Text Mining in Economics and Health Economics using Stata

Carlo Drago

University Niccolo Cusano, Rome

May 6, 2024

XVIII Italian Conference of Users of Stata

Agenda

1. Introduction to Text Mining using Stata
2. What is Document Clustering?
3. Term-Document Matrices in Stata
4. Clustering Documents in Stata
5. Challenges in Clustering
6. Case Studies in Stata

Introduction to Text Mining using Stata

- **Text Mining is an important area of Data Science** that focuses on extracting insights and essential information from unstructured text data (see Aggarwal & Zhai 2012, Hotho et al. 2005).
- In the digital age, the **exponential growth of text data** requires appropriate approaches to interpret, evaluate, and efficiently utilize this relevant resource.
- Text mining uses **algorithms and models** to facilitate the discovery of patterns, trends, and relationships in data. This aids in strategic planning and informed decision-making.
- See also Ke et al. 2024, Cohen and Hunter, 2008

Introduction to Text Mining using Stata

- **Text mining is an interdisciplinary field** that merges techniques from linguistics, computer science, and statistics to extract meaningful information from unstructured textual data. This approach is particularly valuable in economics and health economics where vast amounts of textual data such as academic articles, policy documents, and patient records exist.
- **The process involves several key steps**: data collection, preprocessing of text, feature extraction, and deployment of analytical models. By applying these techniques, researchers can uncover patterns and insights that are not readily apparent, supporting a wide range of applications from market trend analysis to patient outcome studies.

See Aggarwal & Zhai 2012, Gaikwad et al. 2014, Hassan et al. 2022.

Introduction to Text Mining using Stata

- Using Stata for text mining in these domains, researchers leverage its powerful statistical tools along with text-specific commands and user-written ado-files, enabling them to effectively manage and analyze textual data.
- The integration allows for creating **new type of variables** at every level
- This integration allows for **robust analysis of text data**, linking econometric models directly with textual analysis outputs to generate empirically grounded insights.
- **Relevant libraries in Stata** for text mining exists yet there is much space for developing in a growing field (see for instance Provalis Research 2024 and William and Williams 2014 and Schonlau et al. 2017)

Text Mining in Economics

- **The world of business** benefits greatly from text mining, applying it to analyze diverse types of data, such as research papers, news articles, and financial reports
- This analytical power assists in gauging market sentiment, spotting emerging trends, and enhancing the accuracy of economic forecasts.
- Specifically, text mining can uncover terms and phrases indicative of investment behaviors and sentiment shifts, offering economists a valuable tool for predicting market movements.

Examples of Applications of Text Mining in Economics

1. Economic Sentiment Analysis
2. Policy Evaluation
3. Market Trend Analysis
4. Regulatory Compliance Monitoring
5. Consumer Behavior Analysis
6. Economic News Tracking
7. Credit Risk Assessment
8. Labor Market Analysis
9. Intellectual Property and Patent Analysis
10. Macroeconomic Forecasting

See for instance: Kumar & Ravi (2016)

Text Mining in Economics

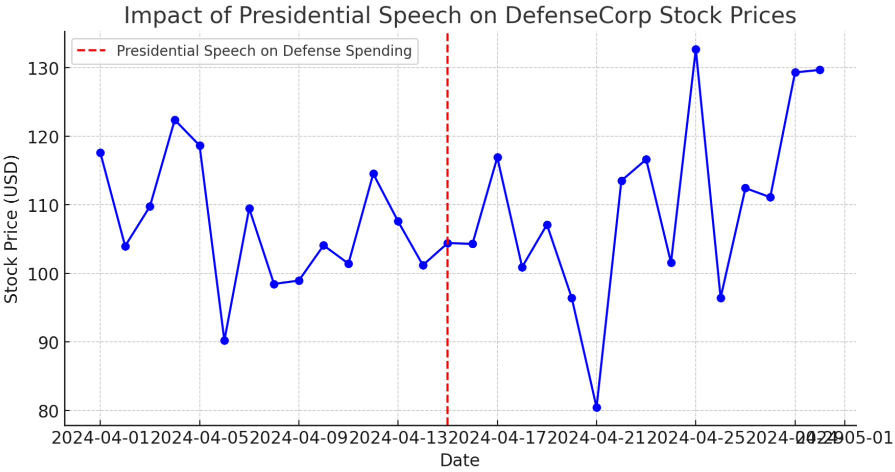


Figure: Impact of a Presidential Speech

Text Mining in Healthcare

- In **healthcare**, text mining proves instrumental in extracting insights from various textual sources, including patient surveys, clinical trials, and medical records.
- This enables a deeper understanding of healthcare utilization patterns, the factors affecting patient outcomes, and the efficacy of different treatments.
- Furthermore, by analyzing electronic health records, text mining can identify trends in disease incidence, treatment effectiveness, and overall healthcare service use.

