Bloom Filter Project Report

Name : Weifeng Lyu

Task 1 Write-up :

Did one of the functions work better than the other? Did data choice matter? Why or why not? Justify your answer with relevant plots (on page 2 of the report template) and discussion. More important is how do these hash functions compare when utilized in your implementation of a Bloom filter.

The first hash function is $hi(x)=r(si+x)$ and the second hash is $hi(x)=((ai*x + bi)\ mod\ P)\ mod\ n$
From the plots we can see they work equally well. If we look closer, it might be type2 is slightly better. The first hash function is built in with one random number generator's randint and the second hash function is built with two random number generator's randint, mod by two prime number. Every run is independent with previous run, and the second hash function's process is more stochastic.

The data choice of only even value or all value does not matter and the scatterplots are distributed uniformly as we can see from the figure, so the data choice does not have impact in this random process because all the even values and nature values will be produced with a different sequence by the same stochastic process.

Task 2 Write-up :
Discuss the details of your Bloom filter implementation on report page 1, in the space provided, along with any design choices you made.

The Bloom Filter implementation:
1. Initiate configData, enter configData into the bloom filter.
2. The implement will separate the data into two types of hashes from task1, then generate a hash table consists of vector of n boolean values, initially all set to 0 as well as k independent hash functions, h0, h1, …, hk-1, each with range 0 to n-1/
3. After all the integers are loaded from the file, it will enter each element into the bloom filter, the entry indexes depend on the certain number of hash functions (here we have k hash functions) and calculate the hashes for a given input, then those indexes' value with those hashes will be set to 1.
4. As the integer entries, the index's value may be set to 1 several times.
5. When the function checks whether the integer exists in the table, it goes through the same process. If the integer was entered before, it would be processed by same hash functions and search in those indexes by old hashes, check whether all values are one, if one of them are 0, the bloom filter will return false. Otherwise, return true.

Task 3 Write-up :
Make sure to include the 4 relevant plots in your report on page 3 in the spaces provided. Discuss the results of your experiments on page 1. Briefly describe how you derived your theoretical values. How do your results compare to the theoretical false positive rate which is a function of k and c? Do your experiments agree with the theoretical result for the optimal choice of k? Justify your discussion with the relevant plot/graphs. Any other conclusions, insights, or observations?

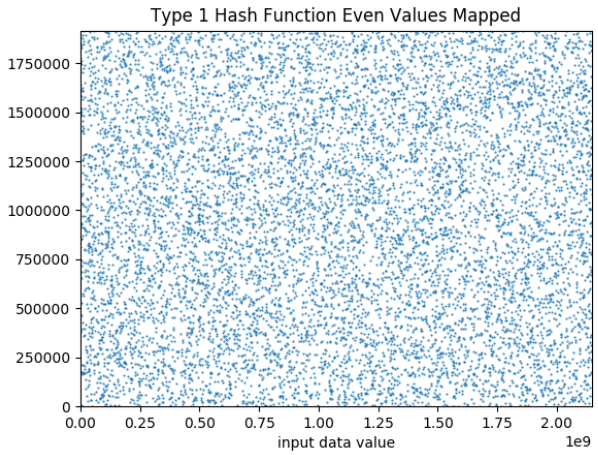
To derive the theoretical values, we fix c first to be 10, and then 15, and sweep k from .4 * c to 1.0 * c, then we append the c=10 array and the c=15 array with $f(k) = \left(1 - e^{-\left(\frac{k}{c}\right)}\right)^k$ theoretical value of false positive rate, where k is increasing number from 4 to c, then all the theoretical values are derived. The real false positive rate values are generated by the number of appearances of those values marked missing, also taking the average after 10 times of trials. Comparing to those real values with the theoretical values, the results with c=10 and c=15 are well aligned. And the margin between real and theoretical value with c=15 is better aligned than c=10.

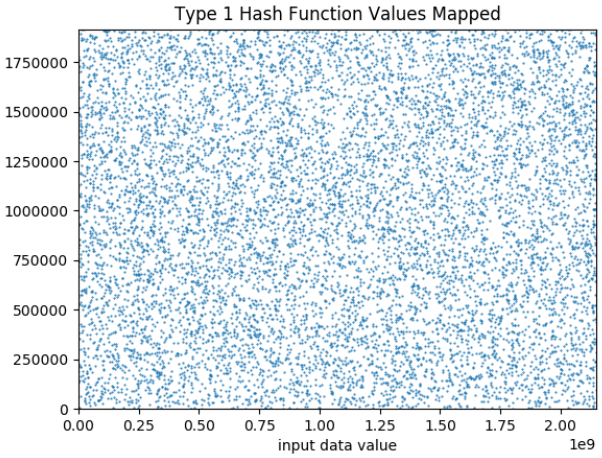
The experiments agree with the theoretical result for the optimal choice of k because k is optimal at 7 with 0.8% false positive rate, and k is optimal around 10 with 0.05% false positive rate. Thus, we can conclude that the theoretical false positive value function correctly reflects the real number of missing marked items with actual presence, one of the insights is that increasing c value (The size of the array over the number of value entry) can reduce the false positive rate significantly.

