Encyclopedia of Data Warehousing and Mining

Second Edition

John Wang
Montclair State University, USA

Volume III
K–Pri

Information Science REFERENCE

INFORMATION SCIENCE REFERENCE
Hershey · New York

Director of Editorial Content: Kristin Klinger
Director of Production: Jennifer Neidig
Managing Editor: Jamie Snavely
Assistant Managing Editor: Carole Coulson
Typesetter: Amanda Appicello, Jeff Ash, Mike Brehem, Carole Coulson, Elizabeth Duke, Jen Henderson, Chris Hrobak, Jennifer Neidig, Jamie Snavely, Sean Woznicki
Cover Design: Lisa Tosheff
Printed at: Yurchak Printing Inc.

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue, Suite 200
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com/reference

and in the United Kingdom by
Information Science Reference (an imprint of IGI Global)
3 Henrietta Street
Covent Garden
London WC2E 8LU
Tel: 44 20 7240 0856
Fax: 44 20 7379 0609
Web site: http://www.eurospanbookstore.com

Library of Congress Cataloging-in-Publication Data

Encyclopedia of data warehousing and mining / John Wang, editor. -- 2nd ed.
p. cm.
Includes bibliographical references and index.
Summary: "This set offers thorough examination of the issues of importance in the rapidly changing field of data warehousing and mining"--Provided by publisher.
ISBN 978-1-60566-010-3 (hardcover) -- ISBN 978-1-60566-011-0 (ebook)
1. Data mining. 2. Data warehousing. I. Wang, John,
QA76.9.D37E52 2008
005.74--dc22
2008030801

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Mining Data Streams

Tamraparni Dasu
AT&T Labs, USA

Gary Weiss
Fordham University, USA

INTRODUCTION

When a space shuttle takes off, tiny sensors measure thousands of data points every fraction of a second, pertaining to a variety of attributes like temperature, acceleration, pressure and velocity. A data gathering server at a networking company receives terabytes of data a day from various network elements like routers, reporting on traffic throughput, CPU usage, machine loads and performance. Each of these is an example of a data stream. Many applications of data streams arise naturally in industry (networking, e-commerce) and scientific fields (meteorology, rocketry).

Data streams pose three unique challenges that make them interesting from a data mining perspective.

1. **Size:** The number of measurements as well as the number of attributes (variables) is very large. For instance, an IP network has thousands of elements each of which collects data every few seconds on multiple attributes like traffic, load, resource availability, topography, configuration and connections.
2. **Rate of accumulation:** The data arrives very rapidly, like "water from a fire hydrant". Data storage and analysis techniques need to keep up with the data to avoid insurmountable backlogs.
3. **Data transience:** We get to see the raw data points at most once since the volumes of the raw data are too high to store or access.

BACKGROUND

Data streams are a predominant form of information today, arising in areas and applications ranging from telecommunications, meteorology and sensor networks, to the monitoring and support of e-commerce sites. Data streams pose unique analytical, statistical and computing challenges that are just beginning to be addressed. In this chapter we give an overview of the analysis and monitoring of data streams and discuss the analytical and computing challenges posed by the unique constraints associated with data streams.

There are a wide variety of analytical problems associated with mining and monitoring data streams, such as:

1. Data reduction,
2. Characterizing constantly changing distributions and detecting changes in these distributions,
3. Identifying outliers, tracking rare events and anomalies,
4. "Correlating" multiple data streams,
5. Building predictive models,
6. Clustering and classifying data streams, and
7. Visualization.

As innovative applications in on-demand entertainment, gaming and other areas evolve, new forms of data streams emerge, each posing new and complex challenges.

MAIN FOCUS

The data mining community has been active in developing a framework for the analysis of data streams. Research is focused primarily in the field of computer science, with an emphasis on computational and database issues. Henzinger, Raghavan & Rajagopalan (1998) discuss the computing framework for maintaining aggregates from data using a limited number of passes. Domingos & Hulten (2001) formalize the challenges, desiderata and research issues for mining data streams. Collection of rudimentary statistics for data streams is addressed in Zhu & Sasha (2002) and Babcock, Datar, Matwani & O'Callaghan (2003). Clustering (Aggarwal, Han, Wang & Yu, 2003), classification, association rules (Charikar, Chen & Farach-

Colton, 2002) and other data mining algorithms have been considered and adapted for data streams.

Correlating multiple data streams is an important aspect of mining data streams. Guha, Gunopulous & Koudas (2003) have proposed the use of singular value decomposition (SVD) approaches (suitably modified to scale to the data) for computing correlations between multiple data streams.

A good overview and introduction to data stream algorithms and applications from a database perspective is found in Muthukrishnan (2003). Aggarwal (2007) has a comprehensive collection of work in the computer science field on data streams. In a similar vein, Gaber (2006) maintains a frequently updated website of research literature and researchers in data streams.

However, there is not much work in the statistical analysis of data streams. Statistical comparison of signatures of telecommunication users was used by Cortes & Pregibon (2001) to mine large streams of call detail data for fraud detection and identifying social communities in a telephone network. Papers on change detection in data streams (Ben-David, Gehrke & Kifer, 2004; Dasu, Krishnan, Venkatasubramanian & Yi, 2006) use statistical approaches of varying sophistication. An important underpinning of statistical approaches to data mining is density estimation, particularly histogram based approaches. Scott (1992) provides a comprehensive statistical approach to density estimation, with recent updates included in Scott & Sain (2004). A tutorial by Urbanek & Dasu (2007) sets down a statistical framework for the rigorous analysis of data streams with emphasis on case studies and applications. Dasu, Koutsofios & Wright (2007) discuss application of statistical analysis to an e-commerce data stream. Gao, Fan, Han & Yu (2007) address the issue of estimating posterior probabilities in data streams with skewed distributions.

Visualization of data streams is particularly challenging, from the three perspectives dimensionality, scale and time. Wong, Foote, Adams, Cowley & Thomas (2003) present methods based on multi dimensional scaling. Urbanek & Dasu (2007) present a discussion of viable visualization techniques for data streams in their tutorial.

Data Quality and Data Streams

Data streams tend to be dynamic and inherently noisy due to the fast changing conditions.

An important but little discussed concern with data streams is the quality of the data. Problems could and do arise at every stage of the process.

Data Gathering: Most data streams are generated automatically. For instance, a router sends information about packets at varying levels of detail. Similarly an intrusion detection system (IDS) automatically generates an alarm on a network when a predefined rule or condition is met. The data streams change when the rule settings are changed either intentionally by an operator or due to some software glitch. In either case, there is no documentation of the change to alert the analyst that the data stream is no longer *consistent* and *can not be interpreted* using the existing data definitions. Software and hardware components fail on occasion leading to gaps in the data streams (*missing or incomplete data*).

Data Summarization: Due to the huge size and rapid accumulation, data streams are usually summarized for storage -- for instance using 5-minute aggregates of number of packets, average CPU usage, and number of events of a certain type in the system logs. However, the average CPU usage might not reflect abnormal spikes. Or, a rare but catastrophic event might be unnoticed among all the other types of alarms. The trade-off between data granularity and aggregation is an important one. There has been much interest in representing data streams using histograms and other distributional summaries (Guha, Koudas & Shim, 2001) but largely for univariate data streams. Options for multivariate data streams and the use of sufficient statistics (Moore, 2006) for building regression type models for data streams are explored in Dasu, Koutsofios & Wright (2007).

Data Integration: Creating a comprehensive data set from multiple data sources always poses challenges. Sometimes there are no well defined join keys – only soft keys like names and addresses that can be represented in many different ways. For example, "J. Smith", "John Smith" and "John F. Smith" might be different variations of the same entity. Disambiguation is not easy. One data source might contain only a fraction of the entities contained in the other data sources, leading to gaps in the data matrix. Data streams pose additional complexities such as synchronization of multiple streams. There are two ways the temporal aspect could be a problem. First, if the clocks that timestamp the data streams are out of step and second, if the aggregation granularity does not allow the two data streams to be synchronized in any meaningful fashion.

Data quality is of particular concern in data streams. We do not have the luxury of referring back to the raw data to validate or correct mistakes. Furthermore, data quality mistakes could get compounded rapidly due to the high rate of accumulation, resulting in a significant divergence from reality and accuracy.

A Framework

A data stream is typically a sequential set of measurements in time. In most extant literature, a data stream is univariate i.e. measures just one attribute. It is represented in a reduced, more manageable form by maintaining statistical summaries for a given slice of the data stream, called a window, where every chunk of *N* points constitutes a window.

In Figure 1, the gray dots are data points that have been processed and summarized into aggregates *{A(t-1)}*, the black dots represent the data in the current time window that will be stored as aggregates *{A(t)}* and the white dots are data points yet to be processed. Another approach to maintaining summaries is to compute cumulative summaries based on the entire history of the data stream and updating these as the data comes in.

Aggregates are typically counts, sums and higher order moments like sums of squares; extreme values like minimum and maximum; and other percentiles. It is important to select these summaries carefully since there is no further opportunity to access the raw data to either update the existing summaries or compute additional aggregates. The amount of type and kind of statistics to be maintained can be customized depending on the application and may include:

1. Characterizing the distribution of a data stream by building histograms – see Guha, Koudas & Shim (2001) and Muthukrishnan (2003).
2. Detecting changes in the distribution and updating the distribution to reflect the changes – change detection has received much attention since it plays a critical role in network monitoring, security applications and maintaining robust and reliable aggregates for data stream modeling. We will discuss this in a little greater detail later on in the context of the application.
3. Comparing two or more data streams (Cormode, Datar, Indyk & Muthukrishnan (2002)) or the same data stream at different points in time (Dasu, Krishnan, Venkatasubramanian & Yi (2006)).
4. Detecting anomalies and outliers in a data stream.
5. Discovering patterns and sequences in data stream anomalies. See Aggarwal (2006).
6. Building predictive models.

Figure 1

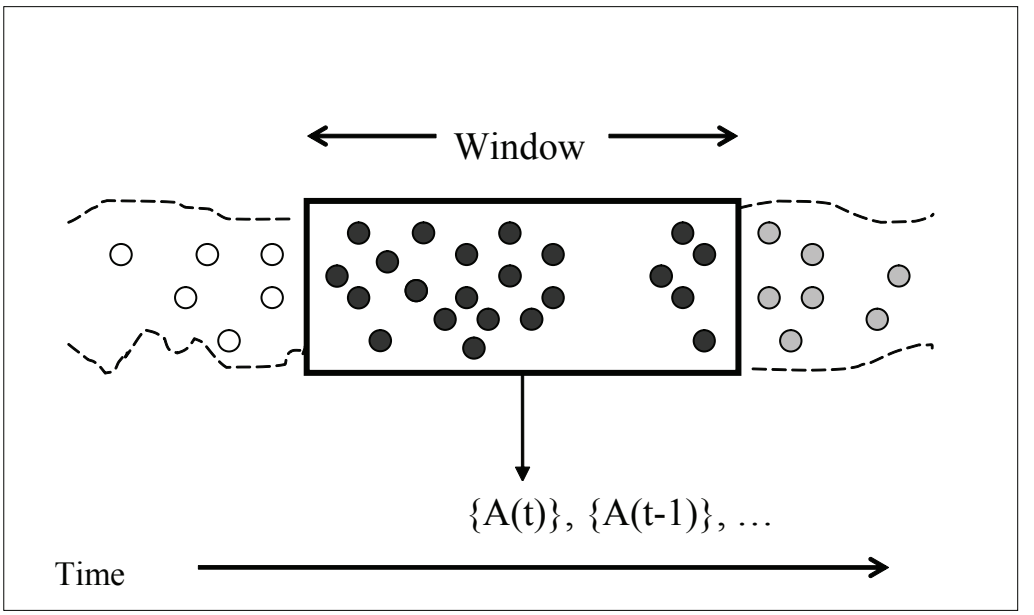
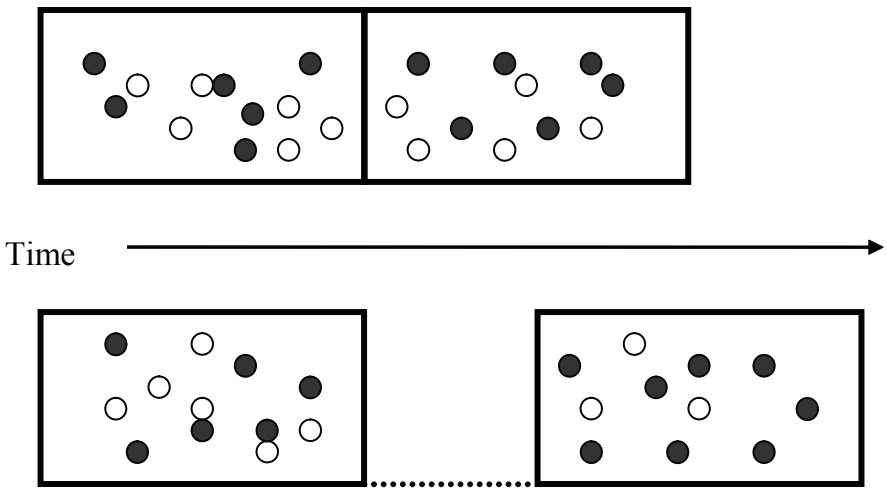


