CSE 332
Introduction to Visualization

High-Dimensional Data

Klaus Mueller

Computer Science Department
Stony Brook University
<table>
<thead>
<tr>
<th>Lecture</th>
<th>Topic</th>
<th>Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Intro, schedule, and logistics</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Applications of visual analytics, data types</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Data sources and preparation</td>
<td>Project 1 out</td>
</tr>
<tr>
<td>4</td>
<td>Data reduction, similarity &amp; distance, data augmentation</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Dimension reduction</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Introduction to D3</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Visual communication using infographics</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Visual perception and cognition</td>
<td>Project 2 out</td>
</tr>
<tr>
<td>9</td>
<td>Visual design and aesthetic</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>D3 hands-on presentation</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Cluster analysis</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Visual analytics tasks and design</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td><strong>High-dimensional data VIS, dimensionality reduction</strong></td>
<td>Project 3 out</td>
</tr>
<tr>
<td>14</td>
<td>Visualization of spatial data: volume visualization intro</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Visualization of spatial data: raycasting, transfer functions</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Midterm</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Illumination and isosurface rendering</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>Scientific visualization</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Non-photorealistic and illustrative rendering</td>
<td>Project 4 out</td>
</tr>
<tr>
<td>20</td>
<td>Midterm discussion</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>Principles of interaction</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Visual analytics and the visual sense making process</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>Visualization of graphs and hierarchies</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Visualization of time-varying and streaming data</td>
<td>Project 5 out</td>
</tr>
<tr>
<td>25</td>
<td>Maps</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>Memorable visualizations, visual embellishments</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>Evaluation and user studies</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>Narrative visualization, storytelling, data journalism, XAI</td>
<td></td>
</tr>
</tbody>
</table>
Feature vectors are typically high dimensional
- this means, they have many elements
- high dimensional space is tricky
- most people do not understand it
- why is that?
- well, because you don’t learn to see high-D when your vision system develops

Object permanence (Jean Piaget)
- the ability to create mental pictures or remember objects and people you have previously seen
- thought to be a vital precursor to creativity and abstract thinking
The curse of dimensionality
As $n \to \infty$
- Cube: side length $l$, diagonal $d$, volume $V$
- $V \to \infty$ for $l > 1$
- $V \to 0$ for $l < 1$
- $V = 0$ for $l = 1$
- $d \to \infty$

and very sparse

most points are here

and not here
High-D Space is Tricky

Essentially hypercube is like a “hedgehog”
Points are all at about the same distance from one another

- concentration of distances
- fundamental equation (Bellman, ’61)

\[
\lim_{n \to \infty} \frac{\text{Dist}_{\text{max}} - \text{Dist}_{\text{min}}}{\text{Dist}_{\text{min}}} \to 0
\]

- so as \( n \) increases, it is impossible to distinguish two points by (Euclidian) distance
  - unless these points are in the same cluster of points
Space gets extremely sparse

- with every extra dimension points get pulled apart further
- distances become meaningless
Space gets extremely sparse

- with every extra dimension points get pulled apart further
- distances become meaningless

1D – points are very close

2D – points spread apart

3D – getting even sparser

4D, 5D, ... – sparseness grows further
Indexing (and storage) also gets very expensive

- exponential growth in the number of dimensions

- 4D: 65k cells  
- 5D: 1M cells  
- 6D: 16M cells  
- 7D: 268M cells

- keep a keen eye on storage complexity
Parallel Coordinates
The N=7 data axes are arranged side by side
  - in parallel
Hard to see the individual cars?
- what can we do?
Grouping the cars into sub-populations

- a clustering operation
- an be automated or interactive (put the user in charge)
Computes the mean and superimposes it onto the lines
  - allows one to see trends
PC With Illustrative Abstraction

individual polylines
completely abstracted away
PC With Illustrative Abstraction

blended partially
PC With Illustrative Abstraction

all put together – three clusters

[McDonnell and Mueller, 2008]
Interaction is Key

Interaction in Parallel Coordinate
PATTERNS IN PARALLEL COORDINATES

correlation  r=-1.0  r=0  r=1.0
Patterns in Parallel Coordinates

# points

correlation
Li et al. found that twice as many correlation levels can be distinguished with scatterplots.
There are $n!$ ways to order the $n$ dimensions

- how many orderings for 7 dimensions?
- 5,040
- but since can see relationships across 3 axes a better estimate is $n!/(n-3)! 3!) = 35$
- still a lot of axes orderings to try out → we need help
The below is not an optimal ordering, why?
This ordering is better, why?

