Pruebas de bondad de generadores de aleatorios

750098M Simulación computacional

Contenido

Introducción

Los números aleatorios deben cumplir dos propiedades

1. Pruebas de uniformidad: Comprobar que los datos siguiente una distribución dada (Generalmente Uniforme U[0,1])

2. Prueba de independencia: Es determinar que los datos no tengan un patrón dado

Un buen generador debe cumplir (estadisticamente) estas dos propiedades

Introducción

Pruebas de uniformidad

Fix

1. Los números deben distribuirse uniformemente en el intervalo [0,1]. No deben ser densos en un intervalo y escasos en otros. Se parte el intervalo debe observarse el mismo comportamiento.
2. Un buen generador de pseudoaleatorios debe verificar si se producen datos aleatorios

Introducción

Pruebas de independencia

1. Los números pseudoaleatorios no pueden tener patrones definidos
2. Un ejemplo: Se tienen 100 números. Se encuentra que se tiene una secuencia creciente: 0, 0.1, 0.11, 0.12, 0.2, 0.21. Esta secuencia puede pasar una prueba de uniformidad sin problema pero no una de independencia. En este caso NO son considerados como pseudoaleatorios.

Introducción

Pruebas de uniformidad	Pruebas de independencia
Se determina si el computamiento de los datos está de acuerdo a una distribución dada (usualmente Uniforme entre 0 y 1) • Prueba Chi Cuadrado • Prueba de Kolmogorov-Smirnov	Se determina si no existen patrones o estructuras dentro de los números generados • Pruebas de series • Pruebas de corridas • Pruebas póker

Prueba de Chi cuadrado

Introducción.

1. Es una prueba de uniformidad que se aplica sobre datos generados. La idea es observar si siguen una distribución (Generalmente Uniforme U[0,1]

2. Está basada en la distribución chi-cuadrado con un grado de confianza dado. Por lo que debemos usar tablas.

Prueba de Chi cuadrado

Variables:

1. n: Número de datos
2. c: Número de clases.
3. gl: Grados de libertad
4. FE: Frecuencia esperada
5. FO: Frecuencia observada

$$c = \lceil \sqrt{n} \rceil$$

$$gl = c - 1$$

Prueba de Chi cuadrado

Proceso

1. Se clasifican los datos observados (FO) en c clases

2. Se determina la frecuencia esperada de cada clase. En este caso cómo trata de una distribución uniforme U[0,1]

$$FE = \frac{n}{c}$$

3. Se determina el valor calculado de Chi-cuadrado

$$\chi^2_{calc} = \sum_{i=1}^{c} \frac{(FE_i - FO_i)^2}{FE_i}$$

Prueba de Chi cuadrado

Proceso

4. Se determina el valor crítico de chi cuadrado con gl grados de libertad con un grado de confianza α dado. Para esto buscamos en la tabla.

5. Si nuestro valor de chi-cuadrado es menor o igual que el valor crítico (de la tabla) aceptamos la hipótesis que los datos se distribuyen uniformemente, en caso contrario rechazamos el generador.

Prueba de Chi cuadrado

Tabla de distribución Chi Cuadrado

gl υ	$\chi^2_{0.005}$	$\chi^2_{0.01}$	$\chi^2_{0.025}$	$\chi^2_{0.05}$	$t^2_{0.10}$
1	7.88	6.63	5.02	3.84	2.71
2	10.60	9.21	7.38	5.99	4.61
3	12.84	11.34	9.35	7.81	6.25
4	14.96	13.28	11.14	9.49	7.78
5	16.75	15.09	12.83	11.07	9.24
6	18.55	16.81	14.45	12.59	10.64
7	20.28	18.48	16.01	14.07	12.02
8	21.95	20.09	17.53	15.51	13.36
9	23.59	21.67	19.02	$\boxed{16.92}$	14.68
10	25.19	23.21	20.48	18.31	15.99
11	26.76	24.73	21.92	19.68	17.28
12	28.30	26.22	23.34	21.03	18.55
13	29.82	27.69	24.74	22.36	19.81
14	31.32	29.14	26.12	23.68	21.06
15	32.80	30.58	27.49	25.00	22.31
16	34.27	32.00	28.85	26.30	23.54

Prueba de Chi cuadrado

Ejemplo

El generador definido por

- $a = 106$
- $c = 1283$
- $m = 6075$

¿Para la prueba de uniformidad Chi cuadrado?

