CSE 494/598
Lecture-11: Clustering & Classification

LYDIA MANIKONDA
HTTP://WWW.PUBLIC.ASU.EDU/~LMANIKON/

**With permission, content adapted from last year’s slides and from Intro to DM dmbook@cs.umn.edu**
Announcements

• Project part – 3 is released

• Final exam: **May 2\textsuperscript{nd} 2016, 12:10 – 2:00 PM BYAC 240** (same as our regular class room)

• In-class homework/exercise – April 22\textsuperscript{nd} 2016
  • Closed book
  • 2 sets of questions
Today’s lecture

• Clustering
  • Types of clustering
  • K-means clustering algorithm
    • Evaluation
    • Limitations

• Classification
Clustering
What is Clustering?

Clustering is the process of grouping a set of physical or abstract objects into classes of similar objects.

- Objects in a class will be
  - Similar (or related) to one another
  - Different from (or unrelated to) the objects in other groups

- It is also called unsupervised learning.
- It is a common and important task that finds many applications.
Applications of Cluster Analysis

Applications in Search engines:
- Understanding
  - Group related documents that are similar
- Summarization
  - Reduces the size of large data sets
- Structuring search results
- Suggesting related pages
- Automatic directory construction/update
- Finding near identical/duplicate pages

<table>
<thead>
<tr>
<th>Discovered Clusters (content-based)</th>
<th>Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Bank ; River bank</td>
<td>Geography</td>
</tr>
<tr>
<td>2 The Bank of the River</td>
<td>Fiction</td>
</tr>
<tr>
<td>3 The Left Bank at River Oak Rentals</td>
<td>Apartments</td>
</tr>
</tbody>
</table>
Text Clustering in Search

Clustering can be done at:
- Indexing time
- At query time
  - Applied to documents
  - Applied to snippets

Clustering can be based on:
- URL source
  - Put pages from the same server together
- Text Content
  - Polysemy ("bat", "banks")
  - Multiple aspects of a single topic
- Links
  - Look at the connected components in the link graph (A/H analysis can do it)
  - Look at co-citation similarity (e.g. as in collaborative filtering)
What is not Cluster Analysis?

• Supervised classification
  • Have class label information

• Simple segmentation
  • Dividing students into different registration groups alphabetically, by last name

• Graph partitioning
  • Some mutual relevance and synergy, but areas are not identical
Notion of a Cluster can be Ambiguous

How many clusters?

Six Clusters

Two Clusters

Four Clusters
Types of Clustering

• A clustering is a set of clusters

• Important distinction between hierarchical and partitional sets of clusters

• Partitional Clustering
  • A division of data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset

• Hierarchical Clustering
  • A set of nested clusters organizes as a hierarchical tree
Partitional Clustering

Original Points

A Partitional Clustering
Hierarchical Clustering

Traditional Hierarchical Clustering

Non-traditional Hierarchical Clustering

Traditional Dendrogram

Non-traditional Dendrogram
Clustering issues

--Hard vs. Soft clusters

--Distance measures
cosine or Jaccard or.

--Cluster quality:
Internal measures
--intra-cluster tightness
--inter-cluster separation

External measures
--How many points are put in wrong clusters.
Cluster Evaluation

- Clusters can be evaluated with “internal” as well as “external” measures
  - Internal measures are related to the inter/intra cluster distance
    - A good clustering is one where
      - *(Intra-cluster distance)* the sum of distances between objects in the same cluster are minimized,
      - *(Inter-cluster distance)* while the distances between different clusters are maximized
    - Objective to minimize: F(Intra, Inter)
  - External measures are related to how representative are the current clusters to “true” classes. Measured in terms of purity, entropy or F-measure
    - Note that in real world, you often *don’t know* what the true classes are. (This is why clustering is called unsupervised learning)
Inter and Intra Cluster Distances

**Intra-cluster distance/tightness**
(Sum/Min/Max/Avg) the (absolute/squared) distance between
- All pairs of points in the cluster OR
- "diameter"—two farthest points
- Between the centroid/medoid and all points in the cluster

**Inter-cluster distance**
Sum the (squared) distance between all pairs of clusters
Where distance between two clusters is defined as:
- distance between their centroids/medoids
- Distance between farthest pair of points (complete link)
- Distance between the closest pair of points belonging to the clusters (single link)
Cluster Evaluations

- Clusters can be evaluated with “internal” as well as “external” measures
  - Internal measures are related to the inter/intra cluster distance
    - A good clustering is one where
      - (Intra-cluster distance) the sum of distances between objects in the same cluster are minimized,
      - (Inter-cluster distance) while the distances between different clusters are maximized
    - Objective to minimize: $F(\text{Intra}, \text{Inter})$
  - External measures are related to how representative are the current clusters to “true” classes. Measured in terms of purity, entropy or F-measure
Cluster Purity (Given gold standard classes)

Cluster I
Cluster II
Cluster III

Pure size of a cluster = # elements from the majority class

Sum of pure sizes of clusters

Total number of elements across clusters

Will it work if you allow # of clusters to increase?

