# "UNIT - 02"

DATE: __/__/____
PAGE: _____

Data Stream: Continuous, fast-changing, and ordered chain of data transmitted at a very high speed.

⇒ An ordered sequence of information for a specific interval.

⇒ Sender's data is transferred from sender's side and immediately shows in data streaming at the receiver's side.

⇒ Streaming does not mean downloading the data or storing the information on storage devices.

⇒ A data stream is potentially unbounded sequence of tuples. Each tuple consist of a set of attributes similar to a row in database table.

⇒ Transactional data stream: log interactions between entities.
- Credit card.
- Telecommunications
- Web

⇒ Measurement data streams: monitor evolution of entity states
- Sensor networks - road traffic
- IP networks, traffic at routers interfaces
- Earth climate - temperature, moisture

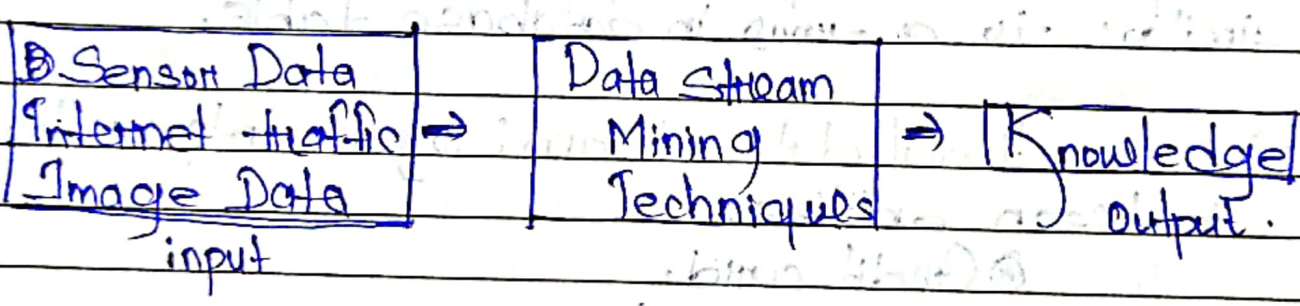
Good Write

DATE: __/__/____
PAGE: _____

## Sources of data stream:

i) Internet traffic
ii) Sensors data
iii) Real time ATM transaction.
iv) Live event data.
v) Call records
vi) Satellite data
vii) Audio listening
viii) Watching videos.

Data stream Generators




Sensor Data: In navigation systems, sensor data is used. Imagine a temperature sensor floating about in the ocean, sending back to the base station a reading of the surface temperature each hour. The data generated by this sensor is a stream of real numbers.

Image Data: Satellites frequency send down to earth streams containing many terabyte of images per day. Surveillance cameras generate images with lower resolution than satellites, but there can be numerous of them, each producing a stream of images at a break of 1 second each.

3) Internet and Web traffic: A bobbing node in the center of the internet receives streams of IP packets from many inputs and paths them to it's outputs. Websites receive streams of heterogenous types. For example: Google receives a hundred million search queries per day.

## Data streams in Data mining:

⇒ Data streams in Data mining is extracting knowledge and valuable insights from a continuous stream of data using stream processing software.

⇒ Data streams in data mining can be considered a subset of general concepts of machine learning, knowledge extration and data mining.

