CSE 564
Visualization & Visual Analytics

High-D Space and Dimension Reduction

Klaus Mueller

Computer Science Department
Stony Brook University

Lecture	Topic	Projects
1	Intro and logistics	
2	Basic visualizations and tasks, data types, examples, ethical considerations	
3	Data preparation (cleaning, imputation, data set integration)	
4	AI-assisted coding for VIS applications (design, debugging, refactoring)	Project #1 out
5	Big data and data reduction (distance/sim metrics, intro to clustering)	
6	High-D data: concept, subspaces, dimension reduction, PCA	
7	Cluster analysis: numerical data, categorical data	
8	Perception and cognition (human visual system, color, contrast, bias)	Project #2(a) out
9	Visual design and aesthetics	
10	Visualization of multivariate and high-dimensional data: direct methods	
11	Visualization of multivariate and high-D data: projections & embeddings	
12	Visualization and AI: mutual support and capabilities (VIS4AI, AI4VIS)	Project #2(b) out
13	Principles of interaction: drive what is visualized, analyzed & how (HCI4VIS)	
14	Visual analytics (VA), human-centered AI, mixed-initiative system	
15	Midterm #1 (tentative date)	
16	VA system design and evaluation, collaborative VA, uncertainty, provenance	
17	Midterm #1 discussion (tentative date)	Final proj. proposal call out
18	Visualization of hierarchical data	
19	Visualization of maps and data with geo-reference	
20	Visualization of graphs, networks (incl. derivation of causal networks)	Final project proposal due
21	Vis. of time-varying, time-series, streaming data, progressive visualization	
22	Visualization of text, LLMs, and semantic data	
23	Ed Tufte revisited: principles, critiques and limits, responsible visualization	
24	Design of effective infographics	Final proj. prelim report due
25	Foundations scientific and medical visualization, intro to volume rendering	
26	Scientific visualization	Bonus project out (Vol Ren)
27	Story telling with data, data journalism	
28	Midterm #2 (tentative date)	
Final	Final project demo on zoom (public)	All final proj. materials due

High-Dimensional Data

Rectangular Dataset

One data item

The variables or *features*
→ the attributes or properties we measured

The data items or *feature vectors*
→ the samples (observations) we obtained from the population of all instances

	A	B	C	D	E	F	G	H	I
1	Name	Country	Miles Per Gallon	Accceleration	Horsepower	weight	cylinders	year	price
2	Volkswagen Rabbit Dl	Germany	43,1	21,5	48	1985	4	78	2400
3	Ford Fiesta	Germany	36,1	14,4	66	1800	4	78	1900
4	Mazda GLC Deluxe	Japan	32,8	19,4	52	1985	4	78	2200
5	Datsun B210 GX	Japan	39,4	18,6	70	2070	4	78	2725
6	Honda Civic CVCC	Japan	36,1	16,4	60	1800	4	78	2250
7	Oldsmobile Cutlass	USA	19,9	15,5	110	3365	8	78	3300
8	Dodge Diplomat	USA	19,4	13,2	140	3735	8	78	3125
9	Mercury Monarch	USA	20,2	12,8	139	3570	8	78	2850
10	Pontiac Phoenix	USA	19,2	19,2	105	3535	6	78	2800
11	Chevrolet Malibu	USA	20,5	18,2	95	3155	6	78	3275
12	Ford Fairmont A	USA	20,2	15,8	85	2965	6	78	2375
13	Ford Fairmont M	USA	25,1	15,4	88	2720	4	78	2275
14	Plymouth Volare	USA	20,5	17,2	100	3430	6	78	2700
15	AMC Concord	USA	19,4	17,2	90	3210	6	78	2300
16	Buick Century	USA	20,6	15,8	105	3380	6	78	3300
17	Mercury Zephyr	USA	20,8	16,7	85	3070	6	78	2425
18	Dodge Aspen	USA	18,6	18,7	110	3620	6	78	2700
19	AMC Concord D1	USA	18,1	15,1	120	3410	6	78	2425
20	Chevrolet MonteCarlo	USA	19,2	13,2	145	3425	8	78	3900
21	Buick RegalTurbo	USA	17,7	13,4	165	3445	6	78	4400
22	Ford Futura	Germany	18,1	11,2	139	3205	8	78	2525
23	Dodge Magnum XE	USA	17,5	13,7	140	4080	8	78	3000
24	Chevrolet Chevette	USA	30	16,5	68	2155	4	78	2100

