
CONTROL ESTADÍSTICO DE CALIDAD

ENTREGA 1. 2024-II

Michel Mendivenson Barragán Zabala
Departamento de Estadística
Universidad Nacional de Colombia

mbarraganz@unal.edu.co

Juan Sebastián Huertas Pirajan
Departamento de Estadística
Universidad Nacional de Colombia

juhuertasp@unal.edu.co

Diego Andres Paez Molina
Departamento de Estadística
Universidad Nacional de Colombia

dpaezm@unal.edu.co

30 de julio de 2024

Ejercicio 1: Sea $X \sim N(\mu, \sigma)$ una característica de calidad. Mediante simulaciones, establezca el comportamiento del ARL (en control y fuera de él) de las Cartas R y S para observaciones normales con límites 3σ y muestras de tamaño (a) $n = 3$ y (b) $n = 10$ ¿Qué regularidades observa?

Para la solución de este ejercicio se creó la siguiente función para la carta R :

```
RunLengthR_optimized = function(mu = 0, sigma = 1, k= 0, n = 3, m = 1000){  
  #matrixStats: Librería para operaciones rowDiff y rowRange más eficientes (apply es muy lento)  
  # Constantes d3 y d2 para carta R  
  d3 = c(0.853, 0.888, 0.880, 0.864, 0.848, 0.833, 0.820, 0.808, 0.797, 0.787, 0.778, 0.770,  
         0.763, 0.756, 0.750, 0.744, 0.739, 0.734, 0.729, 0.724, 0.72, 0.716, 0.712, 0.708)  
  d2 = c(1.128, 1.693, 2.059, 2.326, 2.534, 2.704, 2.847, 2.970, 3.078, 3.173, 3.258, 3.336,  
         3.407, 3.472, 3.532, 3.588, 3.640, 3.689, 3.735, 3.778, 3.819, 3.858, 3.895, 3.931)  
  
  d3 = d3[n-1]; d2 = d2[n-1] # Selección de constantes específicas dado el tamaño de muestra  
  UCL = (d2 + 3 * d3) * sigma; # Límite superior para carta R  
  LCL = (d2 - 3 * d3) * sigma # Límite inferior para carta R  
  
  RL = c(); # Vector para guardar las longitudes de corrida  
  last_count = 0 # Z longitudes de corrida sin señal al final de la matriz  
  
  while(length(RL) < m){  
    # Se generan m*100 subgrupos racionales, se toma el rango y la diferencia  
    rangos = matrixStats::rowDiffs(matrixStats::colRanges(matrix(rnorm(n*m*100,  
                                                                    mean = mu,  
                                                                    sd = sigma * (1 - k)),  
                                                                    nrow = n, byrow = F)))  
  
    # ¿Qué índices se salieron de control?  
    OutControl = which(rangos < LCL | rangos > UCL)  
    # ¿Cuánto tiempo hay entre estos índices?  
    OutControl = diff(c(0, OutControl))  
    # Se suman al primer RL la cantidad de subgrupos racionales sin señal  
    # del final del ciclo anterior  
    OutControl[1] = OutControl[1] + last_count  
    # Se actualiza a z subgrupos racionales al final de este ciclo sin dar señal
```

```

last_count = length(rangos) - sum(OutControl)

# Se agregan los resultados al final del vector de longitudes de corrida
RL = c(RL, OutControl)}
return(RL[1:m])}

```

Y la siguiente función para la carta S . De esta función, se debe tener en cuenta que se usan las expresiones $LCL = B_5 \times \sigma_0$ y $UCL = B_6 \times \sigma_0$ para los límites de control de la carta en fase II calculando B_5 y B_6 directamente:

```

RunLengthS_optimized = function(mu = 0, sigma = 1, k = 0, n = 3, m = 1000){
  # Constante c4 para corrección de sesgo
  c4 = sqrt(2/(n-1)) * gamma(n/2) / gamma((n-1)/2)
  LCL = sigma * (c4 - 3 * sqrt(1 - c4**2))
  UCL = sigma * (c4 + 3 * sqrt(1 - c4**2))

  RL = c()
  last_count = 0
  while (length(RL) < m){
    desviaciones = matrixStats::colSds(matrix(rnorm(n*m*100,
                                                    mean = mu,
                                                    sd = sigma * (1 - k)),
                                                    nrow = n, byrow = F))

    OutControl = which(desviaciones < LCL | desviaciones > UCL)
    OutControl = diff(c(0, OutControl))
    OutControl[1] = OutControl[1] + last_count
    last_count = length(desviaciones) - sum(OutControl)
    RL = c(RL, OutControl)}
  return(RL[1:m])}

