



## Tomando decisiones basados en programación y cálculo computacional: Un caso práctico

Wilfredo Alejandro Díaz Cruz<sup>1</sup>

[wdiaz@secmca.org](mailto:w Diaz@secmca.org)

### Conocimiento especializado vs programar

En “Peleados con los algoritmos: los economistas y su talón de Aquiles” (Escudero, 2017) el autor comparte la interrogante que un economista argentino emitió en Twitter sobre cuántos paquetes de cinco figuritas habría que comprar, en promedio, para poder llenar un álbum de 200 figuritas. El autor indica que la respuesta correcta —233 paquetes—, puede alcanzarse mediante dos abordajes: (i) un cálculo matemático basado en teoría de probabilidad, y (ii) una simulación computacional. Escudero plantea que, en general, los colegas economistas no otorgaron respuestas concretas a esta interrogante, y los que plantearon respuestas fue mediante el primer abordaje.

Lo relatado por Escudero no es de sorprender puesto que en muchos países, incluidos los de Centroamérica y República Dominicana (CARD), la formación académica de economistas contiene una cantidad significativa de cursos cuantitativos, pero no así de cursos de programación o cálculo computacional. Esto resulta paradójico, ya que en la actualidad la tecnología permite administrar de forma eficiente gran cantidad de información. Así pues, esta nota tiene por objetivo mostrar cómo el uso de la programación de computadores resulta una herramienta indispensable para los nuevos economistas, ilustrando su utilidad con dos programas de R que muestran cómo determinar el número de sobres de figuritas que se requerirían para llenar el álbum Panini del Mundial de Rusia 2018.

---

<sup>1</sup>Economista Consultor de la Secretaría Ejecutiva del Consejo Monetario Centroamericano (SECMCA). Magister en Macroeconomía de la Pontificia Universidad católica de Chile. El autor agradece las recomendaciones y comentarios recibidos de Randall Romero Aguilar y Jorge Madrigal, que ayudaron a mejorar este documento. Las opiniones expresadas son las del autor y no necesariamente representan la posición de la SECMCA, ni de los miembros del CMCA.



## Acerca de la importancia de aprender a programar

¿Por qué no utilizar los recursos tecnológicos de forma más intensiva para la formación profesional en distintos campos académicos? En gran medida la respuesta a esta pregunta conlleva revisar el proceso de formación superior. Para los economistas —y para un número nada despreciable de otras profesiones— encontrar patrones, realizar cálculos, enfrentar incertidumbre, y gestionar gran cantidad de información son retos diarios que indudablemente pueden abordarse mediante la programación y el cálculo computacional, y que por esta vía pueden no solo resolverse de una manera rápida y replicable, sino que también permite entender mejor la racionalidad de un proceso o un problema, permitiendo así desarticular el mismo en partes más pequeñas. A su vez, dado que los programas de computadora son replicables, esta manera de trabajar mejora la transmisión de conocimiento y el traslado de tareas en un ambiente laboral.

Es importante también considerar que personas con diferentes perfiles académicos, no necesariamente profundos en campos como la estadística o matemática, podrían hacer frente a problemas numéricos o probabilísticos mediante el procesamiento de información con una computadora, aplicando lenguaje de programación, algoritmos y cierta lógica secuencial. El principal beneficio que se obtiene de una formación académica que incluya programación computacional es la creación de un círculo virtuoso, “aprender a codificar, codificar para aprender (Resnick 2012), por lo tanto adquirir esta habilidad tiene un efecto multiplicador, intensificado por la evolución constante de la tecnología.

Las aplicaciones de la programación y el cálculo computacional en diferentes campos laborales son diversas, que van desde analizar el movimiento de los mercados financieros, la evaluación de políticas públicas, publicidad, actividades artísticas, analizar el comportamiento del consumidor y hasta en la vida cotidiana como se revisará posteriormente.

## Caso práctico: El álbum Panini del mundial 2018

Utilizando el programa R<sup>2</sup>, es posible elaborar un programa que permita contestar las siguientes preguntas: ¿cuál es el costo esperado de llenar el álbum?, ¿cómo podría reducirse el costo del llenar el álbum?<sup>3</sup> Ambos cuestionamientos contienen intrínsecamente conceptos económicos y estadísticos aplicados que pueden afrontarse mediante programación.

