文字探勘
(Text Mining)
文本相似度和分群
(Text Similarity and Clustering)

1082TM08
MBA, BDABI, TKU (E3611) (8480) (Spring 2020)
Mon, 7, 8, 9 (14:10-17:00) (B206)

Chichang Jou
Associate Professor
周清江
副教授
cjou@mail.tku.edu.tw

Min-Yuh Day
Associate Professor
戴敏育
副教授
myday@mail.tku.edu.tw

Dept. of Information Management, Tamkang University
淡江大學 資訊管理學系
課程大綱 (Syllabus)

週次 (Week) | 日期 (Date) | 內容 (Subject/Topics)
---|---|---
1 | 2020/03/02 | 文字探勘課程介紹 (Course Orientation on Text Mining)
2 | 2020/03/09 | 文字探勘基礎：自然語言處理 (Foundations of Text Mining: Natural Language Processing; NLP)
3 | 2020/03/16 | Python自然語言處理 (Python for Natural Language Processing)
4 | 2020/03/23 | 處理和理解文本 (Processing and Understanding Text)
5 | 2020/03/30 | 文本表達特徵工程 (Feature Engineering for Text Representation)
6 | 2020/04/06 | 人工智慧文本分析個案研究 I (Case Study on Artificial Intelligence for Text Analytics I)
<table>
<thead>
<tr>
<th>週次 (Week)</th>
<th>日期 (Date)</th>
<th>內容 (Subject/Topics)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>2020/04/13</td>
<td>文本分類 (Text Classification)</td>
</tr>
<tr>
<td>8</td>
<td>2020/04/20</td>
<td>文本摘要和主題模型 (Text Summarization and Topic Models)</td>
</tr>
<tr>
<td>9</td>
<td>2020/04/27</td>
<td>期中報告 (Midterm Project Report)</td>
</tr>
<tr>
<td>10</td>
<td>2020/05/04</td>
<td>文本相似度和分群 (Text Similarity and Clustering)</td>
</tr>
<tr>
<td>11</td>
<td>2020/05/11</td>
<td>語意分析和命名實體識別 (Semantic Analysis and Named Entity Recognition; NER)</td>
</tr>
<tr>
<td>12</td>
<td>2020/05/18</td>
<td>情感分析 (Sentiment Analysis)</td>
</tr>
</tbody>
</table>
課程大綱 (Syllabus)

週次 (Week) 日期 (Date)  內容 (Subject/Topics)
13 2020/05/25 人工智慧文本分析個案研究Ⅱ  
    (Case Study on Artificial Intelligence for Text Analytics II)
14 2020/06/01 深度學習和通用句子嵌入模型  
    (Deep Learning and Universal Sentence-Embedding Models)
15 2020/06/08 問答系統與對話系統  
    (Question Answering and Dialogue Systems)
16 2020/06/15 期末報告 I (Final Project Presentation I)
17 2020/06/22 期末報告 II (Final Project Presentation II)
18 2020/06/29 教師彈性補充教學
Outline

• Text Similarity
• Text Clustering
  • Cluster Analysis
• K-Means Clustering
Text Similarity and Clustering
Text Similarity and Clustering

- **Text Dataset** (Unsupervised)
- **Text Pre-Processing**
- **Feature Extraction** (Vectorization) (TF-IDF) (Embedding)
- **Text Similarity**
- **Text Clustering**
Text Similarity and Clustering

• How do we measure similarity between terms and documents?
• How can we use distance measures to find the most relevant documents?
• How can we build a recommender system from text similarity?
• How do we group similar documents (document clustering)?

Text Similarity and Clustering

• Information Retrieval (IR)
• Feature Engineering
• Similarity Measures
• Unsupervised Machine Learning Algorithms

Text Similarity

• Lexical similarity
  – Syntax, structure, and content of the documents

• Semantic similarity
  – Semantics, meaning, and context of the documents

Cosine Similarity

$\theta$

$P_1 = (3, 4)$

$P_2 = (8, 5)$
NLP

Classical NLP

Documents

Language Detection

English
Spanish
Arabic

Pre-processing

Tokenization (English)

POS Tagging (English)

Stopword Removal (EN)

...