Example of Applications of Text Mining in Health Economics

1. Cost-Effectiveness Analysis of Treatments
2. Health Policy Formulation
3. Epidemiological Surveillance
4. Patient Satisfaction Analysis
5. Medical Fraud Detection
6. Resource Allocation
7. Pharmaceutical Research
8. Clinical Trial Research Synthesis
9. Healthcare Quality Control
10. Predictive Modelling of Health Outcomes

See for instance: Kim & Chung (2019).

Introduction to Text Mining using EHR Data

- Objective: Analyze electronic health records (EHR) to identify patterns in treatment outcomes.

Age	Gender	Diagnosis	Doctor's Notes
45	Male	Diabetes	Patient reports increased thirst and frequent urination.
30	Female	Hypertension	Blood pressure elevated; medication dosage adjusted.
55	Female	Breast Cancer	Chemotherapy started, patient feeling nauseous.

Note: This data is hypothetical and is used here for illustrative purposes to demonstrate the application of text mining in health economics.

What is Document Clustering?

- Document clustering is a very relevant approach in text mining helpful **to classify and identify different documents based on their content and similarities.**
- Key challenges include dealing with high-dimensional data, managing the variability of language, and the need for sophisticated algorithms to understand context and semantics.
- **Document clustering** groups a set of documents into clusters so that documents in the same cluster are more similar to each other than to those in other clusters. It helps organize, navigate, and summarize a large corpus of text data.

See Steinbach et al. (2000)

Pseudocode Overview for Document Clustering in Stata

Algorithm 1 Data Preprocessing and Cluster Analysis

```
procedure TextDataPreprocessing(texts)
    cleanedTexts ← remove noise(texts)
    tokenizedTexts ← tokenize(cleanedTexts)
    stemmedTexts ← stem(tokenizedTexts)
    return stemmedTexts
end procedure
procedure MultipleCorrespondence(data)
    factorScores ← MCA(data)
    return factorScores
end procedure
procedure ClusterAnalysis(data, numClusters)
    clusters ← hierarchical clustering(data, numClusters)
    return clusters
end procedure
```

Pseudocode for Cluster Validation in Stata

Algorithm 2 Cluster Validation

```
procedure ValidateClusters($clusters, data$)
    $davies \leftarrow$ calculate Caliński-Harabasz($clusters, data$)
    return $davies$
end procedure
$validation \leftarrow$ ValidateClusters($clusterResult, scores$)
Output the results of Caliński-Harabasz index validation
```

Representation of a Corpus in Text Mining

In text mining, a *corpus* C is defined as a collection of documents:

$$C = \{d_1, d_2, \ldots, d_n\}$$

where:

- C is the corpus, encompassing a set of documents.
- Each d_i is a document within C.
- n represents the total number of documents in the corpus.

Each document d_i can be further described as a sequence of terms:

$$d_i = [t_{i1}, t_{i2}, \ldots, t_{im}]$$

where:

- t_{ij} denotes a term in the i^{th} document.
- m is the number of terms in d_i.

Representation of a Corpus in Text Mining

Vector Space Model: Documents are often represented as vectors in a multidimensional space, where each dimension corresponds to a unique term in the corpus, facilitating the application of machine learning and statistical analysis.

$$\vec{d_i} = (w_{i1}, w_{i2}, \ldots, w_{ik})$$

- $\vec{d_i}$ is the vector representation of d_i.
- w_{ij} is the weight of term j in d_i, such as term frequency or TF-IDF score.
- k represents the number of unique specific terms in the corpus.

