Homework 6 Write-Up

David Enders

October 12, 2022

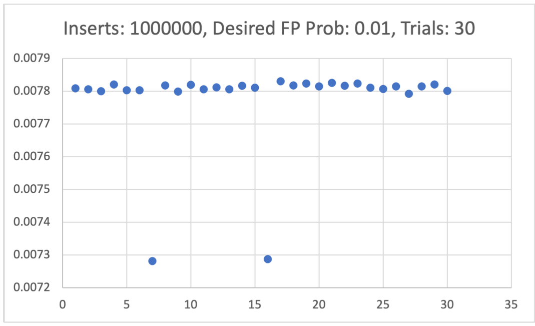
Introduction

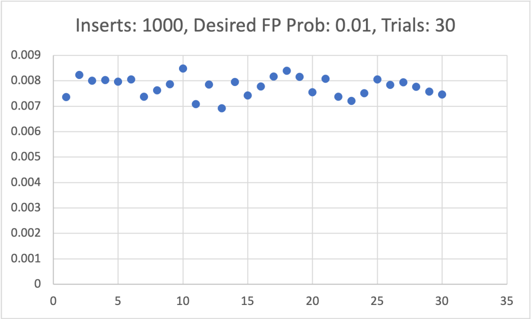
In this experiment, we are answering the question: how does a bloom filter's actual false-positive probability compare to the theoretical probability for different amounts of input values and different desired false-positive probability. To rephrase: how does the actual false-positive probability compare to the theoretical one with small vs. large number of inputs?

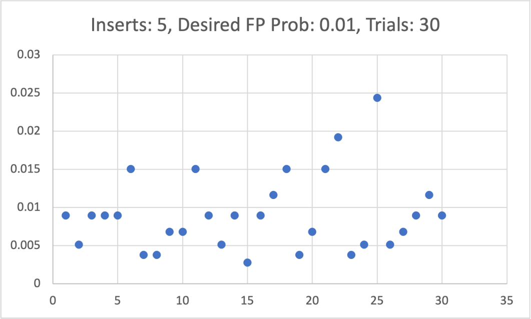
Method

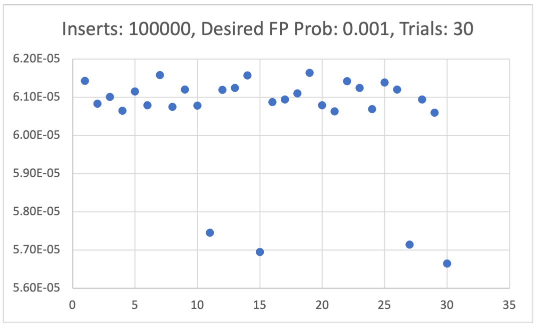
To measure/test this I measured the probability that a value being checked for would report a false positive after 1000000 inserts in a bloom filter with desired false-positive probability — or Prob(FP) — of 0.01, after 1000 inserts in a bloom filter with desired prob(FP) of 0.01, after 5 inserts in a bloom filter with desired prob(FP) of 0.01, and after 100000 inserts in a bloom filter with desired prob(FP) of 0.001. I ran 30 trials of each variation, where I inserted the above number of strings into the bloom filter and then measured the probability that k randomly selected bits would be set to 1, which is equal to Prob(FP). The results are below. The x-axis denotes the trial, and the y-axis marks Prob(FP).

Results









Interpretation

As we can see, for all of the variations but the variation with n=5, our maximum value for Prob(FP) is comfortably below the desired false-positive probability. This is likely due to rounding in the formula that calculates the necessary number of hash functions and bits to store; we must be overestimating the required numbers somewhere in the formula. For n=5, the number of bits to store is so small that variations in which random bucket was chosen significantly impact Prob(FP) by the end of the 5 inserts. This is because, while the formulas used to calculate the number of hash functions and number of bits to store still hold, the rounding that occurs have a much greater impact when the final values are relatively small. However, with any number of inputs large enough to make bloom filters a reasonable option for storing data, the data from these experiments suggests that this implementation will result in false-positive probability at least as good as the desired Prob(FP) inputted.