
-PROYECTO INTEGRADOR-

SIMULACIÓN

Comisión:4k10

Integrantes:

- Falabella, Sol
- Illanes, Daniela
- Luconi, Martina
- Vicente, Julieta

Índice

SIMULACIÓN.....	1
Selección del Tema.....	3
Definición de objetivos.....	3
Identificación de las etapas del modelo (Vensim).....	3
Formulación del problema.....	3
Definición del sistema.....	4
Formulación del modelo.....	5
Colección de datos.....	5
Implementación del modelo en la computadora.....	6
Verificación.....	9
Validación.....	9
Diseño de experimentos.....	9
Análisis de resultados.....	9
Identificación de las etapas del modelo (Python).....	12
Formulación del problema.....	12
Definición del sistema y alcance del mismo.....	12
Formulación del modelo. Descripción de las herramientas a utilizar.....	12
Implementación del modelo en la computadora.....	13
Descripción del Uso y Manejo de la Aplicación:.....	16
Verificación del código.....	16
Validación.....	16
Diseño de experimentos.....	16
Conclusiones.....	17
Documentación.....	18

Selección del Tema

El tema elegido fue las redes sociales y el ingreso de las redes en cuanto a la cantidad de publicaciones monetizadas. Las publicaciones del sistema se dividen en normales o no monetizadas y en publicaciones monetizadas, que son las que generan ingresos.

Para todas las variables cuyo valor sea aleatorio, la generación de los números estará hecha con python, usando un generador congruencial mixto, que luego será evaluado usando el método de la chi cuadrada para verificar la independencia y aleatoriedad de los números obtenidos.

Como el modelo de simulación planteado recibe externamente los datos de las variables cuyos valores dependen de la aleatoriedad. El modelo de python plantea la generación de números aleatorios y la validación de aleatoriedad de los mismos, para usar estos datos como fuente aleatoria de algunas variables como los ingresos.

Definición de objetivos

Nuestro objetivo es simular el modelo planteado usando el software Vensim, y poder visualizar si las ganancias de la empresa, haciendo énfasis en las obtenidas por las publicaciones pagas, son suficientes para solventar los gastos y poder obtener un margen positivo de ingresos.

También implementar un generador de números pseudoaleatorios basado en el método congruencial lineal mixto, validar su aleatoriedad utilizando la prueba de Kolmogorov-Smirnov utilizando el lenguaje de programación Python y utilizar esta secuencia de números generados para las variables del modelo que lo requieran.

Identificación de las etapas del modelo (Vensim)

Formulación del problema

El modelo de simulación planteado consta de analizar una red social y poder ver cómo fluctúan los usuarios activos e inactivos, las conversiones de un usuario normal a un tipo de empresa y ver cómo todo esto influye en las ganancias que se tienen por las publicaciones pagadas por las empresas para publicitarlas. Se cuenta con un tiempo de aproximadamente dos semanas y media para desarrollar el modelo, del que se espera luego poder definir cuales son las ganancias, ganancias por publicaciones, y la variación de usuarios y empresas a través del tiempo.

El modelo recibe externamente los datos de las variables cuyos valores dependen de la aleatoriedad pero luego el usuario no debe introducir más datos ya que el resto de valores se encuentran en el programa.

El modelo realizado tiene como finalidad la simulación del sistema ya existente para ver su comportamiento y en función de eso que las personas implicadas en el área de redes sociales vean cómo pueden optimizar sus pagos y demás actividades para obtener un mejor rendimiento

Las variables de interés principales son:

- Los usuarios activos, ya que el modelo recibe como flujo de entrada los nuevos usuarios que comenzarán a consumir el contenido no monetizado. Estos a su vez, pueden dejar de utilizar la red social o se pueden convertir en empresas.
- Las empresas son los usuarios activos que se convirtieron para generar contenido monetizable, que generarán ganancias para la red social.
- Los ingresos generados o totales, son aquellos que provienen de dos fuentes: los ingresos extras y los generados por publicidad. Los primeros hacen referencia a ganancias por publicidad y patrocinio, como por ejemplo: publicaciones pagadas por creadores de contenido, donaciones de usuarios que apoyan la plataforma, comisiones por ventas generadas a través de enlaces afiliados, entre otros. Mientras que los segundos son aquellos que provienen del contenido monetizado que generan las empresas. Dependiendo el tipo de contenido, se asignará un valor de publicidad que le dará un determinado número de visualizaciones.
- El contenido total proviene de 2 categorías: contenido monetizable y no monetizable. El primer tipo de contenido no genera ganancias, y es utilizado por los usuarios para disfrute personal. Por otro lado, el segundo tipo de contenido es visualizado por los usuarios activos pero es generado por empresas. Todos ellos pueden ser eliminados en cualquier momento.

