

## Modelo de utilidad: (Software)

Se excluye el uso del cero (0) debido a su naturaleza.

Este método consta de un arreglo de nueve (9) reglas o números que interactúan entre si siguiendo un patrón constante e inalterable de forma natural. Cabe mencionar la posibilidad de alterar dicho patrón para obtener resultados diferentes de dichas reglas (cifrado dinámico), más no es el caso para este modelo.

En un ecosistema base (9) es posible calcular, almacenar, predecir y / o desplazarse en ambas direcciones sin alterar el resultado esperado. Estas propiedades se implementan en este modelo para comprimir y extraer datos a partir de un fractal o arreglo numérico.

A continuación se muestra el orden o disposición lógica de dichos números basándonos en sus propiedades:

Lógica de Avance: [3,5,7,9,2,4,6,8,1]

Lógica de Reversa (principal): [6,4,2,9,7,5,3,1,8]

e observa que: la disposición de los mismos varia en lo referente a el orden habitual aceptado y estandarizado (1,2,3,4,5,6,7,8,9). Una vez comprendido lo anterior pasamos a enfocarnos en las propiedades primarias de dichos números o reglas, siendo las dos más fundamentales la constante de salto [K] y la paridad o espejo de dichos números [KPN], esto referente a el estado de menor entropía o de génesis. Dicho estado esta definido por las reglas del (9) y a partir de ahí se derivan todas las demás reglas (números). Las respectivas parejas formadas entre si por los mismos son:

Pares en lógica de Avance: [1,8] [2,7] [3,6] [4,5] [9,9]

Pares en lógica de Reversa: [8,1] [7,2] [6,3] [5,4] [9,9]

$$\frac{(9)(n)(n1)}{\downarrow} \\ (n2)$$

$$Kpn\sim(n) = (9 - n) = [K+-]$$

$$[K+] = \overrightarrow{\text{Diff}} (Kpn\sim(n) + 1 \text{ ---> } 9)$$

$$[K-] = (n)$$

$\overrightarrow{[ss]}$

ejemplo 1:

$\frac{(9)32}{122}$	$\frac{(9)35}{125}$	$\frac{(9)37}{127}$
$\frac{32}{5}$	$\frac{35}{8}$	$\frac{37}{1}$

ejemplo 2:

$\frac{(9)72}{162}$	$\frac{(9)75}{165}$	$\frac{(9)77}{167}$
$\frac{72}{9}$	$\frac{75}{12}$	$\frac{77}{14}$
	$\frac{12}{3}$	$\frac{14}{5}$

entonces:

$$\text{if: } (n2) < Kpn\sim(n): (n1) = (n2 - n) \text{ ---> } [K-]$$

$$\text{if: } (n2) > Kpn\sim(n): (n1) = (n2 + Kpn\sim(n)) \text{ ---> } [K+]$$

Ver tabla de desplazamiento (documento anexo)

## Concepto matemático de una neurona de calculo:

dado que todas las reglas o números están relacionados con su base (9) es posible extraer la totalidad de los mismos partiendo de uno de ellos. A excepción de (3, 6) y la propia base en si. Esto dado que esos tres valores integran la base.

Ejemplo:

$$1 = (Kpn \sim 1) = (8) \rightarrow (8 - 1) = (7) \rightarrow (Kpn \sim 7) = (2) \rightarrow (7 - 2) = (5) \rightarrow (Kpn \sim 5) = (4)$$

para obtener de lo anterior los valores 3,6,9 podemos utilizar:

Retorno a (9)

$$(8 - 5) = 3; (7 - 4) = 3; (5 - 2) = 3; (4 - 1) = 3$$

$$(8 - 2) = 6; (7 - 1) = 6$$

Avance a (9)

$$(1 + 2) = 3; (3 + 9) = (12 \rightarrow) = 3; (4 + 8) = (12 \rightarrow) = 3; (5 + 7) = (12 \rightarrow) = 3; (6 + 6) = (12 \rightarrow) = 3$$

$$(1 + 5) = 6; (2 + 4) = 6; (3 + 3) = 6; (6 + 9) = (15 \rightarrow) = 6; (7 + 8) = (15 \rightarrow) = 6$$

Avance a (9)

$$(1 + 8) = 9; (2 + 7) = 9; (3 + 6) = 9; (4 + 5) = 9$$

Retorno a (9)

$$(1 - 1) = 9; (2 - 2) = 9; (3 - 3) = 9; (4 - 4) = 9; (5 - 5) = 9; (6 - 6) = 9; (7 - 7) = 9; (8 - 8) = 9;$$

$$(9 - 9) = 9$$

En este método de calculo no se adicionan o restan valores, lo que se computa o calcula son los desplazamientos y la dirección de los mismos (avance, retorno) siendo posible retornar en dos formas distintas dependiendo cual regla (numero) inicie o formule la petición de desplazamiento.

Ejemplo:

$$(8 - 5) = 3; (8 - (5 \rightarrow 14)) = 6$$

En el ejemplo anterior expandimos el valor (5) para obtener 14, partiendo de esto restamos (8) para obtener (6). Es importante considerar este punto para definir cual regla realiza la petición de desplazamiento ya que en reversa podemos obtener dos valores distintos validos y correalacionados con los factores de inicio de calculo.

U4:

Una vez comprendido lo anterior podemos armar nuestra neurona de calculo siendo esta una (U4) (unsigned) de 4 elementos o valores (a,b,c,d). Si intentamos crear una neurona de menos de 3 elementos obtendríamos resultados ambiguos para los mismos factores de calculo. Es decir tendríamos colisiones de valor o más de un resultado valido.

Toda neurona antes de iniciar el proceso de calculo o integrarse a los demás componentes se considera U4, en función de la finalidad o uso que se desee se puede ecualizar la misma para ordenar los valores o indicar directamente el uso o modo de calculo que se va a emplear.

En este ejemplo de compresión de datos para obtener un fractal o resumen matemático de sus componentes o factores iniciales hemos utilizado modo de calculo R4 para cargar la información, LB para el procesamiento y transmisión (herencia) de la misma y SH para obtener el resultado final.

Hexa – C:

La neurona (Hexa -C) corresponde a un modelo U6 compuesto por una R4 y un conector de herencia el cual se desplaza al interior de esta para fungir como elemento de salida de datos o herencia a la siguiente neurona de calculo.

En otras publicaciones se ahondara en los distintos tipos de neuronas, así como el uso de las mismas según sea el sentido de desplazamiento.

comprendiendo estos conceptos podemos diseñar ecuaciones o modelos matemáticos para fines de predicción, análisis, cuantificación y demás. Siendo el objetivo primario de esta publicación la utilización de lo anterior con fines de compresión de datos(modelo de utilidad: software) en un fractal matemático que hereda de manera vinculante las propiedades de todos los elementos que componen el mismo. De esta forma diseñamos una ecuación para descomponer el resultado final de dicho calculo en los componentes o factores primarios así como respetar o mantener su correlación y / o ubicación con respecto a los demás miembros del calculo inicial. Esta ultima propiedad es fundamental a la hora de revertir o descomprimir un fractal, ya que nos devuelve los datos de la misma forma en a que fueron introducidos. Siendo lo anterior coherente y por lo tanto (legible) tanto para un dispositivo o software así como para humanos.

En este método (publico) se omiten algunos procedimientos por razones de copyright (c) y de aprovechamiento comercial.

Julio C. Gómez.

Todos los derechos reservados (c)