```

Estas funciones muestrean mediante simulación de la distribución de la longitud de corrida de cada carta. Se toman 80 corrimientos a intervalos regulares con k siendo el valor del corrimiento y $k \in [-2, 0]$ (Los resultados se representarán teniendo $\sigma_1 = \sigma_0(1 - k)$), tomando 50000 muestras de la distribución de la longitud de corrida de cada una de las cartas por corrimiento para calcular la media. La distribución real del proceso, sigue siendo la normal estándar. Obtenemos los siguientes resultados:

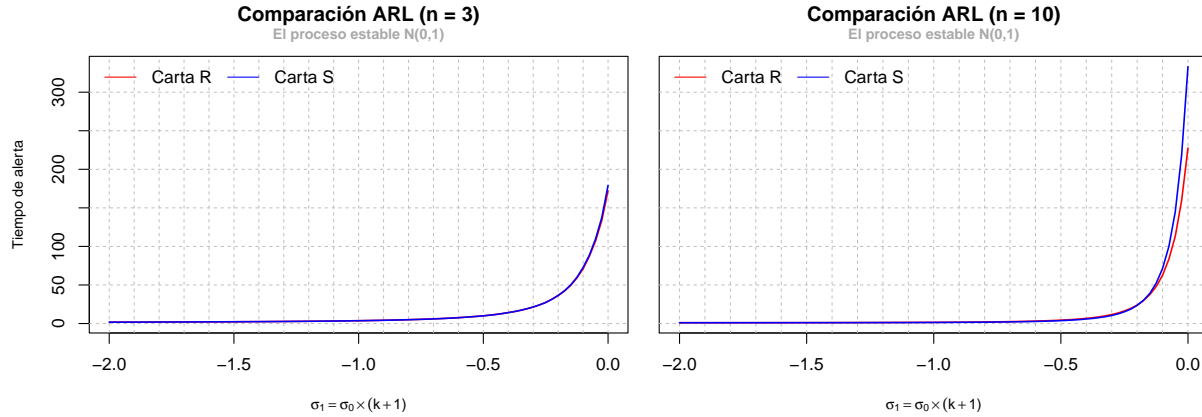


Figura 1: Comparación de ARL para distintos tamaños de muestra (Cartas S y R).

De estas gráficas, es posible concluir que bajo normalidad las cartas R y S parecen tener un comportamiento similar para muestras de tamaño 3. Mientras que para las muestras de tamaño 10 la carta S presenta mejor comportamiento.

	Carta R (n = 3)	Carta R (n = 10)	Carta S (n = 3)	Carta S (n = 10)
$k = -2$	1.79300	1.04396	1.78138	1.02794
$k = -1.8$	1.94530	1.07136	1.93302	1.04692

k = -1.6	2.15982	1.11722	2.13912	1.07720
k = -1.4	2.47186	1.18710	2.44322	1.12928
k = -1.2	2.92476	1.31540	2.92482	1.22208
k = -1	3.66378	1.54954	3.64788	1.40298
k = -0.8	4.98072	2.02762	4.95418	1.77894
k = -0.6	7.60930	3.16820	7.55606	2.70180
k = -0.4	14.10988	6.69224	14.15872	5.88376
k = -0.2	36.08124	23.77134	36.56638	23.60296
k = 0	172.16808	227.63636	179.10096	333.20418

Tabla 1: Comparación de ARL para distintos tamaños de muestra (Cartas S y R).

