An Information-Theoretic Approach to Detecting Changes in Multi-Dimensional Data Streams

Authors:

Dasu, T., Krishnan, S., Venkatasubramanian, S., & Yi, K. (2006).

Presentation:

Vincent Chu
22 November 2017

Content

Introduction

Motivation

Data streams can change over time as the underlying processes that generate them change.

Some changes are:

- Spurious and pertain to glitches in the data.
- Genuine, caused by changes in the underlying distributions.
- Gradual or more precipitous.

We would like to detect changes in a variety of settings:

- Data cleaning,
- Data modeling, and
- Alarm systems.

Motivation: Settings (1/2)

Data cleaning

Spurious changes affect the quality of the data.

- Missing values, default values erroneously set, discrepancy from an expected stochastic process, etc.

Data modeling

Shifts in underlying probability distributions can cause models to fail.

- While much effort is spent in building, validating and putting models in place, very little done is in terms of detecting changes.
- Sometimes, models might be too insensitive to change, reflecting the change only after a big shift in the distributions.

Motivation: Settings (2/2)

Alarm systems

Some changes are transient, and yet important to detect.

- Example: Network traffic monitoring
  - *Hard to posit realistic underlying models, yet some anomaly detection approach is needed to detect (in real time) shifts in network behavior along a wide array of dimensions.*

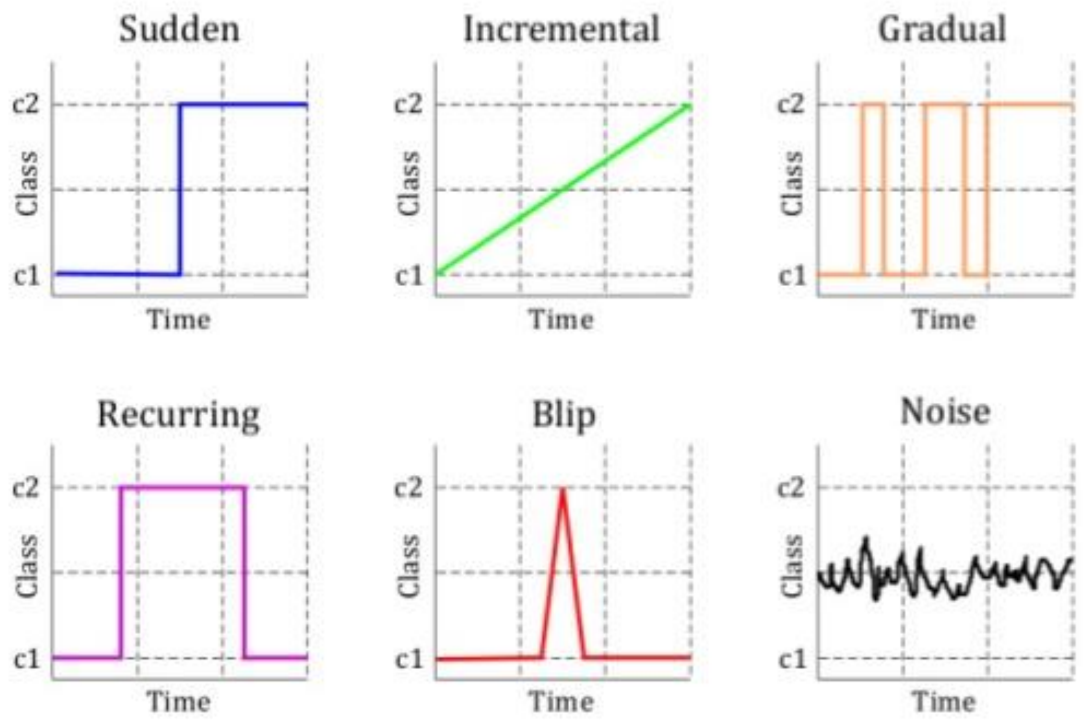


Image: D. Brzeziński thesis

Desiderata — Something that is needed or wanted.

Any change detection mechanism has to satisfy a number of criteria to be viable:

- Generality

  Applications for change detection come from a variety of sources, and the notion of "change" varies from setting to setting.

- Scalability

  Any approach must be scalable to very large datasets, and be able to adapt to streaming settings as well if necessary.

  Must be able to work with multidimensional data directly in order to capture spatial relationships and correlations.

- Statistical soundness:

  Key problems with a change detection mechanism is determining the significance of an event.

  Ensure that any changes reported by the method can be evaluated objectively

  Allowing the method to be used for a diverse set of applications.

Approach

A natural approach to detecting change in data is to model the data via a distribution.

