# Chapter XIII
# Mining Data-Streams

**Hanady Abdulsalam**
*Queen's University, Canada*

**David B. Skillicorn**
*Queen's University, Canada*

**Pat Martin**
*Queen's University, Canada*

## ABSTRACT

*Data analysis or data mining have been applied to data produced by many kinds of systems. Some systems produce data continuously and often at high rates, for example, road traffic monitoring. Analyzing such data creates new issues, because it is neither appropriate, nor perhaps possible, to accumulate it and process it using standard data-mining techniques. The information implicit in each data record must be extracted in a limited amount of time and, usually, without the possibility of going back to consider it again. Existing algorithms must be modified to apply in this new setting. This chapter outlines and analyzes the most recent research work in the area of data-stream mining. It gives some sample research ideas or algorithms in this field and concludes with a comparison that shows the main advantages and disadvantages of the algorithms. It also includes a discussion and possible future work in the area.*

## INTRODUCTION

Since many recent applications such as Internet traffic monitoring, telecommunications billing, near-earth asteroid tracking, closed-circuit television, and sales tracking produce a huge amount of data to be monitored, it is not practical to store the data physically. The data is instead presented as continuous streams. We define a data-stream to be an endless, real-time, and ordered sequence of records. Systems that analyze such streams have been called data-stream management systems

(DSMSs). Because streams are endless, results and models that depend on observing the entire data cannot be computed exactly, and some kind of approximation is required. Because streams are real-time, analysis should be fast enough to accommodate high input rates. Otherwise, the underfitting problem might occur; that is, although there is enough data to produce complex models, only simple and inaccurate models are produced since the system is unable to take full advantage of the data (Domingos & Hulten, 2001). Analysis, moreover, cannot require more than amortized constant time for each record, and analysis that depends on multiple passes over the data cannot be carried out, at least not without new algorithms.

A number of example DSMSs appear in the literature. Some are general DSMSs, for example, STREAM (Arasu, et al., 2003; Babcock, Babu, Datar, Motwani, & Widow, 2002), the Stanford data-stream management system, and Aurora (Abadi, et al., 2003). Others were developed for special applications; for example, COUGAR (Bonnet, Gehrke, & Seshadri, 2001) is a sensor system developed at Cornell University, used in sensor networks for monitoring and managing data, and the Tribeca network monitoring system (Sullivan & Heybey, 1998) is a DSMS designed to support network traffic analysis.

We now consider the main subject of this chapter: data-stream mining. Data-stream mining poses new challenges, such as understanding the trade-offs between accuracy and limited access to the data records; developing new algorithms that avoid multiple passes over the data while still producing similar results; and understanding the relationship between the amount of data seen and accuracy. Performance issues are of critical importance since they determine how much processing can be applied per data object.

Three kinds of data-stream mining can be distinguished:

1. **Occurrence mining:** The stream is continuously scanned for occurrences of a particular pattern or set of patterns. For example, the stream may be scanned for records with particular attribute values that trigger an alarm, or for certain combinations of records occurring in close proximity. Occurrence mining is similar to the use of continuous queries (Terry, Goldberg, Nichols, & Oki, 1992) in database systems, which are queries issued once and executed continuously over a data-stream upon receiving new data points. We will not discuss occurrence mining further.
2. **Multipass mining:** Extracting information requires more than one pass over the data. Clearly such a model cannot be built from the stream directly but requires some sample to be collected and used as if it were a standard dataset.
   Windows of adjacent records are often used to extract a sample that can be made available for off-line analysis. Windows may be defined as:
   - **Time based:** An interval of timestamps on the data records, for example, all records from the last hour.
   - **Order based:** An interval of record identifiers, for example, the last 100 records or the recent set of records that can fit into the memory buffer.

   Standard data-mining techniques can be applied in multipass mining with little, if any, change so we will not discuss them further.
3. **Online mining:** A model of the data is built continuously and incrementally from the records as they flow into the system. Such models can be simple, for example, accumulating the sum of some attribute of each of the records, or can be complicated, for example, building a decision tree based on the stream as training data.
   There are three important classes of online mining techniques: