150B/355B
Introduction to Machine Learning for Social Science
TA Section 8

Haemin Jee and Tongtong Zhang

March 2, 2018

Road Map

1. K-means clustering
2. Topic modeling
3. R Exercise

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters.

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters. $\rightsquigarrow$ Unsupervised Learning

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters. $\rightsquigarrow$ Unsupervised Learning

How does unsupervised learning differ from supervised learning?

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters. $\rightsquigarrow$ Unsupervised Learning

How does unsupervised learning differ from supervised learning?

In supervised learning, we know what are the meaningful classes (MENA vs. West; Positive vs. Negative), and the goal is to assign each document into one of these pre-defined classes using tools like regressions, dictionary method (dinstinctive words).

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters. $\rightsquigarrow$ Unsupervised Learning

How does unsupervised learning differ from supervised learning?

In supervised learning, we know what are the meaningful classes (MENA vs. West; Positive vs. Negative), and the goal is to assign each document into one of these pre-defined classes using tools like regressions, dictionary method (dinstinctive words).
$\rightarrow$ check the quality of result by comparing with human coding

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters. $\rightsquigarrow$ Unsupervised Learning

How does unsupervised learning differ from supervised learning?

In supervised learning, we know what are the meaningful classes (MENA vs. West; Positive vs. Negative), and the goal is to assign each document into one of these pre-defined classes using tools like regressions, dictionary method (dinstinctive words).
$\rightarrow$ check the quality of result by comparing with human coding

In unsupervised learning, we do not know what are the meaningful classes and the goal is to figure out the meaningful classes while assigning each document into one of them at the same time.

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters. $\rightsquigarrow$ Unsupervised Learning

How does unsupervised learning differ from supervised learning?

In supervised learning, we know what are the meaningful classes (MENA vs. West; Positive vs. Negative), and the goal is to assign each document into one of these pre-defined classes using tools like regressions, dictionary method (dinstinctive words).
$\rightarrow$ check the quality of result by comparing with human coding

In unsupervised learning, we do not know what are the meaningful classes and the goal is to figure out the meaningful classes while assigning each document into one of them at the same time.
$\rightarrow$ no clear way to check the quality of result

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters.

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters.

- similar (small geometric distance) documents are in the same cluster

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters.

- similar (small geometric distance) documents are in the same cluster
- dissimilar (large geometric distance) documents are apart

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters.

- similar (small geometric distance) documents are in the same cluster
- dissimilar (large geometric distance) documents are apart

Last week: distance between documents (euclidean, cosine)

Clustering

Task: Partition a set of unlabeled documents into meaningful classes / clusters.

- similar (small geometric distance) documents are in the same cluster
- dissimilar (large geometric distance) documents are apart

Last week: distance between documents (euclidean, cosine)

This week: using these distance metrics, how do we find a good partition of documents?

K-Means Clustering

Task: Assign each document into exactly one cluster

K-Means Clustering

Task: Assign each document into exactly one cluster

Inputs

1. A document term matrix (or any multidimensional dataset)
2. K: the desired number of clusters.

K-Means Clustering

Task: Assign each document into exactly one cluster

Inputs

1. A document term matrix (or any multidimensional dataset)
2. K: the desired number of clusters.

Outputs

1. C_k: The set of documents assigned to each cluster.
2. μ_k: The mean for each K – a vector representing the average values of all documents in that cluster. Also called centroid.

K-Means Clustering

Task: Assign each document into exactly one cluster

Inputs

1. A document term matrix (or any multidimensional dataset)
2. K: the desired number of clusters.

Outputs

1. C_k: The set of documents assigned to each cluster.
2. μ_k: The mean for each K – a vector representing the average values of all documents in that cluster. Also called centroid.

   Q: What is the dimension of a centroid?

K-Means Clustering

Algorithm

K-Means Clustering

Algorithm

1) Randomly initialize K cluster centroids $(\mu_1, \mu_2, \cdots, \mu_k)$ in the multi-dimensional space

K-Means Clustering

Algorithm

1) Randomly initialize K cluster centroids $(\mu_1, \mu_2, \cdots, \mu_k)$ in the multi-dimensional space
2) Repeat:
   - Assignment: Given the centroids, assign each document $\boldsymbol{X}$ to the cluster that has the closest centroid μ_k with $\boldsymbol{X}$.
   - Update: Calculate new centroids μ_k by averaging all documents assigned to each cluster.