- One can compare representative statistics like means or fit simple models like linear regression to capture variable interactions.
- Such approaches aim to capture some simple aspects of the joint distribution rather than the entire multivariate distribution.
  - *e.g. centrality, relationships between some specific attributes*

Approach: Parametric vs Nonparametric

Parametric approach

- Very powerful when data is known to come from specific distributions
- Wide variety of methods can be used to estimate distributions precisely.
- If distributional assumptions hold, require very little data in order to work successfully.
- However, generality is violated.
  - *Data that one typically encounters may not arise from any standard distribution, and thus parametric approaches are not applicable.*

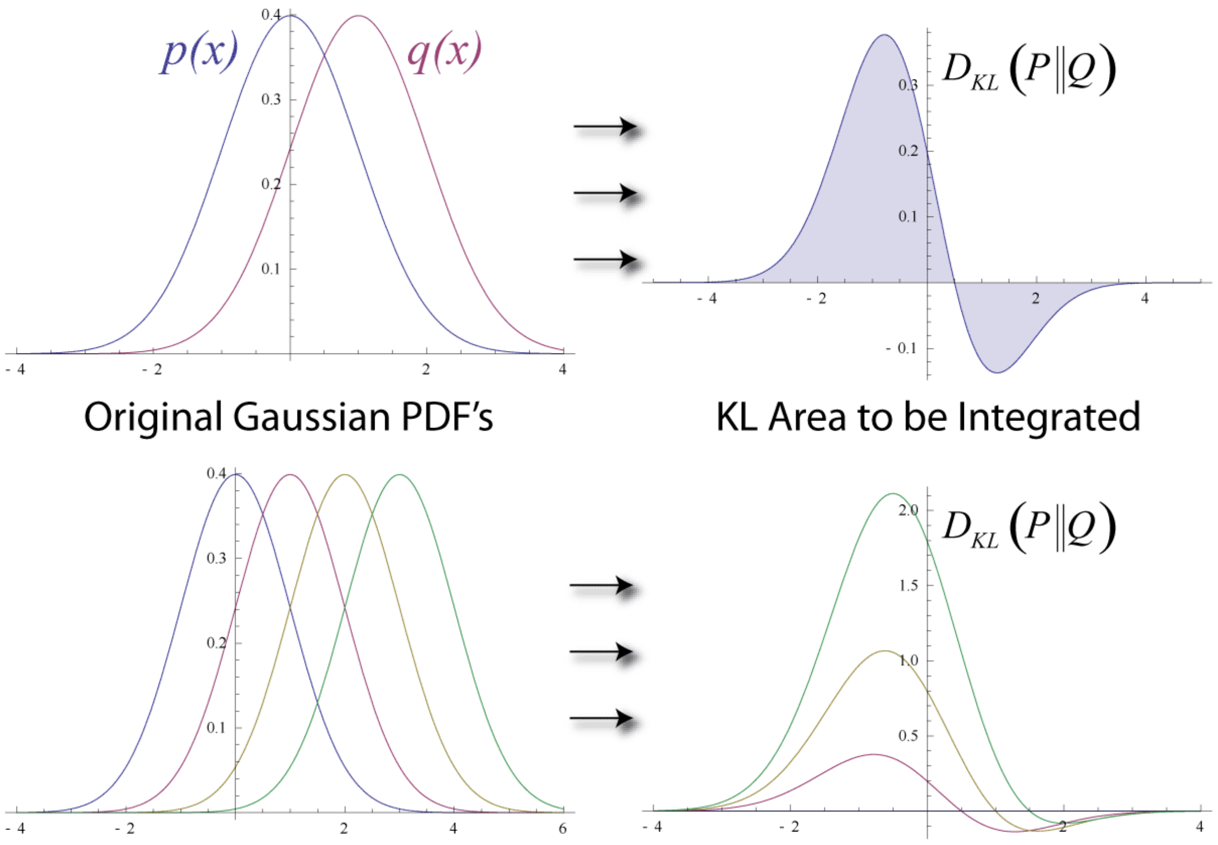
Nonparametric approach

- Make no distributional assumptions on the data.
- As before, computes a test statistic (a scalar function of the data), and compares the values computed to determine whether a change has occurred.

Approach: Information-theoretic (1/2)

Tests attempt to capture a notion of distance between two distributions.

A measure that is one of the most general ways of representing this distance is the relative entropy from information theory, also known as the Kullback-Leibler (or KL) distance.



Approach: Information-theoretic (2/2)

The KL-distance has many properties that make it ideal for estimating the distance between distributions:

- Given a set of data that we wish to fit to a distribution in a family of distributions, the maximum likelihood estimator is the one that minimizes the KL-distance to the true distribution.
- KL-distance generalizes standard tests of difference like: the t-test, chi-square and the Kulldorff spatial scan statistic.
- Optimal classifier that attempts to distinguish between two distributions p and q will have a false positive (or false negative) error proportional to an exponential in the KL-distance from p to q (the exponent is negative, so the error decreases as the distance increases).
- Example of an α-divergence

Approach: Statistical Significance

How do we determine whether the measure of change returned is significant or not?

- A statistical approach poses the question by specifying a null hypothesis (in this case, that change has not occurred), and then asking "How likely is it that the measurement could have been obtained under the null hypothesis?"
- The smaller this value "p-value", the more likely it is that the change is significant

Parametric tests: significance testing is fairly straightforward.

Some nonparametric tests: significance testing can be performed by exploiting certain special properties of the tests used.

But If we wish to determine statistical significance in more general settings, we need a more general approach to determining confidence intervals.