Definición del sistema

Frontera del sistema:

- **Incluye:**
 - Usuarios nuevos y activos
 - Contenido total, contenido monetizable y contenido eliminado
 - Visualizaciones (normales y por publicidad)
 - Tasas de publicación (paga y no paga)
 - Tasa de inactividad y tasa de eliminación de publicaciones
 - Ingresos (por publicidad y extra)
 - Gastos (costos fijos y variables)
 - Empresa que gestiona estos elementos
- **Interacciones con el medioambiente:**
 - Incremento en el valor de la publicidad con el tiempo
 - Cambio en las tasas de usuarios nuevos e inactivos

Formulación del modelo

Aspectos relevantes:

- **Para la red social (desde una perspectiva de ingresos):**
 - La cantidad de usuarios activos afecta directamente las visualizaciones de anuncios y, por ende, los ingresos por publicidad.
 - El contenido monetizable y no monetizado afecta el total de visualizaciones y la capacidad de generar ingresos.
 - Las tasas de publicación y eliminación de publicaciones afectan la cantidad de contenido disponible en la plataforma.
 - Gastos y costos (fijos y variables) determinan la rentabilidad de la red social.

Modelo inicial:

- **Estructura básica:**
 - Un stock de "Usuarios Activos" que se incrementa con la tasa de usuarios nuevos y disminuye con la tasa de inactividad.
 - Un stock de "Contenido Total" que se divide en "Contenido Monetizable" y "Contenido no Monetizado".
 - Ingresos generados a partir de las visualizaciones de anuncios, que dependen del contenido monetizable y las visualizaciones por publicidad.
 - Un componente de gastos que incluye costos fijos y variables.

Colección de datos

En nuestro estudio se dan por válidos los siguientes valores:

- Tasa de nuevos usuarios: 0.45
- Tasa de inactividad: 0.17
- Tasa de conversión: 0.37
- Tasa de publicación no paga: 0.52
- Tasa de publicación paga: 0.48
- Tasa de eliminación de publicaciones: 0.23
- Costos fijos: 800000 dólares

Además, se definen las variables de nivel de la siguiente manera:

- Usuarios activos en instante inicial: 500000 personas
- Empresa en instante inicial: 200000 empresas
- Contenido total en instante inicial: 100000 publicaciones
- Ingreso Total en instante inicial: 900000 dólares

Los datos para la simulación fueron obtenidos de observaciones directas del sistema y de registros de datos históricos del mismo.

Implementación del modelo en la computadora

Para esta etapa elegimos el software Vensim, ya que con él podemos modelar el sistema dinámico de “funcionamiento de una red social” y analizar su comportamiento a lo largo del tiempo. Se utilizó para la construcción del diagrama causal y de simulación, permitiéndonos analizar no solo las relaciones entre las variables, sino que también la simulación a través de un modelo matemático basado en ecuaciones diferenciales.

Pasos de implementación:

1. Definir los stocks (usuarios activos, contenido total, contenido monetizable).
2. Establecer los flujos (tasas de usuarios nuevos, inactivos, eliminación de publicaciones, publicación paga y no paga).
3. Implementar las variables auxiliares (visualización normal, por publicidad, tasas de inactividad, eliminación de publicaciones).
4. Configurar las ecuaciones de ingresos y gastos.
5. Conectar los elementos con flechas de influencia según el diagrama proporcionado.

