Two Technique Papers on High Dimensionality

Allan Rempel
December 5, 2005

Papers


- well written, clear, appropriately detailed
- High-dim and MDS can be complicated

Dimensionality reduction

- Mapping high-dimensional data to 2D space
- Could be done many different ways
- Different techniques satisfy different goals
- Familiar example - projection of 3D to 2D preserves geometric relationships
- Abstract data may not need that

Multidimensional scaling (MDS)

- Display multivariate abstract point data in 2D
  - Data from bioinformatics, financial sector, etc.
  - No inherent mapping in 2D space
  - p-dim embedding of q-dim space (p < q) where inter-object relationships are approximated in low-dimensional space
- Proximity in high-D -> proximity in 2D
  - High-dim distance between points (similarity) determines relative (x,y) position
  - Absolute (x,y) positions are not meaningful
- Clusters show closely associated points

Multidimensional scaling (MDS)

- Eigenvector analysis of \( N \times N \) matrix – \( O(N^2) \)
  - Need to recompute if data changes slightly
  - Iterative \( O(N^2) \) algorithm – Chalmers,1996
- This paper – \( O(N\sqrt{N}) \)
- Next paper – \( O(N \log N) \)

Multidimensional scaling (MDS)

- Proximity data
  - In social sciences, geology, archaeology, etc.
  - E.g. library catalogue query – find similar points
    - Multi-dimensional scatterplot not possible
  - Want to see clusters, curves, etc.
    - Features that stand out from the noise
- Distance function
  - Typically use Euclidean distance – intuitive
Spring models

- Used instead of statistical techniques (PCA)
  - Better convergence to optimal solution
  - Iterative – steerable – Munzner et al, 2004
- Good aesthetic results – symmetry, edge lengths
- Basic algorithm – $O(N^3)$
  - Start: place points randomly in 2D space
  - Springs reflect diff btwn high-D and 2D distance
  - #iterations required is generally $O(N)$

Chalmers’ 1996 algorithm

- Approximate solution works well
- Caching, stochastic sampling – $O(N^2)$
  - Perform each iteration in $O(N)$ instead of $O(N^2)$
  - Keep constant-size set of neighbours
  - Constants as low as 5 worked well
- Still only worked on datasets up to few 1000s

Hybrid methods of clustering and layout

- Diff clustering algorithms have diff strengths
  - Kohonen’s self-organising feature maps (SOM)
  - K-means iterative centroid-based divisive alg.
- Hybrid methods have produced benefits
- Neural networks, machine learning literature

New hybrid MDS approach

- Start: run spring model on subset of size $\sqrt{N}$
  - Completes in $O(N)$ ($O(\sqrt{N} \times \sqrt{N})$)
- For each remaining point:
  - Place close to closest ‘anchor’
  - Adjust by adding spring forces to other anchors
- Overall complexity $O(N \sqrt{N})$

Experimental results

- 3-D data sets: 5000 – 50,000 points
- 13-D data sets: 2000 – 24,000 points
- Took less than 1/3 the time of the $O(N^3)$
- Achieved lower stress when done
- Also compared against original $O(N^3)$ model
  - 9 seconds vs. 577; and 24 vs. 3642
  - Achieved much lower stress (0.06 vs. 0.2)
Future work

- Hashing
- Pivots – Morrison, Chalmers, 2003
  - Achieved $O(N\sqrt[3]{N})$
- Dynamically resizing anchor set
- Proximity grid
  - Do MDS, then transform continuous layout into discrete topology

Improving parent-finding strategy

- Select constant-size subset $P \subset S$
- For each $p$ in $P$ create sorted list $L_p$
- For each remaining point $u$, binary search $L_p$ for point $u_p$ as distant from $p$ as $u$ is
  - Implies that $u$ and $u_p$ are very close
- Place $u$ according to location of $u_p$

Comparison

- Chalmers et al is better for $N < 5500$
- Main diff is in parent-finding, represented by Fig. 3

Quality of output

- MDS theory uses stress to objectively determine quality of placement of points
- Subjective determinations can be made too
  - 2D small world network example (500 – 80,000 nodes)
Recursively defining the initial kernel set of points can yield much better real-time performance

Conclusions and future work

- Series of results yielding progressively better time complexities for MDS
- 2D mappings provide good representations
- Further examination of multiscale approach
- User-steerable MDS could be fruitful