Approach: Bootstrap Method

Data-centric approach to determining confidence intervals for inferences on data.

By repeated sampling (with or without replacement) from the data, determines whether a specific measurement on the data is significant or not.

Can make strong inferences from small datasets

Satisfy the goal of generality & statistical soundness

Well suited for use with nonparametric methods

Scope

The paper presents a general information theoretic approach to the problem of multi-dimensional change detection. Specifically:

- Use of Kullback-Leibler distance as a measure of change in multi-dimensional data.
- Use of bootstrap methods to establish the statistical significance of distances computed.
- An efficient algorithm for change detection on streaming data that scales well with dimension.
- An approach for identifying sub-regions of the data that have the highest changes.
- Empirical demonstration (both on real and synthetic data) of the accuracy of approach.

Algorithm

Overview: Definitions

Let $x_1, x_2, \ldots$ be a stream of objects, over $x_i \in \mathbb{R}^d$.

A window $W_{i,n}$ denotes the sequence of points ending at x_i of size n:

$$W_{i,n} = (x_{i-n+1}, \ldots, x_i).$$

Distances are measured between distributions constructed from points in two windows W_t and $W_{t'}$.

Overview: Sliding Windows (1/2)

Using different-sized windows allows one to detect changes at different scales.

- Can run scheme with different window sizes in parallel.
  - *Each window size can be processed independently.*
- Will choose window sizes that increase exponentially
  - *Having sizes n, 2n, 4n, and so on.*

Note that we assume that the time a point arrives is its time stamp; we do not consider streams where data might arrive out of (time) order.

We consider two sliding window models:

1. *Adjacent windows model*
2. *Fix-slide windows model*

Overview: Sliding Windows (2/2)

Adjacent Windows Model

The two windows that we measure the difference between are W_t and W_{t-n}, where t is the current time.

- Better captures the notion of "rate of change" at the current moment
- Will repeatedly only detect small changes

Fix-slide Windows Model

We measure the difference between a fixed window W_n and a sliding window W_t.

- More suitable for change detection when gradual changes may cumulate over time

Overview

1. Constructed windows W_t and $W_{t'}$
2. Each window W_t defines an empirical distribution F_t.
3. Compute the distance

   $$d_t = d(F_t, F_{t'}) \text{ from } F_t \text{ to } F_{t'}$$

   where t' is either $\mathrm{t-n}$ or n depending on the sliding window model.

   This distance is our measure of the difference between the two distributions.
4. Determine whether this measurement is statistically significant

   Assert the null hypothesis: $H_0 : F_t = F_{t'}$ to determine the probability of observing the value d_t if H_0 is true.

To determine the probability of observing the value $\boldsymbol{d_t}$ if $\boldsymbol{H_0}$ is true, we use bootstrap estimates:

1. Generate a set of k bootstrap estimates:

   $$\widehat{d}_i, i = 1 \dots k.$$
2. Form an empirical distribution from which we construct a critical region (d_{hi}, ∞).
3. If d_t falls into this region, we consider that H_0 is invalidated.
4. Since we test H_0 at every time step, we only signal a change after we have seen γn distances larger than d_{hi} in a row

   where γ is a small constant defined by the user.

   True change should be more persistent than a false alarm. $\boldsymbol{\gamma}$ is the persistence factor.
5. If no change has been reported, we update the windows and repeat the procedure.

Overview

Algorithm 2.1 Change detection algorithm (for a fixed window size)

$t \leftarrow 2n$;
$t' \leftarrow n$;
Construct windows W_t and $W_{t'}$;
Compute $d_t = d(F_t, F_{t'})$;
Compute bootstrap estimate $\hat{d}_i, i = 1, \ldots, k$ and critical region (d_{hi}, ∞);
$c \leftarrow 0$;
while not at end of stream **do**
  if $d_t > d_{\text{hi}}$ **then**
    $c \leftarrow c + 1$;
    if $c \geq \gamma n$ **then**
      Signal change;
      Start over;
    end if
  else
    $c \leftarrow 0$;
  end if
  Slide window W_t (and $W_{t'}$ if required);
  Update d_t;
end while

Information-theoretic Distances

The measure we use to compare distributions is the **Kullback-Leibler distance** or the **relative entropy**.

KL-distance between two probability mass functions $p(x)$ and $q(x)$ is defined as:

$$D(p\|q) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)},$$

where the sum is taken (in the discrete setting) over the atoms of the space of events X.

However, the relative entropy is defined on a pair of probability mass functions.

How do we map sequences of points to distributions?

Theory of types

Information-theoretic Distances

Constructing a Distribution from a Stream (1/3)

Let $w = \{a_1, a_2, \dots, a_n\}$ be a multiset of letters from a finite alphabet $\mathcal{A}$.

The type P_w of w is thus vector representing the relative proportion of each element of $\mathcal{A}$ in w

$$P_{\mathbf{w}}(a) = \frac{N(a \mid \mathbf{w})}{n}.$$

Each set w defines a empirical probability distribution P_w.

For each set, we compute the corresponding empirical distribution, and compute the distance between the two distributions, viewed as mass functions.