Diagrama de simulación del proyecto

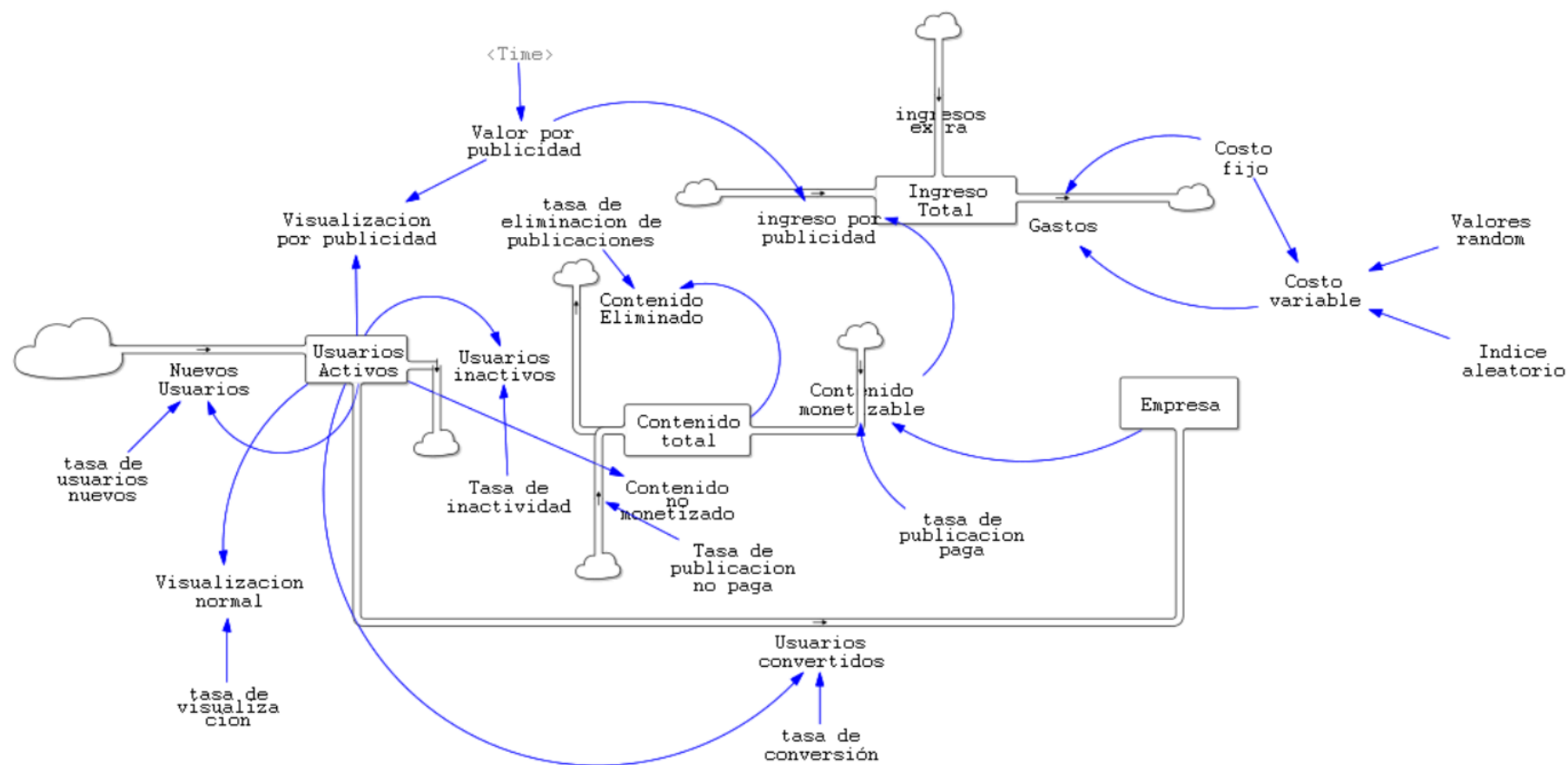
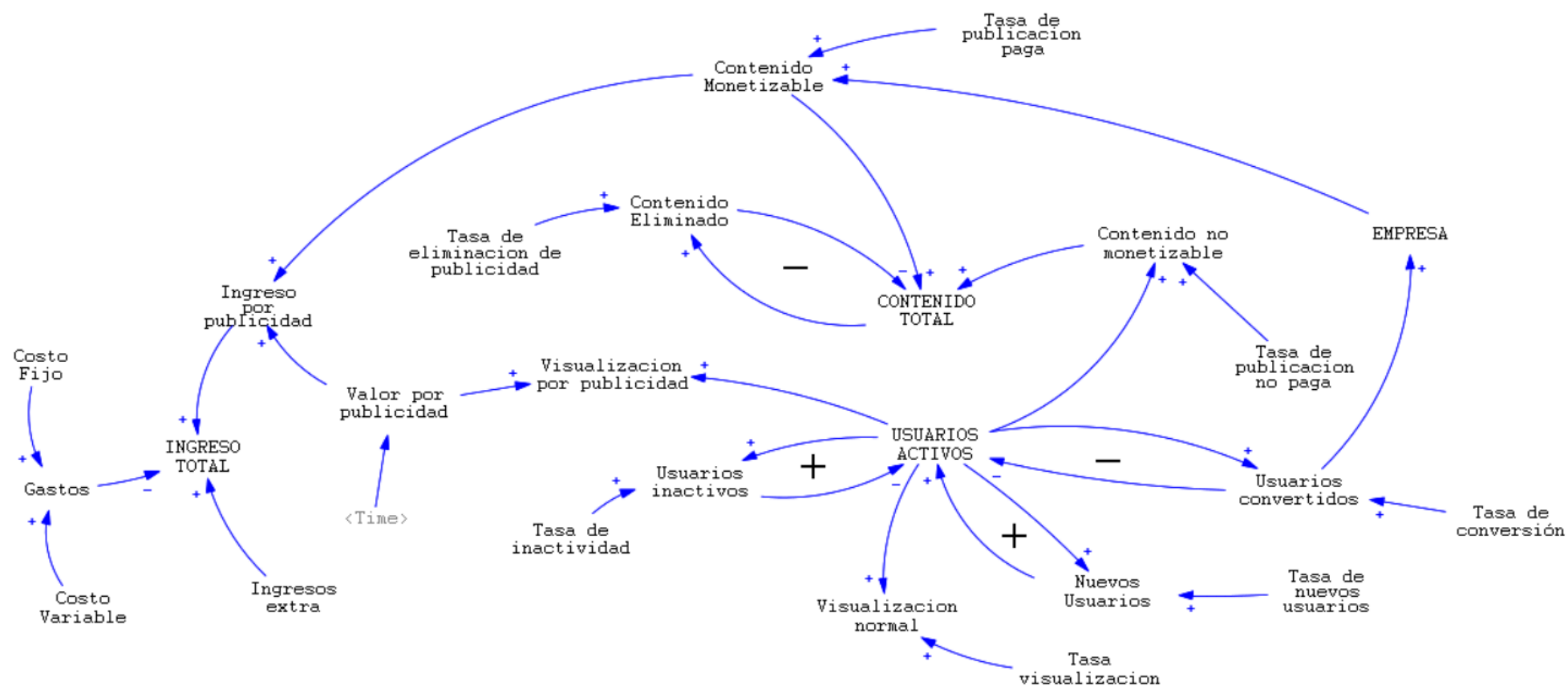


