UNIVERSIDAD NACIONAL DE COLOMBIA

Sede-Medellin

Facultad de Ciencias

Actuaría de riesgos de contingencias de vida

Estadistica

Trabajo 2
Anualidades con tasas aleatorias

Juan David Garcia Zapata
Juan Diego Espinosa

2024

Nos correspondio el siguiente problema:

Grupo	Anualidad	fondo	modelo de tasas
13	Geometrica	2	NIG

Cuadro 1: NIG = Normal Inversa Gaussiana

Enunciado

Un departamento de estructuración de un Banco diseña una anualidad a n = 10 años, financiada mediante tasas de rendimiento aleatorio de un fondo de inversiones. La anualidad es de tipo Geométrico ó Lineal, con pagos mes ($m = 12$) ó trimestre ($m = 4$) vencido, con valor inicial de C = 2.5 unidades monetarias. Se pacta una tasa de incremento de costo de vida anual, de iq efectiva anual. Desarrolle los siguientes puntos.

a) (30/30) Con los datos de rendimientos diarios del archivo asignado, agreguelos a mes o a trimestrel. Ajuste el modelo LogNor, NIR o VG a las tasas geométricas $\delta_m(K) = \log(1 + i_m(K))$. Reporte: los parámetros estimados, la gráfica de ajuste de log-densidades observadas versus estimadas y la varianza $\sigma^2 = Var(\delta_m(K))$ (para VG use **vgVar()** en VarianceGamma y nigVar para la NIG en **GeneralizedHyperbolic**)

b) (20/20) Calcule valor de la anualidad Geométrica o Lineal, cierta, asumiendo: una tasa efectiva anual de ia, equivalente a la media de las tasas diarias positivas del fondo (ver procedimiento en los ejemplos). Escoja una tasa de inflación efectiva anual $0 < i_q < i_a$, de tal forma que los pagos se incrementan anualmente, q = 1, a esta tasa.

c) (30/30) Para los casos NIG o VG. Calcule la matriz de varianzas-covarianzas de las Λ(j) y simule N = 500 valores de estas variables con el procedimiento de simulación de una cópula Gaussiana con marginales NIG o VG. Reporte algunos histogramas y diagramas de dispersión con dos de estas variables. Genere la exponencial de estas variables y luego genere la suma S de estas exponenciales, que es una LogNIG o LogVG. Muestre el histograma. Para el caso LogNormal con método WKG realice el cálculo de la matriz de varianzas-covarianzas de las Λ(j) y simule N = 500 valores de estas variables con la función rmvnorm de la librería mvtnorm. Genere la exponencial de estas variables y luego genere la suma S de estas exponenciales, que es una suma de LogNormales correlacionadas.

d) (20/20) Encuentre una distribución que ajuste los datos de la variable S, en el caso NIG o VG con la librería gamlss (posiblemente sea una distribución Gamma Inversa). Y en el caso LogNormal con el método WKG. Con estas aproximaciones calcule el cuantil de qS(0.9) como el costo con recargo. Muestre el ajuste superponiendo al histograma de S la densidad estimada. Encuentre el porcentaje de recargo con respecto a la media $q_S(0.9) = (1+)E(X)$.

Se lee la base de datos:

```r
D  <- read.csv("C:\\Users\\Juan\\Documents\\FMRTX.csv",sep = ",")
```

Nota 1: Aunque las ecuaciones definidas en los puntos 1 y 2 no se desarrollan completamente en esos puntos, algunas de sus partes se utilizarán en los demás literales en los que sean necesarias.

Nota 2: A lo largo del trabajo, se irá incluyendo el código correspondiente a cada análisis para proporcionar un mayor contexto. Sin embargo, al final del documento, se presentarán los códigos de cada punto de manera compacta, permitiendo una replicación y verificación más sencilla de los resultados obtenidos.

1. Literal a

:

En el contexto de las contingencias de vida en actuaría, es esencial modelar correctamente los rendimientos de un fondo de inversiones que financia una anualidad a 10 años. Este análisis se centrará en el ajuste de un modelo Normal Inverse Gaussian (NIG) a las tasas geométricas derivadas de los rendimientos agregados, ya sea mensuales o trimestrales. Este ajuste permitirá estimar parámetros clave y evaluar la varianza de los rendimientos, lo cual es crucial para la valoración y predicción del comportamiento de la anualidad.

Para este punto partimos de la siguiente ecuación:

$$F(k) = \exp^{\delta_m(k)} F(k-1) - \frac{1}{m}(1+1_q)^{\frac{1}{q}\lfloor \frac{q_k}{m} \rfloor}) \quad k = 1, 2, ...2, nm$$

Donde:

1. $\delta_m(k)$: Son la tasas geometricas,
2. $F(k)$: Es el saldo de la anualidad en el periodo k.

La solucion dada usando $\Lambda(j) = \sum_{s=1}^{j} \delta_m(s)$ es:

$$F(k) = \exp^{\Lambda(k)}(F(0) - \frac{1}{m}\sum_{j=1}^{k}(1+1_q)^{\frac{1}{q}\lfloor \frac{q(j-1)}{m} \rfloor} \exp^{-\Lambda(j)})$$

Donde:

1. $\Lambda(j)$: Representa el acumulado de las tasas geometricas

Para conseguir los parametros estimados primero definamos los datos que necesitamos para la ecuacion:

cálculo de rendimientos diarios

A partir de la base de datos historicos de precio de cierre de fondo de inversion. a partir A partir de estos precios, se calcularon los rendimientos diarios utilizando la fórmula $\delta_m(K) = \log(\frac{P(k)}{P(k-1)})$ donde P(k) representa el precio de cierre en el dia k. a continuacion, se derivan las tasas geometricas $\delta_m(K) = \log(1 + i_m(K)$ que son fundamentales para la modelización de los rendimientos bajo un modelo geométrico.

```
imk = diff(log(D$Close),1,1)
dmk = log(1+imk)
fechas = as.Date(D$Date[-1],format="%Y-%m-%d")
```