Information-theoretic Distances

Constructing a Distribution from a Stream (2/3)

For d-dimensional data, the “alphabet” will consist of a letter for each leaf of the quad tree used to store the data, with the count being the number of points in that cell.

One advantage of the use of types is that categorical data can be processed in exactly the same way (with a letter associated with each value in the domain).

One problem with this approach is that the ratio p/q is undefined if $q = 0$. A simple correction replaces the estimate $P_w(a)$ by the estimate:

$$P_{\mathbf{w}}(a) = \frac{N(a|\mathbf{w}) + 0.5}{n + |\mathcal{A}|/2}.$$

Information-theoretic Distances

Constructing a Distribution from a Stream (3/3)

In summary:

Given:

Two windows W_1, W_2, *and*

Their associated multisets of letters $\boldsymbol{w}_1$, $\boldsymbol{w}_2$

Constructed from the alphabet defined over quad tree leaf cells

The KL-distance from W_1 to W_2 is:

$$D(W_1 || W_2) = \sum_{a \in \mathcal{A}} P_{\mathbf{w_1}}(a) \frac{P_{\mathbf{w_1}}(a)}{P_{\mathbf{w_2}}(a)}.$$

Bootstrap Methods + Hypothesis Testing

The bootstrap method is a method for determining the significance (or p-value) of a test statistic, eliminating bias and improving confidence intervals when doing statistical testing.

1. Given the empirical distributions $\hat{P}$ derived from the counts P
2. Sample k sets $S_1,\ldots, S_k$, each of size $2n$
3. Treat first n elements S_{i1} as coming from one distribution F
4. Treat remaining n elements $S_{i2} = S_i - S_{i1}$ as coming from other distribution G
5. Compute bootstrap estimates $\widehat{d_i} = D(S_i \parallel S_{i2})$.
6. Once the desired ASL α is fixed, choose the (1 − α)-percentile of these bootstrap estimates as d_{hi}; (d_{hi}, ∞) is the critical region.
7. If $\hat{d} > d_{hi}$, measurement is statistically significant and invalidates H_0.

Data Structures

Assume that the data points in the streams lie in a d-dimensional hypercube.

In order to maintain the KL-distance between two empirical distributions, we need a way of defining the “types”

- i.e.: a space partitioning scheme that subdivides the space into cells.