Considere un nivel de confianza $\alpha = 0.05$

Prueba de Chi cuadrado

Ejemplo

[0- 0,1) , [0,1 - 0,2) [0,2 - 0,3] ... [0,9 - 1.0]

1. Se general n = 100 datos y los clasificamos en

$$c = \lceil \sqrt{100} \rceil = 10 \text{ clases}$$

2. Se calcula la frecuencia esperada $FE = \frac{100}{10} = 10$

3. Se determina el valor calculado de $\chi^2_{calc} = 10.6$

4. Se determinan los grados de libertad $gl = c - 1 = 9$

5. Se busca en la tabla con 9 grados de libertad y nivel de confianza de 0.05 $\chi^2_{crit} = 16.92$

Prueba de Chi cuadrado

Ejemplo

Clase	FO	FE	$\frac{(FE-FO)^2)}{FE}$
[0-0.1)	11	10	0.1
[0.1-0.2)	6	10	1.6
[0.2-0.3)	16	10	3.6
[0.3-0.4)	9	10	0.1
[0.4-0.5)	7	10	0.9
[0.5-0.6)	11	10	0.1
[0.6-0.7)	10	10	0
[0.7-0.8)	9	10	0.1
[0.8-0.9)	15	10	2.5
[0.9-1)	6	10	1.6

100

$$\chi^2_{calc} = 10.6$$

Prueba de Chi cuadrado

Ejemplo

El generador definido por

- a = 106
- c = 1283
- m = 6075

¿Para la prueba de uniformidad Chi cuadrado?

Como se puede observar $$\chi^2_{calc} \leq \chi^2 crit$$

El generador no puede ser **rechazado** por esta prueba

Prueba de Chi cuadrado

Miremos otro ejemplo

Clase	FO	FE	$\frac{(FE-FO)^2)}{FE}$
[0-0.1)	14	10	1.6
[0.1-0.2)	6	10	1.6
[0.2-0.3)	16	10	3.6
[0.3-0.4)	6	10	1.6
[0.4-0.5)	7	10	0.9
[0.5-0.6)	14	10	1.6
[0.6-0.7)	7	10	0.9
[0.7-0.8)	8	10	0.4
[0.8-0.9)	16	10	3.6
[0.9-1)	6	10	1.6

$$\chi^2_{calc} = 17.4$$

Prueba de Chi cuadrado

Otro ejemplo

¿Para la prueba de uniformidad Chi cuadrado?

Como se puede observar

$$\chi^2_{calc} > \chi^2_{crit}$$

El generador es **rechazado** por esta prueba. El generador no ha pasado la prueba de **uniformidad**

Prueba de Kolmogorov-Smirvov

Introducción.

1. Permite calcular la distancia entre dos distribuciones de probabilidad y determinar estadísticamente si son la misma

2. En nuestro caso se van a comparar la distribución Uniforme y la que se obtiene a partir del generador de pseudoaleatorios. Es más confiable que la prueba de Chi-Cuadrado ya se considera la probabilidad acumulada

Prueba de Kolmogorov-Smirvov

Variables:

1. n: Número de datos
2. c: Número de clases. $$c = \lfloor \sqrt{n} \rfloor$$
3. gl: Grados de libertad $$gl = n$$
4. FO: Frecuencia observada
5. FOA: Frecuencia observada acumulada
6. PO: Probabilidad observada
7. POA: Probabilidad observada acumulada
8. PEA: Probabilidad esperada acumulada

Prueba de Kolmogorov-Smirvov

Grados de libertad (N)	$D_{0.10}$	$D_{0.05}$	$D_{0.01}$
1	0.950	0.975	0.995
2	0.776	0.842	0.929
3	0.642	0.708	0.828
4	0.564	0.624	0.733
5	0.510	0.565	0.669
6	0.470	0.521	0.618
7	0.438	0.486	0.577
8	0.411	0.457	0.543
9	0.388	0.432	0.514
10	0.368	0.410	0.490
11	0.352	0.391	0.468
12	0.338	0.375	0.450
13	0.325	0.361	0.433
14	0.314	0.349	0.418
15	0.304	0.338	0.404
16	0.295	0.328	0.392
17	0.286	0.318	0.381
18	0.278	0.309	0.371
19	0.272	0.301	0.363
20	0.264	0.294	0.356
25	0.24	0.27	0.32
30	0.22	0.24	0.29
35	0.21	0.23	0.27
Más de 35	$\frac{1.22}{\sqrt{N}}$	$\frac{1.36}{\sqrt{N}}$	$\frac{1.63}{\sqrt{N}}$