Understanding High-D Objects

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)

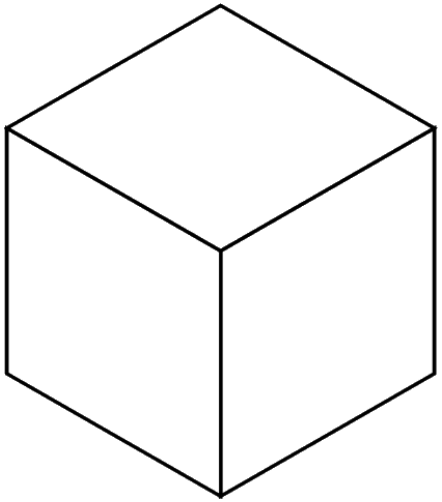
- the ability to create mental pictures or remember objects and people you have previously seen
- thought to be a vital precursor to creativity and abstract thinking

High-D Space is Tricky

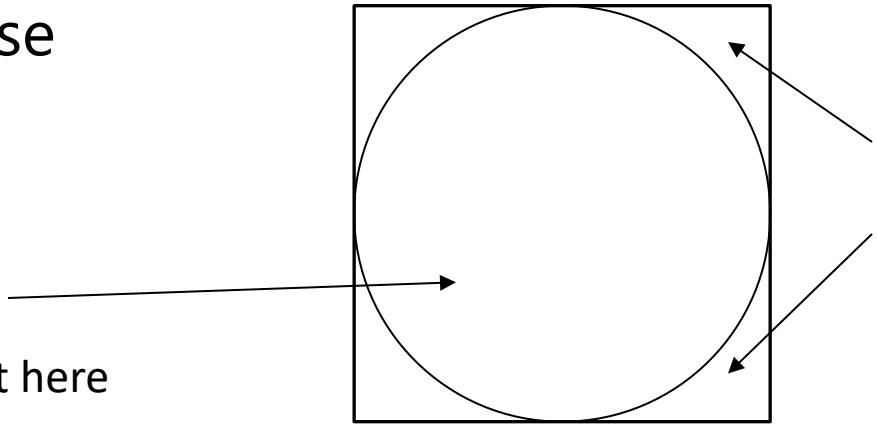
The curse of dimensionality

As $n \to \infty$

- Cube: side length l, diagonal d, volume V
- $V \to \infty$ for $l > 1$
- $V \to 0$ for $l < 1$
- $V = 1$ for $l = 1$
- $d \to \infty$

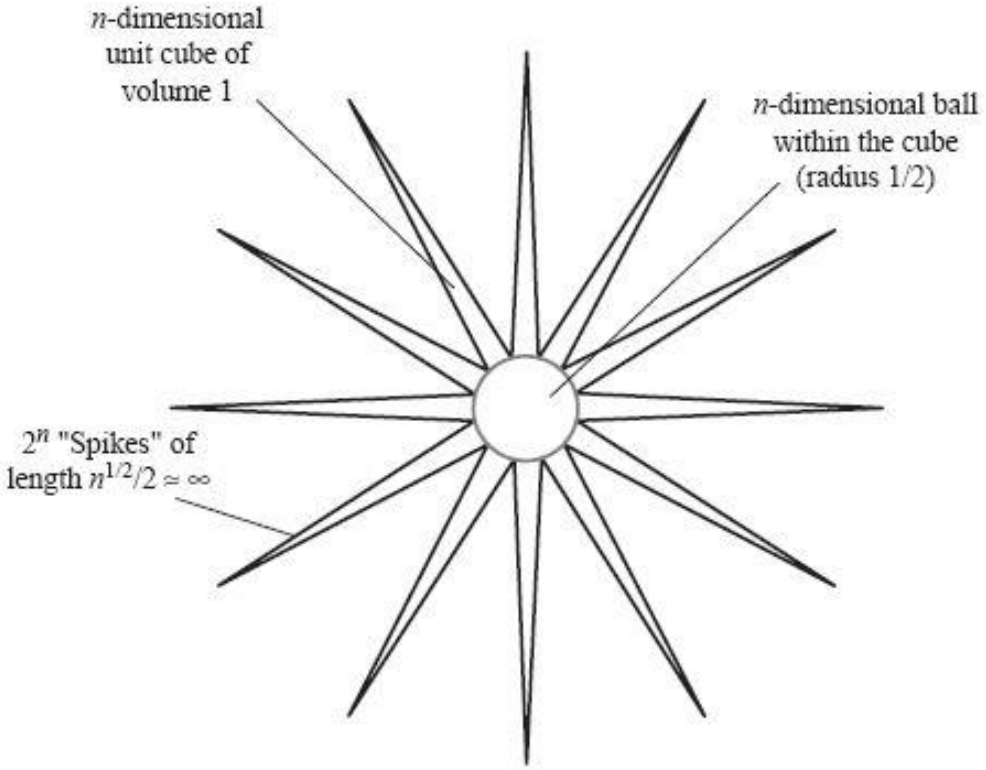


and very sparse



High-D Space is Tricky

Essentially hypercube is like a “hedgehog”



Curse of Dimensionality

Points are all at about the same distance from one another

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

$$\lim_{n\to\infty} \frac{Dist_{\max} - Dist_{\min}}{Dist_{\min}} \to 0$$

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

Sparseness Demonstration

Space gets extremely sparse

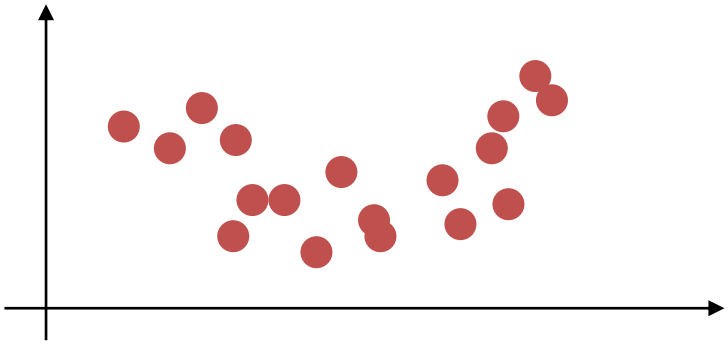
- with every extra dimension points get pulled apart further
- distances become meaningless

