Hubs in Nearest-Neighbor Graphs: Origins, Applications and Challenges

Miloš Radovanović

Department of Mathematics and Informatics
Faculty of Sciences, University of Novi Sad, Serbia

Thanks

My host

- **Laurent Amsaleg**, Inria, Rennes, France

My host in July, prompting the second version of this talk

- **Michael Houle**, National Institute of Informatics, Tokyo, Japan

My host in March, prompting the first version of this talk

- **Kenji Fukumizu**, Institute of Statistical Mathematics, Tokyo, Japan

My coauthors

- **Mirjana Ivanović**, Department of Mathematics and Informatics, Novi Sad, Serbia
- **Alexandros Nanopoulos**, Ingolstadt School of Management, Germany
- **Nenad Tomašev**, ex Jožef Stevan Institute, Ljubljana, Slovenia; Google

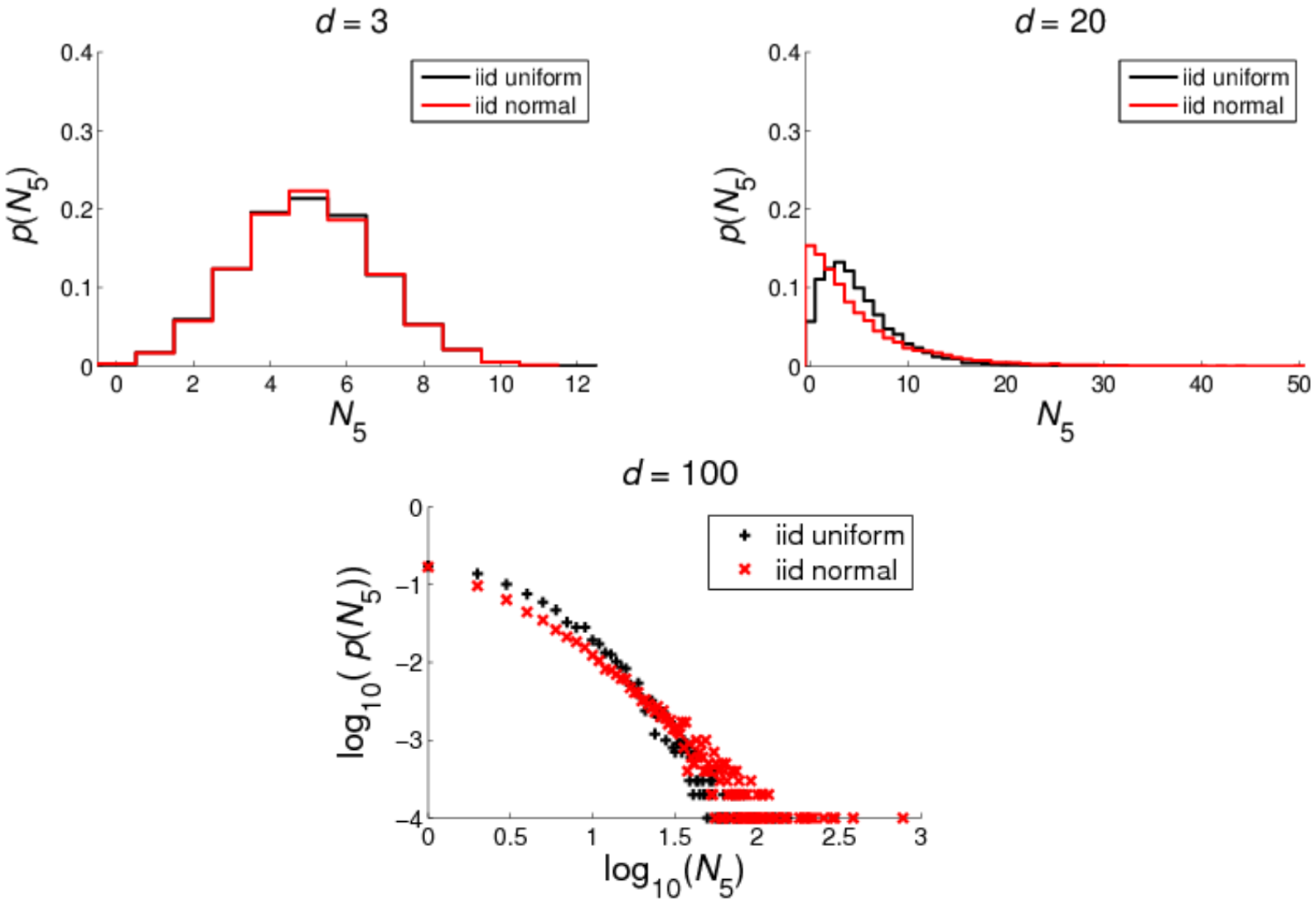
Outline

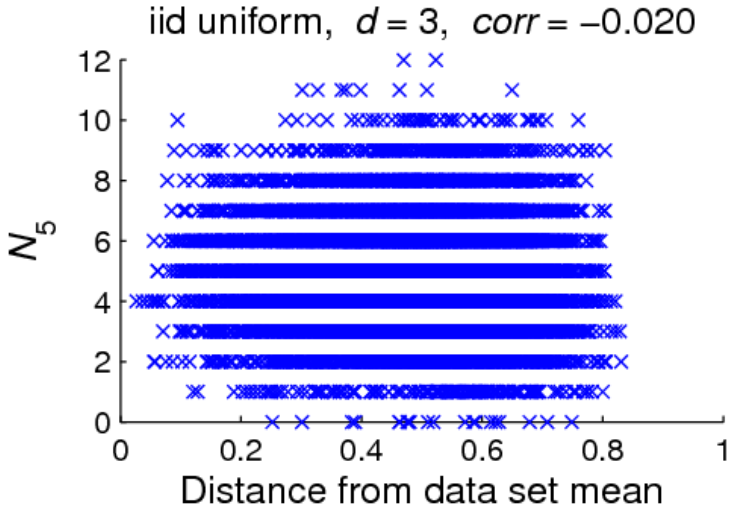
- Origins
  - Definition, causes, distance concentration, real data, dimensionality reduction, large neighborhoods
- Applications
  - Approach 1: Getting rid of hubness
  - Approach 2: Taking advantage of hubness
  - Software
- Challenges
  - Outlier detection, kernels, causes – theory, *k*NN search, dimensionality reduction, others…

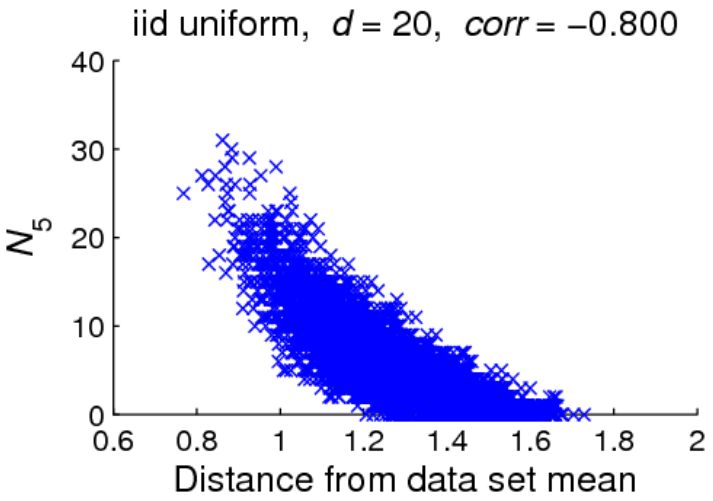
The Hubness Phenomenon

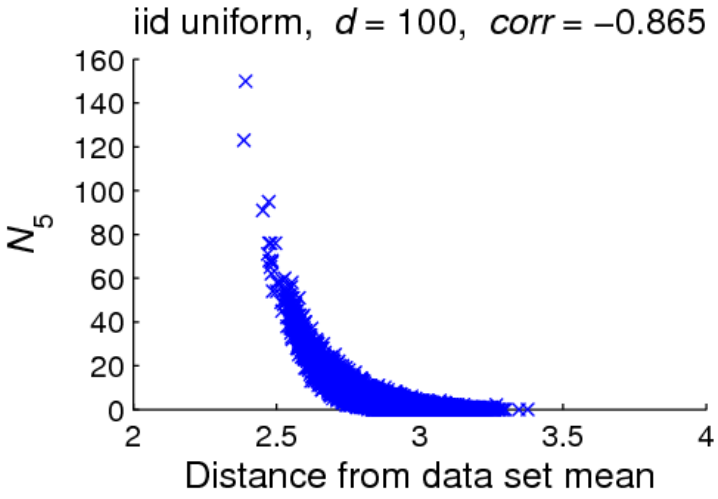
[Radovanović et al. ICML'09, Radovanović et al. JMLR'10]

- $N_k(x)$, 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
  - $N_k(x)$ is the in-degree of node x in the kNN digraph
- Observed that the distribution of N_k can become skewed, resulting in **hubs – points with high N_k**, 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
iid uniform
iid normal
p(N5)
N5
d = 20
iid uniform
iid normal
p(N5)
N5
d = 100
iid uniform
iid normal
log10(p(N5))
log10(N5)

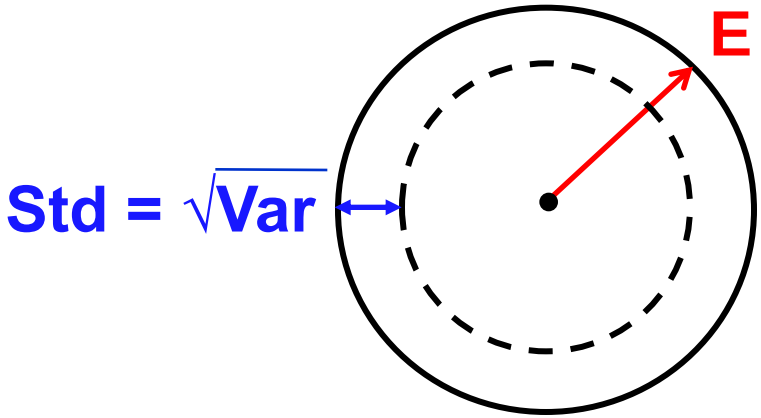

iid uniform, d = 3, corr = –0.020
N5
Distance from data set mean
0
2
4
6
8
10
12
0.2
0.4
0.6
0.8
1


iid uniform, d = 20, corr = –0.800
N5
Distance from data set mean
0
10
20
30
40
0.6
0.8
1
1.2
1.4
1.6
1.8
2


iid uniform, d = 100, corr = –0.865
N5
Distance from data set mean
0
20
40
60
80
100
120
140
160
2
2.5
3
3.5
4

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**



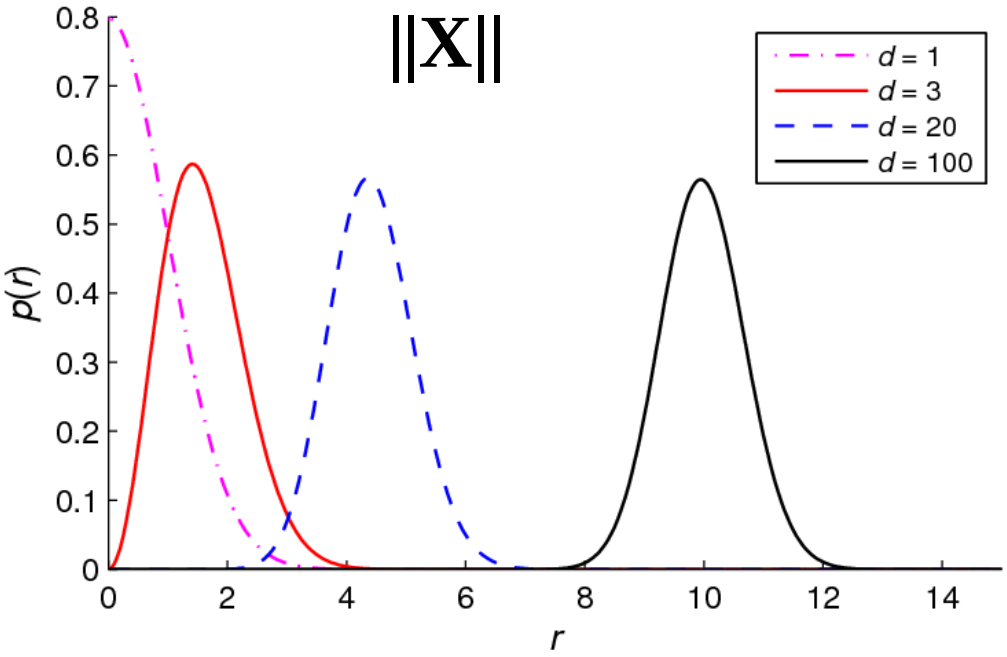
- **Centrality:** points closer to the data set center tend to be closer to all other points (regardless of dimensionality)

Centrality is amplified by high dimensionality

Causes of Hubness

Standard normal distribution of data

Distribution of Euclidean distances of points to data set center (0) = Chi distribution with *d* degrees of freedom