Figure 2.



An important example of the use of data stream summaries is change detection in data streams. Changes are detected by comparing data in two or more windows.

Short term changes are detected using adjacent windows that move in lock-step across time. Long term changes are detected using fixed-slide windows where one window is kept fixed while the second window moves forward in time. When a change is detected, the fixed window is moved to the most recent position and the process starts all over again. Ben-David, Gehrke & Kifer (2004) use a rank based method (Lehmann, 1974) to detect changes in the two windows. However, this can not be extended to higher dimensional data streams since there is no ordering of data in higher dimensions. Nor can it be used for categorical attributes. The method proposed by Dasu, Krishnan, Venkatasubramanian & Yi (2006) based on the Kullback-Leibler information theoretic distance measure addresses these shortcomings. We give a brief description of the basic methodology since our case study relies on this technique.

1. First, use any data partitioning scheme to "bin" the data in the two windows being compared. The partition can be predefined, based on the values of categorical attributes (e.g. gender) or intervals of continuous attributes (e.g., income), or a simple data-driven grid, based on the quantiles of individual attributes. A partition can also be induced by a model such as a clustering or classification algorithm. In our application, we use a DataSphere partition (Johnson & Dasu, 1998) that has the property that the number of bins increases linearly with the number of dimensions or attributes. It is characterized by *distance layers* that divide the data into groups of data points that are within a distance range of a given reference point such as the mean; and *directional pyramids* characterized by the direction of greatest deviation. A detailed discussion is beyond the scope of this chapter.
2. Next, we compute two histograms *H1* and *H2,* one for each of the two windows being compared. The histograms are represented by the frequency vectors

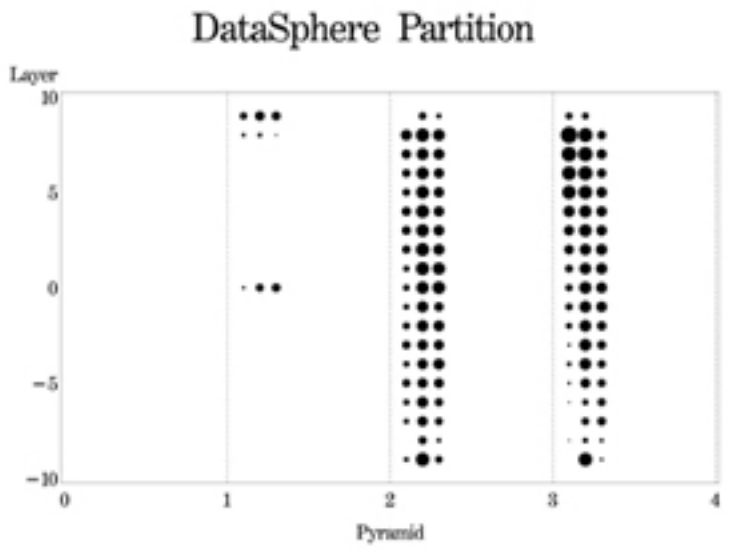
   (p1, p2, …, pB)

   and

   (q1, q2, …, qB),

   where *B* is the number of bins and *pi, qi* are the frequency counts.
3. We compare the distance between the two histograms using a range of statistical tests, like the naïve multinomial Chi-square or a similar test based on the Kullback-Leibler divergence. We use bootstrapping to simulate a sampling distribution when an exact test is not known.

Figure 3



4. Finally, we choose a desired level of confidence (e.g., 95%) and use the sampling distribution to see if the difference is significant.