- because it doesn’t waste axis pairs on uncorrelated relationships
- only region 3 is uncorrelated
- regions 1 and 2 are subspace clusters
For each axis pair, compute correlation

Compute optimal-cost path across all attributes

What algorithm does this?
- Traveling Salesman Solver

Do the same for the correlation plot
Parallel Sets

Developed by [Kosara et al. TVCG, 2006]

Parallel coordinates for categorical data
- for example, census and survey data, inventory, etc.
- data that can be summed up in a cross-tabulation

Example
- Titanic dataset
- what can we see here?
Story Telling with Parallel Coordinates
ANATOMY OF A SALES PIPELINE

lead pool

lead generator

lead
qualified lead
opportunity
opportunity¹
opportunity²
opportunity³
account manager

responds
responds²
requests info (RFI)
requests pricing (RFP)
shapes deal
signs/buys

customer

action

potential

prospect
Scene:
  ▪ a meeting of sales executives of a large corporation, Vandelay Industries

Mission:
  ▪ review the strategies of their various sales teams

Evidence:
  ▪ data of three sales teams with a couple of hundred sales people in each team
Meet Kate, a sales analyst in the meeting room:

“OK...let’s see, cost/won lead is nearby and it has a positive correlation with #opportunities but also a negative correlation with #won leads”
“Let’s go and make a revealing route!”

- she uses the mouse and designs the route shown
- she starts explaining the data like a story ...
Kate notices something else:

- now looking at the red team
- there seems to be a spread in effectiveness among the team
- the team splits into three distinct groups

She recommends: “Maybe fire the least effective group or at least retrain them”
SCATTERPLOTS

Projection of the data items into a bivariate basis of axes
How does 2D projection work in practice?

- N-dimensional point \( x = \{x_1, x_2, x_3, \ldots, x_N\} \)
- A basis of two orthogonal axis vectors defined in N-D space
  - \( a = \{a_1, a_2, a_3, \ldots, a_N\} \)
  - \( b = \{b_1, b_2, b_3, \ldots, b_N\} \)
- A projection \( \{x_a, x_b\} \) of \( x \) into the 2D basis spanned by \( \{a, b\} \) is:
  - \( x_a = a \cdot x^T \)
  - \( x_b = b \cdot x^T \)

where \( \cdot \) is the dot product, \( T \) is the transpose.
Projection causes inaccuracies

- close neighbors in the projections may not be close neighbors in the original higher-dimensional space
- this is called projection ambiguity
Appropriate for the display of bivariate relationships
What to do when there are more than two variables?

- arrange multivariate relationships into scatterplot matrices
- not overly intuitive to perceive multivariate relationships
Scatterplot Matrix (SPLOM)

Climatic predictors

- WetDays
- TempJuly
- TempJan
- TempAnn
- RHJuly
Scatterplot version of parallel coordinates

- distributes \( n(n-1) \) bivariate relationships over a set of tiles
- for \( n=4 \) get 16 tiles
- can use \( n(n-1)/2 \) tiles

For even moderately large \( n \):
- there will be too many tiles

Which plots to select?
- plots that show correlations well
- plots that separate clusters well
Several metrics, a good one is Distance Consistency (DSC)

\[ DSC = \frac{|x' \in v(X) : CD(x', \text{centr}'(c_{\text{clabel}(x))) = true|}{k} \]

- measures how “pure” a cluster is
- pick the views with highest normalized DSC

Plots data points and dimension axes into a single visualization

- uses first two PCA vectors as the basis to project into
- find plot coordinates \([x] [y]\)
  - for data points: \([\text{PCA}_1 \cdot \text{data vector}] [\text{PCA}_2 \cdot \text{data vector}]\)
  - for dimension axes: \([\text{PCA}_1[\text{dimension}]] [\text{PCA}_2[\text{dimension}]]\)
See data distributions into the context of their attributes
See data points into the context of their attributes
Do be aware that the projections may not be fully accurate

- you are projecting N-D into 2D by a linear transformation
- if there are more than 2 significant PCA vectors then some variability will be lost and won’t be visualized
- remote data points might project into nearby plot locations suggesting false relationships \(\rightarrow\) projection ambiguity
- always check out the PCA scree plot to gauge accuracy
Also called multivariate scatterplot

- biplot-axes length vis replaced by graphical design
- less cluttered view
- but there’s more to this .....