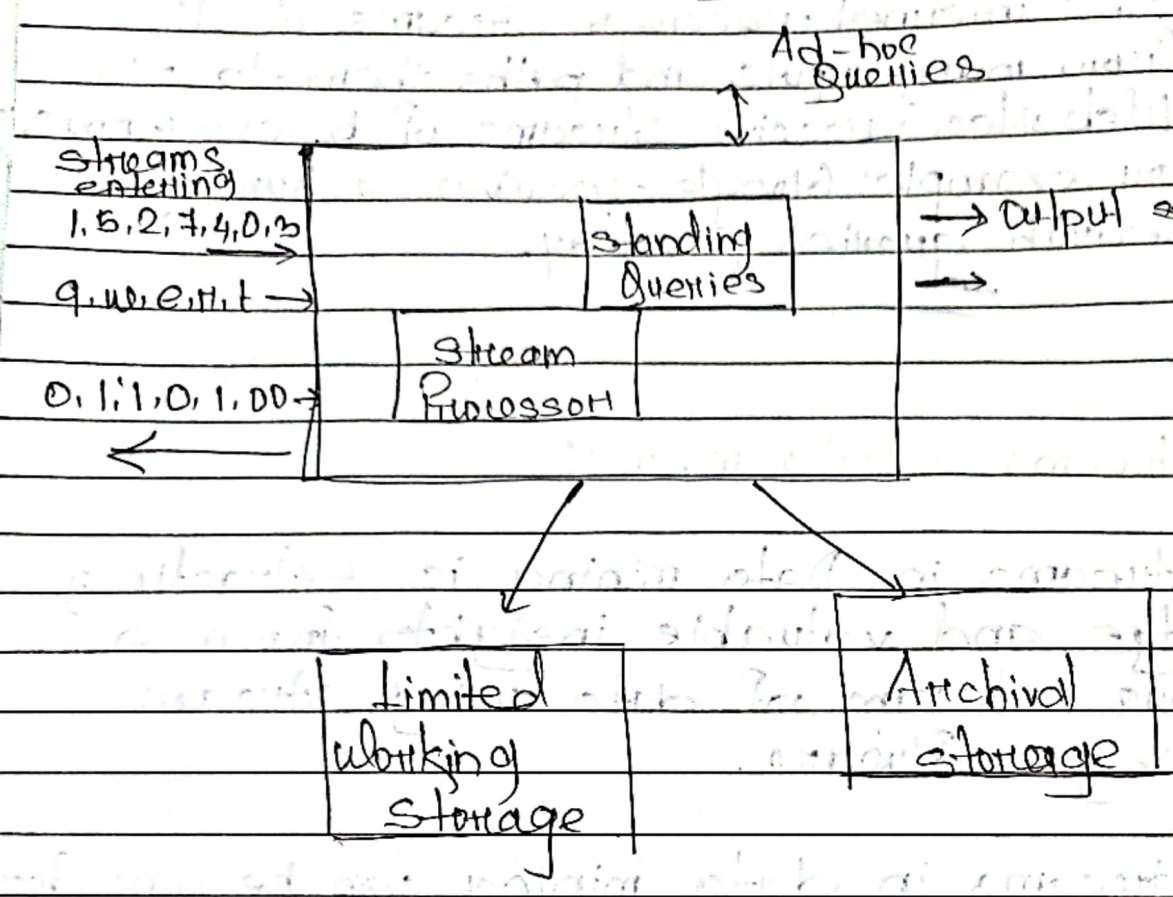
⇒ Data analysis of large amount of data needs to be done in real-time.

⇒ The structure of knowledge is extracted is represented in the case of models and patterns of infinite streams of information.

Good Write

DATE: __/__/____
PAGE: _____

## A data-stream management system (DSMS):




⇒ Streams may be archived in a large archival store, but we assume it is not possible to answer queries from the archival store.

⇒ DSMS would be examined only under special circumstances using time-consuming retrieval process.

⇒ There is also a working store, into which summaries or parts of streams may be placed, and which can be used for answering queries.

⇒ The working store might be disk, or it might be main memory, depending on how fast we need to process queries.

Good Write

DATE: __/__/____
PAGE: _____

⇒ But either way, it is of sufficiently limited capacity that it cannot store all the data from all the streams.

⇒ Any number of streams can enter the system.
⇒ Each stream can provide elements at its own schedule; they need not have the same data rates or data types, and the time between elements of one stream need not be uniform.

## Problems on Data Streams:

i) Sampling data from a stream: Construct a random sample.
ii) Queries over sliding windows: Number of items of type x in the last k elements of the stream.
iii) Filtering a data stream: Select elements with property x from the stream.
iv) Counting distinct elements: Number of distinct elements in the last k elements of the stream.
v) Estimating moments: Estimate avg / std dev of last k elements.
vi) Finding frequent elements.

