Exploring Large Data Sets

Daniel L Boley
University of Minnesota

Find Reduced Order Representations for Large Unstructured Data Collections to facilitate finding patterns, connections, outliers, and to reduce noise.

NSF Grant 0916750
Exploring Large Data Sets

- Many large unstructured data sets must be analysed
  - Text documents (news, laws, WWW documents).
  - Gene expression profiles
  - Attributes for individual people, transactions, locations, ecosystems, . . .

- Gene-gene or protein-protein interaction networks
- WWW connectivity graph
- Computer inter-connect in Internet
- People-people affinities in Social Media

- Many example datasets can easily have up to $O(10^9+)$ data points.
- Many datasets have much noise or many attributes.
- Many example datasets are sampled, subject to sampling bias.
Tools to Explore

- **Dimensionality Reduction**
  - Represent each data sample with a reduced set of attribute values
  - Minimize loss of information
  - Implicit assumption: data is subject to some level of noise.

- **Graph Properties**
  - partitioning
  - identify important nodes or links
  - aggregate properties

- **Sparse Representation**
  - Hard to interpret individual components in traditional dimensionality reduction methods.
  - Seek to represent each data sample as a combination of only a few components.
  - Possibly also seek to represent each component as a combination of only a few original attributes.
  - Maintain desire for small approximation error.
Outline

• Dimensionality Reduction
  • Principal Component Analysis – PCA
  • Latent Semantic Indexing
  • Clustering

• Graph Partitioning
  • Principal Direction Divisive Partitioning
  • Spectral Partitioning

• Sparse Representation – Examples
  • almost shortest path routing.
  • constrained clustering.
  • image/vision,
  • Graph Connection Discovery.

• Finding Sparse Representation
Outline

- Dimensionality Reduction
  - Principal Component Analysis – PCA
  - Latent Semantic Indexing
  - Clustering

- Graph Partitioning
  - Principal Direction Divisive Partitioning
  - Spectral Partitioning

- Sparse Representation – Examples
  - almost shortest path routing.
  - constrained clustering.
  - image/vision,
  - Graph Connection Discovery.

- Finding Sparse Representation
Singular Value Decomposition – SVD

\[
\begin{align*}
(M - \mu_1^T \text{ or } M) \cdot U &= D \cdot V^T,
\end{align*}
\]

genes or words

assays or docs

eigenassays or concepts

assay/concept loadings (modes)
Singular Value Decomposition – SVD

- Eliminate Noise
- Reduce Dimensionality
- Expose Major Components

Suppose samples are columns of $m \times n$ matrix $M$.

Try to find $k$ pseudo-data columns such that all samples can be represented by linear combinations of those $k$ pseudo-data columns.

Primary criterion: minimize the 2-norm of the discrepancy between the original data and what you can represent using $k$ pseudo-data columns.

Answer: Singular Value Decomposition.

Sometimes, for statistical reasons, want to remove uniform signal:

- $M \leftarrow M - \mu 1^T$,
  where $\mu = M \cdot 1$.
- Then $M^T M$ is the Sample Covariance Matrix.
- Even without centering, $M^T M$ is a "Gram" matrix.
Principal Component Analysis – PCA

• Suppose samples are columns of $m \times n$ matrix $M$.
  
  ◦ Optionally center columns of matrix $M \leftarrow M - \mu 1^T$.
  ◦ Form sample covariance matrix or Gram matrix: $C = M^T M$, where $\mu = \frac{1}{n} M 1 = \text{sample mean}$, $1^T = [1, \ldots, 1]$.
  ◦ Diagonalize $C = V D^2 V^T$ to get principal components $V$, where $D^2 = \text{diag}(\sigma_1^2, \sigma_2^2, \cdots)$, $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$.

• Compute above via Singular Value Decomposition

$$M = U D V^T$$

• Top $k$ principal components $\implies$ best rank $k$ approximation:

$$U_{*,1\ldots k} \cdot D_{1\ldots k,1\ldots k} \cdot V_{*,1\ldots k}^T$$
Text Documents – Data Representation

• Each document represented by $n$-vector $d$ of word counts, scaled to unit length.
• Vectors assembled into Term Frequency Matrix $M = (d_1 \cdots d_m)$.
• Stay length-independent: compare using just angles.
Latent Semantic Indexing – LSI

- Loadings of top two concepts on set of 98 documents with 5623 words. (Berry et al., 1995; Boley, 1998)
Five Concepts

<table>
<thead>
<tr>
<th>PC 1</th>
<th></th>
<th>PC 2</th>
<th></th>
<th>PC 3</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>plus end</td>
<td></td>
<td>minus end</td>
<td></td>
<td>plus end</td>
<td></td>
</tr>
<tr>
<td>manufactur</td>
<td>manufactur</td>
<td>pipe</td>
<td>manufactur</td>
<td>edi</td>
<td></td>
</tr>
<tr>
<td>system</td>
<td>employ</td>
<td>seam</td>
<td>employ</td>
<td>behavior</td>
<td></td>
</tr>
<tr>
<td>develop</td>
<td>engin</td>
<td>convert</td>
<td>manufactur</td>
<td>chronolog</td>
<td></td>
</tr>
<tr>
<td>process</td>
<td>servic</td>
<td>processor</td>
<td>wherev</td>
<td></td>
<td></td>
</tr>
<tr>
<td>inform</td>
<td>employe</td>
<td>transmitt</td>
<td>busi</td>
<td>ink</td>
<td></td>
</tr>
<tr>
<td>applic</td>
<td>mean</td>
<td>waste</td>
<td>electron</td>
<td>incomplet</td>
<td></td>
</tr>
<tr>
<td>technologi</td>
<td>integr</td>
<td>chip</td>
<td>standard</td>
<td>height</td>
<td></td>
</tr>
<tr>
<td>integr</td>
<td>action</td>
<td>clock</td>
<td>action</td>
<td>slightli</td>
<td></td>
</tr>
<tr>
<td>standard</td>
<td>affirm</td>
<td>chicago</td>
<td>job</td>
<td>pump</td>
<td></td>
</tr>
<tr>
<td>engin</td>
<td>system</td>
<td>scheme</td>
<td>compani</td>
<td>label</td>
<td></td>
</tr>
<tr>
<td>program</td>
<td>job</td>
<td>highli</td>
<td>engin</td>
<td>clerk</td>
<td></td>
</tr>
<tr>
<td>employ</td>
<td>technologi</td>
<td>phd</td>
<td>mean</td>
<td>french</td>
<td></td>
</tr>
<tr>
<td>edi</td>
<td>process</td>
<td>robin</td>
<td>affirm</td>
<td>embassi</td>
<td></td>
</tr>
<tr>
<td>design</td>
<td>public</td>
<td>reprogramm</td>
<td>capit</td>
<td>mainli</td>
<td></td>
</tr>
<tr>
<td>servic</td>
<td>law</td>
<td>serc</td>
<td>data</td>
<td>thirti</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>employe</td>
<td>interv</td>
<td></td>
</tr>
</tbody>
</table>

- Words in concepts are somewhat informative.
- But high degree of overlap.
Model Avian Influenza Virus

from Lam&Boley 2011

- Evolution is a flow, naturally falls in chronological order.
- Without vaccine, picture is more a random cloud of points.
- Suggests vaccine use does affect evolution of virus.
Model Avian Influenza Virus

(Lam et al., 2012)

- Avian Flu Virus characterized by the HA protein, which the virus uses to penetrate the cell.

- The protein is described by a string of 566 symbols, each representing one of 20 Amino Acids.

- Embed in high dimensional Euclidean space by replacing each Amino Acid with a string of 20 bits:
  
  - E.g. 3rd amino acid = → 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

- Result is a vector of length $566 \cdot 20 = 11230$.

- Use PCA to reduce dimensions from 11320 to 6.

- Use first 2 components to track evolution of this protein in a simple visual way.
Outline

- Dimensionality Reduction
  - Principal Component Analysis – PCA
  - Latent Semantic Indexing
  - Clustering

- Graph Partitioning
  - Principal Direction Divisive Partitioning
  - Spectral Partitioning

- Sparse Representation – Examples
  - almost shortest path routing.
  - constrained clustering.
  - image/vision,
  - Graph Connection Discovery.

- Finding Sparse Representation
Principal Direction Divisive Partitioning

(Boley, 1998)
Divisive Partitioning for Unsupervised Clustering

• Unsupervised, as opposed to Supervised:
  • No predefined categories;
  • No previously classified training data;
  • No a-priori assumptions on the number of clusters.