Purity of clustering:

\[
\frac{\text{Sum of pure sizes of clusters}}{\text{Total number of elements across clusters}} = \frac{5 + 4 + 3}{6 + 6 + 5} = \frac{12}{17} = 0.71
\]
**Rand Index Example**

The following table classifies all pairs of entities (of which there are \( n \) choose 2) into one of four classes:

<table>
<thead>
<tr>
<th>Number of points</th>
<th>Same Cluster in clustering</th>
<th>Different Clusters in clustering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same class in ground truth</td>
<td>TP</td>
<td>FN</td>
</tr>
<tr>
<td>Different classes in ground truth</td>
<td>FP</td>
<td>TN</td>
</tr>
</tbody>
</table>

**Is the cluster putting non-class items in?**

\[
P = \frac{TP}{TP + FP}
\]

**Is the cluster missing any in-class items?**

\[
R = \frac{TP}{TP + FN}
\]

**Compare to Standard Precision & Recall**

\[
RI = \frac{TP + TN}{TP + TN + FP + FN}
\]
Rand Index Example

Cluster I

Cluster II

Cluster III

Elementary combinatorics
TP+FP (total pairs in the same clusters)
= 6 \binom{2}{} + 6 \binom{2}{} + 5 \binom{2}{} = 40
To get TP
= 5 \binom{2}{} + 4 \binom{2}{} + 3 \binom{2}{} + 2 \binom{2}{} = 20
You can compute FN/TN similarly

\[
RI = \frac{20 + 72}{20 + 20 + 24 + 72} = 0.68
\]
Unsupervised?

Clustering is normally seen as an instance of unsupervised learning algorithm

- So how can you have external measures of cluster validity?
- The truth is that you have a continuum between unsupervised vs. supervised
  - Answer: Think of “no teacher being there” vs. “lazy teacher” who checks your work once in a while.
  - Examples:
    - Fully unsupervised (no teacher)
    - Teacher tells you how many clusters are there
    - Teacher tells you that certain pairs of points will fall or will not fill in the same cluster
    - Teacher may occasionally evaluate the goodness of your clusters (external measures of validity)
How hard is clustering?

One idea is to consider all possible clusterings, and pick the one that has best inter and intra cluster distance properties.

Suppose we are given $n$ points, and would like to cluster them into $k$-clusters:

- How many possible clusterings?

\[
\sum_{k=1}^{n} \frac{k^n}{k!}
\]

- Too hard to do it brute force or optimally
- Solution: Iterative optimization algorithms
  - Start with a clustering, iteratively improve it (eg. K-means)
Types of Clusters

- Well-separated clusters
- Center-based clusters
- Contiguous clusters
- Density-based clusters
- Property or Conceptual
- Described by an Objective Function
Types of Clusters: Well-Separated

- Well-separated clusters:
  - A cluster is a set of points such that any point in a cluster is closer (or more similar) to every other point in the cluster than to any point not in the cluster

3 well-separated clusters
Types of Clusters: Center-Based

- Center-based
  - A cluster is a set of objects such that an object in a cluster is close (more similar) to the “center” of a cluster, than to the center of any other cluster.
  - The center of a cluster is often a centroid, the average of all the points in the cluster, or a medoid, the most “representative” point of a cluster.

