Universidad Politécnica de Cartagena

industriales
etsii UPCT

CONSTRUCCIÓN DE UN ENTORNO AMIGABLE PARA EL ESTUDIO DE LA DEPENDENCIA EN SERIES TEMPORALES

Titulación: Ingeniero Industrial
Intensificación: Sistemas Eléctricos
Alumno/a: Salvador Vera Nieto
Director/a/s: José Salvador Cánovas Peña
Antonio Guillamón Frutos

Cartagena, 6 de Febrero de 2012

A mis padres

PREFACIO

Las series temporales son un campo de conocimiento presente en muchos y diversos ámbitos sociales, políticos y científicos. La importancia del estudio de éstas radica en la gran variedad de situaciones que requieren el análisis de una serie temporal. Hay que recurrir al estudio de series temporales para responder a cuestiones tan interesantes y actuales como: ¿es cierto el fenómeno del calentamiento global del planeta?, ¿cuándo saldrá Europa de la actual crisis económica?

El presente trabajo se centra en tres métodos para el estudio de la dependencia en series temporales; que son, concretamente, los siguientes:

- Test de M. Matilla y M. Ruíz.
- Test de J.S. Cánovas y A. Guillamón.
- Test BDS.

El objetivo de nuestro estudio es doble. Por un lado se pretende llevar a cabo una comparación de los tres métodos anteriormente citados, para dilucidar cuál es el más adecuado para el estudio de la dependencia en series temporales. Por otro lado se quiere programar una aplicación que sea capaz de aplicar estos métodos a una serie temporal dada; y así poder obtener y comparar los resultados proporcionados por estos.

En el capítulo 1 se presenta el concepto de serie temporal, se destaca la omnipresencia de las series temporales en diferentes ramas de conocimiento, y se analizan los tipos de series temporales atendiendo a distintos criterios de clasificación.

En el capítulo 2 entramos a describir los tres métodos que son objeto de nuestro trabajo; incluyendo ejemplos clarificadores sobre la aplicación de dichos métodos a series temporales concretas.

En el capítulo 3 se dan las características más importantes de los dos entornos utilizados: Mathematica y Matlab. Además se incluye la descripción de los algoritmos utilizados para la aplicación de cada test.

En el capítulo 4 se ofrecen los resultados del análisis comparativo llevado a cabo entre los métodos en estudio. Se obtienen también algunas conclusiones de cierto interés.

En el capítulo 5 se presenta una aplicación diseñada para el estudio de la dependencia de una serie temporal a través de los tres métodos de análisis que nos ocupan.

ÍNDICE

CAPÍTULO 1

GENERALIDADES SOBRE SERIES TEMPORALES

En este primer capítulo se presenta la definición de serie temporal, y se citan varios ejemplos de series temporales en distintos contextos, como el científico, el social o el político.

Además se describen los tipos de series temporales atendiendo a distintos criterios de clasificación.

Explicamos también cuáles son los objetivos y las finalidades de estudio de las series temporales.

1. DEFINICIÓN DE SERIE TEMPORAL. EJEMPLOS.

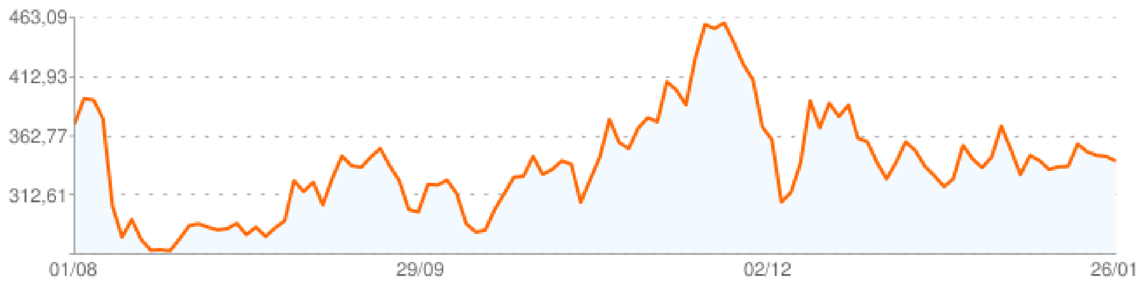
Una serie temporal o cronológica es la realización de n variables aleatorias ordenadas en el tiempo. Cada dato representa una observación de tamaño uno de la variable X_t, que representa las posibles realizaciones en el estado t. El análisis de series temporales comprende métodos que ayudan a interpretar este tipo de datos, extrayendo información representativa, tanto referente a los orígenes o relaciones subyacentes, como a la posibilidad de extrapolar y predecir su comportamiento futuro.

Es habitual encontrar series temporales en contextos económicos, empresariales, físicos, médicos, matemáticos o biológicos. Resulta difícil imaginar una rama de las ciencias en la que no aparezcan datos que puedan ser considerados como series temporales. Enunciamos a continuación algunos ejemplos agrupados en distintos campos:

Economía y marketing

- Evolución del IPC en España.

- Evolución de la prima de riesgo española.



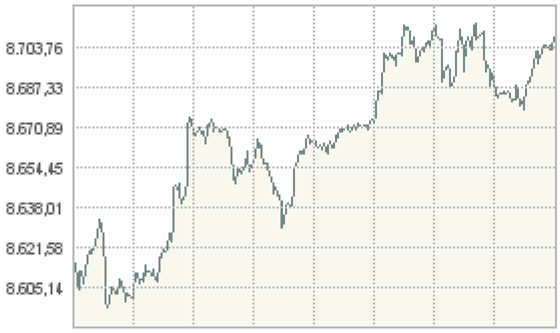
Gráfica. Evolución de la prima de riesgo española desde el 1 de agosto de 2012 hasta el 26 de enero de 2012. (Fuente: www.eleconomista.es)

- Precio de la vivienda en un cierto período de tiempo.

- Evolución del índice del precio de un bien o servicio.

- Beneficios netos anuales de una determinada empresa.

- Series de cotizaciones de bolsa, como por ejemplo el índice Ibex 35.



Gráfica. Evolución del índice Ibex 35 durante el 26 de enero de 2012. (Fuente: www.elpais.com)

- Series de exportaciones o importaciones.

- Existencias en un almacén.

- Gastos en publicidad de un sector.

Demografía y sociedad

- Número de habitantes en cierto país.

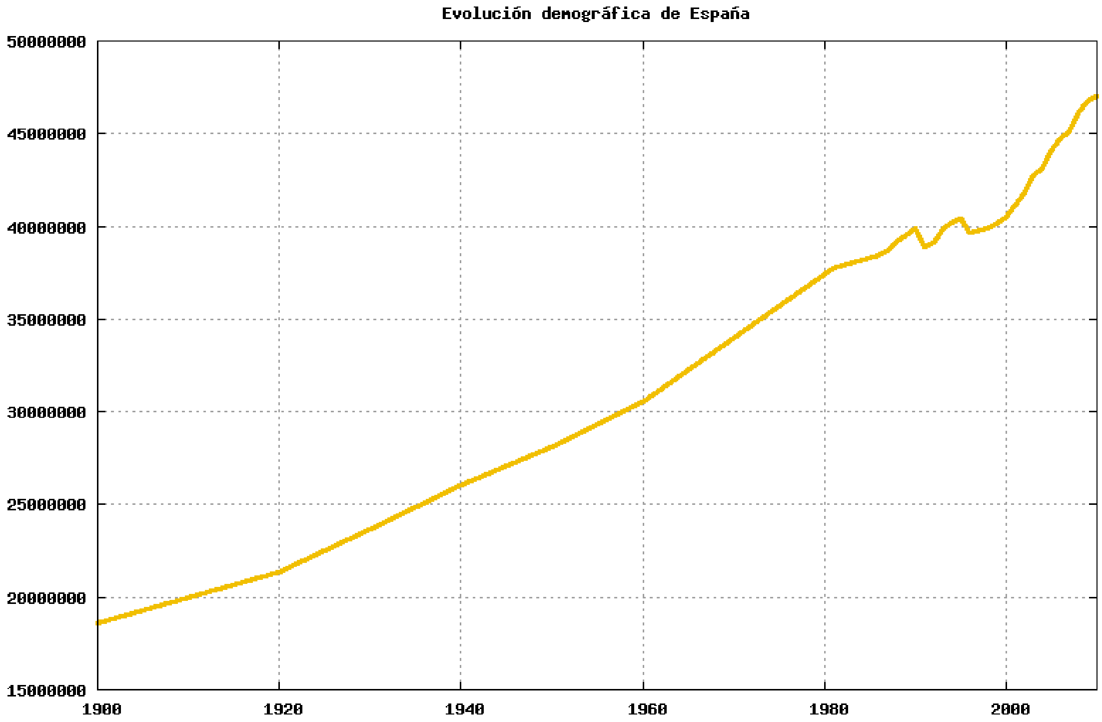
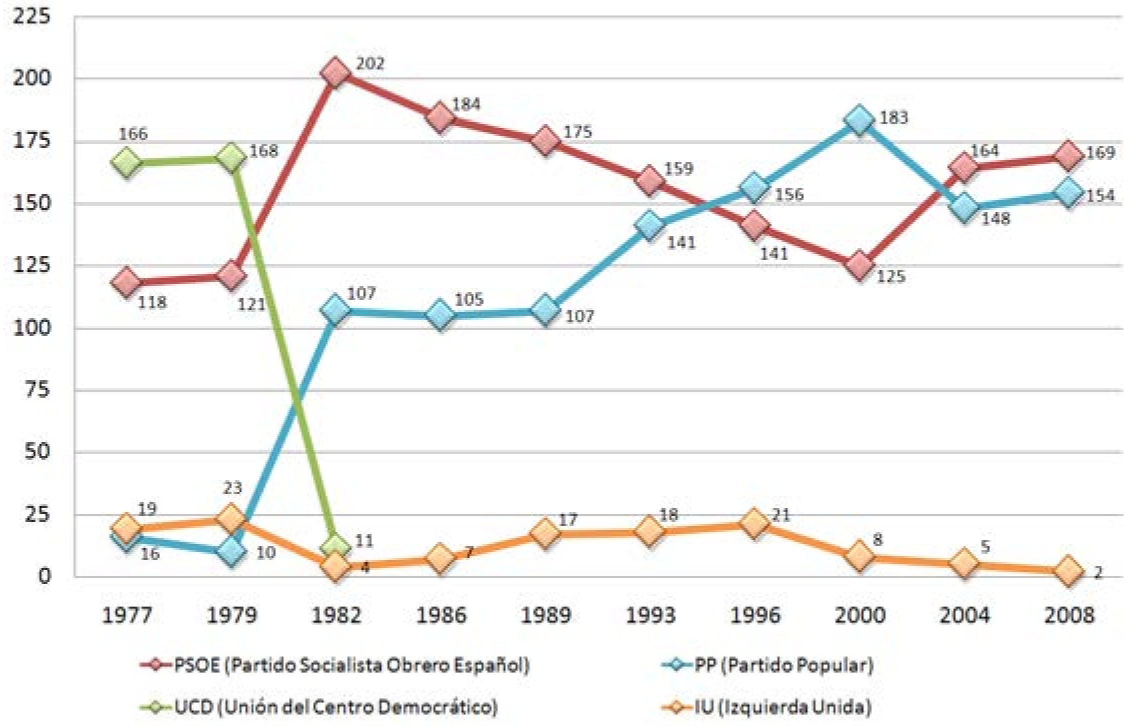


Gráfico. Evolución demográfica de España desde el año 1900 (Fuente: es.wikipedia.org)

- Tasa de mortalidad infantil por año.

- Número de matrimonios por año.

- Evolución del número de escaños de un determinado partido político.



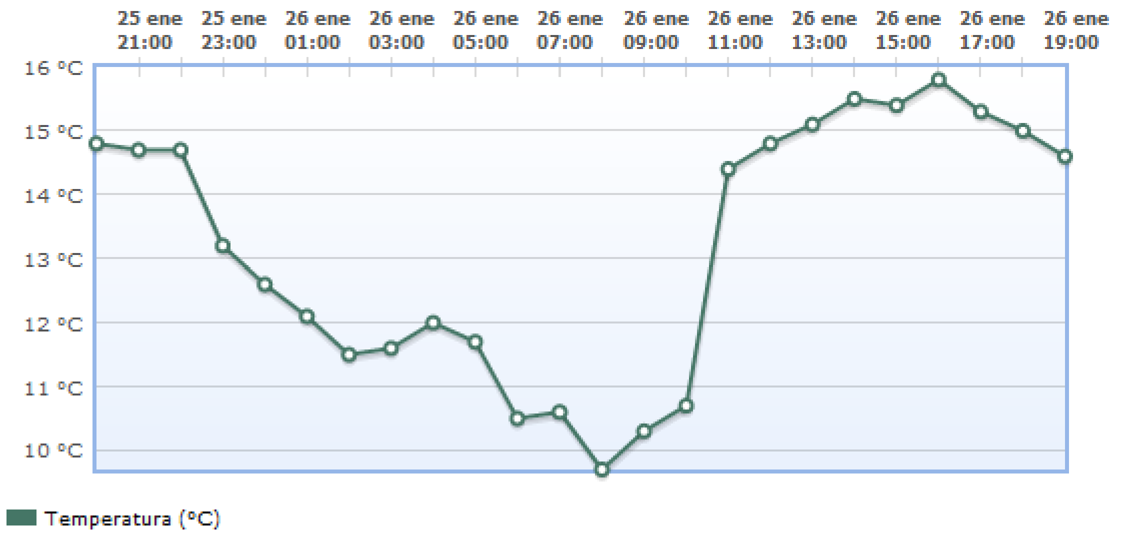
Gráfica. Evolución del número de escaños obtenidos por diversos partidos políticos en el período 1977-2008. (Fuente: es.wikipedia.org)

Medioambiente

- Evolución de los niveles de sustancias contaminantes en una determinada ciudad.

- Lluvia recogida diariamente en cierta región.

- Evolución de la temperatura de una localidad.



Gráfica. Variación de la temperatura en el municipio de Cartagena desde las 21.00 horas del 25 de enero de 2012 hasta las 19.00 hora del 26 de enero de 2012. (Fuente: www.aemet.es)

- Medición del contenido de residuos tóxicos en un río.

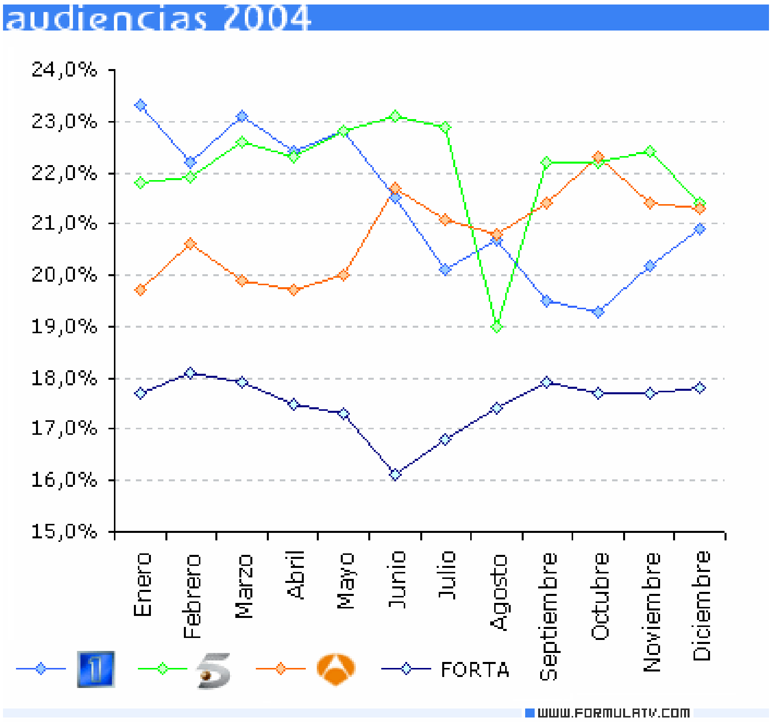
- Velocidad del viento en una central eólica.

- Caudal de un río.

Otros:

- Demanda de energía eléctrica (esta serie suele obtenerse con periodicidad horaria).

- Evolución de la audiencia de una determinada cadena de televisión.



Gráfica. Evolución de las audiencias de distintas cadenas de televisión en el año 2004. (Fuente: www.formulatv.com)

Una serie temporal puede interpretarse como una distribución estadística bidimensional en la que las variables son el tiempo y la característica en estudio:

Tiempo	**Valor**
t_1	$Y(t_1)$
t_2	$Y(t_2)$
…	…

Al iniciar el estudio de una determinada serie temporal se suele proceder a la representación gráfica de la misma. Esto nos permite detectar las características más importantes de la evolución de la serie, y se eliminan los problemas de visualización que conlleva una tabla con un gran volumen de datos. En este aspecto hay que prestar una atención especial a la elección de las escalas en los ejes en los que se realiza la gráfica; una elección errónea puede conducir a asumir unas consideraciones iniciales con cierta distorsión.

Si para cada instante de tiempo se dispone de varias observaciones, los valores se pueden presentar en una tabla como la siguiente:

	Observaciones				
Tiempo	**1**	**2**	**…**	**…**	**s**
$\mathbf{t_1}$	t_{11}	t_{12}	…	…	t_{1s}
$\mathbf{t_2}$	t_{21}	t_{22}	…	…	t_{2s}
…	…	…	…	…	…
$\mathbf{t_n}$	t_{n1}	t_{n2}	…	…	t_{ns}

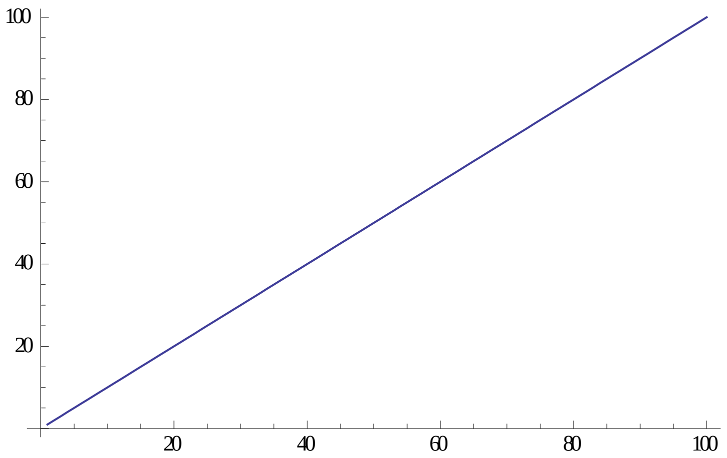
Un caso típico en el que se recurriría a una tabla como la anterior sería el de un conjunto de datos que estuvieran tomados en los diferentes trimestres de cada año. En este caso se tendría un año distinto en cada fila, y los cuatrimestres de cada año se agruparían en las columnas de la tabla.

2. CLASIFICACIÓN DE LAS SERIES TEMPORALES.

Vamos a comentar los distintos tipos de series temporales que nos podemos encontrar. Pero hay que tener en cuenta que éstas no se clasifican, únicamente, según un único criterio. Hay diferentes clasificaciones de las series temporales atendiendo a distintos aspectos.

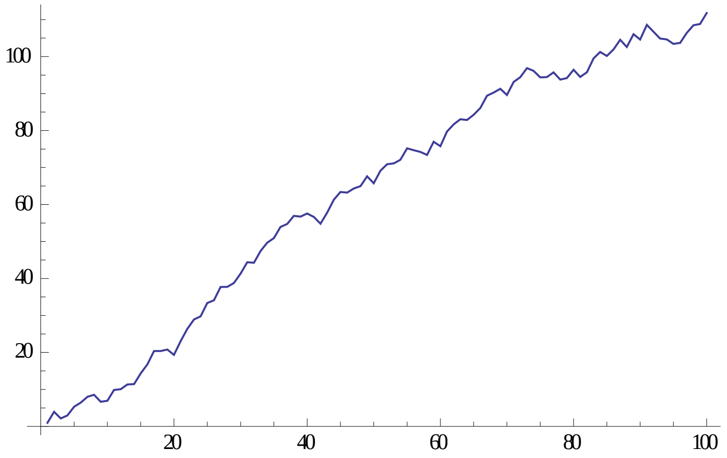
En función de lo predecible que sean los valores futuros se tiene la siguiente clasificación:

· Series temporales deterministas: son aquellas en las que se puede predecir con total exactitud los valores que las componen.



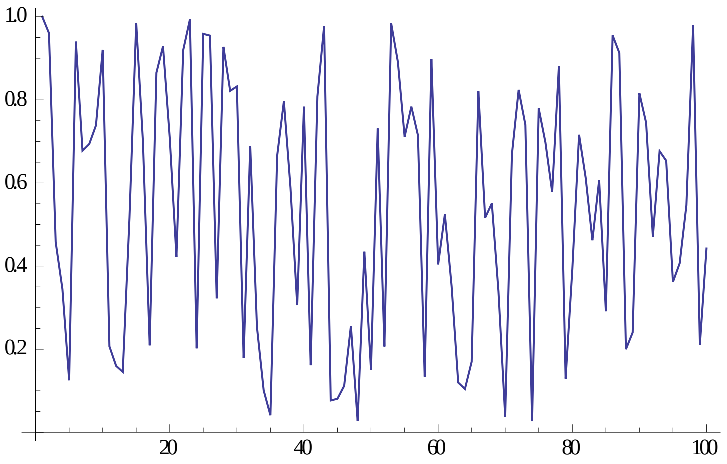
Gráfica. Serie temporal determinista generada con MATHEMATICA.

· Series temporales estocásticas: son aquellas en las que los valores futuros no se pueden determinar de forma exacta, o no son nada predecibles. En el caso de que se pueda determinar de manera parcial su comportamiento, se dice que existen patrones de regularidad en el tiempo; lo que hace posible el modelado, y en su caso, la predicción. Por lo tanto, si podemos encontrar patrones de regularidad en diferentes secciones de la serie temporal, podremos también describirla mediante modelos basados en distribuciones de probabilidad.



Gráfica. Serie temporal estocástica generada con MATHEMATICA.

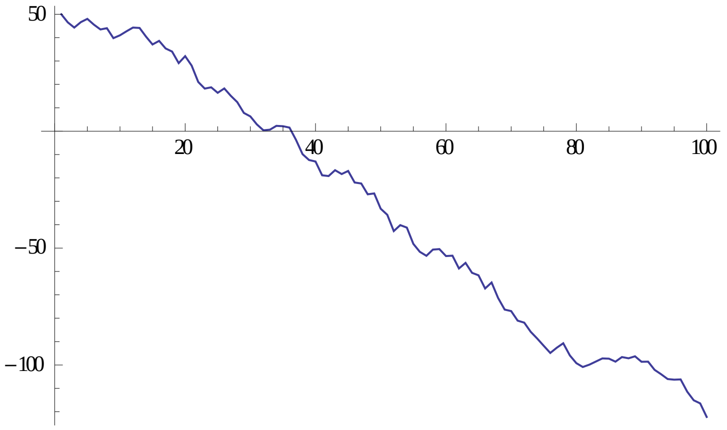
· Series temporales idénticamente distribuidas: son aquellas en las que todas las observaciones provienen de un mismo modelo probabilístico, aunque exista independencia en cuanto al instante en el que aparecen.



Gráfica. Serie temporal idénticamente distribuida generada con MATHEMATICA.

Según las variaciones que puede experimentar la media de los valores de la serie, se tiene el siguiente tipo de series:

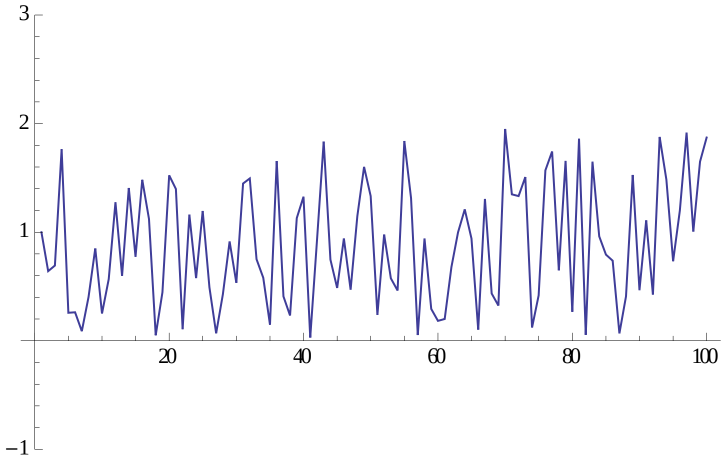
· Series temporales con tendencia: son aquellas en las que se aprecia un patrón de comportamiento.



Gráfica. Serie temporal con tendencia descendente generada con MATHEMATICA.

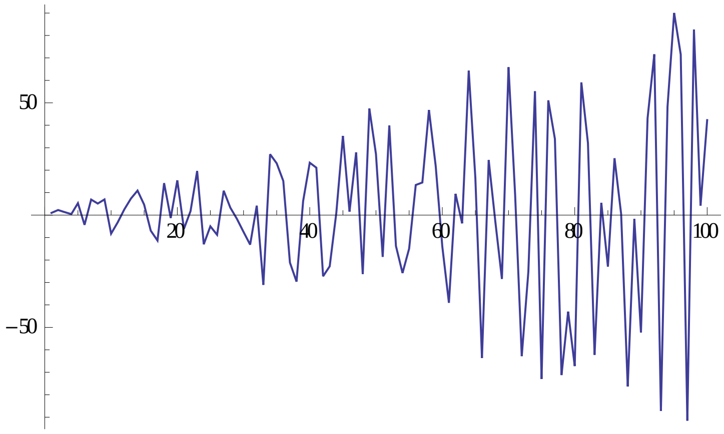
Atendiendo a los cambios que puede sufrir la variabilidad, se puede tener:

· Series temporales homocedásticas: son aquellas en las que la dispersión asociada a las variables se mantiene constante.



Gráfica. Serie temporal homocedástica generada con MATHEMATICA.

· Series temporales heterocedásticas: son aquellas para las que la variabilidad no permanece constante a lo largo del tiempo.



Gráfica. Serie temporal heterocedástica generada con MATHEMATICA.

Considerando de forma conjunta los cambios que pueden experimentar la media y la variabilidad, podemos tener:

· Series temporales estacionarias (en sentido débil): una serie es estacionaria cuando es estable, es decir, cuando la media y la variabilidad son constantes a lo largo del tiempo. Esto se refleja gráficamente en que los valores de la serie tienden a oscilar en torno a la media, que es constante; mientras que la variabilidad también permanece constante en el tiempo. Es una serie básicamente estable, sin que se aprecien aumentos o disminuciones sistemáticas de sus valores. Para este tipo de series tienen sentido conceptos como la media y la varianza.

Dependiendo de la naturaleza de las observaciones, algunos autores clasifican las series temporales en discretas y continuas. En cualquier caso, y siendo coherentes con la definición que dimos de serie temporal, nosotros entenderemos como tal las que son de tipo discreto.

3. OBJETIVOS DEL ANÁLISIS DE SERIES TEMPORALES.

El análisis de series temporales puede tener objetivos muy diversos. Podemos destacar dos de ellos:

· Descripción: cuando se estudia una serie temporal, lo primero que se tiene que hacer es representarla gráficamente y considerar las medidas descriptivas básicas, para así lograr una mejor comprensión del fenómeno en estudio. Así, se tiene que considerar:

- Si los datos presentan forma creciente o decreciente (tendencia).

- Si existe influencia de ciertos períodos de cualquier unidad de tiempo (estacionalidad).

- Si aparecen observaciones extrañas o discordantes.

· Predicción: cuando se observan los valores de una serie se pretende, normalmente, no sólo explicar el pasado, sino también predecir el futuro. Se trata de encontrar características o patrones de comportamiento para que, a partir de ellos, se realicen predicciones sobre valores futuros de la variable analizada.

4. APUNTES Y OBSERVACIONES.

La diferencia esencial entre las series temporales y los análisis no temporales (Estadística descriptiva, Diseño de experimentos o Regresión) es que en estos no importa el orden en que están tomadas las observaciones, mientras que en las series temporales el orden es muy importante, y variarlo supone cambiar la información contenida en la serie.

Hay que tener en cuenta que si se hacen predicciones para instantes de tiempo muy lejanos al del período del que se disponen datos, se corre el riesgo de cometer grandes errores. Una de las razones que podrían propiciar dichos errores sería que, entre los instantes de tiempo para los que se tienen valores, y los instantes de tiempo para los que se quieren realizar predicciones, tuviera lugar algún cambio estructural de cierta importancia. Un ejemplo sobre este aspecto sería la predicción de venta de ordenadores de sobremesa en el momento en que los ordenadores portátiles aún no se comercializaban.

Otra observación importante sobre el estudio de las series temporales es el hecho de que, en una mayoría de casos, los datos u observaciones que componen la serie se toman en instantes de tiempo equidistantes. La diferencia entre los instantes de tiempo depende de la característica en estudio y de los objetivos del análisis; dicha diferencia se denomina periodicidad de la serie. Algunos ejemplos de periodicidad son:

- Anual: se toma un dato cada año.
- Semestral: se toma un dato cada seis meses.
- Trimestral: se toma un dato cada tres meses.
- Mensual: se toma un dato cada mes.
- Semanal: se toma un dato cada semana.
- Diaria: se toma un dato cada día.

5. REFERENCIAS.

- M.A. Palacios, F.A. López Hernández, J. García Córdoba y M. Ruíz Marín. Introducción a la estadística para la empresa. Librería Escarabajal.
- F. Martín-Pliego López. Introducción a la estadística económica y empresarial. Ed. Thomson.
- J.M. Casas, J. Callealta, J. Núñez, M. Toledo y C. Ureña. Curso Básico de Estadística Descriptiva. INAP (1986).
- J.A. Hermoso Gutiérrez y A. Hernández Bastida. Curso Básico de Estadística Descriptiva y Probabilidad. Ed. Némesis (1997).
- J.C. Brocklebank y D.A. Dickey. SAS for forecasting Time Series. Wiley Inter-Science (2003).
- B. Abraham y J. Ledolter. Statistical Methods for Forecasting. Wiley (1983).
- A. Aznar y F.J. Trívez. Métodos de Predicción en Economía II. Análisis de Series Temporales. Ariel Economía (1993).
- P.J. Brockwell y R.A. Davis. Introduction to Time Series and Forecasting. Springer texts in Statistics (2002).
- D. Peña Sánchez de Rivera. Análisis de series temporales. Alianza Editorial (2005).
- D. Peña, G.C. Tiao y R.S. Tsay. A course in time series analysis. John Wiley & Sons (2001).
- E. Uriel y A. Peiró. Introducción al Análisis de Series Temporales. Editorial AC (2000).

CAPÍTULO 2

DESCRIPCIÓN Y APLICACIÓN DE LOS TEST UTILIZADOS

En este capítulo vamos a explicar los siguientes test:

- Test de M. Matilla y M. Ruíz.
- Test de J.S. Cánovas y A. Guillamón.
- Test BDS.

Daremos un resumen teórico de cada uno de ellos, explicaremos los conceptos teóricos en los que se apoyan y daremos ejemplos explicativos para mostrar la forma de aplicar cada uno de ellos.

1. TEST DE M. MATILLA Y M. RUÍZ.

1.1 DESCRIPCIÓN DEL TEST. M. MATILLA Y M. RUÍZ (2008).

En este apartado vamos a presentar la descripción del test que dieron sus propios autores en M. Matilla y M. Ruíz (2008).

Definiciones y notación.

Sea $\{X_t\}_{t \in I}$ una serie temporal de valores reales. Para un entero positivo $m \geq 2$, denotemos con S_m al grupo simétrico de orden $m!$, que es el grupo formado por todas las permutaciones de longitud m. Sea $\pi_i = (i_1, i_2, \dots, i_m) \in S_m$. Llamemos π_i a un símbolo del grupo simétrico S_m. El entero positivo *m* es normalmente conocido como *dimensión de inmersión*.

Definamos ahora un patrón de orden para un símbolo $\pi_i = (i_1, i_2, \dots, i_m) \in S_m$ en un tiempo $t \in I$ dado. Finalmente consideremos que la serie temporal está inmersa en un espacio m-dimensional, tal y como sigue:

$$X_m(t) = (X_{t+1}, X_{t+2}, \dots, X_{t+m}) \; para \; t \in I \quad (1)$$

Entonces decimos que t es del tipo π_i si y sólo si $\pi_i = (i_1, i_2, \dots, i_m)$ es el único símbolo en el grupo S_m cumpliendo las siguientes dos condiciones:

(a) $X_{t+i_1} \leq X_{t+i_2} \leq \dots \leq X_{t+i_m} \quad (2)$
(b) $i_{s-1} < i_s \; si \; X_{t+i_{s-1}} = X_{t+i_s} \quad (3)$

La condición (b) garantiza la unicidad del símbolo π_i. Esto está justificado si los valores de X_t tienen una distribución continua tal que valores iguales sean muy poco comunes, con una probabilidad teórica de ocurrencia igual a 0.

Notar que para todo t, tal que t es de tipo π_i, la m-historia $X_m(t)$ es convertida en un único símbolo π_i. El símbolo π_i describe cómo la ordenación de los instantes $t + 0 < t + 1 < \dots < t + (m - 1)$ es convertida en la ordenación de los valores de la serie temporal analizada. Con objeto de ver esto, el siguiente ejemplo será de ayuda.

Tomemos como dimensión de inmersión $m = 3$. De este modo el grupo simétrico es:

$$S_3 = \{(0,1,2), (0,2,1), (1,0,2), (1,2,0), (2,0,1), (2,1,0)\} \quad (4)$$

Consideremos la serie temporal de siete valores:

$$\{X_1 = 2, X_2 = 8, X_3 = 6, X_4 = 5, X_5 = 4, X_6 = 9, X_7 = 3\} \quad (5)$$

Entonces para $t = 2$ tenemos que $X_{t+2} = 5 < X_{t+1} = 6 < X_{t+0} = 8$, y por tanto tenemos que el período $t = 2$ es de tipo $(2,1,0)$.

También, dada una serie temporal $\{X_t\}_{t \in I}$ y una dimensión de inmersión m, se puede calcular fácilmente la frecuencia relativa de un símbolo $\pi \in S_m$ mediante:

$$p(\pi) \coloneqq p_\pi = \frac{\# \{t \in I | \; t \; es \; de \; tipo \; \pi\}}{|I| - m + 1} \quad (6)$$

donde $|I|$ denota el cardinal del conjunto I.

Entonces para la serie temporal dada en (5) tenemos que la 3-historia $X_3(1) = (X_1 = 2, X_2 = 8, X_3 = 6)$ es representada por el símbolo $(0, 2, 1)$; $X_3(2) = (8, 6, 5)$ y $X_3(3) = (6, 5, 4)$ son representadas por el símbolo $(2, 1, 0)$; $X_3(4) = (5, 4, 9)$ es representado por el símbolo $(1, 0, 2)$ y finalmente $X_3(5) = (4, 9, 3)$ es representado por el símbolo $(1, 2, 0)$. Por tanto obtenemos que $p\big((0, 2, 1)\big) = \frac{1}{5}$, $p\big((2, 1, 0)\big) = \frac{2}{5}$, $p\big((1, 0, 2)\big) = \frac{1}{5}$, $p\big((1, 2, 0)\big) = \frac{1}{5}$, $p\big((0, 1, 2)\big) = 0$ y $p\big((2, 0, 1)\big) = 0$.

Ahora, en este escenario, podemos definir la entropía de permutación de la serie temporal $\{X_t\}_{t\in I}$ para una dimensión de inmersión $m \geq 2$. Esta entropía es definida como la entropía de Shannon de los $m!$ distintos símbolos que siguen:

$$h(m) = -\sum_{\pi\in S_m} p_\pi \ln(p_\pi) \qquad (7)$$

La entropía de permutación, $h(m)$, es la información contenida en la comparación de m valores consecutivos de la serie temporal. Está claro que $0 \leq h(m) \leq \ln(m!)$, donde el límite inferior se alcanza para una serie que sólo presente una permutación admisible, y el límite superior para un sistema completamente aleatorio (secuencia i.i.d.), donde las $m!$ posibles permutaciones aparecen con la misma probabilidad. Para la serie dada en (5) tenemos que $h(3) = -3\frac{1}{5}\ln\left(\frac{1}{5}\right) - \frac{2}{5}\ln\left(\frac{2}{5}\right) \approx 1.332179$.

Construcción y propiedades del test de independencia.

Sea $\{X_t\}_{t\in I}$ una serie temporal y sea m una determinada dimensión de inmersión. Con el objeto de construir un test para la independencia de series, vamos a considerar la siguiente hipótesis nula:

$$H_0\colon \{X_t\}_{t\in I}\ i.i.d. \qquad (8)$$

frente a la hipótesis alternativa de que las observaciones no provienen de una secuencia i.i.d..

Ahora, para un símbolo $\pi_i = (i_1, i_2, \dots, i_m) \in S_m$, definimos la variable aleatoria $Z_{\pi_i t}$ como sigue:

$$Z_{\pi_i t} = \begin{cases} 1\ si\ X_{t+i_1} \leq X_{t+i_2} \leq\ \dots\ \leq X_{t+i_m} \\ 0\ en\ los\ demás\ casos \end{cases} \qquad (9)$$

es decir, tenemos $Z_{\pi_i t} = 1$ si y sólo si t es del tipo π_i, $Z_{\pi_i t} = 0$ en los demás casos.

Entonces $Z_{\pi_i t}$ es una variable de Bernoulli con probabilidad de 'éxito' p_{π_i}, donde 'éxito' significa que t es del tipo π_i. Es sencillo notar que:

$$\sum_{i=1}^{m!} p_{\pi_i} = 1 \qquad (10)$$

Ahora asumamos que el conjunto I es finito y de orden T. Entonces estamos interesados en conocer cuantas $t´s$ son de tipo π_i para todos los símbolos $\pi_i \in S_m$. Sea $K = T - m + 1$. Con la intención de dar respuesta a la pregunta, construimos la siguiente variable:

$$Y_{\pi_i} = \sum_{t=1}^{K} Z_{\pi_i t} \qquad (11)$$

La variable Y_{π_i} puede tomar los valores $\{0, 1, 2, \dots, K\}$. Entonces se tiene que la variable Y_{π_i} es la variable aleatoria binomial:

$$Y_{\pi_i} \sim B(K, p_{\pi_i}) \qquad (12)$$

Para cada símbolo $\pi_i \in S_m$, vamos a denotar:

$$n_{\pi_i} = \# \{t \in I |\ t\ es\ de\ tipo\ \pi_i\} \qquad (13)$$

para $i = 1, 2, \dots, m!$. Entonces, bajo la hipótesis nula H_0, la función de densidad conjunta de las $m!$ variables $(Y_{\pi_1}, Y_{\pi_2}, \dots, Y_{\pi_{m!}})$ es:

$$P(Y_{\pi_1} = a_1, Y_{\pi_2} = a_2, \dots, Y_{\pi_{m!}} = a_{m!}) = \frac{(a_1 + a_2 + \ \dots + a_{m!})!}{a_1!\, a_2! \dots\ a_{m!}!} p_{\pi_1}^{a_1} p_{\pi_2}^{a_2} \dots\ p_{\pi_{m!}}^{a_{m!}} \qquad (14)$$

donde $a_1 + a_2 + \ \dots + a_{m!} = K$. En consecuencia, la distribución conjunta de las $m!$ variables $(Y_{\pi_1}, Y_{\pi_2}, \dots, Y_{\pi_{m!}})$ es una distribución multinomial.

