2024-2 Unstructured Data Analysis
2024.10.16

Dimensionality Reduction

Week 7 – Code Practice

Professor: Misuk Kim
Teaching Assistant: Minjoo Son
minjoo77@hanyang.ac.kr

HANYANG UNIVERSITY

Dimensionality Reduction
Week 7 – Code Practice

Contents

❖ Week 7 Objective

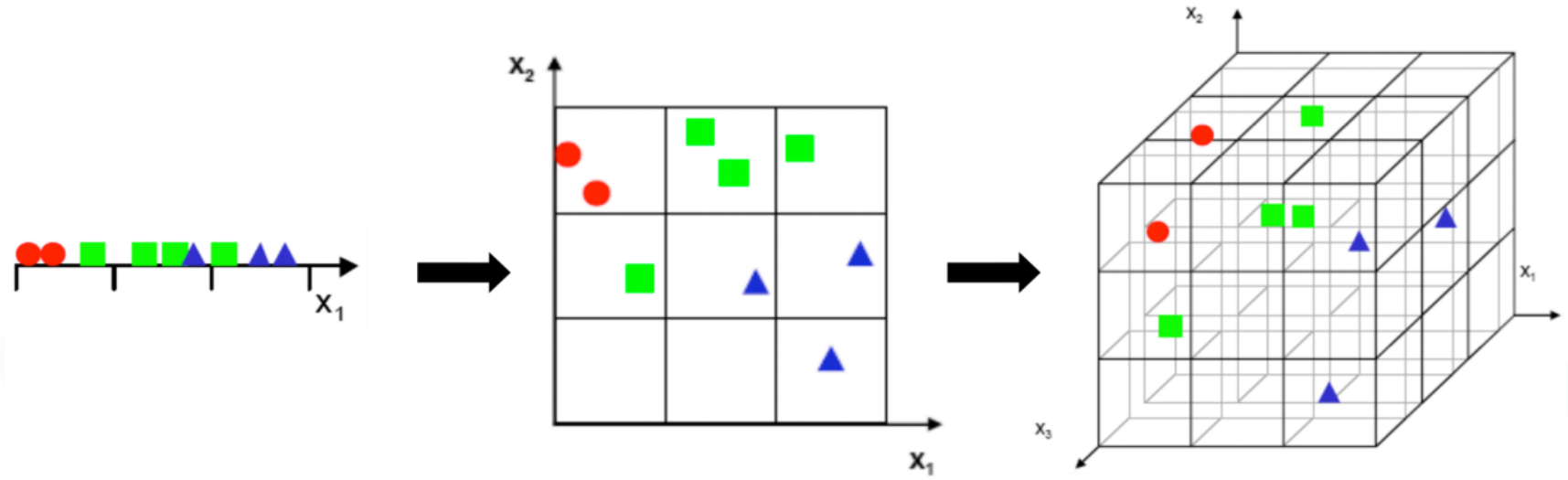
- Dimensionality Reduction
  - Dimensionality Reduction
  - The Curse of Dimensionality and Reasons for Dimensionality Reduction
  - Dimensionality Reduction using PCA
  - Dimensionality Reduction and Meaning Interpretation using LSA
    - Performance of Dimensionality Reduction using LSA
    - Semantic Similarity Calculation between Documents using LSA
    - Analysis of Latent Topics
    - Analysis of Word Semantic Similarity
  - Visualization and Effects of Dimensionality Reduction using tSNE

❖ Dimensionality Reduction

- In problems such as document classification, dimensionality refers to the number of features that represent the document.
- In text mining based on the Bag of Words (BoW) model, the number of features can become extremely large, inherently leading to issues like the curse of dimensionality.

❖ Dimensionality Reduction

- Curse of Dimensionality: This refers to problems that arise when the dimensionality increases, which do not occur in lower dimensions.
  - It refers to the increase in the number of observed variables, and as the dimensionality grows, the space formed by the dimensions becomes larger, causing the distance between the data points in this space to increase.
  - As a result, the data becomes sparse. When data becomes sparse, the models learned based on this data typically have reduced explanatory power, leading to poorer performance.