4 center-based clusters
Types of Clusters: Contiguity-Based

• Contiguous Cluster (Nearest Neighbor or Transitive)
  • A cluster is a set of points such that a point in a cluster is close (or more similar) to one or more other points in the cluster than to any point not in the cluster

8 contiguous clusters
Types of Clusters: Density-Based

- Density-based
  - A cluster is a dense region of points, which is separated by low-density regions, from other regions of high density
  - Used when the clusters are irregular or inter-twined, and when noise and outliers are present
Types of Clusters: Conceptual Clusters

- Shared Property or Conceptual Clusters
  - Finds clusters that share some common property or represent a particular concept

2 Overlapping Circles
Types of Clusters: Objective Function

- Clusters defined by an objective function
  - Find clusters that minimize or maximize an objective function
  - Enumerate all possible ways of dividing the points into clusters and evaluate the ‘goodness’ of each potential set of clusters by using the given objective function (NP Hard)
  - Can have global or local objectives
  - A variation of the global objective function approach is to fit the data to a parameterized model
    - Parameters for the model are determined from the data
    - Mixture models assume that the data is a ‘mixture’ of a number of statistical distributions
- Map the clustering problem to a different domain and solve a related problem in that domain
  - Proximity matrix defines a weighted graph, where the nodes are the points being clustered, and the weighted edges represent the proximities between points
  - Clustering is equivalent to breaking the graph into connected components, one for each cluster
  - Want to minimize the edge weight between clusters and maximize the edge weight within clusters
Characteristics of the Input Data are Important

• Type of proximity or density measure
  • This is a derived measure, but central to clustering

• Sparseness
  • Dictates type of similarity
  • Adds to efficiency

• Attribute type
  • Dictates type of similarity

• Type of data
  • Dictates type of similarity
  • Other characteristics e.g., autocorrelation

• Dimensionality

• Noise and outliers

• Type of distribution
Classical clustering methods

Partitioning methods
  ◦ k-Means (and EM), k-Medoids

Hierarchical methods
  ◦ agglomerative, divisive, BIRCH

Model-based clustering methods
K-means Clustering

• Partitional Clustering approach
• Each cluster is associated with a centroid (center point)
• Each point is assigned to the cluster with the closest centroid
• Number of clusters, K, must be specified
• The basic algorithm is very simple
K-Means Clustering Algorithm

Works when we know k, the number of clusters we want to find

Algorithm:

1: Select $K$ points as the initial centroids.
2: repeat
3: Form $K$ clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: until The centroids don’t change

Iterative improvement of the objective function:

Sum of the squared distance (or Error -- SSE) from each point to the centroid of its cluster

(Notice that since $K$ is fixed, maximizing tightness also maximizes inter-cluster distance)
K-means Example (k=2)

Pick seeds
Reassign clusters
Compute centroids
Reassign clusters
Compute centroids
.....
Converged!
K-means Clustering Algorithm

• Initial centroids are often chosen randomly
• The centroid is typically the mean of the points in the cluster
• ‘Closeness’ is measured by Euclidean distance, cosine similarity, correlation, etc.
• K-means will converge for common similarity measures mentioned above
• Most of the convergence happens in the first few iterations
  • Often the stopping condition is changed to ‘Until relatively few points change clusters’
• Complexity is $O(n \cdot K \cdot I \cdot d)$
  • $n =$ number of data points
  • $K =$ number of clusters
  • $I =$ number of iterations
  • $d =$ number of attributes
Two different K-means Clusterings

Original Points

Optimal Clustering

Sub-optimal Clustering
Evaluating K-means Clusters

Most common measure is Sum of Squared Error (SSE)

- For each point, the error is the distance to the nearest cluster
- To get SSE, we square these errors and sum them.

\[ SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x) \]

- \( x \) is a data point in cluster \( C_i \) and \( m_i \) is the representative point for cluster \( C_i \)
- can show that \( m_i \) corresponds to the center (mean) of the cluster
- Given two clusters, we can choose the one with the smallest error
- One easy way to reduce SSE is to increase \( K \), the number of clusters
- A good clustering with smaller \( K \) can have a lower SSE than a poor clustering with higher \( K \)
Handling Empty Clusters

• Basic K-means algorithm can yield empty clusters

• Several strategies:
  ◦ Choose the point that contributes most to SSE
  ◦ Choose a point from the cluster with the highest SSE
  ◦ If there are several empty clusters, the above can be repeated several times.
Updating Centers Incrementally

• In the basic K-means algorithm, centroids are updated after all points are assigned to a centroid

• An alternative is to update the centroids after each assignment (incremental approach)
  ◦ Each assignment updates zero or two centroids
  ◦ More expensive
  ◦ Introduces an order dependency
  ◦ Never get an empty cluster
  ◦ Can use “weights” to change the impact
Limitations of K-means

K-means has problems when clusters are of differing
  ◦ Sizes
  ◦ Densities
  ◦ Non-globular shapes

K-means has problems when the data contains outliers.
Limitations of K-means: Differing Sizes

Original Points

K-means (3 Clusters)
Limitations of K-means: Differing Density

Original Points

K-means (3 Clusters)
Limitations of K-means: Non-globular Shapes

Original Points

K-means (2 Clusters)
Overcoming K-means Limitations

One solution is to use many clusters

Find parts of clusters, but need to put together
Overcoming K-means Limitations

Original Points

K-means Clusters
Overcoming K-means Limitations

Original Points

K-means Clusters
Classification
Classification

• Given a collection of records (training set)
  • Each record contains a set of attributes, one of the attributes is the class.