La función de probabilidad de la distribución (14) es:

$$L(p_{\pi_1}, p_{\pi_2}, \dots, p_{\pi_{m!}}) = \frac{K!}{n_{\pi_1}!\, n_{\pi_2}! \dots\ n_{\pi_{m!}}!} p_{\pi_1}^{n_{\pi_1}} p_{\pi_2}^{n_{\pi_2}} \dots\ p_{\pi_{m!}}^{n_{\pi_{m!}}} \qquad (15)$$

y teniendo en cuenta $\sum_{i=1}^{m!} p_{\pi_i} = 1$ se tiene:

$$L(p_{\pi_1}, p_{\pi_2}, \dots, p_{\pi_{m!}}) = \frac{K!}{n_{\pi_1}!\, n_{\pi_2}! \dots\ n_{\pi_{m!}}!} p_{\pi_1}^{n_{\pi_1}} p_{\pi_2}^{n_{\pi_2}} \dots\ (1 - p_{\pi_1} - p_{\pi_2} - \ \dots - p_{\pi_{m!-1}})^{n_{\pi_{m!}}} \qquad (16)$$

Entonces el logaritmo de esta función de probabilidad es:

$$\ln\left(L(p_{\pi_1}, p_{\pi_2}, \dots, p_{\pi_{m!}})\right) = \ln\left(\frac{K!}{n_{\pi_1}!\, n_{\pi_2}! \dots\ n_{\pi_{m!}}!}\right) + \sum_{i=1}^{m!-1} n_{\pi_i} \ln(p_{\pi_i}) + n_{\pi_{m!}} \ln(1 - p_{\pi_1} - p_{\pi_2} - \ \dots - p_{\pi_{m!-1}}) \qquad (17)$$

Para determinar el estimador de probabilidad máximo $\hat{p}_{\pi_i}$ de p_{π_i} para todo $i = 1, 2, \dots, m!$, resolvemos la siguiente ecuación:

$$\frac{\partial \ln\left(L(p_{\pi_1}, p_{\pi_2}, \dots, p_{\pi_{m!}})\right)}{\partial p_{\pi_i}} = 0 \quad (18)$$

y obtenemos:

$$\hat{p}_{\pi_i} = \frac{n_{\pi_i}}{K} \quad (19)$$

Entonces el estadístico de índice de probabilidad es (ver Lehman (1986)):

$$\lambda(\mathrm{Y}) = \frac{\frac{K!}{n_{\pi_1}!\,n_{\pi_2}!\dots\,n_{\pi_{m!}}!} p_{\pi_1}^{n_{\pi_1}} p_{\pi_2}^{n_{\pi_2}} \dots\, p_{\pi_{m!}}^{n_{\pi_{m!}}}}{\frac{K!}{n_{\pi_1}!\,n_{\pi_2}!\dots\,n_{\pi_{m!}}!} \hat{p}_{\pi_1}^{n_{\pi_1}} \hat{p}_{\pi_2}^{n_{\pi_2}} \dots\, \hat{p}_{\pi_{m!}}^{n_{\pi_{m!}}}} = \frac{\prod_{i=1}^{m!} p_{\pi_i}^{n_{\pi_i}}}{\prod_{i=1}^{m!} \left(\frac{n_{\pi_i}}{K}\right)^{n_{\pi_i}}} = K^{\sum_{i=1}^{m!} n_{\pi_i}} \prod_{i=1}^{m!} \left(\frac{p_{\pi_i}}{n_{\pi_i}}\right)^{n_{\pi_i}}$$
$$= K^K \prod_{i=1}^{m!} \left(\frac{p_{\pi_i}}{n_{\pi_i}}\right)^{n_{\pi_i}} \quad (20)$$

Por otro lado, $G(m) = -2\ln(\lambda(\mathrm{Y}))$ converge asintóticamente a una distribución Chi-cuadrado con $m! - 1$ grados de libertad (ver Lehman (1986)). De ahí:

$$G(m) = -2\ln(\lambda(\mathrm{Y})) = -2\left[K\ln(K) + \sum_{i=1}^{m!} n_{\pi_i} \ln\left(\frac{p_{\pi_i}}{n_{\pi_i}}\right)\right] \sim \chi^2_{m!-1} \quad (21)$$

Ahora, bajo la hipótesis nula H_0 se tendrá que todas las permutaciones tienen igual probabilidad, y por tanto $p_{\pi_i} = \frac{1}{m!}$ para todo $i = 1, 2, \dots, m!$. Entonces se deduce que:

$$G(m) = -2K\left[\ln(K) + \sum_{i=1}^{m!} \frac{n_{\pi_i}}{K} \ln\left(\frac{p_{\pi_i}}{n_{\pi_i}}\right)\right] = -2K\left[\ln(K) + \sum_{i=1}^{m!} \frac{n_{\pi_i}}{K}\left(\ln\left(\frac{1}{m!}\right) - \ln(n_{\pi_i})\right)\right]$$
$$= -2K\left[\ln(K) + \sum_{i=1}^{m!} \frac{n_{\pi_i}}{K}\left(\ln\left(\frac{1}{m!}\right) - \ln\left(\frac{n_{\pi_i}}{K}\right) - \ln(K)\right)\right] \quad (22)$$

Teniendo en cuenta ahora que $h(m) = -\sum_{i=1}^{m!} p_{\pi_i} \ln(p_{\pi_i}) = -\sum_{i=1}^{m!} \frac{n_{\pi_i}}{K} \ln\left(\frac{n_{\pi_i}}{K}\right)$ tenemos que:

$$G(m) = -2K\left[\ln\left(\frac{1}{m!}\right) + h(m)\right] = -2K[h(m) - \ln(m!)] = 2K[\ln(m!) - h(m)] \quad (23)$$

Por tanto hemos probado el siguiente teorema.

Teorema 1. Sea $\{X_t\}_{t\in I}$ una serie temporal de valores reales, con $|I| = T$. Denotemos mediante $h(m)$ la entropía de permutación definida en (7) para una dimensión de inmersión entera $m > 2$, con $m \in \mathbb{N}$. Si la serie temporal $\{X_t\}_{t\in I}$ es i.i.d., entonces la transformación afín de la entropía de permutación:

$$G(m) = 2(T - m + 1)[\ln(m!) - h(m)] \quad (24)$$

está asintóticamente $\chi^2_{m!-1}$ distribuida.

Sea α un número real tal que $0 \leq \alpha \leq 1$. Sea χ_α^2 tal que:

$$P(\chi_{m!-1}^2 > \chi_\alpha^2) = \alpha \quad (25)$$

Entonces para testar:

$$H_0: \{X_t\}_{t \in I} \; i.i.d. \quad (26)$$

la regla de decisión en la aplicación del test $G(m)$ al $100(1-\alpha)\%$ de nivel confianza es:

$$\begin{cases} Si\ 0 \leq G(m) \leq \chi_\alpha^2 \ aceptamos\ H_0, \\ \quad en\ otro\ caso\ rechazamos\ H_0. \end{cases} \quad (27)$$

Consistencia del test $G(m)$.

A continuación probamos que el test $G(m)$ es consistente para una amplia variedad de procesos dependientes en serie. Esto es una propiedad valiosa, ya que el test rechazará asintóticamente independencia en serie siempre que haya dependencia en serie dentro de las m-historias. Vamos a denotar mediante $\hat{G}(m)$ el estimador de $G(m)$.

Teorema 2. Sea $\{X_t\}_{t \in I}$ un proceso estrictamente estacionario, y $m > 2$ con $m \in \mathbb{N}$. Entonces $\lim_{T\to\infty} Pr(\hat{G}(m) > C) = 1$ bajo la dependencia en serie de orden $\leq m$ para todo $0 < C < \infty$, $C \in \mathbb{R}$.

De este modo, el test basado en $G(m)$ es consistente frente a toda dependencia en serie de orden $\leq m$. Ya que el Teorema 2 implica que $G(m) \to \infty$ con probabilidad tendiendo a 1 bajo la dependencia en serie de orden $\leq m$, entonces los valores críticos de la cola superior son apropiados. Subrayemos en este punto que estas condiciones sobre el proceso son menos restrictivas que las requeridas en los procedimientos de otros test (ver Hong y White (2005), Skaug y Tjøstheim (1996)).

Selección del parámetro de libertad m (dimensión de inmersión).

Es importante señalar, desde un punto de vista práctico, que el investigador tiene que elegir la dimensión de inmersión m para computar la entropía de permutación, y así calcular es estadístico $G(m)$. Afortunadamente, es fácil llevar a cabo esta elección. Notar que T debe ser más grande que el número de símbolos de permutación ($m!$) para tener al menos el mismo número de m-historias como posibles símbolos (eventos) π_i, con $i = 1, 2, \dots, m!$. Cuando la distribución χ^2 es utilizada en la práctica, y todas las frecuencias esperadas son ≥ 5, el límite tabulado de la distribución χ^2 da, como una regla, el valor de χ_α^2 con una aproximación suficiente para propósitos habituales (ver Rohatgi (1976), capítulo 10). Por esta razón, necesitamos trabajar con conjuntos de datos que contengan, al menos, cinco veces el número de posibles eventos (símbolos). Por ejemplo, un conjunto de datos de 200 observaciones es suficiente para computar $G(4)$, porque 24 símbolos son contenidos para $m = 4$; de modo similar, 600 observaciones es el conjunto de datos más pequeño que puede ser considerado para una dimensión de inmersión $m = 5$, ya que en este caso 120 (=5!) símbolos pueden ser encontrados. Por otro lado, para $m = 3$, sólo son analizados seis posibles símbolos, y entonces el nivel de información captada de estos símbolos es muy limitado, y por tanto no sugerimos el uso de $m = 3$. Calcularemos la entropía de permutación de manera que el investigador no tenga que elegir la dimensión de inmersión: para un conjunto dado de datos de T observaciones, la dimensión de inmersión será el mayor m que satisfaga $5m! \leq T$ con $m = 2, 3, 4, ...$. Por ejemplo, en caso de $T = 500$, tomaremos $m = 4$. Por otro lado, fijémonos en que si T es demasiado larga ($T > 25200$), entonces la m seleccionada será demasiado grande (sería $m > 7$), y entonces el procedimiento sería demasiado costoso en términos de tiempo de computación. Por esta razón, y dada la longitud habitual de las series temporales económicas, recomendamos operar con $m = 6$ para $T > 3600$.

Como ha sido indicado antes, merece la pena notar que la posible dependencia detectada por el test $G(m)$ tiene que ser de orden $\leq m$. Esto es debido al hecho de que si la estructura de dependencia del proceso es de orden $> m$, entonces esta dependencia no estará presente en muchas m-historias, y por tanto los símbolos podrían no ser captados.

1.2 EJEMPLO ILUSTRATIVO.

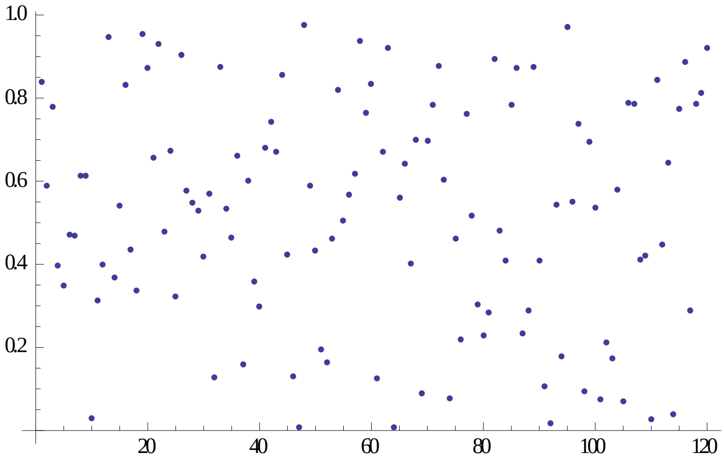
Incluimos a continuación un ejemplo de aplicación del test que desarrollaron Matilla y Ruíz. Se pretende mostrar la forma en que se realiza este test, explicando cada uno de los pasos que son necesarios realizar.

Los cálculos que se van a presentar han sido realizados con un programa hecho en el entorno MATHEMATICA (se puede ver su código en los Anexos).

Comenzamos creando una serie aleatoria de 120 elementos ($T = 120$) utilizando la distribución uniforme $[0,1]$:

{0.839409, 0.588551, 0.778073, 0.397896, 0.348357, 0.471474, 0.468131, 0.614143, 0.613195, 0.0289137, 0.313489, 0.400435, 0.946485, 0.367859, 0.541083, 0.832542, 0.43537, 0.336363, 0.95532, 0.872107, 0.65757, 0.92973, 0.479397, 0.673856, 0.321719, 0.903258, 0.577217, 0.547652, 0.529578, 0.41839, 0.568839, 0.128183, 0.874918, 0.534439, 0.464147, 0.661403, 0.158829, 0.602115, 0.358398, 0.297761, 0.680279, 0.742006, 0.672014, 0.856503, 0.422664, 0.130124, 0.00759055, 0.976151, 0.588305, 0.433604, 0.194095, 0.164221, 0.462386, 0.819786, 0.505276, 0.56709, 0.619041, 0.937387, 0.765605, 0.834317, 0.126486, 0.670503, 0.919557, 0.00890141, 0.560714, 0.641891, 0.402641, 0.700451, 0.0901355, 0.697564, 0.782702, 0.877927, 0.60382, 0.0767138, 0.462744, 0.220163, 0.762244, 0.517364, 0.302864, 0.229136, 0.283183, 0.894947, 0.481594, 0.408009, 0.782776, 0.87347, 0.234633, 0.2891, 0.876154, 0.409021, 0.106186, 0.0175708, 0.544408, 0.179469, 0.970995, 0.550357, 0.738709, 0.0936467, 0.695791, 0.536083, 0.0761648, 0.211064, 0.174647, 0.58063, 0.0712581, 0.789619, 0.786204, 0.411443, 0.419956, 0.0269872, 0.844011, 0.446741, 0.644742, 0.0383106, 0.775063, 0.887509, 0.288843, 0.786023, 0.812506, 0.920627}

La representación gráfica de ésta es:



Para el análisis de esta serie tenemos que comenzar eligiendo el valor de la dimensión de inmersión m. Siguiendo la recomendación dada en Matilla y Ruíz (2008), tomaremos para m el valor más grande que cumpla la condición $T \geq 5m!$; es decir, la dimensión de inmersión tomada tiene que ser tal que tengamos un número de m-historias que sea, al menos, 5 veces el número admisible de permutaciones diferentes. Este criterio nos conduce al valor $m = 4$, ya que $120 = T \geq 5m! = 5 \cdot 4! = 120$; notar que no sería válido el valor $m = 5$, puesto que $120 = T \ngeq 5m! = 5 \cdot 5! = 600$.

Por otra parte, también hay que fijar el nivel de confianza con el que se desea realizar el test. Se tomará $nivel\ de\ confianza = 95\%$, que es el valor más usual.

Concretados los valores de la dimensión de inmersión y del nivel de confianza, podemos adentrarnos en el análisis de la serie.

En primer lugar vamos a especificar las m-historias con las que se llevará a cabo la aplicación del test. Seguidamente remarcamos cuál sería la primera de estas historias (resaltada en negrita):

{**0.839409, 0.588551, 0.778073, 0.397896,** 0.348357, 0.471474, 0.468131, …}

la segunda sería:

{0.839409, **0.588551, 0.778073, 0.397896, 0.348357,** 0.471474, 0.468131, …}

y la tercera:

{0.839409, 0.588551, **0.778073, 0.397896, 0.348357, 0.471474,** 0.468131, …}

Iríamos tomando cada trozo de la sería desplazándonos una posición con respecto al trozo inmediatamente anterior. Hay que prestar atención a que las m-historias que se están seleccionando se superponen unas con otras; ya que en el test de Cánovas y Guillamón que estudiaremos más adelante no se da dicha superposición. El último trozo de la serie es:

{…, 0.0383106, 0.775063, 0.887509, **0.288843, 0.786023, 0.812506, 0.920627**}

Utilizando la misma notación que los autores de este test, cada uno de estos trozos se denotaría como sigue:

$$X_4(1) = (0.839409, 0.588551, 0.778073, 0.397896)$$

$$X_4(2) = (0.588551, 0.778073, 0.397896, 0.348357)$$

$$X_4(3) = (0.778073, 0.397896, 0.348357, 0.471474)$$

...

$$X_4(117) = (0.288843, 0.786023, 0.812506, 0.920627)$$

donde cada vector $X_m(t)$ representa la historia de longitud m, cuyo primer elemento es el que está en la posición t dentro de la serie.

A cada uno de los vectores que han resultado le tenemos que asociar un símbolo $\pi_i = (i_1, i_2, i_3, i_4) \in S_4$, donde S_4 denota el grupo formado por todas las permutaciones de longitud 4:

$$S_4 = \{(0,1,2,3), (0,1,3,2), (0,2,1,3), (0,2,3,1), (0,3,1,2), (0,3,2,1), \dots\}$$

La asignación de un símbolo π_i a cada permutación se lleva a cabo siguiendo las dos condiciones siguientes:

(a) $X_{t+i_1} \leq X_{t+i_2} \leq X_{t+i_3} \leq X_{t+i_4}$
(b) $i_{s-1} < i_s \ si \ X_{t+i_{s-1}} = X_{t+i_s}$

Como vemos, cada símbolo π_i define el orden en que están los elementos del vector al que está asociado.

Procedemos ya a la asignación de un símbolo π a cada vector $X_m(t)$. Empecemos con la 4-historia $X_4(1)$:

$$X_4(1) = (0.839409, 0.588551, 0.778073, 0.397896) \Rightarrow \pi = (3,1,2,0)$$

ya que $0.397896 < 0.588551 < 0.778073 < 0.839409$.

Para $X_4(2)$ se tiene:

$$X_4(2) = (0.588551, 0.778073, 0.397896, 0.348357) \Rightarrow \pi = (2,3,1,0)$$

Y para el vector $X_4(3)$:

$$X_4(3) = (0.778073, 0.397896, 0.348357, 0.471474) \Rightarrow \pi = (3,1,0,2)$$

Repetiríamos este procedimiento para todos los vectores $X_m(t)$ hasta llegar al último, que es:

$$X_4(117) = (0.288843, 0.786023, 0.812506, 0.920627) \Rightarrow \pi = (0,1,2,3)$$

El siguiente paso es calcular la frecuencia relativa de cada símbolo $\pi \in S_4$, tal y como indica la siguiente fórmula:

$$p(\pi) \coloneqq p_\pi = \frac{\#\,\{t \in I \mid t \ es \ de \ tipo \ \pi\}}{|I| - m + 1}$$

donde $|I| = 120$ es el cardinal del conjunto $I = \{1, 2, 3, \dots, 120\}$.

La lista de todas las frecuencias relativas de cada $\pi \in S_4$, exceptuando aquellas cuyo valor es cero, se expone a continuación:

{2/39, 4/117, 4/117, 1/13, 4/117, 1/39, 5/117, 5/117, 4/117, 5/117, 2/39, 2/39, 8/117, 2/39, 7/117, 2/117, 7/117, 4/117, 4/117, 4/117, 4/117, 1/39, 4/117, 1/39}

Una vez determinadas todas las frecuencias relativas p_π, es muy sencillo calcular la entropía de permutación:

$$h(m) = -\sum_{\pi \in S_m} p_\pi \ln(p_\pi) = 3.121341$$

Recordemos que si la serie temporal es i.i.d., entonces la transformación afín de la entropía de permutación:

$$G(m) = 2(T - m + 1)[\ln(m!) - h(m)]$$

está asintóticamente $\chi^2_{m!-1}$ distribuida.

Dado que $nivel\ de\ confianza = 95\%$, entonces el nivel de significación α es:

$$\alpha = 1 - \frac{nivel\ de\ confianza}{100} = 0.05$$

Entonces para testar:

$$H_0: \{X_t\}_{t \in I}\ i.i.d.$$

la regla de decisión con $nivel\ de\ confianza = 95\%$ es:

$$\begin{cases} Si\ 0 \leq G(m) \leq \chi^2_{0.05}\ aceptamos\ H_0, \\ \quad en\ otro\ caso\ rechazamos\ H_0. \end{cases}$$

Calculamos el estadístico de contraste $G(m)$:

$$G(m) = 2(T - m + 1)[\ln(m!) - h(m)] = 13.270883$$

Teniendo en cuenta que $\chi^2_{0.05} = 35.172462$, entonces:

$$0 \leq G(m) = 13.270883 \leq \chi^2_{0.05} = 35.172462 \Rightarrow la\ serie\ es\ i.i.d.$$

1.3 REFERENCIAS.

Referencia principal.

- M. Matilla García y J. Rodríguez Ruíz. A non-parametric Independence test using permutation entropy. Journal of Econometrics 144 (2008), 139-155.

Otras referencias.

- Y. Hong y H. White. Asymptotic distribution theory for nonparametric entropy measures of serial dependence. Econometrica 73 (2005), 837-901.

- E.L. Lehmann. Testing Statistical Hypothesis. Wiley, New York (1986).
- V.K. Rohatgi. An Introduction to Probability Theory and Mathematical Statistics. Wiley, New York (1976).
- H.J. Skaug y D. Tjøstheim. Measures of Distance Between Densities with Application to Testing for Serial Independence. En: P. Robinson, P. Rosenblatt, M. (Eds.), Time Series Analysis in Memory of E.J. Hanan. New York, Springer (1996), 363-377.

2. TEST DE J.S. CÁNOVAS Y A. GUILLAMÓN.

Vamos a exponer el test desarrollado por J.S. Cánovas y A. Guillamón en 2009. Al igual que como se ha hecho con el test anterior, la descripción que damos de éste es justamente la que dieron sus propios autores.

2.1 DESCRIPCIÓN DEL TEST. J.S. CÁNOVAS Y A. GUILLAMÓN (2009).

Introducción.

Recientemente varios autores han señalado que las permutaciones pueden ser una buena herramienta para estudiar las series temporales. El concepto de entropía de permutación ha sido utilizado como medida del grado de predictibilidad de una serie temporal; vamos a recordar dicho concepto.

Sea $(x_n)_{n=1}^T$, $T \in \mathbb{N}$, una secuencia procedente de datos reales o simulados. Para $m \geq 2$, sea S_m el grupo de las permutaciones de longitud m, cuyo cardinal es $m!$. Sea $x_m(r) = (x_r, x_{r+1}, \ldots, x_{r+m-1})$, $1 \leq r \leq T - m + 1$, una ventana deslizante tomada de la secuencia $(x_n)_{n=1}^T$. Decimos que esa ventana $x_m(r)$ es de tipo $\pi \in S_m$, si y sólo si $\pi = (i_1, i_2, \ldots, i_m)$ es el único elemento de S_m cumpliendo las dos siguientes condiciones:

(c1) $x_{r+i_1} \leq x_{r+i_2} \leq \ \ldots \ \leq x_{r+i_m} \quad (1)$

(c2) $i_{s-1} < i_s \ si \ x_{r+i_{s-1}} = x_{r+i_s} \quad (2)$

Por ejemplo, si tenemos la serie temporal $(1.1, 1.2, 0.9, 1.2)$, entonces tenemos que $(2,3,1) \in S_3$ es la permutación asociada a $x_3(1)$ y $(2,3,1,4) \in S_4$ es la asociada a $x_4(1)$.

El entero positivo m es usualmente conocido como dimensión de inmersión. Fijemos $k \leq T - m + 1$. Para cada $\pi \in S_m$, la probabilidad de que ocurra π es estimada por la frecuencia relativa:

$$p(\pi) = \frac{\# \{x_m(j), j = 1, 2, \ldots, k : x_m(j) \ es \ de \ tipo \ \pi\}}{k} \quad (3)$$

Una permutación $\pi \in S_m$ con $p(\pi) > 0$ para algún $k \leq T - m + 1$ es denominada una permutación admisible de $(x_n)_{n=1}^T$. De ahí, la entropía de permutación viene dada por el número no negativo:

$$\mathcal{H}_S(x_n, m) = -\sum_{\pi \in S_m} p(\pi) \ln p(\pi) \quad (4)$$

La expresión anterior fue introducida en Bandt y Pompe (2002) para estudiar la complejidad de las series temporales, y fue útil para crear un test que comprobara la independencia de dichas series (ver Matilla y Ruíz (2008), y Amigó, Zambrano y Sanjuán (2007)). Sin embargo, el principal resultado que muestra que este número (o de forma más exacta, el número admisible de permutaciones) puede ser usado como una herramienta para el estudio de la complejidad de las series temporales, procede de una definición topológica análoga que denominamos entropía de permutación topológica:

$$\mathcal{H}_T(x_n, m) = \ln \# \mathcal{A}_m \qquad (5)$$

donde $\mathcal{A}_m = \{\pi \in S_m: existe\ j\ tal\ que\ x_m(j)\ es\ de\ \pi\ tipo\}$ es el conjunto de permutaciones admisibles. La motivación de esta definición viene de Bandt, Keller y Pompe (2002) (ver también Amigó y Kennel (2007)); el principal resultado relacionado con ésta se explica más abajo.

Sea $I = [0,1]$ y $f: I \longrightarrow I$ un mapa monótono a trozos y continuo, esto es, existen $0 = x_0 < x_1 < \ \dots\ < x_n = 1$ tal que $f_{[x_i, x_{i+1}]}$ es estrictamente monótona para $i = 0, 1, \dots, n-1$. Para algún $\pi \in S_m$, definimos la partición $\mathcal{P}_\pi = \{x \in I: f^{\pi(1)}(x) < f^{\pi(2)}(x) < \ \dots\ < f^{\pi(n)}(x)\}$. Entonces:

$$h(f) = \lim_{m \to \infty} \frac{1}{m} \ln \# \{\pi \in S_m: \mathcal{P}_\pi \neq \emptyset\} \qquad (6)$$

donde $h(f)$ denota la entropía topológica de f (ver la definición en Bowen (1971)). Recordemos que esa entropía topológica es un número que es interpretado como una medida de la complejidad dinámica de f. En particular, si la entropía topológica es positiva, entonces f es caótico en el sentido de Li y Yorke, y fuertemente caótico en el sentido de Block y Coppel (ver Block y Coppel (1992)). Observemos que $\mathcal{P}_\pi \neq \emptyset$ nos conduce a que existe $x \in I$ para el que su órbita $Orb(x, f) = \left(f^n(x)\right)_{n=0}^{\infty}$ contiene π como una permutación admisible. Este hecho es usado para probar el resultado siguiente que conecta nociones de entropía topológica del análisis de series temporales, con sistemas dinámicos, como sigue. Si $f: I \longrightarrow I$ es continua y monótona a trozos, entonces, para algún $y \in I$ se espera que:

$$h_T\big(Orb(y, f)\big) \leq h(f) = \sup_{x \in I} h_T\big(Orb(x, f)\big) \qquad (7)$$

donde, para una secuencia infinita $(x_n)_{n=1}^{\infty}$, definimos el número

$$h_T(x_n) = \lim_{m \to \infty} \sup \frac{1}{m} \mathcal{H}_T((x_n)_{n=1}^{\infty}, m) \qquad (8)$$

con $\mathcal{H}_T((x_n)_{n=1}^{\infty}, m) = \sup_{k \in \mathbb{N}} \mathcal{H}_T\big((x_n)_{n=1}^{k}, m\big)$.

Pero si nos fijamos en la fórmula (5), podemos comprobar fácilmente que lo único que importa es el número de permutaciones admisibles; el logaritmo es únicamente un factor de escala. Este hecho nos motiva a estudiar la construcción de un test para comprobar la independencia en series temporales basadas únicamente en el número de permutaciones admisibles. La principal ventaja de este test con respecto a otros (como Amigó, Zambrano y Sanjuán (2007); Amigó, Zambrano y Sanjuán (2008) o Matilla y Ruíz (2008)) es que en él se manejan funciones de distribución teóricas de las variables aleatorias W: el número de permutaciones admisibles en la serie $(x_n)_{n=1}^{T}$ (para un valor de m dado). Como resultado se han construido diferentes test estadísticos (asintóticos y no asintóticos). Se debe remarcar que en las referencias mencionadas arriba sólo es construido un test asintótico (Chi-cuadrado).

Análisis estadístico.

Sea $(x_n)_{n=1}^T$, $T \in \mathbb{N}$, una secuencia y sea m una dimensión de inmersión. Vamos a considerar k ventanas deslizantes de longitud m sin solapamiento; se tiene $k = [T/m]$, donde $[x]$ denota la parte entera de un número real x. Vamos a representar los símbolos de cada ventana mediante:

$$x_m(j) = \left(x_{m(j-1)+1}, x_{m(j-1)+2}, \dots, x_{m(j-1)+m}\right) \; para \; j = 1, 2, \dots, k \qquad (9)$$

Proponemos el estudio de la variable aleatoria $W = \# \{\pi \in S_m : p(\pi) > 0\}$, el número de permutaciones admisibles diferentes sobre las k ventanas deslizantes. De este modo, $W = k$ significaría que cada ventana deslizante proporciona un π diferente, mientras que $W = 1$ significaría que todas las ventanas deslizantes tienen el mismo π.

Para obtener la distribución asociada a la variable aleatoria W, necesitamos recordar un problema clásico de análisis combinatorio: el problema de determinar el número de particiones de un conjunto de k objetos en j cajas, para el que la solución viene dada por el siguiente resultado (ver Hazewinkel, Enciclopaedia of Mathematics).

Lema 1. El número de disposiciones de k objetos diferentes en j cajas diferentes con ninguna caja vacía ($k \geq j$) es igual a $j!\, S_2(k, j)$, donde:

$$S_2(k, j) = \frac{1}{j!} \sum_{i=0}^{j} (-1)^i \binom{j}{i} (j - i)^k \qquad (10)$$

es denominado el número de Stirling de segunda especie.

Bajo la suposición de que la secuencia $(x_n)_{n=1}^T$ es obtenida de un conjunto $(X_n)_{n=1}^T$ de variables aleatorias i.i.d. (idénticamente distribuidas), podemos obtener la distribución exacta asociada a la variable aleatoria W como sigue.

Teorema 2. Sea $(X_n)_{n=1}^T$ una secuencia de variables aleatorias separadas en k ventanas deslizantes sin solapamiento ($k = [T/m]$). Sea m la dimensión de inmersión y sea W la variable aleatoria definida arriba. Entonces, bajo la hipótesis de que la secuencia $(X_n)_{n=1}^T$ es un conjunto de variables aleatorias i.i.d., se espera que:

$$Pr(W = j) = \begin{cases} S_2(k, j) \dfrac{(m!)!}{(m! - j)!} \left(\dfrac{1}{m!}\right)^k, j = 1, \dots, \min(k, m!) \\ 0, en \; otro \; caso \end{cases} \qquad (11)$$

donde $S_2(k, j)$ es el correspondiente número de Stirling de segunda especie.

Demostración. Vamos a considerar el siguiente problema equivalente:

“Supongamos que k objetos (π observado en cada ventana deslizante) son distribuidos dentro de $m!$ (el cardinal de S_m) cajas con reemplazamiento, de manera que cada objeto tenga una probabilidad de $1/m!$ de estar en una caja en particular”.

Nuestro problema es calcular la probabilidad de que k objetos sean repartidos en, exactamente, j cajas (para $j = 1, \dots, k$).

· Primero calculamos el número de formas diferentes de elegir j cajas entre $m!$ ($j < m!$). Dadas $m!$ cajas ($\# S_m$), la cantidad de formas de seleccionar j de éstas (notar que hay exactamente j diferentes $\pi \in S_m$ con $p(\pi) \neq 0$) viene dada por:

$$\binom{m!}{j} = \frac{(m!)!}{(m!-j)!\,j!} \quad (12)$$

· Seguidamente, usando el Lema 1, se espera que el número de disposiciones de k objetos diferentes (esto es, π observado en cada ventana deslizante) en j cajas diferentes (sin celdas vacías), es decir $\# \{\pi_i / p(\pi_i) \neq 0\} = j$, viene dado por:

$$j!\,S_2(k,j) \quad (13)$$

· Finalmente, bajo la hipótesis de que la secuencia $(x_n)_{n=1}^{T}$ proceda de un conjunto $(X_n)_{n=1}^{T}$ de variables aleatorias i.i.d., cada objeto ($\pi \in S_m$) tiene una probabilidad de $1/m!$, y la probabilidad de cada selección posible de un conjunto de k objetos (π obtenido sobre las k ventanas deslizantes) viene dado por:

$$\left(\frac{1}{m!}\right)^k \quad (14)$$

Entonces, teniendo en cuenta los cálculos de arriba, obtenemos:

$$Pr(W=j) = j!\,S_2(k,j)\binom{m!}{j}\left(\frac{1}{m!}\right)^k = S_2(k,j)\frac{(m!)!}{(m!-j)!}\left(\frac{1}{m!}\right)^k \quad (15)$$

para $j = 1, \dots, \min(k, m!)$, finalizando así la demostración.

Como consecuencia del Teorema 2 podemos obtener la función de distribución de la variable aleatoria W como:

$$F_W(t) = Pr(W \leq t) = \begin{cases} 0, si\ t < 1 \\ \displaystyle\sum_{j=1}^{[t]} S_2(k,j)\frac{(m!)!}{(m!-j)!}\left(\frac{1}{m!}\right)^k, si\ 1 \leq t \leq \min(k, m!) \\ 1, si\ t > \min(k, m!) \end{cases} \quad (16)$$

Esta función de distribución es esencial para construir diferentes test con el objeto de comprobar si los datos son independientes. Abajo resumimos estos test.

Test 1. Sea $(x_n)_{n=1}^{T}$ una secuencia de valores reales que proceden de un conjunto de variables aleatorias $(X_n)_{n=1}^{T}$. Fijemos $\alpha \in [0,1]$ y sea $f_{W,\alpha}$ un valor real tal que $Pr\big(W < f_{W,\alpha}\big) = \alpha$, donde F_W es la función de distribución dada por (16). Entonces, el test:

$$\begin{cases} H_0: (X_n)_{n=1}^{T}\ es\ un\ conjunto\ de\ variables\ aleatorias\ i.i.d. \\ H_1: (X_n)_{n=1}^{T}\ no\ es\ un\ conjunto\ de\ variables\ aleatorias\ i.i.d. \end{cases} \quad (17)$$

tiene la regla de decisión, con un $100(1-\alpha)\%$ de nivel de confianza, que sigue: si ϖ es el número de permutaciones admisibles diferentes $\pi \in S_m$ sobre la secuencia $(x_n)_{n=1}^{T}$, entonces si $\varpi \geq f_{W,\alpha}$, aceptamos H_0. En otro caso la rechazamos.

La tabla siguiente muestra los valores críticos usando diferentes dimensiones de inmersión m y niveles de confianza usuales. Notar que los valores críticos dependen del número de puntos en las series.

T	90%	95%	99%	90%	95%	99%	90%	95%	99%
1200	24	24	24	100	99	96	169	168	165
2400	24	24	24	116	115	114	299	296	291
4800	24	24	24	120	120	119	472	469	463
7200	24	24	24	120	120	120	574	571	565
	m=4			**m=5**			**m=6**		

Tabla. Valores críticos para niveles de confianza usuales. Los valores son presentados de izquierda a derecha para $m = 4$, $m = 5$ y $m = 6$. Para $m = 4$ aceptamos la hipótesis nula si y sólo si todas las posibles permutaciones son admisibles.

La tabla muestra la dependencia entre el valor crítico, la dimensión de inmersión y el número de puntos en la serie. Asignar una permutación es equivalente a codificar las series. De ahí, cuando el número de códigos diferentes aumenta, la información obtenida de las series temporales usando las permutaciones (códigos) también incrementa. Por tanto, si tenemos una serie de datos con más de 1000 puntos, en general 24 permutaciones pueden no ser suficientes para describir la complejidad, y así, son necesarias dimensiones de inmersión mayores de 4 para construir buenos test.

Test 2. Sea $(X_n)_{n=1}^{T}$ una secuencia de variables aleatorias dividida en N ventanas, sin solapamiento, de longitud n ($N = [T/n]$). Sea m una dimensión de inmersión y sea ϖ_j el número de permutaciones admisibles diferentes π sobre la ventana j, $1 \leq j \leq N$. Entonces, bajo la hipótesis de que la secuencia $(X_n)_{n=1}^{T}$ es un conjunto de variables aleatorias i.i.d., se espera que:

$$\chi^2 = \sum_{i=1}^{\min(k,m!)} \frac{(f_i - e_i)^2}{e_i} \qquad (18)$$

esté asintóticamente distribuido como una $\chi^2_{\min(k,m!)-1}$ cuando $N \to \infty$ ($T \to \infty$), donde $k = [n/m]$, f_i representa la frecuencia observada del evento $\varpi_j = i$ (número de ventanas con exactamente i permutaciones admisibles diferentes) y e_i la frecuencia esperada ($e_i = N \cdot Pr(W = i)$) de cada posible valor de la variable aleatoria W.

El principal problema con el test χ^2 es la elección del número y tamaño de los intervalos: estos es, para una aplicación práctica del test, tenemos que dividir los datos en un número de intervalos de datos, y para cada intervalo tenemos que saber cuántas permutaciones diferentes de S_m están asociadas con cada intervalo. Aunque las reglas de agrupamiento pueden ayudar a producir buenos resultados (por ejemplo, la división en intervalos debe realizarse de forma tal que $e_i \geq 5$ para todo i), el valor del test estadístico depende en cómo sean agrupados los datos. El test de Kolmogorov puede ser usado como un test asintótico alternativo al test χ^2 que no requiere que los datos sean agrupados en intervalos. Por otro lado, el test de Kolmogorov proporciona resultados conservadores para conjuntos de datos discretos (ver Noether (1963), Conover y Kolmogorov (1972) y Gibbons y Chakrabortis (1992)).

Test 3. Sea $W_1, W_2, \dots W_n$ una muestra de variables aleatorias de tamaño n de una variable aleatoria W con función de distribución $F_W(\varpi)$ dada por (16) y sea $F_n(\varpi)$ la función de distribución empírica, esto es $F_n(\varpi) = j/n$ si exactamente j de los n valores W_i son menores o iguales que ϖ ($\varpi = 1, \dots, \min(k, m!)$). Si la hipótesis H_0: $F_n(\varpi) = F_W(\varpi)$ es cierta, entonces se espera que:

$$\lim_{n\to\infty} Pr\left(\sqrt{n}D_n \le t\right) = KS(t) \quad (19)$$

donde $D_n = \max|F_n(\varpi) - F_W(\varpi)|$ y $KS(t) = 1 - 2\sum_{i=1}^{\infty}(-1)^{i-1}e^{-2i^2t^2}$ es la función de distribución continua de la distribución de Kolmogorov.

De acuerdo con el test de Kolmogorov, para un nivel de significación α, la hipótesis H_0 debe ser rechazada si $D_n > \lambda(n)$, donde $\lambda(n)$ es el valor crítico del test de Kolmogorov correspondiente a un nivel de significación α.

2.2 EJEMPLOS ILUSTRATIVOS.

En este apartado vamos a dar un ejemplo de aplicación de cada uno de los tres test que han sido presentados más arriba.

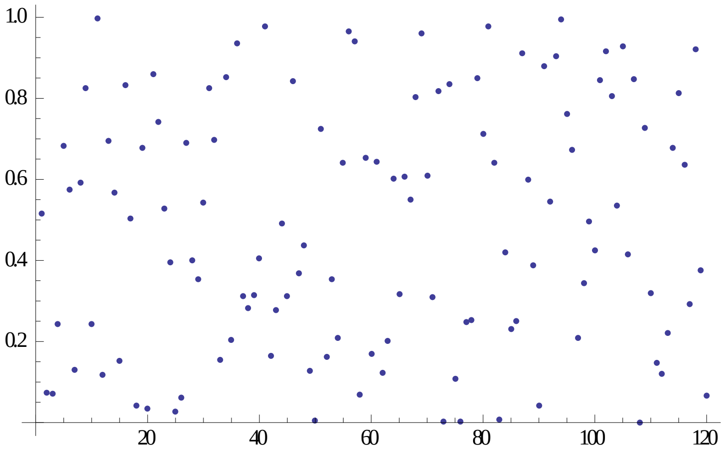
Para realizar los cálculos se han hecho tres programas, uno para cada test, en el entorno MATHEMATICA.

