The Effect of Data Centering on k-nearest neighbor

-Centering Similarity Measures-

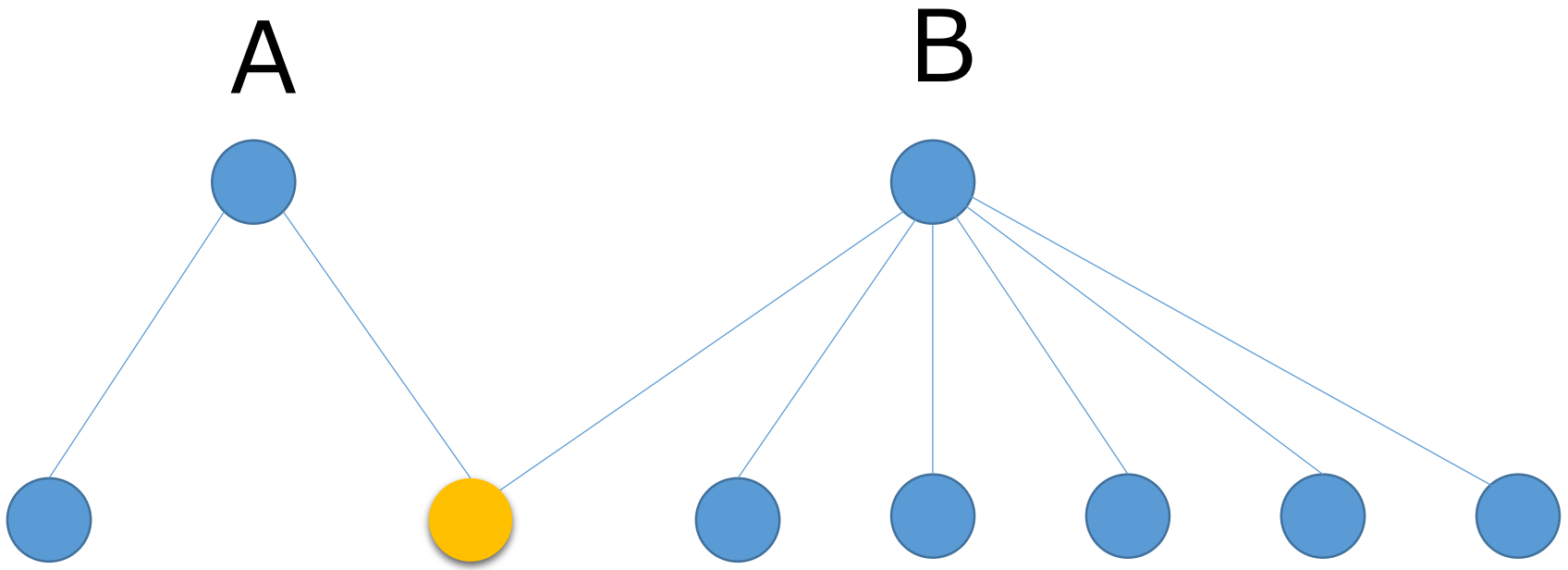
Ikumi Suzuki (鈴木郁美)
From National Institute of Genetics
Mishima, Shizuoka JAPAN

Outline

1. A lot of friends or a few friends?
   - **Laplacian based kernels** do not prefer common terms
2. k-nearest neighbor (kNN) problem in high dimension
   - **Hubness**
3. How to solve the hubness problem
   - Laplacian-based kernels from hubness point of view
   - Centering similarity measures to reduce hubs
4. Experiments
5. Theoretical analysis why centering reduce hubs

1. A lot of friends or a few friends?

A lot of friends or a few friends ?



Question: For , which is more important (A or B) ?

If the edge weights are given by

Cosine similarity	B > A : B is more important than A.
Laplacian based kernels	A > B : A is more important than B.

Small Example : Cosine similarity Prefers Common Terms

Common term

A) **Metric learning** is an important problem in data mining.
B) **High dimensional analysis** is common in data mining.
C) **Distance Optimization** in high dimensions is a challenging.

[Davis et al., 2009]

Because common terms appear in various contexts, their feature vectors become rich. Hence common terms have many chance to share features with other terms.
On the other hand, specific terms appear limited contexts, their feature vectors become sparse. Hence terms rarely share features with other terms.

Term	A	B	C
metric	1	0	0
learn	1	0	0
important	1	0	0
problem	1	1	0
data	1	1	0
mining	1	1	0
common	0	1	0
high	0	1	1
dimension	0	1	1
optimize	0	0	1
distance	0	0	1
challenge	0	0	1

Cosine similarity

	A	B	C
A	1	0.08	0
B	0.08	1	0.08
C	0	0.08	1

Common Terms or Specific Terms?

Thiazide : A class of diuretics, Antihypertensive drug

- [Specific Knowledge] If I were a hypertensive patient, I know this is an antihypertensive drug and look for other drugs for my treatment
  → Laplacian based Kernels ?
- [General Knowledge] If I were not a hypertensive patient, I just need to know Thiazide is an antihypertensive drug
  → Cosine Similarity

- Do Laplacian based Kernels prefer specific terms?

Experiment: Biomedical term synonym acquisition

A) Treatment is usually initiated with a **thiazide** type diuretic.
B) The most important types of **antihypertensives** include the ACE inhibitors, the calcium channel blockers.
C) **Norvasc** is a long-acting dihydropyridine type calcium channel blocker, which blocks the entry of calcium into muscle cells in artery walls.

[Task Objective] To acquire similar terms
Construct feature vectors for each term
Measure similarity between terms by
» Cosine similarity
» Laplacian based Kernels
[what to know] Do Laplacian based Kernels prefer specific terms?

Term	A	B	C
treatment	1	0	0
usually	1	0	0
:	:	:	:
type	1	1	1
ACE	0	1	0
calcium	0	1	1
channel	0	1	1
blocker	0	1	1
:	:	:	:
muscle	0	0	1
cell	0	0	1
artery	0	0	1
wall	0	0	1

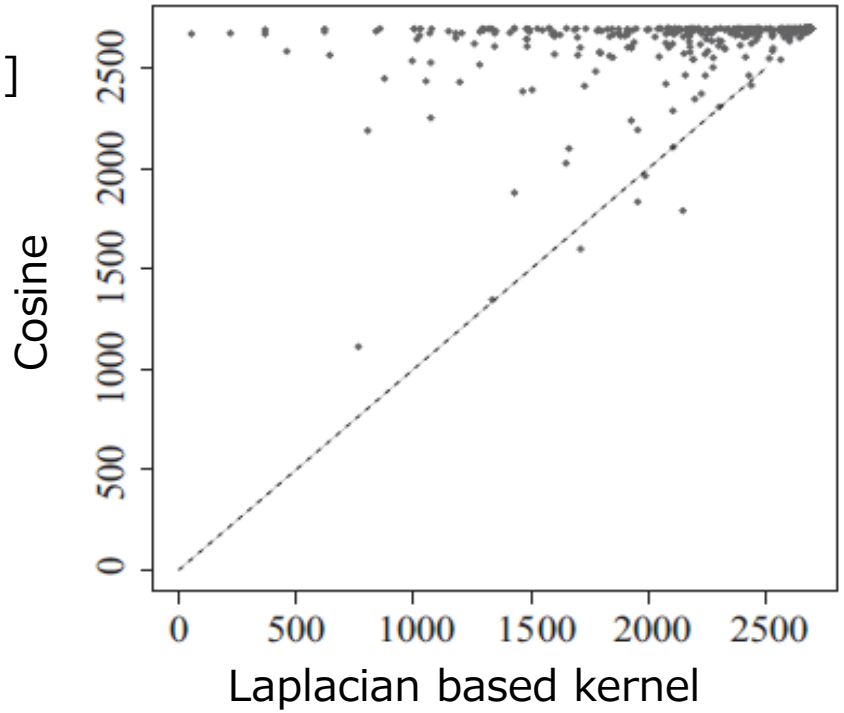
Laplacian-based kernel depreciate common terms

- For 2700 biomedical terms, features are constructed by contextual words in GENIA corpus
- Construct a graph in which a node represents a term and an edge weight is given by cosine similarity between terms.
- Compare the number of edges of the 1st ranked term for cosine and Laplacian based kernel

[Suzuki et al. DTMBIO 2009]

Query term: Human

	COS	Laplacian-based Kernels
1st	Cells	Human T-lymphotropic virus 1
2nd	Genes	Leukemia, t-cell
3rd	T-lymphocytes	Viruses



- Laplacian based kernels depreciate pivotal nodes (common terms). Why?