In principle any space partitioning scheme works in the framework

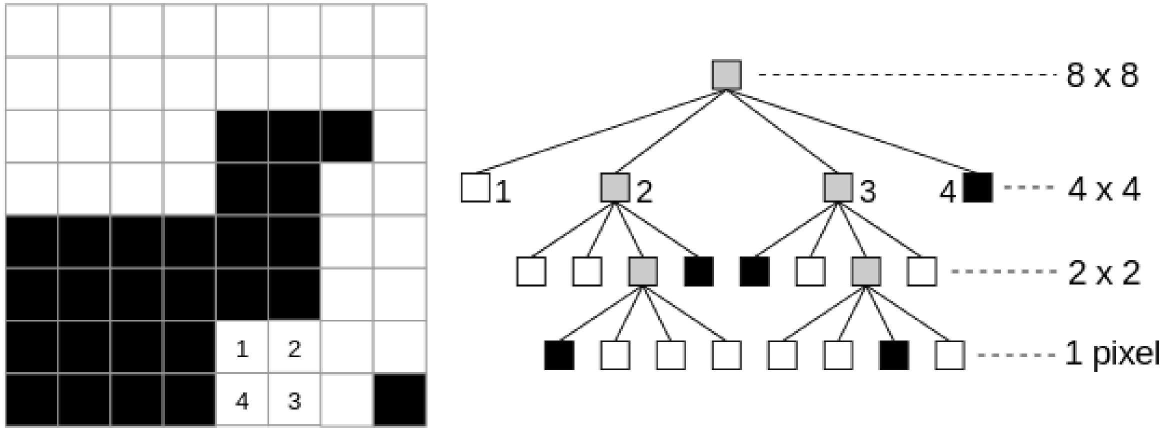
- e.g.: quad tree or k-d-tree

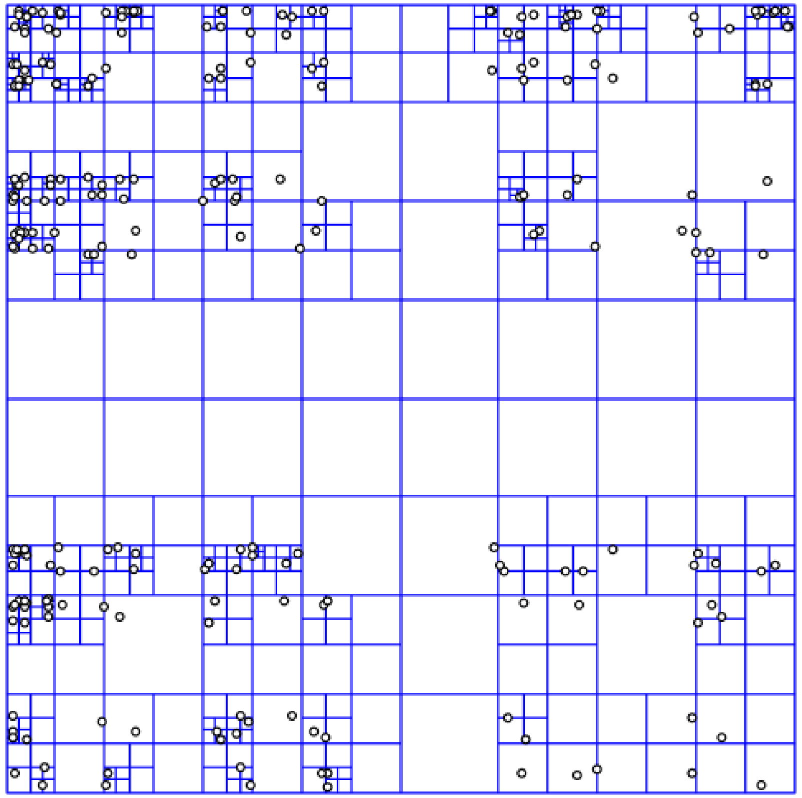
But would like to use a structure that:

- Scales well with the size and dimensionality of the data, and
- Produces “nicely shaped” cells at the same time.

Data Structures: Quad tree

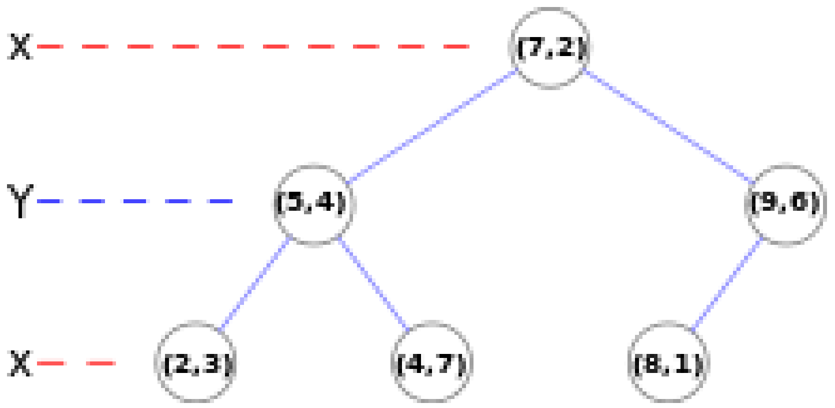
The square cells induced by a quad tree are intuitively good, but its 2^d fan-out might hurt its scalability in high dimensions.

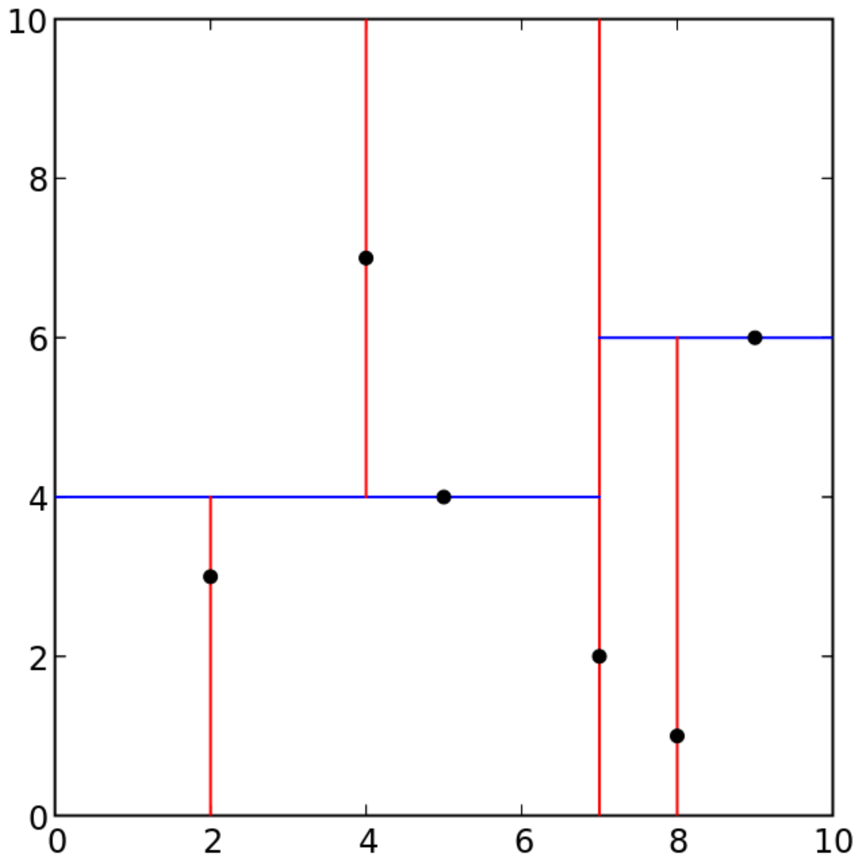




Data Structures: k-d tree

A k-d-tree scales well with dimensionality, but it might generates very skinny cells.