2.2.1 EJEMPLO DE APLICACIÓN DEL TEST 1.

Creamos una serie aleatoria de 120 elementos ($T = 120$) apoyándonos en la distribución uniforme $[0,1]$:

{0.516236, 0.0747436, 0.0721802, 0.24344, 0.682054, 0.573665, 0.13006, 0.590739, 0.824687, 0.241903, 0.997947, 0.117165, 0.69619, 0.568526, 0.151799, 0.832137, 0.504377, 0.0413851, 0.676638, 0.0334647, 0.858872, 0.742478, 0.52796, 0.396365, 0.0268669, 0.0621499, 0.689333, 0.400192, 0.353812, 0.543083, 0.825854, 0.696973, 0.154125, 0.85168, 0.204175, 0.935534, 0.311071, 0.28306, 0.313261, 0.405748, 0.976488, 0.164888, 0.277752, 0.491618, 0.312501, 0.842188, 0.369297, 0.437341, 0.128294, 0.00460613, 0.725204, 0.162605, 0.354585, 0.20953, 0.64092, 0.966238, 0.94077, 0.0693528, 0.652908, 0.169245, 0.64341, 0.121755, 0.201179, 0.602215, 0.316055, 0.606253, 0.549204, 0.802482, 0.959986, 0.608491, 0.310457, 0.819006, 0.0035578, 0.835819, 0.107415, 0.00122853, 0.248392, 0.251776, 0.849021, 0.711804, 0.976729, 0.642107, 0.00854908, 0.420478, 0.230184, 0.249827, 0.91075, 0.598971, 0.389176, 0.0413715, 0.880465, 0.546036, 0.902927, 0.995675, 0.761048, 0.672559, 0.209591, 0.344499, 0.49686, 0.425239, 0.844704, 0.91514, 0.805114, 0.534597, 0.928447, 0.415722, 0.847732, 0.000763412, 0.726174, 0.319024, 0.146648, 0.120502, 0.221036, 0.678784, 0.812145, 0.637111, 0.292414, 0.919943, 0.376175, 0.0660696}

La representación gráfica de ésta es la siguiente:



Empecemos estableciendo el valor que ha de tener la dimensión de inmersión m. Para la elección de éste se seguirá el mismo criterio que se siguió en el test recogido en Matilla y Ruíz (2008). Tomaremos el valor mayor valor de m que verifique $T \geq 5m!$. Es muy fácil deducir que el valor buscado es $m = 4$.

Así mismo tenemos que seleccionar el nivel de confianza con el que queremos aplicar el test. Vamos a tomar $nivel\ de\ confianza = 95\%$, que es el valor más habitual, y es también el valor que utilizamos en el ejemplo que expusimos sobre la aplicación del test dado en Matilla y Ruíz (2008).

Habiendo fijado ya la dimensión de inmersión $m = 4$ y $nivel\ de\ confianza = 95\%$, ya podemos proceder al estudio la serie propuesta.

Lo primero de todo es especificar cuáles son las m-historias con las que vamos a realizar nuestro análisis. En este aspecto encontramos una diferencia importante con respecto al test de Matilla y Ruíz (2008); y es que mientras en este último las m-historias se tomaban con solapamiento entre ellas, en el test que nos proponemos realizar ahora no se da dicho solapamiento. De este modo, el primero trozo de la serie es:

{**0.516236, 0.0747436, 0.0721802, 0.24344,** 0.682054, 0.573665, 0.13006, 0.590739, 0.824687, 0.241903, 0.997947, 0.117165, 0.69619, 0.568526, 0.151799, …}

pero el segundo sería:

{0.516236, 0.0747436, 0.0721802, 0.24344, **0.682054, 0.573665, 0.13006, 0.590739,** 0.824687, 0.241903, 0.997947, 0.117165, 0.69619, 0.568526, 0.151799, …}

y el tercero:

{0.516236, 0.0747436, 0.0721802, 0.24344, 0.682054, 0.573665, 0.13006, 0.590739, **0.824687, 0.241903, 0.997947, 0.117165,** 0.69619, 0.568526, 0.151799, …}

Tal y como se observa, en este caso el desplazamiento que realizamos en los elementos de la serie para tomar el trozo inmediatamente posterior a uno dado, es de m elementos. Así, tal y como comentábamos antes, las m-historias que se seleccionan van una a continuación de la otra, pero sin superponerse nunca. Este proceso se continuaría hasta llegar al último trozo de la serie:

{…, 0.415722, 0.847732, 0.000763412, 0.726174, 0.319024, 0.146648, 0.120502, 0.221036, 0.678784, 0.812145, 0.637111, **0.292414, 0.919943, 0.376175, 0.0660696**}

Hay que comentar un par de detalles en relación a esta forma de seleccionar las m-historias de la serie. En primer lugar, con esta forma de proceder se garantiza que todas las m-historias se toman de manera que no tengan elementos comunes, evitando así que tengan una relación entre ellas que venga impuesta desde el principio. En segundo lugar, con este modo de tomar las m-historias es posible que al final de la serie queden elementos sobrantes, que no sean incluidos dentro de ningún trozo; el número de elementos sobrantes es igual a $T - m[T/m]$. En nuestro ejemplo los elementos sobrantes son $T - m[T/m] = 120 - 4[120/4] = 0$; es decir, no hay ningún elemento sobrante, y la última m-historia llega hasta el último elemento de la serie; esto se debe a que el tamaño de la serie $T = 120$ es múltiplo de la dimensión de inmersión m.

Recurriendo a la notación utilizada por los autores del test, cada uno de los trozos de la serie se representa como se muestra a continuación:

$$x_4(1) = (0.516236, 0.0747436, 0.0721802, 0.24344)$$

$$x_4(5) = (0.682054, 0.573665, 0.13006, 0.590739)$$

$$x_4(9) = (0.824687, 0.241903, 0.997947, 0.117165)$$

...

$$x_4(117) = (0.292414, 0.919943, 0.376175, 0.0660696)$$

siendo cada vector $x_m(t)$ el trozo de tamaño m, cuyo primer elemento se encuentra en la posición t dentro del conjunto de elementos de la serie.

Procediendo como en el test de Matilla y Ruíz (2008), asociamos a cada vector de los anteriores un símbolo $\pi_i = (i_1, i_2, i_3, i_4) \in S_4$, siendo S_4 el grupo simétrico compuesto por todas las permutaciones de 4 elementos:

$$S_4 = \{(0,1,2,3), (0,1,3,2), (0,2,1,3), (0,2,3,1), (0,3,1,2), (0,3,2,1), \dots\}$$

La asociación entre vectores y símbolos π_i se lleva a cabo de forma que se cumplan las condiciones que siguen:

(c1) $x_{r+i_1} \leq x_{r+i_2} \leq \ \dots \ \leq x_{r+i_m}$

(c2) $i_{s-1} < i_s \ si \ x_{r+i_{s-1}} = x_{r+i_s}$

Los símbolos π_i determinan el orden de los elementos que conforman el vector en cuestión.

Veamos ahora cómo asignamos un símbolo π a cada vector $x_m(t)$. Se hace de forma análoga a como se hizo en Matilla y Ruíz (2008). Para el vector $x_4(1)$ se tendría:

$$x_4(1) = (0.516236, 0.0747436, 0.0721802, 0.24344) \Rightarrow \pi = (4, 2, 1, 3)$$

ya que $0.0721802 < 0.0747436 < 0.24344 < 0.516236$.

Para el siguiente vector, el $x_4(5)$, se tendría:

$$x_4(5) = (0.682054, 0.573665, 0.13006, 0.590739) \Rightarrow \pi = (4, 2, 1, 3)$$

Y para el $x_4(9)$:

$$x_4(9) = (0.824687, 0.241903, 0.997947, 0.117165) \Rightarrow \pi = (3, 2, 4, 1)$$

Continuaríamos la asignación de elementos π a cada vector hasta llegar al último de todos, que es:

$$x_4(117) = (0.292414, 0.919943, 0.376175, 0.0660696) \Rightarrow \pi = (2, 4, 3, 1)$$

Ya estaríamos en condiciones de determinar el estadístico de contraste, que no es otra cosa que:

$$\varpi = \# \{\pi \in S_m : p(\pi) > 0\},$$

Es decir, es el número de permutaciones admisibles diferentes que aparecen en la serie. Para calcularlo, lo único que habría que hacer sería contar el número de símbolos π distintos que han aparecido. El resultado que se obtiene para la serie analizada es:

$$\varpi = 16$$

Calculamos el valor crítico $f_{W,\alpha}$ a partir del cual aceptaríamos la hipótesis nula:

$$H_0: (X_n)_{n=1}^{T} \; es\ un\ conjunto\ de\ variables\ aleatorias\ i.i.d.$$

Dicho valor $f_{W,\alpha}$ queda determinado por el nivel de confianza con que se quiera realizar el test; y es tal que:

$$Pr\left(W < f_{W,\alpha}\right) = \alpha$$

siendo W la variable aleatoria que expresa el número de permutaciones admisibles diferentes, y α el nivel de significación del test.

La regla de decisión con un $100(1-\alpha)\%$ de nivel de confianza es que si $\varpi \geq f_{W,\alpha}$, aceptamos H_0, pero en caso contrario la rechazamos.

Recordemos que tomamos $nivel\ de\ confianza = 95\%$, y por tanto, el nivel de significación asociado es:

$$\alpha = 1 - \frac{nivel\ de\ confianza}{100} = 0.05$$

Determinamos ahora $f_{W,0.05}$ haciendo uso de la función de distribución, que vimos que era:

$$F_W(t) = Pr(W \leq t) = \begin{cases} 0, si\ t < 1 \\ \sum_{j=1}^{[t]} S_2(k,j) \frac{(m!)!}{(m!-j)!} \left(\frac{1}{m!}\right)^k, si\ 1 \leq t \leq \min(k, m!) \\ 1, si\ t > \min(k, m!) \end{cases}$$

El valor de $f_{W,0.05}$ será igual al mayor valor t que cumpla $Pr(W \geq t) \geq 1 - \alpha$. Para el caso que analizamos se obtiene:

$$f_{W,0.05} = 15$$

Es interesante notar que el nivel de confianza real no coincide con el nivel de confianza que, en un principio, se estableció ($nivel\ de\ confianza = 95\%$). Esto se debe a que la variable aleatoria W es discreta; y por tanto, la función de distribución es escalonada. Lo que se verifica es que $Pr(W \geq t) = 0.964358 \geq 0.95$; con lo que:

$$nivel\ de\ confianza\ real = 96.4358\%$$

Finalmente, y volviendo al valor calculado $f_{W,\alpha} = 15$, ya podemos obtener el resultado del test:

$$16 = \varpi \geq f_{W,\alpha} = 15 \Rightarrow la\ serie\ es\ i.i.d.$$

2.2.2 EJEMPLO DE APLICACIÓN DEL TEST 2.

Para aplicar este test tenemos que construir una serie temporal de un tamaño superior a las series que se han utilizado para explicar los test anteriores. Este test se basa en la idea de dividir la serie en N ventanas de longitud n ($n = [T/N]$), contabilizar el número de permutaciones admisibles distintas en cada ventana, y comprobar si los valores calculados presentan variaciones importantes con respecto a los resultados que cabría esperar.

Para que el test sea lo más fiable posible, el número de ventanas en que se divide la serie conviene que sea lo mayor posible.

Por otro lado, la dimensión de inmersión m con que se analiza cada ventana debe ser el máximo entero que verifique $n \geq 5m!$, siendo n el tamaño de cada ventana. Nosotros empezaremos fijando un valor para la dimensión de inmersión m; para luego tomar ventanas de tamaño $n = 5m!$.

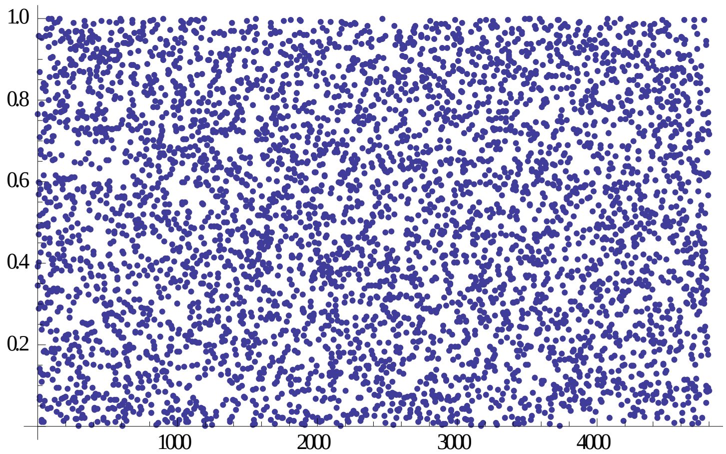
Tomaremos, como en el resto de ejemplos, $nivel\ de\ confianza = 95\%$.

Vamos a construir una serie aleatoria de 4800 elementos ($T = 4800$) haciendo uso de la distribución uniforme $[0,1]$:

{0.391329, 0.346207, 0.765142, 0.958118, 0.403017, 0.56831, 0.288214, 0.470515, 0.490695, 0.598627, 0.0660514, 0.708693, 0.86817, 0.141974, 0.518422, 0.0717768, 0.594935, 0.667948, 0.581001, 0.677925, 0.30111, 0.132326, 0.364612, 0.425986, 0.295389, 0.951955, 0.109308, 0.493838, 0.237809, 0.789479, 0.0103069, 0.0148085, 0.126044, 0.49394, 0.252, 0.355328, 0.521386, 0.0513213, 0.830602, 0.840651, 0.0119256, 0.443473, 0.181991, 0.722602, 0.614873, 0.126509, 0.766779, 0.0635287, 0.185387, 0.659375, 0.290804, 0.459822, 0.558096, 0.495229, 0.751253, 0.529824, 0.339525, 0.821455, 0.956839, 0.835869, 0.18005, 0.974637, 0.251053, 0.732077, 0.586676, 0.593878, 0.699537, 0.32436, 0.303773, 0.497513, 0.491392, 0.548404, 0.0433938, 0.125848, 0.998579, 0.589001, 0.734308, 0.4205, 0.758949, 0.930972, 0.799086, 0.857492, 0.261529, 0.666746, 0.33269, 0.575341, 0.958695, 0.450647, 0.460398, 0.231852, 0.903172, 0.37208, 0.0463616, 0.304652, 0.549733, 0.759728, 0.182292, 0.698921, 0.189198, 0.976863, 0.41034, 0.998641, 0.330633, 0.906087, 0.712444, 0.11311, 0.141531, 0.653733, 0.395747, 0.410721, 0.164713, 0.33875, 0.71253, 0.381449, 0.507374, 0.755658, 0.407988, 0.990064, 0.655737, 0.063908, 0.191224, 0.180325, 0.87981, 0.941206, 0.804474, 0.595827, 0.213677, 0.558355, 0.769795, 0.616514, 0.857968, 0.860009, 0.0947323, 0.387246, 0.539734, …

…, 0.411647, 0.895142, 0.174569, 0.747592, 0.321585, 0.104021, 0.64539, 0.103846, 0.164875, 0.193303, 0.399974, 0.472465, 0.0062715, 0.414011, 0.0412654, 0.197645, 0.307508, 0.335078, 0.293435, 0.0928239, 0.626604, 0.29705, 0.979814, 0.634375, 0.637582, 0.995911, 0.220683, 0.0219502, 0.0862746, 0.0962806, 0.446164, 0.152234, 0.879549, 0.449202, 0.622257, 0.0702316, 0.117118, 0.422731, 0.840723, 0.0731241, 0.332117, 0.653795, 0.701439, 0.962722, 0.925084, 0.274273, 0.476625, 0.861798, 0.575374, 0.00890948, 0.712678, 0.516885, 0.357271, 0.78106, 0.71716, 0.0785338, 0.357519, 0.46454, 0.351814, 0.489864, 0.490397, 0.28097, 0.60383, 0.236529, 0.795693, 0.619886, 0.454928, 0.817972, 0.857619, 0.842623, 0.529367, 0.883473, 0.54127, 0.304518, 0.340755, 0.392234, 0.174296, 0.704732, 0.0214562, 0.415967, 0.448548, 0.428061, 0.729238, 0.369747, 0.766253, 0.324357, 0.412064, 0.725315, 0.618618, 0.402118, 0.367871, 0.44743, 0.311985, 0.327786, 0.30219, 0.484419, 0.234864, 0.99675, 0.522013, 0.613503, 0.612979, 0.204264, 0.24555, 0.765689, 0.428058, 0.388015, 0.0574799, 0.0989704, 0.935522, 0.0865563, 0.945994, 0.363111, 0.188036, 0.615516, 0.672011, 0.333841, 0.823733, 0.219962, 0.672253, 0.084322, 0.0944038, 0.565673, 0.77104, 0.723925, 0.744402, 0.62001, 0.176055, 0.717167, 0.0877138}

La representación gráfica de la serie es la que sigue:



Elegimos un valor para la dimensión de inmersión $m = 4$. Siguiendo el criterio descrito más arriba, trabajaremos con ventanas de longitud $n = 5m! = 120$. De este modo, el número total de ventanas en que dividimos la serie es $N = [T/n] = [4800/120] = 40$. Además, al ser exacta la división $4800/120$, la última ventana abarcará hasta el último elemento de la serie; no perdiendo, por tanto, ningún elemento de ésta.

La primera ventana de la serie es (resaltada en negrita):

{**0.391329, 0.346207, 0.765142, 0.958118, 0.403017, 0.56831, 0.288214, 0.470515, 0.490695, 0.598627, 0.0660514, 0.708693, 0.86817, 0.141974, 0.518422, 0.0717768, 0.594935, 0.667948, 0.581001, 0.677925, 0.30111, 0.132326, 0.364612, 0.425986, 0.295389, 0.951955, 0.109308, 0.493838, 0.237809, 0.789479, 0.0103069, 0.0148085, 0.126044, 0.49394, 0.252, 0.355328, 0.521386, 0.0513213, 0.830602, 0.840651, 0.0119256, 0.443473, 0.181991, 0.722602, 0.614873, 0.126509, 0.766779, 0.0635287, 0.185387, 0.659375, 0.290804, 0.459822, 0.558096, 0.495229, 0.751253, 0.529824, 0.339525, 0.821455, 0.956839, 0.835869, 0.18005, 0.974637, 0.251053, 0.732077, 0.586676, 0.593878, 0.699537, 0.32436, 0.303773, 0.497513, 0.491392, 0.548404, 0.0433938, 0.125848, 0.998579, 0.589001, 0.734308, 0.4205, 0.758949, 0.930972, 0.799086, 0.857492, 0.261529, 0.666746, 0.33269, 0.575341, 0.958695, 0.450647, 0.460398, 0.231852, 0.903172, 0.37208, 0.0463616, 0.304652, 0.549733, 0.759728, 0.182292, 0.698921, 0.189198, 0.976863, 0.41034, 0.998641, 0.330633, 0.906087, 0.712444, 0.11311, 0.141531, 0.653733, 0.395747, 0.410721, 0.164713, 0.33875, 0.71253, 0.381449, 0.507374, 0.755658, 0.407988, 0.990064, 0.655737, 0.063908,** 0.191224, 0.180325, 0.87981, 0.941206, 0.804474, 0.595827, 0.213677, 0.558355, 0.769795, 0.616514, 0.857968, 0.860009, 0.0947323, 0.387246, 0.539734, …}

Contando a partir del último elemento de esta ventana, construiríamos la segunda ventana con los siguientes 120 elementos. Este proceso se continuaría hasta tener la serie dividida en 40 ventanas.

La última ventana que resultaría sería (resaltada en negrita):

{…, 0.411647, 0.895142, 0.174569, 0.747592, 0.321585, 0.104021, 0.64539, 0.103846, 0.164875, **0.193303, 0.399974, 0.472465, 0.0062715, 0.414011, 0.0412654, 0.197645, 0.307508, 0.335078, 0.293435, 0.0928239, 0.626604, 0.29705, 0.979814, 0.634375, 0.637582, 0.995911, 0.220683, 0.0219502, 0.0862746, 0.0962806, 0.446164, 0.152234, 0.879549, 0.449202, 0.622257, 0.0702316, 0.117118, 0.422731, 0.840723, 0.0731241, 0.332117, 0.653795, 0.701439, 0.962722, 0.925084, 0.274273, 0.476625, 0.861798, 0.575374, 0.00890948, 0.712678, 0.516885, 0.357271, 0.78106, 0.71716, 0.0785338, 0.357519, 0.46454, 0.351814, 0.489864, 0.490397, 0.28097, 0.60383, 0.236529, 0.795693, 0.619886, 0.454928, 0.817972, 0.857619, 0.842623, 0.529367, 0.883473, 0.54127, 0.304518, 0.340755, 0.392234, 0.174296, 0.704732, 0.0214562, 0.415967, 0.448548, 0.428061, 0.729238, 0.369747, 0.766253, 0.324357, 0.412064, 0.725315, 0.618618, 0.402118, 0.367871, 0.44743, 0.311985, 0.327786, 0.30219, 0.484419, 0.234864, 0.99675, 0.522013, 0.613503, 0.612979, 0.204264, 0.24555, 0.765689, 0.428058, 0.388015, 0.0574799, 0.0989704, 0.935522, 0.0865563, 0.945994, 0.363111, 0.188036, 0.615516, 0.672011, 0.333841, 0.823733, 0.219962, 0.672253, 0.084322, 0.0944038, 0.565673, 0.77104, 0.723925, 0.744402, 0.62001, 0.176055, 0.717167, 0.0877138**}

Definidas todas las ventanas, habría que realizar, para cada una de ellas, el mismo procedimiento que se explicó en el ejemplo de aplicación del test 1. Recordemos este proceso realizándolo para la primera ventana.

Especificamos cuáles son las m-historias con las que se van a realizar los cálculos en esta primera ventana. Recordemos que éstas las tomábamos sin solapamiento. La primera de ellas sería:

{**0.391329, 0.346207, 0.765142, 0.958118,** 0.403017, 0.56831, 0.288214, 0.470515, 0.490695, 0.598627, 0.0660514, 0.708693, 0.86817, 0.141974, 0.518422, …}

mientras que la segunda sería:

{0.391329, 0.346207, 0.765142, 0.958118, **0.403017, 0.56831, 0.288214, 0.470515,** 0.490695, 0.598627, 0.0660514, 0.708693, 0.86817, 0.141974, 0.518422, …}

y la tercera:

{0.391329, 0.346207, 0.765142, 0.958118, 0.403017, 0.56831, 0.288214, 0.470515, **0.490695, 0.598627, 0.0660514, 0.708693,** 0.86817, 0.141974, 0.518422, …}

Se continuaría tomando todas las m-historias hasta llegar a la última de todas, que abarcaría hasta el último elemento de la ventana:

{…, 0.11311, 0.141531, 0.653733, 0.395747, 0.410721, 0.164713, 0.33875, 0.71253, 0.381449, 0.507374, 0.755658, **0.407988, 0.990064, 0.655737, 0.063908**}

Escribimos los trozos en que hemos dividido la primera ventana en la forma habitual:

$$x_4(1) = (0.391329, 0.346207, 0.765142, 0.958118)$$

$$x_4(5) = (0.403017, 0.56831, 0.288214, 0.470515)$$

$$x_4(9) = (0.490695, 0.598627, 0.0660514, 0.708693)$$

...

$$x_4(117) = (0.407988, 0.990064, 0.655737, 0.063908)$$

siendo cada vector $x_m(t)$ el trozo de tamaño m, cuyo primer elemento se encuentra en la posición t dentro del conjunto de elementos de la serie.

A cada uno de los vectores que hemos anotado hay que asociarle un símbolo $\pi_i = (i_1, i_2, i_3, i_4) \in S_4$, siendo S_4 el conjunto de todas las posibles permutaciones de 4 elementos.

Recordemos que la asociación entre cada vector $x_m(t)$ y cada símbolo π_i se lleva a cabo cumpliendo:

(c1) $x_{r+i_1} \leq x_{r+i_2} \leq \ ... \ \leq x_{r+i_m}$

(c2) $i_{s-1} < i_s \ si \ x_{r+i_{s-1}} = x_{r+i_s}$

Realizando la citada asignación de un símbolo π a cada vector resulta:

$$x_4(1) = (0.391329, 0.346207, 0.765142, 0.958118) \Rightarrow \pi = (2, 1, 3, 4)$$

$$x_4(5) = (0.403017, 0.56831, 0.288214, 0.470515) \Rightarrow \pi = (2, 4, 1, 3)$$

$$x_4(9) = (0.490695, 0.598627, 0.0660514, 0.708693) \Rightarrow \pi = (2, 3, 1, 4)$$

...

$$x_4(117) = (0.407988, 0.990064, 0.655737, 0.063908) \Rightarrow \pi = (2, 4, 3, 1)$$

Determinamos el número de permutaciones admisibles diferentes, que se puede expresar como:

$$\varpi = \# \{\pi \in S_m : p(\pi) > 0\}$$

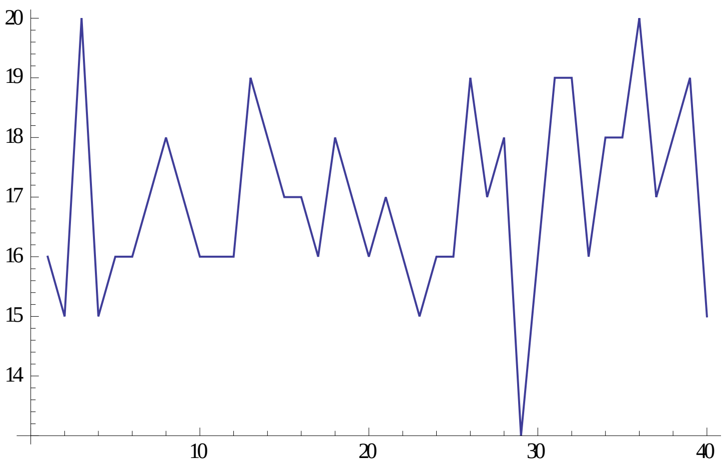
El valor obtenido para esta primera ventana resulta ser $\varpi = 16$.

Los cálculos realizados para esta ventana se tendrían que repetir para el resto de ventanas, obteniendo para cada una el número de permutaciones admisibles diferentes.

El siguiente vector recoge el valor que adopta ϖ en cada ventana:

{16, 15, 20, 15, 16, 16, 17, 18, 17, 16, 16, 16, 19, 18, 17, 17, 16, 18, 17, 16, 17, 16, 15, 16, 16, 19, 17, 18, 13, 16, 19, 19, 16, 18, 18, 20, 17, 18, 19, 15}

A continuación se da una gráfica en la que se muestra la variación de ϖ (número de permutaciones admisibles diferentes que aparecen en cada ventana):



Tal y como expusimos, bajo la hipótesis de que la secuencia $(X_n)_{n=1}^{T}$ es un conjunto de variables aleatorias i.i.d., se espera que:

$$\chi^2 = \sum_{i=1}^{\min(k,m!)} \frac{(f_i - e_i)^2}{e_i}$$

esté asintóticamente distribuido como una $\chi^2_{\min(k,m!)-1}$, siendo $e_i = N \cdot Pr(W = i)$.

Tenemos en cuenta que $\min(k, m!) = m! = 4! = 24$; y calculamos:

$$\chi^2 = \sum_{i=1}^{\min(k,m!)} \frac{(f_i - e_i)^2}{e_i} = 10.192189$$

Éste es el estadístico de contraste.

Como teníamos $nivel\ de\ confianza = 95\%$, el nivel de significación es $\alpha = 0.05$, y entonces:

$$10.192189 = \chi^2 < \chi^2_{0.05} = 35.172462 \Rightarrow la\ serie\ es\ i.i.d.$$

2.2.3 EJEMPLO DE APLICACIÓN DEL TEST 3.

Este test tiene muchas similitudes con el test 2, que acabamos de aplicar. Coincide con éste en que la serie temporal tiene que ser dividida en N ventanas de longitud n ($n = [T/N]$), y en que hay que calcular el número de permutaciones admisibles diferentes en cada una de esas ventanas.

Dado un valor m para la dimensión de inmersión, y siguiendo el mismo criterio que se siguió en el test 2, se van a tomar ventanas de longitud $n = 5m!$.

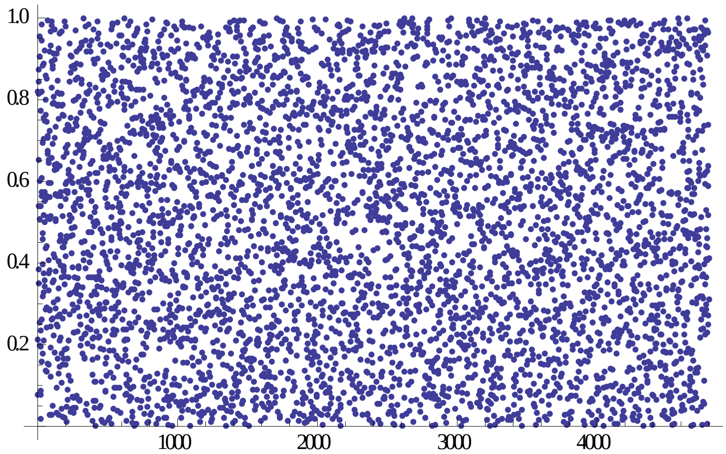
Seleccionamos $nivel\ de\ confianza = 95\%$.

Aplicamos el test a una serie aleatoria de 4800 elementos ($T = 4800$), generada según una distribución uniforme $[0,1]$:

{0.21363, 0.819726, 0.0779806, 0.844573, 0.651977, 0.54062, 0.874644, 0.815074, 0.383454, 0.350087, 0.197391, 0.600342, 0.816996, 0.955361, 0.904558, 0.25312, 0.506819, 0.0864828, 0.771856, 0.234001, 0.540293, 0.0763649, 0.0193576, 0.195128, 0.983697, 0.607517, 0.564303, 0.260674, 0.944031, 0.157754, 0.819573, 0.574813, 0.695235, 0.459343, 0.395752, 0.318164, 0.883143, 0.472985, 0.177257, 0.888162, 0.209736, 0.327145, 0.755072, 0.703972, 0.375459, 0.516574, 0.342902, 0.498973, 0.921282, 0.747452, 0.572748, 0.358219, 0.550984, 0.286994, 0.557718, 0.86009, 0.799893, 0.762295, 0.437147, 0.135808, 0.564504, 0.28999, 0.746179, 0.133144, 0.439866, 0.243079, 0.851992, 0.511476, 0.705486, 0.675433, 0.992877, 0.66657, 0.710617, 0.153593, 0.374811, 0.150721, 0.596069, 0.646186, 0.331938, 0.406798, 0.651782, 0.562724, 0.382589, 0.283023, 0.285912, 0.508447, 0.723406, 0.918297, 0.0279896, 0.193815, 0.966721, 0.122023, 0.594949, 0.932938, 0.514469, 0.245567, 0.658307, 0.262166, 0.892576, 0.992652, 0.829678, 0.980739, 0.0282824, 0.26503, 0.35342, 0.0492611, 0.883373, 0.790271, 0.875108, 0.691933, 0.35786, 0.303615, 0.222685, 0.913732, 0.808345, 0.421176, 0.792543, 0.18154, 0.705314, 0.627514, 0.413902, 0.64031, 0.972245, 0.89914, 0.566876, 0.507564, 0.73388, 0.288676, 0.380195, 0.503366, 0.568602, 0.250419, 0.788618, 0.0941075, …

…, 0.600291, 0.000557652, 0.403273, 0.811637, 0.548254, 0.177531, 0.817232, 0.344744, 0.59603, 0.109888, 0.518351, 0.0451681, 0.303997, 0.568339, 0.184631, 0.0655953, 0.97284, 0.470525, 0.702282, 0.161963, 0.866687, 0.00325605, 0.850199, 0.569376, 0.359267, 0.950938, 0.51872, 0.780748, 0.0678505, 0.583529, 0.670061, 0.934629, 0.972427, 0.131525, 0.388262, 0.458483, 0.0826598, 0.0710407, 0.712443, 0.140727, 0.362884, 0.765967, 0.300475, 0.778883, 0.480384, 0.466626, 0.313524, 0.132348, 0.814868, 0.0372638, 0.914862, 0.462721, 0.425124, 0.14489, 0.394708, 0.600996, 0.723435, 0.9061, 0.773367, 0.63102, 0.446206, 0.943986, 0.954935, 0.61836, 0.387605, 0.644849, 0.613028, 0.216638, 0.482775, 0.872675, 0.80445, 0.898978, 0.318447, 0.976976, 0.261778, 0.928371, 0.236175, 0.511164, 0.950133, 0.0477868, 0.723217, 0.368487, 0.00277175, 0.853721, 0.677112, 0.285778, 0.319031, 0.240481, 0.373015, 0.836065, 0.972638, 0.0610734, 0.919501, 0.314273, 0.615827, 0.737132, 0.397098, 0.0551862, 0.111195, 0.959052, 0.275046, 0.93284, 0.993275, 0.275591, 0.621017, 0.594111, 0.346966, 0.225328, 0.408924, 0.298344, 0.286239, 0.441409, 0.529332, 0.483422, 0.365486, 0.460801, 0.407129, 0.970086, 0.438636, 0.909266, 0.625527, 0.230799, 0.00518241, 0.517702, 0.250387, 0.965732, 0.737305, 0.588409, 0.636893, 0.27226, 0.41212, 0.310769}

La gráfica de los elementos de esta serie es:



Tomamos dimensión de inmersión $m = 4$. Según se comentó antes, el tamaño de cada ventana será $n = 5m! = 120$. El número total de ventanas resulta $N = [T/n] = [4800/120] = 40$; y no se perdería ningún elemento de la serie, puesto que la última ventana llegaría, justamente, hasta el último elemento de la serie.

La primera ventana de la serie es (resaltada en negrita):

{**0.21363, 0.819726, 0.0779806, 0.844573, 0.651977, 0.54062, 0.874644, 0.815074, 0.383454, 0.350087, 0.197391, 0.600342, 0.816996, 0.955361, 0.904558, 0.25312, 0.506819, 0.0864828, 0.771856, 0.234001, 0.540293, 0.0763649, 0.0193576, 0.195128, 0.983697, 0.607517, 0.564303, 0.260674, 0.944031, 0.157754, 0.819573, 0.574813, 0.695235, 0.459343, 0.395752, 0.318164, 0.883143, 0.472985, 0.177257, 0.888162, 0.209736, 0.327145, 0.755072, 0.703972, 0.375459, 0.516574, 0.342902, 0.498973, 0.921282, 0.747452, 0.572748, 0.358219, 0.550984, 0.286994, 0.557718, 0.86009, 0.799893, 0.762295, 0.437147, 0.135808, 0.564504, 0.28999, 0.746179, 0.133144, 0.439866, 0.243079, 0.851992, 0.511476, 0.705486, 0.675433, 0.992877, 0.66657, 0.710617, 0.153593, 0.374811, 0.150721, 0.596069, 0.646186, 0.331938, 0.406798, 0.651782, 0.562724, 0.382589, 0.283023, 0.285912, 0.508447, 0.723406, 0.918297, 0.0279896, 0.193815, 0.966721, 0.122023, 0.594949, 0.932938, 0.514469, 0.245567, 0.658307, 0.262166, 0.892576, 0.992652, 0.829678, 0.980739, 0.0282824, 0.26503, 0.35342, 0.0492611, 0.883373, 0.790271, 0.875108, 0.691933, 0.35786, 0.303615, 0.222685, 0.913732, 0.808345, 0.421176, 0.792543, 0.18154, 0.705314, 0.627514,** 0.413902, 0.64031, 0.972245, 0.89914, 0.566876, 0.507564, 0.73388, 0.288676, 0.380195, 0.503366, 0.568602, 0.250419, 0.788618, 0.0941075, …}

A continuación de esta ventana, y sin solapamiento, se definiría la siguiente. Y se procedería así de forma sucesiva hasta llega a la ventana número 40, que sería la última.

La última ventana que resultaría sería (resaltada en negrita):

{…, 0.600291, 0.000557652, 0.403273, 0.811637, 0.548254, 0.177531, 0.817232, 0.344744, 0.59603, 0.109888, 0.518351, 0.0451681**, 0.303997, 0.568339, 0.184631, 0.0655953, 0.97284, 0.470525, 0.702282, 0.161963, 0.866687, 0.00325605, 0.850199, 0.569376, 0.359267, 0.950938, 0.51872, 0.780748, 0.0678505, 0.583529, 0.670061, 0.934629, 0.972427, 0.131525, 0.388262, 0.458483, 0.0826598, 0.0710407, 0.712443, 0.140727, 0.362884, 0.765967, 0.300475, 0.778883, 0.480384, 0.466626, 0.313524, 0.132348, 0.814868, 0.0372638, 0.914862, 0.462721, 0.425124, 0.14489, 0.394708, 0.600996, 0.723435, 0.9061, 0.773367, 0.63102, 0.446206, 0.943986, 0.954935, 0.61836, 0.387605, 0.644849, 0.613028, 0.216638, 0.482775, 0.872675, 0.80445, 0.898978, 0.318447, 0.976976, 0.261778, 0.928371, 0.236175, 0.511164, 0.950133, 0.0477868, 0.723217, 0.368487, 0.00277175, 0.853721, 0.677112, 0.285778, 0.319031, 0.240481, 0.373015, 0.836065, 0.972638, 0.0610734, 0.919501, 0.314273, 0.615827, 0.737132, 0.397098, 0.0551862, 0.111195, 0.959052, 0.275046, 0.93284, 0.993275, 0.275591, 0.621017, 0.594111, 0.346966, 0.225328, 0.408924, 0.298344, 0.286239, 0.441409, 0.529332, 0.483422, 0.365486, 0.460801, 0.407129, 0.970086, 0.438636, 0.909266, 0.625527, 0.230799, 0.00518241, 0.517702, 0.250387, 0.965732, 0.737305, 0.588409, 0.636893, 0.27226, 0.41212, 0.310769**}

Ya tenemos todas las ventanas definidas. El paso que sigue es realizar un análisis de cada una de éstas para determinar el número de permutaciones admisibles de cada una. Se trata de repetir el proceso que ya realizamos en el test anterior a éste.

Vamos a estudiar la primera de las 40 ventanas. Lo que se va a realizar en esta ventana se tendría que repetir para todas las demás.

Comenzamos definiendo las m-historias en que se divide esta primera ventana:

{**0.21363, 0.819726, 0.0779806, 0.844573,** 0.651977, 0.54062, 0.874644, 0.815074, 0.383454, 0.350087, 0.197391, 0.600342, 0.816996, 0.955361, 0.904558, …}

{0.21363, 0.819726, 0.0779806, 0.844573, **0.651977, 0.54062, 0.874644, 0.815074,** 0.383454, 0.350087, 0.197391, 0.600342, 0.816996, 0.955361, 0.904558, …}

{0.21363, 0.819726, 0.0779806, 0.844573, 0.651977, 0.54062, 0.874644, 0.815074, **0.383454, 0.350087, 0.197391, 0.600342,** 0.816996, 0.955361, 0.904558, …}

Se seguirían definiendo las m-historias, como se ha realizado arriba, hasta llegar a la última, que sería:

{…, 0.413902, 0.64031, 0.972245, 0.89914, 0.566876, 0.507564, 0.73388, 0.288676, 0.380195, 0.503366, **0.568602, 0.250419, 0.788618, 0.0941075**}

Utilizamos la notación que venimos usando para expresar todas estas m-historias:

$$x_4(1) = (0.21363, 0.819726, 0.0779806, 0.844573)$$

$$x_4(5) = (0.651977, 0.54062, 0.874644, 0.815074)$$

$$x_4(9) = (0.383454, 0.350087, 0.197391, 0.600342)$$

...

$$x_4(117) = (0.568602, 0.250419, 0.788618, 0.0941075)$$

siendo $x_m(t)$ la m-historia de tamaño m, cuyo primer elemento sería el que ocupa la posición t dentro de la serie.