The methodology is derived from the classical hypothesis testing framework of statistics. A brief discussion of statistical hypothesis testing along with the bootstrap methodology in the context of change detection is found in Dasu, Krishnan, Venkatasubramanian & Yi (2006). We use additional tests to identify regions of greatest difference. We present below an application that uses this approach.

An Application: Monitoring IP Networks

IP networks carry terabytes of data a day to enable the smooth functioning of almost every aspect of life, be it running corporations, industrial plants, newspapers, educational institutions or simpler residential tasks like exchanging e-mail or pictures. The networks are made of thousands of hardware and software components, and governed by rules called protocols that direct and regulate the data traffic. The traffic, its movement, and the functioning of the components are recorded in daunting detail in various forms. Traffic flows are recorded by netflows that specify the amount and type of data, its origin and destination, and intermediate stops if any. The topology of the network is like a dynamic road map for the data traffic and maintained in configuration tables. The functioning of the components like routers that direct the traffic is recorded in terms of resource usage. Alarms and unusual events are logged by software installed at critical points of the network. We brought together several such data sources to give us a timely and accurate picture of the state of the network.

We present below a brief analysis of a major network observed over a six week period. We are deliberately vague about the details to preserve proprietary information and some of the data distributions have been modified in a manner that does not affect the illustrative purpose of this application. For narrative convenience, we focus on three attributes of the data: the proportion of errors, the total traffic in bytes of type A, and the total traffic of type B. We use a specified week to create a baseline DataSphere partition based on the three attributes and compare the data from other weeks to the baseline. In Figure 3, we focus on a particular network device, D1. The *Y*-axis represents the distance layers, where a "negative" layer corresponds to values of deviations that are below average. The *X*-axis represents the directional pyramids. For example, a data point that has an above average proportion of errors and is

Figure 4.

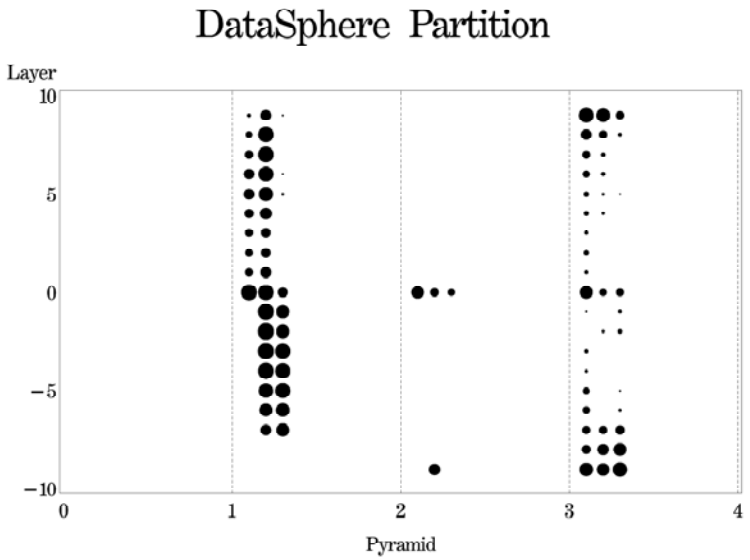


Figure 5.

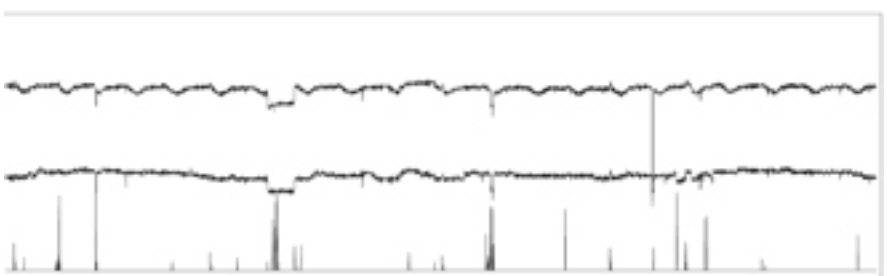
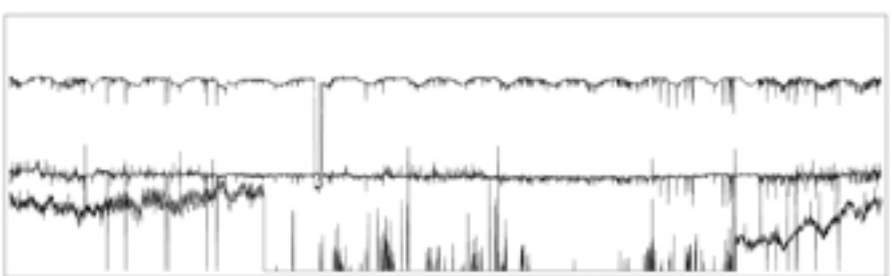