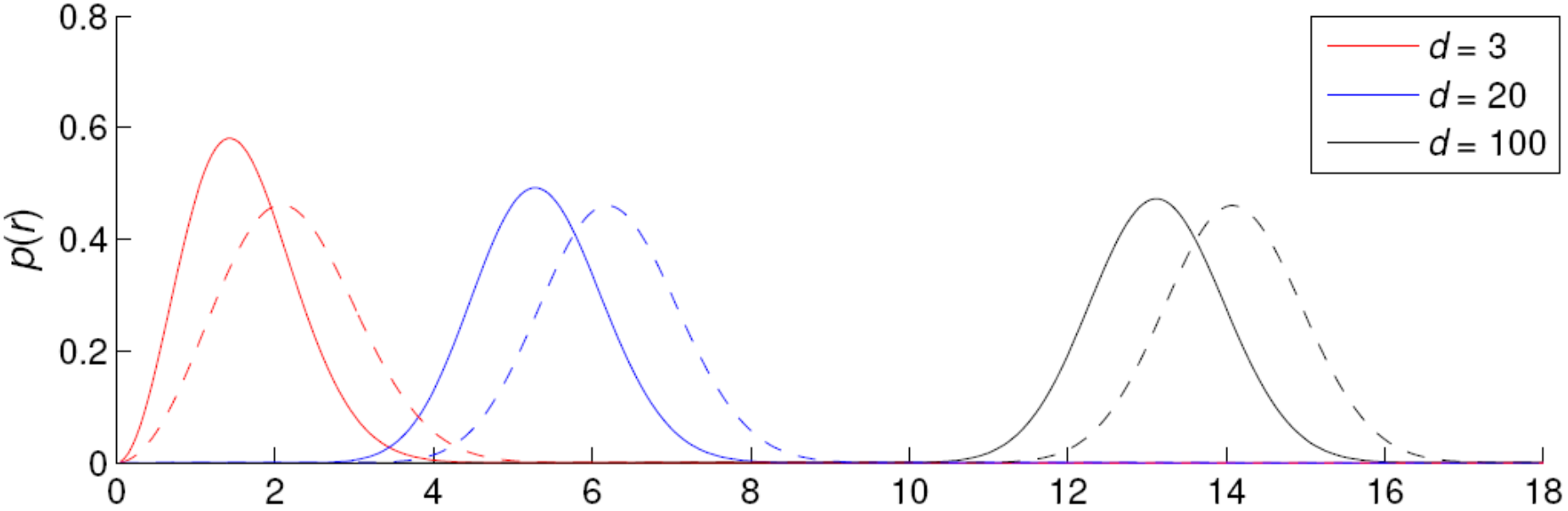
Causes of Hubness

Standard normal distribution of data

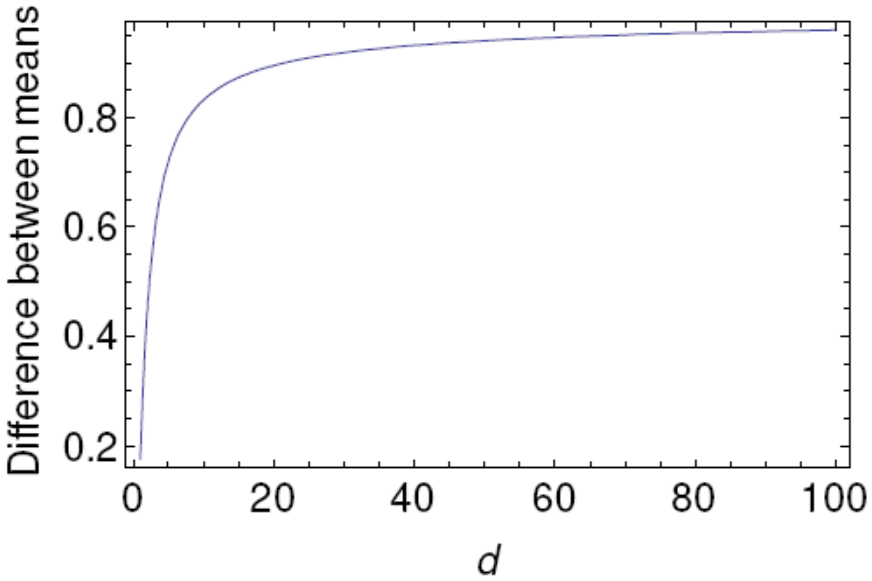
Distribution of Euclidean distances of points to:

- Point at expected distance from 0: E(||X||) (dashed lines)
- Point 2 standard deviations closer: E(||X||) – 2·Std(||X||) (full lines)

= Noncentral Chi distribution with *d* degrees of freedom



Causes of Hubness



Theorem [Radovanović et al. JMLR'10]**:** The ascending behavior holds for iid normal data and any two points at distances E + c_1·Std and E + c_2·Std, for c_1, $c_2 \leq 0$, $c_1 < c_2$

In the above example: $c_1 = -2$, $c_2 = 0$

[Suzuki et al. EMNLP'13] discuss similar result for dot-product similarity and more arbitrary data distribution

Important to Emphasize

- Generally speaking, **concentration does not CAUSE hubness**
- Causation might be possible to derive under certain assumptions
- 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

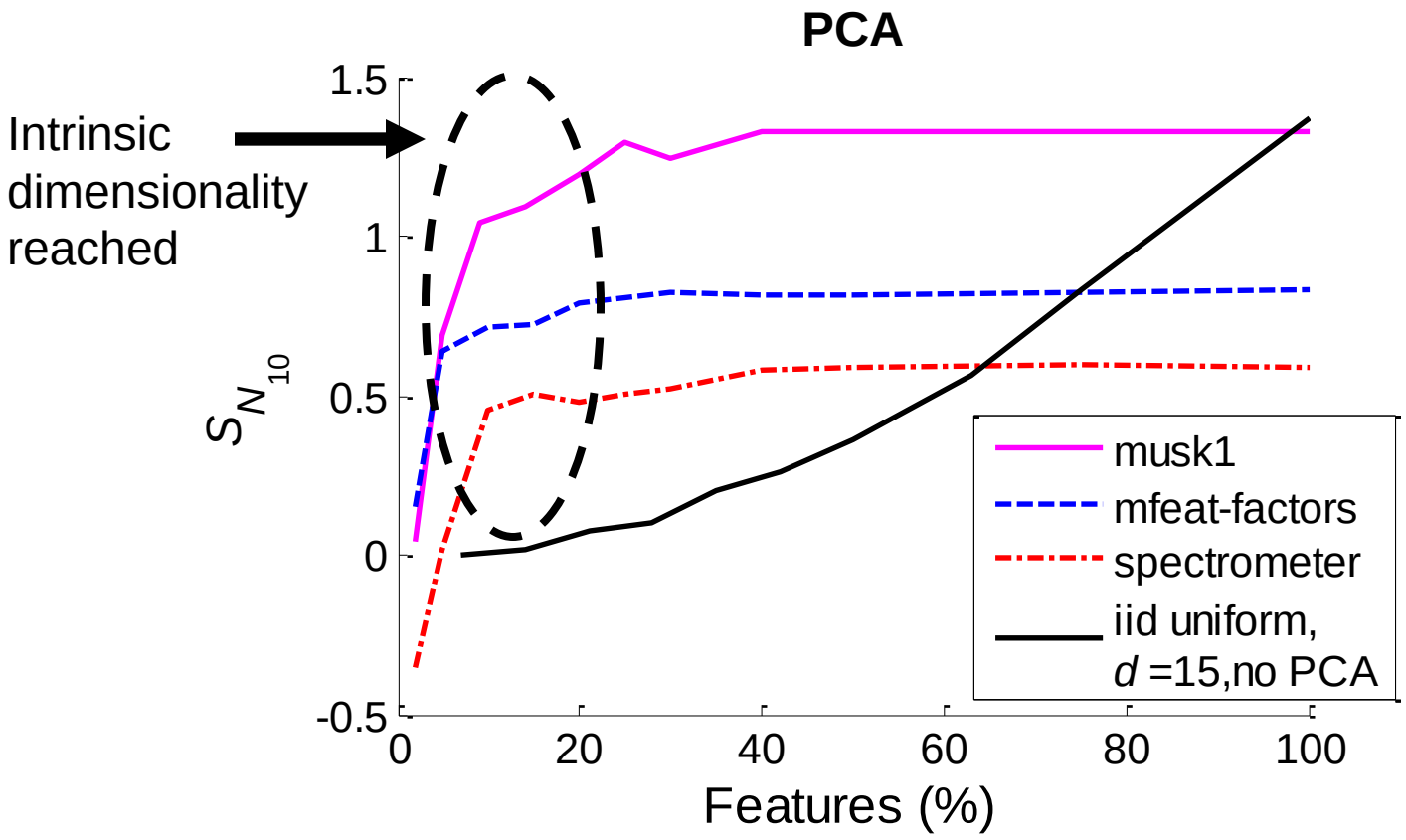
- Important factors for real data
  1) **Dependent attributes**
  2) **Grouping (clustering)**

- 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**

Name	n	d	d_{mle}	Cls.	Clu.	Dist.	$S_{N_{10}}$	$S^S_{N_{10}}$
ecoli	336	7	4.13	8	8	l_2	0.116	0.208
ionosphere	351	34	13.57	2	18	l_2	1.717	2.051
mfeat-factors	2000	216	8.47	10	44	l_2	0.826	5.493
mfeat-fourier	2000	76	11.48	10	44	l_2	1.277	4.001
musk1	476	166	6.74	2	17	l_2	1.327	3.845
optdigits	5620	64	9.62	10	74	l_2	1.095	3.789
page-blocks	5473	10	3.73	5	72	l_2	−0.014	0.470
pendigits	10992	16	5.93	10	104	l_2	0.435	0.982
segment	2310	19	3.93	7	48	l_2	0.313	1.111
sonar	208	60	9.67	2	8	l_2	1.354	3.053
spambase	4601	57	11.45	2	49	l_2	1.916	2.292
spectrometer	531	100	8.04	10	17	l_2	0.591	3.123
vehicle	846	18	5.61	4	25	l_2	0.603	1.625
vowel	990	10	2.39	11	27	l_2	0.766	0.935
lungCancer	181	12533	59.66	2	6	l_2	1.248	3.073
ovarian-61902	253	15154	9.58	2	10	l_2	0.760	3.771
mini-newsgroups	1999	7827	3226.43	20	44	cos	1.980	1.765
reuters-transcribed	201	3029	234.68	10	3	cos	1.165	1.693

Hubness and Dimensionality Reduction



- Similar charts for ICA, SNE, isomap, diffusion maps

Hubness in Real Data

Existence of hubness in real data and dependence on dimensionality verified:

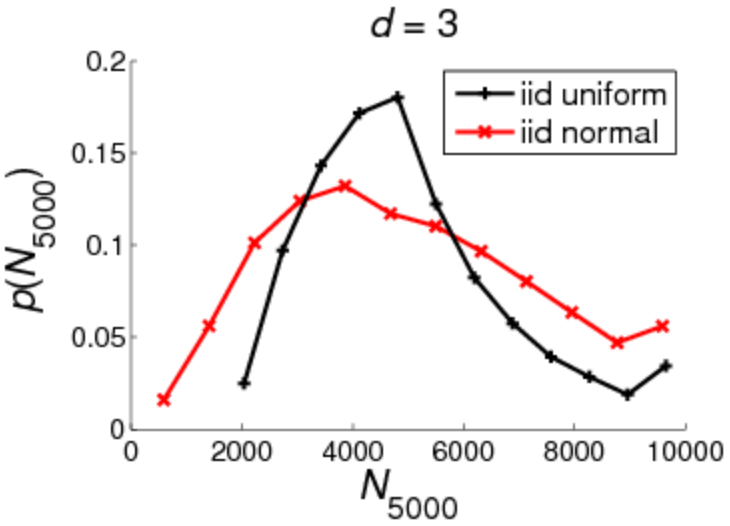
- Various UCI, microarray and text data sets [Radovanović et al. JMLR'10]
- Collaborative filtering data [Nanopoulos et al. RecSys'09, Knees et al. ICMR'14]
- Vector space models for text retrieval [Radovanović et al. SIGIR'10]
- Time series data and "elastic" distance measures (DTW) [Radovanović et al. SDM'10]
- Content-based music retrieval data [Karydis et al. ISMIR'10, Flexer et al. ISMIR'12]
- Doddington zoo in speaker verification [Schnitzer et al. EUSIPCO'13]
- Image data with invariant local features (SIFT, SURF, ORB) [Tomašev et al. ICCP'13]
- Oceanographic sensor data [Tomašev and Mladenić IS'11]
- …