A los vectores anteriores le tenemos que asignar símbolos $\pi_i = (i_1, i_2, i_3, i_4) \in S_4$, donde S_4 es el conjunto simétrico de las permutaciones admisibles de tamaño 4.

Recordemos, una vez más, que la asociación entre un vector $x_m(t)$ y un símbolo π_i debe cumplir las siguientes condiciones:

(c1) $x_{r+i_1} \leq x_{r+i_2} \leq \ \ldots\ \leq x_{r+i_m}$

(c2) $i_{s-1} < i_s \ si \ x_{r+i_{s-1}} = x_{r+i_s}$

Llevando a cabo la asignación que estamos comentando, resulta:

$$x_4(1) = (0.21363, 0.819726, 0.0779806, 0.844573) \Rightarrow \pi = (2, 3, 1, 4)$$

$$x_4(5) = (0.651977, 0.54062, 0.874644, 0.815074) \Rightarrow \pi = (2, 1, 4, 3)$$

$$x_4(9) = (0.383454, 0.350087, 0.197391, 0.600342) \Rightarrow \pi = (3, 2, 1, 4)$$

...

$$x_4(117) = (0.568602, 0.250419, 0.788618, 0.0941075) \Rightarrow \pi = (3, 2, 4, 1)$$

La expresión matemática del número de permutaciones admisibles diferentes es:

$$\varpi = \# \{\pi \in S_m : p(\pi) > 0\}$$

Y resulta que, en esta primera ventana, se obtiene $\varpi = 16$.

Los cálculos realizados para esta ventana se tendrían que repetir para el resto de ventanas, determinando en cada una de ellas el valor de ϖ.

El vector que mostramos a continuación recoge el número de permutaciones admisibles diferentes que han aparecido en las 40 ventanas consideradas:

{18, 18, 18, 18, 18, 19, 18, 17, 16, 17, 19, 16, 17, 16, 16, 18, 13, 20, 18, 16, 15, 17, 17, 15, 21, 15, 15, 16, 15, 18, 16, 16, 19, 18, 18, 19, 17, 17, 19, 20}

El test que vamos a aplicar se basa en la función de distribución continua de la distribución de Kolmogorov. Para realizarlo hay que disponer de la función de distribución empírica $F_n(\varpi)$ y de la función de distribución teórica $F_W(\varpi)$.

La función de distribución empírica se define como $F_n(\varpi) = j/n$, si exactamente j de las n ventanas presentan un valor de permutaciones admisibles diferentes menor o igual que ϖ ($\varpi = 1, \ldots, \min(k, m!)$). En la serie analizada, el vector que mostramos a continuación define los valores que adopta $F_n(\varpi)$ conforme varía ϖ:

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1/40, 1/40, 3/20, 7/20, 21/40, 4/5, 37/40, 39/40, 1, 1, 1, 1}

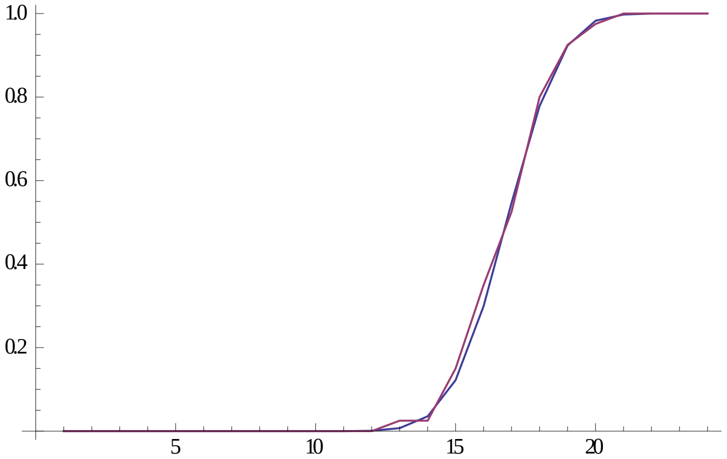
La función de distribución teórica $F_W(\varpi)$ viene dada por:

$$F_W(t) = Pr(W \leq t) = \begin{cases} 0, si\ t < 1 \\ \displaystyle\sum_{j=1}^{[t]} S_2(k,j) \frac{(m!)!}{(m!-j)!} \left(\frac{1}{m!}\right)^k, si\ 1 \leq t \leq \min(k, m!) \\ 1, si\ t > \min(k, m!) \end{cases}$$

Usando otro vector, podemos recoger los valores que toma $F_W(\varpi)$ conforme va variando ϖ, resultando:

{9.41616*10^-41, 1.16271*10^-30, 1.63495*10^-24, 4.80327*10^-20, 1.54395*10^-16, 1.13956*10^-13, 2.86454*10^-11, 3.09819*10^-9, 1.67473*10^-7, 5.00855*10^-6, 0.0000890748, 0.000993561, 0.00724087, 0.0356418, 0.122042, 0.29959, 0.54669, 0.778594, 0.923552, 0.982536, 0.997571, 0.99982, 0.999995, 1.}

Hacemos una representación gráfica conjunta de las funciones $F_n(\varpi)$ (en rojo) y $F_W(\varpi)$ (en azul), resultando:



Podemos observar que ambas funciones son muy parecidas; lo que significa que la función empírica $F_n(\varpi)$ se ajusta bastante bien a la función teórica $F_W(\varpi)$.

A partir de las funciones anteriores, interesa conocer:

$$D_n = \max|F_n(\varpi) - F_W(\varpi)|$$

Y el resultado que obtenemos para éste es $D_n = 0.050410$. El estadístico de contraste se define a partir de D_n, y es:

$$estadístico\ contraste = \sqrt{n} D_n$$

El valor obtenido para el estadístico de contraste es $\sqrt{n}D_n = 0.318820$.

Tenemos en cuenta que la función de distribución de Kolmogorov es la siguiente:

$$KS(t) = 1 - 2\sum_{i=1}^{\infty}(-1)^{i-1}e^{-2i^2t^2}$$

Y entonces se puede calcular el p-valor así:

$$p - valor = 2\sum_{i=1}^{\infty}(-1)^{i-1}e^{-2i^2\left(\sqrt{n}D_n\right)^2}$$

Obteniendo $p - valor = 0.999958$.

Con lo que, siendo α el nivel de significación, el contraste de hipótesis se puede realizar así:

$$Si\ p - valor \geq \alpha \Rightarrow la\ serie\ procede\ de\ un\ conjunto\ de\ variables\ i.i.d.$$

$$Si\ p - valor < \alpha \Rightarrow la\ serie\ no\ procede\ de\ un\ conjunto\ de\ variables\ i.i.d.$$

Y llegamos ya al resultado del test:

$$0.999958 = p - valor \geq \alpha = 0.05 \Rightarrow la\ serie\ es\ i.i.d.$$

2.3 REFERENCIAS.

Referencia principal.

- J.S. Cánovas y A. Guillamón. Permutations and time series analysis. Chaos 19 (2009), 197–235.

Otras referencias.

- J.M. Amigó, S. Zambrano y M.A.F. Sanjuán. True and false forbidden patterns in deterministic and random dynamics. Europhys. Lett. EPL 79 (2007), Art. 50001, 5pp.
- J.M. Amigó, S. Zambrano y M.A.F. Sanjuán. Combinatorial detection of determinism in noisy time series. Europhys. Lett. EPL 83 (2008), Art. 60005, 6pp.
- J.M. Amigó y M.B. Kennel. Topological permutation entropy. Phys. D 231 (2007), 137-142.
- C. Bandt y B. Pompe. Permutation entropy-a natural complexity measure for time series. Phys. Rev. Lett. 88 (2002), 174102.
- C. Bandt, G. Keller y B. Pompe. Entropy of interval maps via permutations. Nonlinearity 15 (2002), 1595-1602.
- L.S. Block y W.A. Coppel. Dynamics in one dimensión. Lecture Notes in Math. 1513, Springer, Berlin, (1992).
- R. Bowen. Entropy for group endomorphism and homogeneous spaces. Trans. Amer. Math. Soc. 153 (1971), 401-414.

- J.S. Cánovas. Estimating topological entropy from individual orbits. Int. J. Comput. Math.
- W.J. Conover. A Kolmogorov goodness-of-fit test for discontinuous distributions. J. Amer. Statist. Assoc. 67 (1972), 591-596.
- J.D. Gibbons y S. Chakrabortis. Nonparametric statistical inference. (Third Edition) Marcel Dekker, New York (1992).
- M. Hazewinkel. Encyclopaedia of Mathematics. Springer-Verlag Berlin Heidelberg New York.
- M. Matilla y M. Ruíz. A non-parametric Independence test using permutation entropy. Journal Econometrics 144 (2008), 139-155.
- G.E. Noether. Note on the Kolmogorov Statistics in the discrete case. Metrika 7 (1963), 115-116.

3. TEST BDS.

Este test fue ideado y desarrollado por Brock, Dechert y Scheinkman en 1986; de ahí que fuera denominado test BDS (Brock, Dechert, Scheinkman). Después de que en 1987 apareciera la primera publicación de este trabajo, LeBaron se sumó al grupo; éste fue el primero en desarrollar un programa informático que permitiera un cálculo viable del estadístico BDS; primero lo hizo en lenguaje Fortran, y más tarde (1997) en lenguaje C. LeBaron analizó las propiedades de algunas muestras finitas simples (Brock, Hsieh y LeBaron (1991)) y aplicó el test a series temporales financieras (Scheinkman y LeBaron (1989)). El trabajo revisado fue publicado en 1996 por Brock, Dechert, Scheinkman y LeBaron

En este apartado se va a explicar en qué consiste el test BDS. Veremos qué pasos hay que seguir para la determinación del estadístico BDS, y qué fórmulas y expresiones matemáticas hay que utilizar para ello.

3.1 DESCRIPCIÓN DEL TEST.

Daremos dos descripciones del test que son, en esencia, equivalentes. La primera descripción es la de Brock, Dechert, Scheinkman y LeBaron (1996); el interés de ésta radica en que sus autores fueron los que desarrollaron y estudiaron por vez primera el test BDS. La segunda descripción, que se debe a Kanzler (1999), se ha incluido por presentar una notación que es, en algunos aspectos, más cómoda; ésta es, además, la notación utilizada en otras muchas publicaciones.

3.1.1 DESCRIPCIÓN DEL TEST. BROCK, DECHERT, SCHEINKMAN Y LEBARON (1996).

Vamos a exponer la descripción dada en Brock, Dechert, Scheinkman y LeBaron (1996).

Definiciones y notación.

Sea $\{u_t\}$ un proceso estocástico estrictamente estacionario de variables aleatorias reales con función de distribución F. Llamemos $(u_t, u_{t+1}, \ldots, u_{t+m-1})$ a una m-historia, y denotémosla mediante u_t^m. Denotemos su función de distribución mediante F_m. Cuando $\{u_t\}$ sea independiente, entonces $F_m(x_1, \ldots, x_m) = \prod_{k=1}^{m} F(x_k)$. Designemos ς_i^j a $\sigma - \{u_i, u_{i+1}, \ldots, u_j\}$, siendo $1 \leq i < j < \infty$. El proceso estocástico $\{u_t\}$ es absolutamente regular (ver Denker y Keller (1983)) si:

$$\beta_k = \sup_{n \geq 1}\{E[sup\{|P(A|\varsigma_1^n) - P(A)| \mid A \in \varsigma_{n+k}^{\infty}\}]\} \qquad (1)$$

converge a cero.

Para $x \in \mathbb{R}^m$ vamos a usar la norma máxima $\|x\| = \max_{1\leq k\leq m}\{|x_k|\}$. Cuando sea importante especificar la dimensión de un espacio, usaremos la notación $\|\cdot\|_m$ para esta norma. La función característica del conjunto A es χ_A, y para el caso concreto en que $A = [0, \epsilon)$ denotaremos la función característica mediante χ_ϵ. Si $\phi: \mathbb{R}^m \longrightarrow \mathbb{R}^1$ es diferenciable, entonces $(D\phi)_x \in \mathbb{R}^m$ es el vector de las derivadas parciales de ϕ evaluadas en $x \in \mathbb{R}^m$. La derivada direccional de ϕ en x según la dirección v es:

$$(D\phi)_x \cdot v = \lim_{\epsilon \to 0} \frac{\phi(x + \epsilon v) - \phi(x)}{\epsilon} \quad (2)$$

La integral de correlación.

En Grassberger y Procaccia (1983) la integral de correlación fue introducida como un método para medir la dimensión fractal de un conjunto de datos determinista. Es una medida de la frecuencia con la que las pautas temporales se repiten en los datos. La integral de correlación para una dimensión de inmersión m viene dada por:

$$C_{m,n}(\epsilon) = \frac{1}{\binom{n}{2}} \sum_{s=1}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(\|u_s^m - u_t^m\|) \quad (3)$$

y

$$C_m(\epsilon) = \lim_{n \to \infty} C_{m,n}(\epsilon) \quad (4)$$

Si los datos son generados por un proceso estocástico estrictamente estacionario que es absolutamente regular, entonces el límite anterior existe. En este caso, el límite de dado en (4) es:

$$C_m(\epsilon) = \iint \chi_\epsilon(\|u - v\|)\, dF_m(u)\, dF_m(v) \quad (5)$$

(En la expresión (5) hay de hecho $2m$ variables de integración. La notación utilizada consiste en escribir un símbolo de integral por cada vector de variables en el integrando.) Cuando el proceso es independiente, y puesto que $\chi_\epsilon(\|u - v\|) = \prod_{i=1}^{m} \chi_\epsilon(u_i - v_i)$, la expresión (5) implica que:

$$C_m(\epsilon) = C_1(\epsilon)^m \quad (6)$$

Distribución asintótica de la integral de correlación.

$C_{m,n}$ es un U-estadístico generalizado (ver Serfling (1980, capítulo 5), y Denker y Keller (1983)) con núcleo simétrico $\chi_\epsilon(\|x - y\|)$. Definimos:

$$K(\epsilon) = \int \left(\int \chi_\epsilon(|u - v|)\, dF(u) \right)^2 dF(v) = \int [F(u + \epsilon) - F(u - \epsilon)]^2\, dF(u) \quad (7)$$

y sea $C(\epsilon) = C_1(\epsilon)$. Cuando el proceso $\{u_t\}$ es independiente, entonces el proceso $\{u_t^m\}$ es absolutamente regular con $\beta_k = 0$ para $k > m$, y por el Teorema 1 de Denker y Keller (1983) tenemos para el U-estadístico generalizado:

Teorema1. Sea $\{u_t\}$ i.i.d. (idénticamente distribuido). Si $K(\epsilon) > C(\epsilon)^2$, entonces:

$$\sqrt{n} \frac{C_{m,n}(\epsilon) - C_1(\epsilon)^m}{\sigma_m(\epsilon)} \quad (8)$$

converge a una distribución $N(0,1)$, donde:

$$\frac{1}{4}\sigma_m^2 = K^m - C^{2m} + 2\sum_{i=1}^{m-1}\left[K^{m-i}C^{2i} - C^{2m}\right] \quad (9)$$

(La dependencia de ϵ de los términos de la ecuación ha sido suprimida para mayor claridad en la notación.)

(La demostración está en la sección 7 de Brock, Dechert, Scheinkman y LeBaron (1996).)

Para $m = 1$, este teorema fue demostrado de forma independiente por Denker y Keller (1986). En la expresión (9), si $K(\epsilon) = C(\epsilon)^2$ entonces $\sigma_m(\epsilon) = 0$, y tenemos un caso de degeneración. Una discusión sobre las propiedades asintóticas del U-estadístico para casos de degeneración puede ser encontrada en Serfling (1980, capítulo 5). Dechert (1989) mostró que $K(\epsilon) = C(\epsilon)^2$ si y sólo si $F(u+\epsilon) - F(u-\epsilon)$ es constante para todo u. Este es el caso de una distribución uniforme circular. También es el caso de $\epsilon = 0$, además de cuando el cierre del apoyo de las variables aleatorias es compacto y ϵ es suficientemente grande. Para todos los demás casos $K(\epsilon) > C(\epsilon)^2$.

Nótese que C_1, C y K aparecen en el estadístico (8), y por tanto, cuando la función de distribución es desconocida, el estadístico no puede ser calculado. Esto lleva al estudio de las propiedades de:

$$T_{m,n}(\epsilon) = C_{m,n}(\epsilon) - C_{1,n}(\epsilon)^m \quad (10)$$

Resulta que $\lambda_1 C_{1,n}(\epsilon) + \lambda_2 C_{m,n}(\epsilon)$ es un U-estadístico generalizado con núcleo $\lambda_1\chi_\epsilon(|u_1 - v_1|) + \lambda_2\chi_\epsilon(\|u - v\|)$, para $u, v \in \mathbb{R}^m$ y para todo λ_1, λ_2. Nuevamente, por Denker y Keller (1983, Teorema 1) esta combinación lineal es asintóticamente normal, y entonces $C_{1,n}$ y $C_{m,n}$ son, en conjunto, asintóticamente normales. Ahora, por el método delta (Pollard (1984), Apéndice A) la varianza asintótica de $\sqrt{n}T_{m,n}$ es la misma que la varianza asintótica de la combinación lineal:

$$\sqrt{n}\left[C_{m,n}(\epsilon) - C_1(\epsilon)^m - mC_1(\epsilon)^{m-1}\left(C_{1,n}(\epsilon) - C_1(\epsilon)\right)\right] \quad (11)$$

Estos son los pasos clave de la prueba de:

Teorema 2. Sea $\{u_t\}$ i.i.d. Si $K(\epsilon) > C(\epsilon)^2$, entonces para $m \geq 2$:

$$\sqrt{n}\frac{T_{m,n}(\epsilon)}{V_m} \quad (12)$$

converge a una distribución $N(0,1)$. La varianza asintótica viene dada por:

$$\frac{1}{4}V_m^2 = m(m-2)C^{2m-2}(K - C^2) + K^m - C^{2m} + 2\sum_{j=1}^{m-1}\left[C^{2j}\left(K^{m-j} - C^{2m-2j}\right) - mC^{2m-2}(K - C^2)\right] \quad (13)$$

(La dependencia de ϵ de los términos de la ecuación ha sido suprimida por mayor claridad en la notación.)

Las constantes C y K en la expresión (13) para el cálculo de la varianza de $T_{m,n}(\epsilon)$ pueden ser consistentemente estimadas por:

$$C_n(\epsilon) = \frac{1}{n^2}\sum_{s=1}^{n}\sum_{t=1}^{n}\chi_\epsilon(|u_s - u_t|) \quad (14)$$

$$K_n(\epsilon) = \frac{1}{n^3}\sum_{r=1}^{n}\sum_{s=1}^{n}\sum_{t=1}^{n}\chi_\epsilon(|u_r - u_s|)\chi_\epsilon(|u_s - u_t|) \quad (15)$$

Sea $V_{m,n}^2(\epsilon)$ el valor de la varianza dada en la expresión (13), evaluada con $C_n(\epsilon)$ y $K_n(\epsilon)$ en lugar de $C(\epsilon)$ y $K(\epsilon)$. $C_n(\epsilon)$ es un V-estadístico (ver Serffling (1980, Capítulo 5)), al igual que $K_n(\epsilon)$. C_n y K_n convergen a C y K, respectivamente. De modo que $V_{m,n}$ converge a V_m, y por el Teorema de Slutsky y el Teorema 2, el estadístico BDS presentado en Brock, Dechert y Scheinkman (1987):

$$W_{m,n}(\epsilon) = \sqrt{n}\frac{T_{m,n}(\epsilon)}{V_{m,n}(\epsilon)} \quad (16)$$

converge a una distribución $N(0,1)$.

La expresión (16) es un estadístico de distribución libre. De modo que tiene la ventaja de no necesitar suposiciones sobre la distribución al usarlo como un estadístico para variables aleatorias i.i.d. Por otro lado tiene la desventaja de no ser el estadístico más potente para testar una hipótesis paramétrica. Sin embargo, presenta una potencia buena frente a varias alternativas que son comunes en economía.

3.1.2 DESCRIPCIÓN DEL TEST. KANZLER (1999).

Presentamos otra descripción del test BDS, concretamente la que realizó Kanzler (1999).

La siguiente explicación del test generaliza la dada por LeBaron (1997).

Consideremos una serie temporal x de un número infinito de observaciones siguiendo cierta distribución F:

$$x \sim F \quad (17)$$

Elijamos un tamaño arbitrario de la distancia dimensional ϵ, con la condición de que no exceda la extensión de la serie temporal (si existe):

$$0 < \epsilon < max(x) - mín(x) \quad (18)$$

Consideremos ahora la probabilidad de que un par de observaciones X_i, X_j estén separadas por una distancia inferior a ϵ:

$$P_1 \equiv P(|X_i - X_j| \leq \epsilon)\; para\; enteros\; i \neq j \quad (19)$$

Una relación similar en dimensión 2 estaría definida por dos observaciones y por sus respectivos vecinos inmediatamente anteriores. Aquí la probabilidad de 'cercanía' está definida como la probabilidad de que dos observaciones estén 'cerca', así como la de sus dos predecesores. La probabilidad de que dos historias de dos observaciones estén separadas por una distancia menor de ϵ es:

$$P_2 \equiv P\left(\left|X_i - X_j\right| \leq \epsilon, \left|X_{i-1} - X_{j-1}\right| \leq \epsilon\right) \; para\ enteros\ i \neq j \quad (20)$$

Está claro que las probabilidades para dimensión 2 son diferentes. Sin embargo si la serie temporal es i.i.d., hay una relación bien definida entre las dos: la probabilidad de que dos historias de dos observaciones estén cerca entre sí es igual al cuadrado de la probabilidad de que dos observaciones estén próximas entre sí:

$$P_2 = P_1^2 \; si\ x \sim F(i.i.d.) \quad (21)$$

La relación anterior se generaliza para otras dimensiones, y el test BDS para una dimensión de inmersión m es un test en donde la hipótesis nula establece que las probabilidades para dimensión 1 y para dimensión m son iguales:

$$H_0: P_m = P_1^m \quad H_1: P_m \neq P_1^m \quad (22)$$

Testar la hipótesis nula de arriba es equivalente a testar si la serie temporal es i.i.d., frente a todas las demás alternativas:

$$H_0: x \sim F(i.i.d.) \quad (23)$$

Para obtener el estadístico del test, la probabilidad P_m es estimada por la integral de correlación $C_{m,n}(\epsilon)$ en el espacio finito. Sea I la función de Heavside que hace que la variable $I_\epsilon(X_i, X_j)$ tome el valor 1 si las observaciones X_i y X_j están a una distancia inferior a ϵ, y tome el valor 0 en caso contrario:

$$I_\epsilon(X_i, X_j) = \begin{cases} 1, si\ \left|X_i - X_j\right| \leq \epsilon \\ 0, en\ otro\ caso \end{cases} \quad (24)$$

En una muestra de n observaciones, la integral de correlación de dimensión m es calculada como la media de todos los productos de m historias:

$$C_{m,n}(\epsilon) = \frac{2}{(n-m+1)(n-m)} \sum_{s=m}^{n} \sum_{t=s+1}^{n} \prod_{j=0}^{m-1} I_\epsilon\left(X_{s-j}, X_{t-j}\right) \quad (25)$$

Como se mostró en Brock, Dechert, Scheinkman y LeBaron (1996), el estadístico BDS para dimensión de inmersión m y distancia dimensional ϵ es estimado de forma consistente en una muestra de n observaciones por:

$$W_{m,n}(\epsilon) = \sqrt{n-m+1} \, \frac{C_{m,n}(\epsilon) - C_{1,n-m+1}(\epsilon)^m}{\sigma_{m,n}(\epsilon)} \quad (26)$$

donde la varianza estimada de $C_{m,n}(\epsilon) - C_{1,n-m+1}(\epsilon)^m$ viene dada por:

$$\sigma_{m,n}^2(\epsilon) = 4\left[K^m + 2\sum_{j=1}^{m-1} K^{m-j}C^{2j} + (m-1)^2C^{2m} - m^2KC^{2m-2}\right] \quad (27)$$

El parámetro C en la integral de correlación de dimensión 1 es:

$$C = C_{1,n}(\epsilon) \quad (28)$$

El parámetro K es la probabilidad de que una terna de observaciones se encuentren a una distancia inferior a ϵ, y se calcula:

$$K_n(\epsilon) = \frac{2}{n(n-1)(n-2)} \sum_{t=1}^{n} \sum_{s=t+1}^{n} \sum_{r=s+1}^{n} [I_\epsilon(X_t, X_s) I_\epsilon(X_s, X_r) + I_\epsilon(X_t, X_r) I_\epsilon(X_r, X_s) + I_\epsilon(X_s, X_t) I_\epsilon(X_t, X_r)] \quad (29)$$

Actualmente hay muchas formas de estimar K_n y C de forma consistente, sin embargo las expresiones mostradas son los estimadores más eficientes.

Finalmente, Brock, Dechert, Scheinkman y LeBaron (1996) muestra que el estadístico BDS converge a una distribución normal estándar:

$$\lim_{n \to \infty} W_{m,n}(\epsilon) \sim N(0,1) para\ cada\ m, \epsilon \quad (30)$$

En principio, aplicar el test BDS es sencillo: fijar el parámetro de distancia ϵ y la dimensión de inmersión m, calcular $C_{1,n}$ y K_n según las ecuaciones (25) y (29), utilizar estos estimadores para calcular $\sigma_{m,n}$ según (27), utilizar (25) para calcular $C_{m,n}$ y $C_{1,n-m+1}$, e introducir todos estos estimadores en (26) para calcular el estadístico BDS. El nivel de significación de la hipótesis nula es evaluado haciendo uso de la distribución normal.

En la práctica está lejos de ser fácil implementar las expresiones anteriores para obtener el estadístico BDS de una manera viable, en cuanto a coste computacional se refiere. En el siguiente capítulo se dará una explicación detallada sobre el modo de obtener un algoritmo que permita el cálculo eficaz del estadístico BDS.

3.2 COMENTARIOS SOBRE EL CÁLCULO DE LOS DISTINTOS PARÁMETROS.

En las dos descripciones del test BDS que han sido desarrolladas, las expresiones de cálculo presentan ciertas diferencias. Algunas se deben, simplemente, a la distinta notación utilizada en uno y otro caso. Pero otras atañen, directamente, a los valores que son calculados; no resulta lo mismo utilizando las fórmulas de Brock, Dechert, Scheinkman y LeBaron (1996), que haciendo uso de las expresiones de Kanzler (1999). Se ha de señalar también que los resultados de unas y otras no difieren en cantidades importantes. De todos modos, las fórmulas de Kanzler (1999) son preferibles, y a continuación se detallará el porqué; también se introducirán las modificaciones pertinentes en las expresiones en donde se ha estimado oportuno.

Cálculo de $C_{m,n}(\epsilon)$.

En Brock, Dechert, Scheinkman y LeBaron (1996) se estableció:

$$C_{m,n}(\epsilon) = \frac{1}{\binom{n}{2}} \sum_{s=1}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(\|u_s^m - u_t^m\|)$$

Aquí se pretende promediar los valores que toma la función característica $\chi_\epsilon(\|u_s^m - u_t^m\|)$ para cada par de m-historias u_s^m y u_t^m, siendo $1 \leq s < t \leq n$. Pero el número total de m-historias que aparecen en la serie temporal es $n-m+1$; y por tanto, el número de parejas diferentes será $\binom{n-m+1}{2}$, y no $\binom{n}{2}$. Además habría que sustituir en los sumatorios $n-1$ por $n-m$, y n por $n-m+1$. Teniendo en cuenta esto quedaría:

$$C_{m,n}(\epsilon) = \frac{1}{\binom{n-m+1}{2}} \sum_{s=1}^{n-m} \sum_{t=s+1}^{n-m+1} \chi_\epsilon(\|u_s^m - u_t^m\|)$$

Ésta última sí es una fórmula equivalente a la dada en Kanzler (1999), que es:

$$C_{m,n}(\epsilon) = \frac{2}{(n-m+1)(n-m)} \sum_{s=m}^{n} \sum_{t=s+1}^{n} \prod_{j=0}^{m-1} I_\epsilon(X_{s-j}, X_{t-j})$$

Sin embargo, sería recomendable introducir un pequeño cambio aquí también. Dado que $s < t$, el valor de s cumpliría $m \leq s \leq n-1$; y entonces se tendría:

$$C_{m,n}(\epsilon) = \frac{2}{(n-m+1)(n-m)} \sum_{s=m}^{n-1} \sum_{t=s+1}^{n} \prod_{j=0}^{m-1} I_\epsilon(X_{s-j}, X_{t-j})$$

Cálculo de $C_{1,n-m+1}(\epsilon)$.

Lo primero que hay que señalar es que en Brock, Dechert, Scheinkman y LeBaron (1996) no se habla de $C_{1,n-m+1}(\epsilon)$, sino de $C_{m,n}(\epsilon)$ particularizada para $m=1$. Lo mejor es utilizar la primera notación, $C_{1,n-m+1}(\epsilon)$, ya que el valor al que hacemos referencia no se calcula a partir de los n elementos de la serie, sino que se hace a partir de $n-m+1$ elementos de la misma; tal y como explicamos a continuación.

El test BDS se apoya en la idea de que un proceso es independiente si se cumple:

$$C_m(\epsilon) = C_1(\epsilon)^m$$

siendo $C_m(\epsilon) = \lim_{n\to\infty} C_{m,n}(\epsilon)$. Por tanto, para llevar a cabo el test hay que evaluar cuán próximos están ambos términos en la serie temporal en estudio. Para ello hay utilizar el mismo número de historias para cada uno de ellos; y como $C_{m,n}(\epsilon)$ se calcula utilizando $n-m+1$ m-historias, entonces ése será también el número de elementos que habrá que emplear para el cálculo del segundo término, que por ello denominamos $C_{1,n-m+1}(\epsilon)$, y no $C_{1,n}(\epsilon)$.

Hay otra cuestión que hay que tratar sobre el cálculo de $C_{1,n-m+1}(\epsilon)$. Se trata de que, como para su cálculo sólo se utilizan $n-m+1$ elementos, y no los n elementos de la serie completa; hay que descartar $m-1$ elementos. Está claro que los elementos de los que vamos a prescindir tendrán que ser del inicio de la serie temporal, o del final de la misma. Hay autores que hacen lo primero, pero hay otros que prefieren lo segundo. Aquí vamos a descartar los $m-1$ primeros elementos, tal y como se hace en Kanzler (1999). En cualquier caso, la diferencia entre eliminar elementos del principio o del final de la serie es pequeña; y más aún si la serie temporal en estudio es grande, y el valor de m pequeño.

Con todo lo dicho, la fórmula de cálculo de $C_{1,n-m+1}(\epsilon)$ es:

$$C_{1,n-m+1}(\epsilon) = \frac{1}{\binom{n-m+1}{2}} \sum_{s=m}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(|u_s^1 - u_t^1|)$$

Y utilizando la notación de Kanzler (1999) sería:

$$C_{1,n-m+1}(\epsilon) = \frac{2}{(n-m+1)(n-m)} \sum_{s=m}^{n-1} \sum_{t=s+1}^{n} I_\epsilon(X_s, X_t)$$

Cálculo de $\boldsymbol{T_{m,n}}(\epsilon)$.

$T_{m,n}(\epsilon)$ se define en Brock, Dechert, Scheinkman y LeBaron (1996) del modo que sigue:

$$T_{m,n}(\epsilon) = C_{m,n}(\epsilon) - C_{1,n}(\epsilon)^m$$

Introduciendo el cambio de $C_{1,n}(\epsilon)$ por $C_{1,n-m+1}(\epsilon)$, queda:

$$T_{m,n}(\epsilon) = C_{m,n}(\epsilon) - C_{1,n-m+1}(\epsilon)^m$$

En Kanzler (1999) no se define $T_{m,n}(\epsilon)$, y se habla directamente de $C_{m,n}(\epsilon) - C_{1,n-m+1}(\epsilon)^m$, que es precisamente su valor.

Cálculo de $\boldsymbol{C_n}(\epsilon)$.

Comenzamos con la expresión matemática de $C_n(\epsilon)$ presentada en Brock, Dechert, Scheinkman y LeBaron (1996):

$$C_n(\epsilon) = \frac{1}{n^2} \sum_{s=1}^{n} \sum_{t=1}^{n} \chi_\epsilon(|u_s - u_t|)$$

Las modificaciones que se van a llevar a término son similares a las hechas en la fórmula de $C_{m,n}(\epsilon)$. Es decir, en lugar de promediar la función característica $\chi_\epsilon(|u_s - u_t|)$ para $1 \leq s \leq n$ y $1 \leq t \leq n$, se hará para $1 \leq s < t \leq n$. Así, pasaremos de promediar n^2 términos a promediar $\binom{n}{2}$ términos. Este cambio se introduce para eliminar las comparaciones de un elemento de la serie consigo mismo (por ejemplo, no tendría sentido calcular $\chi_\epsilon(|u_1 - u_1|)$), y para evitar repetir comparaciones de dos elementos (por ejemplo, calcular $\chi_\epsilon(|u_1 - u_2|)$, es lo mismo que calcular $\chi_\epsilon(|u_2 - u_1|)$).

Con las modificaciones indicadas queda:

$$C_n(\epsilon) = \frac{1}{\binom{n}{2}} \sum_{s=1}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(|u_s - u_t|)$$

Quedando algo muy parecido a la fórmula de $C_{1,n-m+1}(\epsilon)$. La diferencia está en que el cálculo de $C_{1,n-m+1}(\epsilon)$ se lleva a cabo contando a partir del elemento de la serie en la posición m, mientras que el cálculo de $C_n(\epsilon)$ se realiza contando a partir del primer elemento de la serie. Nótese también que $C_n(\epsilon)$ se puede expresar fácilmente haciendo uso de la fórmula de $C_{m,n}(\epsilon)$; de hecho $C_n(\epsilon) = C_{1,n}(\epsilon)$.

Finalmente, con la notación de Kanzler (1999) escribiríamos:

$$C_n(\epsilon) = \frac{2}{n(n-1)} \sum_{s=1}^{n-1} \sum_{t=s+1}^{n} I_\epsilon(X_s, X_t)$$

Cálculo de $\boldsymbol{K_n(\epsilon)}$.

La fórmula que da Brock, Dechert, Scheinkman y LeBaron (1996) es:

$$K_n(\epsilon) = \frac{1}{n^3} \sum_{r=1}^{n} \sum_{s=1}^{n} \sum_{t=1}^{n} \chi_\epsilon(|u_r - u_s|)\chi_\epsilon(|u_s - u_t|)$$

Los cambios a introducir son análogos a los acometidos en $C_n(\epsilon)$. En este caso estamos comprando ternas de valores de la serie temporal. Con la intención de suprimir las ternas repetidas y las ternas en donde un elemento se compara consigo mismo, sustituimos los sumatorios $1 \le r \le n$, $1 \le s \le n$ y $1 \le t \le n$ por $1 \le r < s < t \le n$. De esta manera, el número de ternas sería $\binom{n}{3}$, y no n^3.

La expresión queda como se muestra a continuación:

$$K_n(\epsilon) = \frac{1}{\binom{n}{3}} \sum_{r=1}^{n-2} \sum_{s=r+1}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(|u_r - u_s|)\chi_\epsilon(|u_s - u_t|)$$

Por otro lado, en Kanzler (1999) se da:

$$K_n(\epsilon) = \frac{2}{n(n-1)(n-2)} \sum_{t=1}^{n} \sum_{s=t+1}^{n} \sum_{r=s+1}^{n} [I_\epsilon(X_t, X_s)I_\epsilon(X_s, X_r) + I_\epsilon(X_t, X_r)I_\epsilon(X_r, X_s) + I_\epsilon(X_s, X_t)I_\epsilon(X_t, X_r)]$$

Aquí se pueden hacer dos pequeños cambios con el objeto de lograr algo más de claridad. Estos consistirían en sustituir en el primer sumatorio n por $n-2$; y en el segundo sumatorio n por $n-1$. El resultado es:

$$K_n(\epsilon) = \frac{2}{n(n-1)(n-2)} \sum_{t=1}^{n-2} \sum_{s=t+1}^{n-1} \sum_{r=s+1}^{n} [I_\epsilon(X_t, X_s)I_\epsilon(X_s, X_r) + I_\epsilon(X_t, X_r)I_\epsilon(X_r, X_s) + I_\epsilon(X_s, X_t)I_\epsilon(X_t, X_r)]$$

Cálculo de $\boldsymbol{V_{m,n}(\epsilon)}$.

La desviación típica de $T_{m,n}(\epsilon)$ se denota como $V_{m,n}(\epsilon)$ en Brock, Dechert, Scheinkman y LeBaron (1996); mientras que en Kanzler (1999) se denota con $\sigma_{m,n}(\epsilon)$. De aquí en adelante adoptaremos la primera opción, es decir, representaremos la desviación típica mediante $V_{m,n}(\epsilon)$. Sin embargo, en ambas referencias, las expresiones de cálculo son totalmente equivalentes. La diferencia entre ambas está en que la fórmula dada en Kanzler (1999) está mucho más simplificada; y por tanto será más cómoda de manejar.

En Brock, Dechert, Scheinkman y LeBaron (1996), $V_{m,n}(\epsilon)$ se presenta como se muestra a continuación:

$$\frac{1}{4}V_{m,n}^2 = m(m-2)C_n^{2m-2}(K_n - C_n^2) + K_n^m - C_n^{2m} + 2\sum_{j=1}^{m-1}\left[C_n^{2j}\left(K_n^{m-j} - C_n^{2m-2j}\right) - mC_n^{2m-2}(K_n - C_n^2)\right]$$

Con la intención de conseguir una mayor claridad se ha suprimido la dependencia de ϵ de los términos de la expresión.

Ahora vamos a ver qué proceso es el que simplifica la fórmula anterior, y conduce a la fórmula que da Kanzler (1999).