INTERACTIVE BIPILOTS
Trackball-Based Cluster Exploration
CHASE INTERESTING CLUSTERS –
TRANSITION TO ADJACENT 3D SUBSPACES
EDIT AND ANNOTATE CLUSTERS
Clarify Spatial Relationships
CLARIFY SPATIAL RELATIONSHIPS
Star Coordinates

Coordinate system based on axes positioned in a star

- a point $P$ is vector sum of all axis coordinates

Interactions

- axis rescaling, rotation
- reveal correlations
- resolve plotting ambiguities

\[ P = O + \sum_{i=1}^{m} d_i \vec{c}_i \]
Similar to Star Coordinates

- uses a spring model
- difference is normalization by sum of values

\[
P = \frac{\sum_{i=1}^{m} d_i \tilde{c}_i}{\sum_{i=1}^{m} d_i}
\]

Figure by: Rubio-Sanchez et al. TVCG 2015

[P. Hoffman et al. VIS 1997]
Optimizing the RadViz Layout

Standard RadViz

Optimization on circle using TSP iteratively using similarity constraints

[Cheng and Mueller, Pacific Vis 2015]
Star Coordinates

Coordinate system based on axes positioned in a “star”, or circular pattern

- a point P is plotted as a vector sum of all axis coordinates
Star Coordinates

Operations defined on Star Coords

- scaling changes contribution to resulting visualization
- axis rotation can visualize correlations
- also used to reduce projection ambiguities

Similar paradigm: RadViz
All of these scatterplot displays share the following characteristics

- allow users to see the data points in the context of the variables
- but can suffer from projection ambiguity
- some offer interaction to resolve some of these shortcomings
- but interaction can be tedious

Are there visualization paradigms that can overcome these problems?

- yes, algorithms that optimize the layout to preserve distances or similarities in high-dimensional space
- these are also called *lower-dimensional embeddings*
- very popular is MDS (Multi-dimensional scaling)
MDS is for irregular structures
- scattered points in high-dimensions (N-D)
- adjacency matrices

Maps the distances between observations from N-D into low-D (say 2D)
- attempts to ensure that differences between pairs of points in this reduced space match as closely as possible

The input to MDS is a distance (similarity) matrix
- actually, you use the *dissimilarity* matrix because you want similar points mapped closely
- dissimilar point pairs will have greater values and map farther apart
### The Dissimilarity Matrix

#### Data Matrix

<table>
<thead>
<tr>
<th>point</th>
<th>attribute1</th>
<th>attribute2</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>x2</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>x3</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>x4</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>

#### Dissimilarity Matrix (with Euclidean Distance)

<table>
<thead>
<tr>
<th></th>
<th>x1</th>
<th>x2</th>
<th>x3</th>
<th>x4</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x2</td>
<td>3.61</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x3</td>
<td>2.24</td>
<td>5.1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>x4</td>
<td>4.24</td>
<td>1</td>
<td>5.39</td>
<td>0</td>
</tr>
</tbody>
</table>
MDS turns a distance matrix into a network or point cloud
  - correlation, cosine, Euclidian, and so on

Suppose you know a matrix of distances among cities

<table>
<thead>
<tr>
<th></th>
<th>Chicago</th>
<th>Raleigh</th>
<th>Boston</th>
<th>Seattle</th>
<th>S.F.</th>
<th>Austin</th>
<th>Orlando</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chicago</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Raleigh</td>
<td>641</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boston</td>
<td>851</td>
<td>608</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seattle</td>
<td>1733</td>
<td>2363</td>
<td>2488</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S.F.</td>
<td>1855</td>
<td>2406</td>
<td>2696</td>
<td>684</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Austin</td>
<td>972</td>
<td>1167</td>
<td>1691</td>
<td>1764</td>
<td>1495</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Orlando</td>
<td>994</td>
<td>520</td>
<td>1105</td>
<td>2565</td>
<td>2458</td>
<td>1015</td>
<td>0</td>
</tr>
</tbody>
</table>
RESULT OF MDS
Compare with real Map
MDS Algorithm

- Task:
  - Find that configuration of image points whose pairwise distances are most similar to the original inter-point distances !!!