See Aswani & Srinivas (2009) and Wagner et al. (2012)

Preprocessing Corpus in Text Mining

Preprocessing of the corpus is a critical initial step in text mining, involving several techniques to transform raw text into a structured form able to to be uses in the analysis. This process enhances data quality and analysis efficiency. Key preprocessing steps include:

1. **Tokenization:** Segmenting text into meaningful elements such as words, phrases, or symbols.
2. **Stop Words Removal:** Eliminating common words (e.g., "the", "is", "and") that offer little value for analysis.
3. **Stemming and Lemmatization:** Reducing words to their base or root form. While stemming cuts off prefixes and suffixes, lemmatization considers the morphological analysis of the words.
4. **Normalization:** Standardizing text by converting it to a uniform case (usually lowercase), removing punctuation, and correcting misspellings.
5. **Vectorization:** Transforming text into numerical values or vectors, enabling statistical or machine learning models to process the text data. Common methods include Count Vectorization and TF-IDF

Preprocessing Corpus in Text Mining

- Preprocessing techniques are often tailored to the specific requirements of the text mining task at hand, considering the nature of the text data and the objectives of the analysis. See Toman et al. (2006), HaCohen-Kerner et al. (2020).
- **A term-document matrix** represents text data as a matrix of term frequencies. Each row corresponds to a term, and each column corresponds to a document.
- This structure is foundational for many text mining techniques. See Anandarajan et al. (2019).

Term-Document Matrix: Representation

A term-document matrix, T, is a mathematical representation where each term t_i in the dataset corresponds to a row i, and each document d_j corresponds to a column j. The entry T_{ij} in the matrix represents the frequency of term t_i in document d_j.

$$T = \begin{array}{c} \\ t_1 \\ t_2 \\ \vdots \\ t_m \end{array} \begin{array}{c} \begin{array}{cccc} d_1 & d_2 & \cdots & d_n \end{array} \\ \begin{pmatrix} f_{11} & f_{12} & \cdots & f_{1n} \\ f_{21} & f_{22} & \cdots & f_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ f_{m1} & f_{m2} & \cdots & f_{mn} \end{pmatrix} \end{array}$$

Where:

- f_{ij} is the frequency of term t_i in document d_j.
- n is the number of documents.
- m is the number of distinct terms across all documents.

This structure facilitates the application of various statistical and machine learning methods to extract insights from text data.

Importance of the Term-Document Matrix

The term-document matrix serves as a foundation for many text mining and natural language processing (NLP) tasks, enabling:

- Identification of prevalent themes across documents through topic modeling.
- Analysis of document similarity for clustering or classification.
- Sentiment analysis by assessing the presence of positive or negative terms.

Furthermore, variations such as TF-IDF (Term Frequency-Inverse Document Frequency) adjust the raw frequencies to highlight terms that are particularly relevant to a document.
See Antonellis & Gallopoulos (2006)

TF-IDF Weighting

Term Frequency-Inverse Document Frequency (TF-IDF) is a numerical statistic that reflects how important a word is to a document in a collection or corpus.

$$\text{TF-IDF}(i,j) = TF(i,j) \times IDF(i)$$

where:

$$TF(i,j) = \frac{\text{Number of times term } i \text{ appears in document } j}{\text{Total number of terms in document } j}$$

$$IDF(i) = \log\left(\frac{\text{Total number of documents}}{\text{Number of documents containing term } i}\right)$$

See Chen et al. (2016)

Term-Document Matrix in Stata

- It's possible to adopt **procedures in Stata** and in particular the to preprocessing text data (using txttool by Williams and Williams 2014) and constructing a document-term matrix (DTM), which is crucial to perform many text mining applications
- Alternatively it's possible to use the **Stata integration with Python language** which allow to transform and preprocess the textual data in a document-term matrix

See also Provalis Research (2024)

Overview of Multiple Correspondence Analysis (MCA)

Multiple Correspondence Analysis (MCA) is a statistical technique for dimensionality reduction (Greenacre & Blasius 2006), designed to analyze and visualize the patterns in categorical data. It generalizes simple correspondence analysis to higher-dimensional data tables.

Key Steps:

1. **Indicator Matrix Construction:** For a set of categorical variables, an indicator (or dummy) matrix X is created, where rows correspond to observations and columns to categories across all variables.
2. **Distance Metric:** MCA employs the chi-square metric on X to measure distances between categorical profiles, ensuring that the analysis is sensitive to variations in data distribution.
3. **Singular Value Decomposition (SVD):** MCA applies SVD to the centered and normalized indicator matrix:

$$X^* = UDV^T$$

Multiple Correspondence Analysis (MCA) of the Term-Document Matrix

Multiple Correspondence Analysis (MCA see also Abdi & Valentin 2007) is a technique for dimensionality reduction. that is particularly suited to analyzing the high-dimensional categorical data found in a term-document matrix. It extends Correspondence Analysis (CA) to accommodate multiple categorical variables, offering a way to visualize and interpret the complex relationships in textual data.

