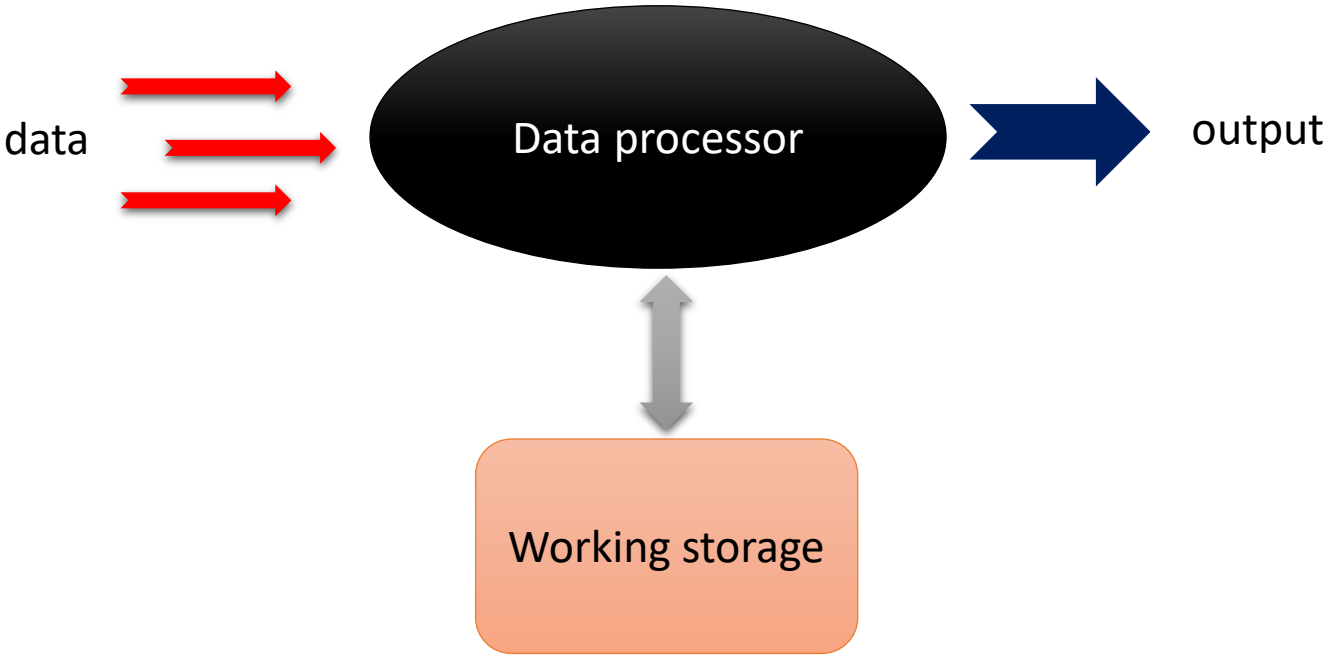
Mining Data Streams

Week 7

Nurul Afiqah Burhanuddin
nurul.afiqah@ukm.edu.my
Room 2119

Introduction

- Data stream: continuous accumulation of data at a rapid rate for real-time processing



Introduction

- In machine learning: online learning.

- Examples:
  - Online advertisement by continuously analyzing clickstreams on shopping sites.
  - Organization tracks changes in public sentiment on online social networks such as Twitter that continuously generate text data.
  - Traffic monitoring systems that collect data using sensors.
  - Online financial transactions, like credit card purchases generate time-critical data that need to be processed for real-time actions.
  - Data from health monitoring devices.