---

<sup>2</sup>R es un software libre para realizar cálculos estadísticos y presentaciones gráficas. Funciona en una amplia variedad de plataformas, como UNIX, Windows y MacOS.

<sup>3</sup>Similar a lo presentado por Sardy y Velenik (2010).



Antes de iniciar, se establecerán tres supuestos relevantes para dar respuesta a las preguntas planteadas:

1. que las estampas presentan una distribución uniforme, siguiendo con lo planteado por Sardy y Velenik (2010)<sup>4</sup>.
2. el álbum está conformado por 670 estampas
3. finalmente, se asume que el álbum y las estampas se obtienen a los precios de referencia oficial en la región CARD, los cuales se reportan en el cuadro 1<sup>5</sup>.

Inicialmente, es importante tener en cuenta que el reto de coleccionar las 670 estampas para completar el álbum conlleva comprar paquetes que contienen 5 estampas cada uno. El problema radica en la aleatoriedad con que aparecen las estampas en los paquetes, siendo un proceso similar a un muestreo con reposición, es decir, es posible obtener estampas repetidas. En síntesis, a medida que se va llenando el álbum, el costo de conseguir las estampas faltantes será cada vez mayor, es decir se requerirá una gran cantidad de paquetes.

En este primer programa, realizamos un experimento de Monte Carlo para determinar cuántos paquetes serían necesarios en el caso de que una persona desee llenar el álbum sin intercambiar estampas repetidas con sus amigos:

```
1 #-----Simulación de llenado de album por una persona (sin intercambio)
2 total.participantes <- 10000
3 total.estampas <- 670
4 estampas.por.paquete <- 5
5 las.estampas <- 1:total.estampas
6 participantes <- rep(0, total.participantes)
7 for (j in 1:total.participantes) {
8   album <- rep(FALSE, total.estampas)
9   paquetes <- 0
10  while(!all(album)){
11    paquetes <- paquetes + 1
12    estampas <- sample(las.estampas, estampas.por.paquete, replace=TRUE)
13    album[estampas] <- TRUE
14  }
15  participantes[j] <- paquetes
16 }
17 promedio.paquetes <- mean(participantes)
```

Este programa realiza un experimento de Monte Carlo de la siguiente manera: en las líneas 2 a la 4 se declaran valores para el total de iteraciones (esto equivale a suponer que 10,000 personas hacen el mismo ejercicio), el total de estampas del álbum y las estampas por paquete. En la línea 5 se declara `las.estampas` como un rango para identificar las

<sup>4</sup>En este artículo los autores realizaron un experimento para comprobar que la distribución de las estampas es uniforme. Éste consiste en comprar 12 cajas de 500 estampas y realizar un registro de la frecuencia con la que aparecen cada una de las mismas.

<sup>5</sup>Esta información se obtuvo a través de las publicaciones de prensa sobre la comercialización a nivel regional.



estampas con número enteros entre 1 y 670. La línea 6 genera el vector `participantes` con 10,000 ceros, en el cual guardaremos el resultado de cada una de las iteraciones del experimento. Entre las líneas 7 y 16 se utiliza un bucle (con el comando `for`) que ejecuta 10,000 iteraciones de llenar un álbum: se empieza con un álbum vacío, indicado por los 670 elementos con valor `FALSE`, los cuales serán sustituidos por `TRUE` conforme vayan apareciendo las distintas estampas. Se declara paquetes ya comprados como cero (línea 9), como un contador de paquetes que irá aumentando (línea 11) conforme se compran paquetes nuevos. Las líneas 10 a 14 definen un bucle `while` que se ejecutará mientras no esté lleno el álbum (es decir, mientras no todos los elementos de `álbum` sean `TRUE`); cada iteración representa el acto de comprar un paquete nuevo, incrementando el total comprado (línea 11), obteniendo 5 estampas aleatoriamente (línea 12) y pegándolas en el álbum (línea 13). Finalmente, la línea 15 almacena el número de estampas que se compraron en esta iteración de Monte Carlo, y la línea 17 calcula el promedio de paquetes de las 10,000 iteraciones.

Al ejecutar el programa anterior determinamos que una persona que desee llenar el álbum sin intercambiar estampas repetidas deberá comprar, en promedio, 949 paquetes, es decir 4,745 estampas.