- Formally:
  - Define: \( D_{ij} = \| x_i - x_j \|_D \), \( d_{ij} = \| y_i - y_j \|_d \)
  - Claim: \( D_{ij} \equiv d_{ij} \), \( \forall i, j \in [1, n] \)

- In general: an exact solution is not possible !!!
- Inter Point distances \( \rightarrow \) invariance features
MDS Algorithm

Strategy (of metric MDS):

- iterative procedure to find a good configuration of image points

  1) Initialization
     → Begin with some (arbitrary) initial configuration

  2) Alter the image points and try to find a configuration of points that minimizes the following sum-of-squares error function:
MDS Algorithm

Strategy (of metric MDS):

- Iterative procedure to find a good configuration of image points
  
  1) Initialization
     → Begin with some (arbitrary) initial configuration
  
  2) Alter the image points and try to find a configuration of points that minimizes the following sum-of-squares error function:

\[
E = \sum_{i<j}^N (D_{ij} - d_{ij})^2
\]
Spring-like system

- insert springs within each node
- the length of the spring encodes the desired node distance
- start at an initial configuration
- iteratively move nodes until an energy minimum is reached
Spring-like system

- insert springs within each node
- the length of the spring encodes the desired node distance
- start at an initial configuration
- iteratively move nodes until an energy minimum is reached
Distance (similarity) metric

- Euclidian distance (best for data)
- Cosine distance (best for data)
- |1-correlation| distance (best for attributes)
- use 1-correlation to move correlated attribute points closer
- use | | if you do not care about positive or negative correlations
MDS EXAMPLES
Tries to unwrap a high-dimensional surface (A) \( \rightarrow \) manifold
- noisy points could be averaged first and projected onto the manifold

Algorithm
- construct neighborhood graph G \( \rightarrow \) (B)
- for each pair of points in G compute the shortest path distances \( \rightarrow \) geodesic distances
- fill similarity matrix with these geodesic distances
- embed (layout) in low-D (2D) with MDS \( \rightarrow \) (C)
- visualize it like an MDS layout
t-Distributed Stochastic Neighbor Embedding
  - innovated by [L. van der Maaten and G. Hinton, 2008]

Works as a two-stage approach
1. Construct a probability distribution over pairs of high-D points based on similarity
2. Define a similar probability distribution over the points in the low-D map
Self-Organizing Maps (SOM)

Introduced by [T. Kohonen et al. 1996]
- unsupervised learning and clustering algorithm
- has advantages compared to hierarchical clustering
- often realized as an artificial neural network

SOMs group the data
- perform a nonlinear projection from N-dimensional input space onto two-dimensional visualization space
- provide a useful topological arrangement of information objects in order to display clusters of similar objects in information space
Map a dataset of 3D color vectors into a 2D plane

- assume you have an image with 5 colors
- want to see how many there are of each
- compute an SOM of the color vectors
SOM Algorithm

Create array and connect all elements to the N input dimensions
- shown here: 2D vector with 4×4 elements
- initialize weights

For each input vector chosen at random
- find node with weights most like the input vector
- call that node the Best Matching Unit (BMU)
- find nodes within neighborhood radius $r$ of BMU
  - initially $r$ is chosen as the radius of the lattice
  - diminishes at each time step
- adjust the weights of the neighboring nodes to make them more like the input vector
  - the closer a node is to the BMU, the more its weights get altered
SOM Example: Poverty Map

SOM – Result Example

World Poverty Map
A SOM has been used to classify statistical data describing various quality-of-life factors such as state of health, nutrition, educational services etc. Countries with similar quality-of-life factors end up clustered together. The countries with better quality-of-life are situated toward the upper left and the most poverty stricken countries are toward the lower right.

‘Poverty map’ based on 39 indicators from World Bank statistics (1992)
Height represents density or number of documents in the region.
Invented at Pacific Northwest National Lab (PNNL).
You will need to use correspondence analysis (CA)
  - CA is PCA for categorical variables
  - related to factor analysis

Makes use of the $\chi^2$ test
  - what’s $\chi^2$?
A chi-square test is used to investigate relationships Relationships between categorical, or nominal-scale, variables representing attributes of people, interaction techniques, systems, etc.
Data organized in a contingency table – cross tabulation containing counts (frequency data) for number of observations in each category
A chi-square test compares the observed values against expected values
Expected values assume “no difference”
Research question:
- Do males and females differ in their method of scrolling on desktop systems? (next slide)
Chi-square – Example #1