Introduction

Challenges in mining data streams:

- One-pass constraint: the data can be processed only once.
- The data may evolve over time
- Limited processing power and memory

Introduction

Batch data	Streaming data
Data processing within a time span	Data processing in real time
Complex data analytics	Simple/fast tools and functions
Data size is known and finite	Data size unknown and volatile
Multiple passes	One pass

Introduction

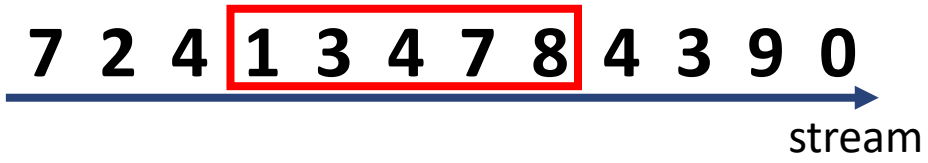
Basic tools used in mining streaming data:

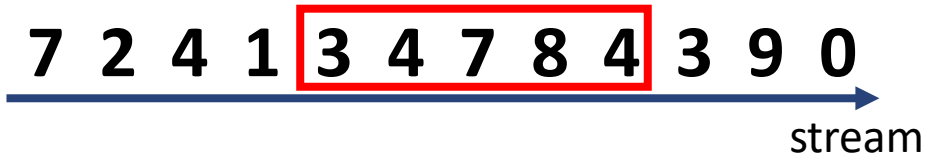
- Stream window
- Reservoir sampling

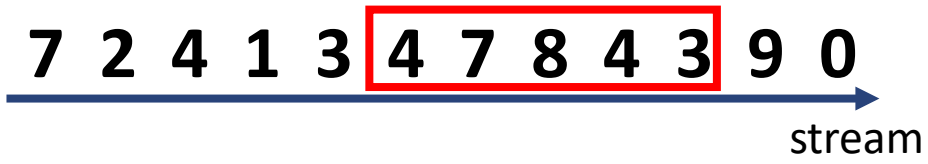
Stream Window

- Assume that recent data is more useful and pertinent than older data.
- Window types:
    1. Count-based window
    2. Time-based window
    3. Punctuation-based window

