

**Laboratory Manual**

**Subject: Machine Learning–IV (PCCS7010T) Semester: VII Class: B. Tech Experiment No. : 4**

**Title :** Approximate the number of unique elements in a data stream or database in one pass using Flajolet-Martin Algorithm.

**Pre-requisite:** Basics of Machine Learning

Hardware Requirements (if any): Windows 64 bit processor

Theory/Concept Explanation (Compulsory):

Introduction:

The Flajolet-Martin algorithm is also known as probabilistic algorithm which is manly used to count the number of unique elements in a stream or database. This algorithm was invented by Philippe Flajolet and G. Nigel Martin in 1983 and since then it has been used in various applications such as, data mining and database management.

The basic idea to which Flajolet-Martin algorithm is based on is to use a hash function to map the elements in the given dataset to a binary string, and to make use of the length of the longest null sequence in the binary string as an estimator for the number of unique elements to use as a value element.

The steps for the Flajolet-Martin algorithm are:

* First step is to choose a hash function that can be used to map the elements in the database to fixed-length binary strings. The length of the binary string can be chosen based on the accuracy desired.
* Next step is to apply the hash function to each data item in the dataset to get its binary string representation.
* Next step includes determining the position of the rightmost zero in each binary string.
* Next we compute the maximum position of the rightmost zero for all binary strings.
* Now we estimate the number of distinct elements in the dataset as 2 to the power of the maximum position of the rightmost zero which we calculated in previous step.

The accuracy of Flajolet Martin Algorithm is determined by the length of the binary strings and the number of hash functions it uses. Generally, with increase in the length of the binary strings or using more hash functions in algorithm can often increase the algorithm’s accuracy.

The Flajolet Martin Algorithm is especially used for big datasets that cannot be kept in memory or analyzed with regular methods. This algorithm, by using good probabilistic techniques, can provide a precise estimate of the number of unique elements in the data set by using less computing.

## **Need of this Algorithm**

The Flajolet-Martin algorithm, can be used to determine how many unique elements are there in a database. It is very helpful in situations where the size of the memory is large and it is difficult to process the complete dataset. The following are some of the main uses and benefits of the Flajolet-Martin algorithm:

* **Scalability:** This algorithm is considered scalable as it can be used to handle large datasets and can estimate the number of unique elements in the dataset without storing the entire dataset in the memory.
* **Memory efficiency:** This algorithm requires a small amount of memory to count the number of unique elements.  This can be achieved by using bit manipulations and hash functions to create a small representation of the data.
* **Speed:**This algorithm is suited for real-time applications since it is computationally efficient and can quickly generate count of the number of unique items by using relatively less computing.

## ****Disadvantages****

* **Accuracy:** Instead of providing an exact result, this algorithm provides us with an estimation of the count of distinct items. The accuracy depends on factors like the number of hash functions used, the length of the binary string representation etc. In some applications, where precise count is required this algorithm is not accurate enough.
* **Sensitive to dataset:** The accuracy of the Flajolet-Martin algorithm is also influenced by the distribution and characteristics of the dataset. It may have better accuracy on datasets having uniform or random distributions, but perform less accurately on datasets with skewed distributions or some specific patterns.
* **Hash Function selection:** The performance and accuracy of this algorithm can be influenced by the hash functions used in the algorithm. It is very important to select appropriate hash functions to maintain a balance between accuracy and efficiency.
* **Limited applicability:** The Flajolet-Martin algorithm is mainly designed for estimating the number of unique elements and can’t be used for any other data analysis tasks. It does not provide insights information about the specific elements or their frequencies. Its main goal is on estimation.

## ****Conclusion****

Finally, the Flajolet Martin Algorithm is an effective method for calculating the count of number of unique items in huge data sets. This approach works by mapping the data set elements to the binary strings using hash functions, and then make use of the length of the longest run of zeros in the binary strings as an estimation for the number of unique elements. While this process is also known as probabilistic as it will not always offer a precise count, it is an effective method for finding insights of big data sets that would be difficult or impossible to analyses using regular methods.

**Implementation:**

import random

import math

def trailing\_zeros(x):

""" Counting number of trailing zeros

in the binary representation of x."""

if x == 0:

return 1

count = 0

while x & 1 == 0:

count += 1

x >>= 1

return count

def flajolet\_martin(dataset, k):

"""Number of distinct elements using

the Flajolet-Martin Algorithm."""

max\_zeros= 0

for i in range(k):

hash\_vals = [trailing\_zeros(random.choice(dataset))

for \_ in range(len(dataset))]

max\_zeros = max(max\_zeros, max(hash\_vals))

return 2 \*\* max\_zeros

# Example

dataset = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

dist\_num = flajolet\_martin(dataset, 10)

print("Estimated number of distinct elements:", dist\_num)

**Output:**

'Estimated number of distinct elements:’ 8