Agregación de rendimientos diarios a rendimientos mensuales

Dado que nuestro objetivo es modelar los rendimientos a una escala temporal mayor, se procedió a agregar los rendimientos diarios para obtener rendimientos mensuales. Para ello, se sumaron los rendimientos diarios dentro de cada mes utilizando herramientas de análisis de series temporales. Las tasas geométricas mensuales se recalcularon aplicando la transformación logarítmica $\delta_m(K) = \log(1 + i_m(K)$ sobre los rendimientos mensuales agregados.

```
#-----------agregar desde dia a mes
library(highfrequency)
library(xts)
imk = xts(x=imk, order.by = fechas)
# Calculo de las tasas geometricas
ts.month <- apply.monthly(imk,FUN=sum)
dmk = log(1+ts.month)

#Resultados de las tasas geometricas
                   [,1]
2023-08-31  0.019012558
2023-09-29 -0.041596027
2023-10-31 -0.026425819
2023-11-30  0.062899253
2023-12-29  0.031370499
2024-01-31 -0.001848430
2024-02-29  0.016364370
2024-03-28  0.021353945
2024-04-30 -0.029308194
2024-05-31  0.023257923
2024-06-28  0.014978536
2024-07-31  0.019015845
2024-08-16  0.006013782
```

Ajuste del modelo NIG y estimación de parámetros

$$NIG(\mu, \delta, \alpha, \beta)$$

El ajuste del modelo **NIG** se realiza utilizando la función **nigFit()** del paquete **GeneralizedHyperbolic**, que estima los parámetros del modelo NIG a partir de las tasas geométricas calculadas previamente. Una vez que se ajusta el modelo, se puede calcular la varianza σ^2 usando la función **nigVar()**.

```
library(GeneralizedHyperbolic)
# Ajustar el modelo NIG a las tasas geométricas
mnig = nigFit(dmk)
nig.est = mnig$param  # Parámetros estimados del modelo NIG
# Calcular la varianza de las tasas geométricas bajo el modelo NIG
sigma2 = nigVar(param = nig.est)
```

1. **Ajuste del modelo**:

```
mu          delta         alpha          beta
0.01880314    0.01305660   21.13256407 -12.80798673
```

2. **Varianza de las tasas geométricas**:

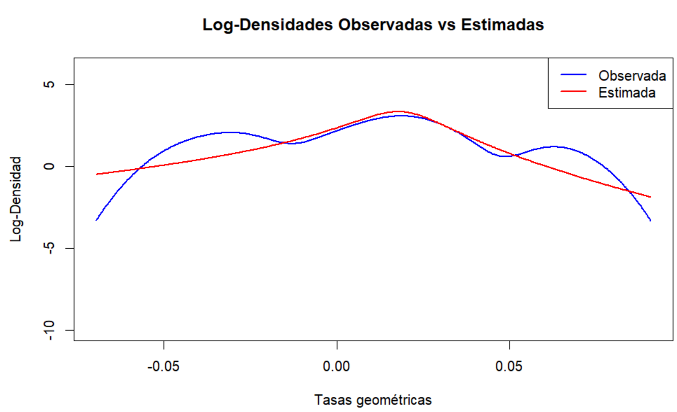
```
0.001227758
```

El ajuste del modelo Normal Inverse Gaussian (NIG) a las tasas geométricas mensuales produjo los siguientes parámetros clave: $\mu = 0.0188$, $\delta = 0.0131$, $\alpha = 21.1326$, y $\beta = -12.8080$. Estos resultados reflejan una ligera asimetría positiva en los rendimientos con una dispersión moderada, y una propensión hacia colas más gruesas a la izquierda, lo que sugiere un riesgo de caídas pronunciadas.

La varianza calculada ($\sigma^2 = 0.0012$) indica que, aunque hay volatilidad, los rendimientos tienden a concentrarse alrededor de la media. En conjunto, el modelo NIG captura adecuadamente las características esenciales de los rendimientos mensuales, proporcionando una base sólida para evaluar el riesgo y el comportamiento de la anualidad.

gráfica de ajuste de log-densidades observadas versus estimadas

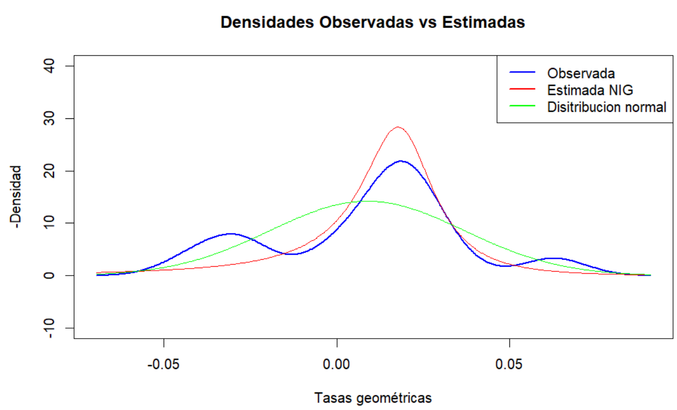
```r
d = density(dmk)
plot(d$x, log(d$y),
     type = "l", col='blue', lwd=2,
     xlab="Tasas geométricas",
     ylab="Log-Densidad",
     main="Log-Densidades Observadas vs Estimadas",ylim = c(-10, 6.0))

lines(d$x, log(dnig(d$x, param = nig.est)),
      lwd=2, col='red')
legend("topright", legend=c("Observada", "Estimada"),
       col=c("blue", "red"), lwd=2, lty=1)
```



El modelo NIG captura adecuadamente la tendencia central de las tasas geométricas, aunque presenta algunas limitaciones en los extremos. Esto sugiere que, si bien el modelo es adecuado para describir la mayoría de los datos, podría haber áreas de mejora, especialmente en la representación de las colas de la distribución. Además, se observó internamente que podría parecer que faltan datos, pero esto se debe a que los datos se agrupan mensualmente, lo que genera esa impresión. Cabe destacar que la gráfica no se asemeja a un ajuste NIG, ya que presenta una curva en la parte superior, más parecida a una distribución normal, en lugar de tener una forma de pico.