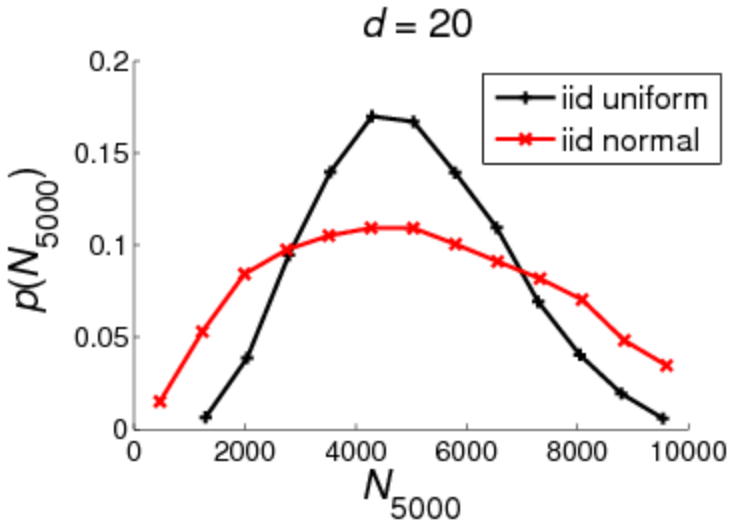
There Are Also Critics

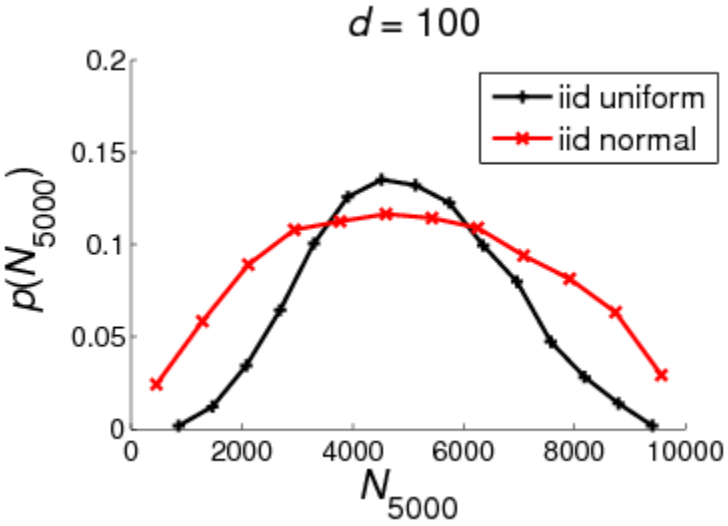
[Low et al. STUDFUZZ'13]

- "The Hubness Phenomenon: Fact or Artifact?"
- "we challenge the hypothesis that the hubness phenomenon is an effect of the dimensionality of the data set and provide evidence that it is rather a boundary effect or, more generally, an effect of a density gradient"
- The "challenge" is easy to overcome by referring to more careful reading of [Radovanović et al. JMLR'10], where boundaries are also discussed in detail (and found to be a dual notion to centrality)
- Nevertheless, the paper articulates the notion of density gradient (empirically), which could prove valuable

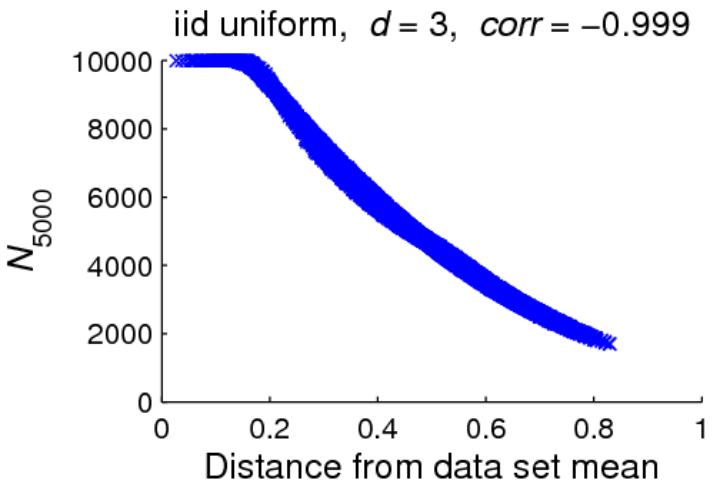
Hubness and Large Neighborhoods

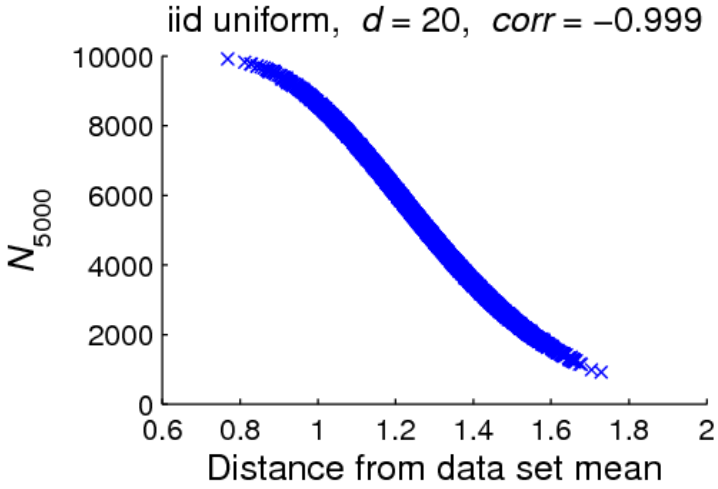


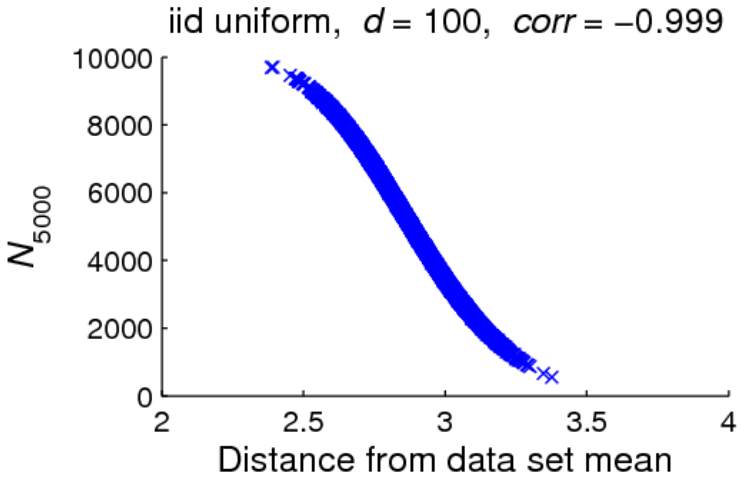




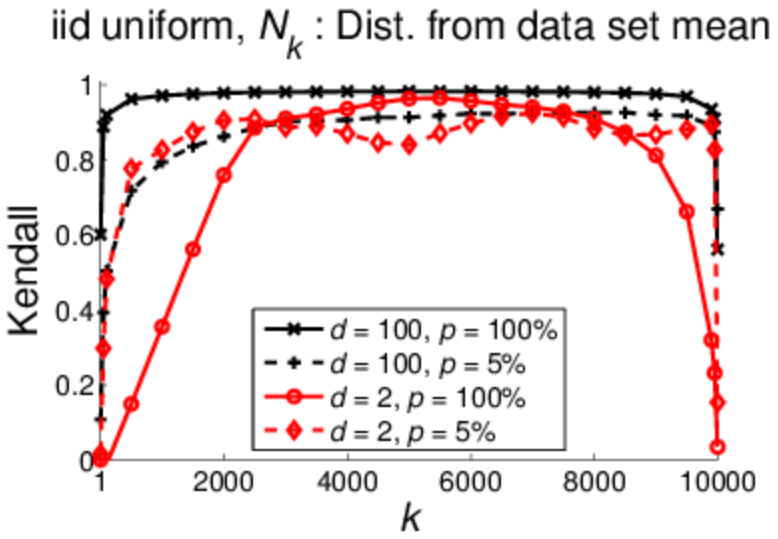
Hubness and Large Neighborhoods

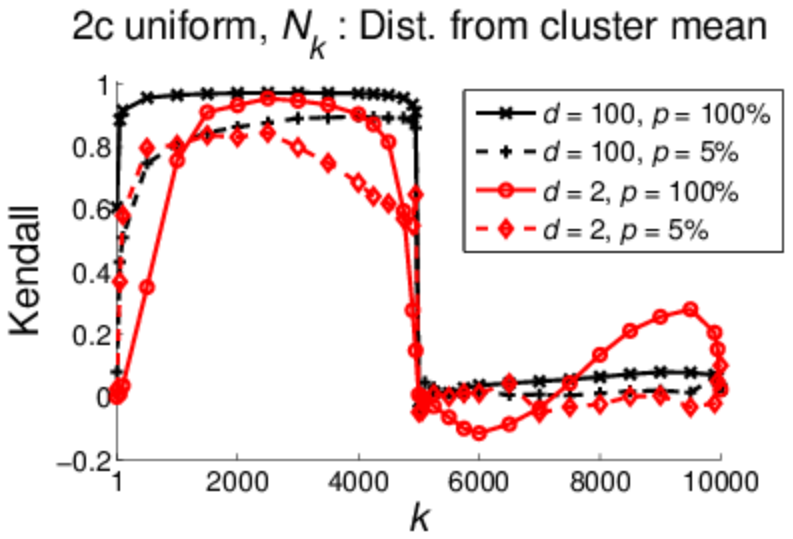






Hubness and Large Neighborhoods





[Radovanović et al. TKDE'15]

Outline

- Origins
  - Definition, causes, distance concentration, real data, dimensionality reduction, large neighborhoods
- Applications
  - Approach 1: Getting rid of hubness
  - Approach 2: Taking advantage of hubness
  - Software
- Challenges
  - Outlier detection, kernels, causes – theory, *k*NN search, dimensionality reduction, others…

Approaches to Handling Hubs

1. **Hubness is a problem** – let's get rid of it
2. **Hubness is OK** – let's take advantage of it

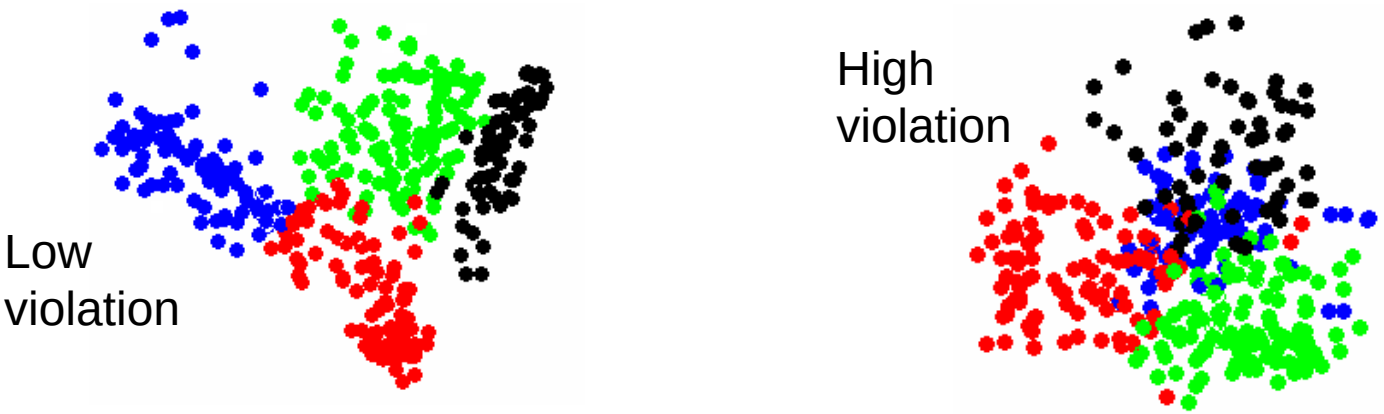
- Hubness is present in many kinds of real data and application domains
- We will review research that actively takes hubness into account (in an informed way)
- But first…

Hubness and Classification

- Based on labels, *k*-occurrences can be distinguished into:
  - **"Bad" *k*-occurrences, $BN_k(x)$**
  - **"Good" *k*-occurrences, $GN_k(x)$**
  - $N_k(x) = BN_k(x) + GN_k(x)$
- **"Bad hubs"** can appear
  - How do bad hubs originate?

How Do “Bad” Hubs Originate?