Application to Term-Document Matrix

Given a term-document matrix T, MCA enables us to:

- Identify patterns and associations between terms and documents.
- Reduce the dimensionality of T to a few synthetic variables (dimensions) that capture the major part of the information in the matrix.
- Facilitate the visualization of documents and terms in a lower-dimensional space, often in two or three dimensions.

Multiple Correspondence Analysis (MCA) of the Term-Document Matrix

Foundation

MCA treats each term and document as categorical variables and applies singular value decomposition (SVD) to the indicator matrix derived from T, producing a set of orthogonal factors that summarize the data. The reduced space is then used to analyze and visualize the data's structure.

Through MCA, complex and high-dimensional term-document matrices can be transformed into more manageable and interpretable forms, aiding in the discovery of underlying structures and relationships in textual data.

Clustering Documents in Stata

- Clustering involves grouping documents in such a way that documents in the same group are more similar to each other than to those in other groups.
- This is typically achieved through the use of algorithms like hierarchical clustering and K-means.

Hierarchical Clustering with Euclidean Distance

Hierarchical clustering is a technique that groups data over a series of partitions. It does not require a pre-specified number of clusters. Instead, it creates a hierarchical decomposition of the dataset using a bottom-up (agglomerative) or top-down (divisive) approach. Using Euclidean distance, it measures the distances between data points to determine their similarity. See Johnson (1967)

Euclidean Distance

The Euclidean distance between two points x and y in a multidimensional space is defined as:

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

where x_i and y_i are the coordinates of x and y respectively, and n is the number of dimensions.

Hierarchical Clustering with Euclidean Distance

Agglomerative Hierarchical Clustering Process

1. Start with each data point as a separate cluster.
2. Find the two clusters that are closest together (using Euclidean distance as the metric), and merge them into a single cluster.
3. Once all data points are merged into one cluster, repeat step 2 until all the data points have been merged into one.
4. The result is a tree-based representation of the observations, called a dendrogram, which shows the order and distance (similarity) of merges.

Euclidean distance provides a straightforward geometric interpretation of distance, making it intuitive for clustering in many real-world applications. However, it's important to note that it is sensitive to the scale of the data, hence preprocessing steps such as normalization may be required.

The Dendrogram

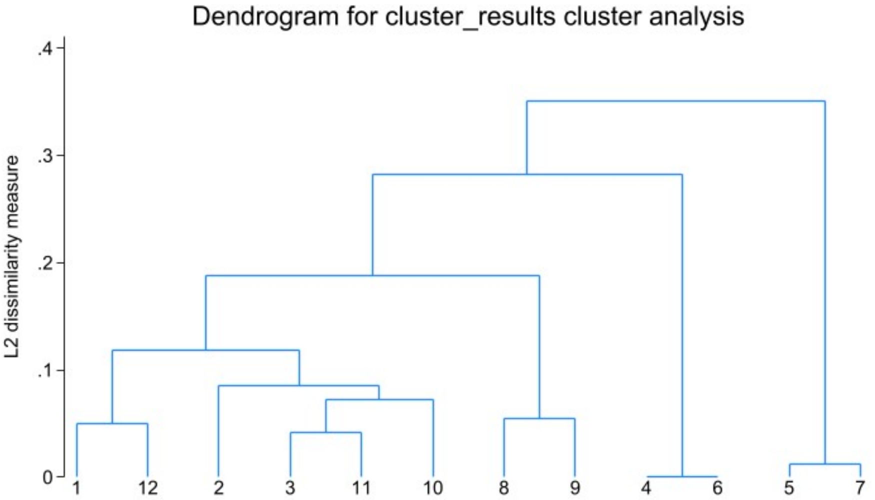


Figure: Dendrogram for the documents clustering

Determinating the Optimal Number of Document Clusters

- The Caliński-Harabasz index, also known as the Variance Ratio Criterion, is used to evaluate the quality of clusters.
- It helps in determining the optimal number of clusters in hierarchical clustering and other methods.

See Wang and Xu 2019 and Halpin 2015,

Caliński-Harabasz Index: Definitions

- **Between-Cluster Variance** (B): This metric quantifies the variance or scatter of the cluster centroids (μ_j) around the grand centroid ($\bar{x}$). It assesses how well-separated the different clusters are, with higher values indicating more distinct clusters.
- **Within-Cluster Variance** (W): This metric captures the variance or scatter of the data points (x_i) within each cluster around their respective cluster centroids (μ_j). Lower values suggest that the clusters are compact, which generally indicates better clustering performance.