Para comparar mejor las observaciones contra los valores estimados, podemos graficarlos sin aplicar la transformación logarítmica. Esto permitirá observar directamente la relación entre los valores observados y los estimados, facilitando la identificación de cualquier discrepancia.

```
d = density(dmk)
plot(d$x,(d$y),
type = "l", col='blue',lwd=2,
ylim = c(-10, 40),
     xlab="Tasas geométricas",
     ylab="-Densidad",
     main="Densidades Observadas vs Estimadas")
lines(d$x,(dnig(d$x, param = nig.est)),
lwd=1,type = 'l', col='red')
lines(d$x,(dnorm(d$x, mean = mean(dmk), sd= sd(dmk))),
lwd=1,type = 'l', col='green')
legend("topright", legend=c("Observada", "Estimada NIG","Disitribucion normal"),
       col=c("blue", "red","green"), lwd=2, lty=1)
```



La comparación entre las densidades observadas y estimadas sugiere que, aunque el modelo NIG captura razonablemente bien la tendencia central de las tasas geométricas, presenta limitaciones en los extremos. Esto es evidente en el hecho de que la densidad estimada por el modelo NIG tiene un pico central más pronunciado y menos dispersión en las colas en comparación con los datos observados. Por otro lado, la distribución normal tampoco parece ser adecuada para modelar estos datos, ya que su suavidad excesiva no refleja bien la estructura observada, especialmente en las áreas más alejadas del centro.

En conclusión, aunque el modelo NIG puede ser útil para describir la mayor parte de los datos, especialmente cerca de la media, podrían explorarse otras alternativas o ajustes adicionales para mejorar la representación de las colas de la distribución y la forma de la curva en la parte superior, ya que esta debería parecerse más a una forma triangular. Esto es importante porque la densidad observada muestra características que el modelo actual no logra capturar completamente

2. Literal b

La ecuacion para el cálculo del valor presente de una serie de pagos en una anualidad geométrica.

$$V(k) = -\frac{1}{m}\sum_{j=1}^{k}(1+1_q)^{\frac{1}{q}\left\lfloor\frac{q(j-1)}{m}\right\rfloor}\exp^{-\Lambda(j)}$$

Donde:

- $V(k)$: Valor presente de los pagos hasta el período k.
- $\frac{1}{m}$: Este factor ajusta el cálculo para la frecuencia de pagos, siendo m el número de pagos por año (por ejemplo, $m = 12$ para pagos mensuales).
- $\sum_{j=1}^{k}$: La suma desde $j = 1$ hasta $j = k$ representa la acumulación de pagos desde el inicio hasta el período k.
- $(1+i_q)^{\frac{1}{q}\left(\frac{q(j-1)}{m}\right)}$: Este término refleja el ajuste de cada pago por la tasa de inflación i_q, que incrementa los pagos a lo largo del tiempo.
- $e^{-\Lambda(j)}$: Este término descuenta los pagos al valor presente, considerando la tasa efectiva anual acumulada. $\Lambda(j)$ es una función acumulativa de las tasas geométricas $\delta_m(k)$.

Filtrado de rendimientos positivos y cálculo de la tasa promedio

Dado que algunos rendimientos diarios fueron negativos, lo que no es ideal para la modelización de anualidades, se procedió a calcular la tasa de rendimiento promedio acumulada. Solo los períodos con rendimientos positivos fueron considerados para operaciones subsecuentes. Esto se realizó graficando el promedio acumulado de los rendimientos y seleccionando aquellos valores donde este promedio era mayor que cero.

```
library(GMCM)
ejex.mes <- seq(min(fechas), max(fechas), by = "month")
ejex.año <- seq(min(fechas), max(fechas), by = "year")
plot(fechas,GMCM:::cummean(imk), xaxt="n", panel.first = grid(),
type='l',ylab='valor unidad fondo FTHRX')
axis.Date(1, at=ejex.mes, format="%m/%y")
axis.Date(1, at=ejex.año, labels = FALSE, tcl = -0.2)
abline(a=0,b=0, col="red")
cimk = GMCM:::cummean(imk)
n1=which(cimk>0)
mimk = mean(imk[n1])
```

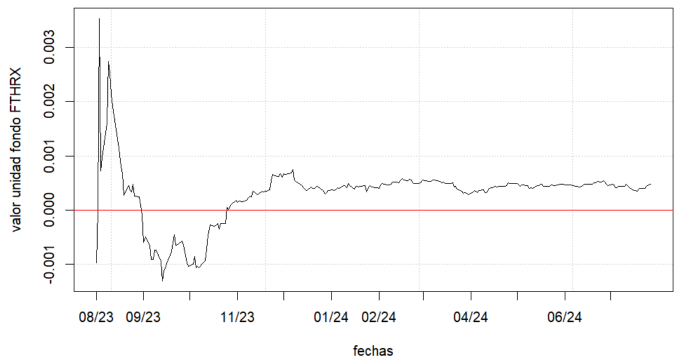


Figura 1: Caption

Este paso asegura que los cálculos posteriores de la anualidad se realicen en condiciones favorables, basadas en rendimientos que son positivos, evitando así problemas asociados a la manipulación de tasas negativas.

Definir parametros

La tasa de rendimiento promedio se estimó a partir de los rendimientos diarios acumulados, seleccionando únicamente aquellos días en los que los rendimientos fueron positivos. Esto permitió calcular una tasa efectiva anual i_a. Adicionalmente, se fijó una tasa de incremento de inflación anual $i_q =?$ (? %). Con un valor inicial de la anualidad unidades monetarias, se generaron los pagos geométricos correspondientes para los 120 meses (10 años) de la anualidad, ajustando los valores según el incremento de inflación.