Prueba de Kolmogorov-Smirvov

Ejemplo: Confianza 0.05

Clase	FO	FOA	POA	PEA	|PEA-POA|
[0-0.1)	11	11	0.11	0.1	0.01
[0.1-0.2)	6	17	0.17	0.2	0.03
[0.2-0.3)	16	33	0.33	0.3	0.03
[0.3-0.4)	9	42	0.42	0.4	0.02
[0.4-0.5)	7	49	0.49	0.5	0.01
[0.5-0.6)	11	60	0.60	0.6	0
[0.6-0.7)	10	70	0.70	0.7	0
[0.7-0.8)	9	79	0.79	0.8	0.01
[0.8-0.9)	15	94	0.94	0.9	0.04
[0.9-1)	6	100	1	1	0

$$DM_{calc} = 0.04$$

Prueba de Kolmogorov-Smirvov

Resultados

1. Se tiene que $DM_{calc} = 0.04$
2. Se busca en la tabla y se tiene que

$$DM_{crit} = \frac{1.36}{\sqrt{100}} = 0.136$$

3. Se observa que $DM_{calc} \leq DM_{crit}$
4. Por lo tanto, se acepta la hipótesis de que el generador es bueno en cuanto a uniformidad

Pruebas de corridas

1. Una corrida es una sucesión de eventos similares
2. La longitud de una corrida es el número de eventos similares en la corrida

Ejemplo:

Se lanza una moneda 10 veces

C S S C C S S S C S

Las corridas son las sucesiones de eventos C y de S:

C S S C C S S S C S

Por lo tanto, tenemos 6 corridas.

Pruebas de corridas

Introducción

1. A partir de una secuencia de números pseudoaleatorios construimos una secuencia proveniente de un experimento de Bernoulli (Eventos con dos posibles resultados)

2. Se determina el valor de la variable aleatorio *número de corridas* que se obtiene a partir de la secuencia

3. Dado que se conoce esta variable aleatorio, se determina si el valor observado es cercano a la media de la variable aleatoria usamos la prueba de la Normal

Pruebas de corridas

Tipos de prueba:

1. Respecto al crecimiento o decrecimiento
2. Respecto a valores por encima o por debajo de la media
3. También se pueden comparar las longitudes con otras obtenidas aleatoriamente (si está por encima o por debajo)

Pruebas de corridas

Ejemplo:

0.08, 0.09, 0.23, 0.29, 0.42, 0.55, 0.58, 0.72, 0.89, 0.91,

0.84, 0.74, 0.73, 0.71, 0.53, 0.41, 0.31, 0.18, 0.16, 0.11,

0.01, 0.09,0.30, 0.32, 0.45, 0.47, 0.69, 0.74, 0.91, 0.95,

0.91, 0.88, 0.86, 0.68, 0.54, 0.38, 0.36, 0.29, 0.13, 0.12

¿Son independientes?

Pruebas de corridas

Miremos patrones de crecimiento o decrecimiento

* + + + + + + + + +

\- - - - - - - - - - -

\- + + + + + + + + +

\- - - - - - - - - -

¿Que observan en esta secuencia?

Pruebas de corridas

Observemos:

1. Es poco probable que una secuencia buena de números pseudoaleatorios tenga muchas o pocas corridas. El número mínimo de corridas es 1 y la máxima es N-1

2. El número de corridas es una variable aleatoria, que se puede describir estadísticamente.

Pruebas de corridas

Entonces:

1. La variable aleatoria se puede describir de acuerdo a su media **μ** y varianza **σ²**.

$$\mu = \frac{2n_1 n_2}{n_1 + n_2} + 1 \qquad \sigma^2 = \frac{2n_1 n_2 (2n_1 n_2 - n_1 - n_2)}{(n_1 + n_2)^2 (n_1 + n_2 + 1)}$$

