□ Bloom Filter Calculator □

Bloom filters are space-efficient probablistic data structures used to test whether an element is a member of a set.

They're surprisingly simple: take an array of **m** bits, and for up to **n** different elements, either test or set **k** bits using positions chosen using hash functions. If all bits are set, the element *probably* already exists, with a false positive rate of **p**; if any of the bits are not set, the element *certainly* does not exist.

Bloom filters find a wide range of uses, including tracking which articles you've read, speeding up Bitcoin clients, detecting malicious web sites, and improving the performance of caches.

This page will help you choose an optimal size for your filter, or explore how the different parameters interact.

---

n

Number of items in the filter (optionally with SI units: k, M, G, T, P, E, Z, Y)

101988

p

Probability of false positives, fraction between 0 and 1 or a number indicating 1-in-p

0.0001

m

Nu**m**ber of bits in the filter (or a size with KB, KiB, MB, Mb, GiB, etc)

k

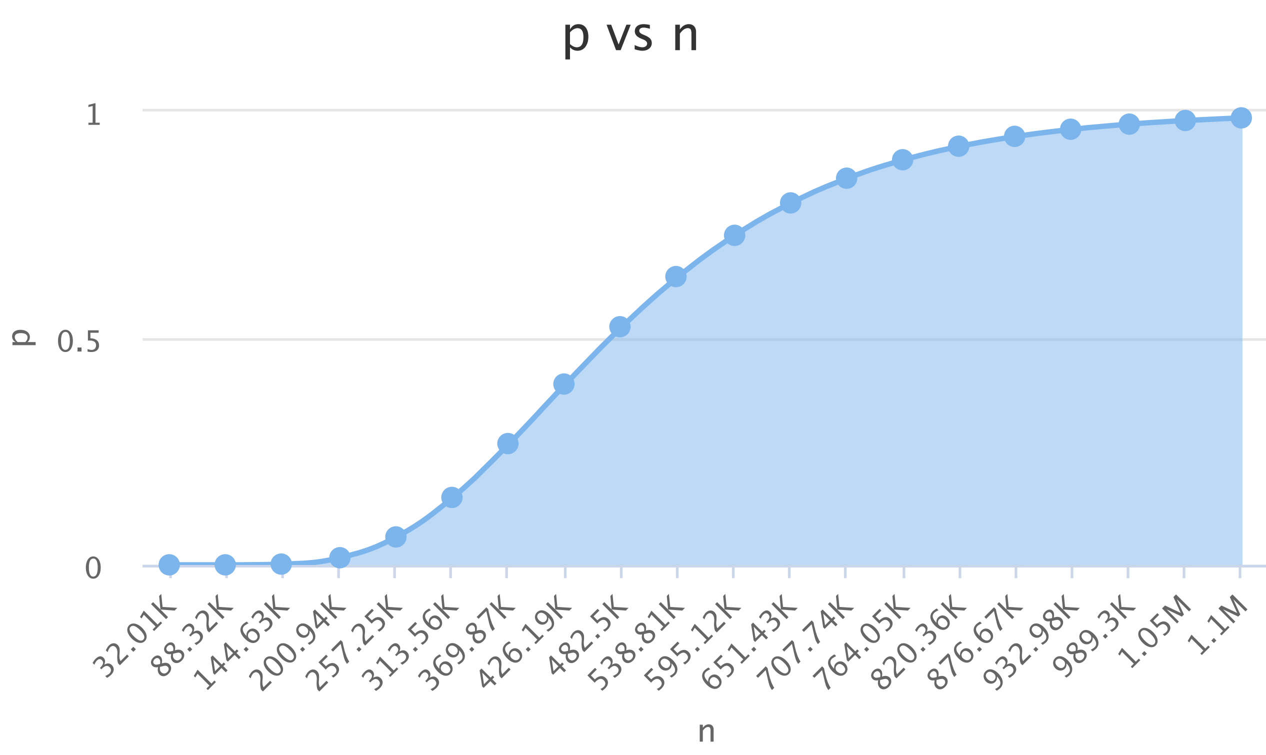
Number of hash fun**c**tions

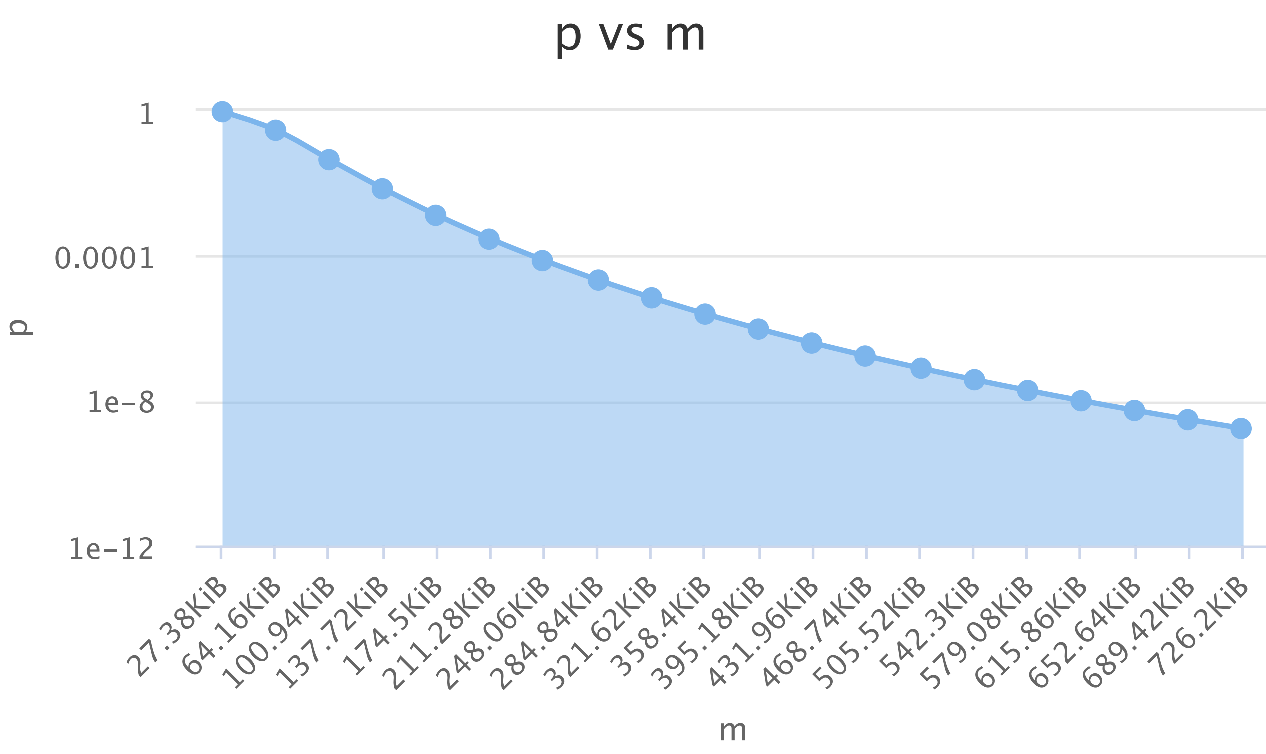
12

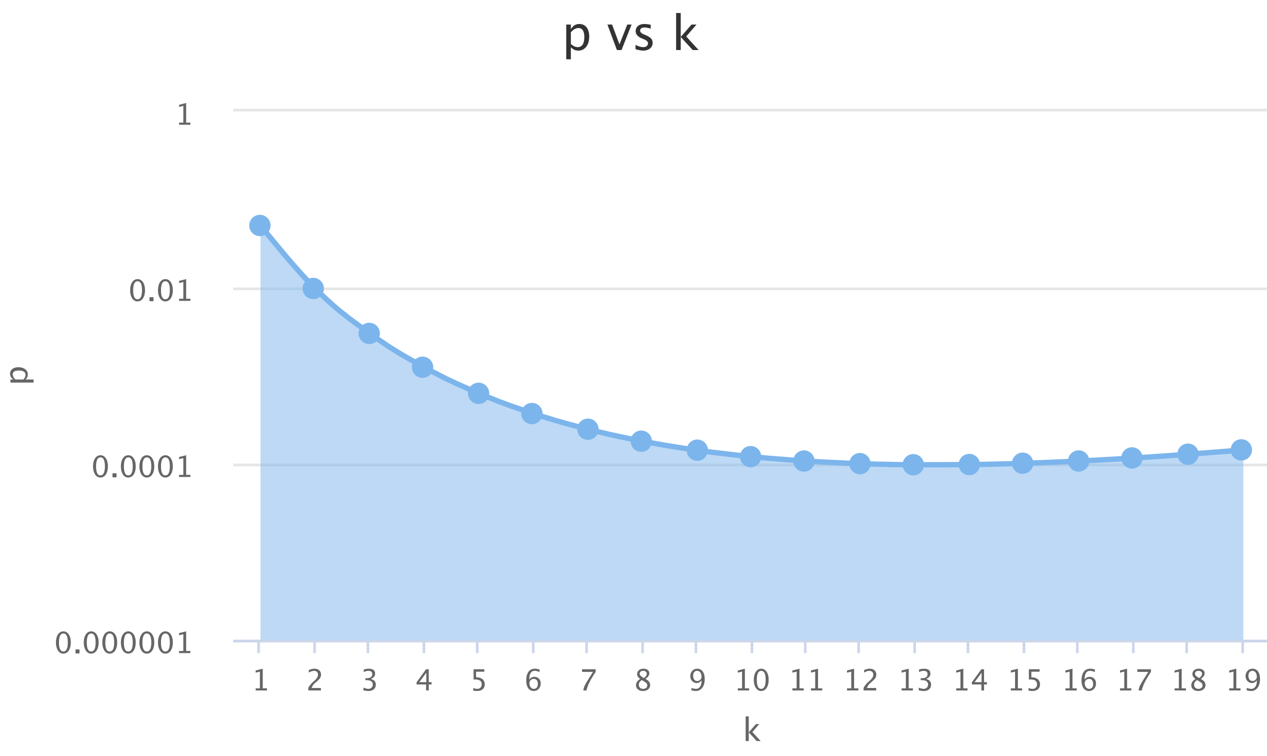
Submit

n = 101,988
p = 0.0001 (1 in 10,000)
m = 1,961,567 (239.45KiB)
k = 12

```
n = ceil(m / (-k / log(1 - exp(log(p) / k))))
p = pow(1 - exp(-k / (m / n)), k)
m = ceil((n * log(p)) / log(1 / pow(2, log(2))));
k = round((m / n) * log(2));
```







Last updated: 2022-02-25T02:40:46+00:00

Provided without warranty. For entertainment purposes only. Avoid contact with eyes.

Thomas Hurst <tom@hur.st>