Clustering Big Data

Anil K. Jain
(with Radha Chitta and Rong Jin)

Department of Computer Science
Michigan State University
November 29, 2012
Outline

• Big Data
• How to extract “information”?
• Data clustering
• Clustering Big Data
• Kernel K-means & approximation
• Summary
How Big is Big Data?

- **Big** is a fast moving target: kilobytes, megabytes, gigabytes, terabytes ($10^{12}$), petabytes ($10^{15}$), exabytes ($10^{18}$), zettabytes ($10^{21}$),......

- Over 1.8 zb created in 2011; ~8 zb by 2015

Source: IDC’s Digital Universe study, sponsored by EMC, June 2011
http://idcdocserv.com/1142

Nature of Big Data: Volume, Velocity and Variety
Big Data on the Web

Over 225 million users generating over 800 tweets per second

~900 million users, 2.5 billion content items, 105 terabytes of data each half hour, 300M photos and 4M videos posted per day

Big Data on the Web

- Over 50 billion pages indexed and more than 2 million queries/min
- Articles from over 10,000 sources in real time
- ~4.5 million photos uploaded/day
- 48 hours of video uploaded/min; more than 1 trillion video views
- No. of mobile phones will exceed the world’s population by the end of 2012
What to do with Big Data?

• Extract information to make decisions

• Evidence-based decision: data-driven vs. analysis based on intuition & experience

• Analytics, business intelligence, data mining, machine learning, pattern recognition

• Big Data computing: IBM is promoting Watson (Jeopardy champion) to tackle Big Data in healthcare, finance, drug design,..

Steve Lohr, “Amid the Flood, A Catchphrase is Born”, NY Times, August 12, 2012
Decision Making

• Data Representation
  • Features and similarity

• Learning
  • Classification (labeled data)
  • Clustering (unlabeled data)

Most big data problems have unlabeled objects
Pattern Matrix

$n \times d$ pattern matrix
Similarity Matrix

Polynomial kernel: $K(x, y) = \left(x^T y + 1\right)^4$
Classification

Given a training set of labeled objects, learn a decision rule
Clustering

Given a collection of (unlabeled) objects, find meaningful groups
Semi-supervised Clustering

Supervised

Unsupervised

Dogs

Cats

Semi-supervised

Pairwise constraints improve the clustering performance
What is a cluster?

“A group of the same or similar elements gathered or occurring closely together”
Clusters in 2D
Challenges in Data Clustering

- Measure of similarity
- No. of clusters
- Cluster validity
- Outliers
Data Clustering

Organize a collection of $n$ objects into a partition or a hierarchy (nested set of partitions)

“Data clustering” returned ~6,100 hits for 2011 (Google Scholar)
Clustering is the Key to Big Data Problem

• Not feasible to “label” large collection of objects
• No prior knowledge of the number and nature of groups (clusters) in data
• Clusters may evolve over time
• Clustering provides efficient browsing, search, recommendation and organization of data
Clustering Users on Facebook

- ~300,000 status updates per minute on tens of thousands of topics
- Cluster users based on topic of status messages

http://searchengineland.com/by-the-numbers-twitter-vs-facebook-vs-google-buzz-36709
Clustering Articles on Google News

Clustering Videos on Youtube

- Keywords
- Popularity
- Viewer engagement
- User browsing history

http://www.strutta.com/blog/blog/six-degrees-of-youtube
Clustering for Efficient Image retrieval

Fig. 1. Upper-left image is the query. Numbers under the images on left side: image ID and cluster ID; on the right side: Image ID, matching score, number of regions.

Retrieval accuracy for the “food” category (average precision):

Without clustering: 47%

With clustering: 61%

Clustering Algorithms

Hundreds of clustering algorithms are available; many are “admissible”, but no algorithm is “optimal”

- K-means
- Gaussian mixture models
- Kernel K-means
- Spectral Clustering
- Nearest neighbor
- Latent Dirichlet Allocation

K-means Algorithm

1. Randomly assign cluster labels to the data points.
2. Compute the center of each cluster.
3. Assign points to the nearest cluster center.
4. Re-compute centers.
5. Repeat until there is no change in the cluster labels.
K-means: Limitations

Prefers “compact” and “isolated” clusters

\[
\min \sum_{i=1}^{n} \sum_{k=1}^{K} u_{ik} \| x_i - c_k \|^2
\]
Gaussian Mixture Model

Kernel K-means

Non-linear mapping to find clusters of arbitrary shapes

\[
\min Trace \left( \sum_{i=1}^{n} \sum_{k=1}^{K} u_{ik} (\phi(x_i) - c_k \phi)(\phi(x_i) - c_k \phi)^T \right)
\]

\[
\phi(x, y) = (x^2, \sqrt{2}xy, y^2)^T \quad K(a, b) = \phi(a)^T \phi(b)
\]