2. n1 y n2 son la cantidad de valores de cada corrida. Se deben tener suficientes datos (N > 20) se puede asumir que la variable aleatoria tiene una distribución normal

Pruebas de corridas

Para el caso anterior:

1. De los 39 datos de la secuencia anterior se tienen 18 valores + y 21 valores -, entonces:

$$\mu = \frac{2n_1n_2}{n_1+n_2} + 1 = 20.38 \qquad \sigma = \sqrt{\frac{2n_1n_2(2n_1n_2 - n_1 - n_2)}{(n_1+n_2)^2(n_1+n_2+1)}} = 2.98$$

2. Esto significa que una buena secuencia debería tener en promedio 20.38 corridas con una desviación de 2.98. En total tuvimos 4 corridas, ¿Esto es suficientemente cercano para ser aceptado?

Pruebas de corridas

TABLA DE APOYO AL CÁLCULO DEL TAMAÑO DE UNA MUESTRA POR NIVELES DE CONFIANZA									
α	5%	6%	7%	8%	9%	10%	20%	37.7%	50%
Z	**1.96**	1.88	1.81	1.75	1.69	1.65	1.28	1	0.67

Pruebas de corridas

Para el caso anterior:

3 Dada una confianza del 0.05, el valor dado de la distribución normal es 1.96 entonces el rango:

$$[Inf_{crit}, Sup_{crit}] = [-1.96 * \sigma + \mu, 1.96 * \sigma + \mu]$$

4 De aquí obtenemos que el rango de corridas válido es [14.53,26.22], como tenemos 4 corridas se puede rechazar el generador, ya que no pasa la prueba de **independencia**

Pruebas de corridas

Para esa secuencia:

0.41, 0.68, 0.89, 0.94, 0.74, 0.91, 0.55, 0.62, 0.36 , 0.27,

0.19, 0.72, 0.75, 0.08, 0.54, 0.02, 0.01, 0.36, 0.16, 0.28,

0.18, 0.01, 0.95, 0.69, 0.18, 0.47, 0.23, 0.32, 0.82, 0.53,

0.31, 0.42, 0.73, 0.04, 0.83, 0.45, 0.13, 0.57, 0.63, 0.29

Comportamiento: (Decrecimiento / Crecimiento)

* + + + - + - + - -

- + + - + - - + - +

- - + - - + - + + -

- + + - + - - + + -

Pruebas de corridas

De los 39 datos se tienen 19 valores + y 20 valores -. En total tenemos 26 corridas

$$\sigma = \sqrt{\frac{2n_1 n_2(2n_1 n_2 - n_1 - n_2)}{(n_1+n_2)^2(n_1+n_2+1)}} = 3.08 \qquad \mu = \frac{2n_1 n_2}{n_1+n_2} + 1 = 20.5$$

$$[Inf_{crit}, Sup_{crit}] = [-1.96 * \sigma + \mu, 1.96 * \sigma + \mu] = [14.5, 26.5]$$

Dado que 26, está dentro del rango de aceptación. Esta secuencia pasa la prueba de independencia con respecto a crecimiento / decrecimiento

Pruebas de corridas

Para esa secuencia:

0.41, 0.68, 0.89, 0.94, 0.74, 0.91, 0.55, 0.62, 0.36 , 0.27,

0.19, 0.72, 0.75, 0.08, 0.54, 0.02, 0.01, 0.36, 0.16, 0.28,

0.18, 0.01, 0.95, 0.69, 0.18, 0.47, 0.23, 0.32, 0.82, 0.53,

0.31, 0.42, 0.73, 0.04, 0.83, 0.45, 0.13, 0.57, 0.63, 0.29

Comportamiento: (Con respecto a la media 0.5)

- \+ + + + + + + - -

- \+ + - + - - - - -

- \- + + - - - - + +

- \- + - + - - + + -

Pruebas de corridas

De los 39 datos se tienen 18 valores + y 22 valores -. En total tenemos 17 corridas

$$\sigma = \sqrt{\frac{2n_1n_2(2n_1n_2-n_1-n_2)}{(n_1+n_2)^2(n_1+n_2+1)}} = 3.09 \qquad \mu = \frac{2n_1n_2}{n_1+n_2} + 1 = 20.8$$

$$[Inf_{crit}, Sup_{crit}] = [-1.96 * \sigma + mu, 1.96 * \sigma + \mu] = [14.8, 26.9]$$

Dado que 17, está dentro del rango de aceptación. Esta secuencia pasa la prueba de independencia con respecto a crecimiento / decrecimiento

Pruebas de serie

Principios.