Operamos dentro del sumatorio:

$$\frac{1}{4}V_{m,n}^2 = m(m-2)C_n^{2m-2}(K_n - C_n^2) + K_n^m - C_n^{2m} + 2\sum_{j=1}^{m-1}\left[C_n^{2j}K_n^{m-j} - C_n^{2m} - mC_n^{2m-2}K_n + mC_n^{2m}\right]$$

Agrupamos términos dentro del sumatorio:

$$\frac{1}{4}V_{m,n}^2 = m(m-2)C_n^{2m-2}(K_n - C_n^2) + K_n^m - C_n^{2m} + 2\sum_{j=1}^{m-1}\left[C_n^{2j}K_n^{m-j} + (m-1)C_n^{2m} - mC_n^{2m-2}K_n\right]$$

Descomponemos el sumatorio en tres nuevos sumatorios:

$$\frac{1}{4}V_{m,n}^2 = m(m-2)C_n^{2m-2}(K_n - C_n^2) + K_n^m - C_n^{2m} + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + 2\sum_{j=1}^{m-1}(m-1)C_n^{2m} - 2\sum_{j=1}^{m-1} mC_n^{2m-2}K_n$$

El segundo y tercer sumatorios son muy fáciles de calcular:

$$\frac{1}{4}V_{m,n}^2 = m(m-2)C_n^{2m-2}(K_n - C_n^2) + K_n^m - C_n^{2m} + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + 2(m-1)^2C_n^{2m} - 2(m-1)mC_n^{2m-2}K_n$$

Operamos ahora en el primer término del segundo miembro:

$$\frac{1}{4}V_{m,n}^2 = m(m-2)C_n^{2m-2}K_n - m(m-2)C_n^{2m} + K_n^m - C_n^{2m} + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + 2(m-1)^2C_n^{2m} - 2(m-1)mC_n^{2m-2}K_n$$

Agrupamos el primer término con el último teniendo en cuenta que tienen como factores comunes $mC_n^{2m-2}K_n$:

$$\frac{1}{4}V_{m,n}^2 = -m(m-2)C_n^{2m} + K_n^m - C_n^{2m} + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + 2(m-1)^2C_n^{2m} + \big(m-2-2(m-1)\big)mC_n^{2m-2}K_n$$

Simplificamos el paréntesis que ha resultado, y volvemos a operar:

$$\frac{1}{4}V_{m,n}^2 = -m(m-2)C_n^{2m} + K_n^m - C_n^{2m} + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + 2(m-1)^2C_n^{2m} - m^2C_n^{2m-2}K_n$$

Agrupamos ahora los términos que tienen como factor común C_n^{2m}:

$$\frac{1}{4}V_{m,n}^2 = K_n^m + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + [-m(m-2) - 1 + 2(m-1)^2]C_n^{2m} - m^2C_n^{2m-2}K_n$$

Operamos dentro del corchete:

$$\frac{1}{4}V_{m,n}^2 = K_n^m + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + [-m^2 + 2m - 1 + 2m^2 - 4m + 2]C_n^{2m} - m^2C_n^{2m-2}K_n$$

Tras estas últimas operaciones, simplificamos:

$$\frac{1}{4}V_{m,n}^2 = K_n^m + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + [m^2 - 2m + 1]C_n^{2m} - m^2C_n^{2m-2}K_n$$

Teniendo en cuenta que $m^2 - 2m + 1 = (m-1)^2$, queda:

$$\frac{1}{4}V_{m,n}^2 = K_n^m + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + (m-1)^2C_n^{2m} - m^2C_n^{2m-2}K_n$$

Despejamos $V_{m,n}^2$:

$$V_{m,n}^2 = 4\left[K_n^m + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + (m-1)^2C_n^{2m} - m^2C_n^{2m-2}K_n\right]$$

Y ha resultado, finalmente, la fórmula que da Kanzler (1999). Tal y como comentábamos antes, ésta es mucho más sencilla que la expresión de partida.

Para terminar, se puede despejar el valor de la desviación típica $V_{m,n}$:

$$V_{m,n} = 2\sqrt{K_n^m + 2\sum_{j=1}^{m-1} C_n^{2j}K_n^{m-j} + (m-1)^2C_n^{2m} - m^2C_n^{2m-2}K_n}$$

Cálculo de $W_{m,n}(\epsilon)$.

El estadístico de contraste se presenta en Brock, Dechert, Scheinkman y LeBaron (1996) como sigue:

$$W_{m,n}(\epsilon) = \sqrt{n}\frac{T_{m,n}(\epsilon)}{V_{m,n}(\epsilon)}$$

Pero aquí hay que volver a hacer un cambio. Hay que tener en cuenta que en el cálculo de $C_{m,n}(\epsilon)$ se utilizan $n - m + 1$ m-historias; y que el cálculo de $C_{1,n-m+1}(\epsilon)$ se basa en $n - m + 1$ elementos de la serie. En consecuencia, en la fórmula de $W_{m,n}(\epsilon)$ hay que sustituir $\sqrt{n}$ por $\sqrt{n - m + 1}$. Resulta:

$$W_{m,n}(\epsilon) = \sqrt{n - m + 1}\frac{T_{m,n}(\epsilon)}{V_{m,n}(\epsilon)}$$

En Kanzler (1999) se expone:

$$W_{m,n}(\epsilon) = \sqrt{n - m + 1}\frac{C_{m,n}(\epsilon) - C_{1,n-m+1}(\epsilon)^m}{V_{m,n}(\epsilon)}$$

Evidentemente, esta expresión es idéntica a la anterior, ya que $T_{m,n}(\epsilon) = C_{m,n}(\epsilon) - C_{1,n-m+1}(\epsilon)^m$.

3.3 RECOMENDACIONES EN LA ELECCIÓN DE LOS PARÁMETROS ϵ y m. KANZLER (1999).

Uno de los problemas que presenta el test BDS es la elección de los parámetros ϵ y m. En Kanzler (1999) podemos encontrar algunas recomendaciones en esta elección. En su trabajó realizó diversas simulaciones con el objeto de estudiar los resultados que proporcionaban distintos valores de ϵ y m. Utilizó series temporales de tamaños 50, 100, 250, 500, 750 y 1000; tomando 25000 muestras para cada uno de dichos tamaños.

El valor del parámetro ϵ, al igual que en Brock, Dechert, Scheinkman y LeBaron (1996), se expresa en función de la desviación típica de los elementos de la serie:

$$\epsilon = r\sigma$$

siendo r una constante. En Brock, Dechert, Scheinkman y LeBaron (1996) se toma $r = 0.5$ (o equivalentemente $\epsilon/\sigma = 0.5$); pero Kanzler (1999) estudia distintos valores de la relación ϵ/σ, para intentar determinar el valor más apropiado para la misma.

Las principales conclusiones que, en este aspecto, obtiene Kanzler (1999) son recogidas en M. Matilla García y J. Rodríguez Ruíz (2005), y son:

- Cuanto menor sea la dimensión de inmersión m, mejores serán las propiedades de las muestras pequeñas, cualesquiera que sean los valores del tamaño de la muestra n, y el valor de la relación ϵ/σ.
- Aumentar el tamaño muestral siempre mejora las propiedades de la muestra pequeña, cualquiera que sea la dimensión m y la relación ϵ/σ.

- La variación de la relación ϵ/σ modifica drásticamente la forma de la distribución, siendo poco evidente cuál es el valor de la relación que ofrece, en general, mejores propiedades.

Por lo que se refiere a la elección del valor de ϵ/σ, en Kanzler (1999), se concluye que $\epsilon/\sigma = 1.5$ proporciona buenos resultados para valores pequeños de la dimensión de inmersión m; sin embargo, la relación $\epsilon/\sigma = 2$ resulta más apropiada para valores de m mayores. Aunque no queda del todo claro a partir de qué valor de m es conveniente pasar de $\epsilon/\sigma = 1.5$ a $\epsilon/\sigma = 2$.

3.4 EJEMPLO ILUSTRATIVO.

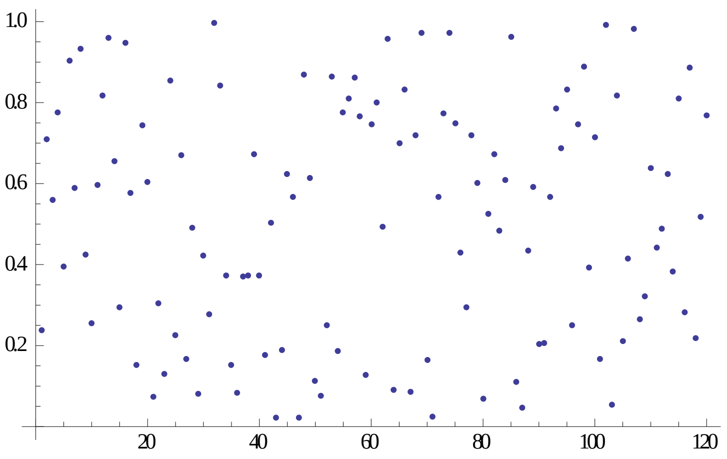
Vamos a explicar cómo se aplica el test BDS ayudándonos de un ejemplo. Nos apoyaremos en las fórmulas dadas en Brock, Dechert, Scheinkman y LeBaron (1996), introduciendo las modificaciones que fueron comentadas anteriormente. El cualquier caso, el cálculo del estadístico BDS, se puede llevar a cabo de otra forma más eficaz; esto será explicado en el siguiente capítulo.

Para llevar a cabo los cálculos utilizaremos MATHEMATICA, entorno en el que se han desarrollado dos programas para aplicar el test BDS.

Generamos una serie aleatoria de 120 elementos ($n = 120$) utilizando la distribución uniforme $[0,1]$:

{0.237727, 0.709933, 0.559636, 0.775993, 0.394606, 0.904602, 0.588895, 0.932412, 0.423695, 0.256358, 0.596915, 0.818158, 0.960795, 0.655254, 0.295724, 0.946789, 0.576478, 0.152283, 0.743392, 0.603243, 0.0745123, 0.303582, 0.130704, 0.855464, 0.225581, 0.669688, 0.166056, 0.491183, 0.0798902, 0.421622, 0.278487, 0.997649, 0.843248, 0.372919, 0.152502, 0.0830969, 0.371279, 0.37299, 0.67247, 0.372585, 0.176238, 0.50409, 0.0229472, 0.18855, 0.623284, 0.566418, 0.0213581, 0.869135, 0.612803, 0.112327, 0.0766239, 0.249931, 0.864705, 0.186896, 0.776738, 0.810325, 0.860925, 0.765948, 0.127848, 0.745497, 0.799541, 0.492358, 0.957174, 0.0918299, 0.700742, 0.832537, 0.0861771, 0.71996, 0.973287, 0.16569, 0.0233899, 0.568087, 0.773139, 0.972469, 0.747657, 0.430234, 0.294528, 0.71837, 0.601147, 0.0676988, 0.526208, 0.671854, 0.484311, 0.60858, 0.962165, 0.109861, 0.0462298, 0.433791, 0.591827, 0.203216, 0.206849, 0.568104, 0.786476, 0.6881, 0.832535, 0.250541, 0.745748, 0.889828, 0.391741, 0.715514, 0.166844, 0.991555, 0.0544439, 0.816973, 0.21092, 0.415865, 0.98247, 0.264778, 0.321162, 0.638711, 0.441141, 0.488938, 0.622725, 0.383031, 0.810381, 0.281239, 0.886714, 0.218508, 0.519281, 0.768932}

La representación gráfica de los elementos de la serie es la siguiente:



Tomamos dimensión de inmersión $m = 4$ y $nivel\ de\ confianza = 95\%$. Seleccionamos la relación $\epsilon/\sigma = 1.5$, de modo que:

$$\sigma = 0.29115 \ \Rightarrow \ \epsilon = 0.436725$$

Recordemos la expresión del estadístico de contraste $W_{m,n}(\epsilon)$:

$$W_{m,n}(\epsilon) = \sqrt{n-m+1}\frac{T_{m,n}(\epsilon)}{V_{m,n}(\epsilon)}$$

Vamos a comenzar calculando $T_{m,n}(\epsilon)$, y para ello hacemos uso de la fórmula:

$$T_{m,n}(\epsilon) = C_{m,n}(\epsilon) - C_{1,n-m+1}(\epsilon)^m$$

Para calcular $T_{m,n}(\epsilon)$ hay que calcular a su vez $C_{m,n}(\epsilon)$ y $C_{1,n-m+1}(\epsilon)$. Comencemos con $C_{m,n}(\epsilon)$:

$$C_{m,n}(\epsilon) = \frac{1}{\binom{n-m+1}{2}}\sum_{s=1}^{n-m}\sum_{t=s+1}^{n-m+1}\chi_\epsilon(\|u_s^m - u_t^m\|)$$

Y para el cálculo de $C_{m,n}(\epsilon)$ hay que dividir la serie en trozos que contengan 4 elementos; de modo que el primer trozo de la serie es (resaltado en negrita):

{**0.237727, 0.709933, 0.559636, 0.775993,** 0.394606, 0.904602, 0.588895, …}

el segundo trozo es:

{0.237727, **0.709933, 0.559636, 0.775993, 0.394606,** 0.904602, 0.588895, …}

y el tercero:

{0.237727, 0.709933, **0.559636, 0.775993, 0.394606, 0.904602,** 0.588895, …}

y así sucesivamente, hasta llegar al último trozo de la serie, que es:

{…, 0.383031, 0.810381, 0.281239, **0.886714, 0.218508, 0.519281, 0.768932**}

Los tramos se denotan tal y como sigue:

$$u_1^4 = (0.237727, 0.709933, 0.559636, 0.775993)$$

$$u_2^4 = (0.709933, 0.559636, 0.775993, 0.394606)$$

$$u_3^4 = (0.559636, 0.775993, 0.394606, 0.904602)$$

...

$$u_{117}^4 = (0.886714, 0.218508, 0.519281, 0.768932)$$

Una vez se tienen estos vectores habría que evaluar, para cada par diferente de ellos, $\chi_\epsilon(\|u_s^m - u_t^m\|)$, con $1 \leq s < t \leq n - m + 1$. Así, por ejemplo:

$$\begin{aligned}\chi_\epsilon(\|u_1^4 - u_2^4\|) &= \chi_\epsilon(\|(0.237727, 0.709933, 0.559636, 0.775993) \\ &\quad - (0.709933, 0.559636, 0.775993, 0.394606)\|) \\ &= \chi_\epsilon(\|(-0.472206, 0.150297, -0.216357, 0.381387)\|)\end{aligned}$$

y calculando el módulo de norma infinito del vector, y teniendo en cuenta que $\epsilon = 0.436725$, queda:

$$\chi_\epsilon(\|(-0.472206, 0.150297, -0.216357, 0.381387)\|) = \chi_\epsilon(0.472206) = 0$$

ya que $0.472206 > \epsilon = 0.436725$.

Este cálculo se realiza para cada par de trozos en que hemos dividido la serie para, posteriormente, promediar los resultados. Dicha media aritmética es el valor de $C_{m,n}(\epsilon)$:

$$C_{m,n}(\epsilon) = \frac{1}{\binom{n-m+1}{2}} \sum_{s=1}^{n-m} \sum_{t=s+1}^{n-m+1} \chi_\epsilon(\|u_s^m - u_t^m\|) = 0.448084$$

$C_{1,n-m+1}(\epsilon)$ se calcula de modo similar a como se ha hecho con $C_{m,n}(\epsilon)$, pero en este caso, los trozos en que se divide la serie contienen un solo elemento de la misma:

$$C_{1,n-m+1}(\epsilon) = \frac{1}{\binom{n-m+1}{2}} \sum_{s=m}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(|u_s^1 - u_t^1|)$$

Los tramos que se utilizan ahora para el cálculo son:

{0.237727, 0.709933, 0.559636, **0.775993,** 0.394606, 0.904602, 0.588895, …}

{0.237727, 0.709933, 0.559636, 0.775993, **0.394606,** 0.904602, 0.588895, …}

{0.237727, 0.709933, 0.559636, 0.775993, 0.394606, **0.904602,** 0.588895, …}

…

{…, 0.383031, 0.810381, 0.281239, 0.886714, 0.218508, 0.519281, **0.768932**}

Y los denotamos:

$$u_1^1 = 0.775993$$

$$u_2^1 = 0.394606$$

$$u_3^1 = 0.904602$$

...

$$u_{117}^1 = 0.768932$$

Con estos valores se evaluará $\chi_\epsilon(|u_s^1 - u_t^1|)$, con $m \leq s < t \leq n$. Así, por ejemplo se tendría:

$$\chi_\epsilon(|u_1^1 - u_2^1|) = \chi_\epsilon(|0.775993 - 0.394606|) = \chi_\epsilon(0.381387) = 1$$

ya que $0.381387 < \epsilon = 0.436725$.

La operación realizada se repetiría para cada pareja de trozos, y después se calcularía la media aritmética de los resultados. Se obtiene lo siguiente:

$$C_{1,n-m+1}(\epsilon) = \frac{1}{\binom{n-m+1}{2}} \sum_{s=m}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(|u_s^1 - u_t^1|) = 0.671842$$

Calculados $C_{m,n}(\epsilon)$ y $C_{1,n-m+1}(\epsilon)$, procedemos a determinar $T_{m,n}(\epsilon)$:

$$T_{m,n}(\epsilon) = C_{m,n}(\epsilon) - C_{1,n-m+1}(\epsilon)^m = -0.003287$$

Pasemos ahora a evaluar $V_{m,n}(\epsilon)$. Para ello utilizamos la fórmula dada en Brock, Dechert, Scheinkman y LeBaron (1996), que es:

$$\frac{1}{4}V_{m,n}^2 = m(m-2)C^{2m-2}(K - C^2) + K^m - C^{2m} + 2\sum_{j=1}^{m-1}\left[C^{2j}\left(K^{m-j} - C^{2m-2j}\right) - mC^{2m-2}(K - C^2)\right]$$

(La dependencia de ϵ de los términos de la ecuación ha sido suprimida para mayor claridad en la notación.)

Como ya vimos en el apartado anterior, simplificando la expresión se puede obtener la fórmula:

$$V_{m,n}^2 = 4\left[K^m + 2\sum_{j=1}^{m-1} K^{m-j}C^{2j} + (m-1)^2C^{2m} - m^2KC^{2m-2}\right]$$

Ésta es la que utiliza Kanzler (1999), y es equivalente a la primera, pero más sencilla. Podemos ver que antes de determinar $V_{m,n}(\epsilon)$ es preciso calcular previamente los valores de $C_n(\epsilon)$ y $K_n(\epsilon)$.

El cálculo de $C_n(\epsilon)$ es:

$$C_n(\epsilon) = \frac{1}{\binom{n}{2}} \sum_{s=1}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(|u_s^1 - u_t^1|)$$

La expresión es muy similar a la de $C_{1,n-m+1}(\epsilon)$. En el cálculo de $C_{1,n-m+1}(\epsilon)$ se hace un promedio de todos los valores de la función característica $\chi_\epsilon(|u_s - u_t|)$ para $m \leq s < t \leq n$; mientras que al determinar $C_n(\epsilon)$ se hace un promedio de los valores que toma la función característica $\chi_\epsilon(|u_s - u_t|)$ para $1 \leq s < t \leq n$. En el cálculo de ambos, $C_{1,n-m+1}(\epsilon)$ y $C_n(\epsilon)$, se trabaja con las parejas de elementos esencialmente distintas (no se distingue, por ejemplo, $\chi_\epsilon(|u_1 - u_2|)$ de $\chi_\epsilon(|u_2 - u_1|)$), excluyendo las parejas formadas por un elemento repetido dos veces (no se considera, por ejemplo, $\chi_\epsilon(|u_1 - u_1|)$). La diferencia está en que, mientras $C_{1,n-m+1}(\epsilon)$ solo tiene en cuenta los elementos de la serie, contando a partir del que está en la posición m; $C_n(\epsilon)$ considera absolutamente todos los elementos de la serie, contando a partir del que está en la primera posición. Esto implica que en el cálculo de $C_{1,n-m+1}(\epsilon)$ se promedian $\binom{n-m+1}{2}$ valores, pero en el cálculo de $C_n(\epsilon)$ se promedian $\binom{n}{2}$ valores.

En cualquier caso, y aunque $C_n(\epsilon)$ no es lo mismo que $C_{1,n-m+1}(\epsilon)$, podemos apoyarnos en los resultados obtenidos en el cálculo de este último para calcular $C_n(\epsilon)$. Operando se obtiene:

$$C_n(\epsilon) = \frac{1}{\binom{n}{2}} \sum_{s=1}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(|u_s^1 - u_t^1|) = 0.671569$$

La expresión de cálculo de $K_n(\epsilon)$ es:

$$K_n(\epsilon) = \frac{1}{\binom{n}{3}} \sum_{r=1}^{n-2} \sum_{s=r+1}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(|u_r - u_s|)\chi_\epsilon(|u_s - u_t|)$$

$K_n(\epsilon)$ expresa la probabilidad de que una terna de observaciones sea tal que la distancia entre dos cualesquiera de ellas sea menor de ϵ. Su determinación se puede llevar a cabo con ayuda de los cálculos que se realizaron para obtener $C_n(\epsilon)$. Sólo habría que promediar el valor de $\chi_\epsilon(|u_r - u_s|)\chi_\epsilon(|u_s - u_t|)$ para $1 \leq r < s < t \leq n$; el número de valores promediados sería $\binom{n}{3}$. El resultado es:

$$K_n(\epsilon) = \frac{1}{\binom{n}{3}} \sum_{r=1}^{n-2} \sum_{s=r+1}^{n-1} \sum_{t=s+1}^{n} \chi_\epsilon(|u_r - u_s|)\chi_\epsilon(|u_s - u_t|) = 0.462819$$

Ya calculados $C_n(\epsilon)$ y $K_n(\epsilon)$, se puede determinar $V_{m,n}(\epsilon)$:

$$V_{m,n}^2(\epsilon) = 4\left[K^m + 2\sum_{j=1}^{m-1} K^{m-j}C^{2j} + (m-1)^2C^{2m} - m^2KC^{2m-2}\right] \Rightarrow V_{m,n}(\epsilon) = 0.023629$$

Finalmente, con $T_{m,n}(\epsilon)$ y $V_{m,n}(\epsilon)$ determinamos el estadístico de contraste $W_{m,n}(\epsilon)$:

$$W_{m,n}(\epsilon) = \sqrt{n-m+1}\,\frac{T_{m,n}(\epsilon)}{V_{m,n}(\epsilon)} = -1.517454$$

Al tener $nivel\ de\ confianza = 95\%$, el intervalo de aceptación (no rechazo) de la hipótesis nula sería $(-1.959964, 1.959964)$, por tanto:

$$W_{m,n}(\epsilon) = -1.517454 \in (-1.959964, 1.959964) \Rightarrow la\ serie\ es\ i.i.d.$$

3.5 REFERENCIAS.

Referencias principales.

- W.A. Brock, W.D. Dechert, J.A. Scheinkman y B. LeBaron. A test for independence based on the correlation dimension. Econometric Reviews 15 (1996), 197-235.
- L. Kanzler. Very fast and correctly sized estimation of the BDS statistic. Department of Economics of Oxford University (1999).
- M. Matilla García y J. Rodríguez Ruíz. Aplicabilidad del test BDS al análisis de series económicas. Estudios de Economía Aplicada, vol. 23, número 002 (2005).

Otras referencias.

- W.A. Brock, W.D. Dechert y J.A. Scheinkman. A Test for Independence Based On the Correlation Dimension. Department of Economics, University of Wisconsin at Madison, University of Houston, and University of Chicago (1987).
- W.D. Dechert. An Application of Chaos Theory to Stochastic and Deterministic Observations. Department of Economics, University of Houston (1989).
- M. Denker y G. Keller. On U-Statistics and von Mises Statistics for Weakly Dependent Processes. Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete, 64 (1983), 505-522.
- M. Denker y G. Keller. Rigorous Statistical Procedures for Data from Dynamical Systems. Journal of Statistical Physics, 44 (1986), 67-93.
- P. Grassberger y I. Procaccia. Measuring the Strangeness of Strange Attractors. Physics, 9D, 189 (1983).
- D. Pollard. Convergence of Stochastic Processes. Springer-Verlag. New York (1984).
- R.J. Serfling. Approximation Theorems of Mathematical Statistics. John Wiley & Sons, New York (1980).

CAPÍTULO 3

PROGRAMACIÓN DE LOS TEST UTILIZADOS

Los test que son objeto de estudio se han programado en dos entornos de programación distintos: Matlab y Mathematica. En este capítulo comentaremos las ventajas de cada programa, teniendo en cuenta que el primero es de cálculo numérico y el segundo de cálculo simbólico.

Por otro lado hablaremos de los algoritmos utilizados para la programación de cada test. Se presta especial atención al algoritmo más rápido de los dos utilizados para la programación del test BDS.

1. PROGRAMAS DE CÁLCULO MATEMÁTICO.

Los programas de cálculo matemático son programas utilizados para realizar cálculos numéricos y/o simbólicos. Estos programas nos permiten realizar operaciones de una manera cómoda y rápida. Con ellos se puede realizar cálculos repetitivos o iterativos que serían casi impracticables con el uso de las calculadoras tradicionales. Entre los programas de cálculo matemático se distinguen dos tipos: los programas de cálculo numérico y los programas de cálculo simbólico. Como indica el propio nombre de cada uno de ellos, los primeros trabajan con ecuaciones y fórmulas numéricamente, mientras que los segundos lo hacen simbólicamente. Así, por ejemplo, un programa de cálculo simbólico interpreta la suma $x + y$ como la suma de dos variables, y no como la suma de dos números que ya tienen valores asignados.

Podemos encontrar programas especializados en cálculo numérico y en cálculo simbólico. Sin embargo, la mayoría de estos programas son capaces de trabajar de ambas formas. A continuación presentamos una lista de programas de cálculo que son utilizados en el ámbito de las matemáticas. En ésta no se incluyen todos, pero sí los más importantes y conocidos. Señalamos también el ámbito en el que son más conocidos, y el tipo de software de cada uno:

Tipo de cálculo	Programa	Tipo de software
Numérico	Matlab	Comercial
	Octave	Libre
	Scilab	Libre
Simbólico	Derive	Comercial
	Maple	Comercial
	Mathematica	Comercial
	Maxima	Libre

Tabla. Programas de cálculo matemático clasificados según el tipo de cálculo y el tipo de software.

La programación llevada a cabo en el presente trabajo se ha realizado con dos programas de cálculo matemático diferentes. Se ha recurrido a un programa de cálculo simbólico, que es Mathematica; y a un programa de cálculo numérico, que es Matlab; cada uno tiene sus ventajas, y puede cubrir posibles deficiencias del otro. Ambos programas son sobradamente conocidos en ámbitos de ingeniería y de matemáticas; y tienen múltiples funciones, como representaciones gráficas, programación estructurada y manipulación de datos.

Vamos a tratar las diferencias existentes entre los programas de cálculo simbólico y los programas de cálculo numérico. Estas diferencias se van a referir a los siguientes aspectos:

- Tratamiento de las expresiones numéricas.
- Tratamiento de las expresiones literales.
- Representación interna de la información.
- Características más representativas de ambos tipos de programas.

Tratamiento de las expresiones numéricas.

Las expresiones numéricas se forman con números o estructuras de números, y operadores que especifican los cálculos que deben ser realizados. Dichas expresiones pueden ser evaluadas por cualquier programa de cálculo numérico, y el resultado siempre será un número en formato decimal, con la precisión que se tenga especificada en ese momento. Si introducimos en Matlab la expresión *sqrt(2)*, el programa nos devuelve el valor numérico *1.4142*.

Sin embargo, los programas de cálculo simbólico devuelven las expresiones en forma simbólica; es decir, el resultado que proporcionan siempre es exacto. En cualquier caso, los programas de cálculo simbólico son capaces, también, de devolver un resultado en formato decimal, con tanta precisión como se desee. Si introducimos en Mathematica la expresión *Sqrt[2]*, el programa arroja el valor √*2*; y si queremos que el resultado sea expresado por una aproximación decimal, podemos escribir *N[Sqrt[2]]*, y en este caso se obtiene *1.4142135623730951*.

Tratamiento de las expresiones literales.

Las expresiones literales se forman con letras, números o estructuras de letras y número, y operadores que indican los cálculos que deben ser realizados. Los programas de cálculo numérico sólo son capaces de evaluar expresiones literales donde las variables tengan un valor asignado previamente. Por otro lado, los programas de cálculo simbólico pueden operar con expresiones literales sin problema, aunque las variables no tengan ningún valor asignado.

Si, por ejemplo, introducimos en Mathematica la expresión *a⁄(Sqrt[b])*, el programa nos da el resultado *a/*√*b*. Pero para que la misma expresión pueda ser evaluada en Matlab, habría que definir antes qué valores tienen las variables. Así, en Matlab podríamos definir *a=2* y *b=3*; de este modo, al introducir la expresión *a/sqrt(b)*, el resultado de salida sería *1.1547* (aproximación decimal de $\frac{2}{\sqrt{3}}$).

Las posibilidades de un programa de cálculo simbólico quedan de manifiesto si la expresión literal admite algún tipo de simplificación. Si introducimos en Mathematica la expresión (Sqrt[a])⁄a, la salida que se obtiene es *1/*√*a* (se ha realizado la simplificación $\frac{\sqrt{a}}{a} = \frac{\sqrt{a}\sqrt{a}}{a\sqrt{a}} = \frac{a}{a\sqrt{a}} = \frac{1}{\sqrt{a}}$).

Además, los programas de cálculo simbólico están preparados para presentar un mismo resultado en formatos diferentes. De esta manera, podemos simplificar una expresión lo máximo posible, o dejarla expresada en un formato estándar.

Representación interna de la información.

En Matlab todo se representa como una matriz, a excepción de otras estructuras más complejas, como las listas, que a su vez pueden contener matrices. De esta manera, se puede acceder a un elemento concreto de forma directa; mientras que si se utilizan listas, para acceder a un determinado dato hay que recorrer todos los datos que hay delante del que nos interesa. Esta mecánica favorece la rapidez en la realización de operaciones, y la consecuente reducción de tiempos de computación.

Mathematica también permite el almacenamiento matricial de datos, siempre y cuando se haya definido previamente el tamaño de la matriz.

Las características más importantes de los programas de cálculo numérico.

Los programas de cálculo numérico tienen las siguientes características:

- Utilizan la notación coma flotante, con mantisa y exponente, que permite la representación de números grandes y pequeños. Pero ésta no es la forma más exacta de representar un número.
- Los cálculos no son exactos, sino que llevan asociados un error. Por ello se suele hablar de la precisión de cálculo.
- Los cálculos pueden involucrar procesos iterativos, que llegan a su fin cuando se agota el número máximo de iteraciones, o cuando el error alcanza el valor deseado.
- Pueden encontrar solución numérica a problemas complejos que, de forma analítica no tienen solución; o problemas que sí tienen solución analítica, pero requieran un alto coste de computación para su determinación.
- Están capacitados para el análisis y almacenamiento de datos.

Las características más importantes de los programas de cálculo simbólico.

En los programas de cálculo simbólico se pueden resaltar las siguientes características comunes:

- Representan y realizan las operaciones de forma exacta, manejando expresiones simbólicas.
- Pueden realizar diversos cálculos con funciones de una o varias variables, como derivación o integración.
- Son capaces de resolver de forma analítica determinadas ecuaciones y sistemas de ecuaciones.

2. ALGORITMOS.

Vamos a describir, brevemente, los algoritmos que se han utilizado para realizar todos los cálculos asociados a la aplicación de cada test.

Los algoritmos de los test de M. Matilla y M. Ruíz, y de J.S. Cánovas y A. Guillamón, son relativamente sencillos. El algoritmo más interesante es el del test BDS; teniendo en cuenta, además, que es importante el diseño de un algoritmo eficiente para este test, ya que es el que requiere, con mucha diferencia, más tiempo de computación.

2.1 ALGORITMO DE PROGRAMACIÓN DEL TEST DE M. MATILLA Y M. RUÍZ.

Paso 1: se introducen la serie temporal a analizar, el valor de la dimensión de inmersión (m) y el valor del nivel de confianza con el que se quiere llevar a cabo el test.

Paso 2: se toma el primer tramo de la serie, de longitud m, y se identifica el tipo de permutación asociada. Esa permutación se almacena en la primera fila de una matriz. Se crea un vector en el que cada componente i almacene la frecuencia absoluta de la permutación que se almacena en la fila i de la matriz de permutaciones.

Paso 3: nos desplazamos una posición en la serie temporal, y tomamos el siguiente trozo de la serie. Determinamos la permutación asociada al mismo, y la comparamos con todas las permutaciones recogidas en la matriz de permutaciones. Si la permutación es nueva, se almacena en una nueva fila de la matriz y se añade una componente al vector de frecuencias absolutas (en la que se almacenaría el valor 1); pero si la permutación es igual a alguna de las que ya hay en la matriz, entonces no añadimos a ésta ninguna fila, y se actualiza la frecuencia absoluta de dicha permutación.

Paso 4: repetimos el paso 3 hasta llegar al elemento de la serie en la posición $n-m+1$.

Paso 5: con el vector de frecuencias absolutas determinamos un nuevo vector cuyos elementos sean las frecuencias relativas de cada permutación.

Paso 6: evaluamos la entropía de permutación a partir de todas las frecuencias relativas de las permutaciones de la serie.

Paso 7: con los valores del tamaño de la serie, la dimensión de inmersión y la entropía de permutación, calculamos el estadístico de contraste $G(m)$.

Paso 8: realizamos el contraste de hipótesis a partir del valor del estadístico de contraste y de la distribución χ^2. Lo hacemos con el nivel de confianza que se estableció en el paso 1.

2.2 ALGORITMO DE PROGRAMACIÓN DEL TEST DE J.S. CÁNOVAS Y A. GUILLAMÓN.

Describimos el algoritmo para cada uno de los tres test descritos por J.S. Cánovas y A. Guillamón. Estos tres test se basan en la misma idea; por ello se tendrán algunas similitudes en los algoritmos empleados para la aplicación de cada uno de ellos.

Test 1.

Los pasos a seguir son:

Paso 1: se introducen la serie a analizar, la dimensión de inmersión (m) y el nivel de confianza.

Paso 2: tomamos el primer trozo de la serie, y determinamos la permutación asociada al mismo. Almacenamos la permutación en la primera fila de una matriz. Introducimos un vector que almacene las frecuencias absolutas de las permutaciones recogidas en la matriz. El primer elemento del vector será 1.

Paso 3: nos desplazamos en la serie m elementos con respecto al trozo anterior. De esta manera estamos tomando el trozo inmediatamente posterior al anterior, pero sin que haya superposición con éste. Identificamos el tipo de permutación de este trozo. Si la permutación no está en la matriz de permutaciones, entonces lo incorporamos en una nueva fila, y añadimos un nuevo elemento al vector de frecuencias absolutas, cuyo valor será igual a la unidad; si en cambio la permutación ya estaba recogida en la matriz, entonces sólo habrá que actualizar la frecuencia absoluta de la permutación correspondiente.

Paso 4: repetimos el paso 3 hasta llegar al último trozo de la serie temporal.

Paso 5: el estadístico de contraste es el número de permutaciones diferentes que hemos detectado en la serie; este número también coincidirá con el número de filas de la matriz de permutaciones, y con el número de elementos del vector de frecuencias absolutas.

Paso 6: llevamos a cabo el contraste de hipótesis con la función de distribución teórica dada en el trabajo de J.S. Cánovas y A. Guillamón. Se realiza con el nivel de confianza establecido con anterioridad.

Test 2.

Paso 1: introducimos al programa la serie temporal que se desea estudiar, y se definen los valores de la dimensión de inmersión y del nivel de confianza.

Paso 2: dividimos la serie temporal en varias ventanas de tamaño igual a $5m!$. Estas ventanas no se superpondrán, y estarán una a continuación de la otra hasta llegar al final de la serie.

Paso 3: tomamos la primera ventana de la serie y llevamos a cabo los pasos 3, 4 y 5 del test anterior (Test 1). De esta manera determinamos el número de permutaciones admisibles distintas en esta primera ventana.

Paso 4: repetimos el paso 3 para cada una de las ventanas en que ha sido dividida la serie temporal. Almacenamos el número de permutaciones diferentes de cada ventana en un vector.

Paso 5: a partir del vector que guarda el número de permutaciones diferentes de cada ventana, determinamos el estadístico de contraste.

Paso 6: llevamos a cabo el contraste de hipótesis haciendo uso del valor del estadístico de contraste y del nivel de confianza que habíamos fijado previamente.

Test 3.

Paso 1: introducimos la serie temporal, la dimensión de inmersión m y el nivel de confianza con que quiere aplicarse el test.

Paso 2: aplicamos los pasos 2, 3 y 4 del test anterior (Test 2); hay que hacer notar que dentro de estos pasos se hace referencia, a su vez, a pasos dados en la aplicación del Test 1.

Paso 3: introducimos un vector con el que se representará la función de distribución empírica. En el elemento i de este vector se guardará la frecuencia relativa del número de ventanas que tienen, exactamente, i permutaciones diferentes.

Paso 4: construimos un nuevo vector, que en este caso representará la función de distribución teórica. Sus componentes recogerán las probabilidades teóricas asociadas a cada número admisible de permutaciones diferentes.

Paso 5: a partir de las distribuciones empírica y teórica determinamos el estadístico de contraste del test.

Paso 6: hacemos uso de la función de distribución de Kolmogorov, y del valor del estadístico de contraste, y determinamos el p-valor.

Paso 7: finalizamos el contraste de hipótesis comparando el p-valor obtenido con el nivel de significación asociado al nivel de confianza que se estableció.

2.3 ALGORITMO DE PROGRAMACIÓN DEL TEST BDS.

Algoritmo lento.

El algoritmo que, en primera lugar, se va a describir, es el algoritmo que implica un mayor coste computacional. Este algoritmo es muy sencillo, y se limita a la aplicación directa de fórmulas y expresiones matemáticas.

Paso 1: se introduce la serie temporal a analizar, y se establecen la dimensión de inmersión m y el nivel de confianza del test. Fijamos también el valor de la relación ε/σ.

Paso 2: calculamos $C_{m,n}(\epsilon)$ y $C_{1,n-m+1}(\epsilon)$. Con estos se determina $T_{m,n}(\epsilon)$.

Paso 3: determinamos $C_n(\epsilon)$ y $K_n(\epsilon)$. Con ambos valores se puede calcular $V_{m,n}(\epsilon)$.

Paso 4: con $T_{m,n}(\epsilon)$ y $V_{m,n}(\epsilon)$ se determina el estadístico de contraste $W_{m,n}(\epsilon)$.

Paso 5: con el estadístico de contraste y el nivel de confianza aplicamos, finalmente, el test.

Algoritmo rápido.

El algoritmo que vamos a describir ahora tiene más interés que el explicado antes. Es así por dos razones. En primer lugar porque con el algoritmo que vamos a presentar ahora se puede aplicar el test BDS de forma simultánea para distintos valores de la dimensión de inmersión m. Y en segundo lugar porque este algoritmo requiere un coste computacional menor; éste elimina, con respecto al algoritmo anterior, cálculos innecesarios, y repeticiones de operaciones.

A diferencia de cómo hemos explicado los algoritmos anteriores, para éste vamos a desarrollar un ejemplo explicativo, para una mejor presentación y comprensión del mismo.

A efectos de entender el funcionamiento del algoritmo, no importa el tamaño de la serie temporal que utilicemos; por lo tanto, por una mayor facilidad para el desarrollo de los cálculos, elegimos una serie de un tamaño muy pequeño (8 elementos):

{0.700742, 0.832537, 0.0861771, 0.71996, 0.973287, 0.16569, 0.0233899, 0.568087}

Para su análisis se utilizará una dimensión de inmersión $m = 4$ y una relación $\varepsilon/\sigma = 1{,}5$.

La desviación típica de los elementos de la serie resulta ser $\sigma = 0.365954$; y en consecuencia, el parámetro de distancia será $\varepsilon = 0.548931$.

Vamos a construir ahora una matriz de 8 filas y 8 columnas. En los elementos de esta matriz almacenaremos el resultado de:

$$I_\epsilon(X_i, X_j) = \begin{cases} 1, si \ |X_i - X_j| \leq \epsilon \\ 0, en\ otro\ caso \end{cases}$$

siendo X_i y X_j dos elementos de la serie.

La matriz resultante sería simétrica, ya que $|X_i - X_j| = |X_j - X_i|$, por lo que sólo mostraremos los elementos que queden por encima de la diagonal principal. Además, los propios elementos de la diagonal principal serán nulos, ya que $|X_i - X_i| = 0 \leq \epsilon$; razón por la cual también se omitirán los elementos de esta diagonal.