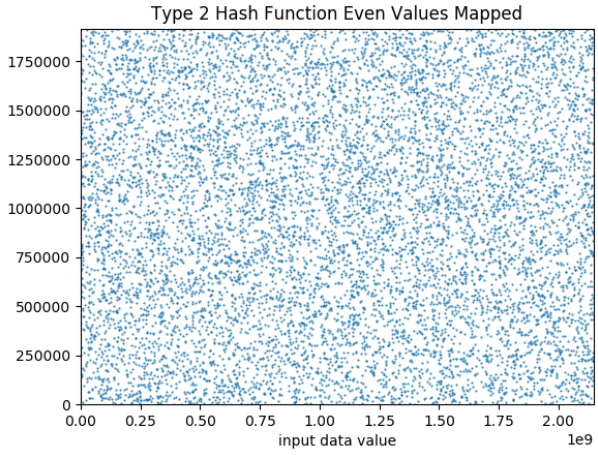
Bloom Filter Project Report

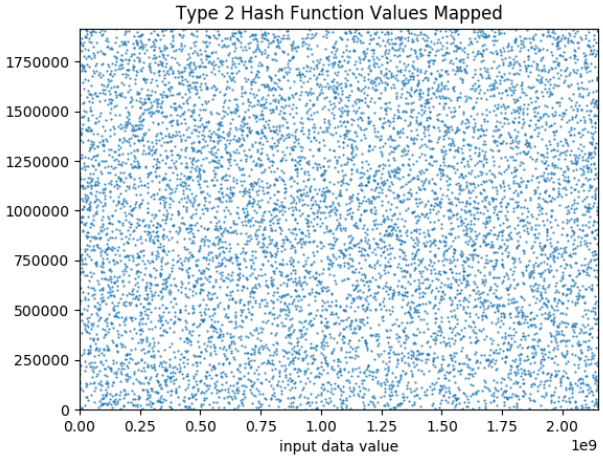
Name : Weifeng Lyu

Task 1 Plots :



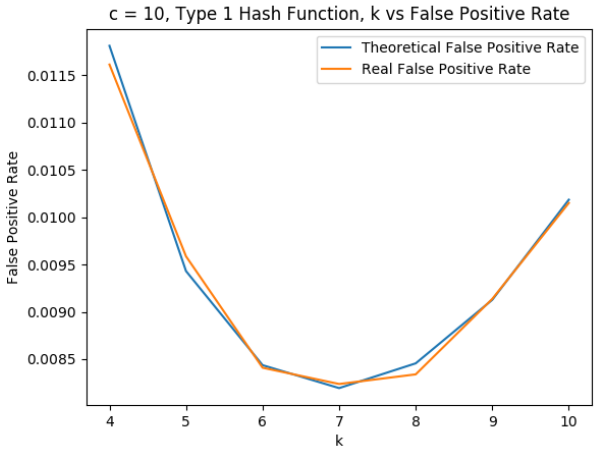



Bloom Filter Project Report

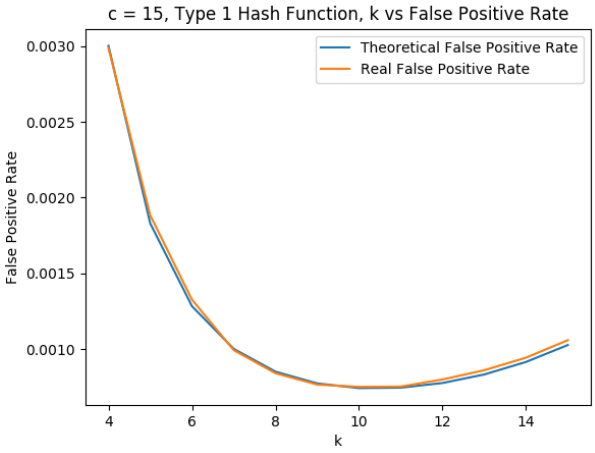
Name : Weifeng Lyu

Task 3 Plots :

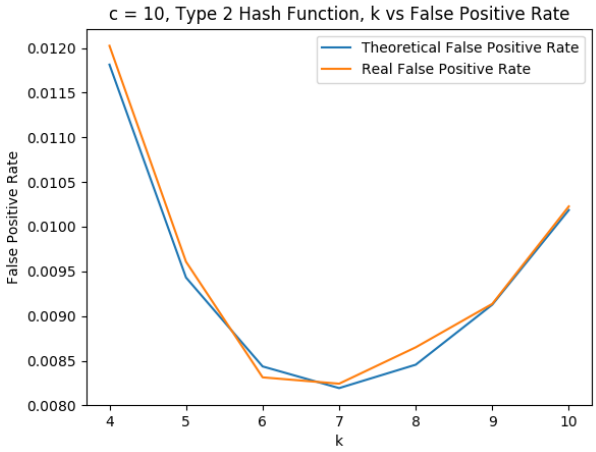
Hash Type 1; $c$ = 10




Hash Type 1; $c$ = 15




Hash Type 2; $c$ = 10




Hash Type 2; $c$ = 15