Caliński-Harabasz Index: Scatter Matrices

- **Total Scatter Matrix** (T): Defined as

$$T = \sum_{i=1}^{n} (x_i - \bar{x})^T (x_i - \bar{x}),$$

  this matrix quantifies the overall variance within the dataset, encapsulating the total scatter of all data points (x_i) around the dataset's grand mean ($\bar{x}$). It serves as a baseline measure of data dispersion, fundamental for normalizing the between-cluster scatter.

- **Between-Cluster Scatter Matrix** (B): Expressed as

$$B = \sum_{j=1}^{k} n_j (\mu_j - \bar{x})^T (\mu_j - \bar{x}),$$

  this matrix measures the weighted scatter of the cluster centroids (μ_j), relative to the grand mean ($\bar{x}$), with weights corresponding to the cluster sizes (n_j). It highlights the extent of separation among different clusters, crucial for assessing the effectiveness of a clustering

Caliński-Harabasz Index: Internal Cluster Cohesion

- **Within-Cluster Scatter Matrix** (W):

$$W = \sum_{j=1}^{k} \sum_{i \in C_j} (x_i - \mu_j)^T (x_i - \mu_j),$$

- this matrix captures the sum of squared distances between each data point (x_i) and its cluster centroid (μ_j), aggregated across all clusters. The within-cluster scatter matrix is integral to evaluating the compactness and homogeneity of clusters, serving as a fundamental measure in cluster validation.
- It assesses how tightly grouped the data points are within each cluster, which is essential for determining the effectiveness of the clustering algorithm in segmenting similar observations into coherent groups.

Caliński-Harabasz Index: Assessing Cluster Validity

The Caliński-Harabasz index, a measure of clustering efficacy, is calculated by the formula:

$$CH = \frac{B/(k-1)}{W/(n-k)}$$

- The numerator, $\frac{B}{k-1}$, calculates the mean between-cluster variance for k clusters, where B encapsulates the total variance between clusters' centroids relative to the overall centroid, and $k-1$ indicates the degrees of freedom, reflecting the number of clusters minus one.
- The denominator, $\frac{W}{n-k}$, measures the mean within-cluster variance, with W representing the accumulated variances within each cluster, and $n-k$ the degrees of freedom, accounting for the total number of data points minus the number of clusters.

Caliński-Harabasz Index: Assessing Cluster Validity

This index is particularly valuable for comparing the effectiveness of different clustering schemes, with higher values indicating better-defined clustering by maximizing between-cluster variance and minimizing within-cluster variance.

Caliński-Harabasz Index: Clustering Validation

- A higher *CH* index suggests that the clusters are dense and well-separated, indicating that the clustering structure is appropriately capturing distinct groups within the data.
- The optimal number of clusters k is the one that maximizes the *CH* index. This maximization reflects the point where the clusters are most distinctly separable and internally homogeneous, providing a strong heuristic for determining the number of clusters in a dataset.
- The Caliński-Harabasz index provides a numerical measure to evaluate different clustering models.
- It is effective in selecting the most appropriate number of clusters especially in complex datasets.

Challenges in Clustering

- Document clustering faces challenges such as choosing the right number of clusters, dealing with high-dimensional data, and ensuring that the algorithm converges to a meaningful solution.
- Another relevant challenge having assigned to each observation a label it is important to **interpret the cluster identified as a whole**

Case Studies

- A brief overview of a practical application of document clustering, demonstrating its utility in organizing large datasets of text documents into coherent groups.
- We simulate different documents as textual data to be clustered.

Experimental Setup for Classifying Sector-Based Economic Reports

Objective: The main objective of this experiment is to classify textual documents into a category of 'economic reports' based on their content related to specific market sectors.

Data Collection: Our dataset consists of several statements reflecting sector-specific performance. These include mentions of technological advancements in tech stocks, fluctuating oil prices impacting the energy sector, and trends in healthcare demand, among others. Each document is a single sentence summarizing quarterly performance across different industries.

Text Preprocessing: Prior to analysis, we perform standard text preprocessing steps using Stata's text utilities:

- *Tokenization* to break down text into individual words or terms.
- *Removing stopwords* to eliminate common words that add no value to the classification.
- *Stemming* to reduce words to their root forms.

Experimental Setup for Classifying Sector-Based Economic Reports

Feature Extraction: We employ the bag-of-words model to transform text data into numerical features suitable for modeling. This involves creating a matrix where each row represents a document and each column represents a unique word, with cell values denoting the frequency of the word in the document.
MCA and Cluster Analysis: Using Stata we perform multiple correspondence analysis and then on the dimension extracted we classify the different documents.
Validation: The classification was validated using appropriate approaches.
Results Discussion: we will present the results, discuss the implications of our findings, and explore potential applications of this model in economic research.