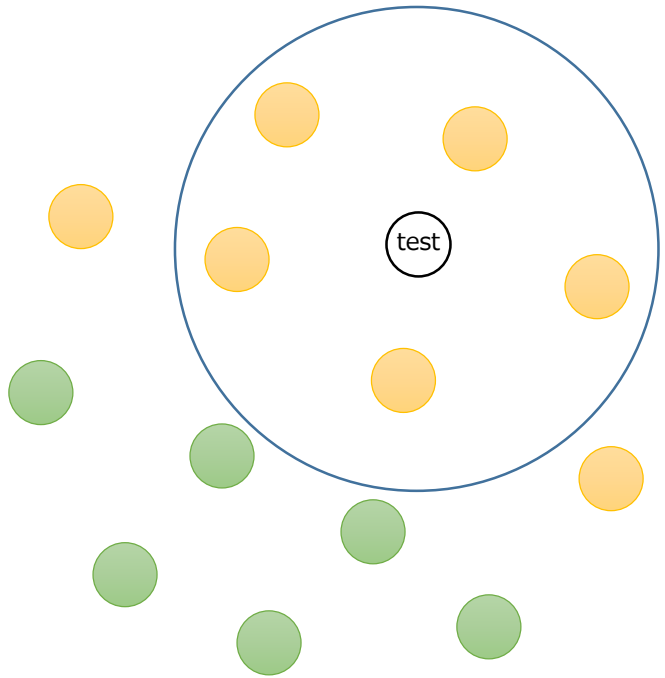
2. Hubness

Hubness Phenomena in High Dimensional Data

- Hubness [Radovanović et al. JMRL 2010]
  - When a sample is represented in a high dimensional feature space, hubness phenomena occurs.
  - A hub is a sample which is similar to many other samples in a dataset.
- Hub samples can affect k-nearest neighbor (kNN)

Classification based on kNN

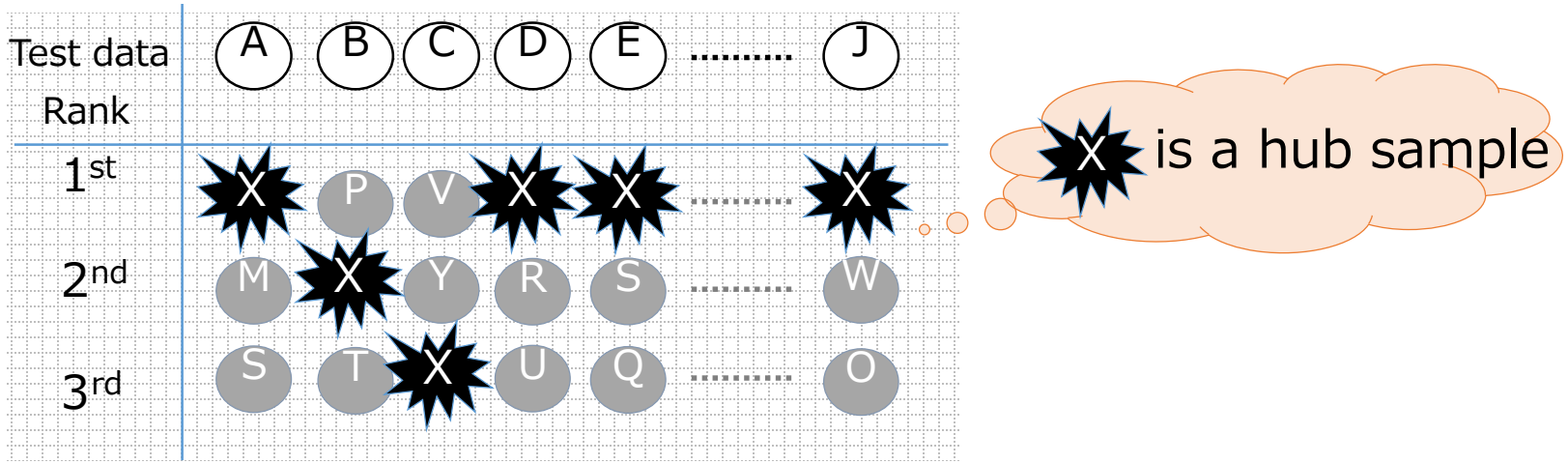
- A label of a test sample is predicted by labels of k training samples which are most similar to the test sample.



k-nearest neighbor (k=5)

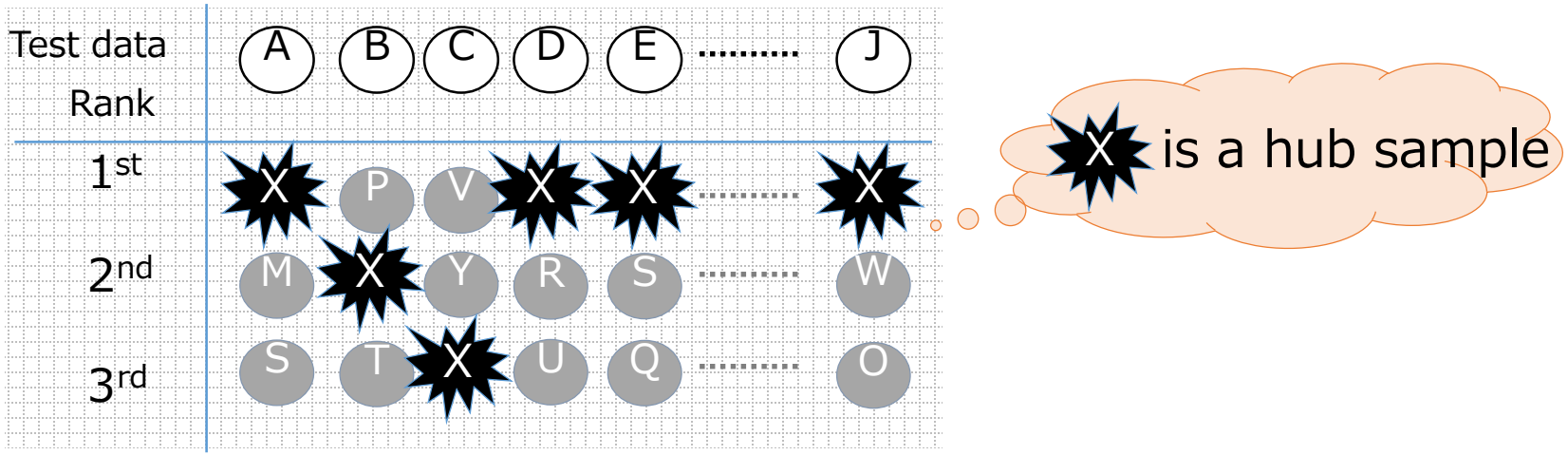
Hub Samples in kNN

- When hubs emerge in a dataset, kNN classification can be affected
  - A hub is a sample which appears in many other samples' kNN
- When the data dimension is high (even tens of dimensions), hubs emerge in a dataset



A Characteristics of Hub Samples

- **Samples which is most similar to the data centroid become hubs** [Radovanović et al. JMLR 2010]
- Hubs emerges in many other samples' kNN not because of label (classes) but because hubs are similar to the data centroid. Hubs can incur kNN based classification or information retrieval tasks.



Emergence of Hubs in Synthetic Data

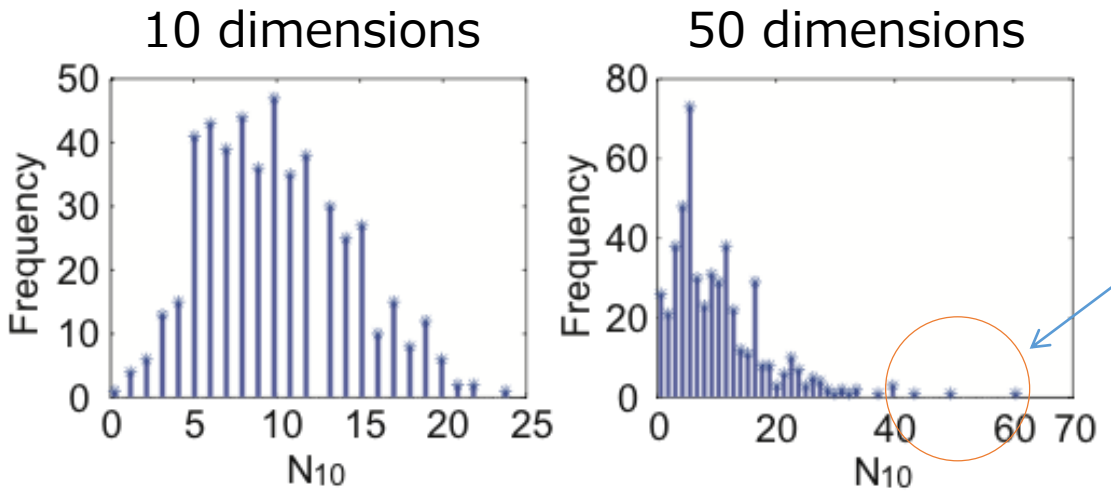
- Hub is a sample which is included in many other samples' kNN

[Characteristics of Hub samples]

- Hub emerges when the data dimension is high
- A sample which is similar to the data centroid tends to become hubs

Emergence of Hubs in Synthetic Data

- Synthetic dataset
  - 500 samples with 10 and 50 dimensions
  - Cosine similarity is used to measure similarity between samples
- Evaluate N_{10} value for each sample in a dataset
  - N_k is the number of times a sample appears in other samples' kNN
- N_k Value is large for hub samples