Para escojer el valor i_q primero calculemos i_a ya que el valor de i_q tiene que ser menor que i_a

```
#ia= tasa de interés efectiva anual
ia = (1+mimk)^(360)-1
[1] 0.2659244
```

Dado que quel valor de la tasa de interes efectiva anual es 0.2659244, entonces para el valor de incremento inflacion anual sera de 0.02(2 %) por que:

1. **Razonamiento**:Esta opción ofrece un buen equilibrio. Refleja una inflación moderada que es común en muchas economías estables, sin sobrepasar el rendimiento anual de la inversión. Con un $i_a = 0.2659244$, una inflación del 2 % anual permitirá que la anualidad siga siendo rentable, con pagos que crecen de manera controlada pero significativa.

2. **Impacto**: Con $i_q = 0.02$ los pagos de la anualidad se ajustarán a un ritmo razonable por la inflación, lo que ayuda a preservar el poder adquisitivo de los pagos en términos reales, sin comprometer la rentabilidad de la inversión.

```
iq = 0.02
C = 2.5

m = 12
n = 10
q = 1
t = seq(1,n*m)
```

Calcular los pagos geométricos

Calcula los pagos geométricos C_j cj que se incrementan anualmente a la tasa de inflación i_q .

```
cj = rep(C,n*m)*(1+iq)^(floor(t*q/m)/q)
```

$C_j =$

```
[1] 2.500000 2.500000 2.500000 2.500000 2.500000 2.500000 2.500000
  [8] 2.500000 2.500000 2.500000 2.500000 2.550000 2.550000 2.550000
 [15] 2.550000 2.550000 2.550000 2.550000 2.550000 2.550000 2.550000
 [22] 2.550000 2.550000 2.601000 2.601000 2.601000 2.601000 2.601000
 [29] 2.601000 2.601000 2.601000 2.601000 2.601000 2.601000 2.601000
 [36] 2.653020 2.653020 2.653020 2.653020 2.653020 2.653020 2.653020
 [43] 2.653020 2.653020 2.653020 2.653020 2.653020 2.706080 2.706080
 [50] 2.706080 2.706080 2.706080 2.706080 2.706080 2.706080 2.706080
 [57] 2.706080 2.706080 2.706080 2.760202 2.760202 2.760202 2.760202
 [64] 2.760202 2.760202 2.760202 2.760202 2.760202 2.760202 2.760202
 [71] 2.760202 2.815406 2.815406 2.815406 2.815406 2.815406 2.815406
 [78] 2.815406 2.815406 2.815406 2.815406 2.815406 2.815406 2.871714
 [85] 2.871714 2.871714 2.871714 2.871714 2.871714 2.871714 2.871714
 [92] 2.871714 2.871714 2.871714 2.871714 2.929148 2.929148 2.929148
 [99] 2.929148 2.929148 2.929148 2.929148 2.929148 2.929148 2.929148
[106] 2.929148 2.929148 2.987731 2.987731 2.987731 2.987731 2.987731
[113] 2.987731 2.987731 2.987731 2.987731 2.987731 2.987731 2.987731
[120] 3.047486
```

Valor presente de la anualidad

```
Gavqmn = function(i,m,q,n,iq){
try(if(iq > i) stop("tasa inflacion invalida"))
try(if(m%%q != 0) stop("m no es divisible por q"))
t = seq(1,n*m,1)
res = (1/m)*sum((1+i)^(-t/m)*(1+iq)^(floor(t*q/m)/q))
return(res)}
(Cp = C*m*Gavqmn(ia,m,q,n,iq))
```

El valor presente es:

```
120.6943
```

El valor presente de la anualidad geométrica calculada, bajo una tasa efectiva anual de $i_a = 0.2659$ y una tasa de inflación anual $i_q = 0.02$, resultó ser **120.6943** unidades monetarias. Este resultado indica que, considerando el crecimiento anual de los pagos debido a la inflación y el descuento de estos pagos a la tasa efectiva anual, la anualidad tiene un valor sustancial en términos actuales.

La elección de una tasa de inflación i_q que es significativamente menor que i_a asegura que los pagos futuros crezcan a un ritmo controlado, mientras que el descuento a la tasa i_a refleja adecuadamente el valor temporal del dinero. Este balance entre el crecimiento debido a la inflación y el descuento por la tasa de rendimiento muestra que la anualidad es una inversión financieramente sólida.

En resumen, el valor presente obtenido refleja una rentabilidad favorable, lo que sugiere que esta anualidad es una opción viable para quienes buscan una inversión que preserve y potencialmente incremente el valor de su capital en términos reales a lo largo del tiempo.

3. Literal c

Primero procederemos a calcular la matriz de varianzas y covarianzas de las $\Lambda(j) = \sum_{s=1}^{j} \delta_m(s)$, donde $\delta_m(s)$ representa transformacion de las tasas geometricas aleatorias $\delta_m(j)$, para hacer esto usaremos el siguiente resultado: $Var(\Lambda(j)) = j\sigma^2$, esto nos va a servir posteriormente para implementar el metodo WKB, el cual sirve para encontrar la distribucion de sumas de variables LogNormal dependientes.

Nota:Se utilizaran los calculos ya hechos en literales anteriores

Matriz de varianzas-covarianza

Por efectos practicos solo mostraremos una pequeña parte de esta ya que la misma tiene una dimension de 120X120.

```
ndim = n*m
R = matrix(0,ndim,ndim)
for(i in 1:ndim){
for(j in 1:ndim){
R[i,j] = ifelse(i <= j, i,j)}}
R = sigma2*R
Rho = cov2cor(R)
Rho[1:10,1:10]
```

```
          [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]      [,8]      [,9]     [,10]
 [1,] 1.0000000 0.7071068 0.5773503 0.5000000 0.4472136 0.4082483 0.3779645 0.3535534 0.3333333 0.3162278
 [2,] 0.7071068 1.0000000 0.8164966 0.7071068 0.6324555 0.5773503 0.5345225 0.5000000 0.4714045 0.4472136
 [3,] 0.5773503 0.8164966 1.0000000 0.8660254 0.7745967 0.7071068 0.6546537 0.6123724 0.5773503 0.5477226
 [4,] 0.5000000 0.7071068 0.8660254 1.0000000 0.8944272 0.8164966 0.7559289 0.7071068 0.6666667 0.6324555
 [5,] 0.4472136 0.6324555 0.7745967 0.8944272 1.0000000 0.9128709 0.8451543 0.7905694 0.7453560 0.7071068
 [6,] 0.4082483 0.5773503 0.7071068 0.8164966 0.9128709 1.0000000 0.9258201 0.8660254 0.8164966 0.7745967
 [7,] 0.3779645 0.5345225 0.6546537 0.7559289 0.8451543 0.9258201 1.0000000 0.9354143 0.8819171 0.8366600
 [8,] 0.3535534 0.5000000 0.6123724 0.7071068 0.7905694 0.8660254 0.9354143 1.0000000 0.9428090 0.8944272
 [9,] 0.3333333 0.4714045 0.5773503 0.6666667 0.7453560 0.8164966 0.8819171 0.9428090 1.0000000 0.9486833
[10,] 0.3162278 0.4472136 0.5477226 0.6324555 0.7071068 0.7745967 0.8366600 0.8944272 0.9486833 1.0000000
```

