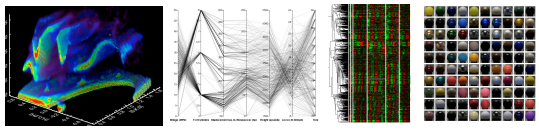
High Dimensional Data & Dimensionality Reduction

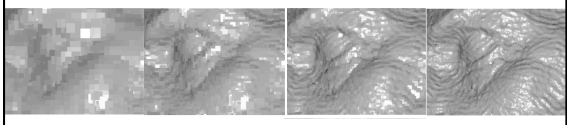


Today's Class

- Last Week's Readings
- Examples of High Dimensional Data
- Parallel Coordinates
- Principle Components Analysis (PCA)
- Next Week's Readings
- Final Project Proposal Discussion

Readings from Last Week:

- "QSplat: A Multiresolution Point Rendering System for Large Meshes", Rusinkiewicz & Levoy, SIGGRAPH 2000



Readings from Last Week:

- "Tree visualization with Tree-maps: A 2-d space-filling approach", Ben Shneiderman, 1991

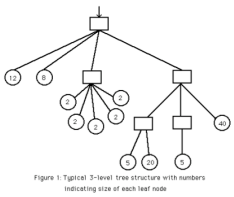


Figure 1: Typical 3-level tree structure with numbers indicating size of each leaf node

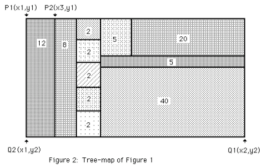


Figure 2: Tree-map of Figure 1

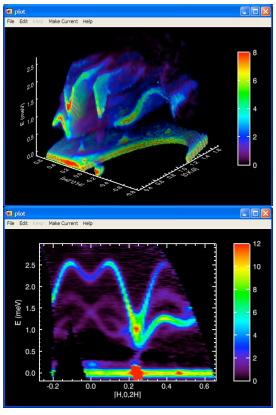
Today's Class

- Last Week's Readings
- Examples of High Dimensional Data
- Parallel Coordinates
- Principle Components Analysis (PCA)
- Next Week's Readings
- Final Project Proposal Discussion

Scientific Data

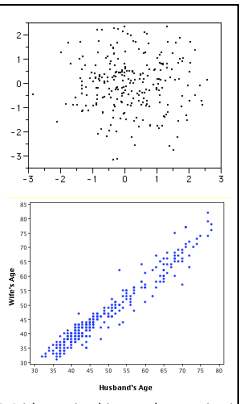
- For many 3D spatial locations during an experiment or simulation
- Time-varying temperature, velocity, pressure, humidity, etc.

http://www.ncnr.nist.gov/dave/screenshots.html



Misc. Personal Data

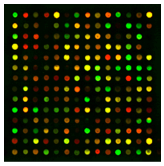
- Height, weight, eye color, phone number, address, IQ, age, cholesterol score, grade in Data Structures, etc.
- Example Hypotheses:
  - There is no correlation between height & phone number
  - There is a positive correlation between the ages of spouses
- Scatter plot: Look at 2 dimensions at a time
- Units on each axis are different!

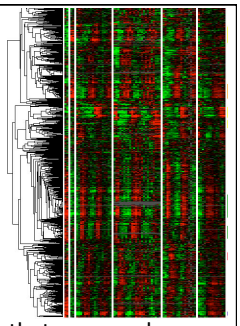


http://www.mzandee.net/~zandee/statistiek/stat-online/chapter4/pearson.html

Gene Expression

- Expression level for hundreds of genes
- For many trials (different individuals/conditions)
- Discover correlations: genes that commonly work together or in opposition





http://www.imbb.forth.gr/people/poirazi/researchEP.html

http://www.bioss.ac.uk/~dirk/essays/GeneExpression/bayes_net.html

High "Dimensional"

- Mathematicians and physicists talk about more than three spatial dimensions
- Not us
- Simply ask "How many numbers does it take to describe a data point?"
- This is the "dimensionality" of the data

Obvious "Dimensions"

- Time
  - A particle moving in 3D can be described by its location (X,Y,Z) at a particular time (t)
  - To describe it, we can specify a vector (X,Y,Z,t)
  - "4D"
- Color
  - A colored point can be specified by its location (X,Y,Z) and its color (R,G,B)
  - To describe it, we can specify a vector (X,Y,Z,R,G,B)
  - "6D"

Really High Dimensions

- "Feature Vectors"
  - Compute properties at each point in a data set and store them all in a long vector associated with each point
- Examples
  - SIFT descriptors (128-D)

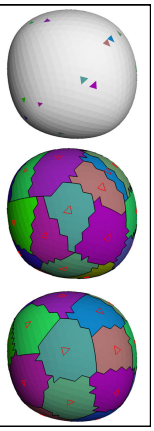
Problems

- Computational
  - Nearest neighbor searches are very expensive
  - People have developed "approximate nearest neighbor" algorithms
- Visualization
  - How can we display/see more than colored 3D points?

