Primos cercanos a los “mosaicos”: un estudio de A000026 con desplazamientos 0, +1, −1 y “sandwich”

Adolfo Catral

30 de agosto de 2025

Resumen

Sea $a(n) = \text{A000026}(n)$ la *función mosaico*: si $n = \prod p^e$ es la factorización prima (con $p^e \parallel n$), entonces

$$a(n) = \prod (p \cdot e).$$

Estudiamos tres familias de índices

$$S_+ = \{n : \ a(n)+1 \text{ es primo}\}, \quad S_- = \{n : \ a(n)-1 \text{ es primo}\}, \quad S_\pm = \{n : \ a(n)\pm 1 \text{ son ambos primos}\},$$

así como la familia de distancia 0 (valores primos de $a(n)$). Damos propiedades elementales (paridad y congruencias), heurísticas de densidad, un método computacional reproducible y resultados empíricos hasta $n = 10^4$: $|S_+| \approx 1721$, $|S_-| \approx 1734$, $|S_\pm| \approx 425$. Además mostramos cómo construir *primos gigantes* de la forma $p = a(n) + 1$ con $p - 1$ *totalmente factorizado*, lo que permite certificados tipo $n-1$ (Pocklington–Lehmer/BLS) o ECPP muy rápidos.

1. Introducción

Las transformaciones *casi primo* del tipo

$$\{\, n : \ \text{isprime}(f(n) \pm 1) \,\}$$

son canónicas en aritmética experimental. Para $f(n) = a(n) = \prod(p \cdot e)$, el valor $a(n)$ entrelaza *bases primas* y *exponentes* de un modo distinto a funciones clásicas como σ, τ o φ. Filtrar por “$+1$ primo”, “-1 primo” y “ambos” detecta índices con estructura multiplicativa marcada; además, cuando $a(n)+1$ es primo, $p-1 = a(n)$ queda factorizado por construcción, lo que facilita obtener *primos grandes* con prueba eficiente.

2. Definiciones y propiedades básicas

Para $n = \prod p^e$ con $p^e \parallel n$,

$$a(n) = \prod (p \cdot e), \qquad a(1) = 1, \qquad a(p^e) = p\, e. \tag{1}$$

La función es multiplicativa.

Paridad. Se tiene

$$a(n) \text{ impar} \iff 2 \nmid n \ \text{ y } \ e \text{ es impar para todo primo } p \mid n.$$

En particular, si n es impar y libre de cuadrados ($e = 1$), entonces $a(n)$ es impar y $a(n) + 1$ es par > 2, por tanto no primo (salvo $n = 1$).

Congruencias mod 6. Los primos > 3 son $\equiv 1$ o $5 \pmod 6$. De aquí:

$$a(n)+1 \text{ primo} > 3 \Rightarrow a(n) \equiv 0,4 \pmod 6, \quad a(n)-1 \text{ primo} > 3 \Rightarrow a(n) \equiv 0,2 \pmod 6.$$

El caso "sandwich" exige $a(n)$ par (si $a(n)$ fuera impar, uno de los vecinos sería par > 2).

3. Las tres familias de índices

Definimos

$S_+ = \{n : a(n)+1 \text{ es primo}\}$, $S_- = \{n : a(n)-1 \text{ es primo}\}$, $S_\pm = \{n : a(n)\pm 1 \text{ son ambos primos}\}$.

- $1 \in S_+$ porque $a(1) = 1$ y $1 + 1 = 2$.
- Si n es impar libre de cuadrados y $n > 1$, entonces $n \notin S_+$ (paridad).
- Para n par o impares con algún exponente par, $a(n)$ suele ser par; entonces $a(n) \pm 1$ son impares y mejores candidatos a primos.

4. Heurística de densidad

Para un entero "aleatorio" M de orden X, $\mathbb{P}(M \pm 1 \text{ primo}) \approx 1/\log X$. Sustituyendo $M = a(n)$, interesa promediar $1/\log a(n)$ sobre $n \le N$. Dado que $a(n)$ tiende a ser más pequeño que n y muy y-suave, la tasa de aciertos para $n \le N$ puede ser apreciable en rangos moderados. Los datos de la Sección 6 concuerdan con esta heurística.

5. Métodos computacionales

A partir de (1), los chequeos son directos. A continuación, implementaciones compactas (ASCII) en Python, PARI/GP y *Mathematica*.

Python (SymPy)

```
from sympy import factorint, isprime

def mosaic(n: int) -> int:
    if n == 0: return 0
    prod = 1
    for p,e in factorint(n).items():
        prod *= p*e
    return prod

def Splus(N): return [n for n in range(1, N+1) if isprime(mosaic(n)+1)]
def Sminus(N): return [n for n in range(1, N+1) if mosaic(n)>1 and isprime(mosaic(n)
    -1)]
def Ssand(N): return [n for n in range(1, N+1)
                      if isprime(mosaic(n)-1) and isprime(mosaic(n)+1)]
```

PARI/GP

```
mosaic(n)={
  if(n==0, return(0));
  my(f=factor(n), k=matsize(f)[1], prod=1);
  for(i=1, k, prod *= f[i,1]*f[i,2]);
  prod
};
Splus(N)=select(n->isprime(mosaic(n)+1), vector(N,i,i));
Sminus(N)=select(n->mosaic(n)>1 && isprime(mosaic(n)-1), vector(N,i,i));
Ssand(N)=select(n->isprime(mosaic(n)-1) && isprime(mosaic(n)+1), vector(N,i,i));
```