Modeling

Feature Extraction (EN)

Modeling (English)

Inference (English)

Output

Sentiment
Classification
Entity Extraction
Translation
Topic Modelling

Deep Learning-based NLP

Documents

Preprocessing

Dense Embeddings

obtained via word2vec, doc2vec, GloVe, etc.

Hidden Layers

Output Units

Output

Sentiment
Classification
Entity Extraction
Translation
Topic Modelling

Source: http://blog.aylien.com/leveraging-deep-learning-for-multilingual/
Modern NLP Pipeline

Documents

Language Detection

Tokenize

POS Tagging

... Token Filtering

Pre-processed Documents

Pre-processed Documents

Build Vocabulary

Bag-of-Words & Vectorization

Machine Learning

Task / Output

Classification

Sentiment Analysis

Entity Extraction

Topic Modeling

Similarity

Pre-processed Documents

Word Embeddings

word2vec
doc2vec
GloVe

(Deep) Neural Network

Modern NLP Pipeline
Deep Learning NLP

Documents → Preprocessing → Dense Word Embeddings → Deep Neural Network

Task / Output
- Classification
- Sentiment Analysis
- Entity Extraction
- Topic Modeling
- Document Similarity

Pre-generated Lookup OR Generated in 1st level of NeuralNet

Source: http://mattfortier.me/2017/01/31/nlp-intro-pt-1-overview/
Natural Language Processing (NLP) and Text Mining

- **Raw text**
- **Sentence Segmentation**
- **Tokenization**
- **Part-of-Speech (POS)**
- **Stop word removal**
- **Stemming / Lemmatization**
- **Dependency Parser**
- **String Metrics & Matching**

- word’s stem: am → am
- word’s lemma: am → be
- having → hav
- having → have

Source: Nitin Hardeniya (2015), NLTK Essentials, Packt Publishing; Florian Leitner (2015), Text mining - from Bayes rule to dependency parsing
# Data Mining Tasks & Methods

<table>
<thead>
<tr>
<th>Data Mining Tasks &amp; Methods</th>
<th>Data Mining Algorithms</th>
<th>Learning Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Prediction</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Classification</td>
<td>Decision Trees, Neural Networks, Support Vector Machines, kNN, Naive Bayes, GA</td>
<td>Supervised</td>
</tr>
<tr>
<td>Regression</td>
<td>Linear/Nonlinear Regression, ANN, Regression Trees, SVM, kNN, GA</td>
<td>Supervised</td>
</tr>
<tr>
<td>Time series</td>
<td>Autoregressive Methods, Averaging Methods, Exponential Smoothing, ARIMA</td>
<td>Supervised</td>
</tr>
<tr>
<td><strong>Association</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Link analysis</td>
<td>Expectation Maximization, Apriori Algorithm, Graph-Based Matching</td>
<td>Unsupervised</td>
</tr>
<tr>
<td>Sequence analysis</td>
<td>Apriori Algorithm, FP-Growth, Graph-Based Matching</td>
<td>Unsupervised</td>
</tr>
<tr>
<td><strong>Segmentation</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clustering</td>
<td>k-means, Expectation Maximization (EM)</td>
<td>Unsupervised</td>
</tr>
<tr>
<td>Outlier analysis</td>
<td>k-means, Expectation Maximization (EM)</td>
<td>Unsupervised</td>
</tr>
</tbody>
</table>

Source: Ramesh Sharda, Dursun Delen, and Efraim Turban (2017), Business Intelligence, Analytics, and Data Science: A Managerial Perspective, 4th Edition, Pearson
Example of Cluster Analysis

<table>
<thead>
<tr>
<th>Point</th>
<th>P</th>
<th>P(x,y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>p01</td>
<td>a</td>
<td>(3, 4)</td>
</tr>
<tr>
<td>p02</td>
<td>b</td>
<td>(3, 6)</td>
</tr>
<tr>
<td>p03</td>
<td>c</td>
<td>(3, 8)</td>
</tr>
<tr>
<td>p04</td>
<td>d</td>
<td>(4, 5)</td>
</tr>
<tr>
<td>p05</td>
<td>e</td>
<td>(4, 7)</td>
</tr>
<tr>
<td>p06</td>
<td>f</td>
<td>(5, 1)</td>
</tr>
<tr>
<td>p07</td>
<td>g</td>
<td>(5, 5)</td>
</tr>
<tr>
<td>p08</td>
<td>h</td>
<td>(7, 3)</td>
</tr>
<tr>
<td>p09</td>
<td>i</td>
<td>(7, 5)</td>
</tr>
<tr>
<td>p10</td>
<td>j</td>
<td>(8, 5)</td>
</tr>
</tbody>
</table>
## K-Means Clustering