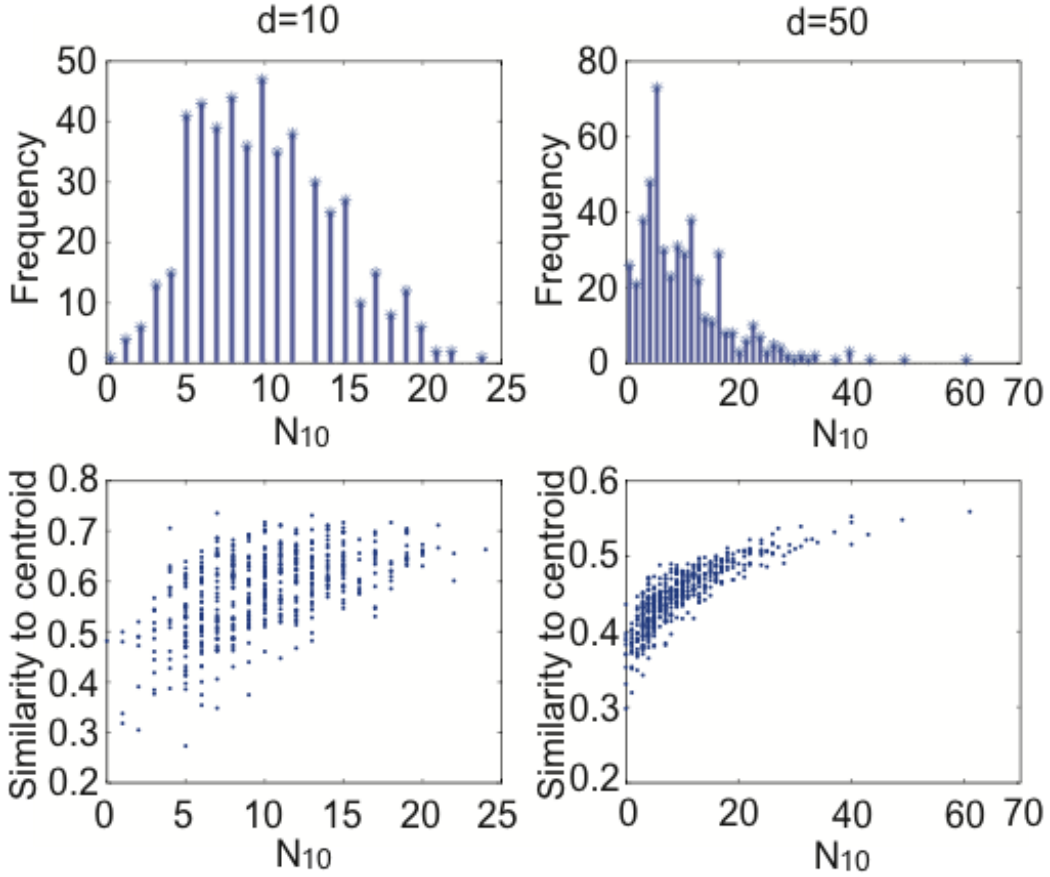
Many samples show small N_{10} value
A few sample(Hubs) shows large N_{10} value

Histogram of N_{10} values for 500 samples

Hubs and the Data Centroid

- A sample which is similar to the data centroid tends to become a hub

[Radovanović et al. JMLR 2010]



Samples with large N_{10} value (hubs) show more similar to the data centroid

N_{10}: Number of times a sample appears in other samples' 10NN

3.How to Solve the Hubness Problem?

Laplacian-based kernels [Suzuki et al. AAAI 2012]

How to tackle with hub samples?

- Any idea to reduce hubs?
- **[Fact]** [Radovanović et al. JMLR 2010]
  Samples which is similar to the data centroid become hubs

- **[Hypothesis]** [Suzuki et al. AAAI 2012]
  Can we reduce hubs if all samples have the same similarity to the data centroid ?
  - Laplacian based Kernels

Laplacian based kernels

- Suppose we have *N* samples with *M* dimensional feature vectors (vector length is normalized to 1) $\mathbf{x}_1, \cdots, \mathbf{x}_N \in \mathrm{R}^{\mathrm{M}}$
- $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ is the inner product between $\mathbf{x}_i$ and $\mathbf{x}_j$ (cosine similarity)
- The *ij*-th element of an adjacency matrix **A** is $\mathbf{A}_{ij} = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$
  - **A** is a cosine similarity matrix

- (Un-Normalized) Laplacian **L** $\mathbf{L} = \mathbf{D} - \mathbf{A}, \ \mathbf{D}_{ii} = \sum_{j=1}^{N} \mathbf{A}_{ij}$
  - Off-diagonal elements are same for **A** and **-L**
- Laplacian based kernels $\mathbf{K}^{\mathrm{Lap}}$
  - Regularized Laplacian $\mathbf{L}_{\mathrm{RL}}$ $\mathbf{L}_{\mathrm{RL}} = (\mathbf{I} + \beta\mathbf{L})^{-1} = \mathbf{I} + \beta(-\mathbf{L}) + \beta^2(-\mathbf{L})^2 + \cdots$
  - Commute Time Kernel $\mathbf{L}_{\mathrm{CT}}$ $\mathbf{L}_{\mathrm{CT}} = \mathbf{L}^{+}\ (\mathrm{pseudo\text{-}inverse})$

Laplacian based kernels Make All Samples Equally Similar to the Data Centroid

- A characteristics of hub samples
  - Samples which show more similar to the data centroid become hubs

- Why Laplacian based kernels can reduce hubs?

  - Regularized Laplacian $\mathbf{L}_{RL}$
    makes all samples equally similar (1) to the data centroid

  - Commute-time Kernel $\mathbf{L}_{CT}$
    makes all samples equally similar (0) to the data centroid

Similarity to the Data Centroid

- Dataset D contains *N* samples which are represented *M* dimensional feature vectors $\mathbf{x}_1, \cdots, \mathbf{x}_N \in \mathrm{R}^M$
- The centroid vector $\bar{\mathbf{x}} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n \in \mathrm{R}^M$
- Similarity of *i*-th sample to the data centroid $\langle \mathbf{x}_i, \bar{\mathbf{x}} \rangle = \langle \mathbf{x}_i, \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n \rangle = \frac{1}{N}\sum_{n=1}^{N} \langle \mathbf{x}_i, \mathbf{x}_n \rangle$

- The similarity of *i*-th sample and the data centroid is obtained by summing up the all elements of the similarity matrix **K** of *i*-th row

$$\begin{aligned} \left\langle \mathbf{x}_i, \bar{\mathbf{x}} \right\rangle &= \left\langle \mathbf{x}_i, \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n \right\rangle \\ &= \frac{1}{N}\sum_{n=1}^{N} \left\langle \mathbf{x}_i, \mathbf{x}_n \right\rangle \\ &= \frac{1}{N}\sum_{n=1}^{N} \mathbf{K}_{in} \end{aligned}$$

$$\begin{array}{c} \\ 1 \\ \vdots \\ i \\ \vdots \\ N \end{array} \begin{array}{c} \begin{array}{ccc} 1 & \cdots & N \end{array} \\ \begin{pmatrix} \langle \mathbf{x}_1, \mathbf{x}_1 \rangle & & \langle \mathbf{x}_1, \mathbf{x}_N \rangle \\ \vdots & & \vdots \\ \langle \mathbf{x}_i, \mathbf{x}_1 \rangle & \mathbf{K} & \langle \mathbf{x}_i, \mathbf{x}_N \rangle \\ \vdots & & \vdots \\ \langle \mathbf{x}_N, \mathbf{x}_1 \rangle & & \langle \mathbf{x}_N, \mathbf{x}_N \rangle \end{pmatrix} \end{array} \begin{pmatrix} \mathbf{1} \\ \mathbf{1} \\ \vdots \\ \mathbf{1} \\ \mathbf{1} \end{pmatrix} = \begin{pmatrix} \langle \mathbf{x}_1, \bar{\mathbf{x}} \rangle \\ \vdots \\ \langle \mathbf{x}_i, \bar{\mathbf{x}} \rangle \\ \vdots \\ \langle \mathbf{x}_N, \bar{\mathbf{x}} \rangle \end{pmatrix} \times N$$

Laplacian based kernels and the similarity to the data centroid

- Use smallest eigenvalue and its eigenvector
  - The Laplacian matrix **L** has all-ones vector $\mathbf{u}_1 = [1,\ldots,1]^{\mathrm{T}}$ as its eigenvector for the smallest eigenvalue ($\lambda_1 = 0$).