"Slicing" the Data

- Simply throw away most of the dimensions and keep some that you can visualize
- This is equivalent to projecting the data onto an axis aligned hyperplane
- "Important" qualities of the data may not be visible

K-Means Clustering



For a set of 2D/3D/*n*D points:

1. Choose k, how many clusters you want (oracle)
2. Select k points from your data at random as initial team representative
3. Every other point determines which team representative it is closest to and joins that team
4. The team averages the positions of all members, this is the team's new representative
5. Repeat 3-5 until change < threshold

"Advanced" Debugging

Applies to software development, and other sciences too!

- Debugging Level 1:
  - Remove syntax errors in compilation
- Debugging Level 2:
  - Produces an answer
- Debugging Level 3:
  - Matches the output provided by the instructor
- Debugging Level 4:
  - Hypothesize system behavior
  - Develop & run experiments
  - Collect data & analyze results
  - Validate (or repeat process)

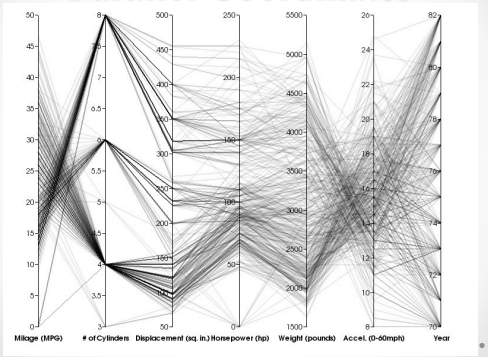
Today's Class

- Last Week's Readings
- Examples of High Dimensional Data
- Parallel Coordinates
- Principle Components Analysis (PCA)
- Next Week's Readings
- Final Project Proposal Discussion

Parallel Coordinates

- A visualization technique used to plot high dimensional data
- Each of the dimensions corresponds to a vertical axis
- Each data "point" is displayed as a series of lines.

Parallel Coordinates



Designing Visualizations using Parallel Coordinates

- How many dimensions (vertical axes)?
- In what order should the axes appear?
- Which direction should each axis run (up or down?)
- How many data points (lines)?
- How could color, line thickness, etc. be used to highlight patterns in the data?
- Data exploration or debugging tool? (Iterate)
- Or final visualization?

Parallel Coordinates in VTK

```
// Set up a 2D scene, add an XY chart to it
vtkContextView* view =
  vtkContextView::New();

vtkChartParallelCoordinates* chart =
  vtkChartParallelCoordinates::New();

view->GetScene()->AddItem(chart);

vtkTable* table =
  vtkTable::New();

vtkFloatArray* array1 =
  vtkFloatArray::New();
array1->SetName("Field 1");
table->AddColumn(array1);

.. Repeat for each dimension/axis
```

Parallel Coordinates in VTK

```
// Generate 4D data points [i, cos(i), sin(i), tan(i)]
int numPoints = 200;
table->SetNumberOfRows(numPoints);
for (int i = 0; i < numPoints; ++i)
{
table->SetValue(i, 0, i); table->SetValue(i, 1, cos(i)); table->SetValue(i, 2,
sin(i)); table->SetValue(i, 3, tan(i));
}

chart->GetPlot(0)->SetInput(table);
view->GetRenderWindow()->SetMultiSamples(0);
view->GetInteractor()->Initialize(); view->GetInteractor()->Start();
```

Parallel Coordinates

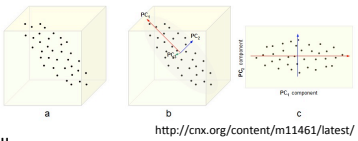
- Interaction really helps
  - Highlighting
  - Filtering
  - Roll-over detail

Today's Class

- Last Week's Readings
- Examples of High Dimensional Data
- Parallel Coordinates
- Principle Components Analysis (PCA)
- Next Week's Readings
- Final Project Proposal Discussion

Principle Components Analysis (PCA)

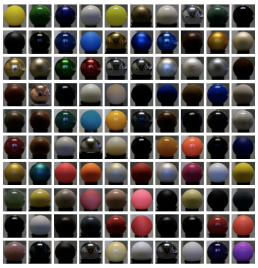
- Takes high dimensional data, where some/many axes are correlated



http://cnx.org/content/m11461/latest/

- Reduce to a smaller set of dimensions that are *not correlated*
- Dimensions/axes form a new basis/coordinate system
  - Each example from the original data can be defined as a linear combination of the new axes
- Essentially we want to find the internal structure that best explains the variance in the data