Simulación

Para simular 500 valores usando una cópula gaussiana con marginal NIG, se seguirían estos pasos:

1. Generar una muestra de 500 valores a partir de la cópula gaussiana utilizando la matriz de correlación Rho.
2. Transformar estas muestras a la distribución marginal NIG.
3. La matriz Rho es crucial porque define las dependencias entre las variables simuladas, asegurando que las correlaciones entre las tasas geométricas aleatorias se mantengan en la simulación.

simulacion de Lambda(j)

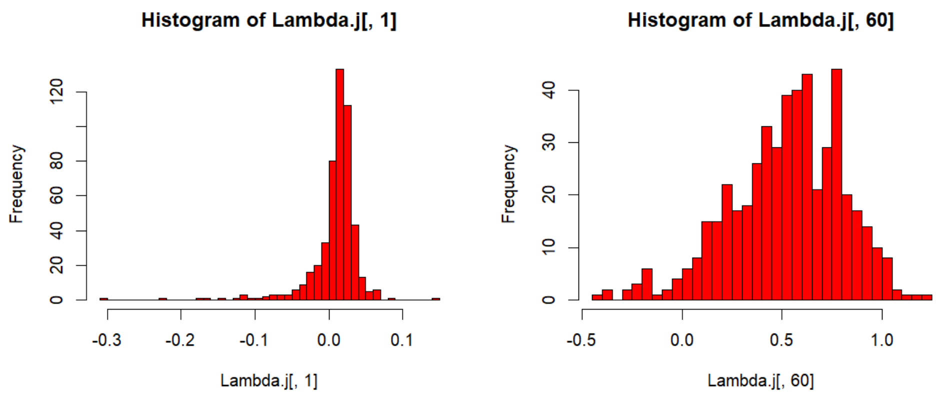
```
set.seed(123)
library(mvtnorm)
sim.GC <- function(n, Rho, qnig){
dat <- rmvnorm(n=n, mean = rep(0,nrow(Rho)), sigma = Rho)
for(j in 1:nrow(Rho)){
dat[,j] <- qnig(pnorm(dat[,j]),param =  c(j,j,1,1)*nig.est)
}
return(dat)
}
Lambda.j = sim.GC(500,Rho,qnig)
Lambda.j[1:8,1:8]
```

```
           [,1]         [,2]         [,3]        [,4]        [,5]        [,6]        [,7]        [,8]
[1,]  0.01283692  0.036597281  0.078580638  0.08208979  0.09861315  0.12995815  0.11716462  0.08775586
[2,]  0.01125381 -0.001398707  0.014765620  0.03884424  0.08132222  0.05530147  0.06962988  0.06977102
[3,]  0.00561964  0.024372714  0.062556155  0.04813682  0.07703894  0.11945754  0.10092598  0.08208719
[4,]  0.01838746  0.047126178  0.065318267  0.09110108  0.10015985  0.09314835  0.10260613  0.09069003
[5,]  0.02346712  0.053787958  0.092247978  0.09999967  0.12519575  0.14328653  0.15573444  0.14827870
[6,]  0.02080549  0.011471086 -0.009653635 -0.07547127 -0.02079697 -0.06650073 -0.11852607 -0.17035611
[7,] -0.01517419 -0.028980363  0.033002535  0.02448751  0.07301148  0.10778304  0.13941679  0.15893533
[8,]  0.02174158  0.099309903  0.079397729  0.10350153  0.13235584  0.14437517  0.16082390  0.21138639
```

Graficos

Histogramas

```
 par(mfrow=c(1,2))
hist(Lambda.j[,1],40)
hist(Lambda.j[,60],40)
```



En los histogramas, se observa la distribución de las variables seleccionadas. Para realizar un análisis más completo, procederemos a calcular el sesgo, lo cual nos permitirá evaluar la asimetría de dichas distribuciones.

La fórmula utilizada para calcular el sesgo es la siguiente:

$$\text{Sesgo} = 3 \times \frac{\text{Media} - \text{Mediana}}{\text{Desviación Estándar}}$$

Este cálculo nos ayudará a comprender mejor la naturaleza de las distribuciones y a identificar posibles desviaciones respecto a la simetría.

sesgo de Lambda.j[,**1**]

```
sesgo_lambda1 = 3*(mean(Lambda.j[,1]) - median(Lambda.j[,1]))/sd(Lambda.j[,1])
[1] -0.4376574
```

$$sesgo\ lambda1 = -0.4376574$$

Esta distribución presenta un sesgo significativo hacia la izquierda, como se observa en la gráfica, lo que indica que tiende a tener valores más altos.

sesgo de Lambda.j[,**60**]