1. Estas pruebas miden si las subsecuencias tomando cada k-esimo dato, son uniformes en [0,1]
2. Si las subsecuencias son uniformes el generador pasa la prueba, en caso contrario se rechaza.

Pruebas de serie

Descripción

1. Agrupe los datos en k en k, interpretando cada grupo como un vector unitario k-dimensional $[0,1]^k$
2. Estos vectores deben ser uniformes en $[0,1]^k$
3. Si no son uniformes, los datos originales no son independientes.

Pruebas de serie

Variables:

1. n: Número de datos
2. El número de grupos es: $\frac{n}{k}$
3. k: Dimensión
4. c: Número total de clases $c = \lceil \sqrt{\frac{n}{k}} \rceil$
5. Como se debe dividir el intervalo [0,1] para cada dimensión se tiene para cada dimensión

$$\lceil \sqrt[k]{\sqrt{\frac{n}{k}}} \rceil$$

clases

Pruebas de serie

Proceso

1. Se cuentan las frecuencias FO en cada clase y se determina las frecuencia FE suponiendo uniforme multidimensional
2. Se aplica prueba chi-Cuadrado
3. Si pasa la prueba se acepta que los datos siguen una distribución uniforme multidimensional, entonces no hay evidencia de dependencia entre ellos.

Pruebas de serie

Se tiene 1200 datos pseudoaleatorios:

0.41 0.68 0.89 0.94 0.74 0.91 0.55 0.62 0.36 0.27 0.19 0.72 0.75 0.08 0.54 0.02 0.01 0.36 ...

Vamos aplicar una prueba bidimensional ($k = 2$), entonces se forman los 600 pares:

(0.41, 0.68), (0.89, 0.94), (0.74, 0.91), (0.55, 0.62), (0.36, 0.27), (0.19 0.72) ...

Pruebas de serie

Ejemplo:

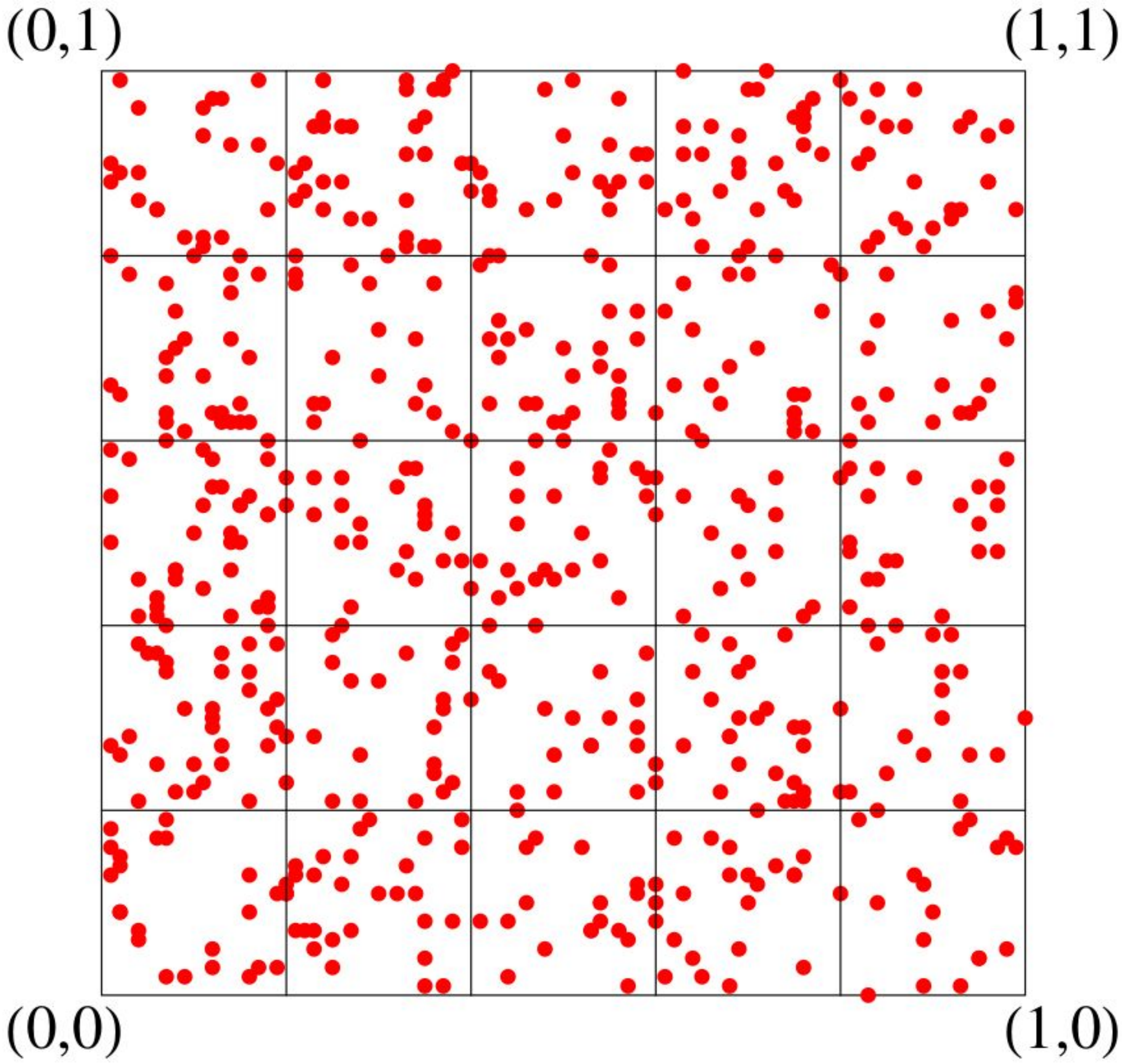
1. Se agrupan los datos en $c = \lceil \sqrt{600} \rceil = 25$ clases
2. Como tenemos dos dimensiones requieren