- The **cluster assumption** [Chapelle et al. 2006]:
  Most pairs of points in a cluster should be of the same class



- Observations and answers [Radovanović et al. JMLR’10]:
  - High dimensionality and skewness of N_k do not automatically induce “badness”
  - Bad hubs originate from a combination of
    1) **high (intrinsic) dimensionality**
    2) **violation of the cluster assumption**

In More General Terms

- General notion of **"error"**
  - Classification error (accuracy)
  - Retrieval error (precision, recall, F-measure)
  - Clustering error (within/between cluster distance)
- Models make errors, but the **responsibility for error** is **not evenly distributed** among data points
- Important to distinguish:
  - Total amount of (responsibility for) error in the data
    - E.g. $\Sigma_x BN_k(x) / \Sigma_x N_k(x)$
  - Distribution of (responsibility for) error among data points
    - E.g. distribution of $BN_k(x)$, i.e. its skewness

In More General Terms

- Hubness generally **does not increase the total amount of error**
- Hubness **skews the distribution of error**, so some points will be more responsible for error than others
- **Approach 1** (getting rid of hubness)
  - May reduce (but also increase) total amount of error in the data
  - Will make distribution of error more uniform
- **Approach 2** (taking advantage of hubness)
  - Will not change total amount of error in the data
  - Will identify points more responsible for error and adjust models accordingly

Outline

- Origins
  - Definition, causes, distance concentration, real data, dimensionality reduction, large neighborhoods
- Applications
  - Approach 1: Getting rid of hubness
  - Approach 2: Taking advantage of hubness
  - Software
- Challenges
  - Outlier detection, kernels, causes – theory, *k*NN search, dimensionality reduction, others…

Mutual *k*NN Graphs

[Ozaki et al. CoNLL'11]

- Graph-based semi-supervised text classification
  - *k*NN graphs
  - Mutual *k*NN graphs + maximum spanning trees
  - *b*-matching graphs [Jebara et al. ICML'09]
- Gaussian random fields (GRF) algorithm
- Mutual *k*NN graphs perform better than *k*NN graphs (and comparably to *b*-matching graphs) due to reduced hubness

Regular Graphs

[Vega-Oliveros et al. JoP'14]

- Graph-based semi-supervised classification
  - *k*NN graphs
  - Undirected regular graphs constructed from *k*NN graphs (regular graph = graph with all degrees equal)
- Local and global consistency (LGC) label propagation algorithm
- Regular graphs perform better than *k*NN graphs due to non-existing hubness

Centering and Hub Reduction

[Suzuki et al. AAAI'12]

- Ranking (IR), multi-class and multi-label *k*NN classification
- Laplacian-based kernels tend to make all points equally similar to the center, thus reducing hubness (compared to plain cosine similarity)
- When hubness is reduced, the kernels work well

[Suzuki et al. EMNLP'13]

- Text classification
- Centering reduces hubness, since it also makes all points equally similar to the center, using dot-product similarity
  - I would add, centering reduces centrality (the existence of centers in the data) w.r.t dot-product similarity
- For multi-cluster data, weighted centering which moves hubs closer to the center achieves a similar effect

Local and Global Scaling

[Schnitzer et al. JMLR'12]

- Content-based music retrieval
- **Idea:** rescale distances between *x* and *y* so that distance is small only if *x* is a close neighbor to *y* and *y* is a close neighbor to *x*
- **Local scaling:** non-iterative contextual dissimilarity measure

$$LS(d_{x,y}) = d_{x,y} / (\mu_x \mu_y)^{½}$$

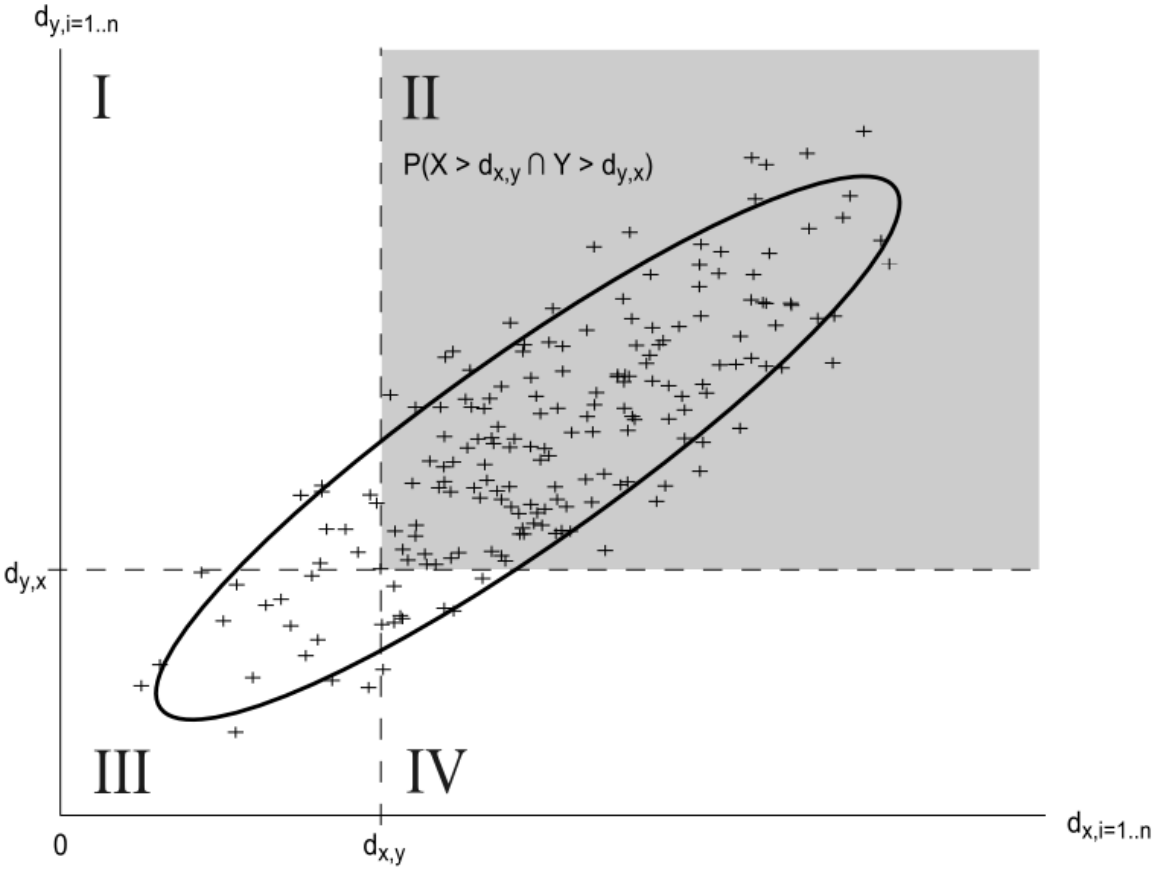
where μ_x (μ_y) is the avg. distance from *x* (*y*) to its *k* NNs

- **Global scaling:** mutual proximity

$$MP(d_{x,y}) = P(X > d_{x,y} \cap Y > d_{y,x})$$

where X (Y) follows the distribution of distances from *x* (*y*) to all other points

Mutual Proximity Visualized



Properties of LS and MP

- Both LS and MP reduce hubness, improving *k*NN classification accuracy
- MP easier to approximate for large data, successfully applied to music retrieval
- Methods do not reduce intrinsic dimensionality of data
- Hubs/anti-hubs remain as such, but to a lesser degree
- Regarding **error ("badness")**, the methods:
  - Reduce badness of hubs
  - Introduce badness to anti-hubs
  - Badness of regular points stays roughly the same, but less than for both hubs and anti-hubs
- LS can benefit from varying neighborhood size based on class labels or clustering [Lagrange et al. ICASSP'12]
- MP successfully applied to neighbor-based collaborative filtering [Knees et al. ICMR'14]
  - MP improves data point coverage in NN graph

Shared Nearest Neighbors

[Flexer & Schnitzer HDM'13]

- Classification
- Consider **shared neighbor similarity**:

$$SNN(x,y) = |D_k(x) \cap D_k(y)| \,/\, k$$

where $D_k(x)$ is the set of k NNs of x

- Use this measure for computing the kNN graph
- SNN reduces hubness, but not as much as LS and MP
- SNN can improve kNN classification accuracy, but overall worse than LS and MP

A Case for Hubness Removal

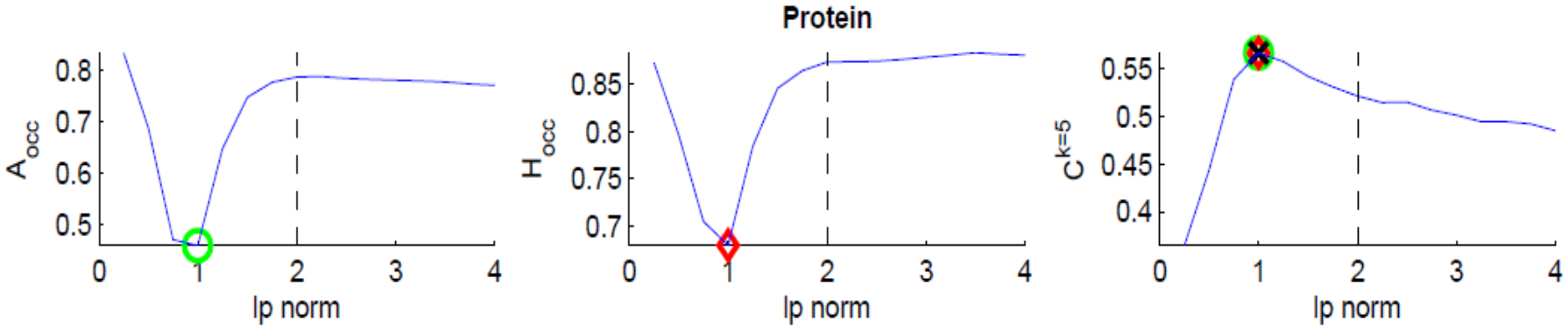
[Schnitzer et al. ECIR'14]

- Multimedia retrieval: text, images, music
- SNN, and especially LS and MP, in all above domains:
  - Reduce hubness
  - Improve data point coverage (reachability)
  - Improve retrieval precision/recall

Choosing the Metric

[Schnitzer & Flexer ESANN'14]

- Examine l_p norms for $p \in \{.25, .5, \ldots, 4\}$
- For a data set compare:
  - Percentage of anti-hubs (points with $N_k = 0$, $k = 1$)
  - Percentage of hubs (points with $N_k > 2k$, $k = 1$)
  - kNN classification accuracy ($k = 5$)
- Values of p with lowest percentage of (anti-)hubs and highest kNN accuracy tend to coincide



Automatic Speech Recognition

[Vincent et al. Interspeech'14]

- Automatic speech recognition (ASR)
- Speech units: hidden Markov models (HMM) with Gaussian mixture model (GMM) observation densities
- Introduce various kinds of normalization to vector observation scores for different states
- Two notions of hubness
  - Of states w.r.t. the *k* most likely feature vectors
  - Of feature vectors w.r.t. the *k* most likely states
- Normalization reduces hubness and increases accurracy

Other Ways to Avoid Hubs

[Murdock and Yaeger ECAL'11]

- Using clustering to identify species in genetic algorithms
- QT clustering algorithm uses ε-neighborhoods, where there is no hubness

[Van Parijs et al. LSM'13]

- Approaches to modify *k*NN graphs of users/items, improving recommendation:
  - Remove strongest hubs
  - Normalize similarities (then recompute graph)
  - Replace similarities with ranks (then recompute graph)

[Lajoie et al. Genome Bilogy'12]

- Regulatory element discovery from gene expression data
- *k*NN graph between genes is first symmetrized
- *k* neighbors sampled with probability inversely proportional to N_k

[Schlüter MSc'11]

- Overview and comparison of methods for hub reduction in music retrieval
- Methods mostly unaware of the true cause of hubness

Outline

- Origins
  - Definition, causes, distance concentration, real data, dimensionality reduction, large neighborhoods
- Applications
  - Approach 1: Getting rid of hubness
  - Approach 2: Taking advantage of hubness
  - Software
- Challenges
  - Outlier detection, kernels, causes – theory, *k*NN search, dimensionality reduction, others…

Extending the *k*NN Classifier

- "Bad" hubs provide erroneous class information to many other points
- hw-*k*NN [Radovanović et al. JMLR'10]:
  - We introduce standardized "bad" hubness:

$$h_B(x, k) = (BN_k(x) - \mu_{BN_k}) / \sigma_{BN_k}$$

  - During majority voting, the vote of each neighbor x is weighted by

$$\exp(-h_B(x, k))$$

Extending the *k*NN Classifier

- Drawbacks of hw-*k*NN:
  - Does not distinguish between classes when computing "badness" of a point
  - Still uses the crisp voting scheme of *k*NN

- Consider class-specific hubness scores $N_{k,c}(x)$:
  The number of *k*-occurrences of *x* in neighbor sets of class *c*

  - h-FNN, Hubness-based Fuzzy NN [Tomašev et al. MLDM'11, IJMLC]:
    Vote in a fuzzy way by class-specific hubness scores $N_{k,c}(x)$
  - NHBNN, Naïve Hubness Bayesian NN [Tomašev et al. CIKM'11]:
    Compute a class probability distribution based on $N_{k,c}(x)$
  - HIKNN, Hubness Information *k*NN [Tomašev & Mladenić ComSIS'12]:
    Information-theoretic approach using $N_{k,c}(x)$
  - ANHBNN, Augmented Naïve Hubness Bayesian NN [Tomašev & Mladenić ECML'13]:
    Extends NHBNN using the Hidden Naïve Bayes model to take into account hub co-occurrences in NN lists

Why Hub-Based Classifiers Work

[Tomašev & Mladenić KBS'13]