A pesar de ser viable, la estrategia de llenar el álbum en autarquía no es óptima, toda vez que la persona termina con más de 4,000 estampas repetidas. Así, una persona interesada en reducir el costo de llenar el álbum podría intercambiar estampas repetidas con sus amigos, aunque esto implique un mayor esfuerzo de negociación por parte de la persona interesada en llenar la colección debido a que tendrá que constituir una red de intercambio que, entre más numerosa, presumiblemente más le reducirá el costo de completar el álbum.

Para simular la experiencia de llenar el álbum cuando se puede intercambiar estampas, este documento propone una estrategia específica de dos fases:

- La primera consiste en que tres amigos compran colectivamente 4 cajas de 100 paquetes de estampas, se reparten 1 caja cada uno y 33 paquetes de estampas de la caja sobrante, con lo cual cada uno de ellos tendrá 665 estampas en total. Esto implica que al menos 5 estampas tendrán que ser adquiridas posteriormente mediante compra individual.
- En la segunda fase cada participante intercambia estampas con 10 personas<sup>6</sup> (incluyendo posiblemente a los dos amigos de la fase inicial).

<sup>6</sup>En principio, todas estas personas tendrían una cantidad similar de estampas repetidas y faltantes



# Notas Económicas Regionales

Secretaría Ejecutiva del Consejo Monetario Centroamericano

No. 95, mayo 2018

```
1 #=====Simulación de llenado de album con intercambio
2 library("plyr") #carga el paquete PLYR, para dividir, calcular y combinar datos
3 total.participantes <- 1000
4 total.estampas <- 670
5 estampas.por.paquete <- 5
6 las.estampas <- 1:total.estampas
7 compradas <- rep(0, total.participantes)
8 intercambiadas <- rep(0, total.participantes)
9 sobrantes <- rep(0, total.participantes)
10 #=====estrategia propuesta de intercambio
11 total.amigos <- 10
12 paquetes.iniciales <- 133
13 for (j in 1:total.participantes){
14   mi.album <- rep(FALSE, total.estampas)
15   mis.estampas = unique(sample(las.estampas, paquetes.iniciales * estampas.por.paquete,
16     replace=TRUE))
17   mi.album[ mis.estampas ] <- TRUE
18   aun.faltan <- sum(mi.album==FALSE)
19   mis.repetidas <- paquetes.iniciales * estampas.por.paquete - sum(mi.album==TRUE)
20   antes.de.intercambiar <- aun.faltan
21   for (k in 1:total.amigos){
22     if (mis.repetidas > 0){
23       # estampas repetidas e intercambios de los 10 amigos
24       sus.estampas <- count(sample(las.estampas, paquetes.iniciales * estampas.por.
25         paquete, replace=TRUE))
26       sus.estampas[ 'repetidas' ] <- sus.estampas[ 'freq' ]>1
27       sus.repetidas <- sus.estampas[ 'x' ][ sus.estampas[ 'repetidas' ]==TRUE]
28       mis.faltantes <- which(mi.album==FALSE)
29       para.intercambiar <- intersect(mis.faltantes, sus.repetidas)
30       # simular el intercambio y pegar las estampas
31       if (mis.repetidas >= length(para.intercambiar)){
32         mi.album[para.intercambiar] <- TRUE
33         mis.repetidas <- mis.repetidas - length(para.intercambiar)
34       } else {
35         mi.album[sample(para.intercambiar, mis.repetidas, replace = FALSE)] <- TRUE
36         mis.repetidas <- 0
37       }
38     }
39     aun.faltan <- sum(mi.album==FALSE)
40   }
41   compradas[j] <- aun.faltan #estampas a comprar por aparte.
42   intercambiadas[j] <- antes.de.intercambiar - aun.faltan # número de estampas
43     intercambiadas
44   sobrantes[j] <- mis.repetidas #registra el número de estampas que sobraron del
45     intercambio.
46 }
47 resultados <- data.frame(compradas=compradas, intercambiadas=intercambiadas, sobrantes=
48   sobrantes)
49 #===== Estimación de costos por país
50 paises <- c( 'CR', 'ES', 'GT', 'HN', 'NI', 'RD' )
51 precio.album <- c(1250.0, 2.0, 15.0, 40.0, 60.0, 180.0) #precios del album por país
52 precio.paquete <- c( 450.0, 0.8, 5.5, 18.0, 20.0, 35.0) #precios de los paquetes por
53   país
54 tipo.cambio <- c( 564.3, 1.0, 7.4, 23.6, 31.0, 49.1) #tipo de cambio por país
55 monedas <- c('Colones', 'US_Dólar', 'Quetzal', 'Lempiras', 'Córdobas', 'Peso_Dominicano')
56 names(precio.paquete) <- paises
57 names(tipo.cambio) <- paises
58 precio.estampa <- 0.5*precio.paquete
59 costo.total <- precio.album + paquetes.iniciales*precio.paquete + mean(compradas) *
60   precio.estampa
61 costo.total.dolares <- costo.total / tipo.cambio #estima el costo total en dólares.
```