Diagrama causal del proyecto



Verificación

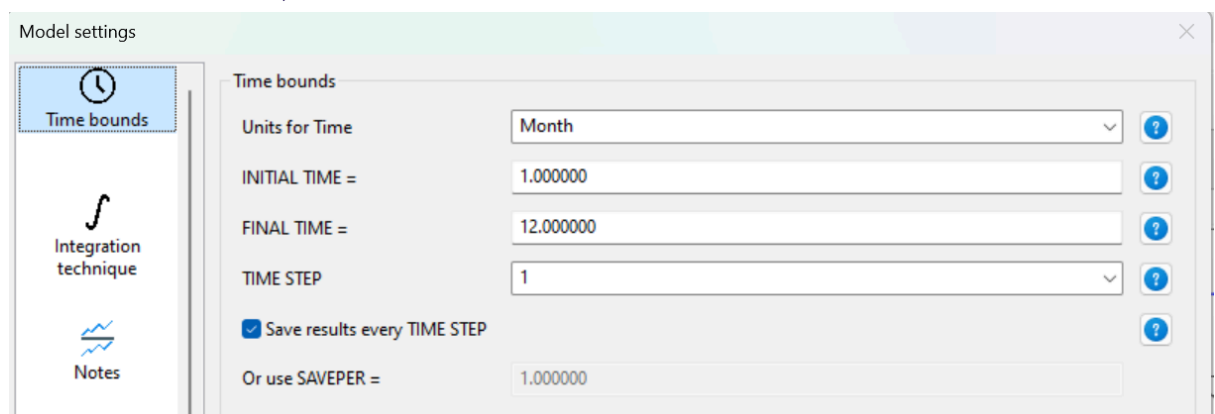
En esta etapa resolvimos los problemas de desbordamiento, ecuaciones inconsistentes y errores sintácticos. Recibimos ayuda de la herramienta de debug de Vensim "Check Model".

Validación

En esta etapa revisamos los datos obtenidos de la simulación y comparamos las gráficas con lo que esperábamos. En nuestro caso en primera instancia vimos que los costos generaban una gráfica que no era la esperada, se hizo una revisión sobre esas variables y encontramos un error en el planteo de la fórmula que no figuró en la verificación por que era un error de lógica ya que estaban mal relacionadas las variables. Una vez corregido, al volver a realizar el experimento obtuvimos los gráficos esperados.

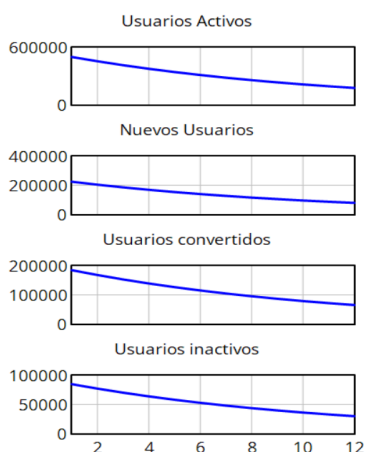
Diseño de experimentos

En nuestro estudio, la simulación comienza en el año 2024 y se extiende hasta 12 meses después. Durante la simulación, el paso del tiempo se mantendrá en intervalos mensuales, avanzando de mes en mes.



Análisis de resultados

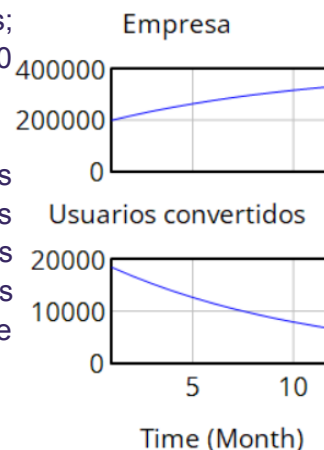
Usuarios



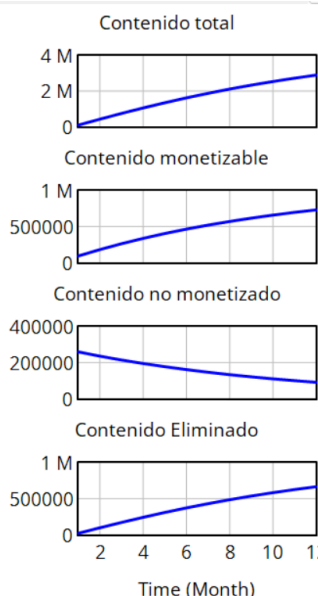
Podemos observar cómo los usuarios en el momento inicial son una gran cantidad, pero luego mientras pasan los meses del año simulado, vemos como bajan los valores tanto de usuarios activos, como de inactivos, y nuevos usuarios. Los usuarios que se convierten del tipo "usuario normal" al tipo "empresa" (que son aquellos usuarios que pueden pagar para publicitar sus publicaciones) tienden a disminuirse en la cantidad de tiempo, pero como se puede observar luego en la gráfica de empresa - usuarios convertidos podemos ver que la cantidad de empresas crece en relación con su valor inicial.

Los valores al iniciar el año son: 500,000 usuarios activos; 225,000 nuevos usuarios; 185,000 usuarios convertidos; 85,000 usuarios inactivos y 200,000 empresas.

Al finalizar los doce meses de simulación, tenemos que los usuarios activos son 177,184; los nuevos usuarios 79,733; los usuarios convertidos 65,558; los usuarios inactivos 30,122 y las empresas 1,52 Millones. Lo que nos demuestra que la tasa de conversión es realmente alta y la mayoría de usuarios de la red social son de este último tipo.