La matriz queda de la siguiente manera:

$$\begin{matrix} - & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ - & - & 0 & 1 & 1 & 0 & 0 & 1 \\ - & - & - & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & 1 & 0 & 0 & 1 \\ - & - & - & - & - & 0 & 0 & 1 \\ - & - & - & - & - & - & 1 & 1 \\ - & - & - & - & - & - & - & 1 \\ - & - & - & - & - & - & - & - \end{matrix}$$

Comenzamos calculando $C_n(\epsilon)$, que es, simplemente, el promedio de los elementos representados en la matriz anterior (no incluimos los elementos de la diagonal principal, ni los que quedarían debajo de ésta):

$$C_n(\epsilon) = \frac{17}{28} \approx 0.607143$$

Calculamos ahora $C_{1,n-m+1}(\epsilon)$. Lo hacemos promediando los elementos de la matriz anterior, a partir de la fila m (no incluimos, aquí tampoco, los elementos de la diagonal principal, ni los que quedarían debajo de ésta):

$$C_{1,n-m+1}(\epsilon) = \frac{6}{10} = \frac{3}{5} = 0.6$$

Para calcular $K_n(\epsilon)$ utilizamos también la matriz anterior. Vamos recorriendo cada uno de los elementos representados, y para cada elemento de la matriz en la fila i y la columna j que sea igual a la unidad, sumamos todos los elementos de la fila que esté en la posición j.

Así, la primera operación que haríamos para el cálculo de $K_n(\epsilon)$ sería, fijando el elemento de la matriz que está en la fila 1 y columna 2 (que es un elemento igual a la unidad, y lo hemos marcado en rojo), sumamos todos los elementos de la fila siguiente (marcada en negrita):

$$\begin{matrix} - & \color{red}{1} & 0 & 1 & 1 & 1 & 0 & 1 \\ - & - & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ - & - & - & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & 1 & 0 & 0 & 1 \\ - & - & - & - & - & 0 & 0 & 1 \\ - & - & - & - & - & - & 1 & 1 \\ - & - & - & - & - & - & - & 1 \\ - & - & - & - & - & - & - & - \end{matrix}$$

El siguiente elemento que habría que tomar ahora sería un cero. Por lo tanto, hay que saltar al siguiente, que sí es un uno. Con éste repetimos la operación descrita antes; pero al tratarse de un elemento en la columna 4, la operación mencionada se hará con la fila 4:

$$\begin{matrix} - & 1 & 0 & \color{red}{1} & 1 & 1 & 0 & 1 \\ - & - & 0 & 1 & 1 & 0 & 0 & 1 \\ - & - & - & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ - & - & - & - & - & 0 & 0 & 1 \\ - & - & - & - & - & - & 1 & 1 \\ - & - & - & - & - & - & - & 1 \\ - & - & - & - & - & - & - & - \end{matrix}$$

Repitiendo este proceso, y acumulando las sumas realizadas, sólo quedaría dividir ésa suma total entre $\binom{8}{3} = 56$ para obtener $K_n(\epsilon)$:

$$K_n(\epsilon) = \frac{16}{56} = \frac{2}{7} \approx 0.285714$$

Vamos a determinar ahora $C_{m,n}(\epsilon)$. Para ello hay que ir recorriendo todos los elementos de la misma matriz, y a partir de cada uno de ellos trazar una diagonal de longitud $m = 4$; después se multiplican los elementos de la diagonal trazada, y se promedian todos los resultados obtenidos para todas las posibles diagonales.

Para el primer elemento tendríamos lo siguiente (señalado en negrita):

$$\begin{array}{cccccccc}
- & \mathbf{1} & 0 & 1 & 1 & 1 & 0 & 1 \\
- & - & \mathbf{0} & 1 & 1 & 0 & 0 & 1 \\
- & - & - & \mathbf{0} & 0 & 1 & 1 & 1 \\
- & - & - & - & \mathbf{1} & 0 & 0 & 1 \\
- & - & - & - & - & 0 & 0 & 1 \\
- & - & - & - & - & - & 1 & 1 \\
- & - & - & - & - & - & - & 1 \\
- & - & - & - & - & - & - & -
\end{array}$$

El producto de los elementos de la diagonal señalada sería cero.

Ahora pasaríamos a trazar una siguiente diagonal, que sería:

$$\begin{array}{cccccccc}
- & 1 & \mathbf{0} & 1 & 1 & 1 & 0 & 1 \\
- & - & 0 & \mathbf{1} & 1 & 0 & 0 & 1 \\
- & - & - & 0 & \mathbf{0} & 1 & 1 & 1 \\
- & - & - & - & 1 & \mathbf{0} & 0 & 1 \\
- & - & - & - & - & 0 & 0 & 1 \\
- & - & - & - & - & - & 1 & 1 \\
- & - & - & - & - & - & - & 1 \\
- & - & - & - & - & - & - & -
\end{array}$$

Y ésta daría nuevamente un producto igual a cero.

El procedimiento se repetiría para todas las diagonales, y finalmente promediaríamos los valores obtenidos. Este resultado sería el valor buscado:

$$C_{m,n}(\epsilon) = \frac{0}{10} = 0$$

El siguiente paso es calcular con $T_{m,n}(\epsilon)$ y $V_{m,n}(\epsilon)$:

$$T_{m,n}(\epsilon) = C_{m,n}(\epsilon) - C_{1,n-m+1}(\epsilon)^m = 0 - \left(\frac{6}{10}\right)^4 = -\frac{81}{625} \approx -0.1296$$

$$V_{m,n}(\epsilon) = 2\sqrt{K_n^m + 2\sum_{j=1}^{m-1} C_n^{2j} K_n^{m-j} + (m-1)^2 C_n^{2m} - m^2 C_n^{2m-2} K_n}$$

$$= 2\sqrt{\left(\frac{2}{7}\right)^4 + 2\sum_{j=1}^{3}\left(\frac{17}{28}\right)^{2j}\left(\frac{2}{7}\right)^{4-j} + 9\left(\frac{17}{28}\right)^8 - 16\left(\frac{17}{28}\right)^6 \frac{2}{7}} = \frac{65\sqrt{1060809}}{307328}$$

$$\approx 0.217836$$

Calculamos el estadístico de contraste $W_{m,n}(\epsilon)$:

$$W_{m,n}(\epsilon) = \sqrt{n-m+1}\,\frac{T_{m,n}(\epsilon)}{V_{m,n}(\epsilon)} = -\frac{8297856\sqrt{\frac{3}{1768015}}}{8125} \approx -1.33033$$

Ya sólo quedaría realizar el contraste de hipótesis.

Según el nivel de confianza, se puede obtener un resultado u otro. Para un nivel de confianza de, por ejemplo, un 95%, se aceptaría la hipótesis nula, y se asumiría que la serie temporal es idénticamente distribuida.

CAPÍTULO 4

ANÁLISIS COMPARATIVO DE LOS TEST UTILIZADOS

Aquí vamos a aplicar los test que se presentaron en el capítulo 2 para analizar un conjunto de series temporales. La idea de este análisis es estudiar la eficacia de los test que estamos estudiando.

Realizaremos un pequeño análisis comparativo; y obtendremos algunas conclusiones a partir de los resultados arrojados por nuestro estudio.

1. SERIES TEMPORALES ESTUDIADAS.

Hemos analizado series temporales deterministas, series temporales idénticamente distribuidas y series temporales estocásticas.

Las series temporales deterministas estudiadas han sido:

- Serie $x_{n+1} = 4x_n(1 - x_n)$, condición inicial: $x_1 = 0.45$.
- Serie $x_{n+2} = 4x_n(1 - x_n)$, condiciones iniciales: $x_1 = 0.45$, $x_2 = 0.3$.

Las series temporales idénticamente distribuidas que se han analizado se presentan a continuación:

- Serie $x_n = \delta_n$, siendo $\delta_n \sim iid, U[-1,1]$.
- Serie $x_n = \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$.

Finalmente, las series temporales estocásticas estudiadas son las siguientes:

- Serie $x_{n+1} = 3.5x_n(1 - x_n) + 0.1\delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.
- Serie $x_{n+1} = 3.5x_n(1 - x_n) + 0.1\varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.
- Serie $x_{n+1} = 4x_n(1 - x_n)\delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.
- Serie $x_{n+1} = 4x_n(1 - x_n)\varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.
- Serie $x_{n+1} = 0.3x_n + \delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.
- Serie $x_{n+1} = 0.3x_n + \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.
- Serie $x_{n+1} = x_n + \delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.
- Serie $x_{n+1} = x_n + \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.

Todas las series temporales citadas más arriba han sido estudiadas para tres tamaños diferentes, que son $n = 120$, $n = 600$ y $n = 3600$. Para todos los tamaños se han utilizado tres niveles de confianza distintos: 90%, 95% y 99%.

Los valores que se han elegido para la dimensión de inmersión, según el tamaño de la serie, han sido:

- Tamaño $n = 120$:
  - Test de M. Matilla y M. Ruíz (test 1): $m = 4$.
  - Test de J.S. Cánovas y A. Guillamón (test 2): $m = 4$.
  - Test BDS (test 3): $m = 2, 3, 4, 5, 6$.
- Tamaño $n = 600$:
  - Test de M. Matilla y M. Ruíz (test 1): $m = 4, 5$.
  - Test de J.S. Cánovas y A. Guillamón (test 2): $m = 4, 5$.
  - Test BDS (test 3): $m = 2, 3, 4, 5, 6$.
- Tamaño $n = 3600$:
  - Test de M. Matilla y M. Ruíz (test 1): $m = 4, 5, 6$.
  - Test de J.S. Cánovas y A. Guillamón (test 2): $m = 4, 5, 6$.

El test BDS se ha aplicado tomando $\varepsilon/\sigma = 1{,}5$. Aunque no se ha utilizado este test para series de tamaño $n = 3600$, por tener un gran coste computacional.

2. RESULTADOS DEL ESTUDIO.

2.1 SERIES TEMPORALES DE TAMAÑO $n = 120$.

Serie $x_{n+1} = 4x_n(1 - x_n)$, condición inicial: $x_1 = 0.45$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	m=4	m=2	m=3	m=4	m=5	m=6
90%	SI	SI	SI	NO	NO	NO	NO
95%	SI	SI	SI	NO	NO	NO	NO
99%	SI	SI	SI	NO	NO	NO	NO

Tabla. Rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 1.

Serie $x_{n+2} = 4x_n(1 - x_n)$, condiciones iniciales: $x_1 = 0.45$, $x_2 = 0.3$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	m=4	m=2	m=3	m=4	m=5	m=6
90%	SI	SI	NO	SI	SI	NO	NO
95%	SI	SI	NO	SI	SI	NO	NO
99%	SI	SI	NO	SI	NO	NO	NO

Tabla. Rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 1.

Serie $x_n = \delta_n$, siendo $\delta_n \sim iid, U[-1,1]$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	m=4	m=2	m=3	m=4	m=5	m=6
90%	10,25	4	23,25	27,5	28,75	33,25	32
95%	6,25	4	16,75	17,75	22,25	23	23
99%	3,5	1,25	9,75	8	7,5	10,75	13

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

Serie $x_n = \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	m=4	m=2	m=3	m=4	m=5	m=6
90%	10,75	2,75	13,25	16,75	18	17,75	18,5
95%	5, 5	2,75	7	11,5	12	12	12,5
99%	1,5	1,25	2	2	3,25	3,5	4,25

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

Serie $x_{n+1} = 3.5x_n(1 - x_n) + 0.1\delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	**m=4**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	100	98	100	100	99,75	99,75	99,75
95%	100	98	100	100	99,75	99,75	99,5
99%	100	94,75	100	99,75	99,5	99,25	99

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

Serie $x_{n+1} = 3.5x_n(1 - x_n) + 0.1\varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	**m=4**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	100	97,5	100	100	100	100	100
95%	100	97,5	100	100	100	100	100
99%	100	96	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

Serie $x_{n+1} = 4x_n(1 - x_n)\delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	**m=4**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	100	89,5	100	100	100	100	100
95%	100	89,5	100	100	100	100	100
99%	100	76	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

Serie $x_{n+1} = 4x_n(1 - x_n)\varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	**m=4**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	100	98,25	100	100	100	100	100
95%	100	98,25	100	100	100	100	100
99%	100	89,75	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

Serie $x_{n+1} = 0.3x_n + \delta_n$, siendo $\delta_n \sim iid, U[-1, 1]$, condición inicial: $x_1 = 0.45$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	**m=4**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	28,25	4,25	52,75	47	45,75	43,25	42
95%	18,25	4,25	44,75	38,5	39,5	36,5	34
99%	9,5	0,75	32,25	27,25	26,5	25,25	23,75

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

Serie $x_{n+1} = 0.3x_n + \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0, 1)$, condición inicial: $x_1 = 0.45$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	**m=4**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	26,25	3,75	60,25	51	45,25	40,25	34,5
95%	19,75	3,75	50,5	41,75	33,25	29	23,5
99%	7,75	1	32,75	23,25	18,25	15,75	14,5

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

Serie $x_{n+1} = x_n + \delta_n$, siendo $\delta_n \sim iid, U[-1, 1]$, condición inicial: $x_1 = 0.45$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	**m=4**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	100	100	100	100	100	100	100
95%	100	100	100	100	100	100	100
99%	100	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

Serie $x_{n+1} = x_n + \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0, 1)$, condición inicial: $x_1 = 0.45$.

	TEST 1	TEST 2	TEST 3 (ε/σ=1,5)				
	m=4	**m=4**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	98,75	34	100	100	100	100	100
95%	97,25	34	100	100	100	100	100
99%	94,25	18,5	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 120. Número de repeticiones: 400.

2.2 SERIES TEMPORALES DE TAMAÑO $n = 600$.

Serie $x_{n+1} = 4x_n(1 - x_n)$, condición inicial: $x_1 = 0.45$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	m=5	m=4	m=5	m=2	m=3	m=4	m=5	m=6
90%	SI	SI	SI	SI	SI	SI	SI	SI	SI
95%	SI	SI	SI	SI	SI	SI	SI	SI	SI
99%	SI	SI	SI	SI	SI	SI	SI	SI	SI

Tabla. Rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 1.

Serie $x_{n+2} = 4x_n(1 - x_n)$, condiciones iniciales: $x_1 = 0.45$, $x_2 = 0.3$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	m=5	m=4	m=5	m=2	m=3	m=4	m=5	m=6
90%	SI	SI	SI	SI	NO	SI	NO	SI	SI
95%	SI	SI	SI	SI	NO	NO	NO	NO	SI
99%	SI	SI	SI	SI	NO	NO	NO	NO	SI

Tabla. Rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 1.

Serie $x_n = \delta_n$, siendo $\delta_n \sim iid, U[-1, 1]$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	m=5	m=4	m=5	m=2	m=3	m=4	m=5	m=6
90%	6,75	17	3,5	9,75	11,25	13	12,25	11,25	11
95%	3	9,75	3,5	2,75	7,5	6	5,75	6	6
99%	1,25	3,25	0	0,25	2,5	1,25	1,25	1	1,5

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400.

Serie $x_n = \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0, 1)$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	m=5	m=4	m=5	m=2	m=3	m=4	m=5	m=6
90%	6,25	15,5	3	6,75	9,25	11	12	11,75	12
95%	4	9,5	3	1,75	5,75	4,75	6	6,25	6,5
99%	0,75	3,25	0	0,75	0,75	1,25	1,25	1,25	1

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400

.

Serie $x_{n+1} = 3.5x_n(1 - x_n) + 0.1\delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	m=5	m=4	m=5	m=2	m=3	m=4	m=5	m=6
90%	100	100	100	100	100	100	100	100	100
95%	100	100	100	100	100	100	100	100	100
99%	100	100	100	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400.

Serie $x_{n+1} = 3.5x_n(1 - x_n) + 0.1\varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	m=5	m=4	m=5	m=2	m=3	m=4	m=5	m=6
90%	100	100	100	100	100	100	100	100	100
95%	100	100	100	100	100	100	100	100	100
99%	100	100	100	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400.

Serie $x_{n+1} = 4x_n(1 - x_n)\delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	m=5	m=4	m=5	m=2	m=3	m=4	m=5	m=6
90%	100	100	100	100	100	100	100	100	100
95%	100	100	100	100	100	100	100	100	100
99%	100	100	100	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400.

Serie $x_{n+1} = 4x_n(1 - x_n)\varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	m=5	m=4	m=5	m=2	m=3	m=4	m=5	m=6
90%	100	100	100	100	100	100	100	100	100
95%	100	100	100	100	100	100	100	100	100
99%	100	100	100	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400.

Serie $x_{n+1} = 0.3x_n + \delta_n$, siendo $\delta_n \sim iid, U[-1, 1]$, condición inicial: $x_1 = 0.45$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	**m=5**	**m=4**	**m=5**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	87,25	93,5	9	26,25	92,5	88,75	84	80,5	74,5
95%	81,75	90,25	9	10,25	89,75	83,75	78,25	74	69
99%	67,25	82,5	0,5	3,5	81,25	75	67	60,25	56

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400.

Serie $x_{n+1} = 0.3x_n + \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0, 1)$, condición inicial: $x_1 = 0.45$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	**m=5**	**m=4**	**m=5**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	85,25	87,75	9	24,5	97,5	95	91	87,5	84,25
95%	77,5	82,75	9	10,5	95,75	92,25	85,5	80,5	74,5
99%	58	69,25	0,25	3,75	89	81,25	69,5	62,5	55,25

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400.

Serie $x_{n+1} = x_n + \delta_n$, siendo $\delta_n \sim iid, U[-1, 1]$, condición inicial: $x_1 = 0.45$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	**m=5**	**m=4**	**m=5**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	100	100	100	100	100	100	100	100	100
95%	100	100	100	100	100	100	100	100	100
99%	100	100	100	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400.

Serie $x_{n+1} = x_n + \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0, 1)$, condición inicial: $x_1 = 0.45$.

	TEST 1		TEST 2		TEST 3 (ε/σ=1,5)				
	m=4	**m=5**	**m=4**	**m=5**	**m=2**	**m=3**	**m=4**	**m=5**	**m=6**
90%	100	100	35,25	99,5	100	100	100	100	100
95%	100	100	35,25	97,25	100	100	100	100	100
99%	100	100	6,75	91,5	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 600. Número de repeticiones: 400.

2.3 SERIES TEMPORALES DE TAMAÑO $n = 3600$.

Serie $x_{n+1} = 4x_n(1 - x_n)$, condición inicial: $x_1 = 0.45$.

	TEST 1			TEST 2		
	m=4	m=5	m=6	m=4	m=5	m=6
90%	SI	SI	SI	SI	SI	SI
95%	SI	SI	SI	SI	SI	SI
99%	SI	SI	SI	SI	SI	SI

Tabla. Rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 1.

Serie $x_{n+2} = 4x_n(1 - x_n)$, condiciones iniciales: $x_1 = 0.45$, $x_2 = 0.3$.

	TEST 1			TEST 2		
	m=4	m=5	m=6	m=4	m=5	m=6
90%	SI	SI	SI	SI	SI	SI
95%	SI	SI	SI	SI	SI	SI
99%	SI	SI	SI	SI	SI	SI

Tabla. Rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 1.

Serie $x_n = \delta_n$, siendo $\delta_n \sim iid, U[-1, 1]$.

	TEST 1			TEST 2		
	m=4	m=5	m=6	m=4	m=5	m=6
90%	9,75	12,25	32,25	0	4	6,25
95%	5,5	8,25	23	0	4	2,25
99%	2	3,75	11	0	0,75	0,5

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

Serie $x_n = \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0, 1)$.

	TEST 1			TEST 2		
	m=4	m=5	m=6	m=4	m=5	m=6
90%	11,5	13,75	34,75	0	2	8
95%	6	7,25	21,25	0	2	4,5
99%	1,25	2,75	10,5	0	0	0,75

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

Serie $x_{n+1} = 3.5x_n(1 - x_n) + 0.1\delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.

	TEST 1			TEST 2		
	m=4	**m=5**	**m=6**	**m=4**	**m=5**	**m=6**
90%	100	100	100	87	100	100
95%	100	100	100	87	100	100
99%	100	100	100	87	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

Serie $x_{n+1} = 3.5x_n(1 - x_n) + 0.1\varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.

	TEST 1			TEST 2		
	m=4	**m=5**	**m=6**	**m=4**	**m=5**	**m=6**
90%	100	100	100	100	100	100
95%	100	100	100	100	100	100
99%	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

Serie $x_{n+1} = 4x_n(1 - x_n)\delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.

	TEST 1			TEST 2		
	m=4	**m=5**	**m=6**	**m=4**	**m=5**	**m=6**
90%	100	100	100	100	100	100
95%	100	100	100	100	100	100
99%	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

Serie $x_{n+1} = 4x_n(1 - x_n)\varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.

	TEST 1			TEST 2		
	m=4	**m=5**	**m=6**	**m=4**	**m=5**	**m=6**
90%	100	100	100	100	100	100
95%	100	100	100	100	100	100
99%	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

Serie $x_{n+1} = 0.3x_n + \delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.

	TEST 1			TEST 2		
	m=4	**m=5**	**m=6**	**m=4**	**m=5**	**m=6**
90%	100	100	100	0	25,75	88,5
95%	100	100	100	0	25,75	76,5
99%	100	100	100	0	6,5	56

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

Serie $x_{n+1} = 0.3x_n + \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.

	TEST 1			TEST 2		
	m=4	**m=5**	**m=6**	**m=4**	**m=5**	**m=6**
90%	100	100	100	0	22,25	72,75
95%	100	100	100	0	22,25	57,25
99%	100	100	100	0	7	35,25

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

Serie $x_{n+1} = x_n + \delta_n$, siendo $\delta_n \sim iid, U[-1,1]$, condición inicial: $x_1 = 0.45$.

	TEST 1			TEST 2		
	m=4	**m=5**	**m=6**	**m=4**	**m=5**	**m=6**
90%	100	100	100	100	100	100
95%	100	100	100	100	100	100
99%	100	100	100	100	100	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

Serie $x_{n+1} = x_n + \varepsilon_n$, siendo $\varepsilon_n \sim iid, N(0,1)$, condición inicial: $x_1 = 0.45$.

	TEST 1			TEST 2		
	m=4	**m=5**	**m=6**	**m=4**	**m=5**	**m=6**
90%	100	100	100	0	99,75	100
95%	100	100	100	0	99,75	100
99%	100	100	100	0	97,25	100

Tabla. Porcentajes de rechazo de la hipótesis nula. Tamaño de la serie: 3600. Número de repeticiones: 400.

3. CONCLUSIONES DEL ANÁLISIS COMPARATIVO.

Antes de ir a las distintas conclusiones obtenidas, recordemos que al test de M. Matilla y M. Ruíz lo hemos denominado test 1, al test de J.S. Cánovas y A. Guillamón lo hemos llamado test 2, y al test BDS lo hemos denominado test 3.

Del análisis comparativo llevado a cabo se pueden destacar las siguientes conclusiones:

- Los test 1 y 2 funcionan mejor que el test 3 en el análisis de series temporales deterministas de tamaños $n = 120$ y $n = 600$. Ambos son igual de eficaces para los tres tamaños analizados, incluido el tamaño $n = 3600$.
- El test que mejor detecta una serie idénticamente distribuida es el test 2, seguido del test 1. El test 3 es el que peor funciona en estos casos.
- En nuestro estudio, para una serie temporal estocástica, y para tamaños $n = 120$ y $n = 600$, el test que mejor funciona es el test 3, seguido del test 1.Para el tamaño $n = 3600$, entre los test 1 y 2, el mejor es el test 1.

Se deben recalcar varios aspectos en relación con las conclusiones expuestas. En primer lugar, hay que destacar que los análisis se han llevado a cabo para series temporales de tres tamaños diferentes; por lo tanto, hay que ser cauteloso a la hora de extrapolar dichas conclusiones para series temporales de otros diferentes. En segundo lugar, también hay que señalar que el test 2 se ha aplicado considerando que la dimensión de inmersión m debe tomarse igual al mayor entero que verifique la inecuación $5m! \leq n$; pero es posible que dicha regla, que es la adecuada en la aplicación del test 1, no sea tan adecuada para este test.

CAPÍTULO 5

PRESENTACIÓN DE LOS RESULTADOS: INTERFAZ GRÁFICA (STEMP 1.0)

En este último capítulo mostramos un entorno amigable, diseñado en Matlab, que permite aplicar los test presentados en el capítulo 2. A este entorno lo hemos denominado STemp 1.0.

Concretamente, Stemp 1.0 permite visualizar la representación gráfica de la serie temporal, así como listar los elementos de dicha serie; también permite analizar la dependencia de la serie según los test de Matilla y Ruíz, de Cánovas y Guillamón, y de Brock, Dechert, Scheinkman y LeBaron.

1. FUNCIONAMIENTO DE STEMP 1.0.

Para trabajar con la aplicación STemp 1.0, hay que abrirla desde la consola de Matlab, donde hay que escribir *Stemp*.

Una vez que la aplicación se ha puesto en funcionamiento, tenemos acceso a un pequeño menú, en el que se pueden seleccionar las opciones *Ver* y *Analizar*.

En la opción *Ver* tenemos, básicamente, dos funciones:

- Ver una lista de los elementos que componen la serie temporal que deseamos estudiar.
- Ver la representación gráfica de la serie.

Por otro lado, en la opción *Analizar*, tenemos acceso a:

- *Mets. basados en permutaciones*:
    - *Matilla y Ruíz*.
    - *Cánovas y Guillamón*.
- *Met. Brock y otros*.

Es decir, con la opción *Analizar* podemos acceder a tres ventanas diferentes, en las que se estudia la dependencia de la serie temporal según los tres métodos citados.

2. DESCRIPCIÓN DE STEMP 1.0.

Vamos a ir mostrando las distintas ventanas que forman parte de Stemp 1.0.; y vamos a ir explicando el funcionamiento de cada una de ellas.

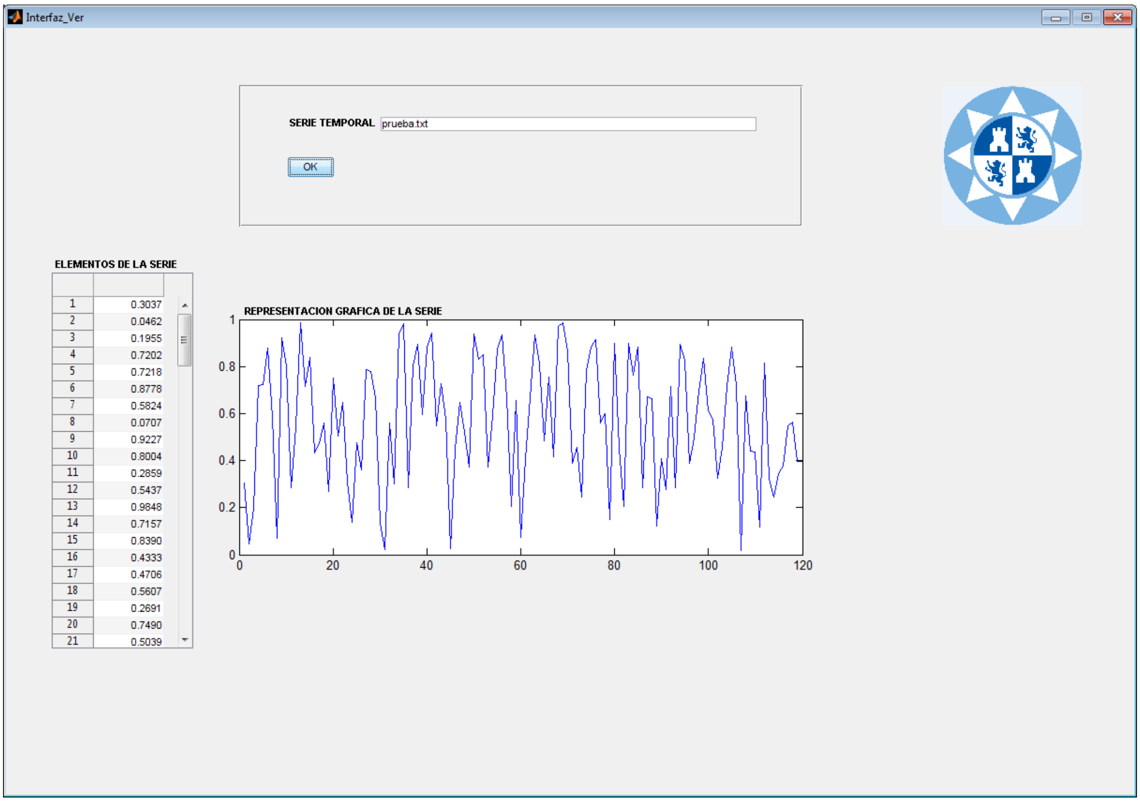
Cuando iniciamos la aplicación, se abre la siguiente ventana:

Se trata de una ventana de presentación, en la que se muestra el nombre de la aplicación.

En la parte superior izquierda tenemos un menú, en el que tenemos acceso a las otras cuatro ventanas de la aplicación. Como ya describimos anteriormente, en la opción *Ver* podemos visualizar la gráfica y los elementos de la serie; y en la opción *Analizar* se puede estudiar la dependencia de la serie según tres métodos distintos.

Pasamos ahora a la descripción de las ventanas a las que se accede desde el menú de la ventana principal.

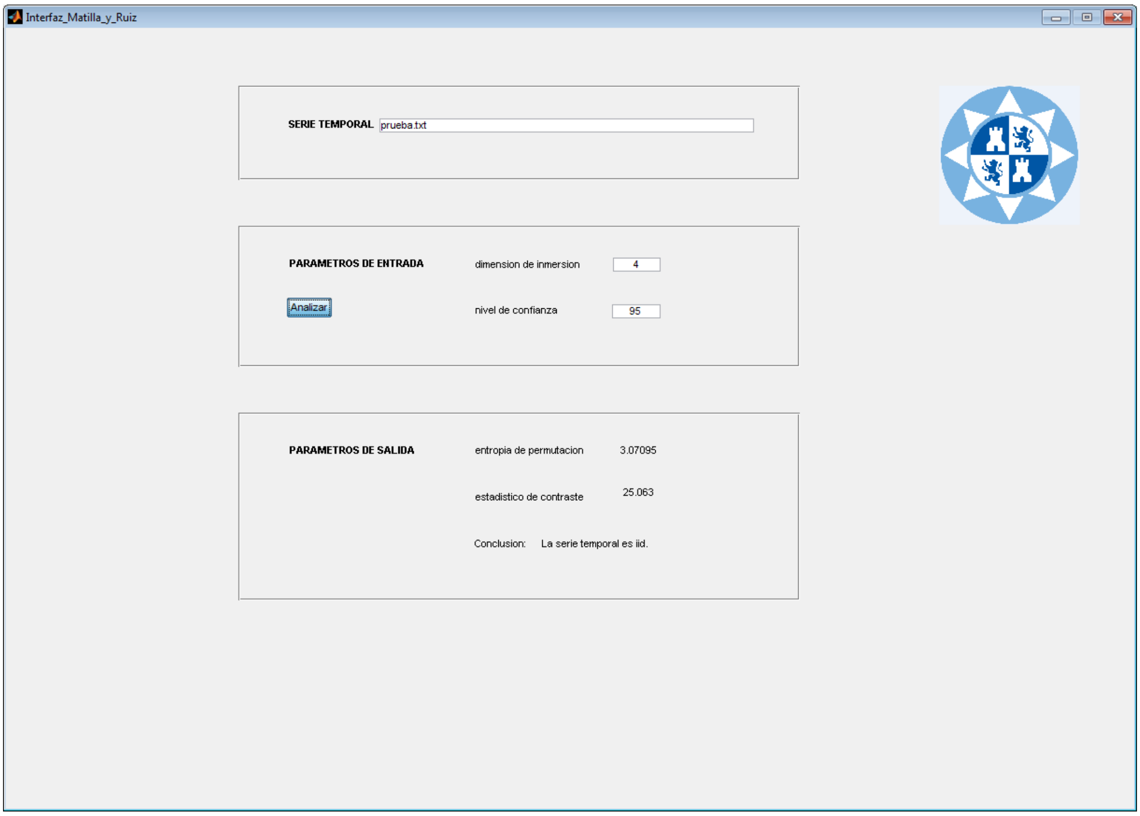
Si seleccionamos la opción *Ver*, nos aparece la siguiente ventana:



Antes de trabajar con una serie temporal hay que hacer unas sencillas operaciones. En primer lugar, hay que almacenar dicha serie en un archivo (en un archivo *.txt, por ejemplo); éste archivo debe situarse en el fichero en el que se encuentran todos los archivos de la aplicación Stemp 1.0. También hay que tener en cuenta que la serie temporal que se desee analizar debe estar almacenada en forma de lista (vector columna); y que si hay elementos que son números decimales, estos deben escribirse con punto, y no con coma.

En el espacio editable que hay junto al texto *SERIE TEMPORAL*, hay que escribir el nombre del archivo que contiene a la serie. Después pulsamos el botón *OK*, y obtendremos la lista con todos los elementos de la serie, y la representación gráfica de la misma.

Si en la ventana de presentación seleccionamos la opción *Analizar*, aparece un pequeño menú desplegable; si elegimos la opción *Mets. basados en permutaciones*, aparece un nuevo menú desplegable; si finalmente seleccionamos la opción *Matilla y Ruíz*, accedemos a la ventana siguiente:

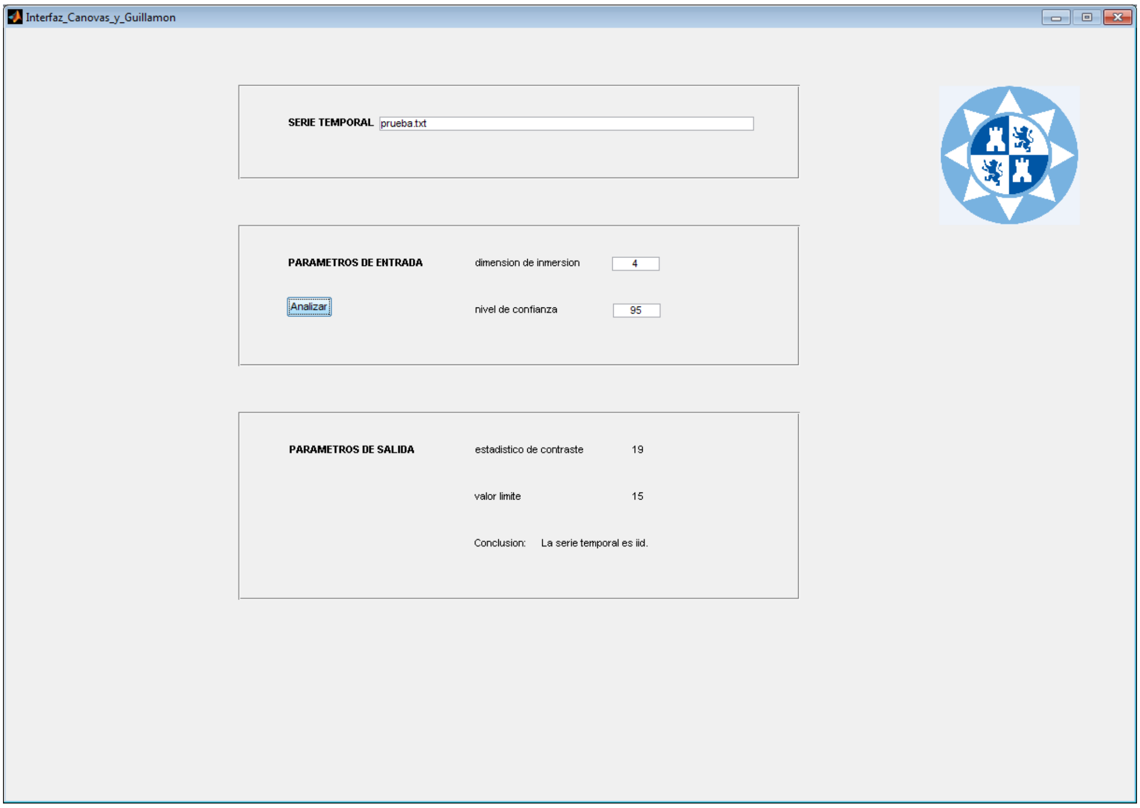


En *SERIE TEMPORAL*, tenemos que escribir el nombre del archivo en el que está almacenada la serie temporal que vamos a analizar. En *PARÁMETROS DE ENTRADA*, se introducen los valores de la dimensión de inmersión y del nivel de confianza.

Para proceder al análisis, le damos al botón Analizar.

En *PARÁMETROS DE SALIDA*, se obtienen la entropía de permutación, el estadístico de contraste y la conclusión del test llevado a cabo.

Si nos situamos ahora en el mismo menú desplegable que antes, pero seleccionamos la opción *Cánovas y Guillamón*, aparece la ventana que se muestra a continuación:

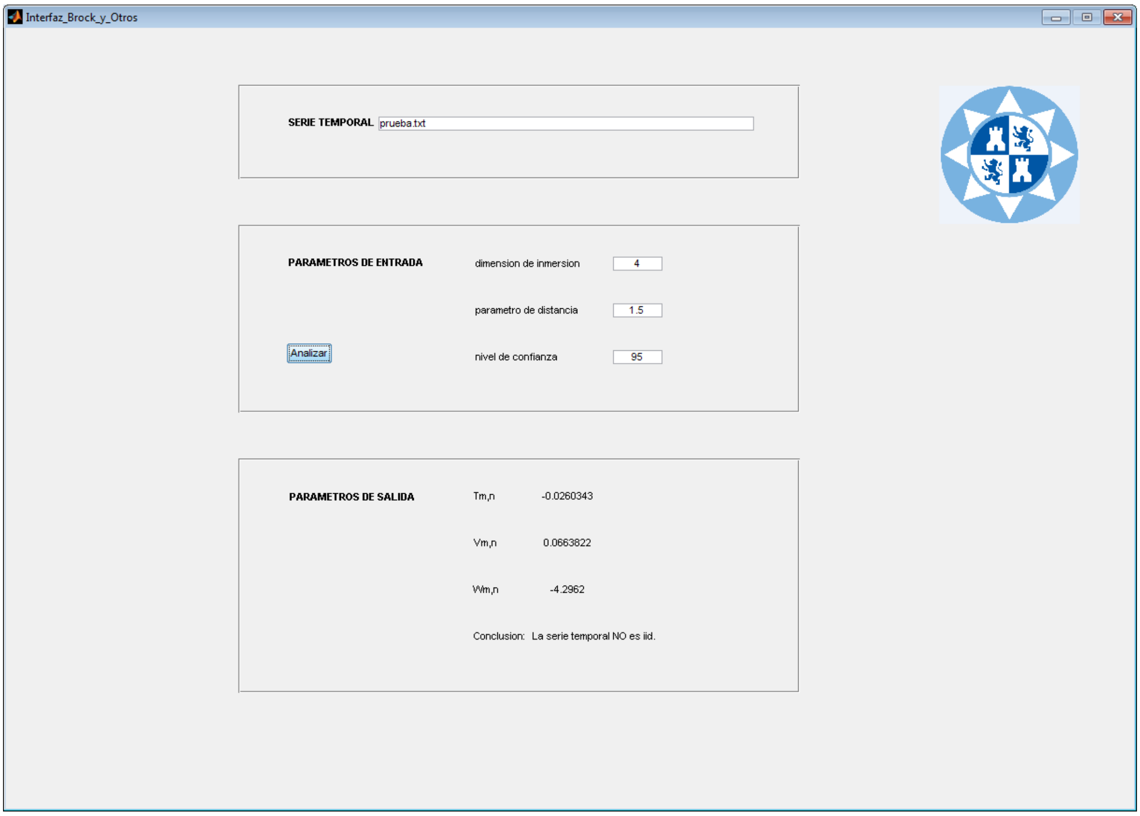


El funcionamiento de ésta es muy similar al de la anterior ventana.