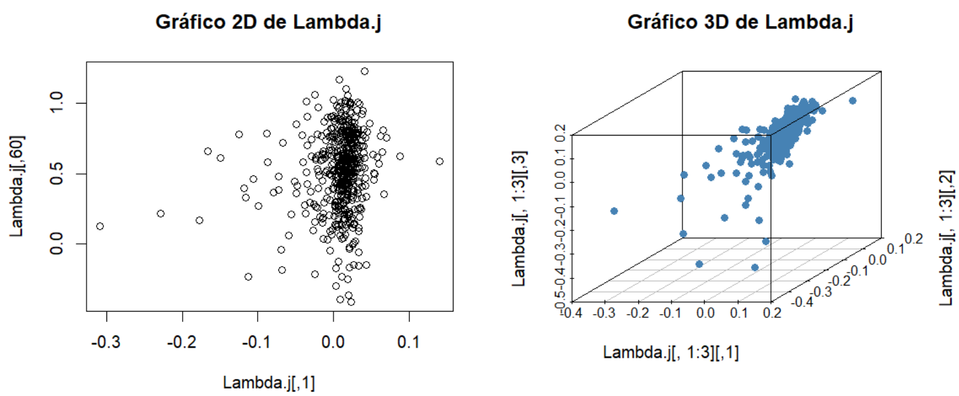
```
sesgo_lambda60 = 3*(mean(Lambda.j[,60]) - median(Lambda.j[,60]))/sd(Lambda.j[,60])
[1] -0.2235571
```

$$sesgo\ lambda60 = -0.2235571$$

Esta distribución presenta un sesgo leve hacia la izquierda, como se observa en la gráfica.

Diagramas

```r
par(mfrow=c(1,2))
plot(Lambda.j[,1], Lambda.j[,60], type='p', main="Gráfico 2D de Lambda.j",
xlab="Lambda.j[,1]", ylab="Lambda.j[,60]")
library(scatterplot3d)
scatterplot3d(Lambda.j[,1:3], pch = 16, color="steelblue", main="Gráfico 3D de Lambda.j")
```

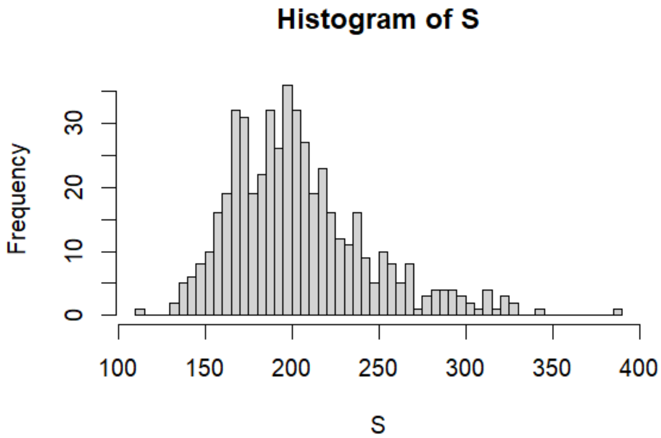


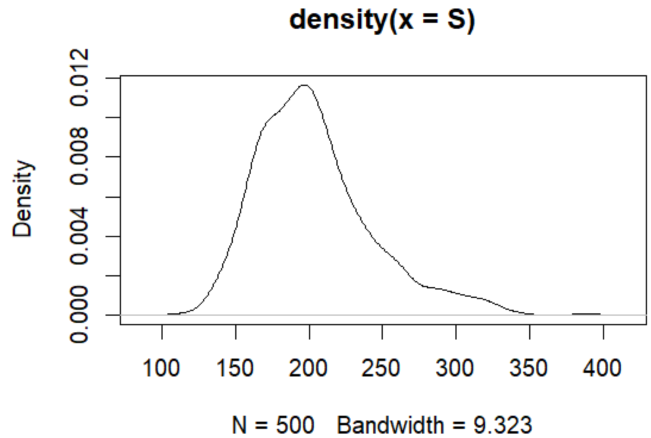
Respecto a la graficas:

1. **2D**: Observamos una correlación entre las dos variables, aunque esta correlación no sigue un patrón común como lineal, exponencial u otro tipo estándar de relación
2. **3D**: Se puede apreciar que las tres variables están altamente correlacionadas. La distribución de los puntos sugiere que existe una fuerte relación entre las variables en el espacio tridimensional, lo que podría indicar una dependencia conjunta significativa entre ellas.

Genere la exponencial de estas variables y luego genere la suma S de estas exponenciales.

```r
par(mfrow=c(1,2))
library(moments)
Y = exp(-Lambda.j)%*%cj
S = apply(Y,1,sum)
hist(S,40)
plot(density(S))
```





El menor dato se encuentra alrededor de 100 y el mayor dato alrededor de 400, lo que define el dominio. También se observa una cola hacia la derecha, la cual se debe a que los valores negativos de los deltas son muy extremos, y cuando estos valores se reemplazan en la fórmula dada en el literal b, específicamente en el exponente $\exp^{-\Lambda}$, el signo negativo en el exponente interactúa con los valores negativos, provocando un aumento significativo en las sumatorias.

4. literal d

¿A qué distribución corresponde esta descripción?

Primero comprobaremos si sigue una distribucion normal y luego no ayudareos de la funcion **gamlss**

Para analizar esto, calcularemos la asimetria:

```
skewness(S)
# Salida
[1] 1.007454
```

ya que es un valor positivo esto nos indica que la distribucion tiene una cola mas larga a la derecha, por lo que la mayoria de valores estan concentrados a la izquierda.

Ahora calcularemos la curtosis:

```
kurtosis(S)
# Salida
[1] 4.348671
```

Como este valor es mayor que 3, nos indica una distribucion leptocurtica, que tiene colas mas pesadas y un pico mas alto que una distribucion normal.