DATE: / /
PAGE:

## Applications:

i) Mining query streams: Google wants to know what queries are more frequent today than yesterd

ii) Mining click streams: Yahoo wants to know which of its pages are getting an unusal number of hits in the past hour.

iii) Mining social network newsfeed: look for trending topics on Twitter, Facebook.

iv) Sensor network: Many sensors feeding int a central controller.

v) Telephone call records: Gather information for optimal routing, Detect denial-of-Ser attacks

vi) Data stream processing: Process queries, apply data mining algorithms.

v) Requirements:

→ Real time processing
→ One-pass processing
→ Bounded storage
→ Possibly consider several streams.

## Characteristics of Data stream in b data mining:

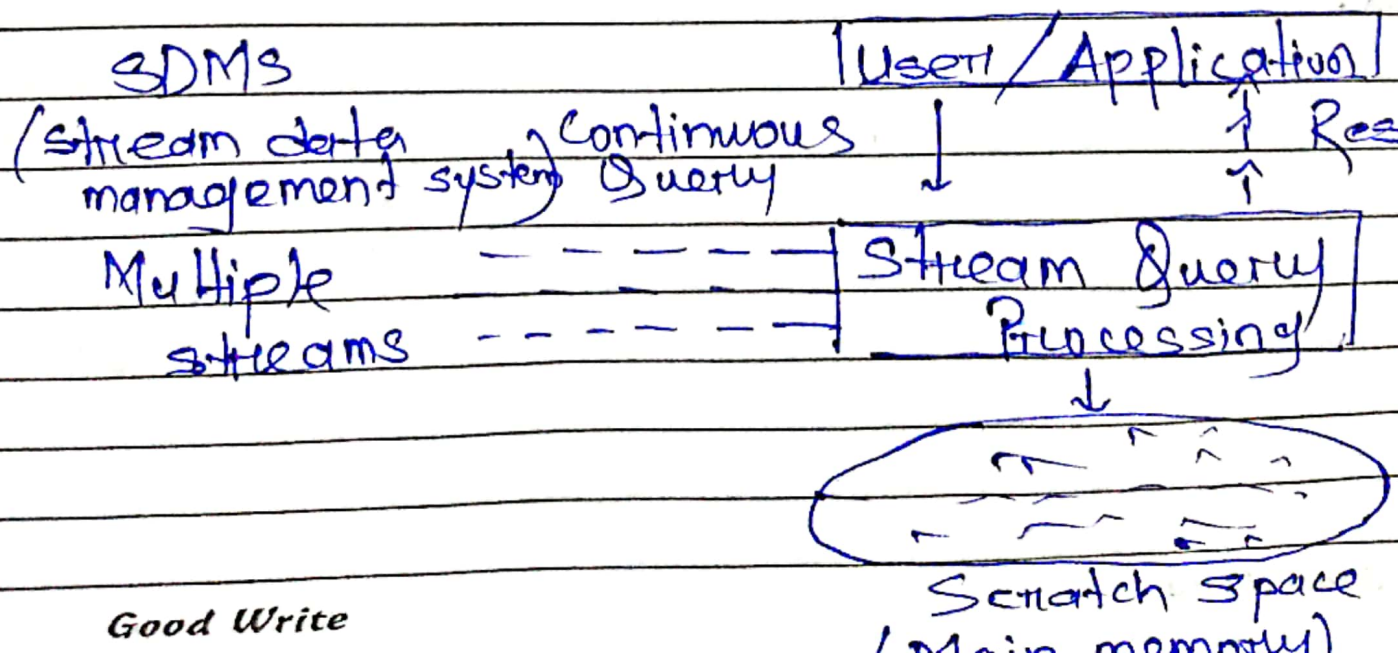
i) Continuous stream of data: Infinite continuous stream resulting in ~~data~~ big data. In data streaming, multiple data streams are passed simultaneously.