  - Laplacian based kernels also have all-ones vector $\mathbf{u}_1 = [1,\ldots,1]^{\mathrm{T}}$ as their eigenvectors. The corresponding eigenvalues are
    - Regularized Laplacian $\mathbf{L}_{\mathrm{RL}}$ $r(\lambda_i) = \frac{1}{1+\beta\lambda_i}$, $\frac{1}{1+\beta\lambda_1} = 1$
    - Commute-time kernel $\mathbf{L}_{\mathrm{CT}}$ $r(\lambda_i) = \frac{1}{\lambda_i}$, $\frac{1}{\lambda_1} \equiv 0$

  - Similarity to the data centroid

$$\mathbf{L}_{\mathrm{RL}}\mathbf{u}_1 = \frac{1}{1+\beta\lambda_1}\mathbf{u}_1 = [1,\ldots,1]^{\mathrm{T}}$$

$$\mathbf{L}_{\mathrm{CT}}\mathbf{u}_1 = \frac{1}{\lambda_1}\mathbf{u}_1 = [0,\ldots,0]^{\mathrm{T}}$$

$$\begin{array}{c} \\ 1 \\ \vdots \\ i \\ \vdots \\ N \end{array}\!\!\begin{array}{c} \begin{array}{ccc} 1 & \cdots & N \end{array} \\ \begin{pmatrix} \langle \mathbf{x}_1, \mathbf{x}_1 \rangle & & \langle \mathbf{x}_1, \mathbf{x}_N \rangle \\ \vdots & & \vdots \\ \langle \mathbf{x}_i, \mathbf{x}_1 \rangle & \mathbf{K} & \langle \mathbf{x}_i, \mathbf{x}_N \rangle \\ \vdots & & \vdots \\ \langle \mathbf{x}_N, \mathbf{x}_1 \rangle & & \langle \mathbf{x}_N, \mathbf{x}_N \rangle \end{pmatrix} \end{array} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} \langle \mathbf{x}_1, \bar{\mathbf{x}} \rangle \\ \vdots \\ \langle \mathbf{x}_i, \bar{\mathbf{x}} \rangle \\ \vdots \\ \langle \mathbf{x}_N, \bar{\mathbf{x}} \rangle \end{pmatrix} \times N$$

3.How to Solve the Hubness Problem?

Data Centering [Suzuki et al. EMNLP 2013]

Centering Similarity Measures

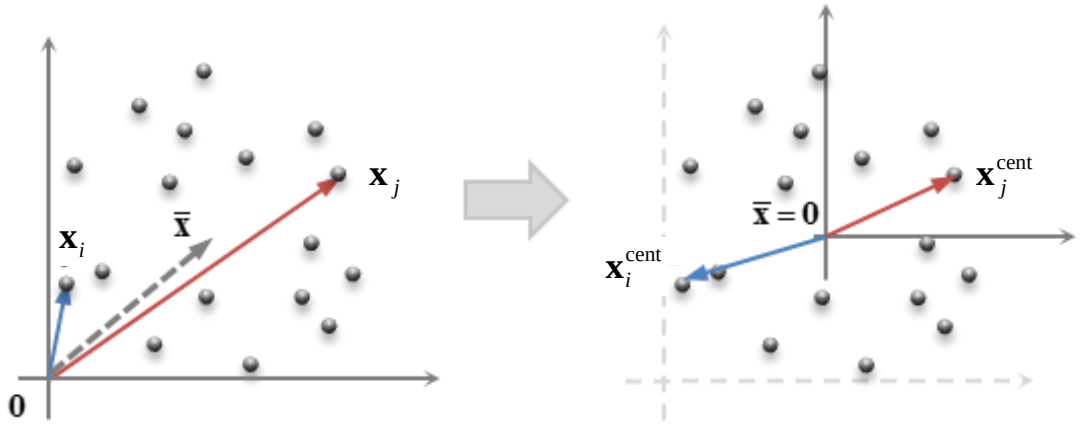
- Laplacian based kernels make all samples in a dataset equally similar to the data centroid
- Is there another way to make all samples equally similar to the data centroid?
- Data centering
  - Shift the origin to the data centroid

Data Centering

- Dataset D contains *N* samples with *M* dimensional feature vectors

$$\mathbf{x}_1, \cdots, \mathbf{x}_N \in \mathrm{R}^{\mathrm{M}}$$

- The centroid vector $\bar{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{x}_n \in \mathrm{R}^{\mathrm{M}}$



- Centering each feature vector
  - $\mathbf{x}_i^{\mathrm{cent}} = \mathbf{x}_i - \bar{\mathbf{x}}$: Shift the origin to the data centroid
- Centered Similarity between *i-j* th sample is given by the inner product between centered *i* and *j*-th samples

$$\left\langle \mathbf{x}_i^{\mathrm{cent}} \mathbf{x}_j^{\mathrm{cent}} \right\rangle = \left\langle \mathbf{x}_i - \bar{\mathbf{x}}, \mathbf{x}_j - \bar{\mathbf{x}} \right\rangle$$

Centering Similarity Matrix

- When n x n similarity matrix **K** is given, the centered similarity matrix $\mathbf{K}^{\text{cent}}$ is

$$\mathbf{K}^{\text{cent}} = \mathbf{Z}^{\mathrm{T}}\mathbf{K}\mathbf{Z}$$

$$\mathbf{Z} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^{\mathrm{T}} = \frac{1}{n}\begin{bmatrix} n-1 & -1 & \cdots & -1 \\ -1 & n-1 & & \\ & & \ddots & \\ -1 & -1 & \cdots & n-1 \end{bmatrix}$$

$\mathbf{1}$ is a vector whose elements are all 1

Why Centered Similarity Matrix Reduce Hubs?

- Centered similarity matrix $\mathbf{K}^{\mathrm{cent}}$ has the eigenvector $\mathbf{1}=[1,\cdots,1]^{\mathrm{T}}$ and the corresponding eigenvalue is 0. Therefore, similarity to the data centroid becomes equally 0 for all samples by centering.

$$\mathbf{K}^{\mathrm{cent}}\begin{pmatrix}\langle \mathbf{x}_1^{\mathrm{cent}},\mathbf{x}_1^{\mathrm{cent}}\rangle & & \langle \mathbf{x}_1^{\mathrm{cent}},\mathbf{x}_N^{\mathrm{cent}}\rangle \\ \vdots & & \vdots \\ \langle \mathbf{x}_i^{\mathrm{cent}},\mathbf{x}_1^{\mathrm{cent}}\rangle & \cdots & \langle \mathbf{x}_i^{\mathrm{cent}},\mathbf{x}_N^{\mathrm{cent}}\rangle \\ \vdots & & \vdots \\ \langle \mathbf{x}_N^{\mathrm{cent}},\mathbf{x}_1^{\mathrm{cent}}\rangle & & \langle \mathbf{x}_N^{\mathrm{cent}},\mathbf{x}_N^{\mathrm{cent}}\rangle \end{pmatrix}\begin{pmatrix}\mathbf{1}\\ \vdots \\ \mathbf{1}\\ \vdots \\ \mathbf{1}\end{pmatrix} = \begin{pmatrix}\langle \mathbf{x}_1^{\mathrm{cent}},\bar{\mathbf{x}}^{\mathrm{cent}}\rangle \\ \vdots \\ \langle \mathbf{x}_i^{\mathrm{cent}},\bar{\mathbf{x}}^{\mathrm{cent}}\rangle \\ \vdots \\ \langle \mathbf{x}_N^{\mathrm{cent}},\bar{\mathbf{x}}^{\mathrm{cent}}\rangle \end{pmatrix} \times N = \mathbf{O}\times\begin{pmatrix}\mathbf{1}\\ \vdots \\ \mathbf{1}\\ \vdots \\ \mathbf{1}\end{pmatrix} = \begin{pmatrix}\mathbf{O}\\ \vdots \\ \mathbf{O}\\ \vdots \\ \mathbf{O}\end{pmatrix}$$

Laplacian based Kernels is Centered

- Laplacian based kernels can be considered as centered in the kernel space

- Let $\boldsymbol{\phi}_i$ be the i-th feature vector in the kernel space. The sample-feature matrix **X** is

Feature

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{\phi}_1^{\mathrm{T}} \\ \boldsymbol{\phi}_2^{\mathrm{T}} \\ \vdots \\ \boldsymbol{\phi}_N^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} \sqrt{r(\lambda_1)}\mathbf{u}_1, & \cdots & , \sqrt{r(\lambda_N)}\mathbf{u}_N \end{bmatrix}$$

Sample

- The Laplacian based Kernel $\mathbf{K}^{\mathrm{Lap}}$ is

$$\mathbf{K}^{\mathrm{Lap}} = \sum_{i=1}^{N} r(\lambda_i)\mathbf{u}_i\mathbf{u}_i^{\mathrm{T}} = \begin{bmatrix} \sqrt{r(\lambda_1)}\mathbf{u}_1 & \cdots & \sqrt{r(\lambda_N)}\mathbf{u}_N \end{bmatrix} \begin{bmatrix} \sqrt{r(\lambda_1)}\mathbf{u}_1^{\mathrm{T}} \\ \vdots \\ \sqrt{r(\lambda_N)}\mathbf{u}_N^{\mathrm{T}} \end{bmatrix} = \mathbf{X}\mathbf{X}^{\mathrm{T}}$$

- Let eigenvalues and eigenvectors of Laplacian **L** denoted as

$$\lambda_1 = 0 < \lambda_2, \cdots, \lambda_N \qquad \mathbf{u}_1 = [1, \ldots, 1]^{\mathrm{T}}, \mathbf{u}_2, \cdots, \mathbf{u}_N$$

- $\boldsymbol{\phi}_i$ is the collection of the i-th element of each eigenvector

$$\boldsymbol{\phi}_i = \left[\sqrt{r(\lambda_1)}\mathbf{u}_1^i, \sqrt{r(\lambda_2)}\mathbf{u}_2^i, \cdots, \sqrt{r(\lambda_N)}\mathbf{u}_N^i\right]$$

