Reverse Nearest Neighbors in Unsupervised Distance-Based Outlier Detection*

* Article accepted in IEEE TKDE

Miloš Radovanović[1] Alexandros Nanopoulos[2]
Mirjana Ivanović[1]

[1]Department of Mathematics and Informatics
Faculty of Science, University of Novi Sad, Serbia

[2]Ingolstadt School of Management
University of Eichstaett-Ingolstadt, Germany

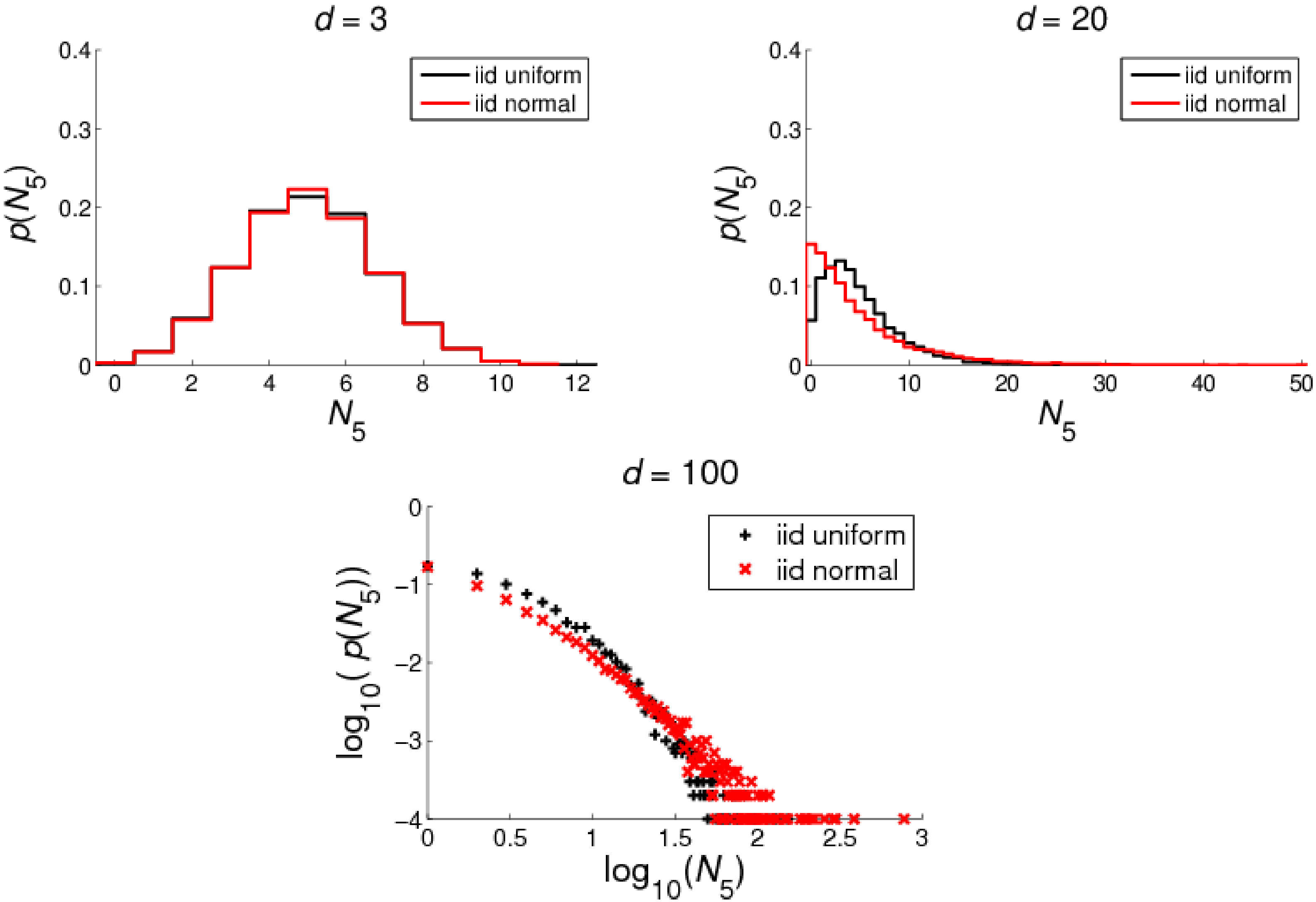
The Hubness Phenomenon

[Radovanović et al. ICML'09, Radovanović et al. JMLR'10]

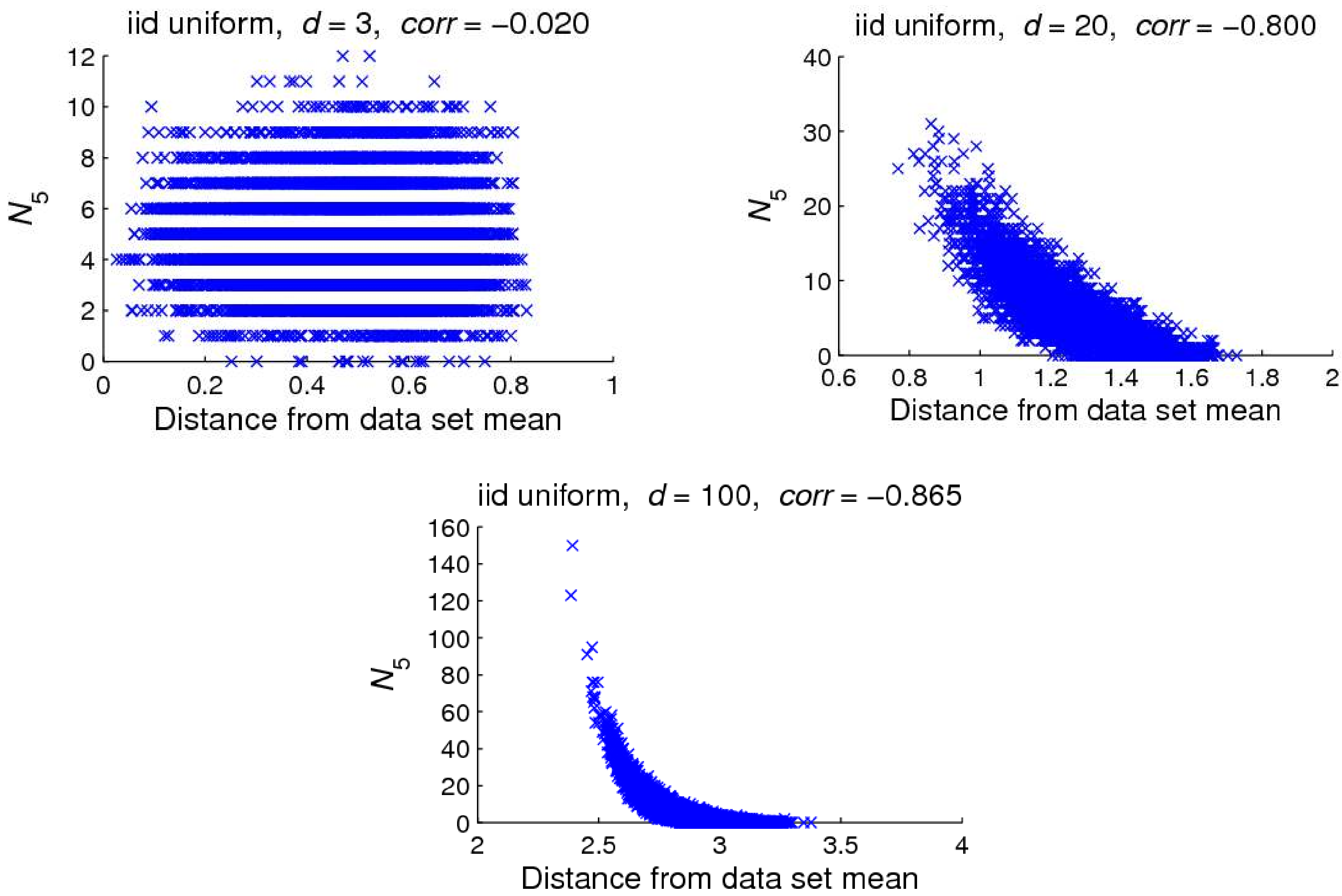
- $N_k(x)$, the number of ***k*-occurrences** of point $x \in \mathbf{R}^d$, is the number of times x occurs among k nearest neighbors of all other points in a data set. In other words:
  - $N_k(x)$ is the reverse k-nearest neighbor count of x
  - $N_k(x)$ is the in-degree of node x in the kNN digraph