<table>
<thead>
<tr>
<th>Point</th>
<th>P</th>
<th>P(x,y)</th>
<th>m1 distance</th>
<th>m2 distance</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>p01</td>
<td>a</td>
<td>(3, 4)</td>
<td>1.95</td>
<td>3.78</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p02</td>
<td>b</td>
<td>(3, 6)</td>
<td>0.69</td>
<td>4.51</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p03</td>
<td>c</td>
<td>(3, 8)</td>
<td>2.27</td>
<td>5.86</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p04</td>
<td>d</td>
<td>(4, 5)</td>
<td>0.89</td>
<td>3.13</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p05</td>
<td>e</td>
<td>(4, 7)</td>
<td>1.22</td>
<td>4.45</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p06</td>
<td>f</td>
<td>(5, 1)</td>
<td>5.01</td>
<td>3.05</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p07</td>
<td>g</td>
<td>(5, 5)</td>
<td>1.57</td>
<td>2.30</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p08</td>
<td>h</td>
<td>(7, 3)</td>
<td>4.37</td>
<td>0.56</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p09</td>
<td>i</td>
<td>(7, 5)</td>
<td>3.43</td>
<td>1.52</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p10</td>
<td>j</td>
<td>(8, 5)</td>
<td>4.41</td>
<td>1.95</td>
<td>Cluster2</td>
</tr>
</tbody>
</table>

m1: (3.67, 5.83)
m2: (6.75, 3.50)
Cluster Analysis
Cluster Analysis

• Used for automatic identification of natural groupings of things
• Part of the machine-learning family
• Employ unsupervised learning
• Learns the clusters of things from past data, then assigns new instances
• There is not an output variable
• Also known as segmentation

Source: Turban et al. (2011), Decision Support and Business Intelligence Systems
Cluster Analysis

Clustering of a set of objects based on the \textit{k-means method}.  
\textit{(The mean of each cluster is marked by a “+”.)}

Source: Han & Kamber (2006)
Example of Cluster Analysis

Point | P | \(P(x,y)\)
---|---|---
p01 | a | (3, 4)
p02 | b | (3, 6)
p03 | c | (3, 8)
p04 | d | (4, 5)
p05 | e | (4, 7)
p06 | f | (5, 1)
p07 | g | (5, 5)
p08 | h | (7, 3)
p09 | i | (7, 5)
p10 | j | (8, 5)
Cluster Analysis for Data Mining

• How many clusters?
  – There is not a “truly optimal” way to calculate it
  – Heuristics are often used
    1. Look at the sparseness of clusters
    2. Number of clusters = \((\frac{n}{2})^{1/2}\) (n: no of data points)
    3. Use Akaike information criterion (AIC)
    4. Use Bayesian information criterion (BIC)

• Most cluster analysis methods involve the use of a distance measure to calculate the closeness between pairs of items
  – Euclidian versus Manhattan (rectilinear) distance

Source: Turban et al. (2011), Decision Support and Business Intelligence Systems
**k-Means Clustering Algorithm**

- **k**: pre-determined number of clusters
- Algorithm ([Step 0](#): determine value of \( k \))

**Step 1**: Randomly generate \( k \) random points as initial cluster centers

**Step 2**: Assign each point to the nearest cluster center

**Step 3**: Re-compute the new cluster centers

**Repetition step**: Repeat steps 2 and 3 until some convergence criterion is met (usually that the assignment of points to clusters becomes stable)

Source: Turban et al. (2011), Decision Support and Business Intelligence Systems
Cluster Analysis for Data Mining - $k$-Means Clustering Algorithm

Source: Turban et al. (2011), Decision Support and Business Intelligence Systems
Similarity
Distance
Similarity and Dissimilarity Between Objects

- **Distances** are normally used to measure the similarity or dissimilarity between two data objects.

