Streaming algorithm

Data stream algorithms are computational methods designed to process and analyze data that arrives continuously or in a continuous flow (known as a data stream), rather than in a fixed dataset.

These algorithms operate in an "online" way, meaning they make immediate decisions or process data as it arrives, often with limited memory or resources.

They are crucial in handling large volumes of data that cannot fit into the memory of a computer or storage system for traditional processing.

Streaming algorithm: Key Characteristics

Continuous Flow:
Data streams consist of a continuous and potentially unbounded sequence of data elements. Examples include sensor data, social media feeds, network traffic, etc.

Limited Memory:
These algorithms are designed to work within limited memory resources. They process data in a single pass or with very limited access to the entire dataset.

Single Pass Processing:
Often, these algorithms only have one chance to process each data point as it arrives. They need to make immediate decisions or summarize information based on the data seen so far.

Data stream algorithms perform various tasks such as summarization, approximate query processing, classification, and anomaly detection.

Count-Min Sketch

The Count-Min Sketch is a probabilistic data structure used in streaming algorithms for estimating the frequency of elements in a data stream. It provides an approximate count of occurrences for each element within the stream while using minimal memory.

How Count-Min Sketch works

Initialization:
The Count-Min Sketch consists of a two-dimensional array of counters, usually represented as rows and columns. The number of rows determines the number of hash functions used, and the columns represent the range of the hash function outputs.

Hashing Process:
When an element arrives in the stream, it is processed by each of the hash functions.
Each hash function maps the element to a specific column in the array.

Incrementing Counters:
For each hash function, the corresponding counter in the array is incremented. This incrementation happens for every occurrence of the element in the stream.

Estimating Frequencies:
To estimate the frequency of an element, the Count-Min Sketch performs the following:

Hashes the element using the same hash functions used during insertion.

Retrieves the values from the counters corresponding to the hashed indices.

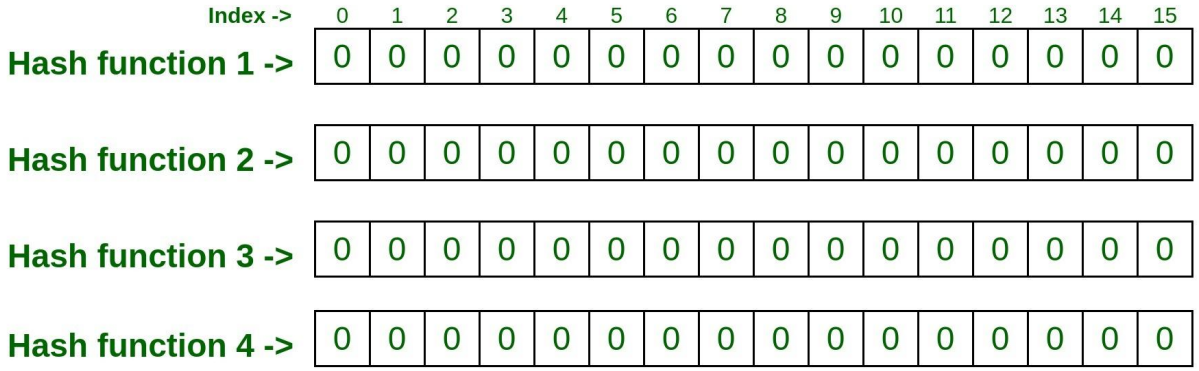
Takes the minimum value among these counters as an estimate of the frequency of the element.

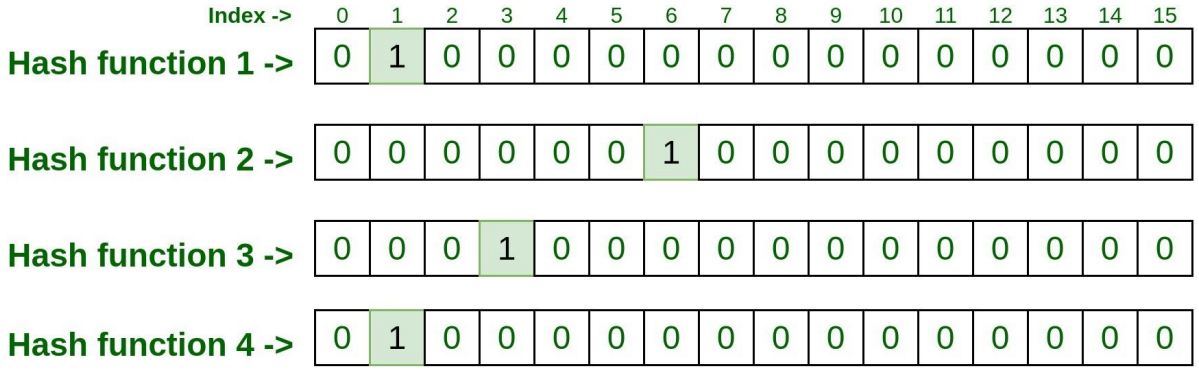
Approximation and Error Bounds:

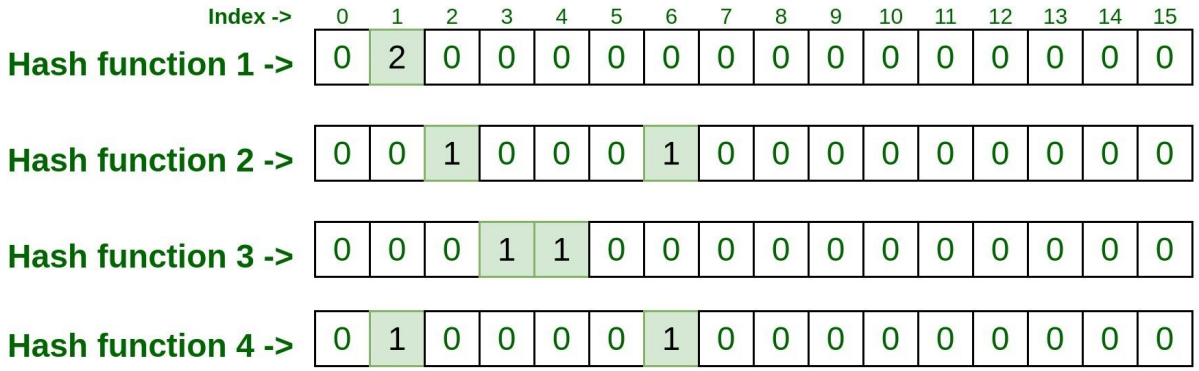
The Count-Min Sketch provides an approximate count of each element's frequency.

The accuracy of the estimate depends on the number of hash functions and the size of the counters used in the sketch.

The sketch allows for controlling the error bounds: increasing the number of hash functions or counter sizes can reduce the error but may increase memory usage.







References:

Streaming Algorithm - *Wikipedia*
https://en.wikipedia.org/wiki/Streaming_algorithm#Applications

Count-Min Sketch Data Structure - *GeeksForGeeks*
https://www.geeksforgeeks.org/count-min-sketch-in-java-with-examples/