- Observed that the distribution of N_k can become skewed, and have high variance, resulting in **hubs – points with high N_k values**, and **anti-hubs – points with low N_k**
  - Music retrieval [Aucouturier & Pachet PR'07]
  - Speaker verification ("Doddington zoo") [Doddington et al. ICSLP'98]
  - Fingerprint identification [Hicklin et al. NIST'05]

- Cause remained unknown, attributed to the specifics of data or algorithms



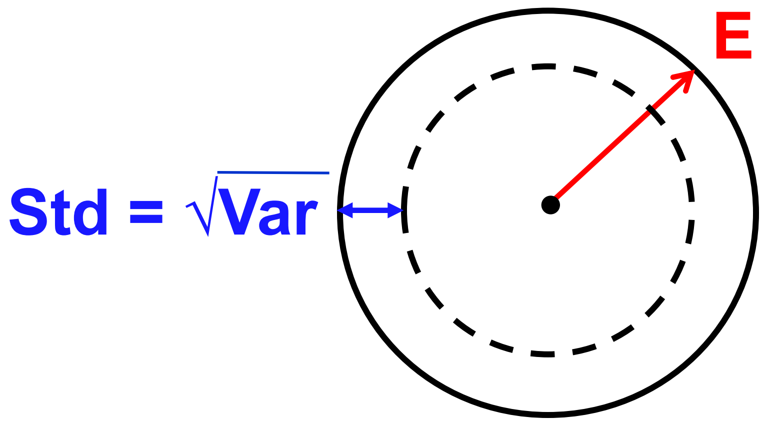
d = 3
0.4
0.3
0.2
0.1
0
p(N5)
iid uniform
iid normal
0
2
4
6
8
10
12
N5
d = 20
0.4
0.3
0.2
0.1
0
p(N5)
iid uniform
iid normal
0
10
20
30
40
50
N5
d = 100
0
−1
−2
−3
−4
log10(p(N5))
iid uniform
iid normal
0
0.5
1
1.5
2
2.5
3
log10(N5)



iid uniform, d = 3, corr = −0.020
N5
Distance from data set mean
iid uniform, d = 20, corr = −0.800
N5
Distance from data set mean
iid uniform, d = 100, corr = −0.865
N5
Distance from data set mean

Causes of Hubness

- **Related phenomenon: concentration of distance / similarity**
  - High-dimensional data points approximately lie on a **sphere** centered at any fixed point [Beyer et al. ICDT'99, Aggarwal & Yu SIGMOD'01]
  - The distribution of distances to a fixed point always has non-negligible variance [François et al. TKDE'07]
  - As the fixed point we observe the data set **center**



- **Centrality:** points closer to the data set center tend to be closer to all other points (regardless of dimensionality)

Centrality is amplified by high dimensionality

Important to Emphasize

- Generally speaking, **concentration does not CAUSE hubness**
- "Causation" might be possible to derive under certain assumptions. My preferred view: they are both manifestations of underlying mechanisms triggered by high dimensionality
- Example settings with(out) concentration and with(out) hubness:
  - C+, H+: iid uniform data, Euclidean dist.
  - C–, H+: iid uniform data, squared Euclidean dist.
  - C+, H–: iid normal data (centered at 0), cosine sim.
  - C–, H–: spatial Poisson process data, Euclidean dist.
- **Two "ingredients" needed for hubness:**
  1) **High dimensionality**
  2) **Centrality** (existence of **centers** / **borders**)

Hubness in Real Data

- Important factors for real data
  1) **Dependent attributes**
  2) **Grouping (clustering)**

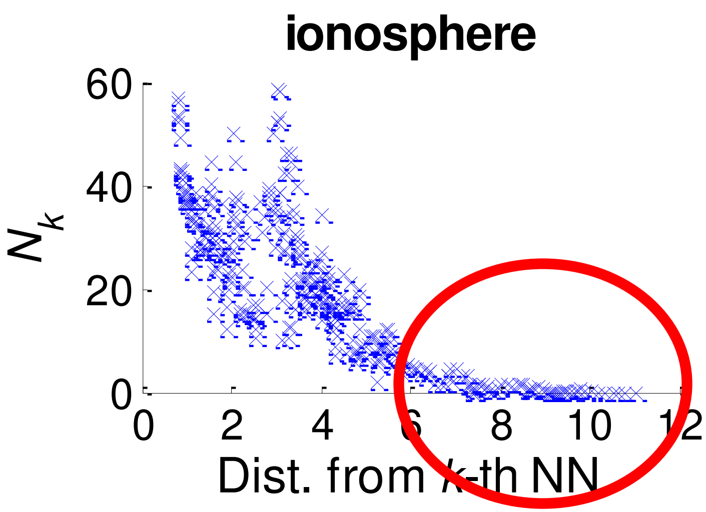
- 50 data sets
  - From well known repositories (UCI, Kent Ridge)
  - Euclidean and cosine, as appropriate

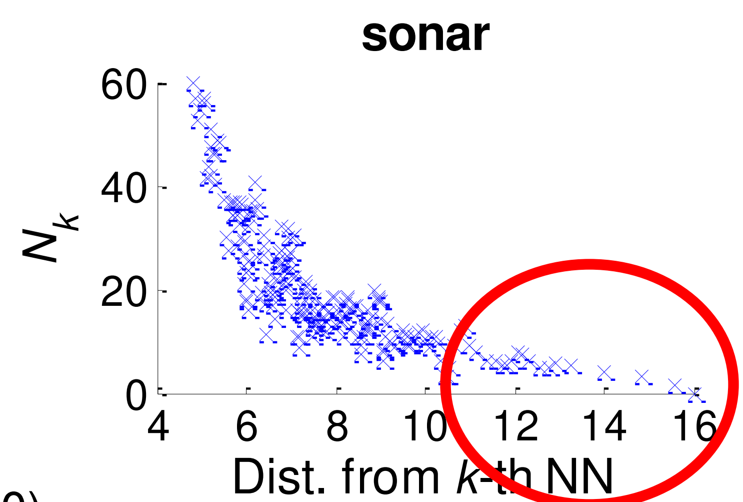
- **Conclusions** [Radovanović et al. JMLR'10]**:**
  1) Hubness depends on **intrinsic dimensionality**
  2) Hubs are in proximity of **cluster centers**

Anti-Hubs in Outlier Detection

[Radovanović et al. JMLR'10]

- In high dimensions, **points with low N_k** – the **anti-hubs** can be considered **distance-based outliers**
  - They are far away from other points in the data set / their cluster
  - High dimensionality contributes to their existence





(k = 20)

Anti-Hubs in Outlier Detection

[Aggarwal and Yu SIGMOD'01]