- Data with **imbalanced classes**
- **"Bad" hubs from MINORITY classes usually responsible for most error**
- Favoring minority class data points (standard approach) makes the problem worse
- Hubness-based classifiers improve precision on minority classes and recall on majority classes
- May be beneficial to combine the hubness-aware voting approaches with the existing class imbalanced *k*NN classifiers
  - Realistically, minority classes need to be favored
  - Minority (bad) hubs need to be taken into account

[Tomašev & Buza Neurocomputing'14]

- Hub-based classifiers (especially h-FNN and NHBNN) are robust to different kinds of label noise

Clustering

[Radovanović et al. JMLR'10]

- Distance-based clustering objectives:
  - **Minimize within-cluster distance**
  - **Maximize between-cluster distance**

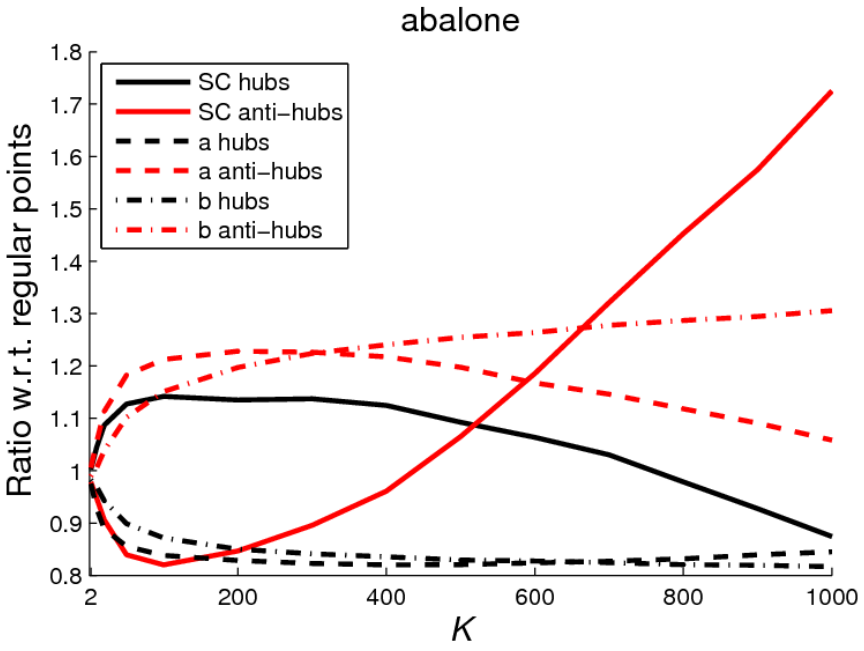
- **Skewness of N_k affects both objectives**
  - Outliers do not cluster well because of high within-cluster distance
  - Hubs also do not cluster well, but because of low between-cluster distance

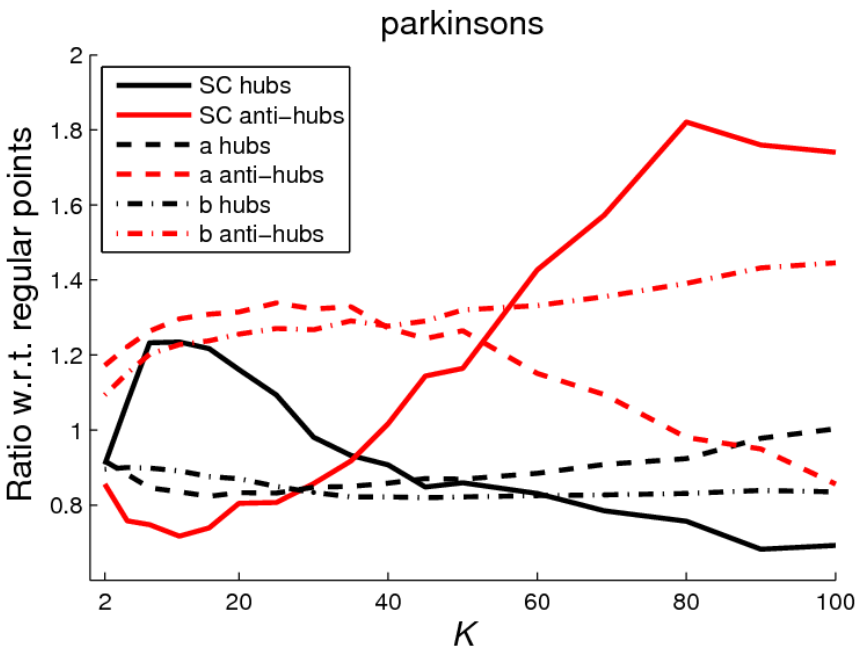
Clustering

- Silhouette coefficient (SC): For *i*-th point
  - a_i = avg. distance to points from its cluster (within-cluster distance)
  - b_i = min. avg. distance to points from other clusters (between-cluster distance)
  - $SC_i = (b_i - a_i) / \max(a_i, b_i)$
    - In range [–1, 1], higher is better
  - SC for a set of points is the average of SC_i for every point *i* in the set

Clustering

[Tomašev et al. PCA'14]





Using Hubs as Cluster Centers

[Tomašev et al. PAKDD'11, TKDE'14]

Algorithm 1 K-hubs

```
initializeClusterCenters();
Cluster[] clusters = formClusters();
repeat
    for all Cluster c ∈ clusters do
        DataPoint h = findClusterHub(c);
        setClusterCenter(c, h);
    end for
    clusters = formClusters();
until noReassignments
return clusters
```

Exploiting the Hubness of Points

Algorithm 2 HPC

```
initializeClusterCenters();
Cluster[] clusters = formClusters();
float t = t_0; {initialize temperature}
repeat
    float θ = getProbFromSchedule(t);
    for all Cluster c ∈ clusters do
        float choice = randomFloat(0,1);
        if choice < θ then
            DataPoint h = findClusterHub(c);
            setClusterCenter(c, h);
        else
```

Exploiting the Hubness of Points

```
            for all DataPoint x ∈ c do
                setChoosingProbability(x, N_k^2(x));
            end for
            normalizeProbabilities();
            DataPoint h = chooseHubProbabilistically(c);
            setClusterCenter(c, h);
        end if
    end for
    clusters = formClusters();
    t = updateTemperature(t);
until noReassignments
return clusters
```

Exploiting the Hubness of Points

- Algorithm 3 **HPKM**
  The same as HPC, except for one line

HPC:

```
if randomFloat(0,1) < θ then
    DataPoint h = findClusterHub(c);
    setClusterCenter(c, h);
```

HPKM:

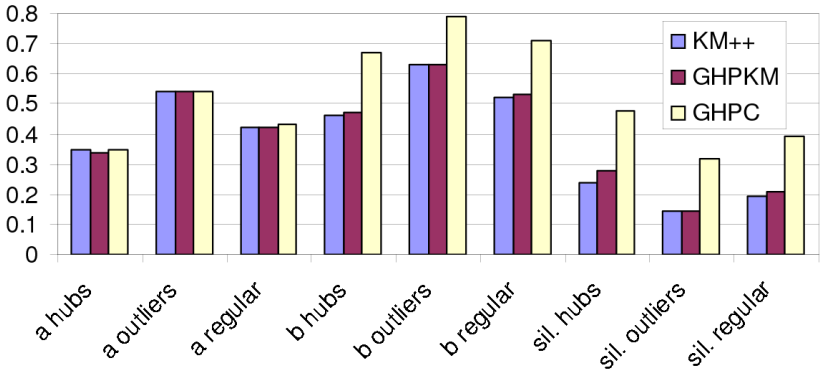
```
if randomFloat(0,1) < θ then
    DataPoint h = findClusterCentroid(c);
    setClusterCenter(c, h);
```

- "Kernelized" extension of HPKM
  [Tomašev et al. PCA'14]

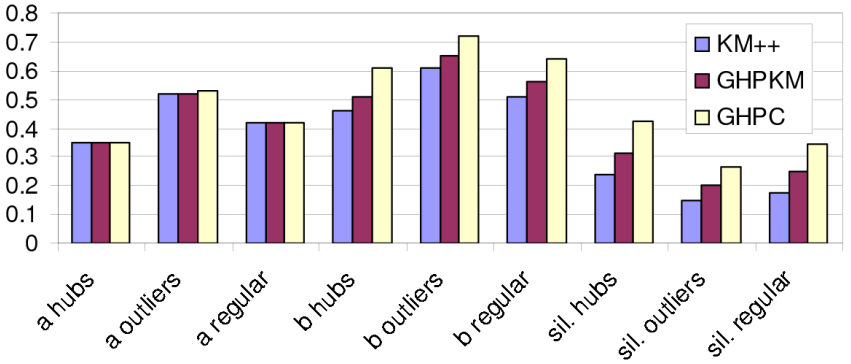
Why Hub-Based Clustering Works

- Hub-based clustering more robust to noise
- Improves between-cluster distance (b component of SC), especially for hubs

Miss-America, Part 1



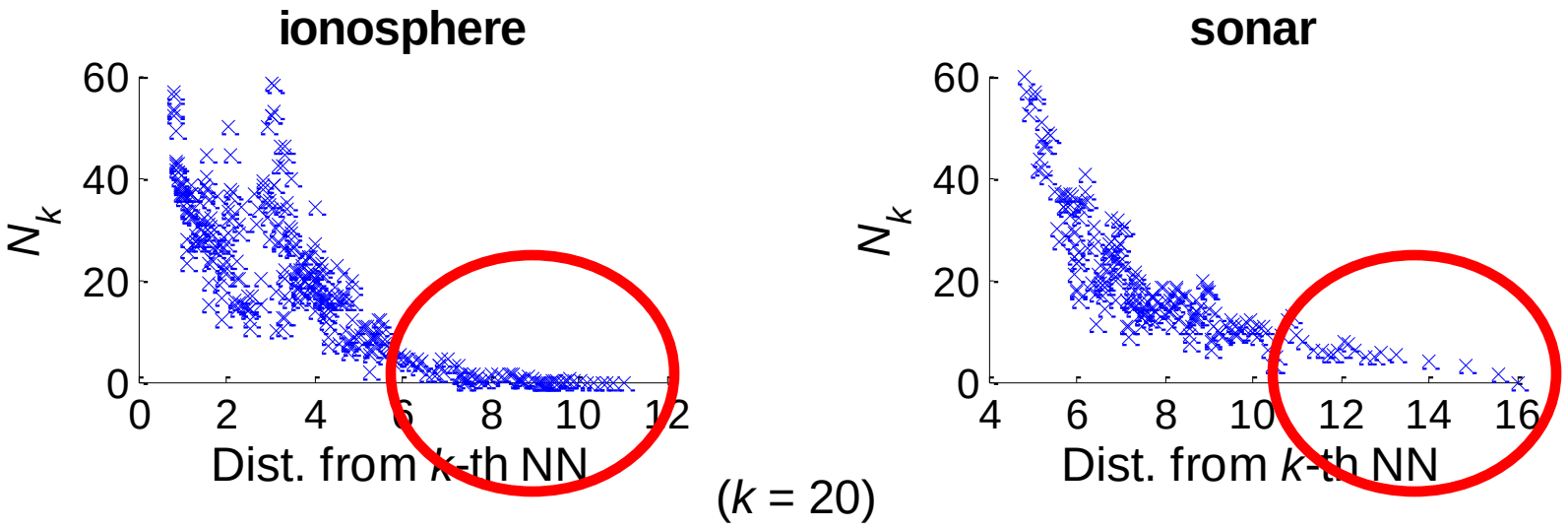
Miss-America, Part 2



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



Outlier Detection

[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…

Outlier Detection

[Radovanović et al. TKDE'15]

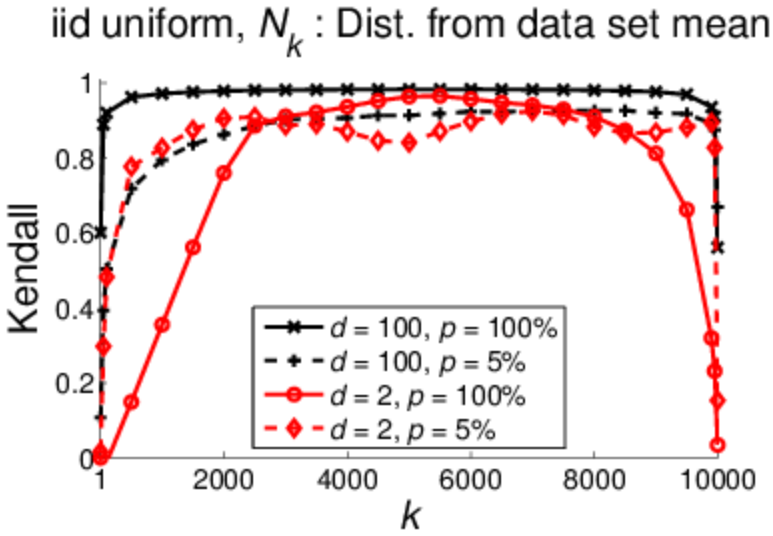
- In method AntiHub, we use $N_k(x)$ as the outlier score of x (same as ODIN, without the threshold)
- Through experiments we identified its strengths and weaknesses with respect to different factors (properties):
  1. Hubness
  2. Locality vs. globality
  3. Discreteness of scores
  4. Varying density
  5. Computational complexity