Data Structures: kdq tree (1/3)

A **kdq-tree** is a binary tree, each of whose nodes is associated with a box.

The box associated with the root v is the entire unit square

1. Divided into two halves by a vertical cut passing through its center.
2. The two smaller boxes are then associated with the two children of the root v_l, v_r.
3. Construct the trees rooted at v_l and v_r recursively, and
4. As we go down the tree, the cuts alternate between vertical and horizontal.
5. Stop the recursion if either:
    1. *The number of points in the box is below τ, or*
    2. *All the sides of the box have reached a minimum length δ*

τ and δ are user specified parameters

Data Structures: kdq tree (2/3)

For a kdq-tree built on n points in d dimensions:

1. Has at most $O(dn \cdot \log(1/\delta)/\tau)$ nodes
2. Height is at most $O(d \cdot \log(1/\delta))$
3. Can be constructed in time $O(dn \cdot \log(1/\delta))$
4. Aspect ratio of any cell is at most 2

Size scales linearly as the dimensionality and the size of data

Generates nicely shaped cells

Very cheap to maintain the counts associated with the nodes

- The cost is proportional to the height of tree.

Data Structures: kdq tree (3/3)

Build the kdq-tree on the first window W_1

- Use the cells induced by this tree as the types to form the empirical distributions for both W_1 and W_2 until a change has been detected, at which point we rebuild the structure. Use structure to compute the bootstrap estimates.

Data Structures: kdq tree

Maintaining the KL-distance (1/2)

Let P_v, Q_v be number of points from sets W_1, W_2 that are inside the cell associated with the leaf v of the kdq-tree.

We would like to maintain the KL-distance between $\mathrm{P} = \{P_v\}$ and $\mathrm{Q} = \{Q_v\}$:

$$
\begin{aligned}
D(P\|Q) &= \sum_v \frac{P_v + 1/2}{|W_1| + L/2} \log \frac{(P_v + 1/2)/(|W_1| + L/2)}{(Q_v + 1/2)/(|W_2| + L/2)} \\
&= \log \frac{|W_2| + L/2}{|W_1| + L/2} + \frac{\sum_v (P_v + 1/2) \log \frac{P_v + 1/2}{Q_v + 1/2}}{|W_1| + L/2},
\end{aligned}
$$

where L is the number of leaves in the kdq-tree.

Data Structures: kdq tree

Maintaining the KL-distance (2/2)

Since $|W_1|$, $|W_2|$ and L are readily known, we only need to maintain:

$$\tilde{D}(P\|Q) = \sum_v (P_v + 1/2) \log \frac{P_v + 1/2}{Q_v + 1/2}.$$

Since counts P_v, Q_v can be updated in $O(d \cdot \log(1/\delta))$ time per time step

KL-distance can also be maintained incrementally in the same time bound.

Data Structures: kdq tree

Identifying regions of greatest difference (1/2)

The kdq-tree structure for KL-distance based change detection can also be used to identify the most different regions between the two datasets, once a change has been reported.

- The idea is to maintain a special case of the KL-distance at each node (internal or leaf) v of the kdq-tree. This special case is the Kulldorff spatial scan statistic, which is defined at a node v as:

$$D_K(v) = P_v \log \frac{P_v}{Q_v} + (|W_1| - P_v) \log \frac{|W_1| - P_v}{|W_2| - Q_v} - |W_1| \log \frac{|W_1|}{|W_2|}.$$

Data Structures: kdq tree

Identifying regions of greatest difference (2/2)

Note that it is simply the KL-distance between W_1 and W_2 when there are only two bins: B_v and its complement $\overline{B_v}$. Kulldorff's statistic basically measures how the two datasets differ only with respect to the region associated with v.

Measures the log likelihood ratio of two hypotheses:

1. The region v has a different density from the rest of space, and
2. All regions have uniform density.

Note that this statistic can be easily maintained as it depends only on P_v and Q_v.

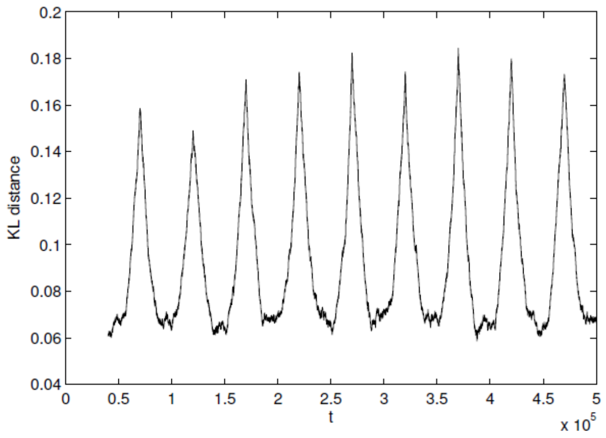
Experiments

Experiments

In all the experiments, we use the following default values for some of the parameters, unless specified otherwise.

