Lecture 10: High Dimensionality Information Visualization
CPSC 533C, Fall 2007
Tamara Munzner
UBC Computer Science
15 October 2007

Readings Covered

Further Reading

Parallel Coordinates
- only 2 orthogonal axes in the plane
- instead, use parallel axes!

PC: Correlation
- rotate-translate
- point-line
- pencil: set of lines coincident at one point

PC: Duality
- geometric interpretations
- hyperplane, hypersphere
- points do have intrinsic order
- Infvis
- no intrinsic order, what to do?
- indeterminate/arbitrary order
  - weakness of many techniques
  - downside: human-powered search
  - upside: powerful interaction technique
- most implementations
- user can interactively swap axes
- Automated Multidimensional Detective
- Inselberg 99
- machine learning approach

Hierarchical Parallel Coords: LOD

Hierarchical Clustering
- proximity-based coloring
- interaction lecture later:
  - structure-based brushing
  - extent scaling

Dimensionality Reduction
- mapping multidimensional space into
  - space of fewer dimensions
  - typically 2D for Infvis
  - keep/explain as much variance as possible
  - show underlying dataset structure
- multidimensional scaling (MDS)
- MDS: minimize differences between interpoint distances in high and low dimensions

Dimensionality Reduction: Isomap
- 4096 D: pixels in image
- 2D: wrist rotation, fingers extension

From Matrix To Graph

MDS vs GLA: Dimensionality
- MDS better when dim = 2D/3D
  - low stress
- GLA better when dim > 2
  - 2D MDS shows MLK intermediate
  - GLA shows MLK part of patriotic group
  - 3D MDS also shows MLK part of patriotic

MDS vs GLA: Outliers
- outliers distort with MDS
- outliers automatically handled with GLA

MDS vs GLA
- intransitivity (triangle inequality doesn’t hold): GLA better
- data asymmetric: GLA better
- interactive exploration (changing filter): GLA allows
- manual node repositioning: GLA allows
- existence/absence of relationships at precise levels: GLA
- overview of all relationships at once: MDS

Visualizing Proximity Data
- characterizing MDS vs. graph layout
- MDS
  - nonmetric: ordering preserved, not exact distances
  - general clusters meaningful, specific local distances less so
- Isomap
- metric (stress=.269) nonmetric (stress=.171)

Parallel Coordinates
- only 2 orthogonal axes in the plane
- instead, use parallel axes!
Critique

- somewhat evangelical pro-graph stance
  but we could use more such characterizations

Faster Spring Model [Chalmers 96]

- compare distances only with a few points
  - maintain small local neighborhood set
  - each time pick some randoms, swap in if closer

Issues

- which distance metric: Euclidean or other?
- computation
  - naive: $O(n^3)$
  - better: $O(n^2)$ Chalmers 96
  - hybrid: $O(n\sqrt{n})$

True Dimensionality: Linear

- how many dimensions is enough?
  - could be more than 2 or 3
  - knee in error curve
- example
  - measured materials from graphics
  - linear PCA: 25
  - get physically impossible intermediate points

True Dimensionality: Nonlinear

- nonlinear MDS: 10-15
- all intermediate points possible
- categorizable by people
  - red, green, blue, specular, diffuse, glossy, metallic, plastic-y, roughness, rubbery, greasiness, dustiness...

MDS Beyond Points

- galaxies: aggregation

- themescapes: terrain/landscapes

Cluster Stability

- display
  - also terrain metaphor
- underlying computation
  - energy minimization (springs) vs. MDS
  - weighted edges
- do same clusters form with different random start points?
  - “ordination”
  - spatial layout of graph nodes

Approach

- criteria
  - geometric distance matching graph-theoretic distance
    - vertices one hop away close
    - vertices many hops away far
    - insensitive to random starting positions
    - major problem with previous work
    - tractable computation
- force-directed placement
  - discussion: energy minimization
  - others: gradient descent, etc
  - discussion: termination criteria

Graph Layout

- repeat for all points
  - compute spring force to all other points
  - difference between high dim, low dim distance
  - move to better location using computed forces

- compute distances between all points
  - $O(n^2)$ iteration, $O(n^2)$ algorithm

Barrier Jumping

- same idea as simulated annealing
  - but compute directly
  - just ignore repulsion for fraction of vertices
  - solves start position sensitivity problem
## Results

- **Efficiency**
  - naive approach: $O(V^2)$
  - approximate density field: $O(V)$
- **Good Stability**
  - rotation/reflection can occur

## Critique

- **Pro**
  - approach on multiple techniques,
  - real data!
- **Con**
  - always show order then space then filter
  - hard to tell which is effective
  - show ordered vs. unordered after zoom/filter?

## Results: Scatterplot Matrices

- raw, filter

## Results: InterRing

- raw, order, distort, rollup (filter)

## Results: Parallel Coordinates

- raw, order/space, zoom, filter

## Results: Star Glyphs

- raw, order/space, distort, filter

## Results: Scatterplot Matrices

- raw, filter

## Software, Data Resources

- www.cs.ubc.ca/~tmm/courses/infovis/resources.html

---

## Dimension Ordering

- in NP, like most interesting infovis problems
- heuristic
- divide and conquer
  - iterative hierarchical clustering
  - representative dimensions
- choices
  - similarity metrics
  - importance metrics
  - variance
  - ordering algorithms
  - optimal
  - random swap
  - simple depth-first traversal

## Spacing, Filtering

- same idea: automatic support
- interaction
  - manual intervention
  - structure-based brushing
  - focus-context, next week

## Critique

- **Pro**
  - approach on multiple techniques,
  - real data!
- **Con**
  - always show order then space then filter
  - hard to tell which is effective
  - show ordered vs. unordered after zoom/filter?