Mathematica

```
mosaic[0]=0;
mosaic[n_Integer?Positive] =
  Times@@(FactorInteger[n]/.{p_,e_}:>p*e);

Splus[nmax_] := Select[Range[nmax], PrimeQ[mosaic[#]+1]&];
Sminus[nmax_] := Select[Range[nmax], mosaic[#]>1 && PrimeQ[mosaic[#]-1]&];
Ssand[nmax_] := Select[Range[nmax], PrimeQ[mosaic[#]-1] && PrimeQ[mosaic[#]+1]&];
```

6. Resultados empíricos

Con los programas anteriores (Python/SymPy), para $1 \le n \le 10^4$ se obtuvo:

$$|S_+| \approx 1721, \qquad |S_-| \approx 1734, \qquad |S_\pm| \approx 425.$$

Esto equivale a $\sim$17,2 %, $\sim$17,3 % y $\sim$4,3 %, respectivamente. Muestras de términos iniciales (índices n):

$$\begin{aligned} S_+ &: \ 1, 2, 4, 6, 8, 9, 10, 12, 18, 22, 24, 25, 28, 30, 32, 40, 42, 45, 46, 52, \ldots \\ S_- &: \ 3, 4, 6, 8, 9, 12, 14, 16, 18, 20, 24, 30, 36, 38, 40, 42, \ldots \\ S_\pm &: \ 4, 6, 8, 9, 12, 18, 24, 30, 40, 42, 45, 54, 56, 60, \ldots \end{aligned}$$

La escasez relativa de $S_\pm$ es compatible con la rareza de los primos gemelos.

7. Primos gigantes vía $a(n) + 1$

Una ventaja clave de S_+ es computacional: si $p = a(n) + 1$ es primo, entonces $p - 1 = a(n)$ está *completamente factorizado* por construcción (sus factores primos son los que aparecen en n y los factores de los exponentes). Por ejemplo, tomando

$$n = (\text{primorial}(283))^{116} \quad \Rightarrow \quad a(n) = \text{primorial}(283) \cdot 116^{61},$$

se obtiene $p = a(n) + 1$, un primo de *244 dígitos*. La factorización de $p - 1$ es explícita:

$$p - 1 = \Big(\prod_{q \le 283} q \Big) \cdot (2^2 \cdot 29)^{61}.$$

Esto permite un certificado $n-1$ o ECPP muy rápido. En general, con $n = (p\#)^a$ se tiene

$$a(n) = p\# \cdot a^{\pi(p)},$$

lo que da control fino del tamaño de $a(n)$ (y, por ende, de $p = a(n) + 1$).

8. Observaciones y conjeturas

(1) Paridad necesaria. Para $n > 1$ en S_+, típicamente $a(n)$ es par (única excepción $n = 1$).

(2) Estructura de exponentes. Cuantos más exponentes pares o más primos pequeños tenga n, más y-suave es $a(n)$ y más barata la certificación cuando $a(n) \pm 1$ es primo.

(3) Conjeturas. Se espera que S_+ y S_- sean infinitos. Para $S_\pm$ (gemelos centrados en $a(n)$) también, aunque es más delicado.

(4) Distribuciones modulares. Estudiar $a(n)$ mód m (p.ej., $m = 3, 4, 6$) dentro de cada familia puede afinar heurísticas y predicciones.

9. Reproducibilidad

Los listados de la Sección 5 permiten regenerar S_+, S_- y $S_\pm$ a voluntad. Para *primos gigantes* $p = a(n) + 1$, se recomienda publicar: (i) el decimal de p; (ii) la lista "primo^exponente" de $p - 1$; (iii) el registro de la prueba ($n-1$/ECPP).

Agradecimientos

Agradezco a la comunidad matemática por herramientas abiertas (PARI/GP, SymPy, *Mathematica*) que facilitan la aritmética experimental.

Referencias

[1] A. A. Mullin, Some related number-theoretic functions, Research Problem 4, *Bull. Amer. Math. Soc.* **69** (1963), 446–447.

[2] B. Gordon y M. M. Robertson, Two theorems on mosaics, *Canad. J. Math.* **17** (1965), 1010–1014.

[3] R. A. Gillman, The Average Size of a Certain Arithmetic Function, *Amer. Math. Monthly* **100** (1993), 296–298.

[4] D. Tsai, A recurring pattern in natural numbers of a certain property, *Integers* **21** (2021), Article A32.

[5] N. J. A. Sloane, *A Handbook of Integer Sequences*, Academic Press, 1973.

[6] N. J. A. Sloane y S. Plouffe, *The Encyclopedia of Integer Sequences*, Academic Press, 1995.

[7] H. C. Pocklington, The determination of the prime or composite nature of large numbers by Fermat's theorem, *Proc. Cambridge Philos. Soc.* **18** (1914), 29–30.

[8] D. H. Lehmer, Tests for primality by the converse of Fermat's theorem, *Bull. Amer. Math. Soc.* **33** (1927), 327–340.

[9] A. O. L. Atkin y F. Morain, Elliptic curves and primality proving, *Math. Comp.* **61** (1993), 29–68.