Para ejecutar exitosamente este programa es necesario instalar previamente el paquete `plyr` (línea 2<sup>7</sup>). En el programa anterior, la primera fase (comprar 133 sobres) está implementado en las líneas 14 a 19; en la 18 se cuenta el número de estampas que están repetidas. Al ejecutar el programa, encontramos que luego de completar esta primera fase a cada uno de los participantes le faltará en promedio 251 estampas y tendrá 246 estampas repetidas para intercambiar.

La segunda fase está implementada en las líneas 20 a 38. Con cada uno de los amigos (línea 20) empezamos el proceso de intercambio, el cual es posible solamente si aún hay estampas repetidas para intercambiar (línea 21). Se determinan cuáles estampas tiene repetidas el amigo (líneas 23 a 25) y cuáles le faltan al participante (línea 26), a partir de ello, se puede determinar cuáles estampas pueden intercambiarse (aquellas que el amigo tenga repetidas y que aún le faltan al participante, línea 27). Las estampas intercambiadas se pegan en el álbum (línea 30), siempre y cuando el participante tenga estampas repetidas suficientes para intercambiar (línea 29), de lo contrario (línea 32) se intercambian al azar la cantidad disponible (línea 33). Ejecutando esta parte del programa, encontramos que luego de intercambiar estampas con 10 amigos, a cada participante le faltaría en promedio 12 estampas (incluyendo las 5 que necesariamente le faltarían en la fase inicial en el caso improbable de no haber obtenido ninguna estampa repetida). Estas últimas estampas se compran individualmente a un precio unitario mayor, el cual se asume igual a la mitad del precio de un paquete.

Este programa lleva además un registro de las estampas faltantes, las repetidas, los intercambios, las estampas compradas y las sobrantes, los cuales se utilizan posteriormente para calcular el costo esperado de llenar el álbum en cada uno de los países (líneas 45 a 54); los resultados se presentan en el cuadro 1.

Cuadro 1: Precios y costos totales de llenar el álbum Panini del Mundial 2018 en la región CARD

País	Moneda	Tipo cambio	Precio Album		Precio Sobre		Costo Total	
			MN	USD	MN	USD	MN	USD
Costa Rica	Colones	564,30	1250	2,2	450,0	0,8	63773,2	113,0
El Salvador	US Dólar	1,00	2	2,0	0,8	0,8	113,2	113,2
Guatemala	Quetzal	7,40	15	2,0	5,5	0,7	779,2	105,3
Honduras	Lempiras	23,60	40	1,7	18,0	0,8	2.540,9	107,7
Nicaragua	Cordobas	31,00	60	1,9	20,0	0,6	2.838,8	91,6
República Dominicana	Peso Dominicano	49,10	180	3,7	35,0	0,7	5.042,9	102,7

Nota: Tipo de cambio corresponde a promedio de marzo de 2018.

<sup>7</sup>El paquete puede instalarse con este comando: `install.packages("plyr")`



Con el anterior programa es posible analizar escenarios alternativos de esta estrategia, cambiando ciertos parámetros (como el número de estampas, los precios, el número de amigos, entre otros). Así, el lector interesado en experimentar su idea podría conocer los resultados de su escenario preferido cambiando los supuestos que considere necesario.

El objetivo que persigue este ejercicio es remarcar la importancia de adquirir la habilidad de programar. Se observa con este ejemplo que mediante la ejecución del programa presentado en este documento se pueden hacer diferentes escenarios, de forma eficiente, eficaz y confiable.