<table>
<thead>
<tr>
<th>Gender</th>
<th>Scrolling Method</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MW</td>
<td>CD</td>
</tr>
<tr>
<td>Male</td>
<td>28</td>
<td>15</td>
</tr>
<tr>
<td>Female</td>
<td>21</td>
<td>9</td>
</tr>
<tr>
<td>Total</td>
<td>49</td>
<td>24</td>
</tr>
</tbody>
</table>

MW = mouse wheel  
CD = clicking, dragging  
KB = keyboard
Chi-square – Example #1

56.0 - 49.0 / 101 = 27.2

(Expected-Observed)^2 / Expected = (28 - 27.2)^2 / 27.2

**Chi-Squares**

<table>
<thead>
<tr>
<th>Gender</th>
<th>Scrolling Method</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MW</td>
<td>CD</td>
</tr>
<tr>
<td>Male</td>
<td>0.025</td>
<td>0.215</td>
</tr>
<tr>
<td>Female</td>
<td>0.032</td>
<td>0.268</td>
</tr>
<tr>
<td>Total</td>
<td>0.057</td>
<td>0.483</td>
</tr>
</tbody>
</table>

\[ \chi^2 = 1.462 \]

Significant if it exceeds critical value (next slide)
Decide in advance on alpha (typically .05)

Degrees of freedom

- \( df = (r - 1)(c - 1) = (2 - 1)(3 - 1) = 2 \)
- \( r = \) number of rows, \( c = \) number of columns

<table>
<thead>
<tr>
<th>Significance Threshold (( \alpha ))</th>
<th>Degrees of Freedom</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>( .1 )</td>
<td>2.71</td>
</tr>
<tr>
<td>( .05 )</td>
<td>3.84</td>
</tr>
<tr>
<td>( .01 )</td>
<td>6.64</td>
</tr>
<tr>
<td>( .001 )</td>
<td>10.83</td>
</tr>
</tbody>
</table>

\( \chi^2 = 1.462 (< 5.99 \therefore \text{not significant}) \)
**Correspondence Analysis (CA)**

Example:

<table>
<thead>
<tr>
<th>Staff Group</th>
<th>Smoking Category</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(1) None</td>
</tr>
<tr>
<td>(1) Senior Managers</td>
<td>4</td>
</tr>
<tr>
<td>(2) Junior Managers</td>
<td>4</td>
</tr>
<tr>
<td>(3) Senior Employees</td>
<td>25</td>
</tr>
<tr>
<td>(4) Junior Employees</td>
<td>18</td>
</tr>
<tr>
<td>(5) Secretaries</td>
<td>10</td>
</tr>
<tr>
<td><strong>Column Totals</strong></td>
<td>61</td>
</tr>
</tbody>
</table>

There are two high-D spaces

- 4D (column) space spanned by smoking habits – plot staff group
- 5D (row) space spanned by staff group – plot smoking habits

Are these two spaces (the rows and columns) independent?

- this occurs when the $\chi^2$ statistics of the table is insignificant
Let’s do some plotting

- compute distance matrix of the rows $CC^T$
- compute Eigenvector matrix $U$ and the Eigenvalue matrix $D$
- sort eigenvectors by values, pick two major vectors, create 2D plot

---

senior employees most similar to secretaries
Next:
- compute distance matrix of the columns $C^TC$
- compute Eigenvector matrix $V$ (gives the same Eigenvalue matrix $D$)
- sort eigenvectors by value
- pick two major vectors
- create 2D plot of smoking categories

Following (next slide):
- combine the plots of $U$ and $V$
- if the $\chi^2$ statistics was significant we should see some dependencies
Interpretation sample (using the $\chi^2$ frequentist mindset)

- relatively speaking, there are more non-smoking senior employees
Plot would now show 193 cases and 9 variables
Let's call it matrix X
Compute X’X to get the Burt Table

Compute Eigenvectors and Eigenvalues
- keep top two Eigenvectors/values
- visualize the attribute loadings of these two Eigenvectors into the Burt table plot (the loadings are the coordinates)
Results of a survey of car owners and car attributes