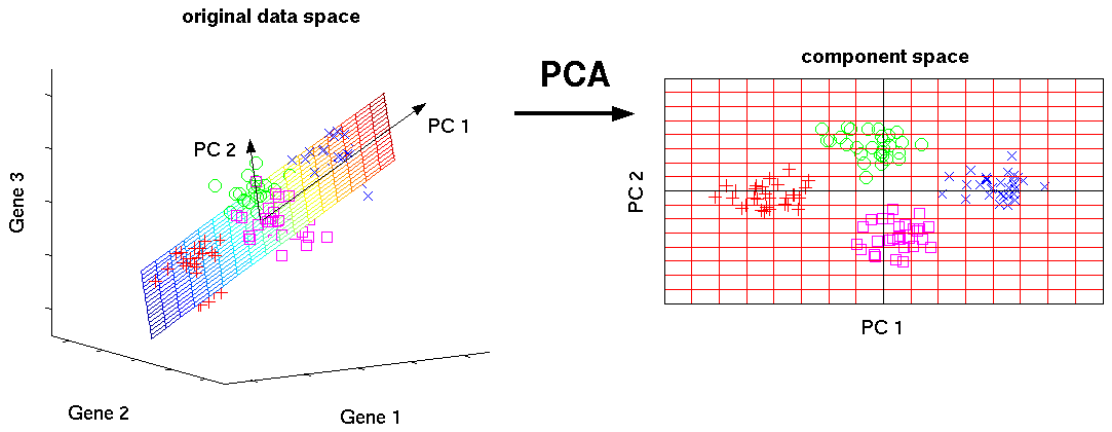
❖ Dimensionality Reduction

- Methods for Solving the Curse of Dimensionality
  1. Increasing the amount of data sufficiently
  2. Reducing the number of features represented by the document using BOW
     - ① Feature Selection: Selecting representative features from the existing ones
     - ② Feature Extraction: Creating new features by combining the existing feature values (e.g., PCA, LSA)

- Code Practice
  - Link: https://drive.google.com/drive/folders/1astmYfYGZKk3mfoiPMb5muRbfCOTo4hI?usp=drive_link

❖ Dimensionality Reduction using PCA (Principal Component Analysis)

- PCA is a method for dimensionality reduction by transforming the data onto new axes that maximize the variance.
  - The most important principle is that the reduction of dimensions should maintain as much information as possible.
  - In other words, even if we reduce 10 independent variables to 9, the new 9 variables should reflect as much of the information carried by the original 10 variables as possible.
  - Statistically, this amount of information can be measured by variance. → By maintaining the largest possible variance when reducing dimensions, we can minimize the loss of information.
  - PCA works by finding new axes (or components) that maximize the variance.



❖ Dimensionality Reduction using PCA (Principal Component Analysis)

- Scikit-learn provides the **PCA** library to support the application of PCA.
  - Link: https://scikit-learn.org/dev/modules/generated/sklearn.decomposition.PCA.html

PCA

```
class sklearn.decomposition.PCA(n_components=None, *, copy=True, whiten=False,
svd_solver='auto', tol=0.0, iterated_power='auto', n_oversamples=10,
power_iteration_normalizer='auto', random_state=None)
```

[source]

- n_components: The target size of the reduced dimensions
- svd_solver: The algorithm ('auto' is the default value, and it automatically selects the solver considering the original and target dimensions)
- explained_variance: The variance explained by each new axis
- explained_variance_ratio: Expresses the ratio of variance explained relative to the original variance. (If the new axes explain all of the original variance, this value will be 1.)

❖Dimensionality Reduction using PCA (Principal Component Analysis)

- Scikit-learn's PCA does not support direct computation on sparse vector formats.
- This means that matrices transformed by CountVectorizer or TfidfVectorizer cannot be directly used as arguments.
- Therefore, you need to use the toarray() method to convert the format before passing it as an argument to the transform() or fit_transform() methods.
- After reduction, you can output the sum of the explained_variance_ratio_ to check how much of the original variance is explained.