ii) Time sensitive: Elements of data streams carry time stamps with them. After a particular time, the data stream loses its significance.

iii) Data Volatility: No data is stored in data streaming. Once the data mining and analysis are done, information is summarized or discarded.

iv) Concept Drifting: Very unpredictable. The data changes or evolves with time, as in this dynamic world, nothing is constan

## Stream Query Processing architecture:



DATE: __/__/____
PAGE: _____

Difference between Database Management system and data stream management system:

| DBMS | DSMS |
|---|---|
| i) Permanent updatable relations. | i) Streams and permanent updatable relations. |
| ii) Data stored in disks. | ii) Streams are processed on fly and permanent relation stored in disks |
| iii) SQL language. | iii) SQL like query language |
| iv) Performs in large volume of data. | iv) Optimization of computer resources to deal with several streams, several queries. |
| v) One-time queries | v) Continuous queries. |
| vi) Random access. | vi) Sequential access. |
| vii) Unbounded disk store. | vii) Bounded disk store |
| viii) Only current state matters. | viii) Historical data is important. |
| ix) No real time service. | ix) Real-time requirements. |
| x) Assume precise data. | x) Data stale/imprecise. |

DATE: __/__/____
PAGE: _____

## Challenges of Stream data processing:

i) Multiple, continuous, rapid, time-varying, ordered streams.
ii) Main memory computations.
iii) Queries are often continuous.
iv) Queries are often complex.
v) Multi-level / Multi-dimensional processing & data mining.

## Stramping Computing Approaches:

⇒ Two approaches for handling such streams:
i) Use a time window
ii) Query the window as static table

⇒ When one can't store collected data, or to keep tracks of historical data
i) Sampling
ii) Filtering
iii) Counting

## Data streams in data mining techniques:

⇒ Data streams in data mining techniques are implemented to extract patterns and insights from a data stream.

DATE: __/__/____
PAGE: _____

## Data Stream Mining Techniques:

1) Classification: Classification is a supervised learning technique. In classification, the classifier model is built based on the training data.
→ This classifier model is then used to predict the label for unlabeled instances or items continuously arriving through the data stream.
→ Prediction is made for the unknown/new items that the model never seen and already known instances are used to train the model.

→ Receives an unlabeled ~~de~~ item and predict it based on it's current model.
→ Receives labels for past known items and use them for the training the model.

i) Lazy Classifier or K-nearest Neighbour.
ii) Naive Bayes
iii) Decision Tree
iv) Logistic Regression.

ii) Regression: Supervised learning technique used to predict real values of label attributes for the stream instances, not the discrete values like classification.
→ The idea of regression is similar to classification either to predict the real values label for unknown items using the regression model.

Algorithms:

i) Lazy classifier or K-nearest Neighbour
ii) Naive Bayes
iii) Decision trees
iv) Linear Regression
v) Esembles.

iii) Clustering: Unsupervised learning technique.

⇒ Clustering is functional when we have unlabeled instances, and we want to find homogenous clusters in them based on the similarities of data item.

⇒ Before the clustering process, the groups are not known. Clusters are formed with continuous data streams based on data and keep on adding items to the different groups.

Algorithms:

i) K-means clustering
ii) Hierarchical Clustering
iii) Density-based clustering

iv) Frequent Pattern mining: An essential task in unsupervised learning.

⇒ It is used to describe the data and find the association rules or discriminative features in data that will further help classification and clustering tasks. Based on two rules:

i) Frequent Item Set: Collection of items occurring together frequently.
ii) Association rules: Indicator of the strong relationship between two items

DATE: __/__/____
PAGE: _____

## Sampling from a Data Stream:

Since we cannot store the entire stream, one obvious approach is to store a sample

Two different problems:

i) Sample a fixed proportion of elements in the stream.

ii) Maintain a random sample of fixed size over a potentially infinite stream.