<table>
<thead>
<tr>
<th></th>
<th>American</th>
<th>European</th>
<th>Japanese</th>
<th>Large</th>
<th>Medium</th>
<th>Small</th>
<th>Family</th>
<th>Sporty</th>
<th>Work</th>
<th>1 Income</th>
<th>1 Incomes</th>
<th>2 Incomes</th>
<th>Own</th>
<th>Rent</th>
<th>Married</th>
<th>Married with Kids</th>
<th>Single</th>
<th>Single with Kids</th>
<th>Female</th>
<th>Male</th>
</tr>
</thead>
<tbody>
<tr>
<td>American</td>
<td>125</td>
<td>0</td>
<td>0</td>
<td>36</td>
<td>60</td>
<td>29</td>
<td>81</td>
<td>24</td>
<td>20</td>
<td>58</td>
<td>67</td>
<td>93</td>
<td>32</td>
<td>37</td>
<td>50</td>
<td>32</td>
<td>6</td>
<td>58</td>
<td>67</td>
<td></td>
</tr>
<tr>
<td>European</td>
<td>0</td>
<td>44</td>
<td>4</td>
<td>4</td>
<td>20</td>
<td>17</td>
<td>23</td>
<td>4</td>
<td>18</td>
<td>26</td>
<td>38</td>
<td>6</td>
<td>13</td>
<td>15</td>
<td>15</td>
<td>1</td>
<td>21</td>
<td>23</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Japanese</td>
<td>0</td>
<td>0</td>
<td>165</td>
<td>2</td>
<td>61</td>
<td>102</td>
<td>75</td>
<td>39</td>
<td>30</td>
<td>74</td>
<td>91</td>
<td>111</td>
<td>54</td>
<td>51</td>
<td>44</td>
<td>62</td>
<td>8</td>
<td>70</td>
<td>92</td>
<td></td>
</tr>
<tr>
<td>Large</td>
<td>35</td>
<td>4</td>
<td>2</td>
<td>42</td>
<td>0</td>
<td>0</td>
<td>30</td>
<td>1</td>
<td>11</td>
<td>20</td>
<td>22</td>
<td>35</td>
<td>7</td>
<td>9</td>
<td>21</td>
<td>11</td>
<td>1</td>
<td>17</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Medium</td>
<td>60</td>
<td>20</td>
<td>61</td>
<td>0</td>
<td>141</td>
<td>0</td>
<td>0</td>
<td>39</td>
<td>13</td>
<td>57</td>
<td>04</td>
<td>106</td>
<td>35</td>
<td>42</td>
<td>51</td>
<td>40</td>
<td>0</td>
<td>70</td>
<td>71</td>
<td></td>
</tr>
<tr>
<td>Small</td>
<td>29</td>
<td>20</td>
<td>102</td>
<td>0</td>
<td>0</td>
<td>151</td>
<td>55</td>
<td>66</td>
<td>30</td>
<td>73</td>
<td>78</td>
<td>101</td>
<td>50</td>
<td>50</td>
<td>37</td>
<td>58</td>
<td>6</td>
<td>62</td>
<td>89</td>
<td></td>
</tr>
<tr>
<td>Family</td>
<td>81</td>
<td>17</td>
<td>76</td>
<td>30</td>
<td>89</td>
<td>55</td>
<td>174</td>
<td>0</td>
<td>0</td>
<td>69</td>
<td>105</td>
<td>130</td>
<td>44</td>
<td>50</td>
<td>79</td>
<td>35</td>
<td>10</td>
<td>83</td>
<td>91</td>
<td></td>
</tr>
<tr>
<td>Sporty</td>
<td>24</td>
<td>23</td>
<td>59</td>
<td>1</td>
<td>39</td>
<td>66</td>
<td>0</td>
<td>106</td>
<td>0</td>
<td>55</td>
<td>51</td>
<td>71</td>
<td>35</td>
<td>35</td>
<td>12</td>
<td>57</td>
<td>2</td>
<td>44</td>
<td>62</td>
<td></td>
</tr>
<tr>
<td>Work</td>
<td>20</td>
<td>4</td>
<td>30</td>
<td>11</td>
<td>13</td>
<td>30</td>
<td>0</td>
<td>0</td>
<td>54</td>
<td>26</td>
<td>28</td>
<td>41</td>
<td>13</td>
<td>16</td>
<td>18</td>
<td>17</td>
<td>3</td>