$$\lceil \sqrt[2]{\sqrt{600}} \rceil = 5$$

En cada dimensión:

Pruebas de serie

Ejemplo:



Pruebas de serie

Tablas de frecuencias para k = 2

	[0-0.2)	[0.2-0.4)	[0.4-0.6)	[0.6-0.8)	[0.8-1)
[0-0.2)	22	31	35	27	25
[0.2-0.2)	30	22	23	17	34
[0.4-0.6)	17	18	24	27	24
[0.6-0.8)	22	27	16	22	30
[0.8.-1)	18	17	23	19	26

La probabilidad teórica de cada celda es 1 / 25

Por eso la frecuencia esperada es 600 / 25 = 24 datos en cada celda. Aplicamos la prueba de Chi-Cuadrado

Pruebas de serie

Prueba chi-cuadrado $$\frac{(FE-FO)^2}{FE}$$

	[0-0.2)	[0.2-0.4)	[0.4-0.6)	[0.6-0.8)	[0.8-1)
[0-0.2)	0.17	2.04	5.04	0.38	0.04
[0.2-0.2)	1.5	0.17	0.04	2.04	4.17
[0.4-0.6)	2.04	1.5	0	0.38	0
[0.6-0.8)	0.17	0.38	2.67	0.17	1.5
[0.8.-1)	1.5	2.04	0.04	1.04	0.17

Se obtiene que el valor calculado es 29.19

Los grados de libertad, sería 25 - 1 = 24

El valor critico es 36.42

Este conjunto de datos pasa la prueba para k = 2

Pruebas de serie

Ahora con k = 3:

1. Ahora tenemos 400 grupos (1200/3)
2. Ahora se tiene en total $\lceil\sqrt{400}\rceil = \mathbf{20}$ clases
3. En cada dimensión se tienen $\lceil\sqrt[3]{\sqrt{400}}\rceil \approx \mathbf{3}$ clases por lo tanto, en total tenemos **27 clases** (son 3 por dimensión y tenemos 3 dimensiones)

Pruebas de serie

Tablas de frecuencias para k = 3

Para facilitar representación codificamos los subintervalos

1 = [0,1/3)

2 = [1/3,2/3)

3 = [2/3,1)

La probabilidad teórica de cada celda es 1 / 27

Por eso la frecuencia esperada es 400 / 27 = 14.8 datos en cada celda. Aplicamos la prueba de Chi-Cuadrado