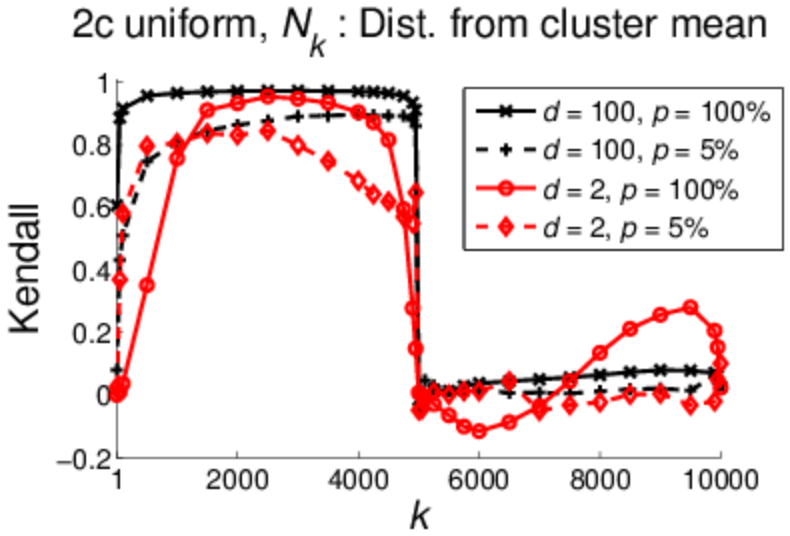
Outlier Detection

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

Outlier Detection





[Radovanović et al. TKDE'15]

Outlier Detection

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” can have problem with discrimination
- Raising *k* can address this, but the notion of outlierness goes global

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

Outlier Detection

Property 4: Varying density

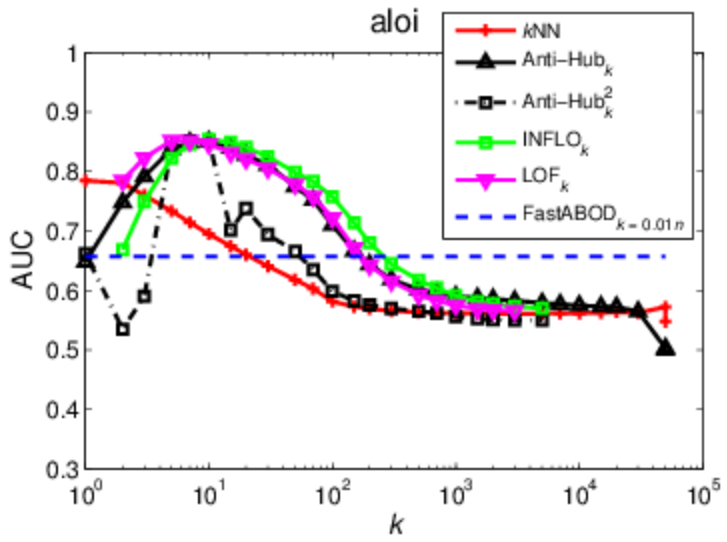
- AntiHub is not sensitive to the scale of distances in the data
- Can effectively detect (local) outliers in clusters of different densities without explicitly modeling density

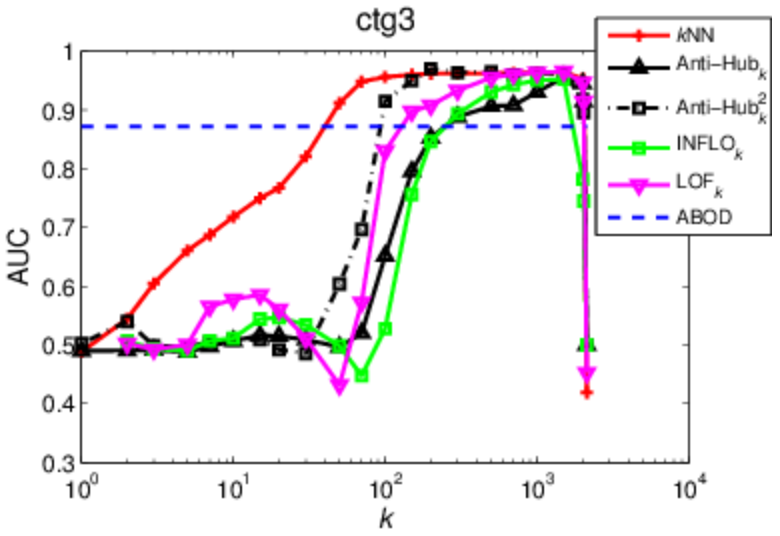
Property 5: Computational complexity

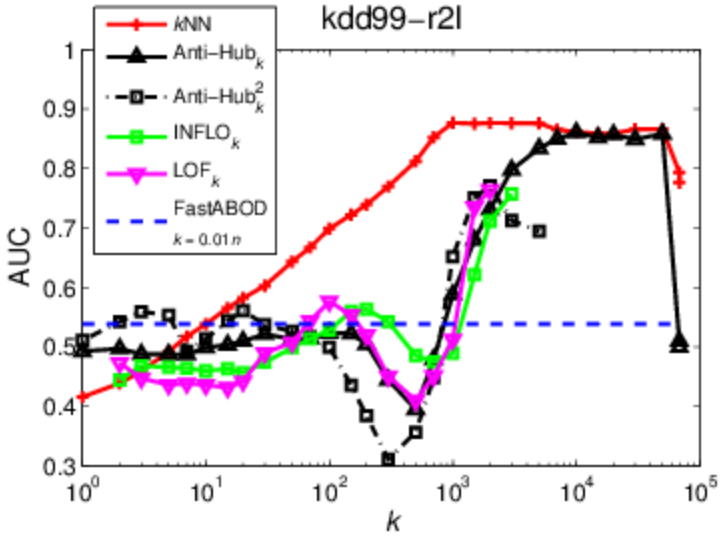
- Using high *k* values can be useful
- However, approximate *k*NN search/indexing methods typically assume $k = O(1)$

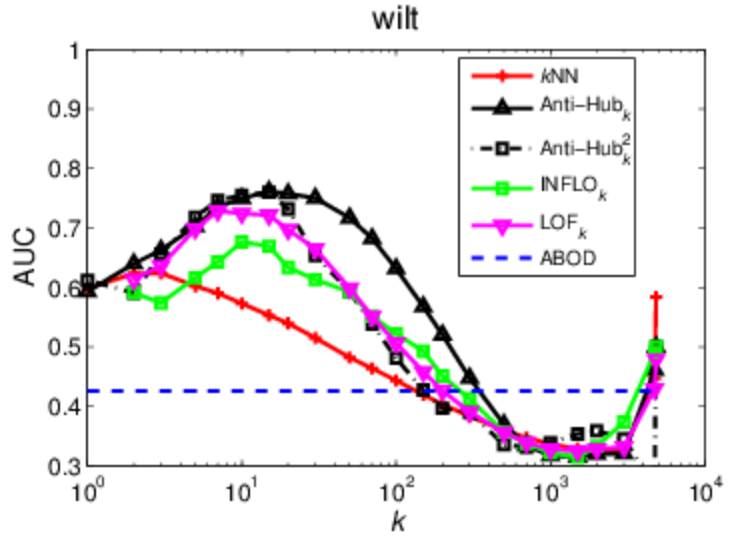
Outlier Detection

- Notable weakness of AntiHub: discrimination of scores, contributed to by two factors: hubness and 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, so as to maximize discrimination
- AntiHub^2 improves discrimination of scores compared to AntiHub (always), as well as AUC (on many data sets)
- 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)


aloi
kNN
Anti-Hub_k
Anti-Hub_k^2
INFLO_k
LOF_k
FastABOD_k=0.01n
AUC
k


ctg3
kNN
Anti-Hub_k
Anti-Hub_k^2
INFLO_k
LOF_k
ABOD
AUC
k


kdd99-r2l
kNN
Anti-Hub_k
Anti-Hub_k^2
INFLO_k
LOF_k
FastABOD
k=0.01n
AUC
k


wilt
kNN
Anti-Hub_k
Anti-Hub_k^2
INFLO_k
LOF_k
ABOD
AUC
k

Instance Selection

[Radovanović et al. JMLR'10]

- Support vector machine classifier
- Bad hubs tend to be good support vectors

[Kouimtzis MSc'11]

- Confirm and refine above observation
- Observe ratio $BN_k(x)$ / $GN_k(x)$
- Two selection methods:
  - RatioOne: Prefer ratios closest to 1 in absolute value
  - BelowOne: Prefer ratios lower than 1
- BelowOne performs better than random selection
- RatioOne comparable to BelowOne only on larger sample sizes
- BelowOne selects instances on the border, but closer to class centers

Instance Selection

[Lazaridis et al. Signal Processing: Image Communication'13]

- Multimodal indexing of multimedia objects (text, 2D image, sketch, video, 3D objects, audio and their combinations)
- Select dimensionality of multimodal feature space (20) to maximize hubness while keeping computational cost reasonable
- Select reference objects for indexing as strongest hubs

[Buza et al. PAKDD'11]

- Improve speed and accuracy of 1NN time-series classification
- INSIGHT: Select a small percentage of instances *x* based on largest
  - $GN_1(x)$
  - $GN_1(x) / (N_1(x) + 1)$
  - $GN_1(x) - 2BN_1(x)$
- The approach using GN_1 is optimal in the sense of producing the best 1NN coverage (label-matching) graph

Feature Construction

[Tomašev et al. FSDPR'14]

- Extension of the approach by [Buza et al. PAKDD'11]
- Construct new features using DTW distances:
  - Split training set into two disjoint sets (T_1 and T_2)
  - Selest some instances from T_1
    - Two approaches: by largest GN_1 (HubFeatures) and randomly (RndFeatures)
  - Compute distances of instances from T_2 to selected ones from T_1
  - Use the distances as feature vectors for instances from T_2
  - Train a classifier (logistic regression) on this
- The two approaches HubFeatures and RndFeatures, and INSIGHT, work well for *k*NN time-series classification

Local Image Feature Selection

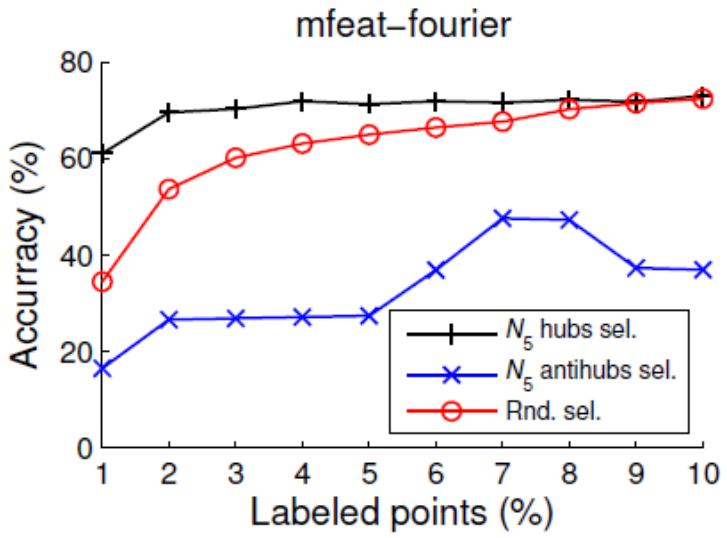
[Wang et al. PR'11]

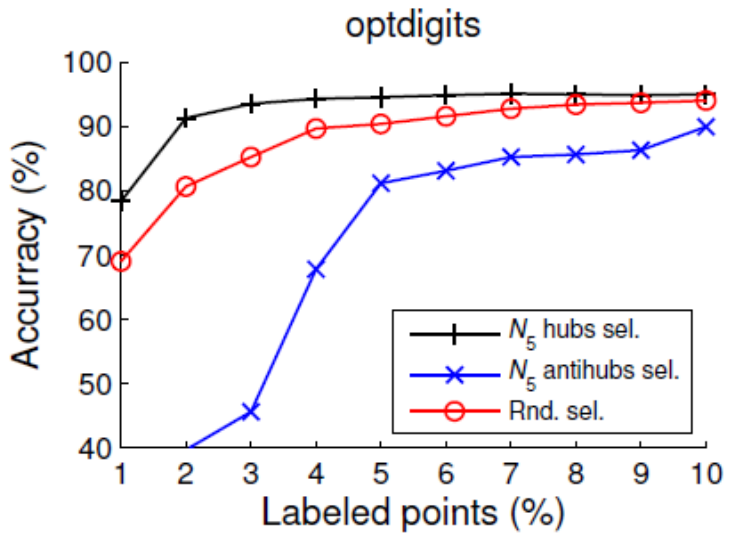
- Improving image-to-class (I2C) distance
- Image features extracted locally from each image, and thus have:
  - Vector descriptors (that can be compared)
  - Associated class information (of the image)
- Reduce cost of NN search in testing phase by:
  - Removing features with low N_k
  - Keeping features with high GN_k / BN_k