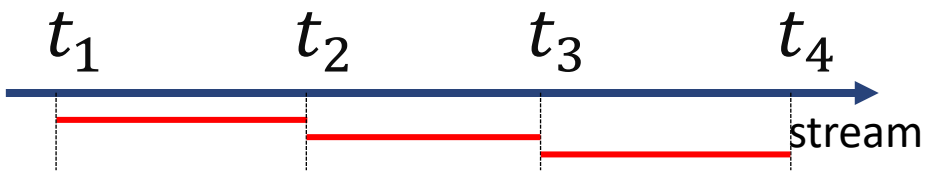
Stream Window – Count-based



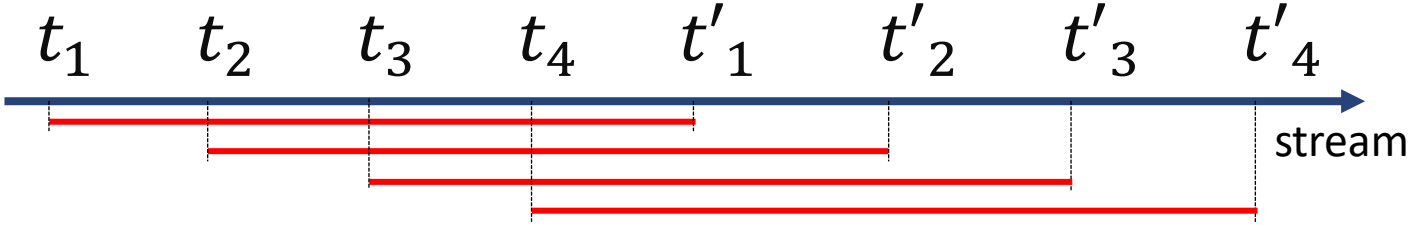




Stream Window – Time-based

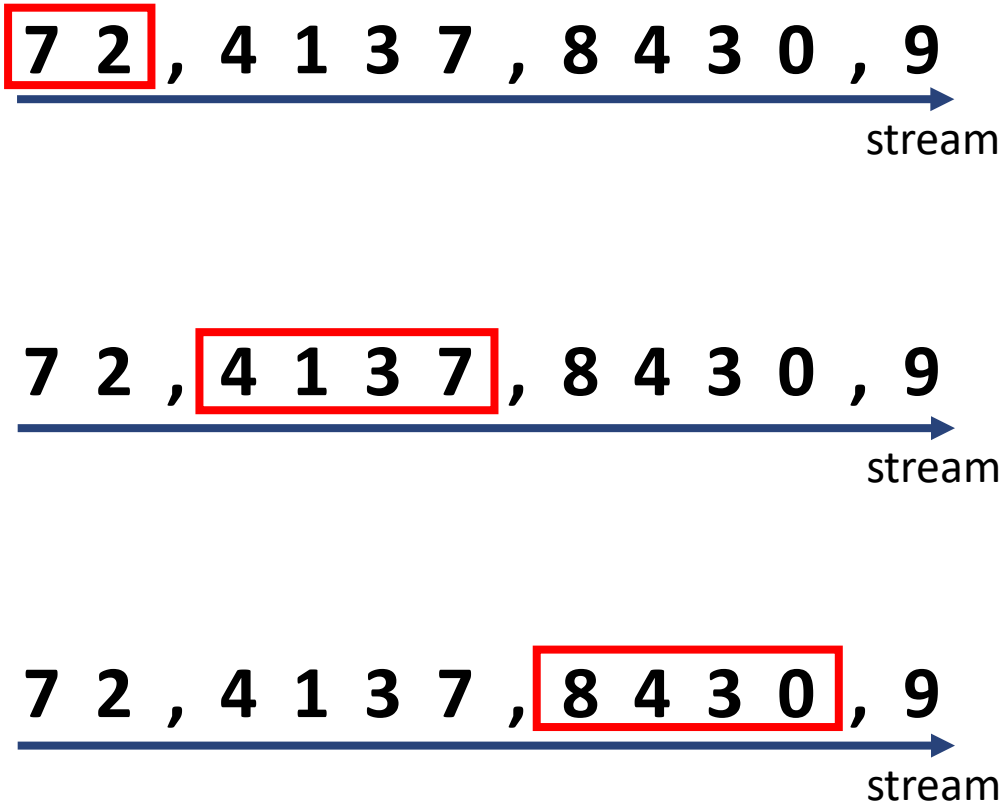


tumbling window



sliding window

Stream Window – Punctuation-based



Reservoir Sampling

- Allows the maintenance of a random sample without replacement of a particular size in an online fashion.
- A reservoir (usually represented by an array) of elements is maintained whilst the input is read sequentially.
- New elements replace those in the reservoir with uniform probability.
- Suppose the objective is to maintain a random sample of n elements from a stream of N elements, where N is not known a priori and $N >> n$. Let the stream elements be $x_1, x_2, \ldots, x_N$.

  We want:

  a simple random sample of n

  with the constraints:

  - x_i can only be read once for $i \in \{1, \ldots, N\}$
  - x_i must be read before x_j for $i < j$ with no guarantee of any structure in the ordering.

Reservoir Sampling

- The reservoir sampling algorithm proceeds as follows. For a reservoir of size n, the first n data points in the stream are always included in the reservoir. Subsequently, for the tth incoming stream data point, the following two admission control decisions are applied:

    1. Insert the tth incoming data point into the reservoir with probability n/t.

    2. If the newly incoming data point was inserted, then eject one of the old data points in the reservoir at random to make room for the newly arriving point.

- A common algorithm for reservoir sampling, Algorithm R:

    1. Initialize an empty reservoir $R = \{R_1, \dots, R_n\}$ with $x_1, \dots, x_n$.

    2. For the subsequent input x_t, generate a random number u uniformly in $\{1, \dots, t\}$.

    3. If $u \in \{1, \dots, n\}$, set $R_u = x_t$. Otherwise discard x_t.