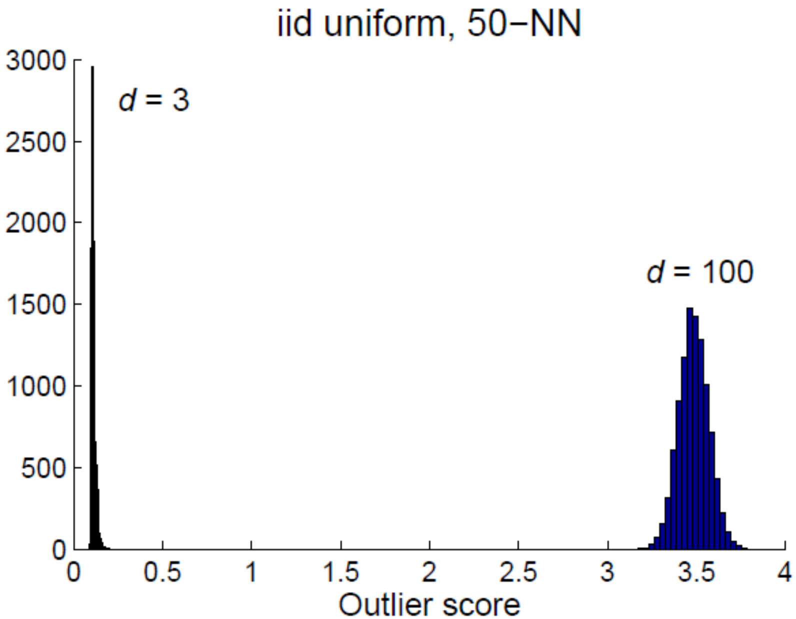
- In high-dimensional space unsupervised methods detect every point as an almost equally good outlier, since distances become indiscernible as dimensionality increases

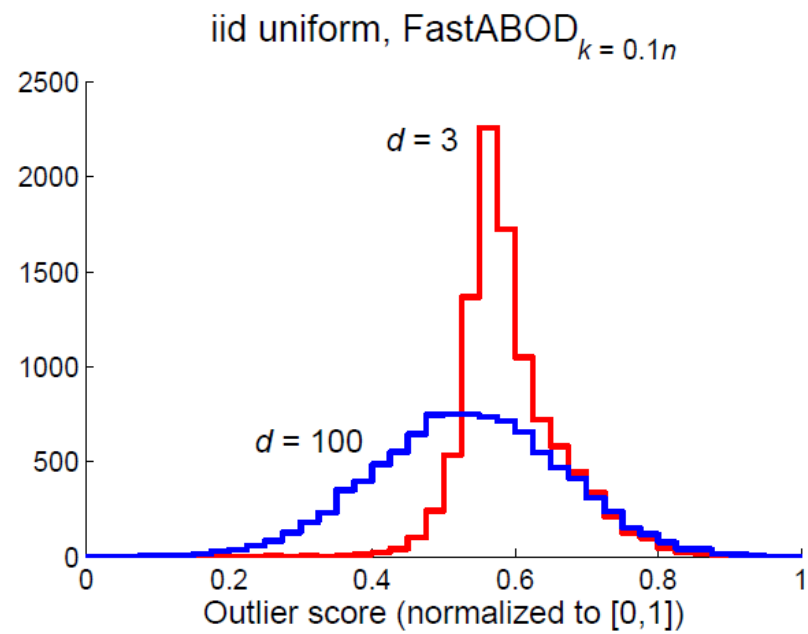
[Zimek et al. SADM'12]

- The above view was challenged by showing that the exact **opposite** may take place
- As dimensionality increases, outliers generated by a different mechanism from the data tend to be detected as more prominent by unsupervised methods
  - Assuming all dimensions carry useful information

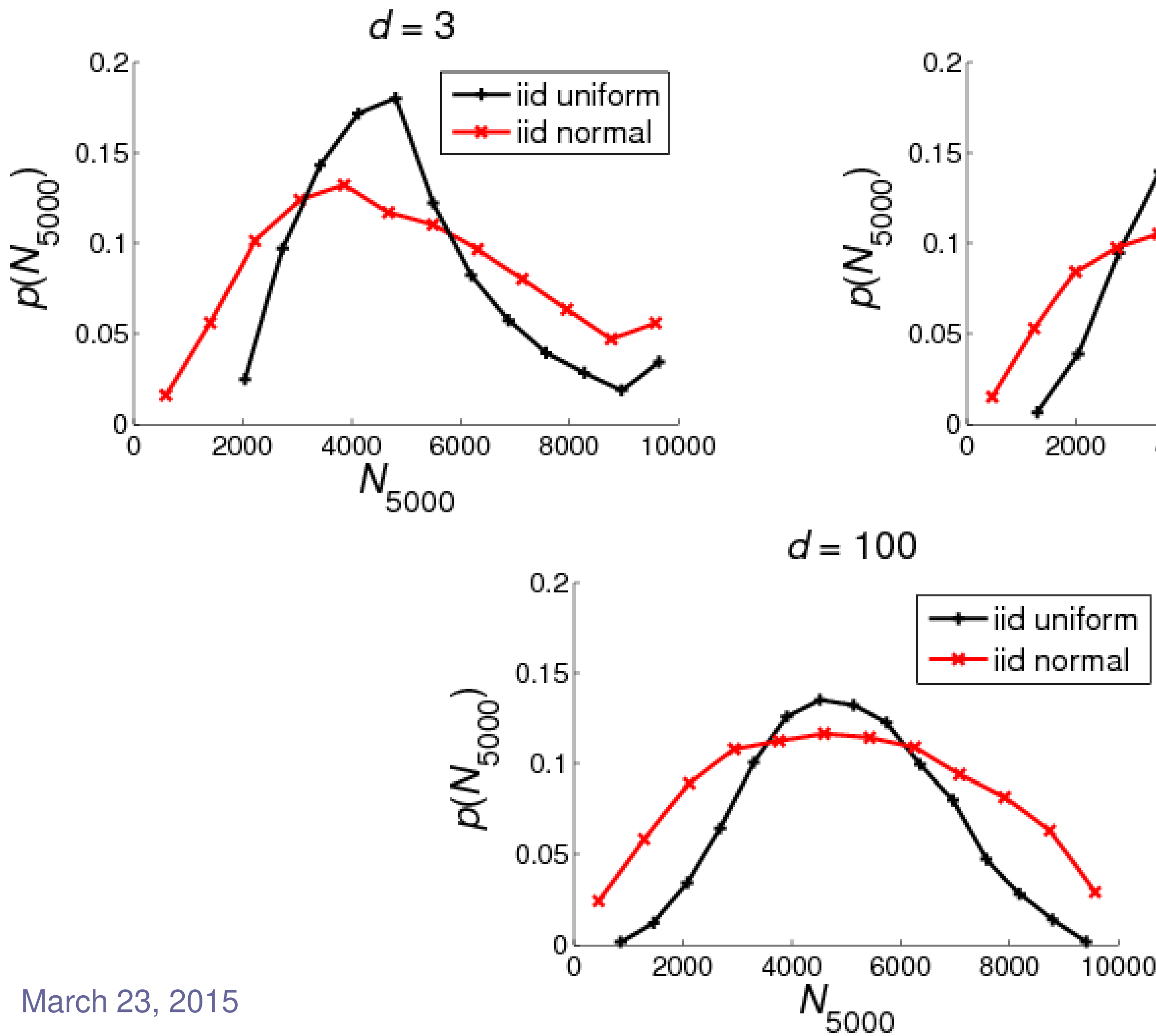
Anti-Hubs in Outlier Detection

- We show that the opposite can take place even when no true outliers exist, in the sense of originating from a different distribution
- This suggests that high dimensionality affects outlier scores and (anti-)hubness in similar ways