## Consideraciones finales

A pesar de que la aplicación anterior puede parecer caricaturesca desde una perspectiva de un documento económico-técnico, la misma ilustra adecuadamente el cómo las herramientas computacionales pueden servir para la toma de decisiones en la vida cotidiana, y reafirma la indudable utilidad de estas herramientas en otros contextos y actividades más especializadas.

En los últimos años se ha observado una mayor demanda por recurso humano que posea la capacidad de programar, siendo una tendencia marcada en varias industrias, en particular con mayor incidencia en las economías avanzadas. Por ejemplo, se proyecta que entre 2016 y 2026 en EE.UU. se crearán 253,400 puestos de trabajo en el grupo ocupacional “Desarrolladores de software(aplicaciones)”, con sueldos anuales promedio de USD 100,080 , siendo uno de los trabajos mejor remunerados en ese país.

Las habilidades para programar no sólo son útiles para desarrolladores y programadores de software, también puestos de trabajo relacionados a ingeniería, diseño, análisis de datos y muchos campos más demandan esta habilidad en la actualidad. Por ejemplo, en el 2015 se abrieron 6.8 millones de puestos laborales en EEUU que requerían este tipo de destreza. Se proyecta que los trabajos que requieran capacidad para programar crecerán 7.2% en los próximos 10 años.

En el campo de la economía la tendencia no es diferente. Sin habilidades computacionales y de programación no sería posible simular los actuales modelos económicos, ni realizar ejercicios prospectivos. Esto se explica en parte por el desarrollo de los nuevos métodos de modelización macroeconómicos y macrofinancieros como los presentados por Wieland y col. (2016), estos autores exponen sobre la facilidad de uso de plataformas (como MATLAB) para el análisis comparativo de políticas económicas. También se ha extendido en el análisis financiero a través del uso de redes neuronales (neural network), como se muestra en Moreno, León y Cely (2016), convirtiéndose en una potente herra-



mienta para la predicción y análisis del sistema financiero, permitiendo el desarrollo de políticas macroprudenciales y el diseño de sistemas de alerta temprana. Así, según el U.S. Bureau of Labor Statistics (BLS), se estima que entre el 2016-2026 el campo laboral de los economistas aumentará sus plazas a un ritmo promedio de 6%, siendo mejores las perspectivas de empleo para aquellos que posean un título de maestría o doctorado, con fuertes habilidades analíticas y experiencia en el uso de software de análisis estadístico.

Finalmente, un tema de frontera es la posibilidad de utilizar machine learning para realizar política monetaria, como lo menciona Hinge (2017), “Machine learning brinda a los bancos centrales un conjunto de nuevas herramientas que pueden usar a un costo relativamente bajo. Por ejemplo, Rendell de Kort, un economista del Banco Central de Aruba, presentó un modelo de aprendizaje automático para pronosticar la demanda del turismo en una conferencia organizada por el Comité Irving Fisher.”

Ante este panorama, es necesario reflexionar sobre la necesidad de cambiar los enfoques de los programas de estudio para los economistas centroamericanos, en línea con el profesional que se requiere actualmente, más enfocado en el uso aplicado de la tecnología, evitando así el divorcio entre los métodos teórico-analíticos y la programación de computadoras.

## Referencias

- Hinge, Daniel (27 de oct. de 2017). *Teaching machines to do monetary policy*. Central Banking.
- Moreno, José Fernando, Carlos León y Jorge Cely (2016). *Whose Balance Sheet is this? Neural Networks for Banks' Pattern Recognition*. Inf. téc. 959. Banco de la República – Colombia.
- Resnick, Mitch (2012). *Let's teach kids to code*. TED Talk. URL: [https://www.ted.com/talks/mitch\\_resnick\\_let\\_s\\_teach\\_kids\\_to\\_code](https://www.ted.com/talks/mitch_resnick_let_s_teach_kids_to_code).
- Sardy, Sylvain e Yvan Velenik (2010). *Paninimania: sticker rarity and cost-effective strategy*. Swiss Statistical Society, Bulletin 66.
- Wieland, Volker y col. (23 de ago. de 2016). *New Methods for Macro-Financial Model Comparison and Policy Analysis*. Inf. téc. 107. Institute for Monetary y Financial Stability (IMFS).