<td>22</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>1 Income</td>
<td>58</td>
<td>18</td>
<td>74</td>
<td>20</td>
<td>57</td>
<td>73</td>
<td>69</td>
<td>53</td>
<td>25</td>
<td>150</td>
<td>0</td>
<td>80</td>
<td>70</td>
<td>10</td>
<td>27</td>
<td>99</td>
<td>14</td>
<td>47</td>
<td>103</td>
<td></td>
</tr>
<tr>
<td>2 Incomes</td>
<td>67</td>
<td>26</td>
<td>91</td>
<td>22</td>
<td>94</td>
<td>78</td>
<td>105</td>
<td>51</td>
<td>20</td>
<td>0</td>
<td>104</td>
<td>162</td>
<td>22</td>
<td>91</td>
<td>92</td>
<td>10</td>
<td>1</td>
<td>102</td>
<td>62</td>
<td></td>
</tr>
<tr>
<td>Own</td>
<td>93</td>
<td>38</td>
<td>111</td>
<td>35</td>
<td>106</td>
<td>101</td>
<td>130</td>
<td>71</td>
<td>41</td>
<td>80</td>
<td>162</td>
<td>242</td>
<td>0</td>
<td>76</td>
<td>106</td>
<td>52</td>
<td>8</td>
<td>114</td>
<td>126</td>
<td></td>
</tr>
<tr>
<td>Rent</td>
<td>32</td>
<td>6</td>
<td>54</td>
<td>7</td>
<td>35</td>
<td>50</td>
<td>44</td>
<td>35</td>
<td>13</td>
<td>70</td>
<td>22</td>
<td>92</td>
<td>25</td>
<td>3</td>
<td>3</td>
<td>57</td>
<td>7</td>
<td>35</td>
<td>57</td>
<td></td>
</tr>
<tr>
<td>Married</td>
<td>37</td>
<td>13</td>
<td>51</td>
<td>9</td>
<td>42</td>
<td>50</td>
<td>50</td>
<td>35</td>
<td>15</td>
<td>10</td>
<td>91</td>
<td>76</td>
<td>25</td>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>53</td>
<td>48</td>
<td></td>
</tr>
<tr>
<td>Married with Kids</td>
<td>50</td>
<td>15</td>
<td>44</td>
<td>21</td>
<td>51</td>
<td>37</td>
<td>79</td>
<td>12</td>
<td>18</td>
<td>27</td>
<td>82</td>
<td>106</td>
<td>3</td>
<td>0</td>
<td>109</td>
<td>0</td>
<td>0</td>
<td>46</td>
<td>61</td>
<td></td>
</tr>
<tr>
<td>Single</td>
<td>32</td>
<td>15</td>
<td>62</td>
<td>11</td>
<td>40</td>
<td>58</td>
<td>35</td>
<td>57</td>
<td>17</td>
<td>99</td>
<td>10</td>
<td>52</td>
<td>57</td>
<td>0</td>
<td>0</td>
<td>109</td>
<td>0</td>
<td>35</td>
<td>74</td>
<td></td>
</tr>
<tr>
<td>Single with Kids</td>
<td>6</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>6</td>
<td>10</td>
<td>2</td>
<td>3</td>
<td>14</td>
<td>1</td>
<td>0</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>15</td>
<td>13</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Female</td>
<td>58</td>
<td>21</td>
<td>70</td>
<td>17</td>
<td>70</td>
<td>62</td>
<td>83</td>
<td>44</td>
<td>22</td>
<td>47</td>
<td>102</td>
<td>114</td>
<td>35</td>
<td>53</td>
<td>46</td>
<td>35</td>
<td>13</td>
<td>149</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Male</td>
<td>67</td>
<td>23</td>
<td>95</td>
<td>25</td>
<td>71</td>
<td>89</td>
<td>91</td>
<td>62</td>
<td>32</td>
<td>103</td>
<td>82</td>
<td>128</td>
<td>57</td>
<td>48</td>
<td>61</td>
<td>74</td>
<td>2</td>
<td>0</td>
<td>185</td>
<td></td>
</tr>
</tbody>
</table>