2. Dimensionality Reduction

❖Dimensionality Reduction and Meaning Interpretation using LSA (Latent Semantic Analysis)

- LSA can analyze the latent meanings in documents as well as the latent meanings in words.
  - In this context, the "meaning" serves as a mediator connecting documents and words, and the reduced dimensions play this role.
  - LSA is implemented using SVD (Singular Value Decomposition), more specifically, Truncated SVD.
- LSA analyzes the latent meanings in both documents and words, with the reduced dimensions serving as the bridge that connects them.
  - It is implemented using SVD (Singular Value Decomposition), specifically Truncated SVD.

❖ Dimensionality Reduction and Meaning Interpretation using LSA (Latent Semantic Analysis)

- Scikit-learn supports LSA for documents using the TruncatedSVD class.
  - Link: https://scikit-learn.org/dev/modules/generated/sklearn.decomposition.TruncatedSVD.html

TruncatedSVD

```
class sklearn.decomposition.TruncatedSVD(n_components=2, *, algorithm='randomized',
n_iter=5, n_oversamples=10, power_iteration_normalizer='auto', random_state=None,
tol=0.0)
```

[source]

n_components: The target size of the reduced dimensions
algorithm: SVD solver to use. Either “arpack” for the ARPACK wrapper in SciPy (scipy.sparse.linalg.svds), or “randomized” for the randomized algorithm due to Halko (2009).
n_iter: Number of iterations for randomized SVD solver. Not used by ARPACK.

❖ Dimensionality Reduction and Meaning Interpretation using LSA (Latent Semantic Analysis)

- Unlike PCA, which cannot directly compute sparse matrices, TruncatedSVD can perform the computation directly, and calling the toarray() method returns the reduced-dimensional document matrix.
- fit_transform() is used on training data, while transform() is used on test data.
  - The reduced dimensions represent latent meanings and connect documents and words.
  - In other words, while the document vector with m dimensions represents the document based on word frequencies, after reducing it to k dimensions, each document is represented by the weights of k latent meanings.
  - In simpler terms, the k reduced dimensions each represent latent meanings, and when applied to words, each

❖ Dimensionality Reduction and Latent Meaning Interpretation using LSA (Latent Semantic Analysis)

- Document Similarity Calculation using LSA (Latent Semantic Analysis)
  - The reason why applying SVD to natural language is called LSA is that the reduced dimensions can be interpreted as showing latent meanings.
  - When documents are transformed into vectors using count-based representations, it becomes possible to calculate distances or similarities between documents.
  - This similarity is calculated based on the frequency of words used in the document.
  - However, with LSA, the document vector is represented not by the words but by the weights of latent meanings, making it possible to calculate similarity based on these latent meanings.

❖ Dimensionality Reduction and Latent Meaning Interpretation using LSA (Latent Semantic Analysis)

- Analysis of Latent Topics
  - When documents are represented by the weights of latent meanings, and these latent meanings are associated with words, we can analyze the words connected to these meanings to identify the latent meanings in the documents.
  - These latent meanings are referred to as "topics," and this type of analysis is known as topic modeling.
  - Before the introduction of LDA (Latent Dirichlet Allocation), LSA was used for topic modeling, but nowadays, LDA is more commonly used.

❖ Dimensionality Reduction and Latent Meaning Interpretation using LSA (Latent Semantic Analysis)

- Analysis of Word Semantic Similarity
  - Just as LSA can be used to compute document similarity, it can also be used to compute word similarity.
  - From the perspective of a count vector matrix, a word can be represented by its occurrence frequency across all documents.
  - Therefore, word similarity can be calculated based on their frequency distributions across documents.In LSA, since documents are represented by the weights of latent meanings, these can also be used to calculate the semantic similarity between words.
  - However, the semantic distribution of each word is limited to the given corpus.