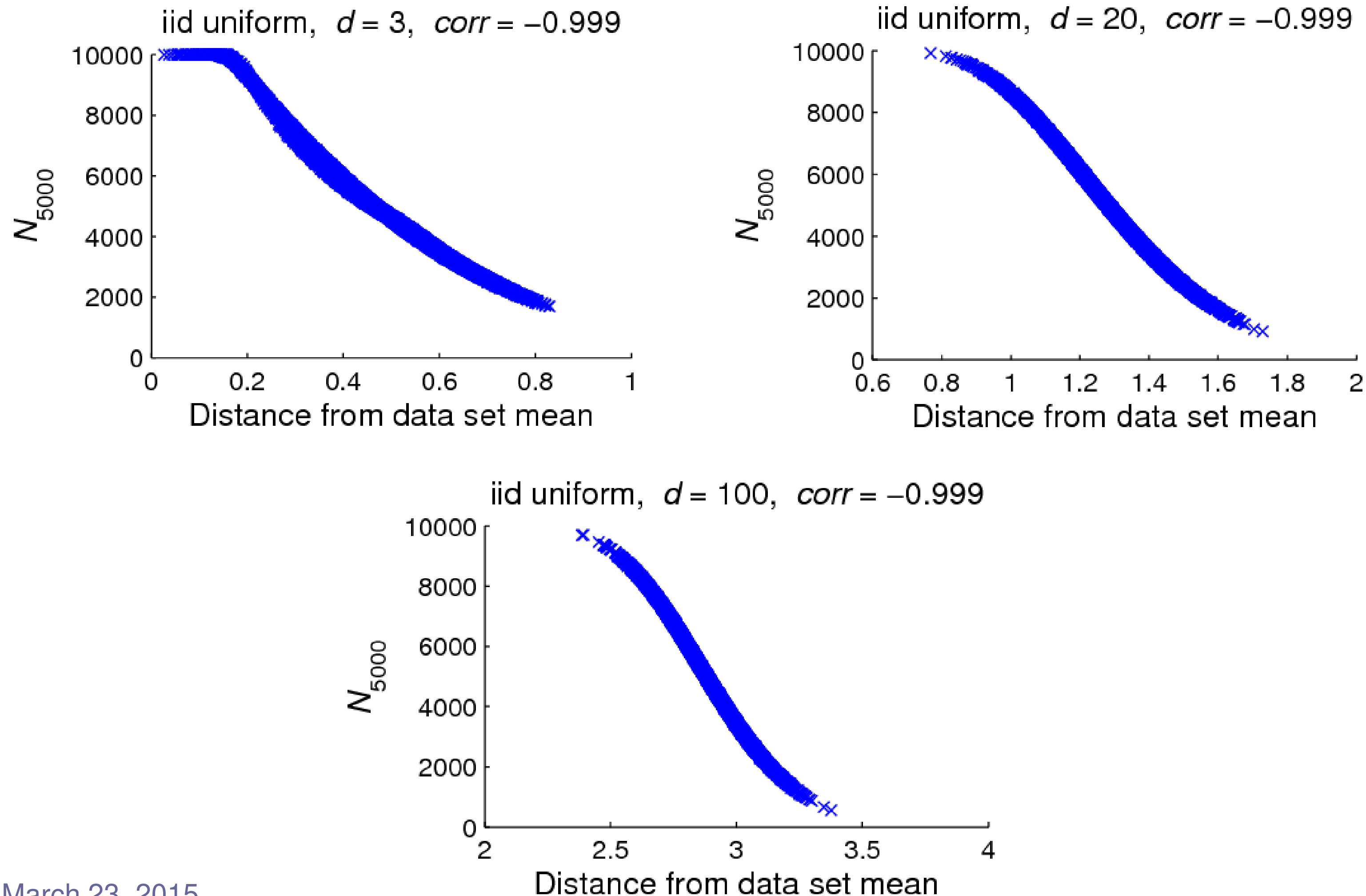




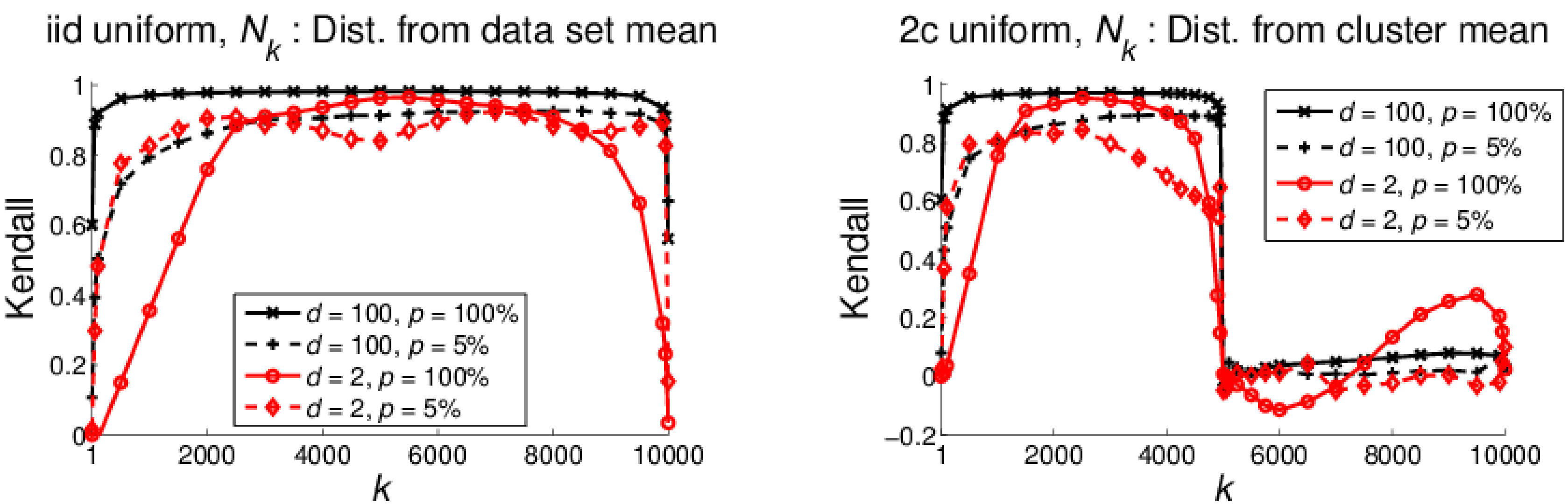
Hubness and Large Neighborhoods



Hubness and Large Neighborhoods



Hubness and Large Neighborhoods



- p = percentage of points with lowest N_k scores
- High dimensionality (d): N_k strong indicator of centrality overall (p = 100%), but weaker for anti-hubs (p = 5%)
- Low d: the opposite, especially w.r.t low k values
- Raising k strengthens correlation, but not when cluster boundary is crossed

The AntiHub Method

[Hautamäki et al. ICPR’04]

- Proposed method ODIN (Outlier Detection using Indegree Number), which selects as outliers points with N_k below or equal to a user-specified threshold
- Experiments on 5 data sets showed it can work better than various *k*NN distance methods
- Not aware of the hubness phenomenon, little insight into reasons why ODIN should work, its strengths, weaknesses…
- In method AntiHub, we use $N_k(x)$ as the outlier score of x (same as ODIN, without the threshold)

The AntiHub Method

Algorithm 1 AntiHub$_{dist}(D, k)$ (based on ODIN [11])

Input:

- Distance measure *dist*
- Ordered data set $D = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$, where $\mathbf{x}_i \in \mathbb{R}^d$, for $i \in \{1, 2, \ldots, n\}$
- No. of neighbors $k \in \{1, 2, \ldots\}$

Output:

- Vector $\mathbf{s} = (s_1, s_2, \ldots, s_n) \in \mathbb{R}^n$, where s_i is the outlier score of $\mathbf{x}_i$, for $i \in \{1, 2, \ldots, n\}$

Temporary variables:

- $t \in \mathbb{R}$

Steps:

1) For each $i \in (1, 2, \ldots, n)$
2) $t := N_k(\mathbf{x}_i)$ computed w.r.t. *dist* and data set $D \backslash \mathbf{x}_i$
3) $s_i := f(t)$, where $f : \mathbb{R} \to \mathbb{R}$ is a monotone function