Contenido de la red social

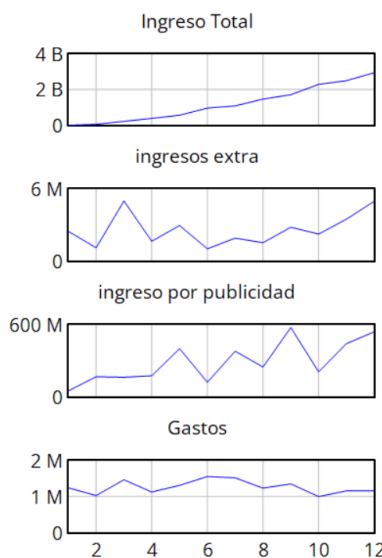


Podemos observar que, como vimos en el punto anterior, las empresas al ser un número ampliamente superior al de los usuarios normales, el contenido monetizable tiende a crecer, mientras que los usuarios regulares al disminuir en el tiempo hace que decrezcan la cantidad de publicaciones no monetizadas. El crecimiento de contenido monetizable es vastamente mayor al del regular, por lo que el contenido total (cuyas entradas son estos tipos de contenido) termina con una tendencia a crecer, como así también la cantidad de contenido eliminado, ya que este último valor crece a medida que crecen la cantidad de publicaciones de la red social.

Los valores al iniciar el año son 100,000 contenido total; 96,000 el contenido monetizable; 260,000 el contenido no monetizado y 23,000 el contenido eliminado.

Al finalizar el año el contenido total es 2,9 Millones; 733,023 el contenido monetizable; 92,136 el contenido no monetizado y 667,186 el contenido eliminado.

Ingresos

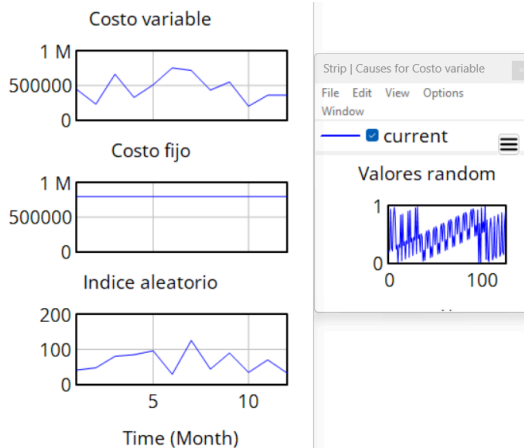


El ingreso total es una variable de nivel cuyo valor depende de los ingresos extra, y los ingresos por publicidad; y ambos de estos valores dependen de la aleatoriedad (tanto porque su valor cambia a través del tiempo o porque dependen en efecto de una función aleatoria), debido a esto, los dos tipos de ingreso tienen ese comportamiento y generan una función con varios máximos y mínimos relativos. Sin embargo, los ingresos totales tienen una tendencia al aumento (siendo el final considerablemente superior al inicial) mientras que los gastos varían pero a diferencia de otros gráficos analizados su comportamiento es más constante.

Al iniciar el año los valores son: \$900,000 ingreso total; \$2,4 Millones el ingreso extra; \$49,8 Millones el ingreso por publicidad y 1,24 Millones el gasto total.

Al finalizar el año el ingreso total es \$2,9 Billones; \$4,9 Millones el ingreso extra; \$540 Millones el ingreso por publicidad y \$1,16 Millones el gasto total.

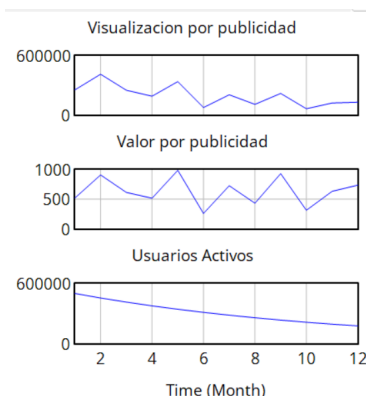
Costos fijo y variable



Esta variable resulta de interés ya que los valores de costo variable son una proporción del costo fijo (cuyo valor se mantiene constante en \$800,000) y esa proporción está dada por los números aleatorios entre 0 y 1 generados en python usando el método Kolmogorov-Smirnov. Podemos observar que los costos variables demuestran en el gráfico el comportamiento estocástico determinado por el índice aleatorio y por los valores random (que tienen en una tabla lookup los valores introducidos por nosotras).

Al iniciar el año el costo variable es de \$443,368; y al finalizarlo el valor es de \$361,587.

Visualizaciones



Los gráficos nos demuestran que las visualizaciones por publicidad caen, esto es debido a que los usuarios que son el objetivo de estas visualizaciones decrecen a través del tiempo. Sin embargo el valor por publicidad, tiene una relativa normalidad en cuanto al hecho de que no tiene un crecimiento o decrecimiento marcado, sino que es notable que depende del tiempo en el que nos encontremos para obtener su valor.

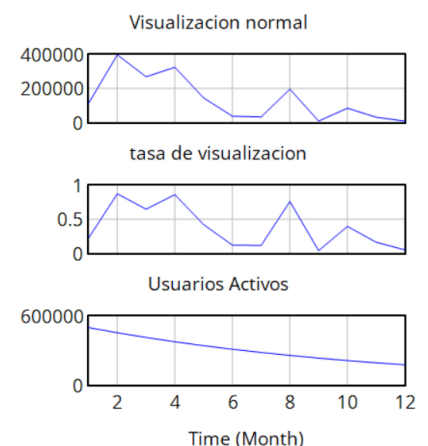
Al iniciar el año las visualizaciones por publicidad son de 253,698; mientras que el valor por publicidad es de \$519,711.