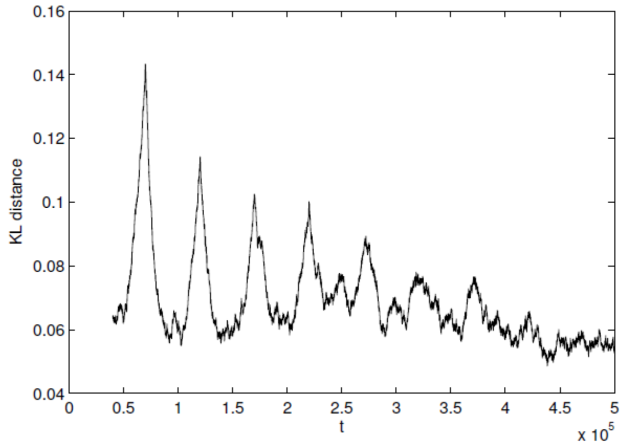
Parameter	Symbol	Value
Minimum side length of a cell	δ	2^{-10}
Maximum number of points in a cell	τ	100
Persistence factor	γ	5%
Achievable significance level (ASL)	α	1%
Number of bootstrap samples	k	500

Evaluation: Accuracy of KL-Distance (1/2)



Varying the mean μ

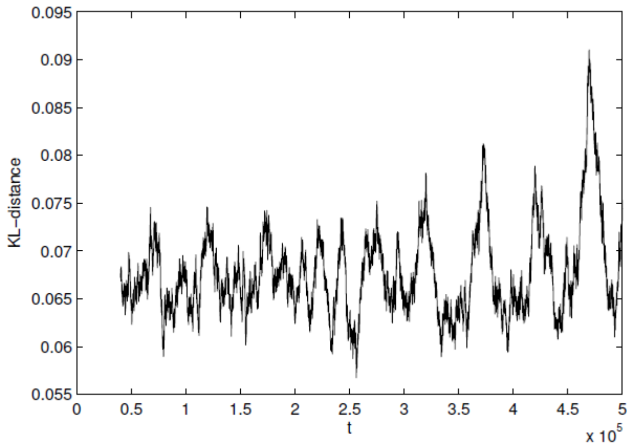
The KL distance between adjacent windows in a stream with varying $(\mu 1, \mu 2)$. Changes occur every 50,000 points.



Varying σ

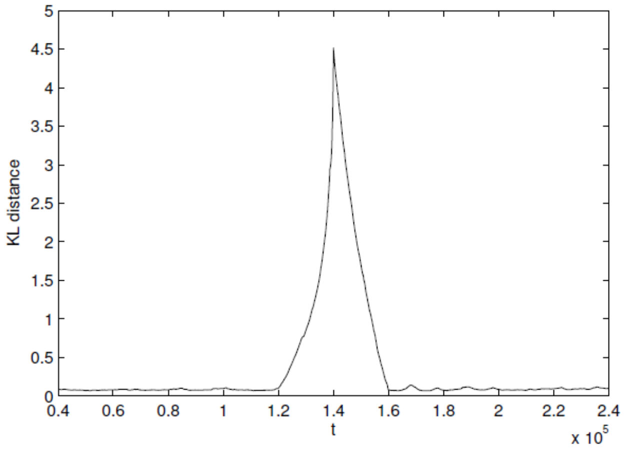
The KL distance between adjacent windows in a stream with varying $(\sigma 1, \sigma 2)$. Changes occur every 50,000 points.

Evaluation: Accuracy of KL-Distance (2/2)



Varying the correlation $\boldsymbol{\rho}$

The KL distance between adjacent windows in a the stream with varying ρ. Changes occur every 50,000 points.



An empirical case study

The KL distance between adjacent windows in a 3D data stream obtained from telephone usage in two urban centers. The change between urban centers occurs at $t = 120{,}000$.

Evaluation: Change Detection Method (1/4)

Stream	Detected	Late	False	Missed
$M(0.01)$	30	17	4	52
$M(0.02)$	70	20	4	9
$M(0.05)$	97	1	4	1
$D(0.01)$	36	20	1	43
$D(0.02)$	95	0	9	4
$D(0.05)$	92	4	7	3
$C(0.1)$	43	18	3	38
$C(0.15)$	83	10	4	6
$C(0.2)$	97	1	4	1

Varying Data Sources

Change detection results on different 2D normal data streams.

Stream	α	Detected	Late	False	Missed
$C(0.1)$	5%	63	15	11	21
	1%	43	18	3	38
	0.2%	36	13	0	51
$C(0.15)$	5%	88	8	21	3
	1%	83	10	4	6
	0.2%	76	12	1	11
$C(0.2)$	5%	96	1	26	2
	1%	97	1	4	1
	0.2%	98	1	3	0

Varying the ASL (Achievable Significance Level)

Change detection results on the streams with different ASLs.

Evaluation: Change Detection Method (2/4)

Stream	n	Detected	Late	False	Missed
$C(0.1)$	5000	30	25	5	44
	10000	43	18	3	38
	20000	62	7	0	30
$C(0.15)$	5000	68	14	17	17
	10000	83	10	4	6
	20000	91	1	1	7
$C(0.2)$	5000	93	5	15	1
	10000	97	1	4	1
	20000	99	0	0	0

Varying the window size

Change detection results on the streams with different window sizes.