Sparseness Demonstration

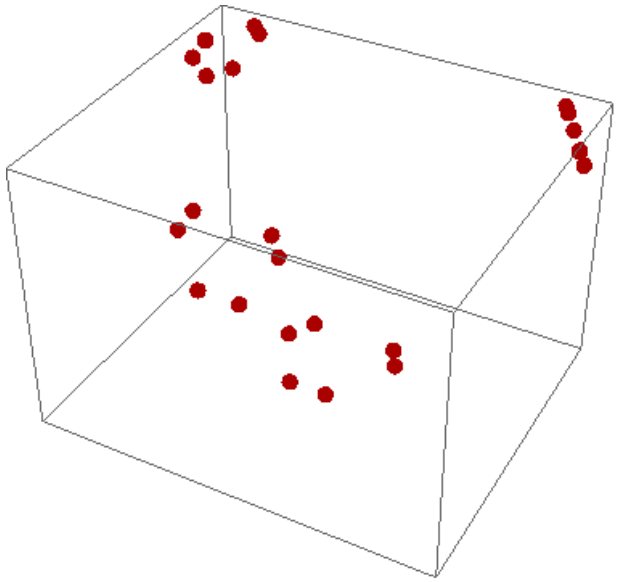
Space gets extremely sparse

- 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

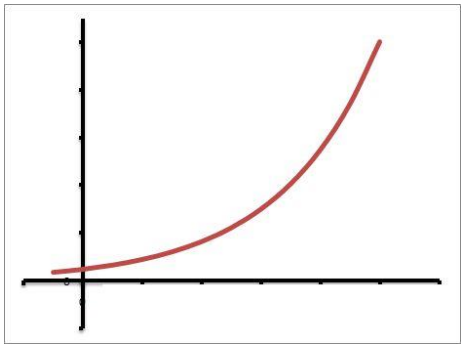
4D, 5D, … – sparseness grows further

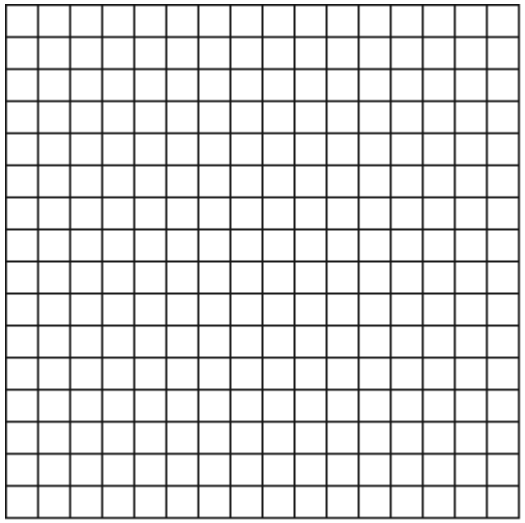
Space and Memory Management

Indexing (and storage) also gets very expensive

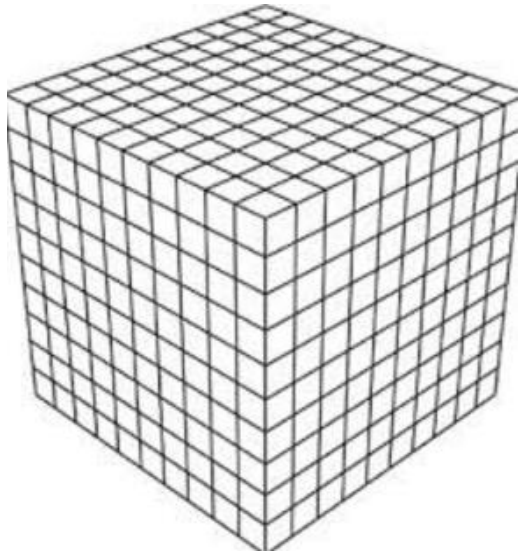
- exponential growth in the number of dimensions

16 cells





$16^2 = 256$ cells



$16^3 = 4{,}096$ cells

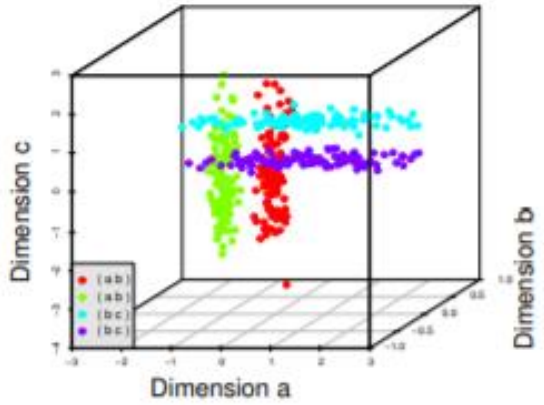
- 4D: 65k cells 5D: 1M cells 6D: 16M cells 7D: 268M cells
- keep a keen eye on storage complexity

High-D Data Subspaces

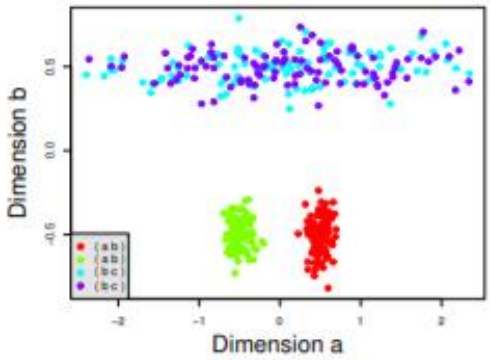
Subspace

A subspace is a projection of data into a reduced dimension set that isolates a particular cluster