Results: the Dendrogram

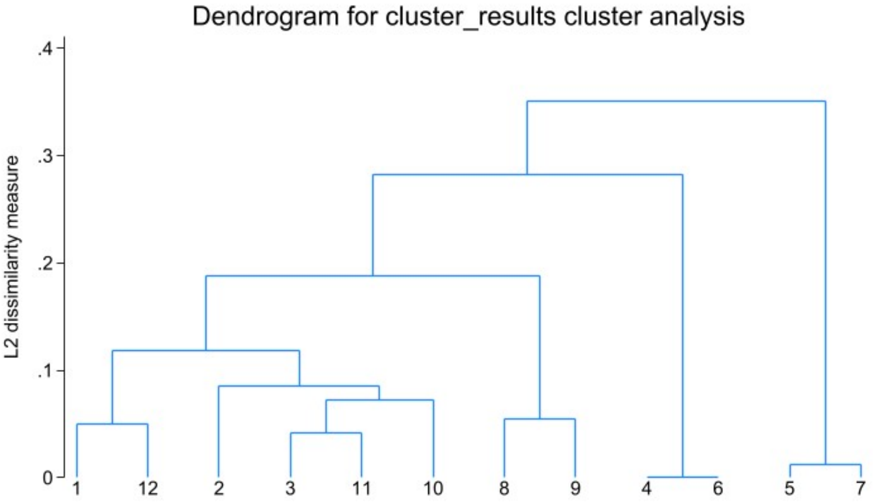


Figure: Dendrogram

Validation of Cluster Analysis

Number of Clusters	Calinski/Harabasz Pseudo-F
2	8.70
3	17.19
4	32.98
5	55.66

This table presents the Calinski-Harabasz index values for different numbers of clusters, illustrating the statistical validity of cluster solutions in our analysis. Higher values generally indicate better cluster separation and validity.

Experiment Design for Classifying Medical EMRs Using Text Mining

Objective:

- To automatically classify EMRs into relevant medical categories such as diagnosis, treatment plans, and patient demographics using text mining techniques.

Data Set:

- **Corpus Description:** The data set consists of 20 realistic EMR entries covering a range of medical scenarios.
- **Source:** Synthetic records created to mimic real-world medical documentation.

Experiment Design for Classifying Medical EMRs Using Text Mining

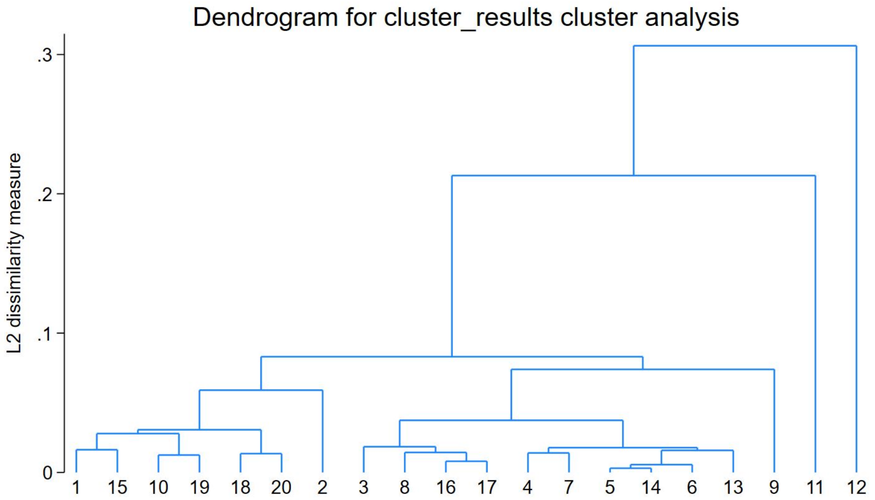
Methodology:

1. **Preprocessing:**
   - Tokenization, Normalization, Stop Word Removal, Stemming.
2. **Feature Extraction:**
   - Vectorization using TF-IDF.
3. **Clustering** item **Cluster Validation**

Expected Outcomes:

- Categorizing medical records.