Polynomial kernel representation
Spectral Clustering

Represent data using the top K eigenvectors of the kernel matrix; equivalent to Kernel K-means
K-means vs. Kernel K-means

Kernel clustering is able to find “complex” clusters
How to choose the right kernel? RBF kernel is the default
Kernel K-means is Expensive

<table>
<thead>
<tr>
<th>No. of Objects (n)</th>
<th>No. of operations</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>K-means</td>
<td>Kernel K-means</td>
</tr>
<tr>
<td></td>
<td>O(nKd)</td>
<td>O(n^2K)</td>
</tr>
<tr>
<td>1M</td>
<td>10^{13} (6412*)</td>
<td>10^{16}</td>
</tr>
<tr>
<td>10M</td>
<td>10^{14}</td>
<td>10^{18}</td>
</tr>
<tr>
<td>100M</td>
<td>10^{15}</td>
<td>10^{20}</td>
</tr>
<tr>
<td>1B</td>
<td>10^{16}</td>
<td>10^{22}</td>
</tr>
</tbody>
</table>

d = 10,000; K=10

* Runtime in seconds on Intel Xeon 2.8 GHz processor using 40 GB memory

A petascale supercomputer (IBM Sequoia, June 2012) with ~1 exabyte memory is needed to run kernel K-means on 1 billion points!
Clustering Big Data

- **Data**
- **n x n similarity matrix**
- **Pre-processing**
- **Clustering**
- **Cluster labels**

Additional terms:
- Sampling
- Summarization
- Incremental
- Distributed
- Approximation
Distributed Clustering

Clustering 100,000 2-D points with 2 clusters on 2.3 GHz quad-core Intel Xeon processors, with 8GB memory in intel07 cluster

Network communication cost increases with the no. of processors
Approximate kernel K-means

Tradeoff between clustering accuracy and running time

Randomly sample $n$ points in the $d$-dimensional space, compute the kernel similarity matrices $K_A (m \times m)$ and $K_B (n \times m)$.

Linear runtime and memory complexity

\[
\min \max_{(x_j)_{j=1}^m} \sum_{k=1}^m \sum_{i=1}^n u_{i,k} K_A(x_i, x_j), \quad \sum_{j=1}^m \sum_{i=1}^n \alpha_{j,k} K_B(y_i, y_j)
\]

(equivalent to running K-means on $K_B K_A^{-1} K_B^T$)

Chitta, Jin, Havens & Jain, Approximate Kernel k-means: solution to Large Scale Kernel Clustering, KDD, 2011
Approximate Kernel K-Means

<table>
<thead>
<tr>
<th>No. of objects ((n))</th>
<th>Kernel K-means</th>
<th>Approximate kernel K-means ((m=100))</th>
<th>K-means</th>
<th>Running time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10K</td>
<td>3.09</td>
<td>0.20</td>
<td>0.03</td>
<td>0.20</td>
</tr>
<tr>
<td>100K</td>
<td>320.10</td>
<td>1.18</td>
<td>0.17</td>
<td>1.18</td>
</tr>
<tr>
<td>1M</td>
<td>-</td>
<td>15.06</td>
<td>0.72</td>
<td>15.06</td>
</tr>
<tr>
<td>10M</td>
<td>-</td>
<td>234.49</td>
<td>12.14</td>
<td>234.49</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Clustering accuracy (%)</th>
<th>Kernel K-means</th>
<th>Approximate kernel K-means ((m=100))</th>
<th>K-means</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>100</td>
<td>93.8</td>
<td>50.1</td>
</tr>
<tr>
<td>100K</td>
<td>100</td>
<td>93.7</td>
<td>49.9</td>
</tr>
<tr>
<td>1M</td>
<td>-</td>
<td>95.1</td>
<td>50.0</td>
</tr>
<tr>
<td>10M</td>
<td>-</td>
<td>91.6</td>
<td>50.0</td>
</tr>
</tbody>
</table>

2.8 GHz processor, 40 GB
Tiny Image Data set

~80 million 32x32 images from ~75K classes (bamboo, fish, mushroom, leaf, mountain,…); image represented by 384-dim. GIST descriptors

Fergus et al., 80 million tiny images: a large dataset for non-parametric object and scene recognition, PAMI 2008
Tiny Image Data set

10-class subset (CIFAR-10): 60K manually annotated images

Airplane
Automobile
Bird
Cat
Deer
Dog
Frog
Horse
Ship
Truck

Krizhevsky, Learning multiple layers of features from tiny images, 2009
Clustering Tiny Images