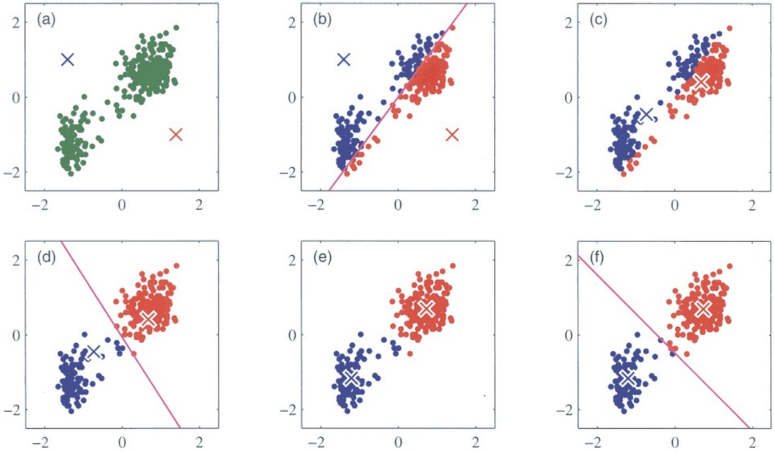
K-Means Clustering

Algorithm

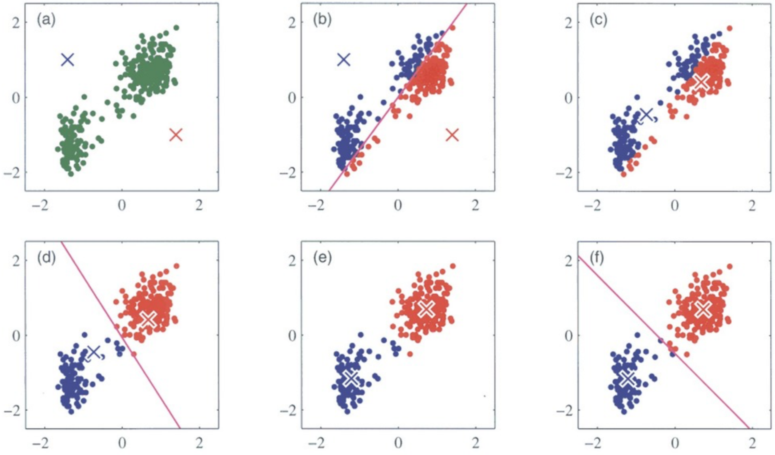
1) Randomly initialize K cluster centroids $(\mu_1, \mu_2, \cdots, \mu_k)$ in the multi-dimensional space
2) Repeat:
   - Assignment: Given the centroids, assign each document $\boldsymbol{X}$ to the cluster that has the closest centroid μ_k with $\boldsymbol{X}$.
   - Update: Calculate new centroids μ_k by averaging all documents assigned to each cluster.

   Stop when cluster assignments stop changing.

K-Means Clustering



K-Means Clustering



The algorithm stops when the partition boundary (red line) stops changing.

K-Means Clustering

Small Decisions with Big Consequences:

K-Means Clustering

Small Decisions with Big Consequences:

1) How should we preprocess the data?
   - k-means are very sensitive to feature scaling / weighting.
   - Common to normalize the DTM in some way, e.g. by dividing each vector by the vector length.

K-Means Clustering

Small Decisions with Big Consequences:

1) How should we preprocess the data?
   - k-means are very sensitive to feature scaling / weighting.
   - Common to normalize the DTM in some way, e.g. by dividing each vector by the vector length.
2) How to chose K?
   - User must assign the number of clusters (K)
   - Different values of K will lead to different partitions.

K-Means Clustering

Small Decisions with Big Consequences:

1) How should we preprocess the data?
    - k-means are very sensitive to feature scaling / weighting.
    - Common to normalize the DTM in some way, e.g. by dividing each vector by the vector length.
2) How to chose K?
    - User must assign the number of clusters (K)
    - Different values of K will lead to different partitions.
3) Random starting values!
    - Results will depend on the initial (random) cluster centroid assignment.