### 1) Problem 1: Sampling fixed proportion:

Scenario ⇒ Search engine query stream

- Stream of tuples: (user, query, time)
- Answer questions such as:
  "How often did a user run the same query in a single day?"
- Have space to store 1/10th of query stream.

Naive solution ⇒

i) Generate a random integer in [0....9] for each query.

ii) Store the query if the integer is 0, otherwise discard.

Solution ⇒

i) Pick 1/10th users and take all their searches in the sample

ii) Use a hash function that hashes the user name or user id uniformly into 10 buckets.

DATE: __/__/____
PAGE: _____

ii) Maintaining a fixed-size sample:

Suppose, we need to maintain a random sample S of size exactly s tuples.

- e.g. main memory size constraint.

Why? Do not know length of stream in advance.

Suppose at time n, we have seen n items.

- Each item is in the sample S with equal probability.

How to think about the problem: say s=2.

Stream: a x c y z k c d e g

At n=5, each of the first 5 tuples is included in the sample S with equal prob.

At n=7, each of the first 7 tuples is included in the sample S with equal prob.

Impractical solution would be to store all the n tuples seen. So, Fair and out of them pick s at random.

Solution: Fixed Size Sample

Reservoir Sampling Algorithm:

- Store all the first s elements of the stream S
- Suppose we have seen n-1 elements, and now the $n^{th}$ element arrives. (n>s).
  - With probability s/n, keep the $n^{th}$ element, else discard it
  - If we picked the $n^{th}$ element, then it replaces one of the s elements in the sample S, picked uniformly at random

Good Write

DATE: __/__/____
PAGE: _____

Claim: This algorithm maintains a sample S with the desired property:
- After n elements, the sample contains each element seen so far with probability s/n.

Filtering of Data Stream: Filtering condition of a stream item is independent of other items of the same stream or any other data stream.

The most common example of such filtering is stream sampling, when each item is filtered out with a certain probability and the remaining items form the desired sample.

Bloom filter:

i) A bloom filter consist of:
- → an array of n bits, initially all 0's.
- → A collection of hash-function h1, h2, ..... hk.
- → A set S of m key values.

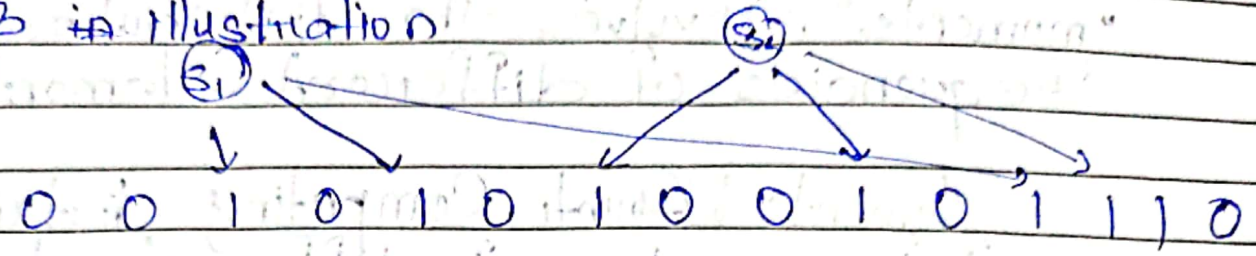
ii) The Bloom filter allows through all stream elements whose keys are in S, while rejecting most of the stream elements whose keys are not in S.

DATE: __/__/____
PAGE: _____

## Illustration of Bloom filters:

→ Use K independent hash functions instead of 1.

→ K = 3 in illustration




## Counting Distinct Elements in a stream:

### Flajolet Martin Algorithm →

Flajolet Martin Algorithm, also known as FM algorithm, is used to approximate the number of unique elements in a data stream or database in one pass.

### Algorithm:

i) Selecting a hash function h so each element in the set is mapped to a string to at least $\log_2 n$ bits.

ii) For each element x, $r(x)$ = length of trailing zeroes in $h(x)$.

iii) $R = \max(r(x))$

= Distinct elements = $2^R$

DATE: __/__/____
PAGE: _____

## Estimating Moments:

→ Estimating moments is a generalization of the problem of counting distint elements in a stream. The problem, called computing "moments", involves the distribution of frequencies of different elements.