The AntiHub Method

- We experimentally identified strengths and weaknesses of AntiHub with respect to different properties (factors):

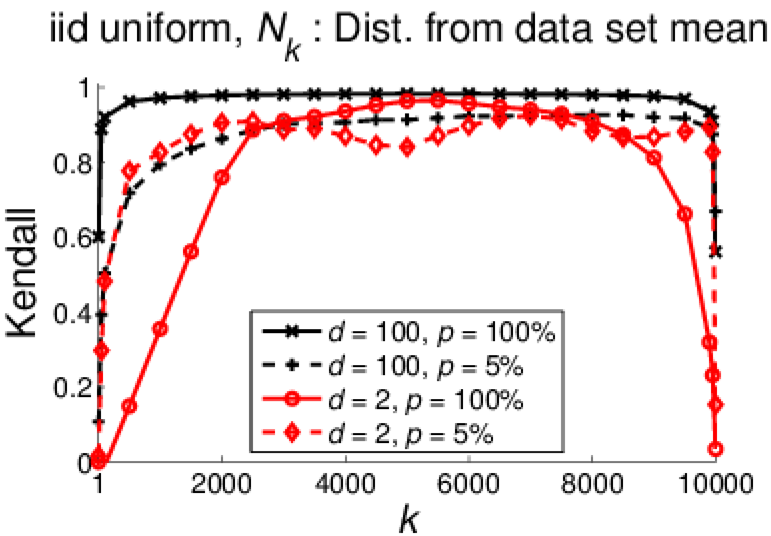
    1. Hubness
    2. Locality vs. globality
    3. Discreteness of scores
    4. Varying density
    5. Computational complexity

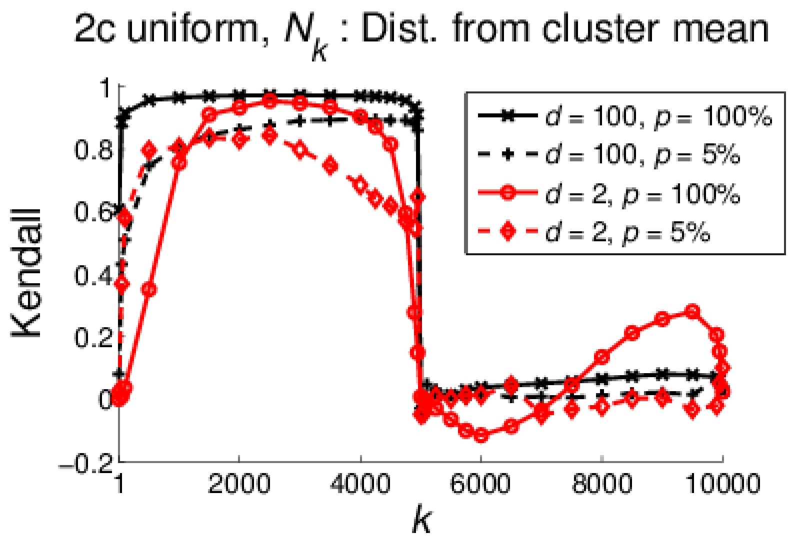
The AntiHub Method

Property 1: Hubness

- High (intrinsic) dimensionality, $k << n$:
  - Good overall correlation between N_k and distance to a center, but
  - Many N_k values of 0 – problem with discrimination
- Low dimensionality, $k << n$
  - Low correlation between N_k and distance to a center, but
  - For a small number of points with low N_k, this correlation is better, so AntiHub/ODIN can be meaningful

The AntiHub Method





The AntiHub Method

Property 2: Locality vs. globality

- For AntiHub and other methods based on *k*NN:
  - $k << n$: notion of outlierness is local
  - $k \sim n$: notion of outlierness is global
- AntiHub in “local mode” may have problems with discrimination
- Raising *k* can address this, but the notion of outlierness goes global
  - This can be problematic if we are interested in local outliers, but *k* crosses cluster boundaries

Property 3: Discreteness of scores

- Regardless of all of the above, N_k scores are integers, hence inherently discrete, which can also cause discrimination problems

The AntiHub Method

Property 4: Varying density

- AntiHub is not sensitive to the scale of distances in the data
- Can effectively detect (local) outliers in clusters of different densities without explicitly modeling density

Property 5: Computational complexity

- Using high k values can be useful
- However, approximate kNN search/indexing methods typically assume $k = O(1)$

The AntiHub^2 Method

- Notable weakness of AntiHub, discrimination of scores, contributed to by two factors:
  - Hubness
  - Discreteness of scores
- Therefore, we proposed method AntiHub^2, which combines the N_k score of a point with N_k scores of it's k nearest neighbors, in order to maximize discrimination
- AntiHub^2 improves discrimination of scores compared to the AntiHub method

The AntiHub^2 Method

Algorithm 2 AntiHub$^2_{dist}(D, k, p, step)$

Input:

- Distance measure $dist$
- Ordered data set $D = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$, where $\mathbf{x}_i \in \mathbb{R}^d$, for $i \in \{1, 2, \dots, n\}$
- No. of neighbors $k \in \{1, 2, \dots\}$
- Ratio of outliers to maximize discrimination $p \in (0, 1]$
- Search parameter $step \in (0, 1]$

Output:

- Vector $\mathbf{s} = (s_1, s_2, \dots, s_n) \in \mathbb{R}^n$, where s_i is the outlier score of $\mathbf{x}_i$, for $i \in \{1, 2, \dots, n\}$

Temporary variables:

- AntiHub scores $\mathbf{a} \in \mathbb{R}^n$
- Sums of nearest neighbors' AntiHub scores $\mathbf{ann} \in \mathbb{R}^n$
- Proportion $\alpha \in [0, 1]$
- (Current) discrimination score $cdisc, disc \in \mathbb{R}$
- (Current) raw outlier scores $\mathbf{ct}, \mathbf{t} \in \mathbb{R}^n$

The AntiHub^2 Method

Local functions:

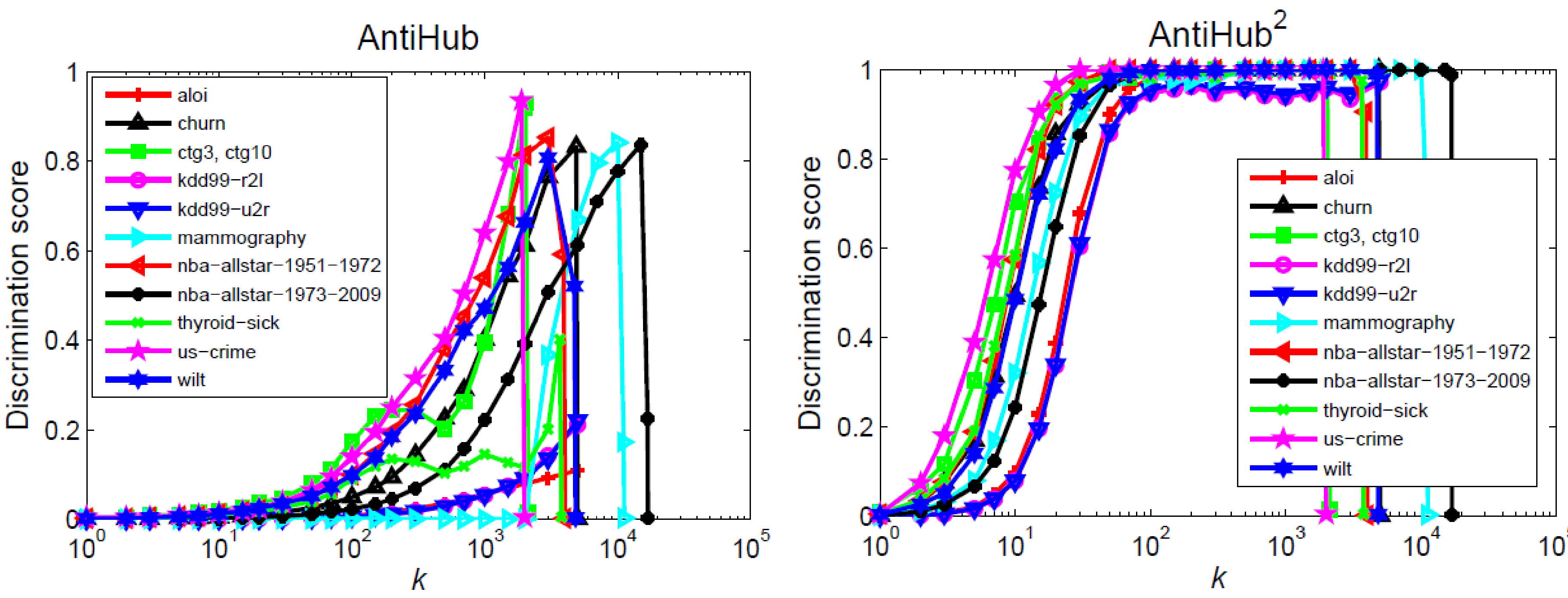
- discScore($\mathbf{y}, p$): for $\mathbf{y} \in \mathbb{R}^n$ and $p \in (0, 1]$ outputs the number of unique items among $\lceil np \rceil$ smallest members of $\mathbf{y}$, divided by $\lceil np \rceil$