- Some popular ones include: *Minkowski distance*:

  \[ d(i, j) = \sqrt[q]{|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \ldots + |x_{ip} - x_{jp}|^q} \]

  where \( i = (x_{i1}, x_{i2}, \ldots, x_{ip}) \) and \( j = (x_{j1}, x_{j2}, \ldots, x_{jp}) \) are two \( p \)-dimensional data objects, and \( q \) is a positive integer.

- If \( q = 1 \), \( d \) is *Manhattan distance*:

  \[ d(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}| \]

Source: Han & Kamber (2006)
Similarity and Dissimilarity Between Objects (Cont.)

• If $q = 2$, $d$ is Euclidean distance:

$$d(i,j) = \sqrt{(|x_{i_1} - x_{j_1}|^2 + |x_{i_2} - x_{j_2}|^2 + \ldots + |x_{i_p} - x_{j_p}|^2)}$$

— Properties

• $d(i,j) \geq 0$
• $d(i,i) = 0$
• $d(i,j) = d(j,i)$
• $d(i,j) \leq d(i,k) + d(k,j)$

• Also, one can use weighted distance, parametric Pearson product moment correlation, or other dissimilarity measures

Source: Han & Kamber (2006)
Euclidean distance vs Manhattan distance

• Distance of two points \( \mathbf{x}_1 = (1, 2) \) and \( \mathbf{x}_2 (3, 5) \)

Euclidean distance:
\[
= \sqrt{(3-1)^2 + (5-2)^2} \\
= \sqrt{2^2 + 3^2} \\
= \sqrt{4 + 9} \\
= \sqrt{13} \\
= 3.61
\]

Manhattan distance:
\[
= |3-1| + |5-2| \\
= 2 + 3 \\
= 5
\]
The **K-Means Clustering Method**

- **Example**

K=2

Arbitrarily choose K object as initial cluster center

Assign each objects to most similar center

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
+---+---+---+---+---+---+---+---+---+---+----|
| 10| 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0   |
| 10| 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0   |
| 10| 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0   |
| 10| 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0   |

Update the cluster means

Reassign

Update the cluster means

Reassign

Source: Han & Kamber (2006)
K-Means Clustering
Example of Cluster Analysis

<table>
<thead>
<tr>
<th>Point</th>
<th>P</th>
<th>P(x,y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>p01</td>
<td>a</td>
<td>(3, 4)</td>
</tr>
<tr>
<td>p02</td>
<td>b</td>
<td>(3, 6)</td>
</tr>
<tr>
<td>p03</td>
<td>c</td>
<td>(3, 8)</td>
</tr>
<tr>
<td>p04</td>
<td>d</td>
<td>(4, 5)</td>
</tr>
<tr>
<td>p05</td>
<td>e</td>
<td>(4, 7)</td>
</tr>
<tr>
<td>p06</td>
<td>f</td>
<td>(5, 1)</td>
</tr>
<tr>
<td>p07</td>
<td>g</td>
<td>(5, 5)</td>
</tr>
<tr>
<td>p08</td>
<td>h</td>
<td>(7, 3)</td>
</tr>
<tr>
<td>p09</td>
<td>i</td>
<td>(7, 5)</td>
</tr>
<tr>
<td>p10</td>
<td>j</td>
<td>(8, 5)</td>
</tr>
</tbody>
</table>
**K-Means Clustering**

**Step by Step**

<table>
<thead>
<tr>
<th>Point</th>
<th>P</th>
<th>P(x, y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>p01</td>
<td>a</td>
<td>(3, 4)</td>
</tr>
<tr>
<td>p02</td>
<td>b</td>
<td>(3, 6)</td>
</tr>
<tr>
<td>p03</td>
<td>c</td>
<td>(3, 8)</td>
</tr>
<tr>
<td>p04</td>
<td>d</td>
<td>(4, 5)</td>
</tr>
<tr>
<td>p05</td>
<td>e</td>
<td>(4, 7)</td>
</tr>
<tr>
<td>p06</td>
<td>f</td>
<td>(5, 1)</td>
</tr>
<tr>
<td>p07</td>
<td>g</td>
<td>(5, 5)</td>
</tr>
<tr>
<td>p08</td>
<td>h</td>
<td>(7, 3)</td>
</tr>
<tr>
<td>p09</td>
<td>i</td>
<td>(7, 5)</td>
</tr>
<tr>
<td>p10</td>
<td>j</td>
<td>(8, 5)</td>
</tr>
</tbody>
</table>
**K-Means Clustering**