Al finalizar el año el valor de la visualización por publicidad se encuentra en 129,494; y el valor por publicidad pasa a ser \$737,242.

En cuanto a las visualizaciones normales, es aleatorio pues una publicación puede ser vista por el total de usuarios o por solamente uno. Podemos observar que mientras disminuye la cantidad de usuarios activos también lo hace la visualización normal.

Al iniciar el año la visualización es de 112,787 y la tasa es de 0,23%.

Al finalizar el año, la visualización es 9,287 y la tasa de 0,05%.



Identificación de las etapas del modelo (Python)

Formulación del problema.

El problema consiste en generar una secuencia de números pseudoaleatorios utilizando un generador congruencial lineal y validar si esta secuencia sigue una distribución uniforme mediante la prueba de Kolmogorov-Smirnov.

Estos números luego serán utilizados para algunas variables de interés en nuestro sistema de simulación de Redes Sociales.

Definición del sistema y alcance del mismo.

Simulamos un generador de números pseudoaleatorios utilizando el método congruencial lineal mixto y un test de Kolmogorov-Smirnov para la validación de la aleatoriedad de la sucesión generada interactuando con el usuario para el ingreso de parámetros solicitados.

Alcance del Sistema:

- Implementación del generador.
- Validación estadística de la secuencia generada.
- Interacción con el usuario para ingreso de parámetros y nivel de significancia para la prueba KS.

Formulación del modelo. Descripción de las herramientas a utilizar.

Modelo del Generador:

- Utilizar el método congruencial lineal mixto para generar números pseudoaleatorios: $x_{n+1}=(a*x_n+b)\bmod m$ y utilizar la prueba de Kolmogorov-Smirnov para comparar la secuencia generada con una distribución uniforme.

Software utilizado y librerías:

- El lenguaje de programación Python para la implementación.
- Librería **math**: Para funciones matemáticas básicas.
- Librería **numpy**: Para manejo de arreglos y operaciones numéricas.
- Librería **pandas**: Para manejo y almacenamiento de datos en formato CSV.

Características y funcionalidad:

- Implementación del método congruencial lineal.
- Generación de secuencias de números pseudoaleatorios.
- Ordenamiento y manipulación de secuencias numéricas.
- Cálculo de diferencias absolutas y evaluación de la prueba KS.
- Validación de la aleatoriedad mediante la prueba KS.
- Generación y almacenamiento de resultados en archivos CSV.

- Interfaz interactiva para ingreso de parámetros y visualización de resultados.

Implementación del modelo en la computadora.

```
import math
import numpy as np
import pandas as pd

def MetodoConguencialLineal(x, a, b, mod):

    sucesion = []
    numerosAleatorios = []
    periodo = 0
    bandera = 0

    while(bandera != x):
        if (periodo == 0):
            bandera = x
        x = ((a * x) + b) % mod
        periodo = periodo + 1
        if (bandera == x): break
        numerosAleatorios.append(x)
        sucesion.append(x/mod)

    return sucesion, periodo, numerosAleatorios

def metodoKS(sucesion, n, ks):
    sucesion.sort()
    i_dividido_n = []
    valorAbs_Yin= []
    num = 0

    for i in range(1, n+1):
        i_n = i/n
        i_dividido_n.append(i_n)
        if i < len(sucesion):
            num = abs(sucesion[i] - i_n)
            valorAbs_Yin.append(num)

    max_valorAbs_Yin = max(valorAbs_Yin)

    if(max_valorAbs_Yin < ks):
```

```
    print("La prueba está aceptada! Los valores son aleatorios. ")
    print("El número máximo de la diferencia  $Y/(i-n)$  es menor al número de
Kolmogorov-Smirnov: ")
    print(round(max_valorAbs_Yin,4), " <", round(ks,4))
else:
    print("No se pasó la prueba. Los valores no son aleatorios. ")
    print("El número máximo de la diferencia  $Y/(i-n)$  es mayor al número de
Kolmogorov-Smirnov: ")
    print(round(max_valorAbs_Yin,4), " >", round(ks,4))

def verificar_alfa(n):

    alfa = float(input("Ingrese el nivel de significancia (alfa): "))
    ks = 0

    if alfa == 0.2:
        ks = 1.07/math.sqrt(n)
    elif alfa == 0.1:
        ks = 1.22/math.sqrt(n)
    elif alfa == 0.05:
        ks = 1.36/math.sqrt(n)
    elif alfa == 0.02:
        ks = 1.52/math.sqrt(n)
    elif alfa == 0.01:
        ks = 1.63/math.sqrt(n)
    elif alfa == 0.005:
        ks = 1.73/math.sqrt(n)
    elif alfa == 0.002:
        ks = 1.85/math.sqrt(n)
    elif alfa == 0.001:
        ks = 1.95/math.sqrt(n)
    else:
        print("El nivel de significancia no es válido, inténtelo nuevamente ")
        print("Niveles de significancia válidos: 0.2 - 0.1 - 0.05 - 0.02 - 0.01 -
0.005 - 0.002 - 0.001 ")
        ks = verificar_alfa(n)
    return ks

def main():

    print("Generación de números pseudoaleatorios con un generador congruencial
mixto: \n")
```

```

m = int(input("Ingrese el valor del módulo: "))
x = int(input("Ingrese el valor de la semilla: "))
a = int(input("Ingrese el valor del multiplicador: "))
b = int(input("Ingrese el valor de la constante aditiva: "))

sucesionGenerada, periodo_n, numerosAleatorios = MetodoConguencialLineal(x,a,b,m)

print("*****
*****")

print("Los numeros de X obtenidos fueron: ", numerosAleatorios)
print(" ")
print("Los numeros aleatorios obtenidos desde el valor de X y el modulo son:
", sucesionGenerada)
print(" ")
print("El periodo fue: ", periodo_n)

if(periodo_n == m):
    print("El generador es de ciclo completo")
else:
    print("El generador es de ciclo incompleto")

print("*****
*****")

print("Comprobar que la sucesión generada sea válida: \n")

ks = verificar_alfa(periodo_n)

metodoKS(sucesionGenerada, periodo_n, ks)

df= pd.DataFrame(sucesionGenerada, columns=['RandomNumbers'])
df.to_csv('random_numbers.csv', index=False)

if __name__ == "__main__":
    main()