- Sum of the all elements in each eigenvector $\mathbf{u}_i$ is 0

$$\sum_{j=1}^{N} \mathbf{u}_i^j = 0 \quad \text{for } i \geq 2, \quad \text{since} \quad \mathbf{u}_i^{\mathrm{T}}\mathbf{1} = 0 \quad (\mathbf{u}_1 = \mathbf{1})$$

- $\boldsymbol{\Phi}_i$ is centered feature vector

$$\overline{\boldsymbol{\phi}} = \sum_{i=1}^{N} \boldsymbol{\phi}_i^{\mathrm{T}} = 0$$

Laplacian based Kernels

- Eigenvalues and eigenvectors of Laplacian **L**

$$\lambda_1 = 0 < \lambda_2, \cdots, \lambda_N \quad \mathbf{u}_1 = [1, \ldots, 1]^{\mathrm{T}}, \mathbf{u}_2, \cdots, \mathbf{u}_N$$

- Laplacian based kernels $\mathbf{K}^{\mathrm{Lap}}$ have the same eigenvectors

$$\mathbf{u}_1 = [1, \ldots, 1]^{\mathrm{T}}, \mathbf{u}_2, \cdots, \mathbf{u}_N$$

  but the eigenvalues are regulated. Let $r(\lambda_i)$ be the eigenvalue function then,

  - Regularized Laplacian $\mathbf{L}_{\mathrm{RL}}$

$$r(\lambda_i) = \frac{1}{1+\beta\lambda_i}$$

  - Commute-Time Kernel $\mathbf{L}_{\mathrm{CT}}$

$$r(\lambda_i) = \begin{cases} 0 & i = 1 \\ \frac{1}{\lambda_i} & i \geq 2 \end{cases}$$

- $\mathbf{K}^{\mathrm{Lap}}$ can be expressed by eigenvalues and eigenvectors

$$\mathbf{K}^{\mathrm{Lap}} = \sum_{i=1}^{N} r(\lambda_i)\mathbf{u}_i\mathbf{u}_i^{\mathrm{T}}$$

Summary

- **[Background]** Samples which are similar to the data centroid tend to become hubs.
- **[What I showed]**
  - Laplacian based kernels
  - Centering similarity measures

  make all samples equally similar to the data centroid. Therefore, they are expected to reduce hubs.

4.Experiments

Can hubs be reduced by making all samples equally similar to the data centroid?

Experiment

- **[Purpose]**
  - Do Laplacian based kernels and centering similarity measures reduce hubs in real dataset and improve kNN-based classification performances?

Dataset	#classes	#objects	#features
Reuters Transcribed	10	201	2730
Mini Newsgroups	20	2000	8811

- **[Task]**
  - Multiclass document classification
  - For each document, a feature vector is constructed by using word frequency occurred in the document. The vector length is normalized to 1 after tf-idf weighting.
  - The class of a test sample is predicted by the majority vote from k=10 most similar samples to the test sample.
  - To measure similarity between samples, I set several options;

Experiment –Similarity Measures-

- **[Purpose]**
  - Do Laplacian based kernels and centering similarity measures reduce hubs in real dataset and improve performances?
- **[Similarity Measures to Compare]**
  - Cosine similarity is a baseline similarity measure and given as the elements of adjacency matrix **A**.

Hubs exist?

$\mathbf{A}$ is a cosine similarity matrix, $\mathbf{A}_{ij} = \left\langle \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|}, \frac{\mathbf{x}_j}{\|\mathbf{x}_j\|} \right\rangle$

Reduce hubs?

$\mathbf{A}^{\text{cent}}$ is a centered cosine similarity matrix, $\mathbf{A}^{\text{cent}} = (\mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}})^{\mathrm{T}}\,\mathbf{A}(\mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}})$

$\mathbf{L}$ is a Graph - Laplacian matrix, $\mathbf{L} = \mathbf{D} - \mathbf{A}, \quad \mathbf{D} = \mathrm{diag}(\mathbf{A}\mathbf{1})$

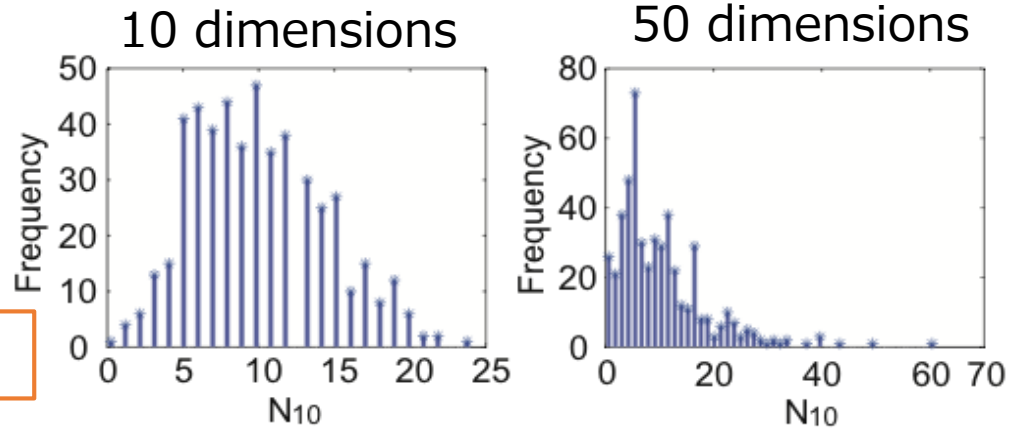
$\mathbf{L}_{\mathrm{RL}}$ is a Regularized Laplacian, $\mathbf{L}_{\mathrm{RL}} = (\mathbf{I} + \beta\mathbf{L})^{-1}$

$\mathbf{L}_{\mathrm{CT}}$ is a Commute - Time Kernel, $\mathbf{L}_{\mathrm{CT}} = \mathbf{L}^{+}$ (pseudo inverse of $\mathbf{L}$)

Experiment -Evaluation Value for Hubness-

- **[Purpose]**
  - Do Laplacian based kernels and centering similarity measures reduce hubs in real dataset and improve performances?
- **[Evaluation Value for Hubness]**
  - To evaluate how much hubs appears in a dataset, the emergence of hubs can be quantified by the skewness of the distribution of the N_{10} (the number of times a sample appears in the kNN of other samples).