Moments: Count. Computing distribution of frequencies of different elements in stream.

→ Suppose a stream consists of elements chosen from universal set. Assume the universal set is ordered so we can speak of the $i^{th}$ element for any i.

→ Let, $m_i$ be the number of occurrences of the ith element for any i. Then the $K^{th}$ order moment of the stream is the sum over all i of $(m_i)^K$

## Alon-Matias-Szegedy Algorithm: (AMS)

→ AMS method works for all moments.
→ Gives an unbiased estimate
→ We pick, & will just concentrate on the 2nd moment of S.

Second moment: (Varience). It measures the spread of values of distribution or how far ... from the normal.

→ Our goal is to compute $S = \sum_i m_i^2$

Algorithm:

i) Choose a random hash function $h: \{1, 2, \ldots m\} \to \{1, 2, \ldots m^3\}$ from a pairwise independent hash family.
ii) Initialize $z = 0$.
iii) For each item $x$ of the stream, update $z = Max(zeros(h(x)), z)$ where zeros(y) denotes the number of training zeros of y in the binary representation.
iv) Output $2^{z+e}$ for $e = 1/2$.

$$E[f(x)] = 1/n \sum_{i=1}^{n} n(2e_i - 1)$$

## Counting oneness of a window:

→ Suppose we have a window of length N on a binary stream. We want at all times to be able to answer queries of the form "how many 1's are there in the last K bits". For any $K \leq N$.

→ To solve this problem, we use DGIM algorithm.

→ DGIM = Datar - Gionis - Indyk - Motwani Algorithm.

DATE: __/__/____
PAGE: _____

## DGIM Algorithm:

Algorithm that uses $O(\log^2 N)$ bits to represent a window of N bit, and allows to estimate the number of 1's in the window with an error of no more than 50%.

→ Each bit stream has a timestamp for the position at which it arrives.
→ The first bit has timestamp 1, the second bit has 2 and so on.
→ The positions are recognized with the window size N.
→ The timestamp is represented with modulo N, and are represented as $\log_2 N$ bits.

The windows are divided into buckets consisting of—

→ The timestamp at it's right end
→ the numbers of 1's must be in the power of 2 which are referred to as size of bucket.

### Rules for forming Bucket:

→ The right side of the bucket should always start with 1.
→ Every bucket should have at least one 1, else no bucket can be formed.
→ All bucket sizes should be a power of 2

Updating the buckets:

- When a new bit comes in, drop the last bucket if it's end-time is prior to N time units before the unit current time.
→ If the current bit is 0, no other changes are needed.

→ If the current bit is 1.
- Create a new bucket of size 1, for just this bit.
End timestamp = current time.
- If there are now three buckets of size 1, Combine the oldest two into a bucket of size 2.

- Decaying Window Algorithm: This algorithm allows you to identify the most popular elements in an incoming data streams.

The decaying window algorithm not only tracks the most recurring elements in an incoming data stream, but also discounts any random spikes or spam requests that might have boosted an element's frequency.

→ In decaying window, assign a score or weight to every element of the incoming data stream.
→ Need to calculate the aggregate sum for each distinct element by adding all the weights assigned.

Good Write

DATE: __/__/____
PAGE: _____

→ In decaying window algorithm, assign more weight to newer elements.

→ For new element, first reduce the weight of all the existing elements by a constant factor k.

→ Assign the new element with specific weight.

The aggregate sum of the decaying exponential weights can be calculated using the following formula:

$$\sum_{i=0}^{t-1} a_{t-i} (1-c)^i$$

c = small constant of order $10^{-6}$ or $10^{-9}$.

New element = $a_{t+1}$

i) Multiply the current sum/score by the value $(1-c)$.

ii) Add the weight corresponding to the new element.