K-Means Clustering

Small Decisions with Big Consequences:

1) How should we preprocess the data?
    - k-means are very sensitive to feature scaling / weighting.
    - Common to normalize the DTM in some way, e.g. by dividing each vector by the vector length.
2) How to chose K?
    - User must assign the number of clusters (K)
    - Different values of K will lead to different partitions.
3) Random starting values!
    - Results will depend on the initial (random) cluster centroid assignment.

In practice, it's important to run the algorithm multiple times with different preprocessing methods, different K, and from different random starting values.

K-Means Clustering

Small Decisions with Big Consequences:

1) How should we preprocess the data?
   - k-means are very sensitive to feature scaling / weighting.
   - Common to normalize the DTM in some way, e.g. by dividing each vector by the vector length.
2) How to chose K?
   - User must assign the number of clusters (K)
   - Different values of K will lead to different partitions.
3) Random starting values!
   - Results will depend on the initial (random) cluster centroid assignment.

In practice, it's important to run the algorithm multiple times with different preprocessing methods, different K, and from different random starting values.

Suppose we ran the algorithm using different starting values, given K and the preprocessing method, how can we know which starting value we should use?

K-Means Clustering

Two kinds of validation criteria:

K-Means Clustering

Two kinds of validation criteria:

1. Quantitative evaluation:
   - A good clustering is one for which the distance between documents within the same cluster is as small as possible.

K-Means Clustering

Two kinds of validation criteria:

1. Quantitative evaluation:
   - A good clustering is one for which the distance between documents within the same cluster is as small as possible.
2. Qualitative evaluation:
   - A good clustering is one for which clusters are substantially / semantically interpretable.

K-Means Clustering: Quantitative Evaluation

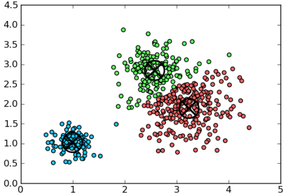
Sum of squared Euclidean distance

K-Means Clustering: Quantitative Evaluation

Sum of squared Euclidean distance

Step 1: For document $\boldsymbol{X}$, calculate its Euclidean distance with the centroid ($\boldsymbol{\mu_k}$) of the cluster it is assigned to:

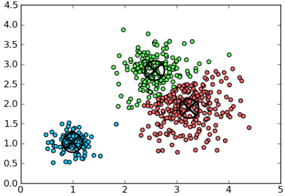
$$D(\boldsymbol{X}, \boldsymbol{\mu_k})^2 = \sum_{p=1}^{P} (x_p - \mu_{kp})^2$$



K-Means Clustering: Quantitative Evaluation

Sum of squared Euclidean distance

Step 1: For document $\boldsymbol{X}$, calculate its Euclidean distance with the centroid ($\boldsymbol{\mu_k}$) of the cluster it is assigned to:



$$D(\boldsymbol{X}, \boldsymbol{\mu_k})^2 = \sum_{p=1}^{P} (x_p - \mu_{kp})^2$$

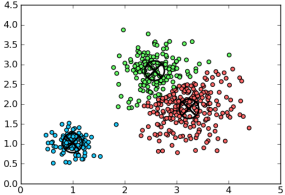
Step 2: Repeat step 1 for each document and sum up all distances

$$W(C_k) = \sum_{i \in C_k} D(\boldsymbol{X}_i, \boldsymbol{\mu_k})^2$$

K-Means Clustering: Quantitative Evaluation

Sum of squared Euclidean distance

Step 1: For document $\boldsymbol{X}$, calculate its Euclidean distance with the centroid ($\boldsymbol{\mu_k}$) of the cluster it is assigned to:



$$D(\boldsymbol{X}, \boldsymbol{\mu_k})^2 = \sum_{p=1}^{P} (x_p - \mu_{kp})^2$$

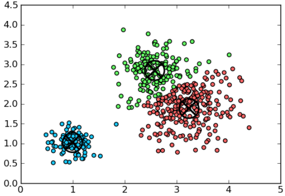
Step 2: Repeat step 1 for each document and sum up all distances

$$W(C_k) = \sum_{i \in C_k} D(\boldsymbol{X}_i, \boldsymbol{\mu_k})^2$$

$$\boldsymbol{W} = \sum_{k=1}^{K} W(C_k)$$

Sum of squared Euclidean distance

Step 1: For document $\boldsymbol{X}$, calculate its Euclidean distance with the centroid ($\boldsymbol{\mu_k}$) of the cluster it is assigned to:



$$D(\boldsymbol{X}, \boldsymbol{\mu_k})^2 = \sum_{p=1}^{P} (x_p - \mu_{kp})^2$$

Step 2: Repeat step 1 for each document and sum up all distances

$$W(C_k) = \sum_{i \in C_k} D(\boldsymbol{X}_i, \boldsymbol{\mu_k})^2$$

$$\boldsymbol{W} = \sum_{k=1}^{K} W(C_k)$$

We want to choose the clustering result that minimizes $\boldsymbol{W}$.

K-Means Clustering: Qualitative Evaluation

Clusters are substantially / semantically interpretable.

K-Means Clustering: Qualitative Evaluation

Clusters are substantially / semantically interpretable.

1. Manual identification
   - Sample set of documents from same cluster
   - Read documents
   - Assign cluster "label" by hand

K-Means Clustering: Qualitative Evaluation

Clusters are substantially / semantically interpretable.

1. Manual identification
   - Sample set of documents from same cluster
   - Read documents
   - Assign cluster "label" by hand
2. Automatic identification
   - Use methods to identify distinctive words between clusters
   - Use these words to infer the semantic meaning of a cluster

K-Means Clustering: Qualitative Evaluation

Clusters are substantially / semantically interpretable.

1. Manual identification
   - Sample set of documents from same cluster
   - Read documents
   - Assign cluster "label" by hand
2. Automatic identification
   - Use methods to identify distinctive words between clusters
   - Use these words to infer the semantic meaning of a cluster
3. Be Transparent
   - Provide documents + code
   - Detail labeling procedures
   - Acknowledge ambiguity

K-Means Clustering: Qualitative Evaluation

Clusters are substantially / semantically interpretable.

1. Manual identification
   - Sample set of documents from same cluster
   - Read documents
   - Assign cluster "label" by hand
2. Automatic identification
   - Use methods to identify distinctive words between clusters
   - Use these words to infer the semantic meaning of a cluster
3. Be Transparent
   - Provide documents + code
   - Detail labeling procedures
   - Acknowledge ambiguity

`R code, Section 1`

Topic Modeling

In clustering, we assign each article to only one topic.

Topic Modeling

In clustering, we assign each article to only one topic.

In topic modeling, we represent each article as a mixture of topics

Topic Modeling

In clustering, we assign each article to only one topic.

In topic modeling, we represent each article as a mixture of topics $\rightsquigarrow$ unsupervised method

Topic Modeling

In clustering, we assign each article to only one topic.

In topic modeling, we represent each article as a mixture of topics $\rightsquigarrow$ unsupervised method

- Describe each topic, defined by a group of distinctive words / high-frequency words.

Topic Modeling

In clustering, we assign each article to only one topic.

In topic modeling, we represent each article as a mixture of topics $\rightsquigarrow$ unsupervised method

- Describe each topic, defined by a group of distinctive words / high-frequency words.
- Measure proportion of each article addressing each topic.

Topic Modeling

In clustering, we assign each article to only one topic.

In topic modeling, we represent each article as a mixture of topics $\rightsquigarrow$ unsupervised method

- Describe each topic, defined by a group of distinctive words / high-frequency words.
- Measure proportion of each article addressing each topic.

Method: Latent Dirichlet Allocation (LDA); Structural Topic Modeling (STM)

Latent Dirichlet Allocation (LDA)

Inputs

1. A document term matrix (or any multidimensional dataset)
2. K: the desired number of topics.

Latent Dirichlet Allocation (LDA)

Outputs

1. π_k: Topic distribution over words - help us interpret each topic semantically

Topic	broccoli	banana	breakfast	kitten	cute	hamster	like	yesterday	*Total*
A	.30	.25	.20	.01	.01	.01	.12	.10	1
B	.01	01	01	.35	.24	.25	.08	.05	1

Latent Dirichlet Allocation (LDA)

Outputs

1. π_k: Topic distribution over words - help us interpret each topic semantically

Topic	broccoli	banana	breakfast	kitten	cute	hamster	like	yesterday	*Total*
A	.30	.25	.20	.01	.01	.01	.12	.10	1
B	.01	01	01	.35	.24	.25	.08	.05	1

2. θ_i: Document distribution over topics - help us identify representative documents in each topic

Document	Topic A Weight	Topic B Weight	Total
1	.99	.01	1
2	.99	.01	1
3	.01	99	1
4	.01	99	1
4	.60	.40	1

Latent Dirichlet Allocation (LDA)

Decisions

Latent Dirichlet Allocation (LDA)

Decisions

1) How should we preprocess the data?
2) How should we choose k?
3) Random starting values!