PCA Example: Material Reflectance Model



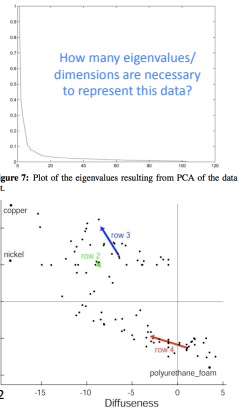
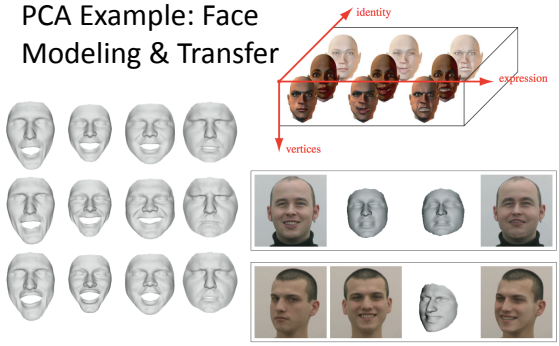


Figure 7: Plot of the eigenvalues resulting from PCA of the data set.

Matusik, Pfister, Brand, & McMillan,
"A Data-Driven Reflectance Model" SIGGRAPH 2002

PCA Example: Face Modeling & Transfer

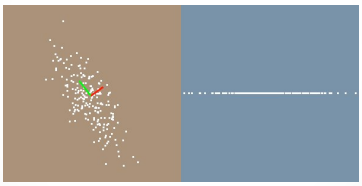


"Face Transfer with Multilinear Models", Vlasic, Brand, Pfister, & Popovic, SIGGRAPH 2005

Principal Component Analysis (PCA)

- Find the "important" "directions" in the data
- "If we could fix the data and rotate the coordinate axes before slicing, how would we rotate them?"
- Use PCA
  - Eigenvectors of the scatter matrix…
  - Not important, just need to know what it does

Principal Component Analysis (PCA)



PCA in VTK

```
// Create the data

vtkDoubleArray* xArray =
  vtkDoubleArray::New();
xArray->SetNumberOfComponents(1);
xArray->SetName("x");

vtkDoubleArray* yArray =
  vtkDoubleArray::New();
yArray->SetNumberOfComponents(1);
yArray->SetName("y");

for(vtkIdType i = 0; i < polydata->GetNumberOfPoints(); i++)
{
double p[3];
polydata->GetPoint(i,p);
xArray->InsertNextValue(p[0]);
yArray->InsertNextValue(p[1]);
}
```

PCA in VTK

```
// Load the data into a table
vtkTable* datasetTable =
  vtkTable::New();
datasetTable->AddColumn(xArray);
datasetTable->AddColumn(yArray);

vtkPCAStatistics* pcaStatistics =
  vtkPCAStatistics::New();

pcaStatistics->SetInput( vtkStatisticsAlgorithm::INPUT_DATA, datasetTable );

pcaStatistics->SetColumnStatus("x", 1 ); pcaStatistics-
>SetColumnStatus("y", 1 );
pcaStatistics->RequestSelectedColumns(); pcaStatistics-
>SetDeriveOption(true);
pcaStatistics->Update();
```

PCA in VTK

```
// Get eigenvalues
vtkSmartPointer<vtkDoubleArray> eigenvalues =
vtkSmartPointer<vtkDoubleArray>::New(); pcaStatistics-
>GetEigenvalues(eigenvalues);
for(vtkIdType i = 0; i < eigenvalues->GetNumberOfTuples(); i++)
{
std::cout << "Eigenvalue " << i << " = " << eigenvalues->GetValue(i) <<
std::endl;
}
```

PCA in VTK

```
// Get eigenvectors
vtkDoubleArray* eigenvectors =
  vtkDoubleArray::New();
pcaStatistics->GetEigenvectors(eigenvectors);
vtkDoubleArray* evec1 =
  vtkDoubleArray::New();
pcaStatistics->GetEigenvector(0, evec1);

std::cout << “evec1: “ << evec1->GetValue(0) << “ “ << evec1->GetValue(1);

vtkDoubleArray* evec2 =
  vtkDoubleArray::New();
pcaStatistics->GetEigenvector(1, evec2);
std::cout << “evec2: “ << evec2->GetValue(0) << “ “ << evec2->GetValue(1);
```

Today’s Class

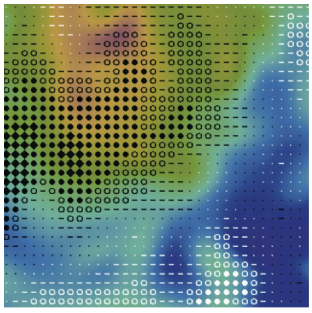
- Last Week’s Readings
- Examples of High Dimensional Data
- Parallel Coordinates
- Principle Components Analysis (PCA)
- Next Week’s Readings
- Final Project Proposal Discussion