Figure 6.



most deviant with respect to that attribute will fall in pyramid 1. The more extreme the values, the greater the distance from the reference point and therefore higher the value of the distance layer. The dots are proportional to the amount of data in that bin. And the three columns within a specific pyramid range correspond to three consecutive weeks, where the middle column is the baseline week. In Figure 3, the distribution has shifted slightly in the three weeks, most notably in pyramid 3 (traffic type B) where the mass has shifted from higher values to more average values. Figure 4, describes the same information but for a different but comparable network device D2. Over the three weeks, we notice major shifts in the data mass in pyramid 1. The patterns for the error proportion are completely reversed from the first week to the third week. There is an unusually high proportion of errors which is fixed during weeks 2 and 3. This pattern is accompanied by an unusually large volume of type B traffic which returns to normal by the third week.

On examining the three individual attributes represented by the three lines in Figure 5, we see that for device D1, the individual attributes are well behaved with slight deviations from the ordinary patterns.

Figure 5 shows the same information for device D2. The erratic patterns in error proportions (bottom line in the plot) are evident, as well as the single big drop in type A traffic (top line in the plot) which corresponds to the big dot in pyramid 2, layer -8, week 2 in Figure 4.

The two dimensional "distance layer-directional pyramid" plots are a convenient and comprehensive way of displaying the distribution of the mass in the bins of the DataSphere partition, irrespective of the number of attributes. Note that line plots like the ones in Figures 5 and 6 become too numerous and overwhelming as the number of attributes increases.

In the case of the data streams above, the differences were clearly significant. In situations where the differences are more subtle, statistical tests of significance are used. See Dasu, Koutsofios & Wright (2007) for a case study that further illustrates the use of these tests.

FUTURE TRENDS

An interesting problem arises while comparing two data streams using multivariate histograms. Given the generally noisy nature of data streams, we can expect standard statistical tests to routinely declare differences. However, can we adapt the test to ignore differences in specified cells which we know a priori to be noisy and which might vary over time?

Research opportunities abound in the warehousing and querying of data streams. Aggarwal (2007) has a comprehensive collection of research articles that provide insights into the current research in the database community as well as open problems that require interdisciplinary solutions.

CONCLUSION

We have provided a brief overview of mining and monitoring data streams. Data streams are an inevitable and challenging form of information in many industrial and scientific applications, particularly the telecommunications industry. The research in this area is in its infancy and provides challenging research opportunities in managing, storing, querying and mining of data streams.

REFERENCES

Aggarwal, C. (2007). *Data Streams: Models and Algorithms*. Springer, USA.

Aggarwal, C., Han, J., Wang, J., & Yu, P. S. (2003). A Framework for Clustering Evolving Data Streams, *Proc. 2003 Int. Conf. on Very Large Data Bases*.

Babcock, B., Datar, M., Motwani, R., & L. O'Callaghan (2003). Maintaining Variance and k-Medians over Data Stream Windows. *Proceedings of the 22nd Symposium on Principles of Database Systems*.

Ben-David, S., Gehrke J., & Kifer, D. (2004). Detecting Change in Data Streams. *Proceedings of VLDB 2004.*

Bickel, P. J., & Doksum, K. A. (2001). *Mathematical statistics*, vol. 1. Prentice Hall, New Jersey.

Charikar, M., Chen K., & Farach-Colton, M. (2002). Finding Frequent Items in Data Streams. *International Colloquium on Automata, Languages, and Programming* (ICALP '02) 508--515.

Cormode, G., Datar, M, Indyk, P., & Muthukrishnan, S. (2002) Comparing data streams using Hamming

norms. In *Proceedings of the International Conference on Very Large Data Bases*, pp. 335-345.

Cortes, C., & Pregibon, D. (2001). Signature-based methods for data streams. *Data Mining and Knowledge Discovery*, *5*, 167-182.

Cover, T., & Thomas, J. (1991). *Elements of Information Theory*. John Wiley & Sons, New York.