Step 1: **K=2**, Arbitrarily choose K object as initial cluster center

Point | P | P(x,y)  
--- | --- | ---  
p01 | a | (3, 4)  
p02 | b | (3, 6)  
p03 | c | (3, 8)  
p04 | d | (4, 5)  
p05 | e | (4, 7)  
p06 | f | (5, 1)  
p07 | g | (5, 5)  
p08 | h | (7, 3)  
p09 | i | (7, 5)  
p10 | j | (8, 5)  

Initial m1  (3, 4)  
Initial m2  (8, 5)
Step 2: Compute seed points as the centroids of the clusters of the current partition

Step 3: Assign each objects to most similar center

<table>
<thead>
<tr>
<th>Point</th>
<th>P</th>
<th>P(x,y)</th>
<th>m1 distance</th>
<th>m2 distance</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>p01</td>
<td>a</td>
<td>(3, 4)</td>
<td>0.00</td>
<td>5.10</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p02</td>
<td>b</td>
<td>(3, 6)</td>
<td>2.00</td>
<td>5.10</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p03</td>
<td>c</td>
<td>(3, 8)</td>
<td>4.00</td>
<td>5.83</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p04</td>
<td>d</td>
<td>(4, 5)</td>
<td>1.41</td>
<td>4.00</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p05</td>
<td>e</td>
<td>(4, 7)</td>
<td>3.16</td>
<td>4.47</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p06</td>
<td>f</td>
<td>(5, 1)</td>
<td>3.61</td>
<td>5.00</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p07</td>
<td>g</td>
<td>(5, 5)</td>
<td>2.24</td>
<td>3.00</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p08</td>
<td>h</td>
<td>(7, 3)</td>
<td>4.12</td>
<td>2.24</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p09</td>
<td>i</td>
<td>(7, 5)</td>
<td>4.12</td>
<td>1.00</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p10</td>
<td>j</td>
<td>(8, 5)</td>
<td>5.10</td>
<td>0.00</td>
<td>Cluster2</td>
</tr>
</tbody>
</table>

$M_1 = (3, 4)$

$M_2 = (8, 5)$

**K-Means Clustering**

Initial m1 (3, 4)

Initial m2 (8, 5)
Step 2: Compute seed points as the centroids of the clusters of the current partition

Step 3: Assign each object to the most similar center.

Euclidean distance
\[ d_{b(3,6), m1(3,4)} = ((3-3)^2 + (4-6)^2)^{1/2} = (0^2 + (-2)^2)^{1/2} = (0 + 4)^{1/2} = (4)^{1/2} = 2.00 \]

Euclidean distance
\[ d_{b(3,6), m2(8,5)} = ((8-3)^2 + (5-6)^2)^{1/2} = (5^2 + (-1)^2)^{1/2} = (25 + 1)^{1/2} = (26)^{1/2} = 5.10 \]
**K-Means Clustering**

**Step 4:** Update the cluster means, repeat Step 2, 3, stop when no more new assignment

<table>
<thead>
<tr>
<th>Point</th>
<th>P</th>
<th>P(x,y)</th>
<th>m1 distance</th>
<th>m2 distance</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>p01</td>
<td>a</td>
<td>(3, 4)</td>
<td>1.43</td>
<td>4.34</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p02</td>
<td>b</td>
<td>(3, 6)</td>
<td>1.22</td>
<td>4.64</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p03</td>
<td>c</td>
<td>(3, 8)</td>
<td>2.99</td>
<td>5.68</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p04</td>
<td>d</td>
<td>(4, 5)</td>
<td>0.20</td>
<td>3.40</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p05</td>
<td>e</td>
<td>(4, 7)</td>
<td>1.87</td>
<td>4.27</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p06</td>
<td>f</td>
<td>(5, 1)</td>
<td>4.29</td>
<td>4.06</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p07</td>
<td>g</td>
<td>(5, 5)</td>
<td>1.15</td>
<td>2.42</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p08</td>
<td>h</td>
<td>(7, 3)</td>
<td>3.80</td>
<td>1.37</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p09</td>
<td>i</td>
<td>(7, 5)</td>
<td>3.14</td>
<td>0.75</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p10</td>
<td>j</td>
<td>(8, 5)</td>
<td>4.14</td>
<td>0.95</td>
<td>Cluster2</td>
</tr>
</tbody>
</table>