Experimental Setup for Classifying Electronic Medical Records (EMR)



Cluster Validation

This table shows the Calinski-Harabasz pseudo-F values for different numbers of clusters in a dataset:

Number of Clusters	**Calinski-Harabasz Pseudo-F**
2	18.14
3	25.98
4	60.69
5	67.73

Cluster Validation

- The table lists the Calinski-Harabasz pseudo-F values for different numbers of clusters. The Calinski-Harabasz index is a method for evaluating the quality of clustering. It is calculated as the ratio of the sum of between-clusters dispersion to within-cluster dispersion for different clusters, multiplied by the factor that adjusts for the number of clusters and data points.
- Higher Values Indicate Better Clustering: As the number of clusters increases from 2 to 5, the Calinski-Harabasz pseudo-F values also increase, suggesting that each increase in the number of clusters provides a statistically more distinct grouping of the data points. The highest value is obtained for 5 clusters (67.73), indicating that dividing the data into five groups provides the best differentiation among them based on this metric.

Cluster Validation

- Choosing the Number of Clusters: Although the index continues to increase, the choice of the number of clusters often depends on other factors as well, such as the interpretability of the clusters in the context of the data, computational constraints, and the diminishing returns of increasing cluster numbers. For instance, while 5 clusters have the highest score, it might be more practical or interpretable to choose fewer clusters depending on the situation.
- These results could be very useful in determining the optimal number of clusters for data analysis tasks as in our case classifying Electronic Medical Records (EMR)

Identifying Optimal Number of Clusters and Assignment

Objective: Determine the optimal number of clusters and accurately assign observations.

Step 1: Identify Optimal Cluster Number

- Use the Calinski-Harabasz criterion to evaluate the optimal cluster count between 2 and 5.
- Command: `cluster stop cluster_results, rule(calinski) groups(2/5)`
- The criterion maximizes the ratio of between-cluster variance to within-cluster variance, suggesting a more distinct grouping.

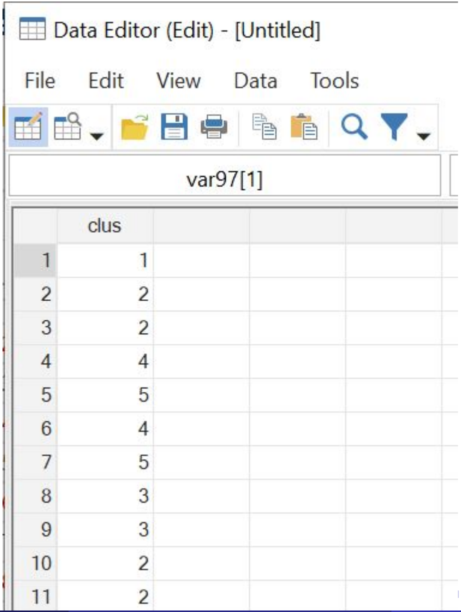
Identifying Optimal Number of Clusters and Assignment

Step 2: Assign Observations to Clusters

- Assign observations to 5 clusters as identified as optimal in Step 1.
- Command: `cluster generate clus = groups(5), name(cluster_results) ties(error)`
- The `ties(error)` option ensures that no ambiguous assignments are made. An error is raised if observations are equidistant to multiple clusters, prompting a review.

Conclusion: This methodical approach using the Calinski-Harabasz rule and strict tie handling ensures precise and reliable clustering, critical for subsequent analysis.

Cluster Membership in Stata



var97[1]

	clus
1	1
2	2
3	2
4	4
5	5
6	4
7	5
8	3
9	3
10	2
11	2

Conclusions

- **Critical Role in Text Mining:** Document clustering is instrumental in organizing unstructured text data, which enhances information retrieval and facilitates comprehensive data analysis.
- **Text Mining Analyses in Stata:** A relevant advantage on using Stata in Text Mining is the possibility to integrate different Stata functionalities with text mining analyses (for instance text mining can allow to construct relevant variables by documents. This could be very powerful.
- **Enhancements through Stata:** Leveraging Stata for methodological improvements has proven to significantly boost processing speeds and improve the accuracy of clustering outcomes.

Conclusions

- **Statistical Challenges:** Issues like determining the optimal number of clusters remain important challenge that drive ongoing research in the field.
- **Practical Applications:** The advancements in document clustering have substantial implications across various industries, improving the way organizations manage and interpret large datasets.