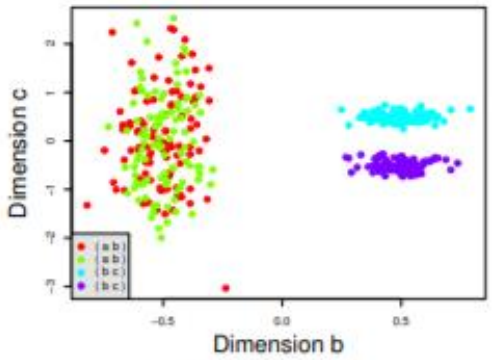
Full 3D dataset



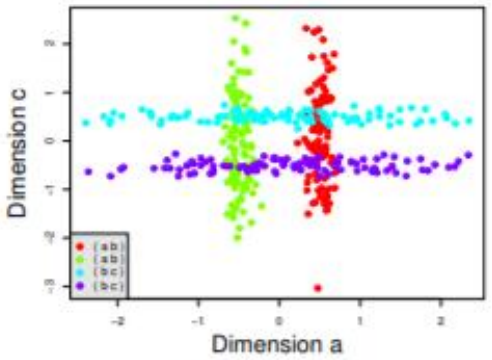
Its three 2D subspaces



(a) Dims a & b



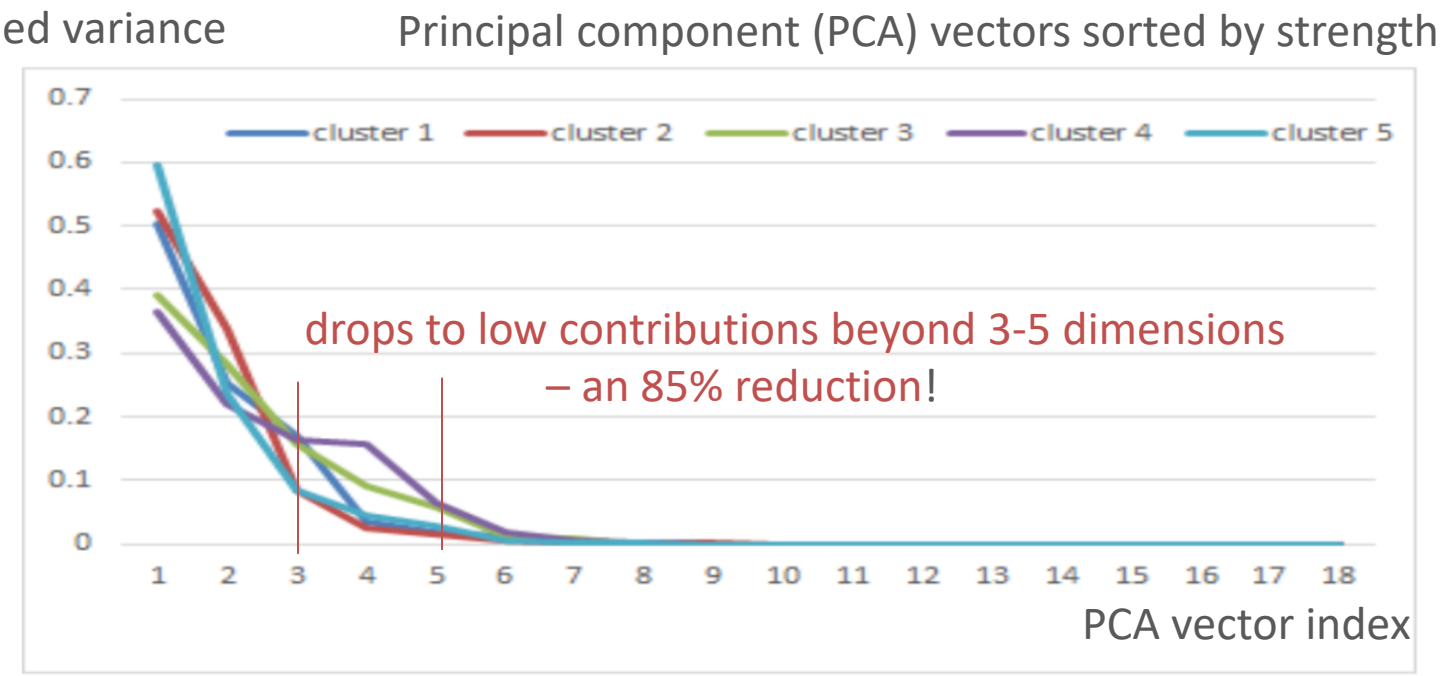
(b) Dims b & c



(c) Dims a & c

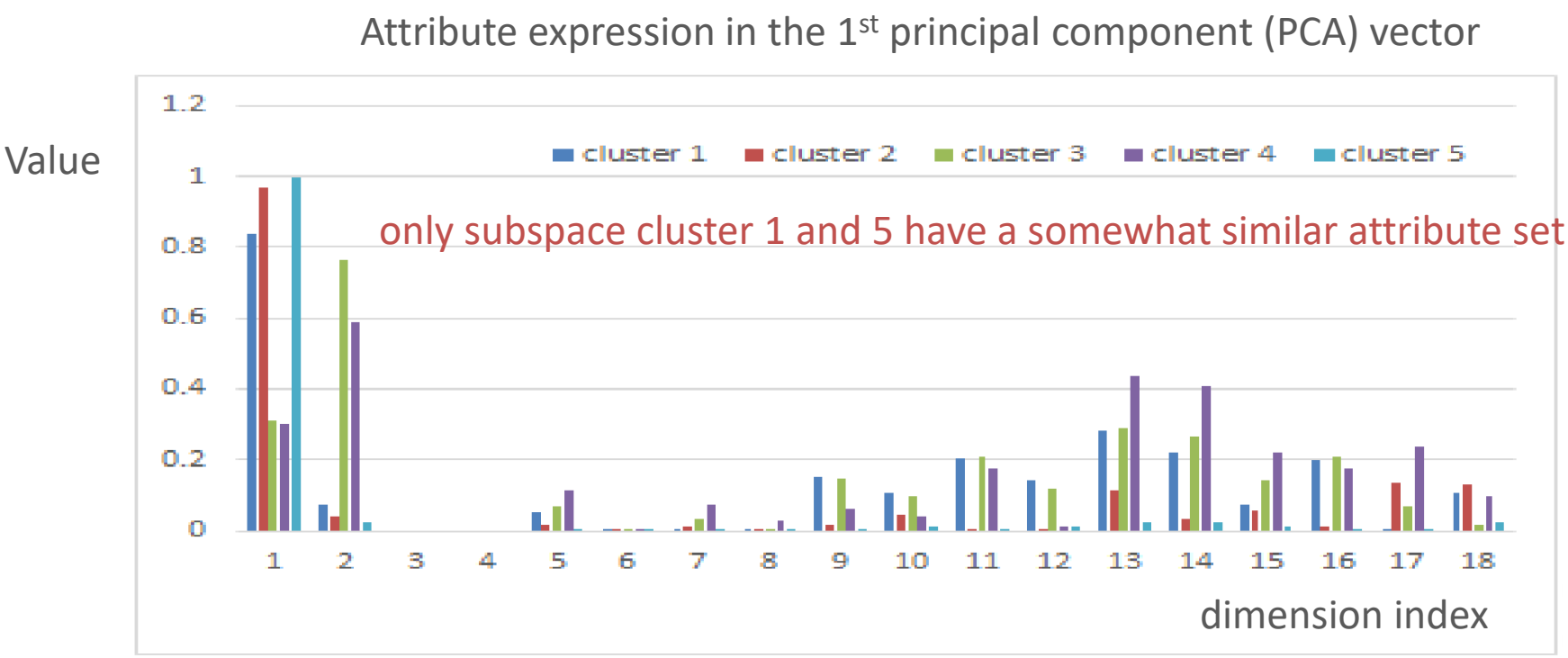
Intrinsic Dimensionality of Subspaces