Latent Dirichlet Allocation (LDA)

Decisions

1) How should we preprocess the data?
2) How should we choose k?
3) Random starting values!

Validation: run topic models with different k or random starting values multiple times, choose the model that yields the most substantially / semantically interpretable topics.

Latent Dirichlet Allocation (LDA)

Decisions

1) How should we preprocess the data?
2) How should we choose k?
3) Random starting values!

Validation: run topic models with different k or random starting values multiple times, choose the model that yields the most substantially / semantically interpretable topics.

1. Look at top / distinctive words for each topic.
2. Read the most representative documents for each topic.

Structural Topic Model (STM)

STM is an extension of LDA

Structural Topic Model (STM)

STM is an extension of LDA

In addition to representing each document as a mixture of topics, we may also want to know in what kind of documents (covariates / metadata) is a certain topic X most prevalent?

Structural Topic Model (STM)

STM is an extension of LDA

In addition to representing each document as a mixture of topics, we may also want to know in what kind of documents (covariates / metadata) is a certain topic X most prevalent?

Example 1: In House floor debates, do Republicans or Democrats conduct more credit claiming in their speeches?

- Topic: credit claiming
- Covariate of document: Republican / Democrats

Structural Topic Model (STM)

STM is an extension of LDA

In addition to representing each document as a mixture of topics, we may also want to know in what kind of documents (covariates / metadata) is a certain topic X most prevalent?

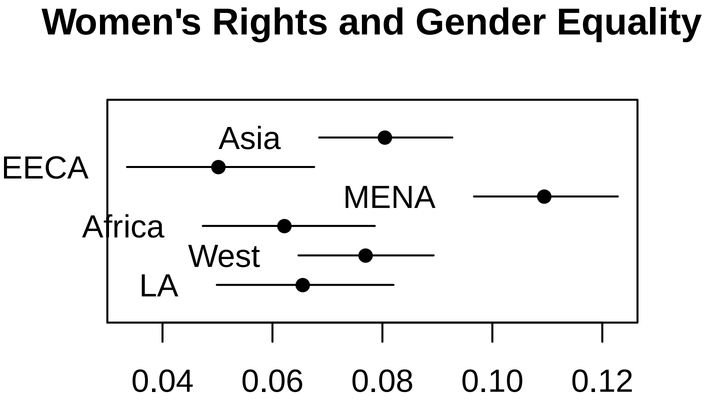
Example 1: In House floor debates, do Republicans or Democrats conduct more credit claiming in their speeches?

- Topic: credit claiming
- Covariate of document: Republican / Democrats

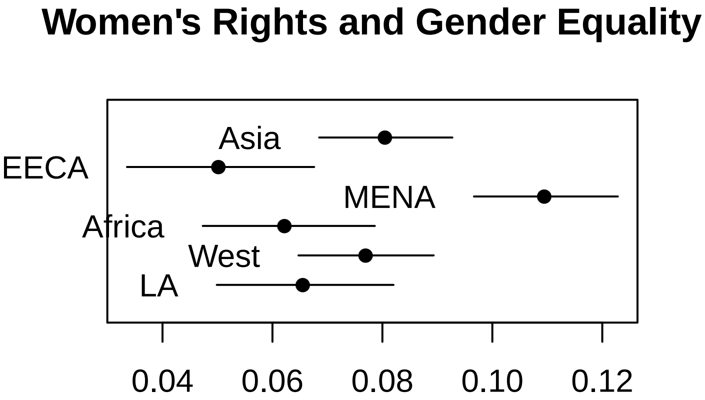
Example 2: In NYT articles, do articles on the Middle East talk more about women's rights and gender equality than articles on other regions do?

- Topic: women's rights and gender equality
- Covariate of document: region (Middle East, West, Asia, etc.)

Structural Topic Model



Structural Topic Model



R code, Section 2