CSE 332
Introduction to Visualization

Dimension Reduction

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, and basic tasks</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Data preparation and representation</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Data reduction, notion of similarity and distance</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td><strong>Dimension reduction</strong></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Introduction to D3, basic vis techniques for non-spatial data</td>
<td>Project 2 out</td>
</tr>
<tr>
<td>7</td>
<td>Visual perception, cognition, design and aesthetics</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Statistics foundations</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Data mining techniques: clusters, text, patterns, classifiers</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Data mining techniques: clusters, text, patterns, classifiers</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>High-dimensional data, dimensionality reduction</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Computer graphics and volume rendering</td>
<td>Project 3 out</td>
</tr>
<tr>
<td>13</td>
<td>Techniques to visualize spatial (3D) data</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Scientific and medical visualization</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Scientific and medical visualization</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Non-photorealistic rendering</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Midterm</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>Principles of interaction</td>
<td>Project 4 out</td>
</tr>
<tr>
<td>19</td>
<td>Visual analytics and the visual sense making process</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>Correlation and causal modeling</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>Big data: data reduction, summarization</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Visualization of graphs and hierarchies</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>Visualization of text data</td>
<td>Project 5 out</td>
</tr>
<tr>
<td>24</td>
<td>Visualization of time-varying and time-series data</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Memorable visualizations, visual embellishments</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>Evaluation and user studies</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>Narrative visualization and storytelling</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>Data journalism</td>
<td></td>
</tr>
</tbody>
</table>
List two attributes that would differentiate the pants well:

- Color
- Length
Last Lecture’s Theme

Data Reduction
Dimension Reduction
Measure of Attribute Similarity

Are there attributes that “go together”?

Can you name a few?
Physical attributes

- color
- number of doors
- number of wheels
- retractable roof
- height
- length
- frames around side windows

Which attributes are useful to distinguish SUVs from convertibles?

- number of doors (4 vs. 2) --> numerical, two levels
- retractable roof (no vs. yes) --> categorical, two levels
- frames around side windows (yes vs. no) --> categorical, two levels
- height (higher vs. lower) --> numerical, many levels
Which attributes are not so useful?

- number of wheels (constant 4) --> no discriminative power
- length (short and long SUVs, convertibles) --> confounding
- color (colors are seemingly random, or are they?)

Is color useful?

- the convertibles seem to have more vibrant colors (red, yellow, ...)
- so maybe we made a discovery
Need to consider more than two attributes

- *height* attribute would have distinguished the Range Rover from the convertibles and caused it to be an outlier.
New classes are constantly evolving over time

- this is known as *cluster evolution*
- measuring more features will increase the chance of discovery

why can empty feature spaces be interesting or useful?

new class: the convertible SUV

retractable roof
The more data (examples) the better
- increases the chances to discover the rare specimen

- but some attributes are useless
- we can cull them away
- perform attribute reduction or *dimension reduction*
We will use the method of Principal Component Analysis (PCA)
Covariance

- measures how much two random variables change together

For N variable we have $N^2$ variable pairs

- we can write them in a matrix of size $N^2 \rightarrow$ the covariance matrix
- for two variables $X_1$ and $X_2$

$$
\text{Var}[X] = \begin{bmatrix}
\text{Var}[X_1] & \text{Cov}[X_1, X_2] \\
\text{Cov}[X_2, X_1] & \text{Var}[X_2]
\end{bmatrix}
$$
Covariance $\text{cov}(X,Y)$

$$\text{COV}(X,Y) = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{n - 1}$$

Correlation $r$

- is covariance normalized by the individual variances for $X$ and $Y$

$$r_{xy} = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2 \sum_{i=1}^{n} (y_i - \bar{y})^2}}$$

mean of all data item values $x_i$ and $y_i$ for attributes $X$ and $Y$, resp.
Ultimate goal:

- find a coordinate system that can represent the variance in the data with as few axes as possible

- rank these axes by the amount of variance (blue, red)
- drop the axes that have the least variance (red)
Principal Components

1st Principal Component, $y_1$

2nd Principal Component, $y_2$
Find the principal components (factors) of a distribution

First characterize the distribution by

- covariance matrix Cov
- correlation matrix Corr
- let's call it C

- perform QR factorization or LU decomposition on that matrix to get

$$ C = Q\Lambda Q^{-1} $$

Q: matrix with Eigenvectors
\( \Lambda \): diagonal matrix with Eigenvalues \( \lambda \)

- now order the Eigenvectors in terms of their Eigenvalues \( \lambda \)
$\lambda_1, \lambda_2$ are the Eigenvalues

- encode the length (and therefore significance) of the Eigenvectors
When to use what?