Example: Image segmentation dataset with 19 dimension



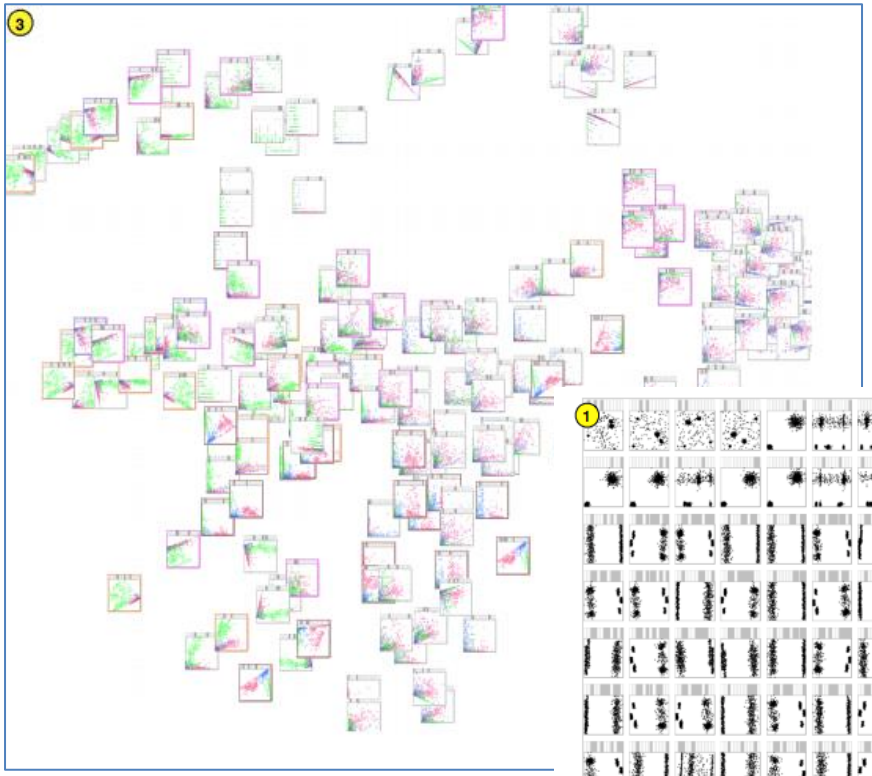
From: B. Wang. K. Mueller, "Does 3D really make sense for visual cluster analysis? Yes!" International Workshop on 3DVis: Does 3D Really Make Sense for Data Visualization? (held jointly with VIS 2014), Paris, France, November 2014.