Semi-Supervised Classification

[Radovanović et al. JMLR'10]

- Gaussian random fields (GRF) algorithm
- Active learning scenario: request labels from hubs, random points, or anti-hubs first





Semi-Supervised Time-Series Classification

[Marussy TDK'12]

- SLINK classifier:
  1. Cluster all points with single-linkage agglomerative hierarchical clustering with:
     - Cannot link constraints (don't link labeled points)
     - DTW distance
  2. Label top-level clusters with their "seeds"
  3. Use 1NN with produced labeling
- Produces as many clusters as there are labeled points
- Works well on many data sets
- Assumption: almost all hubs are (transitively) good hubs

Cross-Lingual Document Retrieval

[Tomašev et al. PAKDD'13]

- Acquis aligned corpus data (labeled), focus on English and French
- Frequent neighbor documents among English texts are usually also frequent neighbors among French texts
- Good/bad neighbor documents in English texts are expected to be good/bad neighbor documents in French
- Canonical correlation analysis (CCA) is a dimensionality reduction technique similar to PCA, but:
  - Assumes the data comes from two views that share some information (such as a bilingual document corpus)
  - Instead of looking for linear combinations of features that maximize the variance it looks for a linear combination of feature vectors from the first view and a linear combination from the second view, that are maximally correlated
- Introduce instance weights that (de)emphasize (bad) hubs in CCA
- Emphasizing hubs gives most improvement in classification and retrieval tasks

Similarity Adjustment

[Radovanović et al. SIGIR'10]

- Document retrieval in the vector space model (TF-IDF + cosine sim., BM25, pivoted cosine)
- For document *x*, query *q*, we adjust similarity sim(*x*, *q*) as follows:

$$\text{sim}_a(x, q) = \text{sim}(x, q) + \text{sim}(x, q) \cdot (GN_k(x) - BN_k(x)) / N_k(x)$$

[Tomašev et al. ITI'13]

- Bug duplicate detection in software bug tracking systems (TF-IDF + cosine sim. over bug report text)
- Similarity adjustment, observing only the past μ occurrences of *x*:

$$\text{sim}_a(x, q) = \text{sim}(x, q) + \text{sim}(x, q) \cdot GN_{k,\mu}(x) / N_{k,\mu}(x)$$

An Approach in Between

[Tomašev & Mladenić, HAIS'12, KAIS'14]

- Image classification
- Consider shared neighbor similarity:

$$SNN(x,y) = |D_k(x) \cap D_k(y)| \,/\, k$$

  where $D_k(x)$ is the set of k NNs of x

- Propose a **modified measure *simhub*** which
  - Increases the influence of rare neighbors
  - Reduces the influence of "bad" hubs (considering class-specific hubness $N_{k,c}(x)$ from an information-theoretic perspective)

- *simhub*:
  - Reduces total amount of error (badness)
  - Reduces hubness
  - Bad hubs no longer correlate with hubs (distribution of error is changed)

Outline

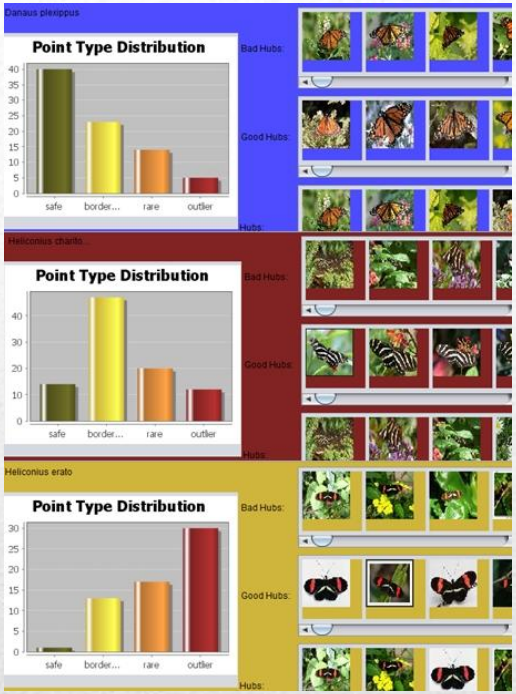
- Origins
  - Definition, causes, distance concentration, real data, dimensionality reduction, large neighborhoods
- Applications
  - Approach 1: Getting rid of hubness
  - Approach 2: Taking advantage of hubness
  - Software
- Challenges
  - Outlier detection, kernels, causes – theory, *k*NN search, dimensionality reduction, others…

Hub Miner: Hubness-aware Machine Learning

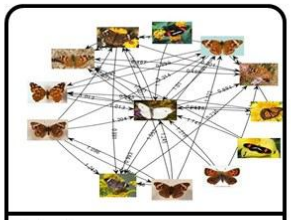
- Java-based library for developing and evaluating hubness-aware machine learning approaches
- About 100k lines of code
- Author: Nenad Tomašev
- GitHub link: https://github.com/datapoet/hubminer
- Project page: http://ailab.ijs.si/tools/hub-miner/
- User Manual: https://github.com/datapoet/hubminer/blob/master/HubMinerManual.pdf

Machine learning support

Classification	• kNN, CBWkNN, NWKNN, AKNN, FNN, dw-kNN, hw-kNN, h-FNN, dwh-FNN, HIKNN, NHBNN, ANHBNN, Naive Bayes, LWNB, KNNNB, ID3
Clustering	• K-means variants, DBScan, LKH, GKH, GHPC, LHPC, GHPKM, Kernel-GHPKM
Metric learning	• Local scaling, NICDM, mutual proximity, simcos, simhub
Instance Selection	• ENN, CNN, GCNN, RT3, RNNR.AL1, INSIGHT
Stochastic Optimization	• Genetic algorithms, simulated annealing, differential evolution, predator-prey particle swarm optimization

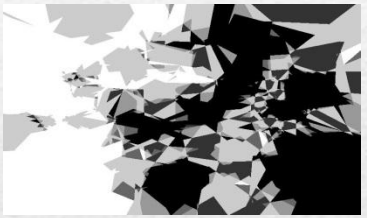


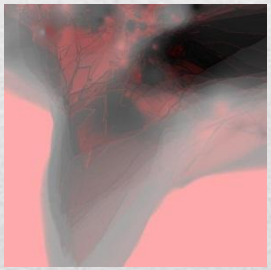
Data overview

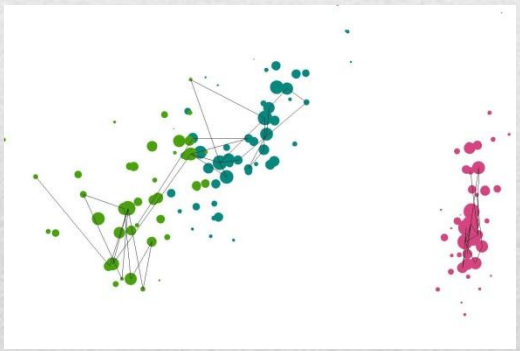


Hub Analysis

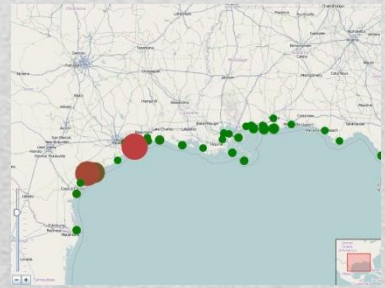
Feature Evaluation

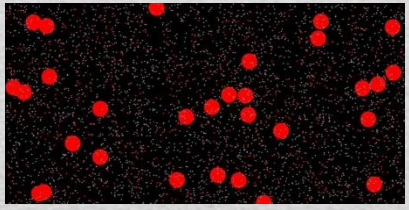


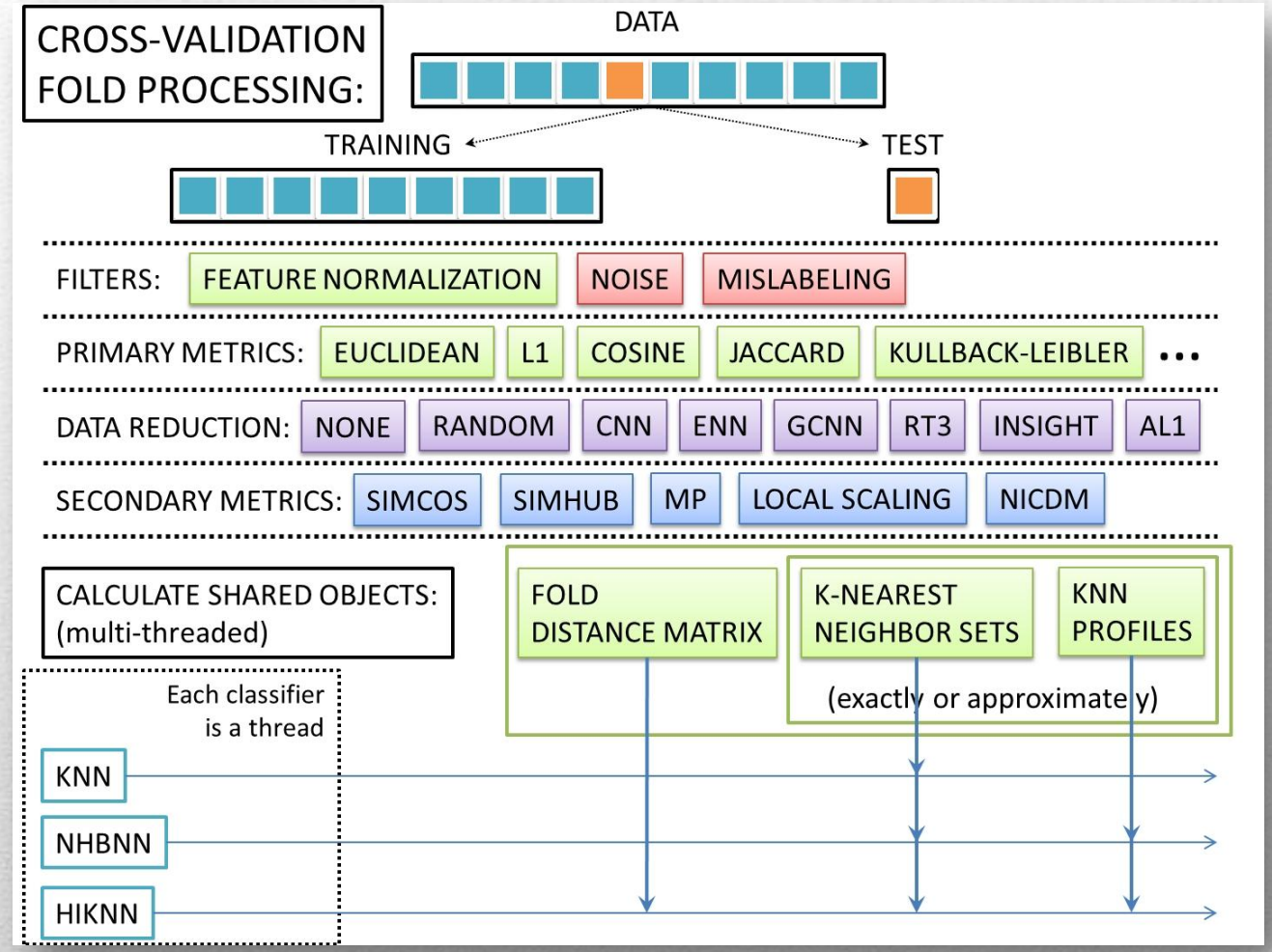




Hub Miner Visualization







Hub Miner Experimentation

Outline

- Origins
  - Definition, causes, distance concentration, real data, dimensionality reduction, large neighborhoods
- Applications
  - Approach 1: Getting rid of hubness
  - Approach 2: Taking advantage of hubness
  - Software
- Challenges
  - Outlier detection, kernels, causes – theory, *k*NN search, dimensionality reduction, others…

Outlier Detection

Challenges [Radovanović et al. TKDE'15]**:**

- High values of *k* can be useful, but:
  - Cluster boundaries can be crossed, producing meaningless results. 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?