Steps:

1) $\mathbf{a} :=$ AntiHub$_{dist}(D, k)$
2) For each $i \in (1, 2, \ldots, n)$
3) $\quad ann_i := \sum_{j \in \mathrm{NN}_{dist}(k,i)} a_j$, where $\mathrm{NN}_{dist}(k, i)$ is the set of indices of k nearest neighbors of $\mathbf{x}_i$
4) $disc := 0$
5) For each $\alpha \in (0, step, 2 \cdot step, \ldots, 1)$
5) $\quad$ For each $i \in (1, 2, \ldots, n)$
6) $\quad\quad ct_i := (1 - \alpha) \cdot a_i + \alpha \cdot ann_i$
7) $\quad cdisc :=$ discScore($\mathbf{ct}, p$)
8) $\quad$ If $cdisc > disc$
9) $\quad\quad \mathbf{t} := \mathbf{ct}$, $disc := cdisc$
10) For each $i \in (1, 2, \ldots, n)$
11) $\quad s_i := f(t_i)$, where $f : \mathbb{R} \to \mathbb{R}$ is a monotone function

Discrimination Improvement

discScore values for real data (p = 10%, $step$ = 0.01)



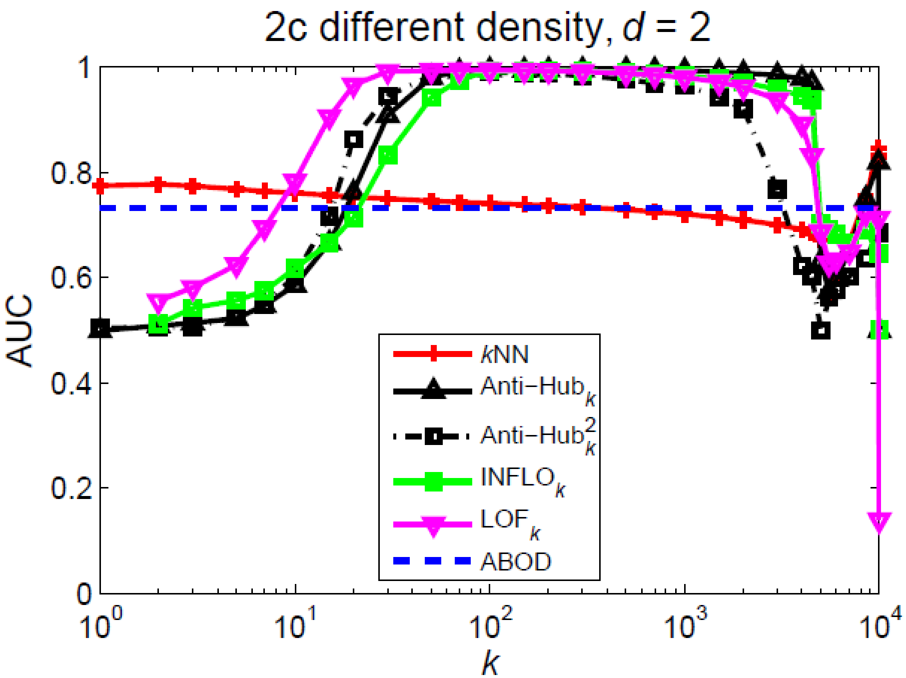
Performance Evaluation

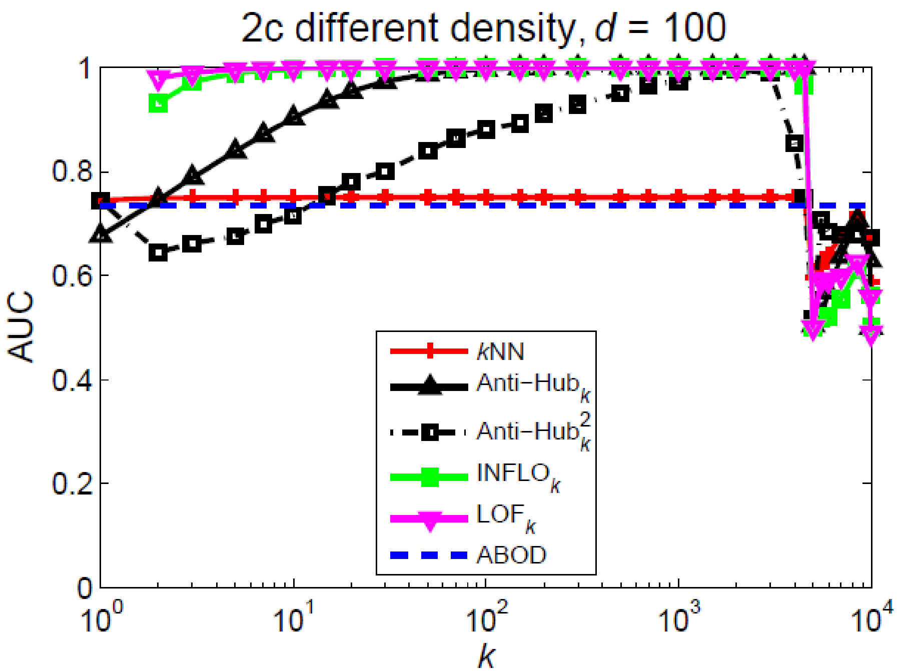
Methods for comparison:

- *k*NN: distance to the *k*th nearest neighbor
  [Ramaswamy et al. SIGMOD Rec'00]
- ABOD: Angle Based Outlier Detection
  [Kriegel et al. KDD'08]
- LOF: Local Outlier Factor
  [Breunig et al. SIGMOD Rec'00]
- INFLO: INFLuenced Outlierness
  [Jin et al. PAKDD'06]

Performance Evaluation

- Synthetic data: two well-separated Gaussian clusters of the same size, std of one 10 times larger than other, outliers 5% of points from each cluster projected 20% farther from respective cluster center