Ejercicio 2: Sea $X \sim N(\mu, \sigma)$ una característica de calidad. Se sabe que los valores objetivo de los parámetros del proceso son $\mu = \mu_0$ y $\sigma = \sigma_0$. Construir las curvas OC de la carta S^2 con límites de probabilidad. Interpretar los resultados

Recordemos que los límites de probabilidad de una carta de control S^2 se calculan usando la distribución chi-cuadrada ($UCL = \frac{\sigma_0^2}{n-1} \chi_{(1-\alpha/2; n-1)}^2$ y $LCL = \frac{\sigma_0^2}{n-1} \chi_{(\alpha/2; n-1)}^2$), esto debido a que la variable aleatoria definida como $\frac{(n-1)S^2}{\sigma^2}$ tiene distribución χ^2 con $n-1$ grados de libertad. Lo que nos interesa calcular es la probabilidad $P(LCL < S^2 < UCL | \sigma^2 = \sigma_1^2)$:

$$\begin{aligned}
 P\left(LCL < \frac{(n-1)S^2}{\sigma^2} < UCL | \sigma^2 = \sigma_1^2\right) &= P\left(LCL < \frac{(n-1)S^2}{\sigma_1^2} < UCL\right) \\
 &= P\left(\frac{(n-1)LCL}{\sigma_1^2} < S^2 < \frac{(n-1)UCL}{\sigma_1^2}\right) \\
 &= P\left(\chi_{(n-1)}^2 < \frac{(n-1)UCL}{\sigma_1^2}\right) - P\left(\chi_{(n-1)}^2 < \frac{(n-1)LCL}{\sigma_1^2}\right)
 \end{aligned}$$

Y calculando estas probabilidades podremos graficar las curvas OC. Las curvas de operación característica para la carta de control S^2 se generarán usando la siguiente función:

```

OCs2 = function(n = 5, sigma2_0 = 1, corrimiento = 1){
  UCL = sigma2_0/(n-1) * qchisq(0.975, df = n - 1)
  LCL = sigma2_0/(n-1) * qchisq(0.025, df = n - 1)
  beta = pchisq((n-1) * UCL / (sigma2_0 + corrimiento) , df = n - 1)
  - pchisq((n-1) * LCL / (sigma2_0 + corrimiento) , df = n - 1)
  return(beta)
}

```

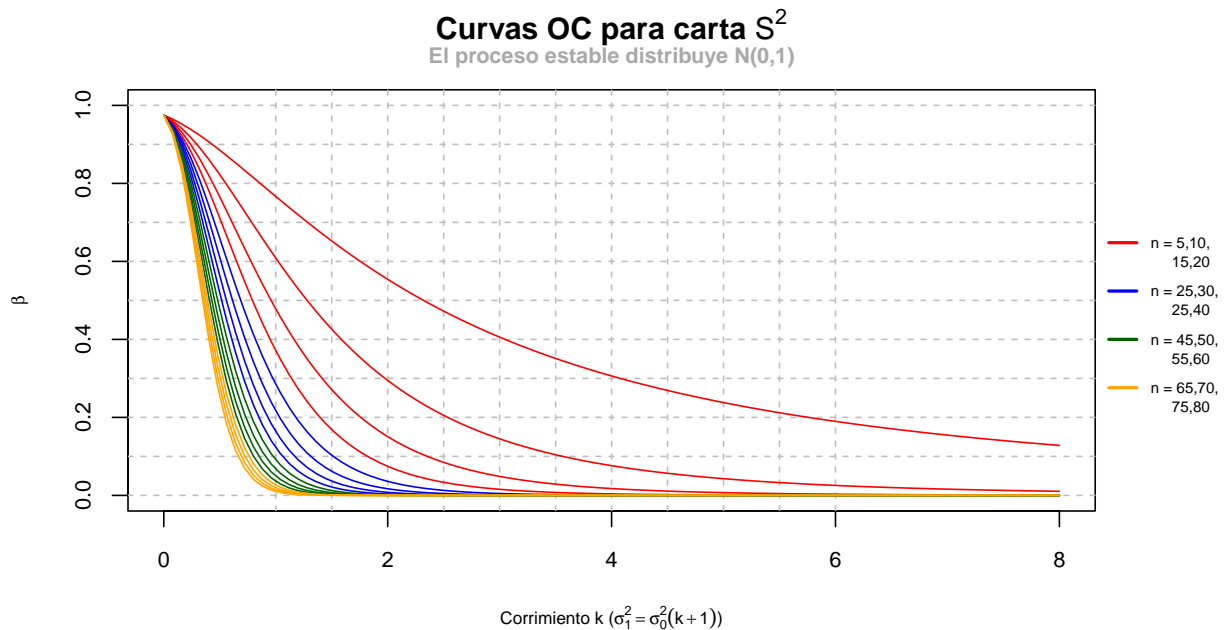


Figura 2: Curvas de operación característica para carta S^2

El incremento en el tamaño de los subgrupos racionales se realiza en múltiplos de 5 para obtener un gráfico más entendible y útil. La carta S^2 demuestra ser más efectiva para muestras más grandes y además de esto, a partir de tamaños de alrededor de 50 no es realmente productivo intentar aumentar n pues la relación costo-beneficio puede llegar a no ser óptima pues la mejora obtenida a cambio del aumento de n no es suficiente. Finalmente, se observa que para corrimientos $\sigma_1^2 = \sigma_0^2 + 1,5$ se obtiene una probabilidad de detectar el cambio en el siguiente subgrupo racional de más del 50% para $n > 15$ por lo que esta carta es más útil para procesos en que sea posible tomar más de 15 muestras por subgrupo.

Ejercicio 3: Sea $X \sim N(\mu_0, \sigma_0)$ una característica de calidad. Construya la carta \bar{X} para el monitoreo de la media del proceso. Genere 10 muestras de tamaño n provenientes de X , de tal modo que la media muestral de ninguna de ellas caiga fuera de los límites de control. A partir del undécimo momento de monitoreo se pide generar muestras del mismo tamaño n provenientes de una distribución normal con media $\mu_1 = \mu_0 + k\sigma_0$ y $\sigma_1 = \sigma_0$ (con $k = 1, 0$) hasta que la carta emita una señal por primera vez. Si se asume que el proceso caracterizado por X es estable y que se desconoce el momento en el cual se produjo el incremento en el nivel medio, ¿en qué muestra ocurrió el cambio en la media del proceso más probablemente?

Tal como se plantea en el ejercicio, se simulan los primeros diez subgrupos muestrales de una distribución normal estándar. Es decir, nuestro valor en control de la media debería ser uno. Además, el tamaño de los subgrupos racionales será $n = 5$:

```
# Parámetros
mu = 0; sigma = 1; k = 1; n = 5

# Límites de la carta de control teóricos
LCL = mu - 3 * (sigma/sqrt(n)); UCL = mu + 3 * (sigma/sqrt(n))

set.seed(13)                                # Para replicabilidad
muestras = c()                               # Objeto para guardar las medidas de control
for (i in 1:10){                             # Generación de muestras bajo control
  muestras[i] = mean(rnorm(n = n, mean = mu, sd = sigma))
}

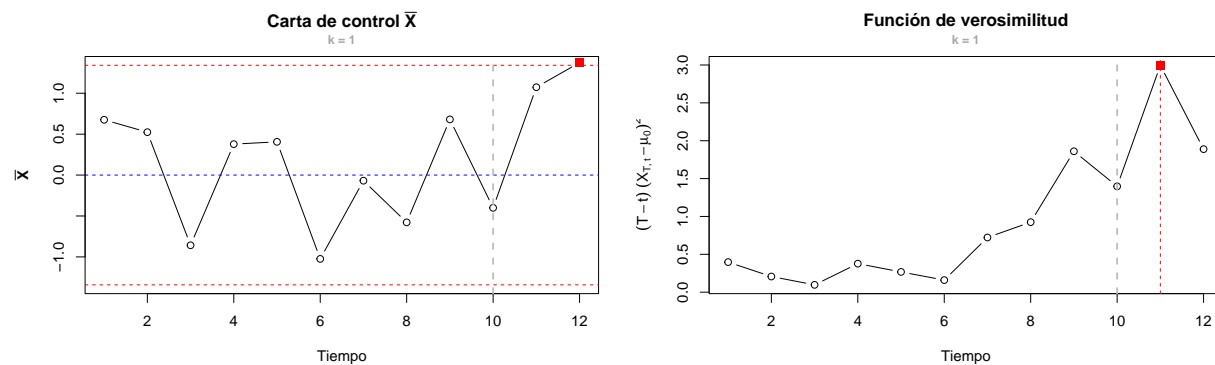
mu = mu + k * sigma                          # Cambiando la media del proceso
media = muestras[10]
```

```
while (media < UCL & media > LCL){
  media = mean(rnorm(n = n, mean = mu, sd = sigma))
  muestras = c(muestras, media)
}
```