- use covariance matrix when the variable scales are similar
- use correlation matrix when the variables are on different scales
- the correlation matrix *standardizes* the data
- in general they give different results, especially when the scales are different
Before PCA

![Scatter plot showing PC 1 and PC 2 before PCA](image)
After PCA

- $\lambda_1 = 9.8783 \quad \lambda_2 = 3.0308 \quad \text{Trace} = 12.9091$
- PC 1 displays ("explains") $\frac{9.8783}{12.9091} = 76.5\%$ of total variance
**Create a scree plot**

- plots a histogram of the Eigenvalues ordered by magnitude
- plots the explained variance as a curve

![Scree Plot]

**Possible threshold**
(explain 75% of data variance)

keep top 3 principal components \(\rightarrow\) reduce dimensions by a factor of \(\frac{4}{7} = 57\%\)
The Eigenvectors are combinations of the true data vectors

- may not be that meaningful to explain the data

**Solution**

- project each data vector into the top $P$ Principal Components using dot product

\[
\begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} \cdot \begin{pmatrix} b_x \\ b_y \\ b_z \end{pmatrix} = a_x b_x + a_y b_y + a_z b_z
\]

- compute the lengths of these projected vectors
- create a scree plot of these vectors
- keep the top $T$ vectors
PCA APPLIED TO FACES

Some familiar faces...
We can reconstruct each face as a linear combination of “basis” faces, or Eigenfaces [M. Turk and A. Pentland (1991)]
90% variance is captured by the first 50 eigenvectors.
Reconstruct existing faces using only 50 basis images.
We can also generate new faces by combining eigenvectors with different weights.
Some vectors:

- Height [VW, GMC, Jeep, Beetle, Audi, Porsche, Rover]
- Number of doors [VW, GMC, Jeep, Beetle, Audi, Porsche, Rover]
- Color [VW, GMC, Jeep, Beetle, Audi, Porsche, Rover]
- Number of wheels [VW, GMC, Jeep, Beetle, Audi, Porsche, Rover]

Use k-means clustering to identify clusters of similar features
Fused dataset of 50 US colleges
US News: academic rankings
College Prowler: survey on campus life attributes
Can you “hallucinate” or “invent” realistic data?

And if so, how would you go about this?
How to Hallucinate More data...
Data Augmentation

Strategy to artificially synthesize new data from existing data

- go from small data to big data
Important topic in deep learning

Common techniques are (for images)

- rotations
- translations
- zooms
- flips
- color perturbations
- crops
- add noise by *jittering*
Definition from dictionary

- act nervously
- "an anxious student who jittered at any provocation"

- small random noise about a steady signal
Generate new samples according to the data distributions

- cluster the data (outliers will form clusters as well)
- the size of each cluster represents its percentage in the population
- randomize new samples – bigger clusters get more samples
- add a small randomized value to either the mean or an existing sample
- do this for every dimension of the chosen mean or sample

sampling rate ~ bin height
augmentation rate ~ cluster size
**Project #2.1: Data Preparation**

Too many data items?
- k-means clustering followed by stratified sampling

Too many attributes?
- PCA followed by dimension projection and scree plot culling

Not enough data items or attributes?
- data synthesis using data augmentation

Use Python library scikit-learn
- lots of simple and efficient tools for data mining and data analysis
- is available on the department’s Linux server via ssh login into allv28.all.cs.stonybrook.edu (port 130)
- use putty
Next week’s class will teach you both theory & practice of D3
  - all of the background needed