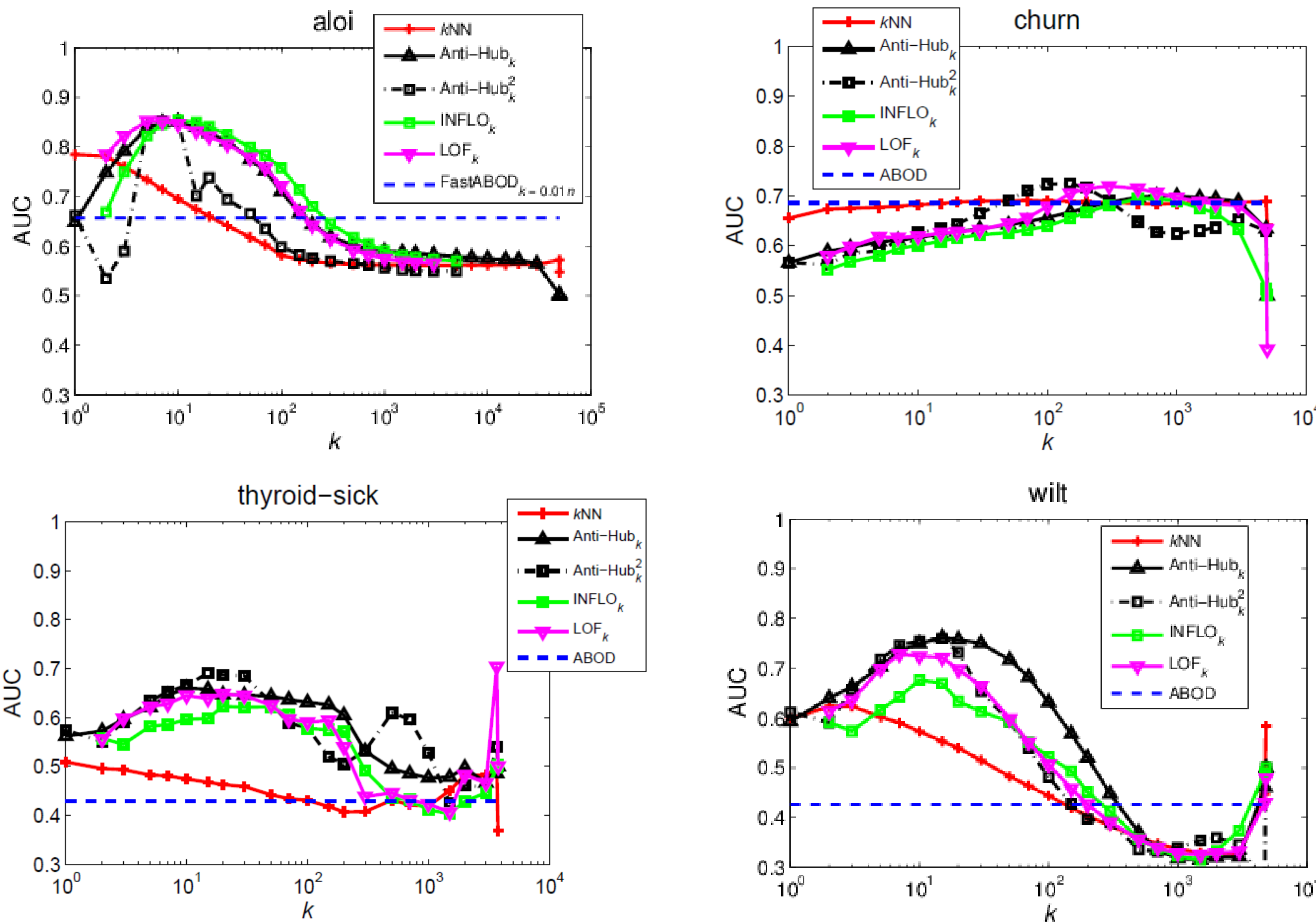


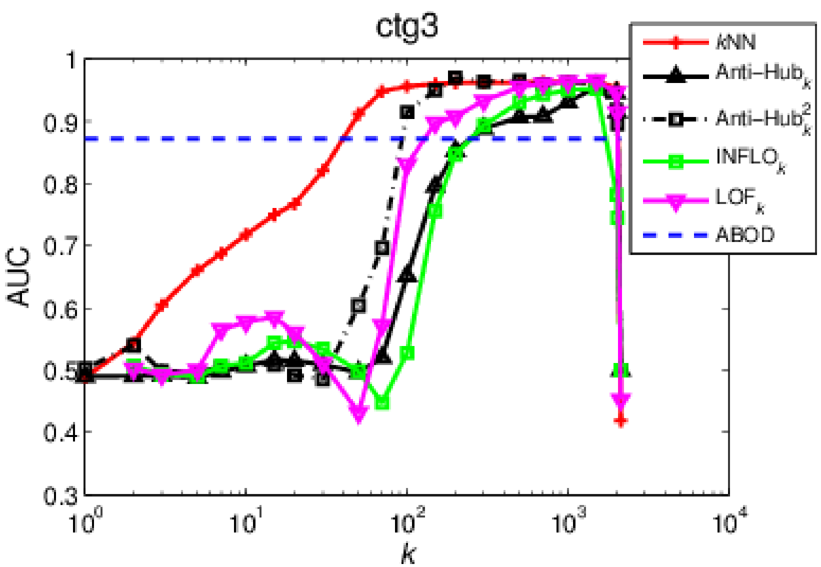


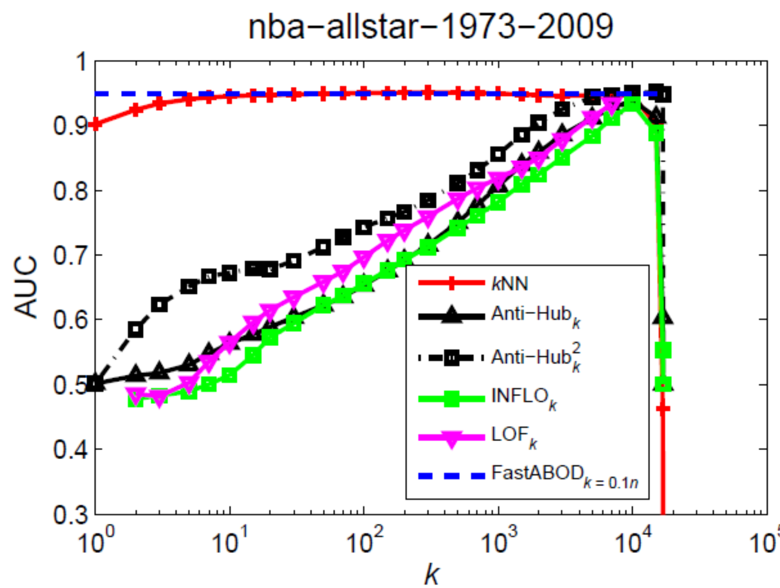
Performance Evaluation

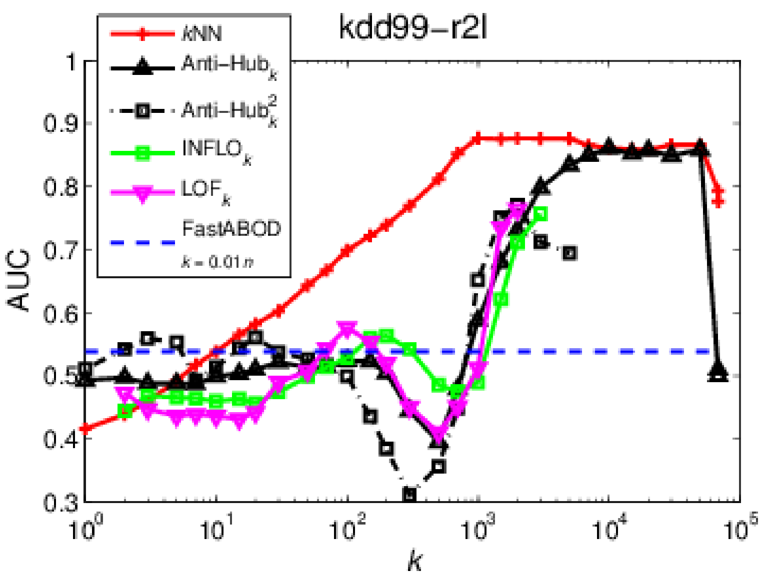
- Real data: mostly natural labeled outliers from various domains

