CS665 HW1: Solutions

Nicolas Garavito-Camargo

January 31, 2017

1 Part I

1. Why are you taking this course?

   I am an Astronomy PhD student and I have realized that more often data science techniques. This is because we have more and more data from telescopes or from computer simulations and the most efficient way to processes and understand this data is with data science. Unfortunately we don't have a formal class in our department and I would like to have a better understanding of data science, that's the reason I am taking this class.

2. What do you think youll learn in this course?

   I hope to learn the algorithms behind the most common techniques, to know more literature about this algorithms.

3. What programming languages are you comfortable with?

   C/C++ and Python

4. Which programming language are you most comfortable with?

   Python

5. During the course of your studies, whats the largest dataset youve had to deal with? What tools did you use?

   $\sim 100GB$, which I could split in chunks of $2Gb$, then I just read the data with Python.

6. Tell me about something cool you learned recently. Possible answers: books, movies, TV shows, blogs, podcasts, etc. It does not have to be about data science. (This is, squarely, old man tries to stay relevant. Take this either as Humor me or Tell me what I should be thinking about with respect to the class)

   A follow a nice blog that make summaries about books (often related with astronomy & arts):

   brainpickings

   There I found this two books which I found amazing: Dear Data and Illustrated field guide to keeping a visual Diary and Cultivating a capacity for creative observation

2 Part II

1. Given a stream of symbols $a_1, a_2, , a_n$ each an integer in $1, , m$, give an algorithm that will select one symbol uniformly at random from the stream. How much memory does your algorithm require?

Solution:

A random probability to select a given symbol of the stream would be $P = \frac{1}{n}$ where n in the number of symbols in the stream. If we are reading one symbol at a time, the probability of reading the symbol j would be $P = \frac{1}{\sum j}$ where j is the index of the symbol. Therefore the probability of selecting the symbol $j + 1$ is $P_{j+1} = \frac{1}{\sum (j+1)}$.

2. Give an algorithm to select an a_i from a stream of symbols $a_1, a_2, , a_n$ with probability proportional to a_i^2.

Solution:

I would start by storing the index of the elements i, then the probability of selecting an element is $P = \frac{1}{S_j}$ where $S_j = \sum a_j^2$.

For the first element that I read I have $i = 1$ and $S_1 = a_1^2/a_1^2 = 1$ therefore I pick the first value. To pick the next element I would compute the probability of picking that element: $S_2 = a_2^2/(a_1^2 + a_2^2)$. Generalizing to n elements the probability of picking a given element would be: $S_n = a_n^2/\sum_n a_n^2$.

3. How would one pick a random word from a very large book where the probability of picking a word is proportional to the number of occurrences of the word in the book?

Solution:

To tackle this problem I am going to assume that the stream is very large with length n such that a simple counting algorithm can't work. Therefore I propose the following algorithm:

- As you read the words make a hash table that stores the number of words that you have read.
- Assign a probability to each word to be picked based on the number of appearances in the hash table. The probability is going to be $P_w = n_w/M$, where M is the total words that you have read (this comes from the hash table). n_w is the number of appearances of the word, (this also comes from the hash table)
- Because the distribution of the number of appearances of the words should be a Gaussian distribution the probability P_w is going to converge as we read words. As such one can stop counting words as soon as the change of P_w is small as you read words, say $\Delta P_w = 1\%$.

4. Consider a matrix where each element has a probability of being selected. Can you select a row according to the sum of probabilities of elements in that row by just selecting an element according to its probability and selecting the row that the element is in?

Solution:

I don't think you can do this, the reason is that if your row probability is $P_r = \sum a_n$ being n the number of elements in that row and a it's values, if there are two rows one r_1 with all elements with the same probability a_1 and the other one r_2 with one element with probability a_1 and the other elements with probability $a_2 < a_1$, therefore $P_{r1} > P_{r2}$ but with the proposed method both will have the same probability of being chosen because both rows have at least one element with the same high probability.

If you the number of appearances of one element in the raw where taken into account, then it would be possible to apply this method.

5. For the streaming model give an algorithm to draw t of indices i independent samples each with the probability proportional to the value of a_i. Justify that your algorithm works correctly.

Solution:

The algorithm would be as follows:

First I would start with a probability to pick the first t indices this is $P_t = \sum a_t/S_{t,1} = 1$, where I define $S_{t,k}$ as the sum of all the indices t that I have read.

$$S_{t,k} = \sum_k \sum_t a_t \tag{1}$$

You store this indices and then you select randomly a set of indices $t' \neq t$, the probability of get the second set of t' is going to be $P_{t'} = \sum a'_t S_{t,2}$ and such, the probability of not picking this indices t' is $1 - P_{t'}$. Generalizing this to n procedures the probability of getting the nth set of t indices is:

$$P_t = \frac{\sum_{a_t}}{S_{t,k}} \tag{2}$$