• Find a model for class attribute as a function of the values of other attributes.

• Goal: previously unseen records should be assigned a class as accurately as possible.
  • A test set is used to determine the accuracy of the model. Usually, the given data set is divided into training and test sets, with training set used to build the model and test set used to validate it.
Classification Task

<table>
<thead>
<tr>
<th>Tid</th>
<th>Attrib1</th>
<th>Attrib2</th>
<th>Attrib3</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Yes</td>
<td>Large</td>
<td>125K</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>No</td>
<td>Medium</td>
<td>100K</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>No</td>
<td>Small</td>
<td>70K</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>Yes</td>
<td>Medium</td>
<td>120K</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>No</td>
<td>Large</td>
<td>95K</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>No</td>
<td>Medium</td>
<td>60K</td>
<td>No</td>
</tr>
<tr>
<td>7</td>
<td>Yes</td>
<td>Large</td>
<td>220K</td>
<td>No</td>
</tr>
<tr>
<td>8</td>
<td>No</td>
<td>Small</td>
<td>85K</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>No</td>
<td>Medium</td>
<td>75K</td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>No</td>
<td>Small</td>
<td>90K</td>
<td>No</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tid</th>
<th>Attrib1</th>
<th>Attrib2</th>
<th>Attrib3</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>No</td>
<td>Small</td>
<td>55K</td>
<td>?</td>
</tr>
<tr>
<td>12</td>
<td>Yes</td>
<td>Medium</td>
<td>80K</td>
<td>?</td>
</tr>
<tr>
<td>13</td>
<td>Yes</td>
<td>Large</td>
<td>110K</td>
<td>?</td>
</tr>
<tr>
<td>14</td>
<td>No</td>
<td>Small</td>
<td>95K</td>
<td>?</td>
</tr>
<tr>
<td>15</td>
<td>No</td>
<td>Large</td>
<td>67K</td>
<td>?</td>
</tr>
</tbody>
</table>

Induction

Deduction

Learning algorithm

Model

Apply Model

Learn Model

Test Set

Training Set
Examples of Classification Task

• Predicting tumor cells as benign or malignant
• Classifying credit card transactions as legitimate or fraudulent
• Classifying secondary structures of protein as alpha-helix, beta-sheet, or random coil
• Categorizing news stories as finance, weather, entertainment, sports, etc
Classification Techniques

- Decision Tree based Methods
- Rule-based Methods
- Memory based reasoning
- Neural Networks
- Naïve Bayes and Bayesian Belief Networks
- Support Vector Machines
Classification vs Clustering

- Coming from Clustering, classification seems significantly simple...
- You are already given the clusters and names (over the training data)
- All you need to do is to decide, for the test data, which cluster it should belong to.
- Seems like a simple distance computation
  - Assign test data to the cluster whose members seem to make the majority of its neighbors
K-Nearest Neighbor for Text

Training:
For each training example \(<x, c(x)> \in D\)

Compute the corresponding TF-IDF vector, \(d_x\), for document \(x\)

Test instance \(y\):
Compute TF-IDF vector \(d\) for document \(y\)
For each \(<x, c(x)> \in D\)

Let \(s_x = \text{cosSim}(d, d_x)\)

Sort examples, \(x\), in \(D\) by decreasing value of \(s_x\)
Let \(N\) be the first \(k\) examples in \(D\).  \((\text{get most similar neighbors})\)
Return the majority class of examples in \(N\)

Find \(k\) nearest neighbors is just retrieving \(k\) closest docs!
K-Nearest Neighbor Algorithm

- The test data point (blue circle) should be classified either to the first class of red cross marks or to the second class of green stars.
- For different values of $k$ its take different class labels.
- If $k = 3$ (solid line circle) it is assigned to the second class because there are 2 stars and only 1 cross mark inside the inner circle.
- If $k = 5$ (dashed line circle) it is assigned to the first class (3 cross marks vs. 2 stars inside the outer circle).
Why clustering-based methods are not enough?