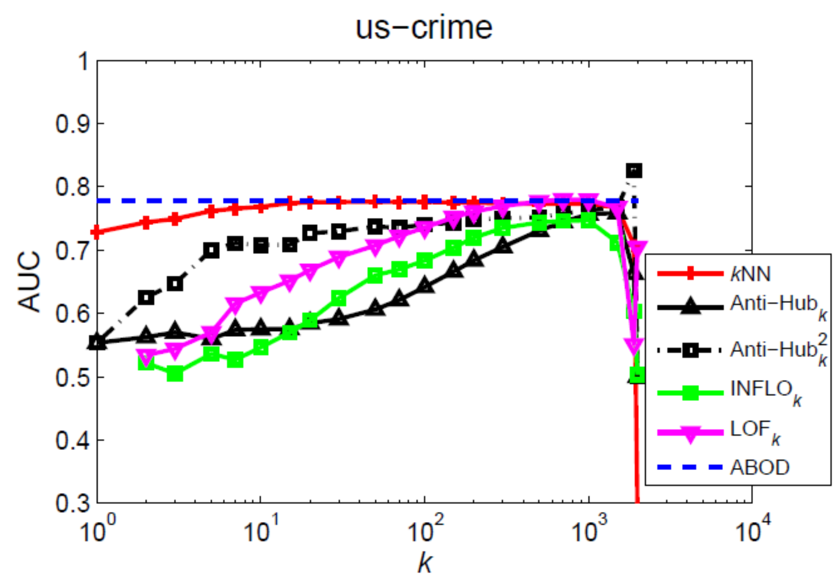
Name	n	d	$S_{N_{10}}$	Outlier%
aloi	50,000	64	0.260	3.016
churn	5,000	17	0.849	14.140
ctg3	2,126	35	0.652	8.279
ctg10	2,126	35	0.652	2.493
kdd99-r2l	68,338	38	0.018	1.456
kdd99-u2r	67,395	38	0.031	0.077
mammography	11,183	6	0.103	2.325
nba-allstar-1951-1972	4,018	15	0.483	15.903
nba-allstar-1973-2009	16,916	17	0.730	5.669
thyroid-sick	3,772	52	0.371	6.124
us-crime	1,994	100	1.327	7.523
wilt	4,839	5	-0.075	5.394


aloi
kNN
Anti–Hub$_k$
Anti–Hub$_k^2$
INFLO$_k$
LOF$_k$
FastABOD$_{k=0.01n}$
AUC
k
churn
ABOD
thyroid–sick
wilt


ctg3
AUC
k
kNN
Anti-Hub$_k$
Anti-Hub^2_k
INFLO$_k$
LOF$_k$
ABOD


nba-allstar-1973-2009
AUC
k
kNN
Anti-Hub$_k$
Anti-Hub^2_k
INFLO$_k$
LOF$_k$
FastABOD$_{k=0.1n}$


kdd99-r2l
AUC
k
kNN
Anti-Hub$_k$
Anti-Hub^2_k
INFLO$_k$
LOF$_k$
FastABOD $k = 0.01n$


us-crime
AUC
k
kNN
Anti-Hub$_k$
Anti-Hub^2_k
INFLO$_k$
LOF$_k$
ABOD

Performance Evaluation

- Two types of data sets: mostly local and mostly global outliers
- With respect to different *k* values, AUC of AntiHub and AntiHub^2 behaves similarly to density-based methods (LOF, INFLO)
- Very high *k* values can be useful for all methods, especially LOF, INFLO, AntiHub and AntiHub^2, suggesting there may be a relationship between "global" density-based and distance-based outliers
- AntiHub^2 can improve AUC of AntiHub, but not always, thus discrimination is not the only factor that should be addressed

Conclusions

- We provided a unifying view of the role of reverse nearest neighbor counts in unsupervised outlier detection:
  - Effects of high dimensionality on unsupervised outlier-detection methods and hubness
  - Extension of previous examinations of (anti-)hubness to large values of k
  - The article also explores the relationship between hubness and data sparsity
- We formulated the AntiHub method, discussed its properties, and improved it in AntiHub^2 by focusing on discrimination of scores
- Our main hope: clearing the picture of the interplay between types of outliers and properties of data, filling a gap in understanding which may have so far hindered the widespread use of reverse neighbor methods in unsupervised outlier detection

Future Possibilities

- High values of *k* can be useful, but:
  - Cluster boundaries can be crossed, producing meaningless results of local outlier detection. How to determine optimal neighborhood size(s)?
  - Computational complexity is raised; approximate NN search/indexing methods do not work any more. Is it possible to solve this for large *k*?
- AntiHub and AntiHub^2 are no "rock star" methods
  - Can N_k scores be applied to outlier detection in a better way? Through outlier ensembles?
- Extend to (semi-)supervised outlier detection methods

Future Possibilities

- Explore relationships between intrinsic dimensionality, distance concentration, (anti-)hubness, and their impact on subspace methods for outlier detection

- Investigate secondary measures of distance/similarity, such as shared-neighbor distances

References

M. Radovanović et al. Reverse nearest neighbors in unsupervised distance-based outlier detection. IEEE Transactions on Knowledge and Data Engineering, 2015 (forthcoming).

M. Radovanović et al. Nearest neighbors in high-dimensional data: The emergence and influence of hubs. In Proc. 26th Int. Conf. on Machine Learning (ICML), pages 865–872, 2009.

M. Radovanović et al. Hubs in space: Popular nearest neighbors in high-dimensional data. Journal of Machine Learning Research,11:2487–2531, 2010.

J.-J. Aucouturier and F. Pachet. A scale-free distribution of false positives for a large class of audio similarity measures. Pattern Recognition, 41(1):272–284, 2007.

G. Doddington et al. SHEEP, GOATS, LAMBS and WOLVES: A statistical analysis of speaker performance in the NIST 1998 speaker recognition evaluation. In Proc. 5th Int. Conf. on Spoken Language Processing (ICSLP), 1998. Paper 0608.

A. Hicklin et al. The myth of goats: How many people have fingerprints that are hard to match? Internal Report 7271, National Institute of Standards and Technology (NIST), USA, 2005.

K. S. Beyer et al. When is “nearest neighbor” meaningful? In Proc. 7th Int. Conf. on Database Theory (ICDT), pages 217–235, 1999.

C. C. Aggarwal and P. S. Yu. Outlier detection for high dimensional data. In Proc. 27th ACM SIGMOD Int. Conf. on Management of Data, pages 37–46, 2001.

D. François et al. The concentration of fractional distances. IEEE Transactions on Knowledge and Data Engineering, 19(7):873–886, 2007.

A. Zimek et al. A survey on unsupervised outlier detection in high-dimensional numerical data. Statistical Analysis and Data Mining, 5(5):363–387, 2012.

V. Hautamäki et al. Outlier detection using k-nearest neighbour graph. In Proc. 17th Int. Conf. on Pattern Recognition (ICPR), vol. 3, pages 430–433, 2004.

S. Ramaswamy et al. Efficient algorithms for mining outliers from large data sets. SIGMOD Record, 29(2):427–438, 2000.

H.-P. Kriegel et al. Angle-based outlier detection in high-dimensional data. In Proc. 14th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining (KDD), pages 444–452, 2008.

M. M. Breunig et al. LOF: Identifying density-based local outliers. SIGMOD Record, 29(2):93–104, 2000.

W. Jin et al. Ranking outliers using symmetric neighborhood relationship. In Proc. 10th Pacific-Asia Conf. on Advances in Knowledge Discovery and Data Mining (PAKDD), pages 577–593, 2006.