References

- Abdi, H., & Valentin, D. (2007). Multiple correspondence analysis. Encyclopedia of measurement and statistics, 2(4), 651-657.
- Aggarwal, C., & Zhai, C. (2012). Mining Text Data. , 429-455. https://doi.org/10.1007/978-1-4614-3223-4.
- Anandarajan, M., Hill, C.,& Nolan, T. (2019). Term-document representation. Practical Text Analytics: Maximizing the Value of Text Data, 61-73.
- Antonellis, I., & Gallopoulos, E. (2006). Exploring term-document matrices from matrix models in text mining. ArXiv, abs/cs/0602076.
- Aswani Kumar, C., & Srinivas, S. (2009). On the performance of latent semantic indexing-based information retrieval. Journal of computing and information technology, 17(3), 259-264.
- Chen, K., Zhang, Z., Long, J., & Zhang, H. (2016). Turning from TF-IDF to TF-IGM for term weighting in text classification. Expert Syst. Appl., 66, 245-260. https://doi.org/10.1016/j.eswa.2016.09.009.

References

- Cohen, K. B. and Hunter, L. (2008). Getting started in text mining. PLoS Computational Biology, 4(1), e20. https://doi.org/10.1371/journal.pcbi.0040020
- Errichiello, L. and Drago C. (2024) Remote Work and Gender during the COVID-19 Pandemic: A Literature Review Through LDA-Based Topic Modelling, Working Paper.
- Gaikwad, S. V., Chaugule, A., & Patil, P. (2014). Text mining methods and techniques. International Journal of Computer Applications, 85(17).
- Greenacre, M., & Blasius, J. (2006). Multiple correspondence analysis and related methods. Chapman and Hall/CRC.
- HaCohen-Kerner, Y., Miller, D., & Yigal, Y. (2020). The influence of preprocessing on text classification using a bag-of-words representation. PloS one, 15(5), e0232525.
- Halpin, B. (2015). CALINSKI: Stata module to compute Calinski-Harabasz cluster stopping index from distance matrix. Statistical Software Components.

References

- Hassan, S. U., Ahamed, J., & Ahmad, K. (2022). Analytics of machine learning-based algorithms for text classification. Sustainable operations and computers, 3, 238-248.
- Hotho, A., Nürnberger, A., & Paass, G. (2005). A Brief Survey of Text Mining. Journal for Language Technology and Computational Linguistics. https://doi.org/10.21248/jlcl.20.2005.68.
- Johnson, S. (1967). Hierarchical clustering schemes. Psychometrika, 32, 241-254. https://doi.org/10.1007/BF02289588.
- Kim, J. C., & Chung, K. (2019). Associative feature information extraction using text mining from health big data. Wireless Personal Communications, 105, 691-707.
- Ke, Z. T., Ji, P., Jin, J., & Li, W. (2023). Recent advances in text analysis. Annual Review of Statistics and Its Application, 11.
- Kumar, B. S., & Ravi, V. (2016). A survey of the applications of text mining in financial domain. Knowledge-Based Systems, 114, 128-147.

References

- Provalis Research (2024) WordStat https://provalisresearch.com/products/content-analysis-software/
- Schonlau, M., Guenther, N. Sucholutsky, I. Text mining using n-gram variables. The Stata Journal. Dec 2017, 17(4), 866-881
- Steinbach, M., Karypis, G., & Kumar, V. (2000). A comparison of document clustering techniques.
- Tange, H. J., Hasman, A., de Vries Robbé, P. F., & Schouten, H. C. (1997). Medical narratives in electronic medical records. International journal of medical informatics, 46(1), 7-29.

References

- Toman, M., Tesar, R., & Jezek, K. (2006). Influence of word normalization on text classification. Proceedings of InSciT, 4(354-358), 9.
- Venkataraman, G. R., Pineda, A. L., Bear Don't Walk IV, O. J., Zehnder, A. M., Ayyar, S., Page, R. L., ... & Rivas, M. A. (2020). FasTag: Automatic text classification of unstructured medical narratives. PLoS one, 15(6), e0234647.
- Wagner, H., Dłotko, P., & Mrozek, M. (2012). Computational topology in text mining. In Computational Topology in Image Context: 4th International Workshop, CTIC 2012, Bertinoro, Italy, May 28-30, 2012. Proceedings (pp. 68-78). Springer Berlin Heidelberg.
- Wang, X., & Xu, Y. (2019, July). An improved index for clustering validation based on Silhouette index and Calinski-Harabasz index. In IOP Conference Series: Materials Science and Engineering (Vol. 569, No. 5, p. 052024). IOP Publishing.

References

- Williams, U., & Williams, S. P. (2014). txttool: Utilities for text analysis in Stata. The Stata Journal, 14(4), 817-829.