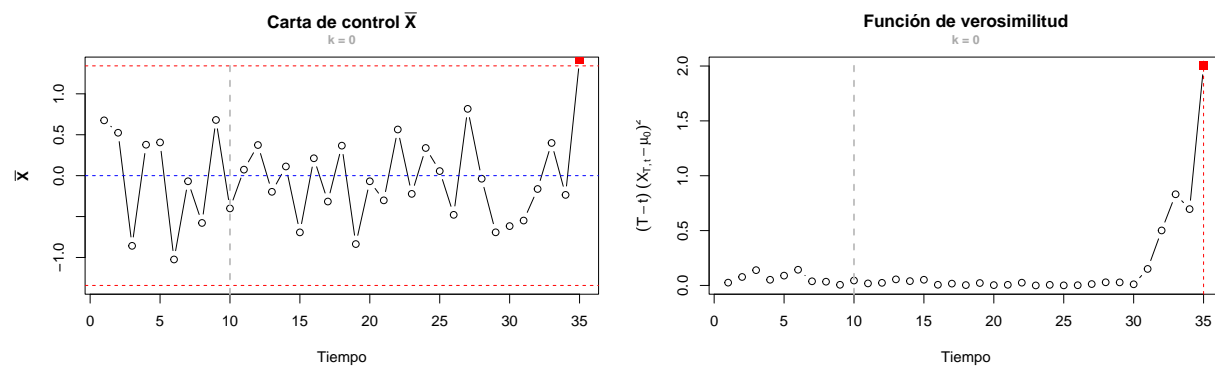
Ahora bien, para calcular el momento t en que más probablemente se produjo el cambio en la medida de control es necesario revisar para que $t < T$ (Teniendo a T como el valor en que la carta emitió una señal) se maximiza la función $\arg\max_{0 \leq t < T} [(T-t)(\bar{X}_{T,t} - \mu_0)^2]$:

```
tiempo = 0:(length(muestras)-1)      # Todos los tiempos antes de la señal
señal = length(muestras)              # Aquí la carta dio una seña

valores = c()
for (t in tiempo){
  Tmt = señal - t
  mediaSeñal = 1/Tmt * sum(muestras[(t+1):señal])
  valores = c(valores, Tmt*(mediaSeñal)^2) # Aquí nos evitamos -mu_0 porque mu_0 = 0
}
```

Figura 3: Ejercicio para $k = 1$

Notemos que en este caso, para una desviación de una desviación estándar de la media este método parece identificar de forma correcta el tiempo en el que se presentó la desviación de la media ya que en la gráfica de la derecha vemos que la función de verosimilitud se maximiza para $t = 11$ por lo que el último subgrupo racional en teoría (Y en la práctica también) fue el subgrupo 10. Ahora veamos cómo se comporta este método para la carta cuando no se han presentado desviaciones de la media (Básicamente reutilizamos el código anterior):

Figura 4: Ejercicio para $k = 0$

En este caso, la gráfica de la derecha nos dice que el momento en que parece más probable que hubiera ocurrido una posible desviación de la media es el momento en que se detectó la señal por lo que se puede justificar a esta como una herramienta útil no solamente para identificar el momento en que se presenta la desviación de la media sino también para obtener pistas de la índole de la desviación de la carta (Aleatoriedad pura o realmente un cambio en el proceso).