$$\text{skewness} = \frac{\mathrm{E}[N_{10} - \mu]^3}{\sigma^3}$$

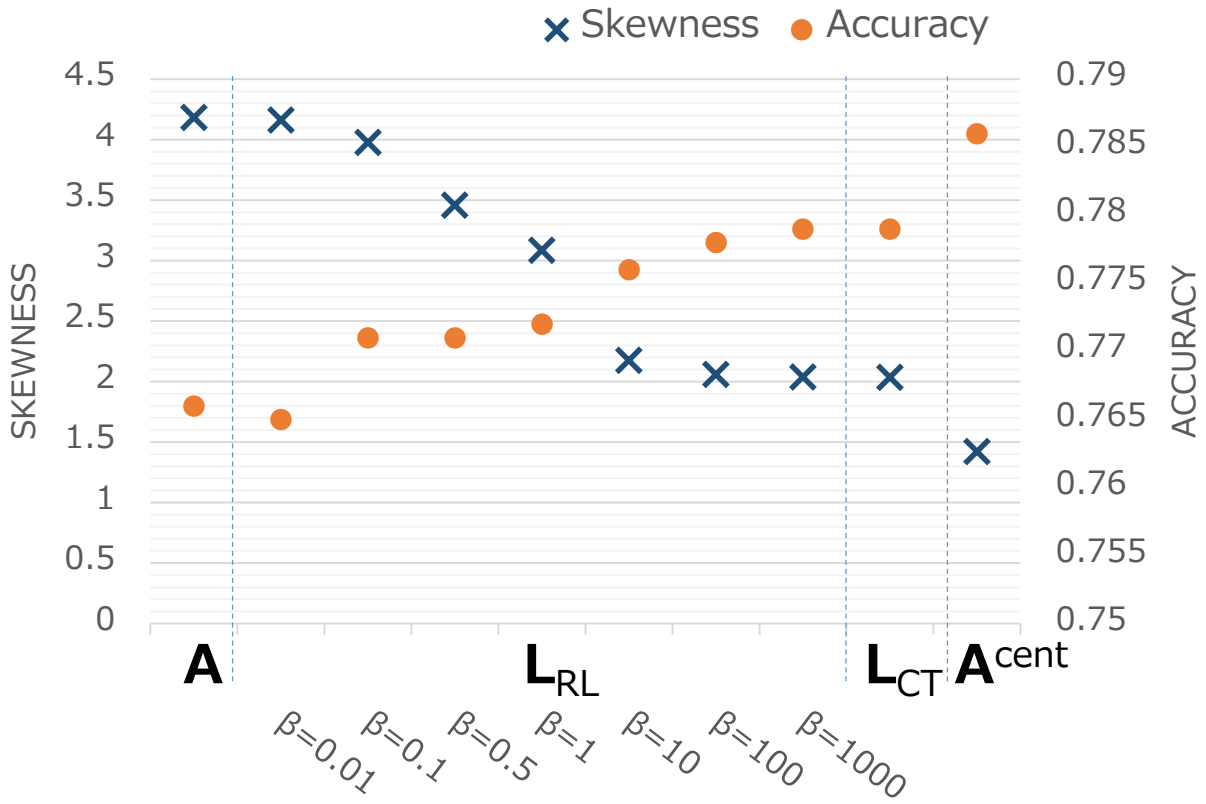


Skewness: small

Skewness: large

Result : Mini-newsgroup

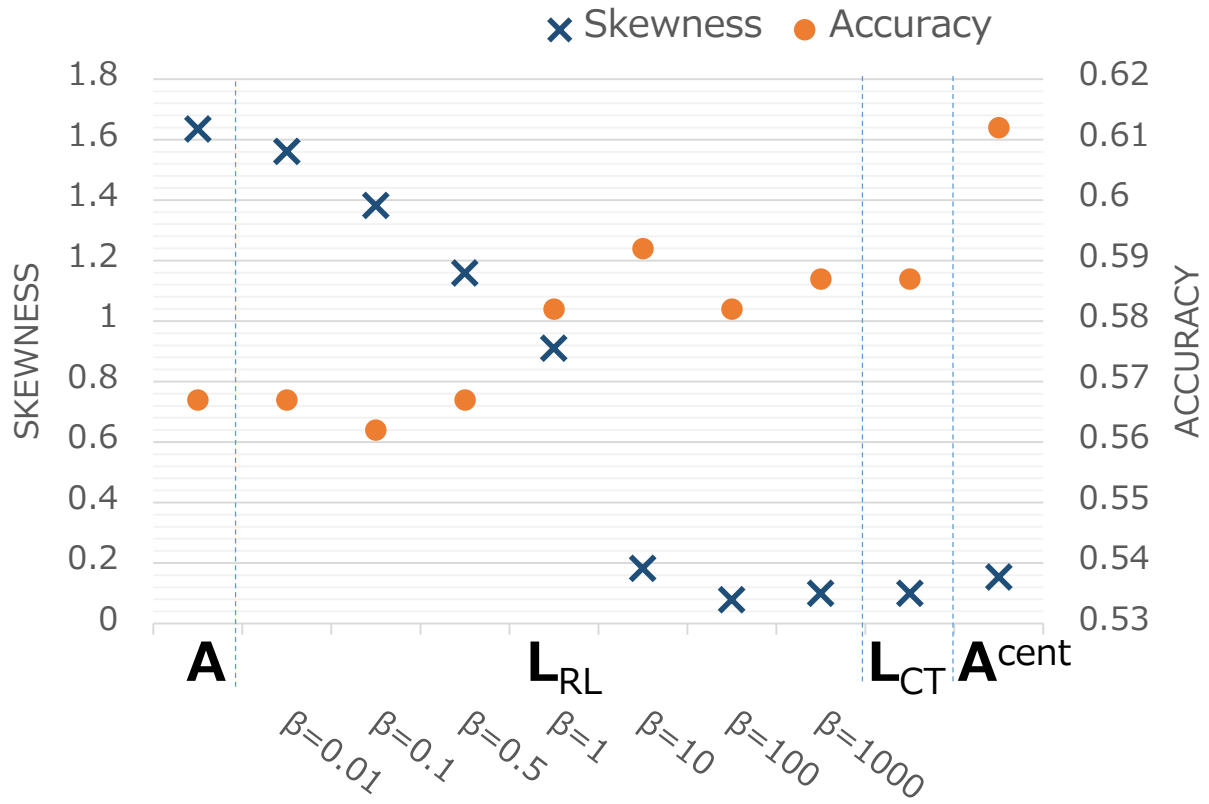
A : cosine
A^{cent} : centered cosine
$\mathbf{L}_{RL}$: Regularized Laplacian
$\mathbf{L}_{CT}$: Commute-Time Kernel



	A	**$\mathbf{L}_{RL}$**							**$\mathbf{L}_{CT}$**	**A**^{cent}
		β=0.01	β=0.1	β=0.5	β=1	β=10	β=100	β=1000		
Skewness	4.1824	4.1619	3.9762	3.458	3.0849	2.1762	2.0605	2.0373	2.0356	1.4231
Accuracy	0.766	0.765	0.771	0.771	0.772	0.776	0.778	0.779	0.779	0.786

Result : Reuters-Transcribed

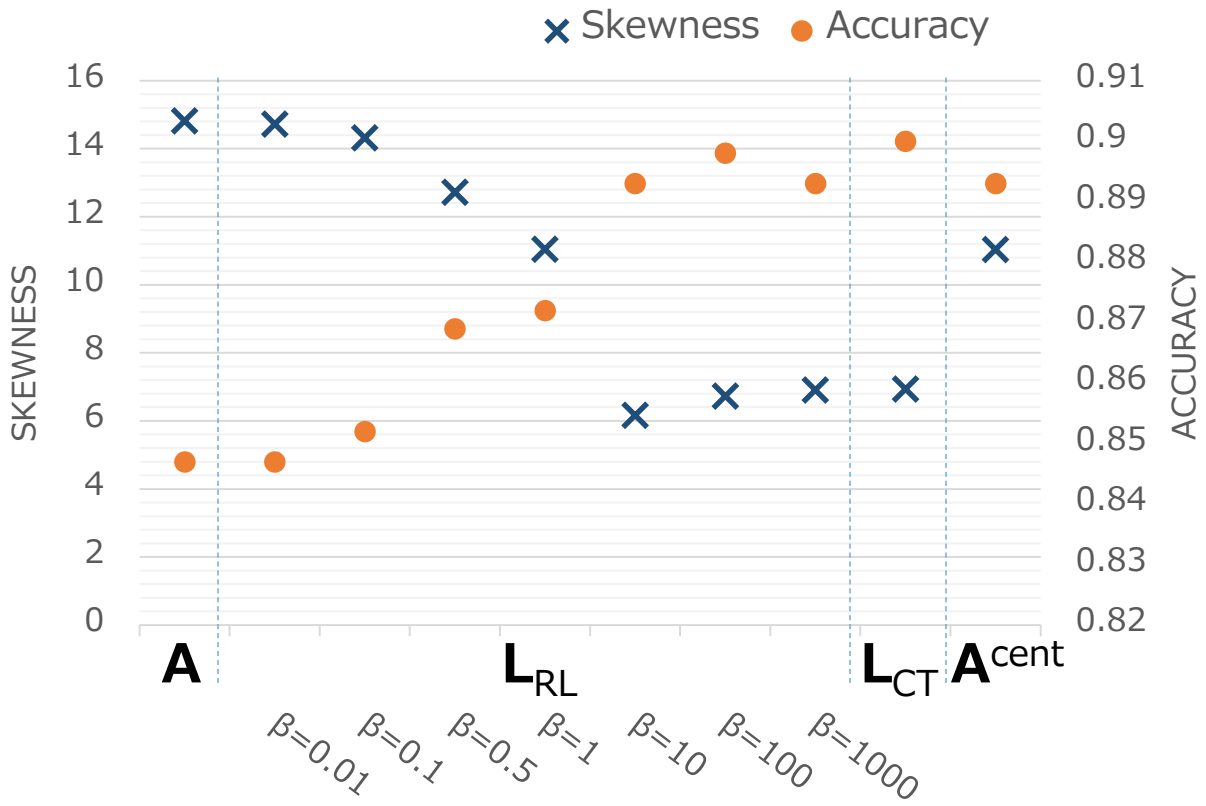
A : cosine
A^{cent} : centered cosine
L$_{RL}$: Regularized Laplacian
L$_{CT}$: Commute-Time Kernel