Stream	k	Detected	Late	False	Missed
$C(0.1)$	100	51	15	2	33
	500	43	18	3	38
	2000	47	15	2	37
$C(0.15)$	100	85	8	9	6
	500	83	10	4	6
	2000	85	7	8	7
$C(0.2)$	100	97	1	10	1
	500	97	1	4	1
	2000	99	0	2	0

Varying number of bootstrap samples

Change detection results on the streams with different number of bootstrap samples.

Evaluation: Change Detection Method (3/4)

Δ	Detected	Late	False	Missed
0.05	60	10	1	29
0.1	67	17	1	15
0.2	98	1	5	0

d	Detected	Late	False	Missed
4	89	1	7	9
6	84	10	8	5
8	83	5	7	11
10	65	12	6	22

Poisson distributions

Change detection results on 2D Poisson data streams.

Higher dimensions

Change detection results on d-dimensional streams.

Evaluation: Change Detection Method (4/4)

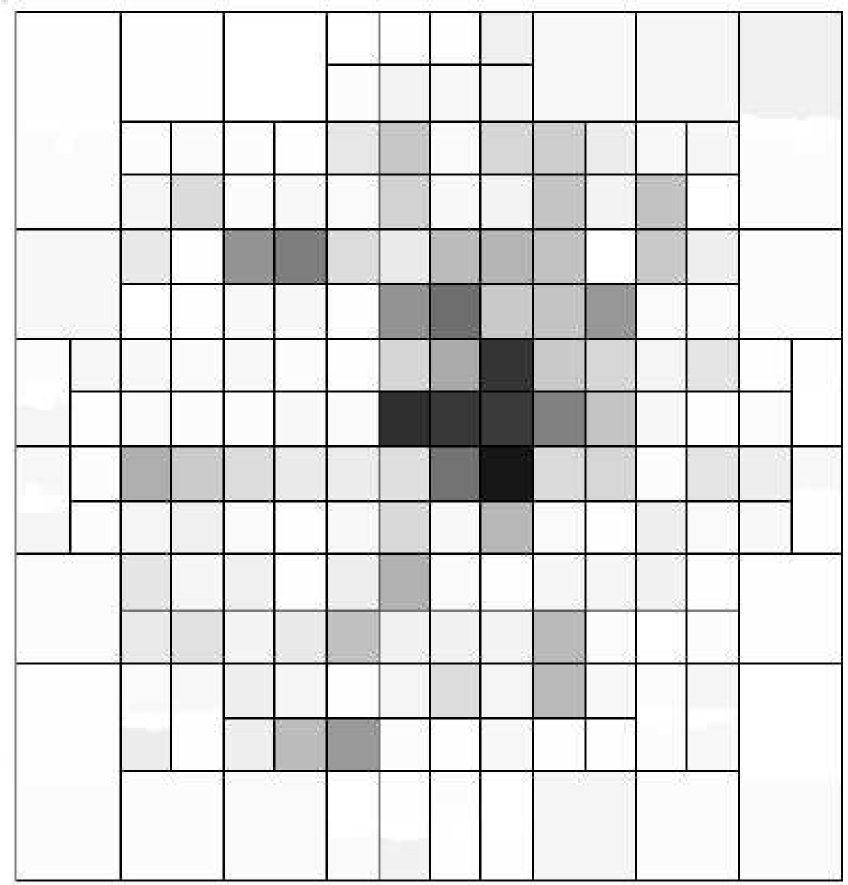
d	n	Construction (sec)	Update (msec)
4	10000	4.52	0.014
6	10000	5.33	0.022
8	10000	5.46	0.029
10	10000	6.08	0.035
10	20000	13.68	0.036
10	30000	22.09	0.035
10	40000	30.83	0.034

Efficiency

Running times with different n's and d's.

Evaluation: Identifying Regions of Greatest Discrepancy

Visualization of the Kulldorff statistic at depth 8 of the kdq-tree. The hole is located at (0.6, 0.6) and has radius 0.2.



Evaluation: Comparison with Prior Work in 1D

Stream	Scheme	Detected	Late	False	Missed
U	Wilcoxon	0	1	1	98
	KS	10	15	3	74
	ϕ	90	6	8	3
	Ξ	81	10	2	8
	KL	72	12	7	15
N_μ	Wilcoxon	90	8	7	1
	KS	89	9	8	1
	ϕ	74	18	4	7
	Ξ	86	13	9	0
	KL	70	23	9	6
N_σ	Wilcoxon	0	3	0	96
	KS	40	19	6	40
	ϕ	58	22	4	19
	Ξ	57	25	4	17
	KL	61	18	9	20

Conclusion

Conclusion

The paper presents a general scheme for nonparametric change detection in multidimensional data streams,

- Based on an information-theoretic approach to the data
- Intrinsically multidimensional
- Can even be used to incorporate categorical attributes in data

Experiments indicate that this approach is comparable to more constrained (but powerful) approaches in one dimension, and works efficiently and accurately in higher dimensions.

Thanks

Any Questions?