Example Clusters

<table>
<thead>
<tr>
<th>C_1</th>
<th>C_2</th>
<th>C_3</th>
<th>C_4</th>
<th>C_5</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Average clustering time (100 clusters)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approximate kernel K-means (m=1,000)</td>
</tr>
<tr>
<td>K-means</td>
</tr>
</tbody>
</table>

2.3GHz, 150GB memory
Clustering Tiny Images

Best Supervised Classification Accuracy on CIFAR-10: 54.7%

<table>
<thead>
<tr>
<th>Clustering accuracy</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Kernel K-means</td>
<td>29.94%</td>
</tr>
<tr>
<td>Approximate kernel K-means (m = 5,000)</td>
<td>29.76%</td>
</tr>
<tr>
<td>Spectral clustering</td>
<td>27.09%</td>
</tr>
<tr>
<td>K-means</td>
<td>26.70%</td>
</tr>
</tbody>
</table>

Ranzato et. Al., Modeling pixel means and covariances using factorized third-order boltzmann machines, CVPR 2010
Fowlkes et al., Spectral grouping using the Nystrom method, PAMI 2004
Distributed Approx. Kernel K-means

For better scalability and faster clustering

\[ \text{Given} \ n \ \text{points in} \ d\text{-dimensional space} \]

Randomly sample \( m \) points (\( m \ll n \))

Split the remaining \( n - m \) randomly into \( p \) partitions and assign partition \( P_t \) to task \( t \)

Run approximate kernel K-means in each task \( t \) and find the cluster centers

Assign each point in task \( s \) (\( s \neq t \)) to the closest center from task \( t \)

Combine the labels from each task using ensemble clustering algorithm
Distributed Approximate kernel K-means

2-D data set with 2 concentric circles

<table>
<thead>
<tr>
<th>Size of data set</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>10K</td>
<td>3.8</td>
</tr>
<tr>
<td>100K</td>
<td>4.8</td>
</tr>
<tr>
<td>1M</td>
<td>3.8</td>
</tr>
<tr>
<td>10M</td>
<td>6.4</td>
</tr>
</tbody>
</table>

2.3 GHz quad-core Intel Xeon processors, with 8GB memory in the intel07 cluster
Limitations of Approx. kernel K-means

Clustering data with more than 10 million points will require terabytes of memory!

Sample and Cluster Algorithm (SnC)

1. Sample s points from data
2. Run approximate kernel K-means on the s points
3. Assign remaining points to the nearest cluster center
Clustering one billion points

Sample and Cluster (s = 1 million, m = 100)

<table>
<thead>
<tr>
<th>Running time</th>
<th>Average Clustering Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-means</td>
<td>K-means</td>
</tr>
<tr>
<td>53 minutes</td>
<td>50%</td>
</tr>
<tr>
<td>SnC</td>
<td>SnC –distributed (8 cores)</td>
</tr>
<tr>
<td>1.2 hours</td>
<td>45 minutes</td>
</tr>
<tr>
<td>SnC</td>
<td>85%</td>
</tr>
</tbody>
</table>
Clustering billions of points

• Work in progress
  – Application to real data sets
  – Yahoo! AltaVista Web Page Hyperlink Connectivity Graph (2002) containing URLs and hyperlinks for over 1.4 billion public web pages

• Challenges
  – Graph Sparsity: Reduce the dimensionality using random projection, PCA
  – Cluster Evaluation: No ground truth available, internal measures such as link density of clusters
Summary

• Clustering is an exploratory technique; used in every scientific field that collects data
• Choice of clustering algorithm & its parameters is data dependent
• Clustering is essential for “Big Data” problem
• Approximate kernel K-means provides good tradeoff between scalability & clustering accuracy
• Challenges: Scalability, very large no. of clusters, heterogeneous data, streaming data, validity
CONSULTANTS SAY THREE QUINTILLION
BYTES OF DATA ARE CREATED EVERY DAY.

IT COMES FROM EVERYWHERE. IT KNOBS ALL.

ACCORDING TO THE BOOK OF WIKIPEDIA,
ITS NAME IS "BIG DATA."

BIG DATA LIVES IN THE CLOUD. IT KNOWS WHAT WE DO.

IN THE PAST, OUR COMPANY DID MANY EVIL THINGS.

BUT IF WE ACCEPT BIG DATA IN OUR SERVERS, WE WILL BE SAVED FROM BANKRUPTCY.

LET US PAY.

IS IT TOO LATE TO SIDE WITH EVIL?

SHHHH! IT HEARS YOU.

http://dilbert.com/strips/comic/2012-07-29/