```

Descripción del Uso y Manejo de la Aplicación:

El modelo se implementa en Python con una interfaz basada en la línea de comandos. El usuario debe ingresar los parámetros del generador (módulo, semilla, multiplicador y constante aditiva) y el nivel de significancia para la prueba KS. El programa genera la secuencia de números, calcula el período y realiza la prueba KS, calculando automáticamente el número de la tabla de KS para compararlo y muestra los resultados en la consola, guardando los números generados en un archivo CSV.

Verificación del código

La verificación del código se realiza mediante pruebas unitarias y de integración, comprobando que el generador produzca secuencias correctas y que la prueba KS se realice adecuadamente.

Utilizamos herramientas de depuración de Python para asegurarse de que el código funciona correctamente y realizamos pruebas con diferentes conjuntos de parámetros para verificar la consistencia de los resultados.

Validación

Una vez generado los números se realiza la validación de la exactitud: Comparamos las predicciones del generador con la secuencia esperada que se encuentra en el excel llamado DatosEsperadosExcel donde se evalúan los métodos para ciertos parámetros bien conocidos.

Modelo Congruencial Lial Mixto				Verificación con KS					
semilla x0	94			alfa = 0,2			n= 128		
multiplicador	81								
aditivo b	89								
módulo	128								
i	xi	ui		Número Aleatorio	Numero Aleatorio Ordenado	i	i/n	y	Mayor valor absoluto
				0,734375	0	1	0,007813	0,00781	0,0078
				0,1796875	0,0078125	2	0,015625	0,00781	
				0,25	0,015625	3	0,023438	0,00781	D(a, n) = 0,09457553
				0,9453125	0,0234375	4	0,03125	0,00781	
				0,265625	0,03125	5	0,039063	0,00781	Si cumple la condición
				0,2109375	0,0390625	6	0,046875	0,00781	0,0078 < 0,09495
				0,78125	0,046875	7	0,054688	0,00781	
				0,9765625	0,0546875	8	0,0625	0,00781	
				0,796875	0,0625	9	0,070313	0,00781	
				0,2421875	0,0703125	10	0,078125	0,00781	
				0,3125	0,078125	11	0,085938	0,00781	
				0,0078125	0,0859375	12	0,09375	0,00781	
				0,328125	0,09375	13	0,101563	0,00781	
				0,2734375	0,1015625	14	0,109375	0,00781	
				0,84375	0,109375	15	0,117188	0,00781	
				0,0390625	0,1171875	16	0,125	0,00781	
				0,859375	0,125	17	0,132813	0,00781	
				0,3046875	0,1328125	18	0,140625	0,00781	
				0,375	0,140625	19	0,148438	0,00781	
				0,0703125	0,1484375	20	0,15625	0,00781	
				0,390625	0,15625	21	0,164063	0,00781	
				0,3359375	0,1640625	22	0,171875	0,00781	
				0,90625	0,171875	23	0,179688	0,00781	
				0,1015625	0,1796875	24	0,1875	0,00781	
				0,921875	0,1875	25	0,195313	0,00781	
				0,3671875	0,1953125	26	0,203125	0,00781	

Diseño de experimentos

Se diseñan experimentos variando los parámetros del generador (módulo, semilla, multiplicador y constante aditiva) y el nivel de significancia de la prueba KS para evaluar la robustez y la calidad de las secuencias generadas.