• Top-down Hierarchical:
  • Imposes a tree hierarchy on unstructured data;
  • Tree is source for some taxonomic information for dataset;
  • Tree is generated from the root down.
  • Result is Principal Direction Divisive Partitioning. (Boley, 1998)

• Multiway Clustering.
  • Project onto first $k$ principal directions. Result: each data sample is represented by $k$ components.
  • Apply classical k-means clustering to projected data.
  • Used for both Graph Partitioning and Data Clustering. (Dhillon, 2001)

• Empirically Best Approach: a hybrid method:
  • Use Divisive Partitioning first (deterministic).
  • Refine with K-means (avoids initialization issues). (Savaresi & Boley, 2004)
PDDP on 98 Document Set

- Loadings of top two concepts on set of 98 documents with 5623 words.
### Top distinctive words in top 3 clusters

<table>
<thead>
<tr>
<th>PC 1</th>
<th>PC 2</th>
<th>PC 3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>minus end</strong></td>
<td><strong>plus end</strong></td>
<td><strong>minus end</strong></td>
</tr>
<tr>
<td>employ</td>
<td>manufactur</td>
<td>busi</td>
</tr>
<tr>
<td>action</td>
<td>engin</td>
<td>capit</td>
</tr>
<tr>
<td>employe</td>
<td>system</td>
<td>fund</td>
</tr>
<tr>
<td>affirm</td>
<td>integr</td>
<td>credit</td>
</tr>
<tr>
<td>servic</td>
<td>process</td>
<td>invest</td>
</tr>
<tr>
<td>mean</td>
<td>technologi</td>
<td>corpor</td>
</tr>
<tr>
<td>law</td>
<td>develop</td>
<td>investor</td>
</tr>
<tr>
<td>job</td>
<td>project</td>
<td>debt</td>
</tr>
<tr>
<td>right</td>
<td>tool</td>
<td>source</td>
</tr>
<tr>
<td>public</td>
<td>design</td>
<td>compani</td>
</tr>
<tr>
<td>feder</td>
<td>industri</td>
<td>offer</td>
</tr>
<tr>
<td>act</td>
<td>product</td>
<td>stock</td>
</tr>
<tr>
<td>copyright</td>
<td>research</td>
<td>click</td>
</tr>
<tr>
<td>osha</td>
<td>machin</td>
<td>tax</td>
</tr>
<tr>
<td>person</td>
<td>data</td>
<td>lease</td>
</tr>
</tbody>
</table>

**labor** manufacturing **business** labor communication manufacturing
Spectral Graph Partitioning

• Model an undirected graph by a random walk.

• Measure distance between two nodes by average round-trip commute time (average number of steps to go from node $i$ to $j$ and back again.)

• Vertices of an undirected connected graph can be embedded in high-dimensional Euclidean space.

• Embedding preserves distances between the vertices.

• Principal Direction splitting on embedding is equivalent to two-way Spectral Graph Partitioning.

• Much more popular in graph setting.

• Can be extended to directed graphs (e.g., commute times still a metric). (Boley et al., 2011)
Outline

• Dimensionality Reduction
  • Principal Component Analysis – PCA
  • Latent Semantic Indexing
  • Clustering

• Graph Partitioning
  • Principal Direction Divisive Partitioning
  • Spectral Partitioning

• Sparse Representation – Examples
  • almost shortest path routing.
  • constrained clustering.
  • image/vision,
  • Graph Connection Discovery.

• Finding Sparse Representation
Sparse Representation

• Many machine learning algorithms can explore massive data:
  K-nearest Neighbors, Kernel-SVM, Boosting, Metric Learning, . . .

• All can benefit from denoising by finding a sparse representation:

\[
\text{raw datum} \quad \begin{array}{c}
\text{dictionary atoms} \\
\end{array} \quad \times \quad \begin{array}{c}
\text{sparse representation} \\
\end{array}
\]

• Must find best fit, subject to sparsity limit.