Dasu, T., Koutsosfios, E., & Wright, J. R. (2007). A Case Study in Change Detection. In *Proc. of the International Workshop on Statistical Modelling, Barcelona, 2007.*

Dasu, T., Krishnan, S., Venkatasubramanian, S., & Yi, K. (2006). An information-theoretic approach to detecting changes in multi-dimensional data streams. *Proceedings of the 38th Symposium on the Interface of Statistics, Computing Science, and Applications (Interface '06)*, Pasadena, CA.

Domingos, P., & Hulten, G. (2001). Catching up with the data: Research issues in mining data streams. *Workshop on Research Issues in Data Mining and Knowledge Discovery, 2001.*

Efron, B., & Tibshirani, R. (1993). *An Introduction to the Bootstrap*. Chapman and Hall.

Gaber, M. M. (2006). *Mining data streams bibliography*. Retrieved from http://www.csse.monash.edu.au/~mgaber/WResources.htm

Gao, J., Fan, W., Han, J., & Yu, P. S. (2007). A general framework for mining concept-drifting data streams with skewed distributions. In *Proc. of the SIAM International Conference on Data Mining.*

Guha, S., Gunopulous D., & Koudas, N. (2003). Correlating synchronous and asynchronous data streams. In *Proc. of International Conference on Knowledge Discovery and Data Mining.*

Guha, S., Koudas, N., & Shim, K. (2001). Data-streams and histograms. In *Proc. ACM Symp. on Theory of Computing,* pp. 471-475.

Henzinger, M., Raghavan, P., & Rajagopalan, S. (1998). Computing on data streams. *Technical Note 1998-011,* Digital Systems Center, Palo Alto, CA.

Johnson, T., & Dasu, T. (1998). Comparing massive high-dimensional data sets. *Proc. 1998 KDD,* pp. 229-233..

Lehmann, E. (1974). *Nonparametric statistics: Statistical methods based on ranks*. Holden-Day.

Moore, A. (2006). *New cached-sufficient statistics algorithms for quickly answering statistical questions.* Keynote address at KDD 2006, Philadelphia.

Muthukrishnan, S. (2003). Data streams: Algorithms and Applications. *Proceedings of the fourteenth annual ACM-SIAM symposium on discrete algorithms.*

Scott, D. W. (1992). *Multivariate density estimation: Theory, practice and visualization.* John Wiley, New York.

Scott, D.W., & Sain, S.R. (2004). Multi-Dimensional Density Estimation. In C. R. Rao & E. J. Wegman (Eds.), *Handbook of Statistics: Data Mining and Computational Statistics*. Elsevier, Amsterdam.

Urbanek, S., & Dasu T. (2007). A statistical framework for mining data streams. *Tutorial presentation, SIAM International Conference on Data Mining.*

Wong, P. C., Foote, H., Adams, D. Cowley, W., & Thomas, J. (2003). Dynamic visualization of transient data streams. In *Proc. of INFOVIS, 2003.* pp. 97-104.

Zhu, Y. and Shasha, D. (2002) StatStream: Statistical monitoring of thousands of data streams in real time**.** In *VLDB 2002*, pages 358--369.

KEY TERMS

Bootstrap: A technique by which multiple samples are created from a single sample to compute error bounds for statistics computed from the original sample. Efron & Tibshirani (1993).

Chi-Square Test: A statistical test based on the Chi-square distribution to determine the statistical significance of a sample statistic.

Histogram: A histogram is a mapping that counts the number of observations that fall into various disjoint categories (known as bins).

Hypothesis Testing: A statistical framework for making decisions using relatively small samples of data. Bickel & Doksum (2001).

IDS: Intrusion detection system is software installed at critical points in a network to monitor the data packets that pass through for suspicious patterns.

Join Key (Match Key): An attribute or field in a database used to join or combine different database tables.

Kullback-Leibler Distance: A measure of the divergence between two probability distributions. Cover & Thomas (1991).

Partition: A method of dividing an attribute space into mutually exclusive classes that completely cover the space.

Quantiles: Values of a random variable that mark off certain probability cut-offs of the distribution. For example, the median is the 50% quantile of a distribution.

Sampling Distribution: The empirical distribution of a statistic computed from multiple samples drawn from the same populations under the same conditions.