Readings for Next Week:

- Ben Schneiderman, “The eyes have it: A task by data type taxonomy for information visualization”, *Visual Languages, 1996*
- *Visual Information-Seeking Mantra:*
  - *overview first*
  - *zoom and filter*
  - *then details on demand*

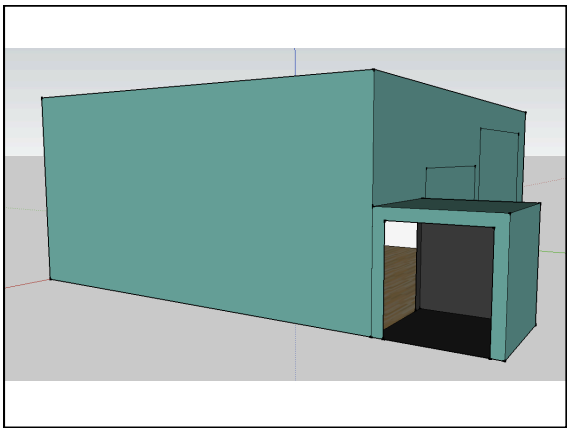
Readings for Next Week:

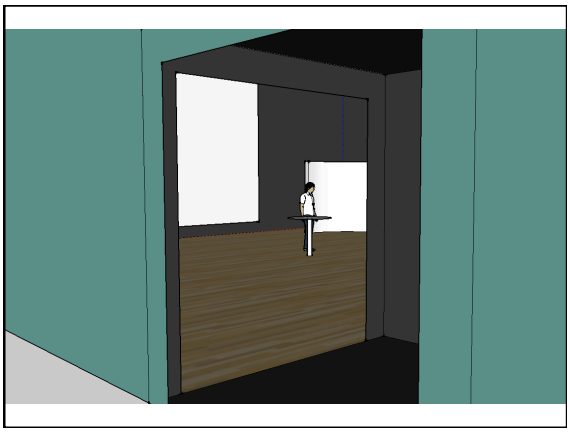
- Colin Ware, "Quantitative Texton Sequences for Legible Bivariate Maps," *IEEE Transactions on Visualization and Computer Graphics*, 2009.

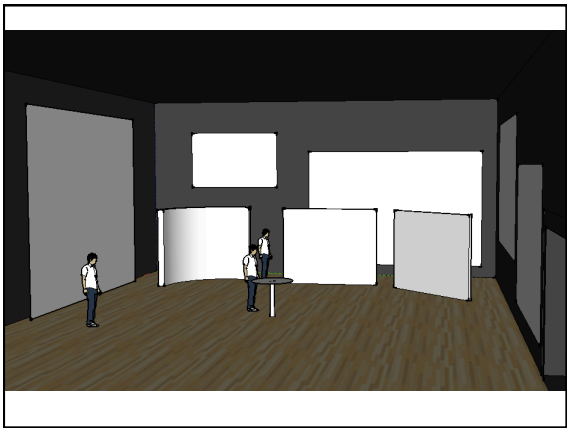


Today’s Class

- Last Week’s Readings
- Examples of High Dimensional Data
- Parallel Coordinates
- Principle Components Analysis (PCA)
- Next Week’s Readings
- Final Project Proposal Discussion

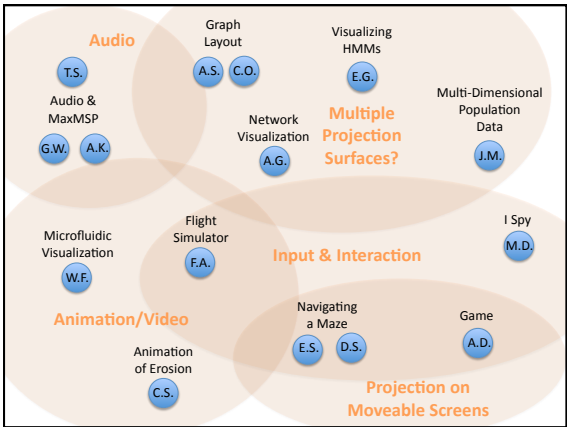






Studio 2 Setup?

- Multiple projection surfaces
  - Fixed & mobile
  - Varied size/purpose/resolution
- Visualization Interaction
  - Central control station
  - Individual input: iClicker, smart phone, etc.



Final Project Guidelines

- Read & summarize 2-3 papers related to your project & incorporate/extend components of this work
- Save early iterations of the visualization (and any "bloopers"), to show the progression of your visualization design and data exploration
- Tues Nov 16th *or* Fri Nov 19th 10-11:50am? pre-review of final projects in *Visual Design* class