• Optionally must learn the dictionary.
Almost Shortest Path Routing

\[
\min_{x} \ x^T W x + \lambda \|x\|_1 = \sum_{ij \in E} X_{ij}^2 w_{ij} + \lambda |X_{ij}| \\
\text{s.t.} \quad \sum_{i : ik \in E} X_{ik} = \sum_{j : kj \in E} X_{kj} \quad \forall k
\]

minimize total flow energy
flow in = flow out at every node \( k \)

(Li et al., 2011)
Constrained Clustering

- Graph Clustering with Must-link and Cannot-link constraints.
- Spectral Graph Cut: $x^T L x$ [where $L$ = Laplacian].
- Previous approach: minimize $x^T L x + \lambda x^T L_c x$ (Shi et al., 2010).
- Our approach: minimize cut with $L1$ penalty on constraint violations: $x^T L x + \lambda \| C_c x \|_1$ [Kawale et al].
Image Descriptors

Image Descriptor

- Pixel Descriptors: for $i$-th pixel $z_i = \phi(x_i, y_i)$ is a vector of descriptors for the pixel at point $(x_i, y_i)$ in the image.

- Example, could use $z_i = (I_x, I_y, |\text{grad}I|, \angle\text{grad}I, I_{xx}, I_{xy}, I_{yy})$ where $I$ is the intensity value. Could also incorporate color information.

Covariance Descriptor (Tuzel et al., 2006)

- Within each small patch around each pixel compute the covariance $C_i$ of the pixel descriptors.

- Covariance descriptors eliminate differences due to scaling, brightness, large shadows, but enhance local features.

- Use for object detection, tracking, recognition, and more . . .

- Each $C_i$ is a small positive semi-definite matrix ($7 \times 7$ in this example).

- Regularize each $C_i$ by adding a small multiple of the identity.
Covariance Descriptor Example

Raw Image

- Raw Image
- first derivatives
  - x-grad
  - y-grad
- second derivatives
  - grad-mag
  - grad-dir
- pixel by pixel descriptor
  - Covariance descriptor

Dxx
Dxy
Dyy
Covariance Descriptor Usage

- **Object Detection and Tracking in Image.**
  
  **Object Detection**
  - face
  - license plate
  - human

  (Opelt et al., 2004; Sivalingam et al., 2011)

  (Porikli & Kocak, 2006)

  (Tuzel et al., 2007)

  (Paliaio et al., 2009)

  **Object Tracking**

  Image Source: Google Image

  Image Source: Google Image

  Image Source: Google Image

- **Object Recognition**
  - face
  - action
  - palmprint

  (Pang et al., 2008)

  KTH dataset

  (Han et al., 2009)
Optimization Setup for Covariances

Notation: (Sivalingam et al., 2010; Sivalingam et al., 2011)

• \( S \) = a raw covariance matrix,
  \( x \) = vector of unknown coefficients.
  \( A = (A_1, A_2, \ldots, A_k) \) = collection of dictionary atoms.
  \( x = (x_1, x_2, \ldots, x_k) \) = vector of unknown coefficients.

• Goal: Approximate \( S = A_1x_1 + \cdots + A_kx_k = A \cdot x \).

• Use “logdet” divergence as measure of discrepancy:
  \( D_{ld}(A \cdot x, S) = tr((A \cdot x)S^{-1}) - \log \det((A \cdot x)S^{-1}) - n. \)

• Logdet divergence measures relative entropy between two different zero-mean multivariate Gaussians.
Optimization Problem for Covariances

(Sivalingam et al., 2010; Sivalingam et al., 2011)

• Leads to optimization problem

\[
\min_x \sum_i x_i \text{tr}(A_i) - \log \det \left[ \sum_i x_i A_i \right] + \lambda \sum_i x_i
\]

\[
\text{Dist}(A \cdot x, S)\quad \text{Dist}(A \cdot x, S)
\]

s.t. \( x \geq 0 \)

\[
\sum_i x_i A_i \preceq 0 \quad \text{(positive semi-definite)}
\]

\[
\sum_i x_i A_i \preceq S \quad \text{(residual positive semi-def.)}
\]

• This is in a standard form for a MaxDet problem.

• The sparsity term is a relaxation of true desired penalty: \# nonzeros in \( x \).

• Convex problem solvable by e.g. the CVX package (Grant & Boyd, 2010).
Graph Connections Discovery

- Signal at node \( i \) is gaussian & correlated to neighbors, but conditionally independent of signal at unconnected node \( j \).