more info see [here](#)
Summary table:

<table>
<thead>
<tr>
<th>Singular Value</th>
<th>Principal Inertia</th>
<th>Chi-Square</th>
<th>Percent</th>
<th>Cumulative Percent</th>
<th>4</th>
<th>8</th>
<th>12</th>
<th>16</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.56934</td>
<td>0.32415</td>
<td>970.77</td>
<td>18.91</td>
<td>18.91</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.48352</td>
<td>0.23380</td>
<td>700.17</td>
<td>13.64</td>
<td>32.55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.42716</td>
<td>0.18247</td>
<td>546.45</td>
<td>10.64</td>
<td>43.19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.41215</td>
<td>0.16987</td>
<td>508.73</td>
<td>9.91</td>
<td>53.10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.38773</td>
<td>0.15033</td>
<td>450.22</td>
<td>8.77</td>
<td>61.87</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.38520</td>
<td>0.14838</td>
<td>444.35</td>
<td>8.66</td>
<td>70.52</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.34066</td>
<td>0.11605</td>
<td>347.55</td>
<td>6.77</td>
<td>77.29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.32983</td>
<td>0.10879</td>
<td>325.79</td>
<td>6.35</td>
<td>83.64</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.31517</td>
<td>0.09933</td>
<td>297.47</td>
<td>5.79</td>
<td>89.43</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.28069</td>
<td>0.07879</td>
<td>235.95</td>
<td>4.60</td>
<td>94.03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.26115</td>
<td>0.06820</td>
<td>204.24</td>
<td>3.98</td>
<td>98.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.18477</td>
<td>0.03414</td>
<td>102.24</td>
<td>1.99</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>1.71429</td>
<td>5133.92</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Degrees of Freedom = 324
Most influential column points (loadings):
MCA Example (4)

Burt table plot:
Plot Observations

Top-right quadrant:
- categories single, single with kids, 1 income, and renting a home are associated

Proceeding clockwise:
- the categories sporty, small, and Japanese are associated
- being married, owning your own home, and having two incomes are associated
- having children is associated with owning a large American family car

Such information could be used in market research to identify target audiences for advertisements
A Gartner Magic Quadrant is a culmination of research in a specific market, providing a wide-angle view of the relative positions of the market's competitors.

This concept can be used for other dimension pairs as well:
- essentially require to think of a segmentation of the 4 quadrants.
Figure 1. Magic Quadrant for Business Intelligence and Analytics Platforms

Source: Gartner (February 2014)
Gartner
Magic Quadrant
Business Intelligence
2013 vs. 2014
Submission site is not operational at this point
- turns out Blackboard supports peer review as well
- will investigate it this evening
- we may switch to Blackboard after all
- will send email by tomorrow morning with more info
- due date extended to Monday 10/22

Video recording
- a good program is Apowersoft Screen Recorder
- captures screen and voice at the same time
- it’s free for a version with sufficient capabilities
Theme: compare several visualization techniques for high-D data
- use D3 for visualization and python for analysis when needed
- use the data you selected with your 10 favorite attributes

Make separate web pages for the following (10 points for each):
1. bivariate scatterplot (user picks any two variables from a menu)
2. $10 \times 10$ correlation matrix (map positive/negative correlations to red/blue with intensity indicating correlation strength)
3. $5 \times 5$ scatter plot matrix (choose attributes with greatest aggregated correlation strength, see next slide)
4. parallel coordinates display with 10 axes (choose pairs by correlation strength, see next slide)
5. PCA plot (top 2 eigenvectors) with associated scree plot (10 bars)
6. biplot with 10 projected axes (project all into top 2 PCA vectors)
7. MDS display of the data (use Euclidian distance)
8. MDS display of the attributes (use 1-$|\text{correlation}|$ distance)
Correlation matrix
- the colors should look like this

Scatterplot matrix plot selection
- add $|\text{correlation}|$ along each correlation matrix column
- pick the 5 attributes with the highest sums and display

Parallel coordinates display axes ordering scheme
- pick pair with greatest $|\text{correlation}| \rightarrow$ axes A1, A2
- axis A1 is the attribute with highest correlation sum
- axis A3 is the attribute that has the highest $|\text{correlation}|$ with A2
- axis A4 is the attribute that has the highest $|\text{correlation}|$ with A3
- and so on....
Scree plot
  - use the bar charts you already have

MDS plots
  - should look like this
  - we will add cluster information in lab 5
Submit by Thursday, November 6, 11:59 pm (no extensions!)

- **report** discussing pros and cons for each of the eight displays
- what information do these displays show well
- what information can’t they show

- **video** that shows all capabilities of your interface

- **archive file** (zip, rar, tar) of your code and data

Point decomposition (the two w’s of lab 3 execution)

- 8 points – works (does the job)
- 2 points – wow (does the job nicely)
Survey

How difficult was lab 2?

How difficult do you think lab 3 is?

Start early!

That’s why you should