$m_1 = (3.86, 5.14)$

$M_2 = (7.33, 4.33)$
Step 4: Update the cluster means,
Repeat Step 2, 3, stop when no more new assignment

Point | P | P(x,y) | m1 distance | m2 distance | Cluster
--- | --- | ----- | -------- | -------- | -----
p01 | a | (3, 4) | 1.95 | 3.78 | Cluster1
p02 | b | (3, 6) | 0.69 | 4.51 | Cluster1
p03 | c | (3, 8) | 2.27 | 5.86 | Cluster1
p04 | d | (4, 5) | 0.89 | 3.13 | Cluster1
p05 | e | (4, 7) | 1.22 | 4.45 | Cluster1
p06 | f | (5, 1) | 5.01 | 3.05 | Cluster2
p07 | g | (5, 5) | 1.57 | 2.30 | Cluster1
p08 | h | (7, 3) | 4.37 | 0.56 | Cluster2
p09 | i | (7, 5) | 3.43 | 1.52 | Cluster2
p10 | j | (8, 5) | 4.41 | 1.95 | Cluster2

m1 = (3.67, 5.83)
m2 = (6.75, 3.50)
**K-Means Clustering**

The algorithm continues until no more new assignments are made.

<table>
<thead>
<tr>
<th>Point</th>
<th>P</th>
<th>P(x,y)</th>
<th>m1 distance</th>
<th>m2 distance</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>p01</td>
<td>a</td>
<td>(3, 4)</td>
<td>1.95</td>
<td>3.78</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p02</td>
<td>b</td>
<td>(3, 6)</td>
<td>0.69</td>
<td>4.51</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p03</td>
<td>c</td>
<td>(3, 8)</td>
<td>2.27</td>
<td>5.86</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p04</td>
<td>d</td>
<td>(4, 5)</td>
<td>0.89</td>
<td>3.13</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p05</td>
<td>e</td>
<td>(4, 7)</td>
<td>1.22</td>
<td>4.45</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p06</td>
<td>f</td>
<td>(5, 1)</td>
<td>5.01</td>
<td>3.05</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p07</td>
<td>g</td>
<td>(5, 5)</td>
<td>1.57</td>
<td>2.30</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p08</td>
<td>h</td>
<td>(7, 3)</td>
<td>4.37</td>
<td>0.56</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p09</td>
<td>i</td>
<td>(7, 5)</td>
<td>3.43</td>
<td>1.52</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p10</td>
<td>j</td>
<td>(8, 5)</td>
<td>4.41</td>
<td>1.95</td>
<td>Cluster2</td>
</tr>
</tbody>
</table>

The initial points are:

- **m1** (3.67, 5.83)
- **m2** (6.75, 3.50)
## K-Means Clustering (K=2, two clusters)

**Stop when no more new assignment**

<table>
<thead>
<tr>
<th>Point</th>
<th>P</th>
<th>P(x,y)</th>
<th>m1 distance</th>
<th>m2 distance</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>p01</td>
<td>a</td>
<td>(3, 4)</td>
<td>1.95</td>
<td>3.78</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p02</td>
<td>b</td>
<td>(3, 6)</td>
<td>0.69</td>
<td>4.51</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p03</td>
<td>c</td>
<td>(3, 8)</td>
<td>2.27</td>
<td>5.86</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p04</td>
<td>d</td>
<td>(4, 5)</td>
<td>0.89</td>
<td>3.13</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p05</td>
<td>e</td>
<td>(4, 7)</td>
<td>1.22</td>
<td>4.45</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p06</td>
<td>f</td>
<td>(5, 1)</td>
<td>5.01</td>
<td>3.05</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p07</td>
<td>g</td>
<td>(5, 5)</td>
<td>1.57</td>
<td>2.30</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p08</td>
<td>h</td>
<td>(7, 3)</td>
<td>4.37</td>
<td>0.56</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p09</td>
<td>i</td>
<td>(7, 5)</td>
<td>3.43</td>
<td>1.52</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p10</td>
<td>j</td>
<td>(8, 5)</td>
<td>4.41</td>
<td>1.95</td>
<td>Cluster2</td>
</tr>
</tbody>
</table>