En *SERIE TEMPORAL* se escribe el nombre del archivo que contiene la serie en cuestión. En *PARÁMETROS DE ENTRADA*, al igual que en la ventana anterior, hay que introducir los valores de la dimensión de inmersión y del nivel de confianza. Pero en este caso no podemos tomar los valores que se quiera; el valor de la dimensión de inmersión sólo puede ser *4*, *5* ó *6*; y el valor del nivel de confianza sólo puede ser *90*, *95* ó *99*.

En el panel inferior se presentan los resultados del análisis. El estadístico de contraste recoge el número de permutaciones admisibles diferentes. Y lo que en la ventana se denomina *valor límite*, representa el valor por encima del que debe encontrarse el estadístico de contraste para aceptar la hipótesis nula (para aceptar que la serie es i.i.d.).

La última ventana de la aplicación es la que aparece cuando, desde la ventana de presentación, seleccionamos *Analizar*, y luego *Met. Brock y otros*. El aspecto de ésta es el siguiente:



Tal como hemos hecho en los dos casos anteriores, en *SERIE TEMPORAL* hay que introducir el nombre del archivo que contiene la serie que queremos estudiar. En este caso, en *PARÁMETROS DE ENTRADA*, tenemos que escribir los valores de la dimensión de inmersión, el parámetro de distancia y el nivel de confianza.

En *PARÁMETROS DE SALIDA* se dan cuatro resultados. Cabe destacar el valor del estadístico de contraste (*Wm,n*), y la conclusión del test, en la que se establece si la serie temporal es i.i.d..

ANEXO 1

PROGRAMAS REALIZADOS CON MATHEMATICA

En este primer anexo damos la relación de los programas que se han realizado con Mathematica para el desarrollo de este trabajo.

No se pretende dar aquí el código de todos los programas realizados con Mathematica, sino sólo aquellos que tienen más importancia.

En todos ellos se ha añadido gran cantidad de comentarios, con el objeto de facilitar la comprensión del funcionamiento de los mismos.

```
(*MÉTODO 1*)
(*TEST 1*)

(*La referencia utilizada ha sido:*)

    (*M. Matilla, M. Ruíz*)
    (*A NON-PARAMETRIC INDEPENDENCE TEST USING PERMUTATION ENTROPY*)
    (*Journal of Econometrics 144 (2008), 139-155*)

(*Realizamos el test a través de los pasos que se exponen a continuación:*)
(*PASO 1: Introducimos la serie temporal con la que vamos a trabajar.*)
(*PASO 2: Calculamos h (entropía de permutación).*)
(*PASO 3: Realizamos el contraste de hipótesis.*)

ClearAll[serie, m, nivelConfianza, nivelSignificacion, T, trozo1Serie,
 permutacion1, Mpermutaciones, NpermutacionesDiferentes, FabsPermutacion,
 cont1, trozoSerie, permutacion, testigo, cont2, FrelPermutacion,
 H, cont3, incrementoH, G, gradosLibertad, chiCuadrado, test1]

(*PASO 1*)
(*INTRODUCIMOS LA SERIE CON LA QUE VAMOS A TRABAJAR*)

serie = RandomReal[{0, 1}, 120]; (*serie temporal*)
m = 4; (*dimensión de inmersión ('embedding dimension')*)
nivelConfianza = 95; (*nivel de confianza del test*)
nivelSignificacion = 1 - (nivelConfianza / 100);
(*nivel de significación del test*)

T = Length[serie]; (*número de elementos de la serie temporal*)

(*PASO 2*)
(*CALCULAMOS h (ENTROPÍA DE PERMUTACIÓN)*)

(*Para el cálculo de la entropía de permutación '
 H' es necesario hacer unos cálculos previos.*)
(*Hay que determinar la frecuencia absoluta de
 cada una de las posibles permutaciones.*)

trozo1Serie = serie[[1 ;; m]];
permutacion1 = Ordering[trozo1Serie];
Mpermutaciones = {permutacion1};
NpermutacionesDiferentes = 1;
FabsPermutacion = {1};

For[cont1 = 2, cont1 ≤ T - m + 1, cont1++,

  trozoSerie = serie[[cont1 ;; cont1 + m - 1]];
  permutacion = Ordering[trozoSerie];

  testigo = 0;
```

```
  For[cont2 = 1, cont2 ≤ NpermutacionesDiferentes, cont2++,

   If[Mpermutaciones[[cont2]] == permutacion,

     FabsPermutacion[[cont2]] = FabsPermutacion[[cont2]] + 1;
     testigo = 1;
     Break[];

    ];

  ];

  If[testigo == 0,

   Mpermutaciones = Append[Mpermutaciones, permutacion];
   NpermutacionesDiferentes = NpermutacionesDiferentes + 1;
   FabsPermutacion = Append[FabsPermutacion, 1];

  ];

 ];

(*Una vez calculadas las frecuencias absolutas de las permutaciones,
calculamos las frecuencias relativas.*)

FrelPermutacion = FabsPermutacion / (T - m + 1);

(*Con las frecuencias relativas de las permutaciones ya se
 puede determinar el valor de la entropía de permutación H.*)
(*La expresión matemática de la entropía de permutación es:*)

    (*h(m)=-∑pπln(pπ)*)

    (*donde pπ es la frecuencia relativa de cada permutación*)

H = 0; (*entropía de permutación*)

For[cont3 = 1, cont3 ≤ NpermutacionesDiferentes, cont3++,

  incrementoH = -FrelPermutacion[[cont3]] * Log[FrelPermutacion[[cont3]]];
  H = H + incrementoH;

 ];

(*PASO 3*)
(*REALIZAMOS EL CONTRASTE DE HIPÓTESIS*)

(*Calculamos 'G' (estadístico de contraste).*)
(*Su fórmula es:*)

    (*G(m)=2(T-m+1)(ln(m!)-h(m))*)
```

```
G = 2 * (T - m + 1) * (Log[m!] - H); (*estadístico de contraste G*)

(*Es interesante hacer notar que el estadístico de contraste no depende,
únicamente, de la entropía de permutación h.*)
(*También depende del número de elementos de la serie T,
y del valor de m ('embedding dimension').*)

gradosLibertad = m! - 1; (*grados de libertad*)
chiCuadrado =
  InverseCDF[ChiSquareDistribution[gradosLibertad], nivelConfianza / 100];
(*función de distribución Chi cuadrado*)

(*Si la variable test1 toma el valor 1, no rechazamos la hipótesis nula.*)
(*Si en cambio, la variable test1 toma el valor 0,
sí rechazamos la hipótesis nula.*)

If[G < chiCuadrado,

  test1 = 1,
  test1 = 0;

 ];

"Representación gráfica de los elementos de la serie temporal"
ListPlot[serie]
"Entropía de permutación 'H'"
N[H]
"Estadístico de contraste 'G'"
N[G]
"Distribución Chi cuadrado"
N[chiCuadrado]
"Resultado del test"
test1
```

```
(*MÉTODO 2*)
(*TEST 2-1*)

(*La referencia utilizada ha sido:*)

    (*J. S. Cánovas, A. Guillamón*)
    (*PERMUTATIONS AND TIME SERIES ANALYSIS*)
    (*Chaos 19 (2009), 197-235*)

(*Seguimos los siguientes pasos para hacer el test:*)
(*PASO 1: Introducimos la serie temporal con la que vamos a trabajar.*)
(*PASO 2: Calculamos H (entropía topológica).*)
(*PASO 3: Realizamos el contraste de hipótesis.*)

(*Aclaración: lo que aquí denominamos entropía topológica no es,
en sentido estricto, el valor real de dicha entropía.*)
(*Lo que en realidad representa H es el número
 de permutaciones diferentes que aparecen en la serie.*)
(*Hemos utilizado el nombre de entropía topológica porque
 este concepto está directamente relacionado con el
 número de permutaciones diferentes en la serie.*)

ClearAll[serie, m, nivelConfianza, nivelSignificacion, T, k,
 trozo1Serie, permutacion1, Mpermutaciones, NpermutacionesDiferentes,
 FabsPermutacion, cont1, trozoSerie, permutacion, testigo, cont2, H,
 NMaxPermutacionesDiferentes, auxiliarSuma, cont3, valorDistrTeorica, test21]

(*PASO 1*)
(*INTRODUCIMOS LA SERIE TEMPORAL CON LA QUE VAMOS A TRABAJAR*)

serie = RandomReal[{0, 1}, 120]; (*serie temporal*)
m = 4; (*dimensión de inmersión ('embedding dimension')*)
nivelConfianza = 95; (*nivel de confianza del test*)
nivelSignificacion = 1 - (nivelConfianza / 100);
(*nivel de significación del test*)

T = Length[serie]; (*número de elementos de la serie temporal*)
k = IntegerPart[T / m];
(*número de trozos en que se divide la serie temporal*)

(*PASO 2*)
(*CALCULAMOS H (ENTROPÍA TOPOLÓGICA)*)

(*La serie temporal se dividirá en varias partes (trozos), cumpliendo:*)
(*1.Cada trozo abarcará un conjunto de elementos consecutivos.*)
(*2.Los trozos no se solaparán.*)
(*3.Entre dos trozos consecutivos no se dejarán elementos libres.*)
(*4.El primer trozo se construirá a
 partir del primer elemento de la ventana.*)
(*5.Todos los trozos tendrán el mismo número de elementos,
y éste será m ('embedding dimension').*)
```

```
trozo1Serie = serie[[1 ;; m]];
permutacion1 = Ordering[trozo1Serie];
Mpermutaciones = {permutacion1};
NpermutacionesDiferentes = 1;
FabsPermutacion = {1};

For[ cont1 = m + 1, cont1 ≤ T - m + 1, cont1 = cont1 + m,

  trozoSerie = serie[[cont1 ;; cont1 + m - 1]];
  permutacion = Ordering[trozoSerie];

  testigo = 0;

  For[ cont2 = 1, cont2 ≤ NpermutacionesDiferentes, cont2++,

   If[Mpermutaciones[[cont2]] == permutacion,

     FabsPermutacion[[cont2]] = FabsPermutacion[[cont2]] + 1;
     testigo = 1;
     Break[];

    ];

  ];

  If[testigo == 0,

   Mpermutaciones = Append[Mpermutaciones, permutacion];
   NpermutacionesDiferentes = NpermutacionesDiferentes + 1;
   FabsPermutacion = Append[FabsPermutacion, 1];

  ];

 ];

(*PASO 3*)
(*REALIZAMOS EL CONTRASTE DE HIPÓTESIS*)

(*Determinamos el estadístico de contraste.*)
(*En este caso el estadístico de
 contrate coincide con la entropía topológica H.*)

H = NpermutacionesDiferentes;
(*entropía topológica H (estadístico de contraste)*)

(*A partir de la distribución teórica de probabilidad,
calculamos el valor que hay que comparar con el estadístico de contraste.*)

prob[k_, m_, t_] := N[StirlingS2[k, t] * ((m!) ! / (m! - t) !) * (1 / m!)^k];
(*función puntual de probabilidad teórica*)
```

```
NMaxPermutacionesDiferentes = Min[m!, k]; (*número máximo
 de permutaciones diferentes que pueden aparecer en la serie*)

auxiliarSuma = 0;

(*Determinamos a partir de qué valor del número de permutaciones diferentes,
la probabilidad asociada es mayor o igual a nivelConfianza/100.*)

For [cont3 = NMaxPermutacionesDiferentes, cont3 ≥ 1, cont3--,

  auxiliarSuma = auxiliarSuma + prob[k, m, cont3];

  If[auxiliarSuma ≥ nivelConfianza / 100,

   Break[];

  ];

 ];

valorDistrTeorica = cont3;

(*Si la variable test21 toma el valor 1, no rechazamos la hipótesis nula.*)
(*Si en cambio, la variable test21 toma el valor 0,
sí rechazamos la hipótesis nula.*)

If[H ≥ valorDistrTeorica,

  test21 = 1,
  test21 = 0;

 ];

"Representación gráfica de los elementos de la serie temporal"
ListPlot[serie]
"Entalpía topológica 'H' (estadístico de contraste)"
H
"Valor obtenido de la distribución teórica"
valorDistrTeorica
"Valor real del nivel de confianza"
100 * auxiliarSuma
"Resultado del test"
test21
```

```
(*MÉTODO 2*)
(*TEST 2-2*)

(*La referencia utilizada ha sido:*)

    (*J. S. Cánovas, A. Guillamón*)
    (*PERMUTATIONS AND TIME SERIES ANALYSIS*)
    (*Chaos 19 (2009), 197-235*)

(*Hacemos el test siguiendo los pasos que se muestran a continuación:*)
(*PASO 1: Introducimos la serie temporal con la que vamos a trabajar.*)
(*PASO 2: Seleccionamos el número de ventanas y su tamaño.*)
(*PASO 3: Calculamos H (entropía topológica) en cada una de las ventanas.*)
(*PASO 4: Realizamos el contraste de hipótesis.*)

(*Aclaración: lo que aquí denominamos entropía topológica no es,
en sentido estricto, el valor real de dicha entropía.*)
(*Lo que en realidad representa H es el número
 de permutaciones diferentes que aparecen en la serie.*)
(*Hemos utilizado el nombre de entropía topológica porque
 este concepto está directamente relacionado con el
 número de permutaciones diferentes en la serie.*)

ClearAll[serie, m, nivelConfianza, nivelSignificacion, T, Tventana,
 Nventanas, NelementosPerdidos, ventana1, Mventanas, cont1, ventanaCont1,
 k, vectorNpermutacionesDiferentes, func1, cont2, ventana, trozo1Ventana,
 permutacion1Ventana, MpermutacionesVentana, NpermutacionesDiferentesVentana,
 FabsPermutacionVentana, cont3, trozoVentana, permutacionVentana,
 testigo, cont4, vectorH, NMaxPermutacionesDiferentes, sumaEstContraste,
 cont5, fi, ei, cont6, EstContraste, gradosLibertad, chiCuadrado, test22]

(*PASO 1*)
(*INTRODUCIMOS LA SERIE TEMPORAL CON LA QUE VAMOS A TRABAJAR*)

serie = RandomReal[{0, 1}, 4800]; (*serie temporal*)
m = 4; (*dimensión de inmersión ('embedding dimension')*)
nivelConfianza = 99; (*nivel de confianza del test*)
nivelSignificacion = 1 - (nivelConfianza / 100);
(*nivel de significación del test*)

T = Length[serie]; (*número de elementos de la serie temporal*)

(*PASO 2*)
(*SELECCIONAMOS EL NÚMERO DE VENTANAS Y SU TAMAÑO*)

(*Vamos a dividir la serie temporal en varias partes (ventanas),
cumpliendo:*)
(*1.Cada ventana abarcará un conjunto de elementos consecutivos.*)
(*2.Las ventanas no se solaparán.*)
(*3.Entre dos ventanas consecutivas no se dejarán elementos libres.*)
(*4.La primera ventana se construirá
 a partir del primer elemento de la serie.*)
```

```
 a partir del primer elemento de la serie.*)
(*5.Todas las ventanas tendrán el mismo número de elementos,
y éste será 5m!.*)

(*La condición 4 implica que, en caso de tener elementos sobrantes,
estos serán los del final de la serie.*)

(*En el caso de que sólo se considere una ventana,
ésta abarcará toda la serie temporal.*)
(*Esta imposición evitará la pérdida de elementos de la serie en el
 caso de que el número total de elementos esté entre 5m! y 2(5m!).*)

If[T ≥ (5 * m!) * 2,

  Tventana = 5 * m!;
  Nventanas = IntegerPart[T / Tventana],

  Tventana = T;
  Nventanas = 1;

 ];

NelementosPerdidos = T - Nventanas * Tventana;
(*número de elementos perdidos (al final de la serie)*)

ventana1 = serie[[1 ;; Tventana]];
Mventanas = {ventana1};

For[cont1 = 2, cont1 ≤ Nventanas, cont1++,

  ventanaCont1 = serie[[(cont1 - 1) * Tventana + 1 ;; cont1 * Tventana]];
  Mventanas = Append[Mventanas, ventanaCont1];

 ];

(*PASO 3*)
(*CALCULAMOS H (ENTROPÍA TOPOLÓGICA) EN CADA UNA DE LAS VENTANAS*)

(*Cada ventana se dividirá en varias partes (trozos), cumpliendo:*)
(*1.Cada trozo abarcará un conjunto de elementos consecutivos.*)
(*2.Los trozos no se solaparán.*)
(*3.Entre dos trozos consecutivos no se dejarán elementos libres.*)
(*4.El primer trozo se construirá a
 partir del primer elemento de la ventana.*)
(*5.Todos los trozos tendrán el mismo número de elementos,
y éste será m ('embedding dimension').*)

(*No se perderán elementos en las ventanas.*)
(*Esto se debe a que la división del número de elementos de cada ventana
 (5m!) entre el tamaño de cada trozo (m) es una división exacta.*)
```

```
k = Tventana / m; (*número de trozos en que se divide cada ventana*)

(*Contamos el número de permutaciones diferentes en cada ventana.*)

vectorNpermutacionesDiferentes = Array[func1, Nventanas];

For[cont2 = 1, cont2 ≤ Nventanas, cont2++,

  ventana = Mventanas[[cont2]];

  trozo1Ventana = ventana[[1 ;; m]];
  permutacion1Ventana = Ordering[trozo1Ventana];
  MpermutacionesVentana = {permutacion1Ventana};
  NpermutacionesDiferentesVentana = 1;
  FabsPermutacionVentana = {1};

  For[ cont3 = m + 1, cont3 ≤ Tventana - m + 1, cont3 = cont3 + m,

   trozoVentana = ventana[[cont3 ;; cont3 + m - 1]];
   permutacionVentana = Ordering[trozoVentana];

   testigo = 0;

   For[ cont4 = 1, cont4 ≤ NpermutacionesDiferentesVentana, cont4++,

    If[MpermutacionesVentana[[cont4]] == permutacionVentana,

      FabsPermutacionVentana[[cont4]] = FabsPermutacionVentana[[cont4]] + 1;
      testigo = 1;
      Break[];

     ];

   ];

   If[testigo == 0,

    NpermutacionesDiferentesVentana = NpermutacionesDiferentesVentana + 1;
    MpermutacionesVentana =
     Append[MpermutacionesVentana, permutacionVentana];
    FabsPermutacionVentana = Append[FabsPermutacionVentana, 1];

   ];

  ];

  func1[cont2] = NpermutacionesDiferentesVentana;

 ];

(*Construimos el vector de entropías topológicas (vectorH).*)
```

```
vectorH = vectorNpermutacionesDiferentes;
(*vector cuyos elementos son las entropías topológicas de cada ventana*)

(*PASO 4*)
(*REALIZAMOS EL CONTRASTE DE HIPÓTESIS*)

(*Calculamos el estadístico de contraste.*)

prob[k_, m_, t_] := N[StirlingS2[k, t] * ((m!) ! / (m! - t) !) * (1 / m!) ^k];
(*función puntual de probabilidad teórica*)

NMaxPermutacionesDiferentes = Min[k, m!];
(*número máximo de permutaciones diferentes en cada ventana*)

sumaEstContraste = 0;

(*Contamos las ventanas que tienen cada uno de los posibles
 valores que puede tener el número de permutaciones diferentes.*)

For[cont5 = 1, cont5 ≤ NMaxPermutacionesDiferentes, cont5++,

  fi = 0;

  ei = Nventanas * prob[k, m, cont5];

  For[cont6 = 1, cont6 ≤ Nventanas, cont6++,

   If[cont5 == vectorNpermutacionesDiferentes[[cont6]],

     fi = fi + 1;

    ];

  ];

  sumaEstContraste = sumaEstContraste + ((fi - ei) ^2) / ei;

 ];

EstContraste = sumaEstContraste; (*estadístico de contraste*)

(*Determinamos el valor límite que
 fijará la zona de rechazo de la hipótesis nula.*)

gradosLibertad = NMaxPermutacionesDiferentes - 1; (*grados de libertad*)
chiCuadrado =
  InverseCDF[ChiSquareDistribution[gradosLibertad], nivelConfianza / 100];
(*función de distribución Chi cuadrado*)

(*Si la variable test22 toma el valor 1, no rechazamos la hipótesis nula.*)
```

```
(*Si en cambio, la variable test22 toma el valor 0,
sí rechazamos la hipótesis nula.*)

If[EstContraste < chiCuadrado,

  test22 = 1,
  test22 = 0;

 ];

"Representación gráfica de los elementos de la serie temporal"
ListPlot[serie]
"Entropía topológica de cada ventana"
vectorH
"Gráfica del número de permutaciones
  diferentes de cada ventana (vector de entropías)"
ListLinePlot[vectorH]
"Estadístico de contraste"
EstContraste
"Distribución Chi cuadrado"
N[chiCuadrado]
"Resultado del test"
test22
```

```
(*MÉTODO 2*)
(*TEST 2-3*)

(*La referencia utilizada ha sido:*)

    (*J. S. Cánovas, A. Guillamón*)
    (*PERMUTATIONS AND TIME SERIES ANALYSIS*)
    (*Chaos 19 (2009), 197-235*)

(*Realizaremos el test siguiendo los siguientes pasos:*)
(*PASO 1: Introducimos la serie temporal con la que vamos a trabajar.*)
(*PASO 2: Seleccionamos el número de ventanas y su tamaño.*)
(*PASO 3: Calculamos H (entropía topológica) en cada una de las ventanas.*)
(*PASO 4: Determinamos la función de distribución teórica FW.*)
(*PASO 5: Determinamos la función de distribución empírica Fn.*)
(*PASO 6: Realizamos el contraste de hipótesis.*)

(*Aclaración: lo que aquí denominamos entropía topológica no es,
en sentido estricto, el valor real de dicha entropía.*)
(*Lo que en realidad representa H es el número
 de permutaciones diferentes que aparecen en la serie.*)
(*Hemos utilizado el nombre de entropía topológica porque
 este concepto está directamente relacionado con el
 número de permutaciones diferentes en la serie.*)

ClearAll[serie, m, nivelConfianza, nivelSignificacion, T, Tventana,
 Nventanas, NelementosPerdidos, ventana1, Mventanas, cont1, ventanaCont1,
 k, vectorNpermutacionesDiferentes, func1, cont2, ventana, trozo1Ventana,
 permutacion1Ventana, MpermutacionesVentana, NpermutacionesDiferentesVentana,
 FabsPermutacionVentana, cont3, trozoVentana, permutacionVentana,
 testigo, cont4, vectorH, NMaxPermutacionesDiferentes, sumaFW, FW,
 func2, cont5, auxiliarFn, func3, cont6, sumaAuxiliarFn, cont7, sumaFn,
 Fn, func4, cont8, diferencias, Dn, EstContraste, pValor, test23]

(*PASO 1*)
(*INTRODUCIMOS LA SERIE TEMPORAL CON LA QUE VAMOS A TRABAJAR*)

serie = RandomReal[{0, 1}, 4800]; (*serie temporal*)
m = 4; (*dimensión de inmersión ('embedding dimension')*)
nivelConfianza = 95; (*nivel de confianza del test*)
nivelSignificacion = 1 - (nivelConfianza / 100);
(*nivel de significación del test*)

T = Length[serie]; (*número de elementos de la serie temporal*)

(*PASO 2*)
(*SELECCIONAMOS EL NÚMERO DE VENTANAS Y SU TAMAÑO*)

(*Vamos a dividir la serie temporal en varias partes (ventanas),
cumpliendo:*)
(*1.Cada ventana abarcará un conjunto de elementos consecutivos.*)
(*2.Las ventanas no se solaparán.*)
```

```
(*2.Las ventanas no se solaparán.*)
(*3.Entre dos ventanas consecutivas no se dejarán elementos libres.*)
(*4.La primera ventana se construirá
 a partir del primer elemento de la serie.*)
(*5.Todas las ventanas tendrán el mismo número de elementos,
y éste será 5m!.*)

(*La condición 4 implica que, en caso de tener elementos sobrantes,
estos serán los del final de la serie.*)

(*En el caso de que sólo se considere una ventana,
ésta abarcará toda la serie temporal.*)
(*Esta imposición evitará la pérdida de elementos de la serie en el
 caso de que el número total de elementos esté entre 5m! y 2(5m!).*)

If[T ≥ (5 * m!) * 2,

  Tventana = 5 * m!;
  Nventanas = IntegerPart[T / Tventana],

  Tventana = T;
  Nventanas = 1;

 ];

NelementosPerdidos = T - Nventanas * Tventana;
(*número de elementos perdidos (al final de la serie)*)

ventana1 = serie[[1 ;; Tventana]];
Mventanas = {ventana1};

For[cont1 = 2, cont1 ≤ Nventanas, cont1++,

  ventanaCont1 = serie[[(cont1 - 1) * Tventana + 1 ;; cont1 * Tventana]];
  Mventanas = Append[Mventanas, ventanaCont1];

 ];

(*PASO 3*)
(*CALCULAMOS H (ENTROPÍA TOPOLÓGICA) EN CADA UNA DE LAS VENTANAS*)

(*Cada ventana se dividirá en varias partes (trozos), cumpliendo:*)
(*1.Cada trozo abarcará un conjunto de elementos consecutivos.*)
(*2.Los trozos no se solaparán.*)
(*3.Entre dos trozos consecutivos no se dejarán elementos libres.*)
(*4.El primer trozo se construirá a
 partir del primer elemento de la ventana.*)
(*5.Todos los trozos tendrán el mismo número de elementos,
y éste será m ('embedding dimension').*)

(*No se perderán elementos en las ventanas.*)
```

```
(*Esto se debe a que la división del número de elementos de cada ventana
 (5m!) entre el tamaño de cada trozo (m) es una división exacta.*)

k = Tventana / m; (*número de trozos en que se divide cada ventana*)

vectorNpermutacionesDiferentes = Array[func1, Nventanas];

For[cont2 = 1, cont2 ≤ Nventanas, cont2++,

  ventana = Mventanas[[cont2]];

  trozo1Ventana = ventana[[1 ;; m]];
  permutacion1Ventana = Ordering[trozo1Ventana];
  MpermutacionesVentana = {permutacion1Ventana};
  NpermutacionesDiferentesVentana = 1;
  FabsPermutacionVentana = {1};

  For[ cont3 = m + 1, cont3 ≤ Tventana - m + 1, cont3 = cont3 + m,

   trozoVentana = ventana[[cont3 ;; cont3 + m - 1]];
   permutacionVentana = Ordering[trozoVentana];

   testigo = 0;

   For[ cont4 = 1, cont4 ≤ NpermutacionesDiferentesVentana, cont4++,

    If[MpermutacionesVentana[[cont4]] == permutacionVentana,

      FabsPermutacionVentana[[cont4]] = FabsPermutacionVentana[[cont4]] + 1;
      testigo = 1;
      Break[];

     ];

   ];

   If[testigo == 0,

    MpermutacionesVentana =
     Append[MpermutacionesVentana, permutacionVentana];
    NpermutacionesDiferentesVentana = NpermutacionesDiferentesVentana + 1;
    FabsPermutacionVentana = Append[FabsPermutacionVentana, 1];

   ];

  ];

  func1[cont2] = NpermutacionesDiferentesVentana;

 ];
```

```
(*Construimos el vector de entropías topológicas (vectorH).*)

vectorH = vectorNpermutacionesDiferentes;
(*vector cuyos elementos son las entropías topológicas de cada ventana*)

(*PASO 4*)
(*DETERMINAMOS LA FUNCIÓN DE DISTRIBUCIÓN TEÓRICA FW*)

prob[k_, m_, t_] := N[StirlingS2[k, t] * ((m!)! / (m! - t)!) * (1 / m!)^k];
(*función puntual de probabilidad teórica*)

NMaxPermutacionesDiferentes = Min[k, m!];
(*número máximo de permutaciones diferentes en cada ventana*)

sumaFW = 0;
FW = Array[func2, NMaxPermutacionesDiferentes];
(*función de distribución teórica*)

For[cont5 = 1, cont5 ≤ NMaxPermutacionesDiferentes, cont5++,

  sumaFW = sumaFW + prob[k, m, cont5];
  func2[cont5] = sumaFW;

 ];

(*PASO 5*)
(*DETERMINAMOS LA FUNCIÓN DE DISTRIBUCIÓN EMPÍRICA Fn*)

auxiliarFn = Array[func3, NMaxPermutacionesDiferentes];

For[cont6 = 1, cont6 ≤ NMaxPermutacionesDiferentes, cont6++,

  sumaAuxiliarFn = 0;

  For[cont7 = 1, cont7 ≤ Nventanas, cont7++,

   If[cont6 == vectorH[[cont7]],

     sumaAuxiliarFn = sumaAuxiliarFn + 1;

    ];

  ];

  func3[cont6] = sumaAuxiliarFn / Nventanas;

 ];

sumaFn = 0;
Fn = Array[func4, NMaxPermutacionesDiferentes];
(*función de distribución empírica*)
```

```
For[cont8 = 1, cont8 ≤ m!, cont8++,

  sumaFn = sumaFn + auxiliarFn[[cont8]];
  func4[cont8] = sumaFn;

 ];

(*PASO 6*)
(*REALIZAMOS EL CONTRASTE DE HIPÓTESIS*)

diferencias = Fn - FW;

(*Calculamos el estadístico de contraste.*)

Dn = Norm[diferencias, Infinity];
EstContraste = Sqrt[Nventanas] * Dn; (*estadístico de contraste*)

(*Determinamos el p-valor.*)
```

$$pValor = 2 * \sum_{i=1}^{\infty} (-1)^{i-1} * e^{-2*i^2*EstContraste^2}; \text{ (*p-valor*)}$$

```
(*Si la variable test23 toma el valor 1, no rechazamos la hipótesis nula.*)
(*Si en cambio, la variable test23 toma el valor 0,
sí rechazamos la hipótesis nula.*)

If[pValor ≥ nivelSignificacion,

  test23 = 1,
  test23 = 0;

 ];

"Representación gráfica de los elementos de la serie temporal"
ListPlot[serie]
"Función de distribución teórica 'FW'"
FW
"Función de distribución empírica 'Fn'"
Fn
"Gráficas de 'FW' y 'Fn'"
ListLinePlot[{FW, Fn}]
"p-valor"
pValor
"Nivel de significación"
N[nivelSignificacion]
"Resultado del test"
test23
```

```
(*MÉTODO 3*)
(*TEST 3: TEST BDS*)
(*VERSIÓN 1*)

(*Las referencias utilizadas han sido, principalmente:*)

    (*W.A. Brock, W.D. Dechert, J.A. Scheinkman, B. LeBaron*)
    (*A TEST FOR INDEPENDENCE BASED ON THE CORRELATION DIMENSION*)
    (*Econometric Reviews 15 (1996), 197-235*)

    (*L. Kanzler*)
    (*VERY FAST AND CORRECTLY SIZED ESTIMATION OF THE BDS STATISTIC*)
    (*Department of Economics of Oxford University (1999)*)

(*Para la realización de este test,
esta primera versión recurre a los dos artículos citados más arriba.*)
(*El algoritmo programado consiste, fundamentalmente,
en una aplicación directa de las expresiones
 matemáticas recogidas en el primer artículo.*)
(*Dichas fórmulas presentan aquí pequeñas modificaciones
 para una correcta aplicación del test.*)
(*El coste computacional de este algoritmo es muy alto,
por lo que es una opción poco práctica para la aplicación de este test.*)
(*Más adelante se presentará un algoritmo más eficaz que éste,
que disminuye de forma notable el coste
 computacional (ver la versión 2 del test).*)

(*Para implementar el algoritmo, los pasos a seguir son:*)

(*PASO 1: Introducimos la serie temporal con la que vamos a trabajar.*)
(*PASO 2: Calculamos el estadístico de contraste W_{m,n}.*)
(*PASO 2.1: Calculamos T_{m,n}.*)
(*PASO 2.2: Calculamos V_{m,n}.*)
(*PASO 2.3: Calculamos W_{m,n}.*)
(*PASO 3: Realizamos el contraste de hipótesis.*)

(*En primer lugar borramos todos los valores de las
 variables y constantes que aparecen en el programa.*)
(*Se pretende evitar que entre dos ejecuciones consecutivas del programa,
algún símbolo adopte un valor equivocado.*)

ClearAll[serie, m, nivelConfianza, nivelSignificacion, n, Ntrozos,
 desviacionTipica, epsilon, Ncombinaciones, sumaCmn, cont1, trozoSerieCmn,
 cont2, diferenciaCmn, normaDiferenciaCmn, Cmn, sumaC1n, cont3,
 elementoSerieC1n, cont4, diferenciaC1n, normaDiferenciaC1n, C1n, Tmn,
 sumaCn, cont5, elementoSerieCn, cont6, diferenciaCn, VabsolutoDiferenciaCn,
 Cn, sumaKn, cont7, elementoSerieKni, cont8, elementoSerieKnj,
 diferenciaKnij, VabsolutoDiferenciaKnij, cont9, elementoSerieKnk,
 diferenciaKnjk, VabsolutoDiferenciaKnjk, Kn, Vmn, Wmn, test3]

(*PASO 1*)
(*INTRODUCIMOS LA SERIE TEMPORAL CON LA QUE VAMOS A TRABAJAR.*)
```

```
(*INTRODUCIMOS LA SERIE TEMPORAL CON LA QUE VAMOS A TRABAJAR*)

serie = RandomReal[{0, 1}, 120]; (*serie temporal*)
m = 4; (*dimensión de inmersión ('embedding dimension')*)
nivelConfianza = 95; (*nivel de confianza del test*)
nivelSignificacion = 1 - (nivelConfianza / 100);
(*nivel de significación del test*)

n = Length[serie]; (*número de elementos de la serie temporal*)
Ntrozos = n - m + 1; (*número de trozos en que dividimos la serie temporal*)

(*PASO 2*)
(*CALCULAMOS EL ESTADÍSTICO DE CONTRASTE W_{m,n}*)

(*El cálculo del estadístico de
 contraste W_{m,n} lo dividimos en tres partes:*)
(*Primera parte: calculamos T_{m,n}.*)
(*Segunda parte: calculamos V_{m,n}.*)
(*Tercera parte: calculamos W_{m,n} a partir de los valores de T_{m,n} y V_{m,n}.*)

(*Los valores de T_{m,n} y V_{m,n} dependen de un parámetro ϵ.*)
(*El valor más apropiado para la relación ϵ/σ
 depende del valor de la dimensión de inmersión m.*)
(*L. Kanzler (1999) establece que para
  valores de m bajos debe tomarse ϵ/σ=1.5.*)

desviacionTipica = StandardDeviation[serie];
(*desviación típica de los elementos de la serie*)
epsilon = 1.5 * desviacionTipica; (*parámetro ϵ*)

(*PASO 2.1*)
(*CALCULAMOS T_{m,n}*)

(*El valor de T_{m,n} viene dado por:*)

    (*T_{m,n}(ϵ)=C_{m,n}(ϵ)-C_{1,n-m+1}(ϵ)^m*)

(*Por tanto, hay que calcular antes C_{m,n} y C_{1,n-m+1}.*)
(*A su vez, para el cálculo de C_{m,n} y C_{1,n-m+1},
hay que calcular el número combinatorio Binomial(n-m+1, 2).*)

Ncombinaciones = Ntrozos * (Ntrozos - 1) / 2;

(*CALCULAMOS C_{m,n}*)

(*Representemos la serie temporal mediante {u_t}.*)
(*Denotemos u_t^m a la m-historia (u_t,u_{t+1},...,u_{t+m-1}).*)
(*C_{m,n} se calcula con la fórmula:*)
```

```
(*C_{m,n}(ϵ) = 1/Binomial[n-m+1, 2] ∑_{s=1}^{n-m} ∑_{t=s+1}^{n-m+1} χϵ(‖u_s^m - u_t^m‖) *)
```

$$(*C_{m,n}(\epsilon)=\frac{1}{\binom{n-m+1}{2}}\sum_{s=1}^{n-m}\sum_{t=s+1}^{n-m+1}\chi_\epsilon\left(\left\|u_s^m-u_t^m\right\|\right)*)$$

```
(*A continuación se evalúa el
 sumatorio que aparece en la fórmula anterior.*)
(*Para ello se recurre a la variable auxiliar sumaCmn.*)

sumaCmn = 0;

For[cont1 = 1, cont1 ≤ Ntrozos - 1, cont1++,

  trozoSerieCmn = serie[[cont1 ;; cont1 + m - 1]];

  For[cont2 = cont1 + 1, cont2 ≤ Ntrozos, cont2++,

   diferenciaCmn = trozoSerieCmn - serie[[cont2 ;; cont2 + m - 1]];
   normaDiferenciaCmn = Norm[diferenciaCmn, Infinity];

   If[normaDiferenciaCmn < epsilon,

    sumaCmn = sumaCmn + 1;

   ];

  ];

 ];

Cmn = sumaCmn / Ncombinaciones; (*valor de Cm,n*)

(*CALCULAMOS C1,n-m+1*)

(*Para calcular C1,n-m+1 utilizaremos la fórmula que se
 utilizó para el cálculo de Cm,n, pero con dos modificaciones:*)

    (*- Consideramos los elementos de la
 serie a partir del que está en la posición m.*)
    (*- Los trozos de la serie pasan a tener un solo elemento.*)

(*Con los cambios descritos, la fórmula queda:*)
```

$$(*C_{1,n-m+1}(\epsilon)=\frac{1}{\binom{n-m+1}{2}}\sum_{s=m}^{n-1}\sum_{t=s+1}^{n}\chi_\epsilon\left(\left\|u_s^1-u_t^1\right\|\right)*)$$

```
(*Procediendo igual que en el cálculo de Cm,n, calcularemos el
 sumatorio de la fórmula con ayuda de la variable auxiliar sumaC1n.*)

sumaC1n = 0;

For[cont3 = m, cont3 ≤ n - 1, cont3++,
```

```
    elementoSerieC1n = serie[[cont3]];

    For[cont4 = cont3 + 1, cont4 ≤ n, cont4++,

     diferenciaC1n = elementoSerieC1n - serie[[cont4]];
     normaDiferenciaC1n = Abs[diferenciaC1n];

     If[normaDiferenciaC1n < epsilon,

      sumaC1n = sumaC1n + 1;

     ];

    ];

   ];
```

C1n = sumaC1n / Ncombinaciones; (*valor de $C_{1,n-m+1}$*)

(*CALCULAMOS $T_{m,n}$*)

(*Con los valores de $C_{m,n}$ y $C_{1,n-m+1}$ calculados,
ya podemos determinar el valor de $T_{m,n}$.*)

Tmn = Cmn - (C1n)^m; (*valor de $T_{m,n}$*)

(*PASO 2.2*)
(*CALCULAMOS $V_{m,n}$*)

(*$V_{m,n}$ es la estimación de la desviación típica de $T_{m,n}$.*)

(*$V_{m,n}$ se calcula mediante la expresión siguiente:*)

(*$V_{m,n}{}^2(\epsilon)=4\left[K_n^m+2\sum_{j=1}^{m-1}K_n^{m-j}C_n^{2j}+(m-1)^2C_{1,n}^{2m}-m^2K_nC_n^{2m-2}\right]$*)

(*En consecuencia,
para calcular $V_{m,n}$ hay que determinar previamente C_n y K_n.*)

(*CALCULAMOS C_n*)

(*La expresión matemática de C_n es:*)

(*$C_n(\epsilon)=\frac{1}{\binom{n}{2}}\sum_{s=1}^{n-1}\sum_{t=s+1}^{n}\chi_\epsilon\left(\left\|u_s^1-u_t^1\right\|\right)$*)