Ejercicio 4: Sea $X \sim N(\mu_0, \sigma_0)$ una característica de calidad. Se pide:

- Mediante simulaciones, establezca el comportamiento del ARL de la Carta \bar{X} con límites tres sigma para observaciones normales.

Utilizaremos una aproximación al problema similar a la presentada en el punto 1 creando una función para poder generar todos los corrimientos necesarios. En este caso, se crea la siguiente función:

```
RunLengthXMod_Optimized = function(mu1 = 0, sigma1 = 1,          # Parámetros estimados del proceso
                                   mu = 0, sigma = 1,           # Parámetros reales del proceso
                                   L = 3, Corrimiento = 0,
                                   n = 3, m = 1000){

  # Note que si los parámetros estimados del proceso son iguales a los parámetros
  # reales del proceso podremos calcular los ARLs teóricos de la carta.

  # Límites de control
  UCL = mu1 + L * sigma1/sqrt(n)          # Tenga en cuenta que sigma1 = Rango/d2 por lo que
  LCL = mu1 - L * sigma1/sqrt(n)          # L/sqrt(n) * sigma1 = A_2 * R. Además, si quisieramos usar
  # otro tamaño de muestra necesitaríamos cambiar a d2.

  medias = c()                            # Medias de los subgrupos racionales
  RL = c()                                # Muestras de longitud de corrida
  last_count = 0                          # Place holder ¿Cuántas muestras quedaron al final de las medias sin s
  # Corrimiento
  muCorr = mu + Corrimiento * sigma        # Media del proceso con corrimientos

  while (length(RL) < m){
    # Se calculan n*m*100 medias y se revisa qué medias quedaron fuera de control
    medias = colMeans(matrix(rnorm(n*m, mean = muCorr, sigma),
                              nrow = n, byrow = F))

    # Se consiguen los índices de las medias que quedaron fuera de control (i.e Tiempo)
    OutControl = which(medias < LCL | medias > UCL)

    # Auxiliar para arreglar el problema
    # aux = length(medias) - OutControl[length(OutControl)]

    # Se calcula el tiempo real (Como reiniciar el proceso después de una señal)
    OutControl = diff(c(0, OutControl))

    # Al final del vector van a quedar z cantidad de muestras que no dieron señal
    # y eso es lo que lleva la variable de last_count
    OutControl[1] = OutControl[1] + last_count

    # El último error estaba en esta línea, la asignación se estaba haciendo de forma errónea
    # last_count = length(medias) - OutControl[length(OutControl)]

    last_count = length(medias) - sum(OutControl)

    RL = c(RL, OutControl[1:length(OutControl)])
  }
  return(RL[1:m])
}
```

NOTA: La función anterior era muy rápida, pero ocupaba demasiada memoria y es posible quedarse sin memoria, en teoría esta función debería ocupar menos memoria.

Note que esta función cuenta con una diferencia importante para la metodología que veníamos utilizando pues declara dos sets de parámetros como argumentos. Cuando $\mu_1 = \mu$ y $\sigma_1 = \sigma$ estaremos hablando del rendimiento teórico de la

carta. En caso contrario, estaremos hablando del rendimiento de la carta haciendo estimaciones de los parámetros (μ_1 y σ_1 son las estimaciones de la media y la desviación estándar respectivamente).

Del mismo modo que anteriormente, se escogen 81 corrimiento $k \in [0, 3]$ para generar los ARL de la carta extrayendo para cada corrimiento 50000 muestras de la longitud de corrida. Obteniendo los siguientes resultados:

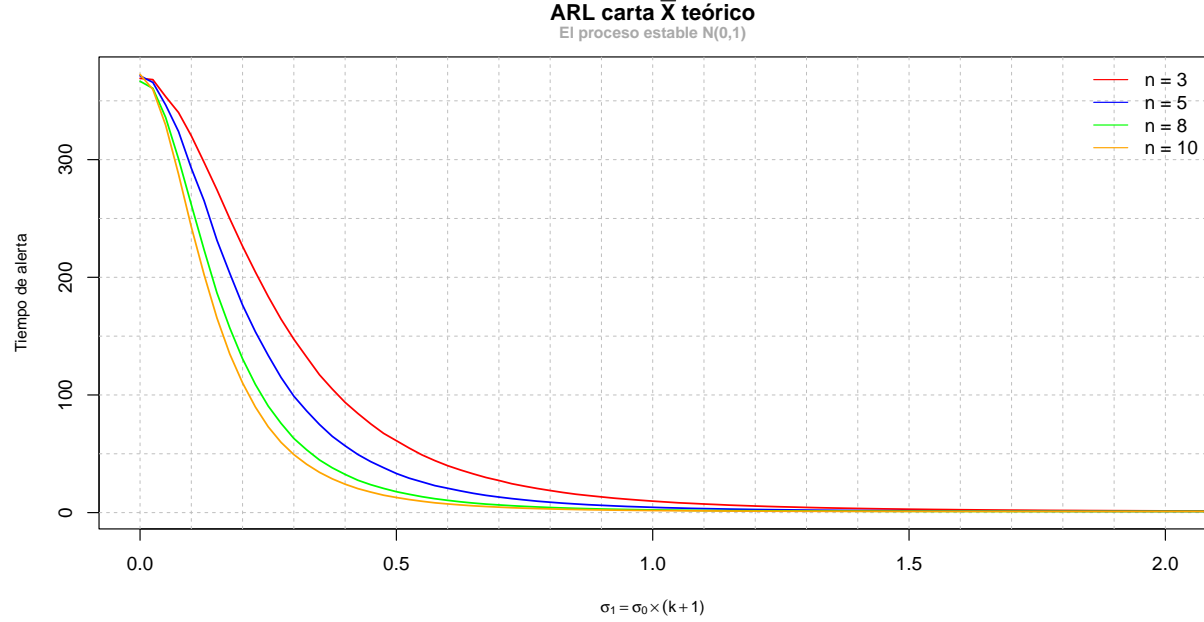


Figura 5: Comparación de ARL para distintos tamaños de muestra (Cartas S y R).

Cuyos resultados numéricos se encuentran consignados en la siguiente tabla:

	n = 3	n = 5	n = 8	n = 10
k = 0	369.11980	371.20572	366.62912	372.43810
k = 0.2	226.35376	176.42326	130.72720	110.20348
k = 0.4	93.70246	56.76764	32.48074	24.15320
k = 0.6	39.88466	20.70026	10.44124	7.38096
k = 0.8	18.77208	8.84864	4.34966	3.15418
k = 1	9.75906	4.50346	2.30920	1.77812
k = 1.2	5.59698	2.65848	1.53628	1.27364
k = 1.4	3.53432	1.81828	1.20776	1.08362
k = 1.6	2.44632	1.38442	1.07040	1.02048
k = 1.8	1.82930	1.18154	1.01862	1.00370
k = 2	1.46686	1.07728	1.00378	1.00054
k = 2.2	1.26256	1.02722	1.00068	1.00008
k = 2.4	1.13912	1.00982	1.00006	1.00000
k = 2.6	1.07032	1.00196	1.00000	1.00000
k = 2.8	1.03518	1.00044	1.00000	1.00000
k = 3	1.01452	1.00004	1.00000	1.00000

Tabla 2: ARL para distintos tamaños de muestra (Carta X).

- b. Genere 20 subgrupos racionales de tamaño $n = 3$ provenientes de X . Asíumase que el proceso es estable en cuanto a dispersión y con los subgrupos iniciales, construya la carta \bar{X} como es habitual hasta verificar la estabilidad del proceso. Establezca el comportamiento del ARL para la carta que se obtiene del análisis de Fase I realizado.

- c. Repetir lo indicado en el literal (b) con 50 subgrupos racionales de tamaño $n = 3$. Comente los resultados.

Ejercicio 5: Calcular el ARL de la carta \bar{X} mediante cadenas de Markov. Diseñar la carta con límites de control ubicados a tres desviaciones estándar de la media y dividiendo la región de control estadístico en franjas de ancho igual a una desviación estándar.