**K-Means Clustering**

m1 (3.67, 5.83)

m2 (6.75, 3.50)
## K-Means Clustering

<table>
<thead>
<tr>
<th>Point</th>
<th>P</th>
<th>P(x,y)</th>
<th>m1 distance</th>
<th>m2 distance</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>p01</td>
<td>a</td>
<td>(3, 4)</td>
<td>1.95</td>
<td>3.78</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p02</td>
<td>b</td>
<td>(3, 6)</td>
<td>0.69</td>
<td>4.51</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p03</td>
<td>c</td>
<td>(3, 8)</td>
<td>2.27</td>
<td>5.86</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p04</td>
<td>d</td>
<td>(4, 5)</td>
<td>0.89</td>
<td>3.13</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p05</td>
<td>e</td>
<td>(4, 7)</td>
<td>1.22</td>
<td>4.45</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p06</td>
<td>f</td>
<td>(5, 1)</td>
<td>5.01</td>
<td>3.05</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p07</td>
<td>g</td>
<td>(5, 5)</td>
<td>1.57</td>
<td>2.30</td>
<td>Cluster1</td>
</tr>
<tr>
<td>p08</td>
<td>h</td>
<td>(7, 3)</td>
<td>4.37</td>
<td>0.56</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p09</td>
<td>i</td>
<td>(7, 5)</td>
<td>3.43</td>
<td>1.52</td>
<td>Cluster2</td>
</tr>
<tr>
<td>p10</td>
<td>j</td>
<td>(8, 5)</td>
<td>4.41</td>
<td>1.95</td>
<td>Cluster2</td>
</tr>
</tbody>
</table>

m1  (3.67, 5.83)
m2  (6.75, 3.50)
gensim

topic modelling for humans

Gensim is a FREE Python library

- Scalable statistical semantics
- Analyze plain-text documents for semantic structure
- Retrieve semantically similar documents

https://radimrehurek.com/gensim/
Fastest in the world

spacy excels at large-scale information extraction tasks. It's written from the ground up in carefully memory-managed Cython. Independent research has confirmed that spacy is the fastest in the world. If your application needs to process entire web dumps, spacy is the library you want to be using.

Get things done

spacy is designed to help you do real work — to build real products, or gather real insights. The library respects your time, and tries to avoid wasting it. It's easy to install, and its API is simple and productive. I like to think of spacy as the Ruby on Rails of Natural Language Processing.

Deep learning

spacy is the best way to prepare text for deep learning. It interoperates seamlessly with TensorFlow, Keras, Scikit-Learn, Gensim and the rest of Python's awesome AI ecosystem. spacy helps you connect the statistical models trained by these libraries to the rest of your application.
Text Similarity and Clustering

Text Similarity

- Spacy Vectors Similarity: https://spacy.io/usage/vectors-similarity

```python
[1] 1 !python -m spacy download en_core_web_sm

[2] 1 !python -m spacy download en_core_web_lg
2 # Restart Runtime

[3] 1 import spacy
 2 nlp = spacy.load("en_core_web_lg")
 3 tokens = nlp("apple banana cat dog notaword")
 4 for token in tokens:
 5   print(token.text, token.has_vector, token.vector_norm, token.is_oov)
```

apple True 7.1346846 False
banana True 6.700014 False
cat True 6.6808186 False
dog True 7.0336733 False
notaword False 0.0 True

```python
[1] 1 import spacy
 2 nlp = spacy.load("en_core_web_lg")
 3 doc1 = nlp("I like cat.")
 4 doc2 = nlp("I like dog.")
 5 doc1.similarity(doc2)
```

https://tinyurl.com/imtkupython101
Python in Google Colab (Python101)

https://colab.research.google.com/drive/1FEG6DnGvwfUbeo4zJ1zTunjMqf2RkCrT

https://tinyurl.com/imtkupython101
Python in Google Colab (Python101)

https://colab.research.google.com/drive/1FEG6DnGvwfUbeo4zJ1zTunjMqf2RkCrT

Hierarchical Clustering Dendrogram

https://tinyurl.com/imtkupython101
Summary

• Text Similarity
• Text Clustering
  • Cluster Analysis
• K-Means Clustering
References