Pruebas de serie

Tablas de frecuencias para k = 3

Clase	FO	Clase	FO	Clase	FO
1-1-1	15	2-1-1	15	3-1-1	13
1-1-2	11	2-1-2	19	3-1-2	24
1-1-3	15	2-1-3	20	3-1-3	16
1-2-1	13	2-2-1	11	3-2-1	10
1-2-2	18	2-2-2	16	3-2-2	7
1-2-3	22	2-2-3	24	3-2-3	8
1-3-1	20	2-3-1	6	3-3-1	11
1-3-2	18	2-3-2	12	3-3-2	9
1-3-3	17	2-3-3	15	3-3-3	11

Pruebas de serie

Ahora con k = 3:

1. Tenemos que $\chi^2_{calc} = 42.74$
2. El valor crítico con 26 grados de libertad y confianza 0.05 es 38.89
3. Como se tiene $\chi^2_{calc} > \chi^2_{crit}$ rechazamos la hipótesis de independencia de los datos y el generador debe ser descartado

Pruebas de póker

Descripción

1. Se buscan patrones que puedan ocurrir en secuencias aleatorias entre 0 y 9.
2. Se comparan estos patrones de acuerdo a una mano de póker suponiendo cartas entre 0 y 9
3. Se calculan las probabilidades de cada patrón FE y se observa la secuencia FO. Se aplica una prueba chi-cuadrado, si el generador no pasa esta prueba es rechazado por falta de independencia

Pruebas de póker

Ejemplo para k = 3, en total tenemos 10*10*10 posibles manos

1. Se toman los 3 primeros decimales de cada número

2. Se observa cuántos tienen los siguientes patrones:
   - 3 Cartas iguales = 10*1*1 / 1000 = 0.01
   - 2 iguales y una diferente 10*9*3 / 1000 = 0.27
   - 3 diferentes 10*9*8 /1000 = 0.72

Recordar técnicas de conteo del curso de Matemáticas Discretas II

Pruebas de póker

Ejemplo para k = 3, en total tenemos 10*10*10 posibles manos

Miremos la siguiente secuencia (n = 40)

0.959, 0.713, 0.178, 0.427, 0.299, 0.153, 0.087, 0.615, 0.188,
0.972, 0.239, 0.425, 0.372, 0.015, 0.316, 0.532, 0.216, 0.466,
0.808, 0.444, 0.084, 0.577, 0.166, 0.182, 0.904, 0.296, 0.854,
0.317, 0.051, 0.229, 0.299, 0.199, 0.185, 0.222, 0.954, 0.582,
0.283, 0.324, 0.913, 0.158.

Para fines didácticos vamos a suponer 3 clases, recuerden que $c = \lceil \sqrt{n} \rceil$ entonces usar más datos para una prueba

Pruebas de póker

Para el ejemplo

Clase	FO	FE	$\frac{(FE-FO)^2)}{FE}$
3 cartas iguales	2	0.01*40 = 0.4	6.4
2 iguales 1 diferente	10	0.24*40=10.8	1.6
3 diferentes	28	0.72*40=28.8	1.6

$$\chi_{calc} = 9.6$$

Para dos grados de libertad, se tiene que el valor crítico es 6.0, como se tiene $\chi_{calc} > \chi_{crit}$ este generador no pasa la prueba de independencia

Conclusiones

Pruebas de uniformidad	Pruebas de independencia
• Se busca que una secuencia de números se ajuste a una distribución de probabilidad dada (en nuestro caso Uniforme) • En términos generales se compara la secuencia con una distribución y se observa si se ajusta con respecto a un confianza dada	• Se busca que no existan patrones o regularidades dentro de una secuencia dada. • Se necesita realizar varias pruebas de independencia, ya que cada una observa diferentes patrones. No es posible que una prueba evalúe todas los posibles patrones o regularidad que existen

Conclusiones

El generador estándar mínimo (GEM) ha sido probado exitosamente en gran variedad de propiedades de uniformidad e independencia .

Los generadores de muchos lenguajes de programación están basados en este generador. Para evitar siempre se genere la misma secuencia (cada ejecución) se debe establecer una semilla diferente, la cual, usualmente depende del tiempo del sistema o el ID de proceso del programa en ejecución.