- The class labels may be describing “non-spherical” (non-isotropic) clusters
  - The coastal cities class of USA

- The class labels may be effectively combining non-overlapping clusters into a single class
  - Hawaii & Alaska are in USA class?

- These problems exist in clustering too—but since it is posed as an unsupervised problem, any clustering with good internal quality is seen as okay.
  - In the case of classification, we are forced to find the clustering that actually agrees with class labels
    - (if the teacher doesn’t know the answer for a test question, all they can see is whether you wrote cleanly and argued persuasively... similarly, in the case of clustering, since no external validation exists in general, you can only hope to see how “tight” the clusters are)
A Classification Example – Predicting when Russell will wait for a table

---

## Attributes

<table>
<thead>
<tr>
<th>Example</th>
<th>Alt</th>
<th>Bar</th>
<th>Fri</th>
<th>Hun</th>
<th>Pat</th>
<th>Price</th>
<th>Rain</th>
<th>Res</th>
<th>Type</th>
<th>Est</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Some</td>
<td>$$ $$</td>
<td>No</td>
<td>Yes</td>
<td>French</td>
<td>0–10</td>
<td>Yes</td>
</tr>
<tr>
<td>X2</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Full</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>Thai</td>
<td>30–60</td>
<td>No</td>
</tr>
<tr>
<td>X3</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Some</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>Burger</td>
<td>0–10</td>
<td>Yes</td>
</tr>
<tr>
<td>X4</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Full</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>Thai</td>
<td>10–30</td>
<td>Yes</td>
</tr>
<tr>
<td>X5</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Full</td>
<td>$$ $$</td>
<td>No</td>
<td>Yes</td>
<td>French</td>
<td>&gt;60</td>
<td>No</td>
</tr>
<tr>
<td>X6</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Some</td>
<td>$$</td>
<td>Yes</td>
<td>Yes</td>
<td>Italian</td>
<td>0–10</td>
<td>Yes</td>
</tr>
<tr>
<td>X7</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>None</td>
<td>$</td>
<td>Yes</td>
<td>No</td>
<td>Burger</td>
<td>0–10</td>
<td>No</td>
</tr>
<tr>
<td>X8</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Some</td>
<td>$$</td>
<td>Yes</td>
<td>Yes</td>
<td>Thai</td>
<td>0–10</td>
<td>Yes</td>
</tr>
<tr>
<td>X9</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Full</td>
<td>$</td>
<td>Yes</td>
<td>No</td>
<td>Burger</td>
<td>&gt;60</td>
<td>No</td>
</tr>
<tr>
<td>X10</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Full</td>
<td>$$ $$</td>
<td>No</td>
<td>Yes</td>
<td>Italian</td>
<td>10–30</td>
<td>No</td>
</tr>
<tr>
<td>X11</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>None</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>Thai</td>
<td>0–10</td>
<td>No</td>
</tr>
<tr>
<td>X12</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Full</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>Burger</td>
<td>30–60</td>
<td>Yes</td>
</tr>
</tbody>
</table>

---

similar to book preferences, predicting credit card fraud, predicting when people are likely to respond to junk mail
Use different biases to predict Russell’s waiting habits

If patrons=full and day=Friday then wait (0.3/0.7)
If wait>60 and Reservation=no then wait (0.4/0.9)

Association rules
--Examples are used to
--Learn support and confidence of association rules

Naïve bayes (bayesnet learning)
--Examples are used to
--Learn topology
--Learn CPTs

K-nearest neighbors

Decision Trees
--Examples are used to
--Learn topology
--Order of questions

SVMs

Neural Nets
--Examples are used to
--Learn topology
--Learn edge weights
Parametric vs Non-Parametric Learners

• K-NN is an example of non-parametric method.
  ◦ The size of the “learned structure” is proportional to the size of training set

• More traditional learners are “parametric”
  ◦ They summarize the training set with a fixed set of parameters
    ◦ E.g. Linear classifiers
K-NN and Higher Dimensions

- Nearest neighbors in high-dimensions are not very near
  - Consider an n-dimensional apple—it is all peel and no core.
  - So, if you consider an n-D sphere centered on a data point, all its neighbors are going to be at the shell!
  - K-NN doesn’t do well in high-dimensions

![Figure 10.27](image)
Today

• Clustering
  • Types of Clusterings
  • Types of clusters
  • K-means
    • Evaluation
    • Limitations

• Classification
  • K-nearest Neighbor