Kernels

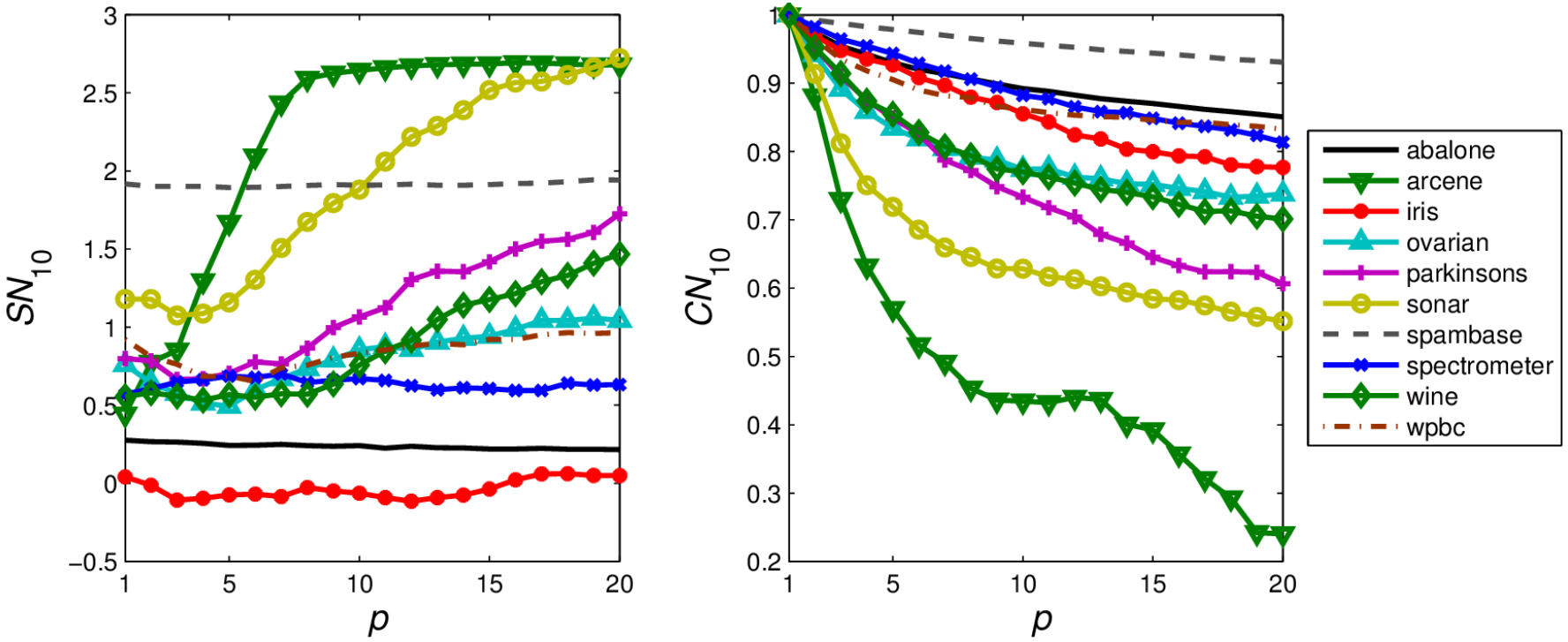
- Little is known about the effects of different kernels (and their parameters) on hubness
- And vice versa, hubness can be a good vehicle for understanding the effects of kernels on data distributions
- For given kernel function $K(x,y)$ and norm distance metric $D(x,y)$ in Hilbert space,

$$D^2(\Psi(x),\Psi(y)) = K(x,x) - 2K(x,y) + K(y,y)$$

- Preliminary investigation in [Tomašev et al. PCA'14], in the context of kernelized hub-based clustering
- Other possible applications: kernelized clustering in general, kernel-*k*NN classifier, SVMs (with only a start given in [Kouimtzis MSc'11])…

Kernels

[Tomašev et al. PCA'14]: polynomial kernel $K(x,y) = (1 + \langle x,y \rangle)^p$



Causes of Hubness: Theory

- Theoretical contribution of [Radovanović et al. JMLR'10] only in terms of properties of distances
- Good strides made in [Suzuki et al. EMNLP'13] for dot-product similarity
- More needs to be done:
  - Explain the causes of hubness theoretically for a large class of distances and data distributions
  - Characterize the distribution of N_k based on the distribution of data, distance measure, number of data points, k
  - Explore the effects of different types of normalization
  - Understand the difference between kNN and ε-neighborhood graphs
- Practical benefits:
  - Geometric models of complex networks (mapping graphs to $\mathbf{R}^d$)
  - Intrinsic dimensionality estimation
  - …

(Approximate) *k*NN Search / Indexing

- HUGE virtually untouched area, with great practical importance
- We did some preliminary experiments, showing that hubness is not severely affected by method from [Chen et al. JMLR'09]
- [Lazaridis et al. 2013] used hubness in a specific multimedia context
- Need for comprehensive systematic exploration of:
  - Interaction between hubness and existing methods
  - Construction of new "hubness-aware" methods
- Possible need for methods that do not assume $k = O(1)$

Dimensionality Reduction

- Apart from simulations in [Radovanović et al. JMLR'10] and instance weighting for CCA in [Tomašev et al. PAKDD'13], practically nothing done

- Many possibilities:
  - Improved objective functions for distance-preserving dimensionality reduction (MDS, PCA)
    - In order to better preserve *k*NN graph structure
    - Or break the *k*NN graph in a controlled way
  - Improve methods based on geodesic distances (Isomap, etc.)
  - …

Other (Possible) Applications

- Information retrieval
  - Investigation of short queries, large data sets
  - Learning to rank
- Local image features (SIFT, SURF…)
  - Hubness affects formation of codebook representations
  - Normalization plays and important role
- Protein folding
- Suggestions?

References

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.

I. Suzuki et al. Centering similarity measures to reduce hubs. In Proc. Conf. on Empirical Methods in Natural Language Processing (EMNLP), pages 613–623, 2013.

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. Nanopoulos et al. How does high dimensionality affect collaborative filtering? In Proc. 3rd ACM Conf. on Recommender Systems (RecSys), pages 293–296, 2009.

P. Knees et al. Improving neighborhood-based collaborative filtering by reducing hubness. Proc. 4th ACM Int. Conf. on Multimedia Retrieval (ICMR), pages 161–168, 2014.

M. Radovanović et al. On the existence of obstinate results in vector space models. In Proc. 33rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 186–193, 2010.

M. Radovanović et al. Time-series classification in many intrinsic dimensions. In Proc. 10th SIAM Int. Conf. on Data Mining (SDM), pages 677–688, 2010.

I. Karydis et al. Looking through the “glass ceiling”: A conceptual framework for the problems of spectral similarity. In Proc. 11th International Society for Music Information Retrieval Conference (ISMIR), pages 267–272, 2010.

A. Flexer et al. A MIREX meta-analysis of hubness in audio music similarity. In Proc. 13th International Society for Music Information Retrieval Conference (ISMIR), pages 175–180, 2012.

D. Schnitzer et al. The relation of hubs to the Doddington zoo in speaker verification. In Proc. 21st European Signal Processing Conf. (EUSIPCO), 2013.

N. Tomašev et al. Object recognition in WIKImage data based on local invariant image features. In Proc. 9th Int. Conf. on Intelligent Computer Communication and Processing (ICCP), pages 139–146, 2013.

N. Tomašev and D. Mladenić. Exploring the hubness-related properties of oceanographic sensor data. In Proc. 4th Int. Multiconf. on Information Society (IS), Volume A, pages 149–152, 2011.

T. Low et al. The hubness phenomenon: Fact or artifact? In C. Borgelt et al (eds.): Towards Advanced Data Analysis by Combining Soft Computing and Statistics, pages 267–278, Springer, 2013.

O. Chapelle et al., editors. Semi-Supervised Learning. MIT Press, 2006.

K. Ozaki et al. Using the mutual k-nearest neighbor graphs for semi-supervised classification of natural language data. In Proc. 15th Conf. on Computational Natural Language Learning (CoNLL), pages 154–162, 2011.

T. Jebara et al. Graph construction and b-matching for semi-supervised learning. In Proc. 26th Int. Conf. on Machine Learning (ICML), pages 441–448, 2009.

I. Suzuki et al. Investigating the effectiveness of Laplacian-based kernels in hub reduction. In Proc. 26th AAAI Conf. on Artificial Intelligence, pages 1112–1118, 2012.

D. Schnitzer et al. Local and global scaling reduce hubs in space. Journal of Machine Learning Research 13:2871–2902, 2012.

M. Lagrange et al. Cluster aware normalization for enhancing audio similarity. In Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), 2012.

A. Flexer and D. Schnitzer. Can shared nearest neighbors reduce hubness in high-dimensional spaces? In Proc. 1st International Workshop on High Dimensional Data Mining (HDM), 2013.

D. Schnitzer et al. A case for hubness removal in high-dimensional multimedia retrieval. In Proc. 36th European Information Retrieval Conference (ECIR), pages 687–692, 2014.

D. Schnitzer and A. Flexer. Choosing the metric in high–dimensional spaces based on hub analysis. In Proc. 22nd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), 2014.

E. Vincent et al. An investigation of likelihood normalization for robust ASR. In Proc. Interspeech, 2014.

D. A. Vega-Oliveros et al. Regular graph construction for semi-supervised learning. Journal of Physics: Conference Series, 490:012022, 2014

J. Murdock and L. S. Yaeger. Identifying species by genetic clustering. In Proc. 20th European Conf. on Artificial Life (ECAL), pages 565–572, 2011.

C. Van Parijs et al. Improving accuracy by reducing the importance of hubs in nearest neighbor recommendations. Louvain School of Management Working Paper 34/2013.

M. Lajoie et al. Computational discovery of regulatory elements in a continuous expression space. Genome Biology 13(11):R109, 2012.

J. Schlüter. Unsupervised Audio Feature Extraction for Music Similarity Estimation. MSc thesis, Faculty of Informatics, Technical University of Munich, Munich, Germany, 2011.

N. Tomašev et al. Hubness-based fuzzy measures for high-dimensional k-nearest neighbor classification. In Proc. 7th Int. Conf. on Machine Learning and Data Mining (MLDM), pages 16–30, New York, USA, 2011.

N. Tomašev et al. Hubness-based fuzzy measures for high-dimensional k-nearest neighbor classification. International Journal of Machine Learning and Cybernetics, 5(3):445–458, 2014.

N. Tomašev et al. A probabilistic approach to nearest-neighbor classification: Naive hubness Bayesian kNN. In Proc. 20th ACM Int. Conf. on Information and Knowledge Management (CIKM), pages 2173–2176, 2011.

N. Tomašev and D. Mladenić. Nearest neighbor voting in high dimensional data: Learning from past occurrences. Computer Science and Information Systems 9(2):691–712, 2012.

N. Tomašev and D. Mladenić. Hub co-occurrence modeling for robust high-dimensional kNN classification. In Proc. 24th European Conf. on Machine Learning (ECML), 2013.

N. Tomašev and D. Mladenić. Class imbalance and the curse of minority hubs. Knowledge-Based Systems, 53:157–172, 2013.

N. Tomašev and K. Buza. Hubness-aware kNN classification of high-dimensional data in presence of label noise. Neurocomputing, 2014 (to appear).

N. Tomašev et al. Hubness-based clustering of high-dimensional data. In M. E. Celebi, editor, Partitional Clustering Algorithms, Springer, 2014 (to appear).

N. Tomašev et al. The role of hubness in clustering high-dimensional data. In Proc. 15th Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD), Part I, pages 183–195, 2011.

N. Tomašev et al. The role of hubness in clustering high-dimensional data. IEEE Transactions on Knowledge and Data Engineering, 26(3):739–751, 2014.

G. Kouimtzis. Investigating the Impact of Hubness on SVM Classifiers. MSc thesis, Department of Information & Communication Systems Engineering, University of the Aegean, Karlovassi, Samos, Greece, 2011.

M. Lazaridis et al. Multimedia search and retrieval using multimodal annotation propagation and indexing techniques. Signal Processing: Image Communication, 28(4):351–367, 2013.

K. Buza et al. INSIGHT: Efficient and effective instance selection for time-series classification. In Proc. 15th Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD), Part II, pages 149–160, 2011.

N. Tomašev et al. Hubness-aware classification, instance selection and feature construction: Survey and extensions to time-series. In U. Stanczyk, L. Jain, editors, Feature Selection for Data and Pattern Recognition, Springer, 2014 (to appear).

K. Marussy. A new approach for more accurate semi-supervised time-series classification. TDK Paper, Budapest University of Technology and Economics, Faculty of Electrical Engineering and Informatics, Budapest, Hungary, 2012.

Z. Wang et al. Improved learning of I2C distance and accelerating the neighborhood search for image classification. Pattern Recognition, 44(10–11):2384–2394, 2011.

N. Tomašev et al. The role of hubs in supervised cross-lingual document retrieval. In Proc. 17th Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD), Part II, pages 185–196, 2013.

N. Tomašev et al. Exploiting hubs for self-adaptive secondary re-ranking in bug report duplicate detection. In Proc. 35th Int. Conf. on Information Technology Interfaces (ITI), 2013.

N. Tomašev and D. Mladenić. Hubness-aware shared neighbor distances for high-dimensional k-nearest neighbor classification. In Proc. 7th Int. Conf. on Hybrid Artificial Intelligence Systems (HAIS), Part II, pages 116–127, 2012.

N. Tomašev and D. Mladenić. Hubness-aware shared neighbor distances for high-dimensional k-nearest neighbor classification. Knowledge and Information Systems, 39(1):89–122, 2014.

J. Chen et al. Fast approximate kNN graph construction for high dimensional data via recursive Lanczos bisection. Journal of Machine Learning Research, 10:1989–2012, 2009.