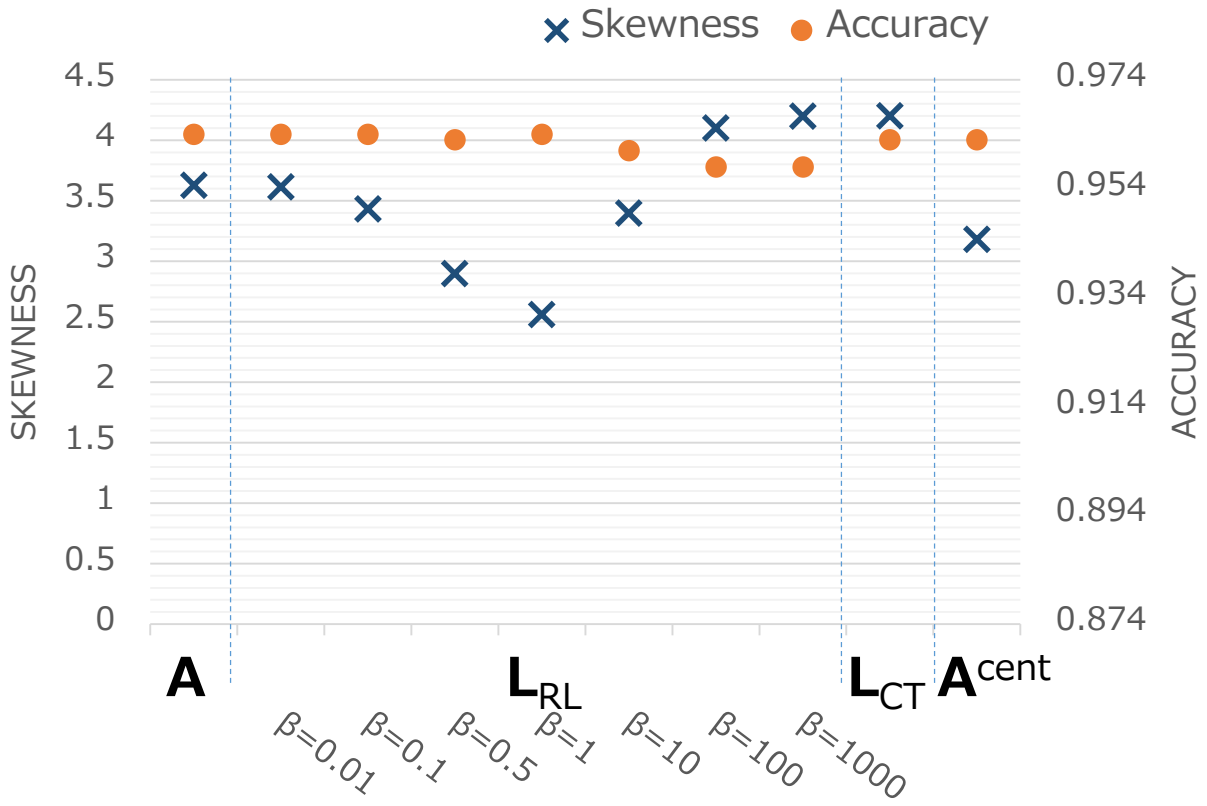
	A	**L**$_{RL}$							**L**$_{CT}$	**A**^{cent}
		β=0.01	β=0.1	β=0.5	β=1	β=10	β=100	β=1000		
Skewness	1.6354	1.5612	1.3819	1.1596	0.91056	0.18239	0.078386	0.099731	0.099731	0.15368
Accuracy	0.567	0.567	0.562	0.567	0.582	0.592	0.582	0.587	0.587	0.612

Results –other datasets-

Reuters 52



TDT2 30



Summary from Experiments

- **[What I did]** To examine if Laplacian based kernels and centering similarity measure reduce hubs in real datasets.
- **[What I found]**
  - Centering similarity measure tend to work well to reduce hubs. But when the feature is sparse, it does not make much change from non-centered similarity measures.
  - For Laplacian-based kernels, the reduction of hubs depends on the parameter (Regularized Laplacian).

5.Theoretical Analysis Why Centering Reduce Hubs

[Suzuki et al. EMNLP 2013]

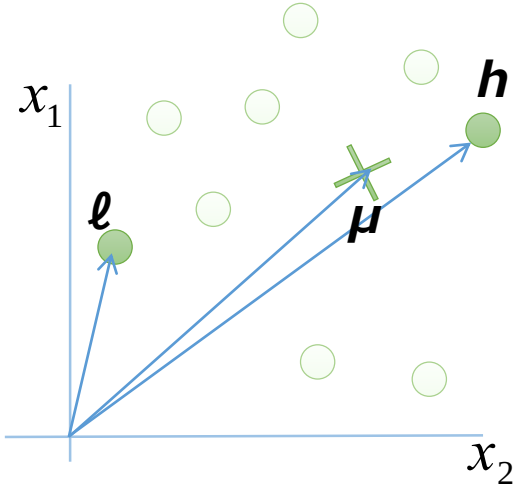
Theoretical Analysis Why Centering Reduce Hubs

- Dataset $\{\mathbf{x}_1,\cdots,\mathbf{x}_N\}\in\mathbb{R}^M$ are generated from a distribution $P(\mathbf{x})$ with mean $\boldsymbol{\mu}$
- Choose two samples $\boldsymbol{h}$ and $\boldsymbol{\ell}$

  $\boldsymbol{h}$ is more similar to the mean $\boldsymbol{\mu}$ than $\boldsymbol{\ell}$

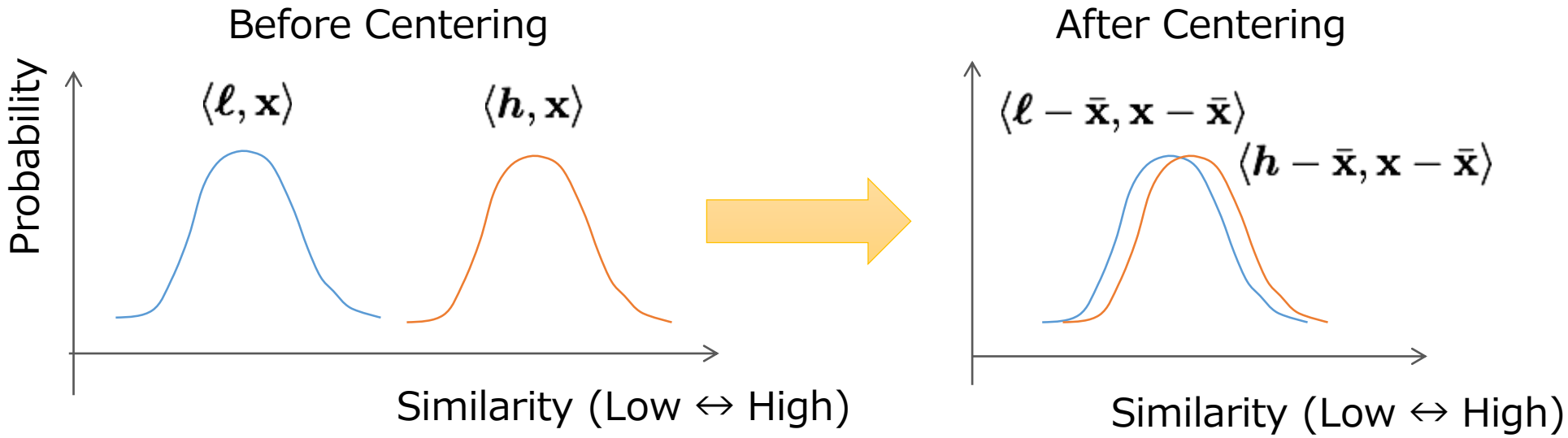
$$\langle \boldsymbol{h}, \boldsymbol{\mu} \rangle > \langle \boldsymbol{\ell}, \boldsymbol{\mu} \rangle$$

- Examine how the similarity distribution of $\boldsymbol{h}$ and $\boldsymbol{\ell}$ with other samples($\mathbf{x}$) changes by centering

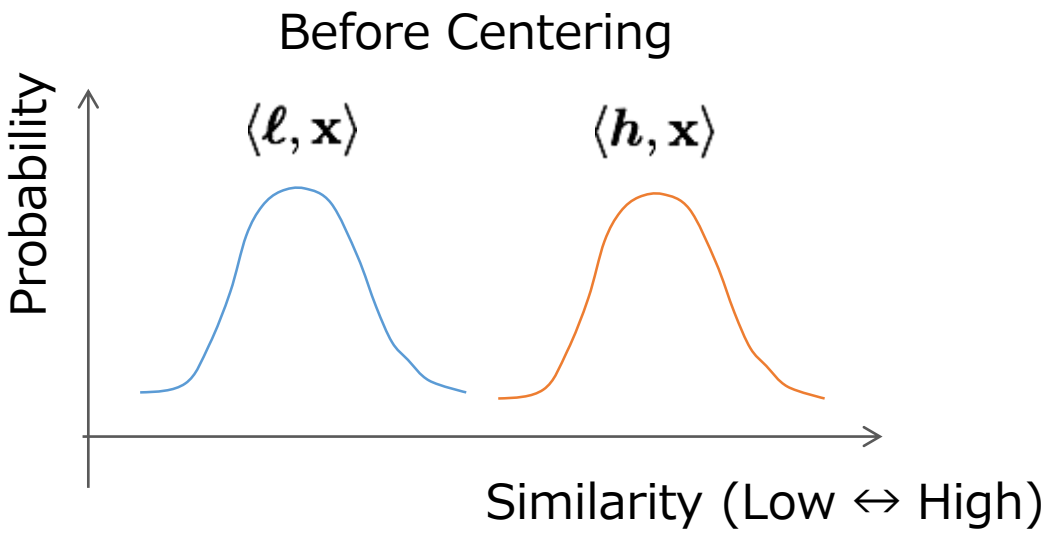


Overview

- Compare the similarity distribution of $\boldsymbol{h}$ and $\boldsymbol{\ell}$ before and after centering



Before Centering : A sample which is more similar to the data centroid ($\boldsymbol{h}$) is more similar with other samples ($\mathbf{x}$) in a dataset than the one less similar to the data centroid ($\boldsymbol{\ell}$)
After Centering : There is no difference in similarity distribution between $\boldsymbol{h}$ and $\boldsymbol{\ell}$