Identificación con la funcion gamlss

```r
library(gamlss)
Sfit1 <- fitDist(y = S, type = "realplus")
```

las mejores 5 con respecto al AIC

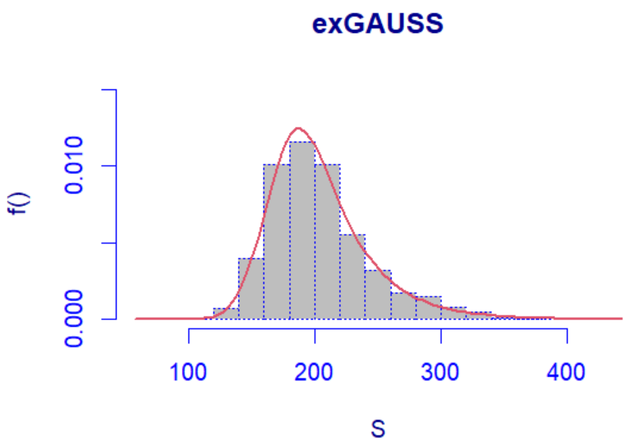
```r
Sfit1$fits[1:5]
```

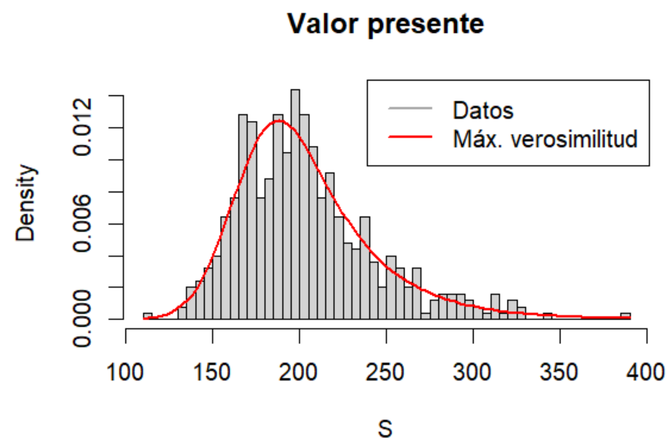
```
exGAUS       GG    BCCGo     BCCG      GB2
5034.665 5036.073 5036.077 5036.077 5037.514
```

De acuerdo con los valores de AIC obtenidos, la distribución **exGAUS** ofrece el mejor ajuste para los datos, seguida de cerca por las distribuciones **GG**, **BCCGo**, y **BCCG**, que presentan un ajuste similar. La distribución **GB2** muestra el ajuste menos favorable entre las cinco evaluadas.

visualizar el ajuste exGAUSS

```r
par(mfrow=c(1,2))
library(gamlss)
library(gamlss.dist)
# Estimación de máxima verosimilitud con exGAUS
histDist(S, family = "exGAUS", main = "exGAUSS", ylim = c(0,0.015))
H_exGAUSS = gamlssML(S, family = exGAUS)
str(H_exGAUSS)
# Coeficientes estimados para la distribución exGAUSS
coef.exGAUSS = c(mu = H_exGAUSS$mu, sigma = H_exGAUSS$sigma, nu = H_exGAUSS$nu)
# Visualización del ajuste exGAUSS
hist(S, 50, probability = TRUE,
     col = 'light gray', main = 'Valor presente')
curve(dexGAUS(x, mu = coef.exGAUSS["mu"], sigma = coef.exGAUSS["sigma"], nu = coef.exGAUSS["nu"],
     log = FALSE),
     col = 'red', lwd = 2, lty = 1, add = TRUE)

# Leyenda del gráfico
legend("topright",
       c("Datos", "Máx. verosimilitud"),
       col = c("darkgray", "red"),
       lwd = c(2, 2), lty = c(1, 1))
```





El gráfico ("Valor presente") indica que el modelo exGAUSS, ajustado mediante máxima verosimilitud, proporciona una buena representación de los datos, especialmente en la región central. La cola derecha también está bien capturada.

Comparacion

Comparemos el valor obtenido mediante ingeniería financiera con el valor obtenido utilizando esta última metodología (la suma S de estas exponenciales). Si utilizamos los promedios de las tasas positivas, obtendremos ciertos valores. Sin embargo, al considerar la variabilidad de las tasas, el valor debería ser mayor, ya que esta metodología capta mejor el riesgo y la incertidumbre asociados con las fluctuaciones de las tasas. (c(Cp,mean(S)))

```
(c(Cp,mean(S)))
```

Como resultado nos da

Método	Valor
Ingeniería Financiera	120.6943
mean(S)	203.8381

Ahora, en lugar de utilizar la media como el costo, lo hemos incrementado ligeramente para crear un margen de seguridad, como un colchón.

$$q_S(0.9) = (1+\theta)E(X)$$

```
q90<- qexGAUS(0.9, mu = coef.exGAUSS[1], sigma = coef.exGAUSS[2], nu = coef.exGAUSS[3])
# Salida
[1] 256.0666
```

El valor 256.0666 es el cuantil del 90 % (costo con recargo). Para calcular el porcentaje de aumento de 203.8381 a 256.0666, calculamos el recargo de la siguiente manera:

$$Recargo = \frac{q_{90} - mean(S)}{mean(S)}$$

```
recargo = (q90 - mean(S)) /  mean(S))
```

$$recargo = 0.2562257 \approx 26\,\%$$

Al incrementar el costo utilizando el cuantil del 90 % en lugar de la media, hemos añadido un margen de seguridad del 26 %. Este enfoque permite cubrir la variabilidad en los datos y ofrece una protección adicional contra posibles desviaciones, asegurando una mayor estabilidad financiera.

5. Codigos compactados

Lectura de base datos:

```
D <- read.csv("C:\\Users\\Juan\\Documents\\FMRTX.csv",sep = ",")
```