Como nuestro objetivo es generar la mayor cantidad de números aleatorios elegimos un módulo $m=128$, una semilla $x_0=94$. una constante aditiva $b=89$ un multiplicador $a=81$; asegurándonos de generar un generador congruencial lineal de ciclo completo, ya que cumplen con las siguientes condiciones:

- ★ m y b son primos entre sí (no tengan factores comunes excepto el 1). ✓
- ★ Si q es un número primo que divide a m , entonces q divide a $(a-1)$. ✓
- ★ Si 4 divide a m , entonces 4 divide a $(a-1)$. ✓

Los resultados obtenidos fueron:

```

Generación de números pseudosaleatorios con un generador congruencial mixto:

Ingrese el valor del módulo: 128
Ingrese el valor de la semilla: 94
Ingrese el valor del multiplicador: 81
Ingrese el valor de la constante aditiva: 89
*****
Los numeros de X obtenidos fueron: [23, 32, 121, 34, 27, 100, 125, 102, 31, 40, 1, 42, 35, 108, 5, 110, 39, 48, 9, 50, 43, 116, 13, 118, 47, 56, 17, 58,
51, 124, 21, 126, 55, 64, 25, 66, 59, 4, 29, 6, 63, 72, 33, 74, 67, 12, 37, 14, 71, 80, 41, 82, 75, 20, 45, 22, 79, 88, 49, 90, 83, 28, 53, 30, 87, 96, 57
, 98, 91, 36, 61, 38, 95, 104, 65, 106, 99, 44, 69, 46, 103, 112, 73, 114, 107, 52, 77, 54, 111, 120, 81, 122, 115, 60, 85, 62, 119, 0, 89, 2, 123, 68, 93
, 70, 127, 8, 97, 10, 3, 76, 101, 78, 7, 16, 105, 18, 11, 84, 109, 86, 15, 24, 113, 26, 19, 92, 117]

Los numeros aleatorios obtenidos desde el valor de X y el modulo son: [0.1796875, 0.25, 0.9453125, 0.265625, 0.2109375, 0.78125, 0.9765625, 0.796875, 0.2
421875, 0.3125, 0.0078125, 0.328125, 0.2734375, 0.84375, 0.0390625, 0.859375, 0.3846875, 0.375, 0.0703125, 0.390625, 0.3359375, 0.90625, 0.1015625, 0.9218
75, 0.3671875, 0.4375, 0.1328125, 0.453125, 0.3984375, 0.96875, 0.1640625, 0.984375, 0.4296875, 0.5, 0.1953125, 0.515625, 0.4609375, 0.03125, 0.2265625, 0
.046875, 0.4921875, 0.5625, 0.2578125, 0.578125, 0.5234375, 0.09375, 0.2890625, 0.109375, 0.5546875, 0.625, 0.3203125, 0.640625, 0.5859375, 0.15625, 0.351
5625, 0.171875, 0.6171875, 0.6875, 0.3828125, 0.703125, 0.6484375, 0.21875, 0.4140625, 0.234375, 0.6796875, 0.75, 0.4453125, 0.765625, 0.7109375, 0.28125,
0.4765625, 0.296875, 0.7421875, 0.8125,

Ingrese el nivel de significancia (alfa): 0.2
La prueba está aceptada! Los valores son aleatorios.
El número máximo de la diferencia  $Y/(i-n)$  es menor al número de Kolmogorov-Smirnov:
0.0078 < 0.0946
  
```

Conclusiones

Luego de analizar el modelo, podemos concluir que:

Las ganancias por publicaciones tienen una tendencia al crecimiento. Se comenzó con \$49,8 millones y al finalizar el año llegaron a \$540 millones. Esto representa un incremento en la cantidad de contenido monetizable por las empresas que creció de 96,000 a 733,023 publicaciones. Lo que implica a su vez, un aumento en la conversión de usuarios a empresas, haciendo a este grupo predominante entre distintos tipos de usuarios.

Con respecto a la variación de usuarios, tiene un decrecimiento a lo largo del tiempo ya que hay un gran flujo de usuarios normales que pasan a convertirse en empresas, haciendo crecer su valor inicial a 1,52 millones. Tras finalizar los doce meses de simulación, vemos cómo los usuarios activos pasan de 500,000 a 177,184; los nuevos usuarios de 225,000 a 79,733; los usuarios convertidos de 185,000 a 65,558; y los usuarios inactivos de 85,000 a 30,122. Lo que nos demuestra que la tasa de conversión es realmente alta y la mayoría de usuarios de la red social son de tipo empresa.

El aumento de las empresas influye a su vez en el contenido monetizable, ya que al haber más empresas, se generará mayor contenido y así aumentarán a su vez las ganancias por publicaciones. Los contenidos eliminados y no monetizados, también variaron pero en menor medida. Esto demuestra que el fuerte de los 3 tipos de contenidos es el monetizable, brindando grandes oportunidades a las empresas.

Por último, los resultados financieros de la red social demostraron que en el transcurso de un año puede generar 2,9 billones, debido al incremento de todos sus ingresos, destacando los referidos a la publicidad. Los gastos totales fueron de 1,16 millones, y se pueden cubrir con los ingresos totales, lo que indica que no existió pérdida.

Esto indica que la red social se puede mantener y aumentar sus ganancias a través del crecimiento del contenido monetizable y la conversión de usuarios en empresas, por sobre los demás tipos de contenidos y usuarios. Esto sugiere, que es un modelo viable y sostenible a lo largo del tiempo.

Documentación

El código fuente se encuentra descrito anteriormente (revisar sección de [implementación del modelo en la computadora](#)). Sin embargo también puede ser visualizado desde el siguiente repositorio de GitHub, que además del código tiene el modelo en Vensim para poder ejecutarlo desde ahí más sencillamente

 [Proyecto Integrador Simulación](#)