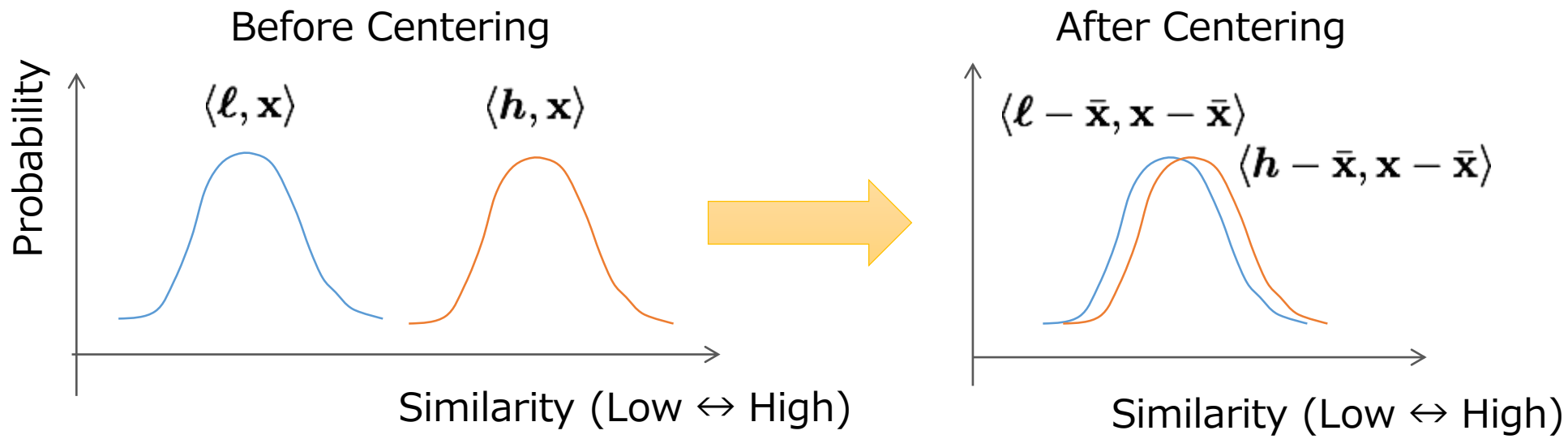
① Select two points $\boldsymbol{h}$ and ℓ, such that $\langle \boldsymbol{h}, \mu \rangle > \langle \ell, \mu \rangle$

Then, compare similarity (inner product) for the selected samples ($\boldsymbol{h}$ and ℓ) with other samples $\mathbf{x}$.

② Before Centering : How different is the mean of the distribution $\langle \boldsymbol{h}, \mathbf{x} \rangle$ and $\langle \ell, \mathbf{x} \rangle$?

$$\mathrm{E}\left[\langle \boldsymbol{h}, \mathbf{x} \rangle\right] - \mathrm{E}\left[\langle \ell, \mathbf{x} \rangle\right] = \langle \boldsymbol{h}, \mathrm{E}[\mathbf{x}] \rangle - \langle \ell, \mathrm{E}[\mathbf{x}] \rangle = \langle \boldsymbol{h}, \boldsymbol{\mu} \rangle - \langle \ell, \boldsymbol{\mu} \rangle > 0$$

$\therefore$ The mean of two distributions are different : $\mathrm{E}\left[\langle \boldsymbol{h}, \mathbf{x} \rangle\right] > \mathrm{E}\left[\langle \ell, \mathbf{x} \rangle\right]$



① Select two points $\boldsymbol{h}$ and ℓ, such that $\langle \boldsymbol{h}, \mu \rangle > \langle \ell, \mu \rangle$

Then, compare similarity (inner product) for the selected samples ($\boldsymbol{h}$ and ℓ) with other samples $\mathbf{x}$.

② Before Centering : How different is the mean of the distribution $\langle \boldsymbol{h}, \mathbf{x} \rangle$ and $\langle \ell, \mathbf{x} \rangle$?

$$\mathrm{E}\left[\langle \boldsymbol{h}, \mathbf{x} \rangle\right] - \mathrm{E}\left[\langle \ell, \mathbf{x} \rangle\right] = \langle \boldsymbol{h}, \mathrm{E}[\mathbf{x}] \rangle - \langle \ell, \mathrm{E}[\mathbf{x}] \rangle = \langle \boldsymbol{h}, \mu \rangle - \langle \ell, \mu \rangle > 0$$

$\therefore$ The mean of two distributions are different : $\mathrm{E}\left[\langle \boldsymbol{h}, \mathbf{x} \rangle\right] > \mathrm{E}\left[\langle \ell, \mathbf{x} \rangle\right]$

③ After Centering : What become of the mean difference of the distribution $\langle \boldsymbol{h} - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle$ and $\langle \ell - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle$?

$$\langle \boldsymbol{h}^{\mathrm{cent}}, \mathbf{x}^{\mathrm{cent}} \rangle = \langle \boldsymbol{h} - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle = \langle \boldsymbol{h}, \mathbf{x} \rangle - \langle \boldsymbol{h}, \bar{\mathbf{x}} \rangle - \langle \mathbf{x}, \bar{\mathbf{x}} \rangle + \|\mathbf{x}\|^2 \qquad \langle \ell^{\mathrm{cent}}, \mathbf{x}^{\mathrm{cent}} \rangle = \langle \ell - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle = \langle \ell, \mathbf{x} \rangle - \langle \ell, \bar{\mathbf{x}} \rangle - \langle \mathbf{x}, \bar{\mathbf{x}} \rangle + \|\mathbf{x}\|^2$$

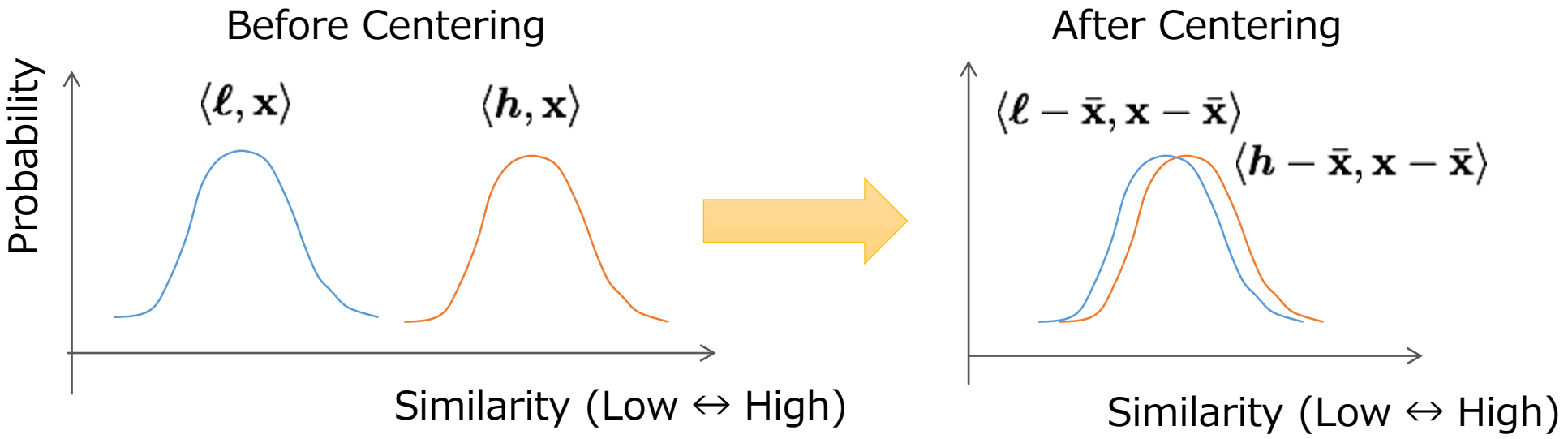
$$\mathrm{E}\left[\langle \boldsymbol{h} - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle\right] - \mathrm{E}\left[\langle \ell - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle\right] = \mathrm{E}\left[\langle \boldsymbol{h}, \mathbf{x} \rangle\right] - \mathrm{E}\left[\langle \boldsymbol{h}, \bar{\mathbf{x}} \rangle\right] - \mathrm{E}\left[\langle \ell, \mathbf{x} \rangle\right] + \mathrm{E}\left[\langle \ell, \bar{\mathbf{x}} \rangle\right] = 0$$

Centroid vector: $\bar{\mathbf{x}} = \sum_{i=1}^{N} \mathbf{x}_i$

$\therefore$ The mean of two distributions are not different : $\mathrm{E}\left[\langle \boldsymbol{h} - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle\right] = \mathrm{E}\left[\langle \ell - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle\right]$

Summary: Hubs are Reduced by Centering

- Before centering, the sample(***h***) which is more similar to the mean(***μ***) is more similar with other samples. The sample ***h*** was a hub sample.
- After centering, the sample ***h*** is not more similar with other samples, so that the sample ***h*** is no longer a hub sample.



Overall Summary

- Hub is a sample which appear many other samples' kNN.
- To reduce hubs,
  - Laplacian based kernels
  - centered similarity measures

  have nice properties to reduce hubs, i.e. make all samples equally similar to the data centroid.
- In experiment, Laplacian based kernels and centering similarity measures are effective to reduce hubs and improve kNN based classification.
- In theoretical analysis, centering similarity measures have effect to reduce hubs.