출처: https://scikit-learn.org/dev/modules/generated/sklearn.decomposition.TruncatedSVD.html

❖Visualization and Dimensionality Reduction Effects using tSNE

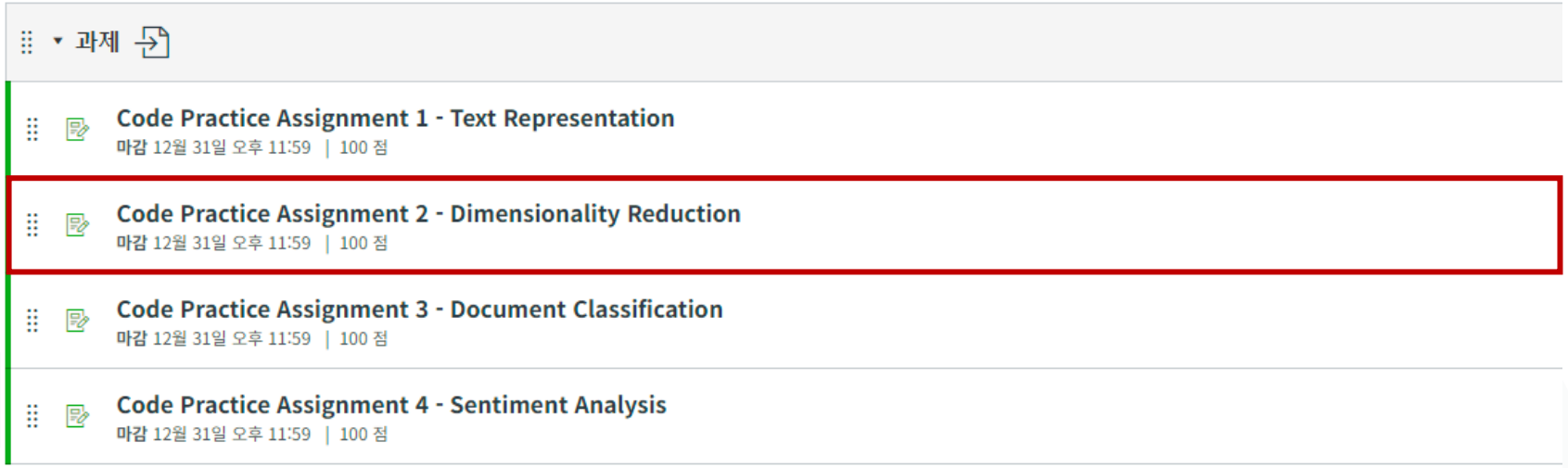
- tSNE is an unsupervised learning algorithm used for visualization, which finds 2D coordinates that best preserve the distances between points in high-dimensional data.
  - By mapping the distances between data points to a 2D space, it allows for an intuitive visual representation.
  - If the graph produced by tSNE appears well-formed, it can be inferred that the distances in the original high-dimensional data have been meaningfully preserved.

❖ Comprehensive Code Practice

- 'Comprehensive Code Practice.ipynb'
- Let's comprehensively apply the concepts we've learned so far:
  - text preprocessing, text representation, and dimensionality reduction.

❖ Assignment 2 – Dimensionality Reduction

- Modify the code based on ‘Comprehensive Code Practice.ipynb’ as per the following details:
  - Dimensionality Reduction: Use LSA or t-SNE instead of PCA
- Save the file as ‘Assignment_YourName_YourStudentID.ipynb’ and submit it to the ‘Code Practice Assignment 2 – Dimensionality Reduction’ section under Assignments
- Deadline: 23:00 on Monday, October 21st.



4. Mid-term Examinations

❖ About Mid-term Exam

- Programming: 20 points
- The Exam focuses only on the core content covered in class.
  - Week 4, 6, 7
  - You are not required to write code from scratch.
  - Example question types
    - Filling in the names of key libraries
    - Provide explanations for why specific arguments are used.

❖ GitHub Link

- I plan to upload all the codes, data, and class materials we've covered during the lab sessions to the following GitHub page starting this Friday Oct 18th at 7 PM.
- Additionally, I will create a folder called "Mid-term Exams" where I will upload the most relevant codes that would be helpful to review before the exam. It would be good to focus on these codes over the weekend and next Monday and Tuesday.
- Upload the code and PowerPoint related to the practice class so far.
- GitHub link: https://github.com/ming9oori/Unstructured-Data-Analysis/tree/main

Dimensionality Reduction
Week 7 – Code Practice

2024-2 Unstructured Data Analysis
2024.10.16

Q & A

Thank you for your attention. Any questions are welcome!

Minjoo Son

HANYANG UNIVERSITY