- Statistical Theory \( \implies (\text{Covariance})^{-1}_{ij} = 0 \).
  \( (\text{Covariance})^{-1} \) is called the Precision Matrix.

- If graph is sparse, expect \( (\text{Covariance})^{-1} \) to be sparse.

- Problem: Graph connections are unknown.

- Task: Given signals at each node, recover graph edges.

- Applications: biology, climate modelling, social networks.

- Method:
  - Compute sample precision matrix from signals.
  - Find best \textit{sparse} approximation to sample precision matrix.
  - Use previous log-det divergence to measure discrepancy between covariance matrices.
Outline

- Dimensionality Reduction
  - Principal Component Analysis – PCA
  - Latent Semantic Indexing
  - Clustering

- Graph Partitioning
  - Principal Direction Divisive Partitioning
  - Spectral Partitioning

- Sparse Representation – Examples
  - almost shortest path routing.
  - constrained clustering.
  - image/vision,
  - Graph Connection Discovery.

- Finding Sparse Representation
Constructing Sparse Basis

- **Matching Pursuit**: (Mallat & Zhang, 1993)
  - Greedy algorithm: try every column not already in your basis;
  - evaluate quality of new column if it were added to your basis;
  - add “best” column to your basis, and repeat until satisfied.

- **Basis Pursuit** (Chen et al., 2001)
  - Minimize \( \|b - Ax\|_2^2 + \lambda \|x\|_0 \).
  - Difficulty: this is a NP-hard combinatorial problem.
  - Relax to \( \|b - Ax\|_2^2 + \lambda \|x\|_1 \).
  - Relaxed problem is convex, so solvable more efficiently.
  - LASSO: Solve for all \( \lambda \) fast (Tibshirani, 1996).
Convex Relaxation $\implies$ LASSO

- Known as Basis Pursuit, Compressed Sensing, "small error + sparse".
- Add penalty for number of nonzeros with weight $\lambda$:
  \[
  \min_x \|Ax - b\|_2^2 + \lambda\|x\|_0.
  \]
- Convert hard combinatorial problem into easier convex optimization problem.
- Relax previous $\|x\|_0$ to convex problem:
  \[
  \min_x \|Ax - b\|_2^2 + \lambda\|x\|_1,
  \]
- or convert to constrained problem:
  \[
  \min_x \|Ax - b\|_2^2 \text{ subject to } \|x\|_1 \leq \text{tol}.
  \]
- Vary parameter $\lambda$ or tol, to explore the trade-off between "small error" and "sparse".
Motivation: find closest sparse point

Find closest point to target \((3.5, 1.5)\) with 1-norm constraint.

As soon as one coordinate reaches zero, it is removed, and the remaining coordinates are driven to zero.

- Find closest point to target \(\ldots\) subject to \(\ell_1\) norm constraint.
Motivation: find closest sparse point

As limit on $\|x\|_1$ is tightened, the coordinates are driven toward zero.

As soon as one coordinate reaches zero, it is removed, and the remaining coordinates are driven to zero.
Example: 17 signals with 10 time points

- As $\lambda$ grows, the error grows, fill (#non-zeros) shrinks.
Methods

• All problems are convex.

• Must work exists on software for convex programming problems

• YALMIP is a front end with links to many solver packages (Löfberg, 2004).

• CVX is a free package of convex solvers with easy matlab interface (Grant & Boyd, 2010).

• ADMM is a paradigm for a simple iterative solver especially adapted for very large but separable problems (Boyd et al., 2011).
Conclusions

- Many different types of data, many highly unstructured.
- Extracting patterns or connections in data involves somehow reducing the volume of data one must look at.
- Data Reduction is an old paradigm that has been updated for the modern digital age.
- Methods discussed here started with classical PCA - SVD based approaches (e.g., assuming independent gaussian noise).
- Connections and pair-wise correlations modeled by graphs.
- Graphs modeled by random walks, counting subgraphs, min-cut/max-flow, models, ....
- Sparse representations: wide variety of sparse approximations: low fill, short basis, non-negative basis, non-squared loss function, count violations of some constraints, low rank (nuclear norm = $L_1$-norm on the singular values), ....
- Leads to need for scalable solvers for very large convex programs.
THANK YOU!
References