(*Para calcular el sumatorio anterior
definimos la variable auxiliar sumaCn.*)

```
sumaCn = 0;
```

```
For[cont5 = 1, cont5 ≤ n - 1, cont5++,

  elementoSerieCn = serie[[cont5]];

  For[cont6 = cont5 + 1, cont6 ≤ n, cont6++,

   diferenciaCn = elementoSerieCn - serie[[cont6]];
   VabsolutoDiferenciaCn = Abs[diferenciaCn];

   If[VabsolutoDiferenciaCn < epsilon,

    sumaCn = sumaCn + 1;

   ];

  ];

 ];

Cn = sumaCn / (n * (n - 1) / 2); (*valor de Cn*)

(*En la expresión de cálculo de Cn se ha tenido en cuenta que (n 2) = n(n-1)/2 *)

(*CALCULAMOS Kn*)

(*El cálculo de Kn se realiza con la fórmula:*)
```

$$(*K_n(\epsilon) = \frac{1}{\binom{n}{3}} \sum_{r=1}^{n-2}\sum_{s=1}^{n-1}\sum_{t=1}^{n}\chi_\epsilon\left(\left\|u_r^1-u_s^1\right\|\right)\chi_\epsilon\left(\left\|u_s^1-u_t^1\right\|\right)*)$$

```
(*Recurrimos, una vez más, a una variable auxiliar.*)
(*En este caso la variable auxiliar es sumaKn,
y se utilizará para calcular el sumatorio de la fórmula anterior.*)

sumaKn = 0;

For[ cont7 = 1, cont7 ≤ n - 2, cont7++,

  elementoSerieKni = serie[[cont7]];

  For[ cont8 = cont7 + 1, cont8 ≤ n - 1, cont8++,

   elementoSerieKnj = serie[[cont8]];

   diferenciaKnij = elementoSerieKni - elementoSerieKnj;
   VabsolutoDiferenciaKnij = Abs[diferenciaKnij];

   If[VabsolutoDiferenciaKnij < epsilon,
```

```
      For [cont9 = cont8 + 1, cont9 ⩽ n, cont9 ++,

        elementoSerieKnk = serie[[cont9]];

        diferenciaKnjk = elementoSerieKnj – elementoSerieKnk;
        VabsolutoDiferenciaKnjk = Abs[diferenciaKnjk];

        If[VabsolutoDiferenciaKnjk < epsilon,

         sumaKn = sumaKn + 1;

        ];

       ];

     ];

    ];

   ];
 (*Una vez determinado el sumatorio es inmediato calcular Kn.*)

 Kn = sumaKn / (n * (n – 1) * (n – 2) / 6); (*valor de Kn*)

 (*Aquí se ha tenido en cuenta que (n 3) = n(n-1)(n-2)/6 *)

 (*CALCULAMOS Vm,n*)

 (*Ya determinados los valores de Cn y Kn, pasamos a determinar Vm,n.*)

 Vmn = 2 * Sqrt[Kn^m + 2 * Sum[Kn^(m-i) * Cn^(2*i), {i, 1, m-1}] + (m – 1)^2 * Cn^(2*m) – m^2 * Kn * Cn^(2*m-2)];
 (*valor de Vm,n*)

 (*PASO 2.3*)
 (*CALCULAMOS Wm,n*)

 (*La fórmula es:*)

     (*Wm,n(ϵ) = Sqrt[n-m+1] Tm,n(ϵ)/Vm,n(ϵ) *)

 Wmn = Sqrt[Ntrozos] * Tmn / Vmn; (*estadístico de contraste Wm,n*)

 (*PASO 3*)
 (*REALIZAMOS EL CONTRASTE DE HIPÓTESIS*)

 (*Para realizar el contraste de hipótesis
  utilizamos la distribución normal estándar.*)
```

```
(*Introducimos la variable test3 para almacenar el resultado del test.*)
(*Si la variable 'test3' toma el valor 1,
no rechazamos la hipótesis nula.*)
(*Si en cambio, la variable 'test3' toma el valor 0,
sí rechazamos la hipótesis nula.*)

If[Abs[Wmn] ≤ InverseCDF[NormalDistribution[0, 1], 1 - nivelSignificacion / 2],

  test3 = 1,
  test3 = 0;

 ];

"Representación gráfica de los elementos de la serie temporal"
ListPlot[serie]
"Estadístico de contraste"
N[Wmn]
"Resultado del test"
test3
```

```
(*MÉTODO 3*)
(*TEST 3: TEST BDS*)
(*VERSIÓN 2*)

(*Las referencias utilizadas han sido, principalmente:*)

    (*W.A. Brock, W.D. Dechert, J.A. Scheinkman, B. LeBaron*)
    (*A TEST FOR INDEPENDENCE BASED ON THE CORRELATION DIMENSION*)
    (*Econometric Reviews 15 (1996), 197-235*)

    (*L. Kanzler*)
    (*VERY FAST AND CORRECTLY SIZED ESTIMATION OF THE BDS STATISTIC*)
    (*Department of Economics of Oxford University (1999)*)

(*Para la aplicación del test se han tenido
 en cuenta los dos artículos citados más arriba.*)
(*El algoritmo utilizado tiene varias similitudes
 con el expuesto en Kanzler (1999).*)
(*Además, con este algoritmo se aplica el test considerando,
de forma simultánea, distintos valores para la dimensión de inmersión m.*)
(*El coste computacional de este algoritmo es
 inferior al del algoritmo presentado en la versión 1.*)
(*En cualquier caso, el coste computacional sigue siendo uno de los
 inconvenientes más importantes en la aplicación del test BDS.*)

(*Para implementar el algoritmo, los pasos a seguir son:*)

(*PASO 1: Introducimos la serie temporal con la que vamos a trabajar.*)
(*PASO 2: Calculamos el estadístico de contraste W_{m,n}.*)
(*PASO 3: Realizamos el contraste de hipótesis.*)

(*Lo primero de todo es borrar los valores de todas
 las variables y constantes que aparecen en el programa.*)
(*Se quiere evitar que alguna de las variables adopte un valor
 erróneo entre dos ejecuciones consecutivas del programa.*)

ClearAll[serie, mMaxima, nivelConfianza, nivelSignificacion,
 n, desviacionTipica, epsilon, matriz, func1, sumaCn, cont1,
 elementoSerie, cont2, normaDiferencia, Cn, vectorC1n, func2,
 sumaAuxiliar, cont3, cont4, sumaKn, cont5, cont6, cont7, Kn,
 vectorCmn, func3, nCmn, m, sumaCmn, cont8, cont9, auxiliar, vectorTmn,
 func4, vectorVmn, func5, vectorWmn, func6, vectorTest3, func7]

(*PASO 1*)
(*INTRODUCIMOS LA SERIE TEMPORAL CON LA QUE VAMOS A TRABAJAR*)

serie = RandomReal[{0, 1}, 100]; (*serie temporal*)
mMaxima = 6;
(*máximo valor de la dimensión de inmersión ('embedding dimension')*)
nivelConfianza = 95; (*nivel de confianza del test*)
nivelSignificacion = 1 - (nivelConfianza / 100);
(*nivel de significación del test*)
```

(*nivel de significación del test*)

```
n = Length[serie]; (*número de elementos de la serie temporal*)
```

(*PASOS 2 Y 3*)
(*CALCULAMOS EL ESTADÍSTICO DE CONTRASTE $W_{m,n}$*)
(*REALIZAMOS EL CONTRASTE DE HIPÓTESIS*)

(*A diferencia de como se hizo en la versión 1,
aquí vamos a realizar los pasos 2 y 3 de forma simultánea.*)
(*La motivación de esto es hacer que el programa sea algo más eficiente.*)

(*El cálculo del estadístico de contraste
$W_{m,n}$ requiere el cálculo previo de $T_{m,n}$ y $V_{m,n}$.*)

(*Los valores de $T_{m,n}$ y $V_{m,n}$ dependen del parámetro de distancia ϵ.*)
(*En Kanzler (1999) se recomienda tomar ϵ/σ=1.5 para valores de m bajos.*)

```
desviacionTipica = StandardDeviation[serie];
(*desviación típica de los elementos de la serie*)
epsilon = 1.5 * desviacionTipica; (*parámetro ϵ*)
```

(*El valor de $T_{m,n}$ viene dado por:*)

(*$T_{m,n}(\epsilon)=C_{m,n}(\epsilon)-C_{1,n-m+1}(\epsilon)^{m}$*)

(*Por otro lado, el valor de $V_{m,n}$ es:*)

(*${V_{m,n}}^{2}(\epsilon)=4\left[K_n^m+2\sum_{j=1}^{m-1}K_n^{m-j}C_n^{2j}+(m-1)^2C_{1,n}^{2m}-m^2K_nC_n^{2m-2}\right]$*)

(*Por tanto, hay que calcular antes $C_{m,n}$, $C_{1,n-m+1}$, C_n y K_n.*)

(*CALCULAMOS C_n*)

(*Representemos la serie temporal mediante $\{u_t\}$.*)
(*Denotemos u_t^m a la m-historia $(u_t,u_{t+1},...,u_{t+m-1})$.*)
(*La expresión matemática de C_n es:*)

(*$C_n(\epsilon)=\frac{1}{\binom{n}{2}}\sum_{s=1}^{n-1}\sum_{t=s+1}^{n}\chi_\epsilon\left(\left\|u_s^1-u_t^1\right\|\right)$*)

(*El valor de C_n no depende de la dimensión de inmersión m.*)
(*Al tiempo que se calcula C_n, definimos una matriz
que servirá de apoyo para el cálculo de $C_{m,n}$, $C_{1,n-m+1}$ y K_n.*)

```
matriz = Array[func1, {n, n}];
sumaCn = 0;

For[cont1 = 1, cont1 ≤ n - 1, cont1++,
```

```
   elementoSerie = serie[[cont1]];

   For[cont2 = cont1 + 1, cont2 ≤ n, cont2 ++,

    normaDiferencia = Abs[elementoSerie - serie[[cont2]]];

    If[normaDiferencia < epsilon,

     func1[cont1, cont2] = 1;
     sumaCn = sumaCn + 1,
     func1[cont1, cont2] = 0;

    ];

   ];

  ];

Cn = sumaCn / (n * (n - 1) / 2); (*valor de Cn*)

(*CALCULAMOS C1,n-m+1*)

(*Para calcular C1,n-m+1 utilizaremos
 la fórmula que se muestra a continuación:*)
```

$$(*C_{1,n-m+1}(\epsilon)=\frac{1}{\binom{n-m+1}{2}}\sum_{s=m}^{n-1}\sum_{t=s+1}^{n}\chi_\epsilon\left(\left\|u_s^1-u_t^1\right\|\right)*)$$

```
(*C1,n-m+1 sí depende del valor de m.*)
(*Esto implica que hay que hacer los cálculos para
 cada uno de los valores de la dimensión de inmersión.*)

vectorC1n = Array[func2, {mMaxima - 1}]; (*vector de los valores de C1,n-m+1*)
sumaAuxiliar = 0;

For[cont3 = 1, cont3 ≤ mMaxima - 1, cont3 ++,

   For[cont4 = cont3 + 1, cont4 ≤ n, cont4 ++,

    sumaAuxiliar = sumaAuxiliar + func1[cont3, cont4];

   ];

   func2[cont3] = (sumaCn - sumaAuxiliar) / ((n - cont3) * (n - cont3 - 1) / 2);

  ];

(*CALCULAMOS Kn*)

(*El cálculo de Kn se realiza con la fórmula:*)
```

$$(*K_n(\epsilon)=\frac{1}{\binom{n}{3}}\sum_{r=1}^{n-2}\sum_{s=1}^{n-1}\sum_{t=1}^{n}\chi_\epsilon\left(\left\|u_r^1-u_s^1\right\|\right)\chi_\epsilon\left(\left\|u_s^1-u_t^1\right\|\right)*)$$

(*Tal y como se puede observar en su expresión matemática,
K_n no depende de la dimensión de inmersión m.*)

```
sumaKn = 0;

For[cont5 = 1, cont5 ≤ n - 2, cont5++,

  For[cont6 = cont5 + 1, cont6 ≤ n - 1, cont6++,

    If[func1[cont5, cont6] == 1,

      For[cont7 = cont6 + 1, cont7 ≤ n, cont7++,

        sumaKn = sumaKn + func1[cont6, cont7];

       ];

     ];

   ];

 ];

Kn = sumaKn / (n * (n - 1) * (n - 2) / 6); (*valor de Kn*)
```

(*CALCULAMOS $C_{m,n}$*)

(*$C_{m,n}$ se calcula con la fórmula:*)

$$(*C_{m,n}(\epsilon)=\frac{1}{\binom{n-m+1}{2}}\sum_{s=1}^{n-m}\sum_{t=s+1}^{n-m+1}\chi_\epsilon\left(\left\|u_s^m-u_t^m\right\|\right)*)$$

(*$C_{m,n}$, al igual que ocurría con $C_{1,n-m+1}$, depende de m.*)
(*Por tanto, determinaremos su valor para cada uno de los valores de m.*)

```
vectorCmn = Array[func3, {mMaxima - 1}]; (*vector de los valores de Cm,n*)
nCmn = n;

For[m = 2, m ≤ mMaxima, m++,

  sumaCmn = 0;

  For[cont8 = 1, cont8 ≤ nCmn - 2, cont8++,

   For[cont9 = cont8 + 1, cont9 ≤ nCmn - 1, cont9++,
```

```
      auxiliar = func1[cont8, cont9] * func1[cont8 + 1, cont9 + 1];
      func1[cont8, cont9] = auxiliar;
      sumaCmn = sumaCmn + auxiliar;

     ];

   ];

   func3[m - 1] = sumaCmn / ((n - m + 1) * (n - m) / 2);
   nCmn = nCmn - 1;

  ];

(*Calculados los valores de C_{m,n}, C_{1,n-m+1},
C_n y K_n, ya podemos determinar T_{m,n} y V_{m,n}.*)

vectorTmn = Array[func4, {mMaxima - 1}]; (*vector de los valores de T_{m,n}*)
vectorVmn = Array[func5, {mMaxima - 1}]; (*vector de los valores de V_{m,n}*)

(*El estadístico de contraste se calcula con la expresión:*)

    (*W_{m,n}(ϵ) = Sqrt[n-m+1] T_{m,n}(ϵ)/V_{m,n}(ϵ) *)

vectorWmn = Array[func6, {mMaxima - 1}]; (*vector de los valores de W_{m,n}*)
vectorTest3 = Array[func7, {mMaxima - 1}];
(*vector que recoge los resultados del test para cada valor de m*)

For[m = 2, m ≤ mMaxima, m++,

   func4[m - 1] = vectorCmn[[m - 1]] - vectorC1n[[m - 1]]^m;

   func5[m - 1] = 2 * Sqrt[Kn^m + 2 * Sum[Kn^(m-i) * Cn^(2*i), {i, 1, m-1}] + (m - 1)^2 * Cn^(2*m) - m^2 * Kn * Cn^(2*m-2)];

   func6[m - 1] = Sqrt[n - m + 1] * vectorTmn[[m - 1]] / vectorVmn[[m - 1]];

   If[Abs[vectorWmn[[m - 1]]] ≤
     InverseCDF[NormalDistribution[0, 1], 1 - nivelSignificacion / 2],

    func7[m - 1] = 1,
    func7[m - 1] = 0;

   ];

  ];

"Representación gráfica de los elementos de la serie temporal"
ListPlot[serie]
"Estadístico de contraste"
N[vectorWmn]
"Resultado del test"
```

$$W_{m,n}(\epsilon)=\sqrt{n-m+1}\,\frac{T_{m,n}(\epsilon)}{V_{m,n}(\epsilon)}$$

$$\text{func5}[m-1] = 2\sqrt{Kn^m + 2\sum_{i=1}^{m-1} Kn^{m-i}\, Cn^{2i} + (m-1)^2\, Cn^{2m} - m^2\, Kn\, Cn^{2m-2}}$$

ANEXO 2

PROGRAMAS REALIZADOS CON MATLAB

En este anexo se presenta el código de los programas de más interés realizados en Matlab.

En concreto, los programas que se presentan son los correspondientes a los cálculos que hay que realizar para aplicar cada uno de los test estudiados. Se omiten los códigos de los archivos utilizados para la realización de la interfaz gráfica.

```
% MÉTODO 1
% TEST 1
%
% La referencia utilizada ha sido:
%
%   M. Matilla, M. Ruíz
%   A NON-PARAMETRIC INDEPENDENCE TEST USING PERMUTATION ENTROPY
%   Journal of Econometrics 144 (2008), 139-155
%
% Realizamos el test a través de los pasos que se exponen a
continuación:
%
% PASO 1: Introducimos la serie temporal con la que vamos a trabajar.
% PASO 2: Calculamos h (entropía de permutación).
% PASO 3: Realizamos el contraste de hipótesis.

function t1=test_1(serie_temp,m,nivel_confianza)

% PASO 1
% INTRODUCIMOS LA SERIE CON LA QUE VAMOS A TRABAJAR

serie=serie_temp; % serie temporal
T=length(serie); % número de elementos de la serie

% PASO 2
% CALCULAMOS h (ENTROPÍA DE PERMUTACIÓN) %

trozo_serie1=serie(1:m);
[trozo_ordenado1,permutacion1]=sort(trozo_serie1);
M_permutaciones=permutacion1;
Npermutaciones_diferentes=1;
Fabs_permutacion(1)=1;

for cont1=2:T-m+1

    trozo_serie=serie(cont1:cont1+m-1);
    [trozo_ordenado,permutacion]=sort(trozo_serie);

    testigo=0;

    for cont2=1:Npermutaciones_diferentes

        if(isequal(M_permutaciones(cont2,:),permutacion)==1)

            Fabs_permutacion(cont2)=Fabs_permutacion(cont2)+1;
            testigo=1;
            break

        end

    end

    if(testigo==0)

    Npermutaciones_diferentes=Npermutaciones_diferentes+1;
    M_permutaciones(Npermutaciones_diferentes,:)=permutacion;
    Fabs_permutacion(Npermutaciones_diferentes)=1;
```

```
    end

end

Frel_permutacion=Fabs_permutacion/(T-m+1);

H=0; % entropía de permutación

for cont3=1:Npermutaciones_diferentes

    incremento_entropia=-
(Frel_permutacion(cont3)*log(Frel_permutacion(cont3)));
    H=H+incremento_entropia;

end

% PASO 3
% REALIZAMOS EL CONTRASTE DE HIPÓTESIS

G=2*(T-m+1)*(log(factorial(m))-H); % estadístico de contraste

grados_libertad=factorial(m)-1;
chi_cuadrado=chi2inv(nivel_confianza/100,grados_libertad);

if(G<chi_cuadrado)

    t1=1;

else

    t1=0;

end

% Agregamos al resultado de salida los valores de H y G

auxiliar=t1;
t1=[H G auxiliar];
```

```
% MÉTODO 2
% TEST 2-1
%
% La referencia utilizada ha sido:
%
%   J. S. Cánovas, A. Guillamón
%   PERMUTATIONS AND TIME SERIES ANALYSIS
%   Chaos 19 (2009), 197-235
%
% Seguimos los siguientes pasos para hacer el test:
%
% PASO 1: Introducimos la serie temporal con la que vamos a trabajar.
% PASO 2: Calculamos H (entropía topológica).
% PASO 3: Realizamos el contraste de hipótesis.
%
% Aclaración:
% Lo que aquí denominamos entropía topológica no es, en sentido
estricto, el valor real de dicha entropía.
% Lo que en realidad representa H es el número de permutaciones
diferentes que aparecen en la serie.
% Hemos utilizado el nombre de entropía topológica porque este
concepto está directamente relacionado con el número de permutaciones
diferentes en la serie.

function t2_1=test_2_1(serie_temp,m,nivel_confianza)

% PASO 1
% INTRODUCIMOS LA SERIE TEMPORAL CON LA QUE VAMOS A TRABAJAR

serie=serie_temp; % serie temporal
T=length(serie); % número de elementos de la serie

% PASO 2
% CALCULAMOS H (ENTROPÍA TOPOLÓGICA)

trozo_serie1=serie(1:m);
[trozo_ordenado1,permutacion1]=sort(trozo_serie1);
M_permutaciones=permutacion1;
Npermutaciones_diferentes=1;
Fabs_permutacion(1)=1;

for cont1=m+1:m:T-m+1

    trozo_serie=serie(cont1:cont1+m-1);
    [trozo_ordenado,permutacion]=sort(trozo_serie);

    testigo=0;

    for cont2=1:Npermutaciones_diferentes

        if(isequal(M_permutaciones(cont2,:),permutacion)==1)

            Fabs_permutacion(cont2)=Fabs_permutacion(cont2)+1;
            testigo=1;
            break

        end

    end
```

```
    if(testigo==0)

    Npermutaciones_diferentes=Npermutaciones_diferentes+1;
    M_permutaciones(Npermutaciones_diferentes,:)=permutacion;
    Fabs_permutacion(Npermutaciones_diferentes)=1;

    end

end

H=Npermutaciones_diferentes;

% PASO 3
% REALIZAMOS EL CONTRASTE DE HIPÓTESIS

valores_limite=load(strcat('VECTORm',num2str(m),'nc',num2str(nivel_con
fianza),'.txt'));
valor_limite=valores_limite(fix(T/m));

if(H>=valor_limite)

    t2_1=1;

else

    t2_1=0;

end

% Agregamos al resultado de salida los valores de Tmn, Vmn y Wmn

auxiliar=t2_1;
t2_1=[H valor_limite auxiliar];
```

```
% MÉTODO 3
% TEST 3: TEST BDS
% VERSIÓN 1
%
% Las referencias utilizadas han sido, principalmente:
%
%   W.A. Brock, W.D. Dechert, J.A. Scheinkman, B. LeBaron
%   A TEST FOR INDEPENDENCE BASED ON THE CORRELATION DIMENSION
%   Econometric Reviews 15 (1996), 197-235
%
%   L. Kanzler
%   VERY FAST AND CORRECTLY SIZED ESTIMATION OF THE BDS STATISTIC
%   Department of Economics of Oxford University (1999)
%
% Para implementar el algoritmo, los pasos a seguir son:
%
% PASO 1: Introducimos la serie temporal con la que vamos a trabajar.
% PASO 2: Calculamos el estadístico de contraste Wmn.
% PASO 2.1: Calculamos Tmn.
% PASO 2.2: Calculamos Vmn.
% PASO 2.3: Calculamos Wmn.
% PASO 3: Realizamos el contraste de hipótesis.

function
t3=test_3(serie_temp,m,parametro_Fcaracteristica,nivel_confianza)

% PASO 1
% INTRODUCIMOS LA SERIE TEMPORAL CON LA QUE VAMOS A TRABAJAR

serie=serie_temp; % serie temporal
n=length(serie); % número de elementos de la serie
Ntrozos=n-m+1; % número de trozos en que se divide la serie

% PASO 2
% CALCULAMOS EL ESTADÍSTICO DE CONTRASTE Wmn

desviacion_tipica=std(serie(:)); % desviación típica de los elementos
de la serie
epsilon=parametro_Fcaracteristica*desviacion_tipica; %parámetro de
distancia

% PASO 2.1
% CALCULAMOS Tmn

suma_Cmn=0;

suma_C1n=0;

Ncombinaciones=Ntrozos*(Ntrozos-1)/2;

% CALCULAMOS Cmn

for cont1=1:Ntrozos-1

    trozo_serie_Cmn=serie(cont1:cont1+m-1);

    for cont2=cont1+1:Ntrozos

        diferencia_Cmn=trozo_serie_Cmn-serie(cont2:cont2+m-1);
```

```
        norma_diferencia_Cmn=norm(diferencia_Cmn,inf);

        if(norma_diferencia_Cmn<epsilon)

        suma_Cmn=suma_Cmn+1;

        end

    end

end

Cmn=suma_Cmn/Ncombinaciones; % valor de Cmn

% CALCULAMOS C1n

for cont3=m:n-1

    elemento_serie_C1n=serie(cont3);

    for cont4=cont3+1:n

        diferencia_C1n=elemento_serie_C1n-serie(cont4);
        norma_diferencia_C1n=abs(diferencia_C1n);

        if(norma_diferencia_C1n<epsilon)

        suma_C1n=suma_C1n+1;

        end

    end

end

C1n=suma_C1n/Ncombinaciones; % valor de C1n

Tmn=Cmn-(C1n)^m; % valor de Tmn

% PASO 2.2
% CALCULAMOS Vmn

suma_Cn=0;

suma_Kn=0;

suma_Vmn=0;

% CALCULAMOS Cn

for cont5=1:n-1

    elemento_serie_Cn=serie(cont5);

    for cont6=cont5+1:n
```

```
        diferencia_Cn=elemento_serie_Cn-serie(cont6);
        Vabsoluto_diferencia_Cn=abs(diferencia_Cn);

        if(Vabsoluto_diferencia_Cn<epsilon)

        suma_Cn=suma_Cn+1;

        end

    end

end

Cn=suma_Cn/(n*(n-1)/2); % valor de Cn

% CALCULAMOS Kn

for cont7=1:n-2

    elemento_serie_Kn_i=serie(cont7);

    for cont8=cont7+1:n-1

        elemento_serie_Kn_j=serie(cont8);

        diferencia_Kn_ij=elemento_serie_Kn_i-elemento_serie_Kn_j;
        Vabsoluto_diferencia_Kn_ij=abs(diferencia_Kn_ij);

        if(Vabsoluto_diferencia_Kn_ij<epsilon)

            for cont9=cont8+1:n

                diferencia_Kn_jk=elemento_serie_Kn_j-serie(cont9);
                Vabsoluto_diferencia_Kn_jk=abs(diferencia_Kn_jk);

                if(Vabsoluto_diferencia_Kn_jk<epsilon)

                    suma_Kn=suma_Kn+1;

                end

            end

        end

    end

end

Kn=suma_Kn/(n*(n-1)*(n-2)/6); % valor de Kn

% CALCULAMOS Vmn

for cont10=1:m-1

    incremento_suma_Vmn=(Kn^(m-cont10))*(Cn^(2*cont10));
```

```
    suma_Vmn=suma_Vmn+incremento_suma_Vmn;

end

Vmn=2*sqrt(Kn^m+2*suma_Vmn+(m-1)^2*Cn^(2*m)-m^2*Kn*Cn^(2*m-2)); %
valor de Vmn

% PASO 2.3
% CALCULAMOS Wmn

Wmn=sqrt(n)*Tmn/Vmn; % valor del estadístico de contraste Wmn

% PASO 3
% REALIZAMOS EL CONTRASTE DE HIPÓTESIS

if (abs(Wmn)<=norminv((1+nivel_confianza/100)/2,0,1))

    t3=1;

else

    t3=0;

end

% Agregamos al resultado de salida los valores de Tmn, Vmn y Wmn

auxiliar=t3;
t3=[Tmn Vmn Wmn auxiliar];
```

```
% MÉTODO 3
% TEST 3: TEST BDS
% VERSIÓN 2
%
% Las referencias utilizadas han sido, principalmente:
%
%   W.A. Brock, W.D. Dechert, J.A. Scheinkman, B. LeBaron
%   A TEST FOR INDEPENDENCE BASED ON THE CORRELATION DIMENSION
%   Econometric Reviews 15 (1996), 197-235
%
%   L. Kanzler
%   VERY FAST AND CORRECTLY SIZED ESTIMATION OF THE BDS STATISTIC
%   Department of Economics of Oxford University (1999)
%
% Para implementar el algoritmo, los pasos a seguir son:
%
% PASO 1: Introducimos la serie temporal con la que vamos a trabajar.
% PASO 2: Calculamos el estadístico de contraste Wmn.
% PASO 3: Realizamos el contraste de hipótesis.

function
t3=test_3_r(serie_temp,mMaxima,parametro_Fcaracteristica,nivel_confian
za)

% PASO 1
% INTRODUCIMOS LA SERIE TEMPORAL CON LA QUE VAMOS A TRABAJAR

serie=serie_temp; % serie temporal
n=length(serie); % número de elementos de la serie

% PASOS 2 Y 3
% CALCULAMOS EL ESTADÍSTICO DE CONTRASTE Wmn
% REALIZAMOS EL CONTRASTE DE HIPÓTESIS

desviacion_tipica=std(serie(:)); % desviación típica de los elementos
de la serie
epsilon=parametro_Fcaracteristica*desviacion_tipica; %parámetro de
distancia

% CALCULAMOS Cn

sumaCn=0;

for cont1=1:n-1

    elementoSerie=serie(cont1);

    for cont2=cont1+1:n

        normaDiferencia=abs(elementoSerie-serie(cont2));

        if(normaDiferencia<epsilon)

            matriz(cont1,cont2)=1;
            sumaCn=sumaCn+1;

        else
```

```
            matriz(cont1,cont2)=0;

        end

    end

end

Cn=sumaCn/(n*(n-1)/2); % valor de Cn

% CALCULAMOS C1n

sumaAuxiliar=0;

for cont3=1:mMaxima-1

    for cont4=cont3+1:n

        sumaAuxiliar=sumaAuxiliar+matriz(cont3,cont4);

    end

    vectorC1n(cont3)=(sumaCn-sumaAuxiliar)/((n-cont3)*(n-cont3-1)/2);
% vector de los valores de C1n
    vectorC1n_modificado(cont3)=vectorC1n(cont3)^(cont3+1);

end

% CALCULAMOS Kn

sumaKn=0;

for cont5=1:n-2

    for cont6=cont5+1:n-1

        if(isequal(matriz(cont5,cont6),1)==1)

            for cont7=cont6+1:n

                sumaKn=sumaKn+matriz(cont6,cont7);

            end

        end

    end

end

Kn=sumaKn/(n*(n-1)*(n-2)/6); % valor de Kn

% CALCULAMOS Cmn

nCmn=n;
```

```
for m=2:mMaxima

    sumaCmn=0;

    for cont8=1:nCmn-2

        for cont9=cont8+1:nCmn-1

            auxiliar=matriz(cont8,cont9)*matriz(cont8+1,cont9+1);
            matriz(cont8,cont9)=auxiliar;
            sumaCmn=sumaCmn+auxiliar;

        end

    end

    vectorCmn(m-1)=sumaCmn/((n-m+1)*(n-m)/2); % vector de los valores
de Cmn
    nCmn=nCmn-1;

end

% CALCULAMOS Tmn, Vmn y Wmn

for m=2:mMaxima

    vectorTmn(m-1)=vectorCmn(m-1)-vectorC1n_modificado(m-1); % vector
de los valores de Tmn

    suma_Vmn=0;

    for cont10=1:m-1

        incremento_suma_Vmn=(Kn^(m-cont10))*(Cn^(2*cont10));
        suma_Vmn=suma_Vmn+incremento_suma_Vmn;

    end

    vectorVmn(m-1)=2*sqrt(Kn^m+2*suma_Vmn+(m-1)^2*Cn^(2*m)-
m^2*Kn*Cn^(2*m-2)); % vector de los valores de Vmn
    vectorWmn(m-1)=sqrt(n-m+1)*vectorTmn(m-1)/vectorVmn(m-1); % vector
de los valores de Wmn

    if (abs(vectorWmn(m-1))<=norminv((1+nivel_confianza/100)/2,0,1))

        vectorTest3(m-1)=1;

    else

        vectorTest3(m-1)=0;

    end

end

t3=vectorTest3;
```

```
% PROGRAMA PARA REALIZAR ENSAYOS
%
% Este programa permite realizar los test 1, 2.1 y 3 a una misma
serie.
% Se pretende utilizar para hacer un análisis comparativo de los
resultados que se obtienen con cada uno de los métodos.
%
% Aclaración:
% Lo que aquí denominamos entropía topológica no es, en sentido
estricto, el valor real de dicha entropía.
% Lo que en realidad representa H es el número de permutaciones
diferentes que aparecen en la serie.
% Hemos utilizado el nombre de entropía topológica porque este
concepto está directamente relacionado con el número de permutaciones
diferentes en la serie.

t1=[0 0 0; 0 0 0; 0 0 0];
t2_1=[0 0 0; 0 0 0; 0 0 0];
t3=[0 0 0 0 0; 0 0 0 0 0; 0 0 0 0 0];
resultadoT1=[0 0 0; 0 0 0; 0 0 0];
resultadoT2=[0 0 0; 0 0 0; 0 0 0];
resultadoT3=[0 0 0 0 0; 0 0 0 0 0; 0 0 0 0 0];

for repeticiones=1:400

    x(1)=0.45;

    for i=2:600

        x(i)=x(i-1)+rand;

    end

    serie=x;
    m=[4 5 6];
    nivel_confianza=[90 95 99];
    n=length(serie);

    % TEST 1

    % CALCULAMOS h (ENTROPÍA DE PERMUTACIÓN) %

    for j=1:3

        trozo_serie1=serie(1:m(j));
        [trozo_ordenado1,permutacion1]=sort(trozo_serie1);
        M_permutaciones=permutacion1;
        Npermutaciones_diferentes=1;
        Fabs_permutacion(1)=1;

        for cont1=2:n-m(j)+1

            trozo_serie=serie(cont1:cont1+m(j)-1);
            [trozo_ordenado,permutacion]=sort(trozo_serie);

            testigo=0;

            for cont2=1:Npermutaciones_diferentes
```

```
                if(isequal(M_permutaciones(cont2,:),permutacion)==1)

                    Fabs_permutacion(cont2)=Fabs_permutacion(cont2)+1;
                    testigo=1;
                    break

                end

            end

            if(testigo==0)

            Npermutaciones_diferentes=Npermutaciones_diferentes+1;
            M_permutaciones(Npermutaciones_diferentes,:)=permutacion;
            Fabs_permutacion(Npermutaciones_diferentes)=1;

            end

        end

        Frel_permutacion=Fabs_permutacion/(n-m(j)+1);

        H1=0;

        for cont3=1:Npermutaciones_diferentes

            incremento_entropia=-
(Frel_permutacion(cont3)*log(Frel_permutacion(cont3)));
            H1=H1+incremento_entropia;

        end

        % REALIZAMOS EL CONTRASTE DE HIPÓTESIS (TEST 1)

        G=2*(n-m(j)+1)*(log(factorial(m(j)))-H1);

        grados_libertad=factorial(m(j))-1;

        for k=1:3


chi_cuadrado=chi2inv(nivel_confianza(k)/100,grados_libertad);

            if(G<chi_cuadrado)

                t1(k,j)=1;

            else

                t1(k,j)=0;

            end

        end
```

```
    end

    resultadoT1=resultadoT1+t1;

    % TEST 2-1

    for j=1:3

        % CALCULAMOS H (ENTROPÍA TOPOLÓGICA)

        trozo_serie1=serie(1:m(j));
        [trozo_ordenado1,permutacion1]=sort(trozo_serie1);
        M_permutaciones=permutacion1;
        Npermutaciones_diferentes=1;
        Fabs_permutacion(1)=1;

        for cont1=m(j)+1:m(j):n-m(j)+1

            trozo_serie=serie(cont1:cont1+m(j)-1);
            [trozo_ordenado,permutacion]=sort(trozo_serie);

            testigo=0;

            for cont2=1:Npermutaciones_diferentes

                if(isequal(M_permutaciones(cont2,:),permutacion)==1)

                    Fabs_permutacion(cont2)=Fabs_permutacion(cont2)+1;
                    testigo=1;
                    break

                end

            end

            if(testigo==0)

            Npermutaciones_diferentes=Npermutaciones_diferentes+1;
            M_permutaciones(Npermutaciones_diferentes,:)=permutacion;
            Fabs_permutacion(Npermutaciones_diferentes)=1;

            end

        end

        H2=Npermutaciones_diferentes;

        % REALIZAMOS EL CONTRASTE DE HIPÓTESIS (TEST 2-1)

        for k=1:3


valores_limite=load(strcat('VECTORm',num2str(m(j)),'nc',num2str(nivel_
confianza(k)),'.txt'));
            tam_valores_limite=length(valores_limite);

            if tam_valores_limite>=fix(n/(m(j)))
```

```
            valor_limite=valores_limite(fix(n/(m(j))));

        else

            valor_limite=valores_limite(tam_valores_limite);

        end

        if(H2>=valor_limite)

            t2_1(k,j)=1;

        else

            t2_1(k,j)=0;

        end

    end

end

resultadoT2=resultadoT2+t2_1;

% TEST 3

mMaxima=6;
desviacion_tipica=std(serie(:));
epsilon=1.5*desviacion_tipica;

% CALCULAMOS Cn

sumaCn=0;

for cont1=1:n-1

    elementoSerie=serie(cont1);

    for cont2=cont1+1:n

        normaDiferencia=abs(elementoSerie-serie(cont2));

        if(normaDiferencia<epsilon)

            matriz(cont1,cont2)=1;
            sumaCn=sumaCn+1;

        else

            matriz(cont1,cont2)=0;

        end

    end
```

```
    end

    Cn=sumaCn/(n*(n-1)/2);

    % CALCULAMOS C1n

    sumaAuxiliar=0;

    for cont3=1:mMaxima-1

        for cont4=cont3+1:n

            sumaAuxiliar=sumaAuxiliar+matriz(cont3,cont4);

        end

        vectorC1n(cont3)=(sumaCn-sumaAuxiliar)/((n-cont3)*(n-cont3-
1)/2); % vector de los valores de C1n
        vectorC1n_modificado(cont3)=vectorC1n(cont3)^(cont3+1);

    end

    % CALCULAMOS Kn

    sumaKn=0;

    for cont5=1:n-2

        for cont6=cont5+1:n-1

            if(isequal(matriz(cont5,cont6),1)==1)

                for cont7=cont6+1:n

                    sumaKn=sumaKn+matriz(cont6,cont7);

                end

            end

        end

    end

    Kn=sumaKn/(n*(n-1)*(n-2)/6);

    % CALCULAMOS Cmn

    nCmn=n;

    for m=2:mMaxima

        sumaCmn=0;

        for cont8=1:nCmn-2
```

```
            for cont9=cont8+1:nCmn-1

                auxiliar=matriz(cont8,cont9)*matriz(cont8+1,cont9+1);
                matriz(cont8,cont9)=auxiliar;
                sumaCmn=sumaCmn+auxiliar;

            end

        end

        vectorCmn(m-1)=sumaCmn/((n-m+1)*(n-m)/2); % vector de los
valores de Cmn
        nCmn=nCmn-1;

    end

    % CALCULAMOS Tmn, Vmn y Wmn
    % REALIZAMOS EL CONTRASTE DE HIPÓTESIS

    for m=2:mMaxima

        vectorTmn(m-1)=vectorCmn(m-1)-vectorC1n_modificado(m-1); %
vector de los valores de Tmn

        suma_Vmn=0;

        for cont10=1:m-1

            incremento_suma_Vmn=(Kn^(m-cont10))*(Cn^(2*cont10));
            suma_Vmn=suma_Vmn+incremento_suma_Vmn;

        end

        vectorVmn(m-1)=2*sqrt(Kn^m+2*suma_Vmn+(m-1)^2*Cn^(2*m)-
m^2*Kn*Cn^(2*m-2)); % vector de los valores de Vmn
        vectorWmn(m-1)=sqrt(n-m+1)*vectorTmn(m-1)/vectorVmn(m-1); %
vector de los valores de Wmn

        for j=1:3

            if (abs(vectorWmn(m-
1))<=norminv((1+nivel_confianza(j)/100)/2,0,1))

                matrizTest3(j,m-1)=1;

            else

                matrizTest3(j,m-1)=0;

            end

        end

    end

    t3=matrizTest3;
```

```
    resultadoT3=resultadoT3+t3;

end

resultadoT1=[100 100 100; 100 100 100; 100 100 100]-
resultadoT1*(100/repeticiones)
resultadoT2=[100 100 100; 100 100 100; 100 100 100]-
resultadoT2*(100/repeticiones)
resultadoT3=[100 100 100 100 100; 100 100 100 100 100; 100 100 100 100
100]-resultadoT3*(100/repeticiones)
```