5.1. Literal 1

:

```
imk = diff(log(D$Close),1,1)
dmk = log(1+imk)
fechas = as.Date(D$Date[-1],format="%Y-%m-%d")
#-----------agregar desde dia a mes
library(highfrequency)
library(xts)
imk = xts(x=imk, order.by = fechas)
# Calculo de las tasas geometricas
ts.month <- apply.monthly(imk,FUN=sum)
dmk = log(1+ts.month)
library(GeneralizedHyperbolic)
# Ajustar el modelo NIG a las tasas geom´etricas
mnig = nigFit(dmk)
nig.est = mnig$param # Par´ametros estimados del modelo NIG
# Calcular la varianza de las tasas geom´etricas bajo el modelo NIG
sigma2 = nigVar(param = nig.est)
d = density(dmk)
plot(d$x, log(d$y),
type = "l", col='blue', lwd=2,
xlab="Tasas geom´etricas",
ylab="Log-Densidad",
main="Log-Densidades Observadas vs Estimadas",ylim = c(-10, 6.0))
lines(d$x, log(dnig(d$x, param = nig.est)),
lwd=2, col='red')
legend("topright", legend=c("Observada", "Estimada"),
col=c("blue", "red"), lwd=2, lty=1)
d = density(dmk)
plot(d$x,(d$y),
type = "l", col='blue',lwd=2,
```

```
ylim = c(-10, 40),
xlab="Tasas geom´etricas",
ylab="-Densidad",
main="Densidades Observadas vs Estimadas")
lines(d$x,(dnig(d$x, param = nig.est)),
lwd=1,type = 'l', col='red')
lines(d$x,(dnorm(d$x, mean = mean(dmk), sd= sd(dmk))),
lwd=1,type = 'l', col='green')
legend("topright", legend=c("Observada", "Estimada NIG","Disitribucion normal"),
col=c("blue", "red","green"), lwd=2, lty=1)
```

5.2. Literal 2

```
library(GMCM)
ejex.mes <- seq(min(fechas), max(fechas), by = "month")
ejex.a <- seq(min(fechas), max(fechas), by = "year")
plot(fechas,GMCM:::cummean(imk), xaxt="n", panel.first = grid(),
type='l',ylab='valor unidad fondo FTHRX')
axis.Date(1, at=ejex.mes, format="%m/%y")
axis.Date(1, at=ejex.a, labels = FALSE, tcl = -0.2)
abline(a=0,b=0, col="red")
cimk = GMCM:::cummean(imk)
n1=which(cimk>0)
mimk = mean(imk[n1])
ia = (1+mimk)^(360)-1
iq = 0.02
C = 2.5
m = 12
n = 10
q = 1
t = seq(1,n*m)
cj = rep(C,n*m)*(1+iq)^(floor(t*q/m)/q)
Gavqmn = function(i,m,q,n,iq){
try(if(iq > i) stop("tasa inflacion invalida"))
try(if(m%%q != 0) stop("m no es divisible por q"))
t = seq(1,n*m,1)
res = (1/m)*sum((1+i)^(-t/m)*(1+iq)^(floor(t*q/m)/q))
return(res)}
(Cp = C*m*Gavqmn(ia,m,q,n,iq))
```

5.3. Literal c

```
ndim = n*m
R = matrix(0,ndim,ndim)
for(i in 1:ndim){
  for(j in 1:ndim){
```

```
    R[i,j] = ifelse(i <= j, i,j)}}
R = sigma2*R
Rho = cov2cor(R)
Rho[1:10,1:10]
set.seed(123)
library(mvtnorm)
sim.GC <- function(n, Rho, qnig){
  dat <- rmvnorm(n=n, mean = rep(0,nrow(Rho)), sigma = Rho)
  for(j in 1:nrow(Rho)){
    dat[,j] <- qnig(pnorm(dat[,j]),param = c(j,j,1,1)*nig.est)
  }
  return(dat)
}
Lambda.j = sim.GC(500,Rho,qnig)
Lambda.j[1:8,1:8]
par(mfrow=c(1,2))
hist(Lambda.j[,1],40)
hist(Lambda.j[,60],40)
sesgo_lambda1 = 3*(mean(Lambda.j[,1]) - median(Lambda.j[,1]))/sd(Lambda.j[,1])
sesgo_lambda60 = 3*(mean(Lambda.j[,60]) - median(Lambda.j[,60]))/sd(Lambda.j[,60])
par(mfrow=c(1,2))
plot(Lambda.j[,1], Lambda.j[,60], type='p', main="Gr´afico 2D de Lambda.j",
     xlab="Lambda.j[,1]", ylab="Lambda.j[,60]")
library(scatterplot3d)
scatterplot3d(Lambda.j[,1:3], pch = 16, color="steelblue", main="Gráfico 3D de Lambda.j")
par(mfrow=c(1,2))
library(moments)
Y = exp(-Lambda.j)%*%cj
S = apply(Y,1,sum)
hist(S,40)
plot(density(S))
```

5.4. Literal d

```
skewness(S)
kurtosis(S)
library(gamlss)
Sfit1 <- fitDist(y = S, type = "realplus")
Sfit1$fits[1:5]
par(mfrow=c(1,2))
library(gamlss)
library(gamlss.dist)
# Estimaci´on de m´axima verosimilitud con exGAUS
histDist(S, family = "exGAUS", main = "exGAUSS", ylim = c(0,0.015))
H_exGAUSS = gamlssML(S, family = exGAUS)
```

```r
str(H_exGAUSS)
# Coeficientes estimados para la distribuci´on exGAUSS
coef.exGAUSS = c(mu = H_exGAUSS$mu, sigma = H_exGAUSS$sigma, nu = H_exGAUSS$nu)
# Visualizaci´on del ajuste exGAUSS
hist(S, 50, probability = TRUE,
     col = 'light gray', main = 'Valor presente')
curve(dexGAUS(x, mu = coef.exGAUSS["mu"], sigma = coef.exGAUSS["sigma"], nu = coef.exGAUSS["nu"],
              log = FALSE),
      col = 'red', lwd = 2, lty = 1, add = TRUE)
# Leyenda del gr´afico
legend("topright",
       c("Datos", "M´ax. verosimilitud"),
       col = c("darkgray", "red"),
       lwd = c(2, 2), lty = c(1, 1))
(c(Cp,mean(S)))
q90<- qexGAUS(0.9, mu = coef.exGAUSS[1], sigma = coef.exGAUSS[2], nu = coef.exGAUSS[3])
recargo = (q90 - mean(S)) / mean(S)
```

también puedes descargar el código completo a través del enlace accediendo con la cuenta institucional:

https://drive.google.com/file/d/1ZI3ohXGI5n7KhSQG5PYZ2PLJWQPBtXZ0/view?usp=sharing

El siguiente enlace es para descargar la base de datos

https://drive.google.com/file/d/1B5ViUFfTsN97yMg_MWLXjIY2nUKLLC4t/view?usp=sharing