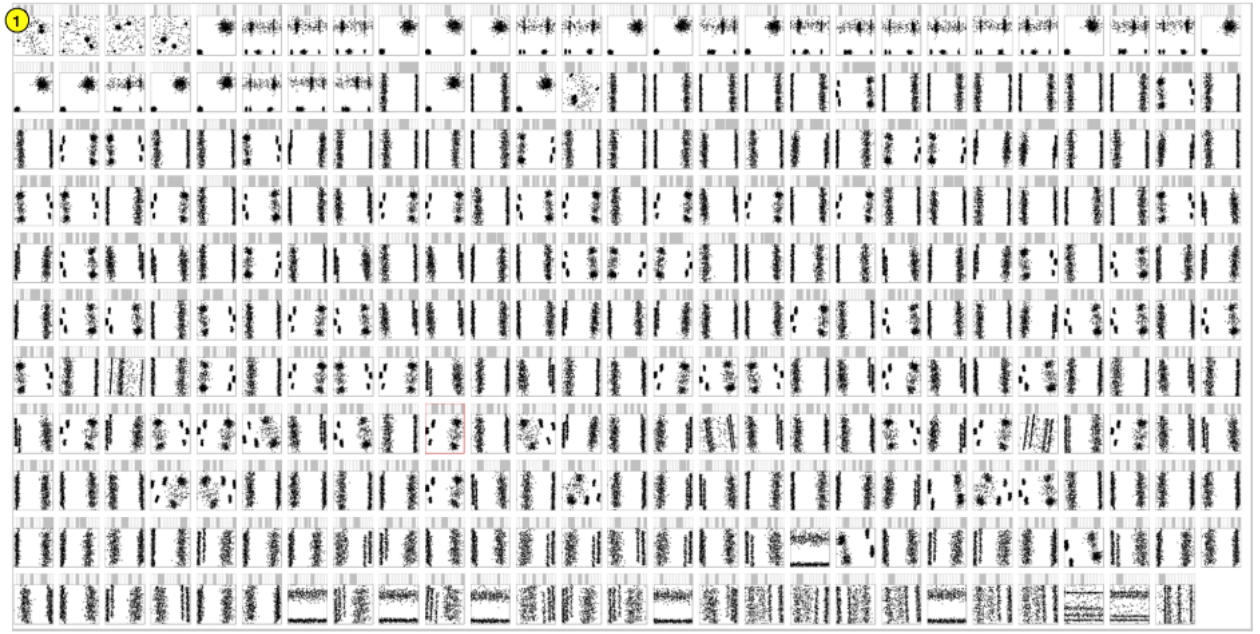
Subspaces are Quite Unique



From: B. Wang. K. Mueller, "Does 3D really make sense for visual cluster analysis? Yes!" International Workshop on 3DVis: Does 3D Really Make Sense for Data Visualization? (held jointly with VIS 2014), Paris, France, November 2014.

An Abundance of Possible Subspaces





Once there are many dimensions

- Subspace explosion

Manage it by Finding *Semantic* Subspaces

Subspaces that have a common theme

--> Semantic clusters derived from word embeddings of the data attributes

Expenditures

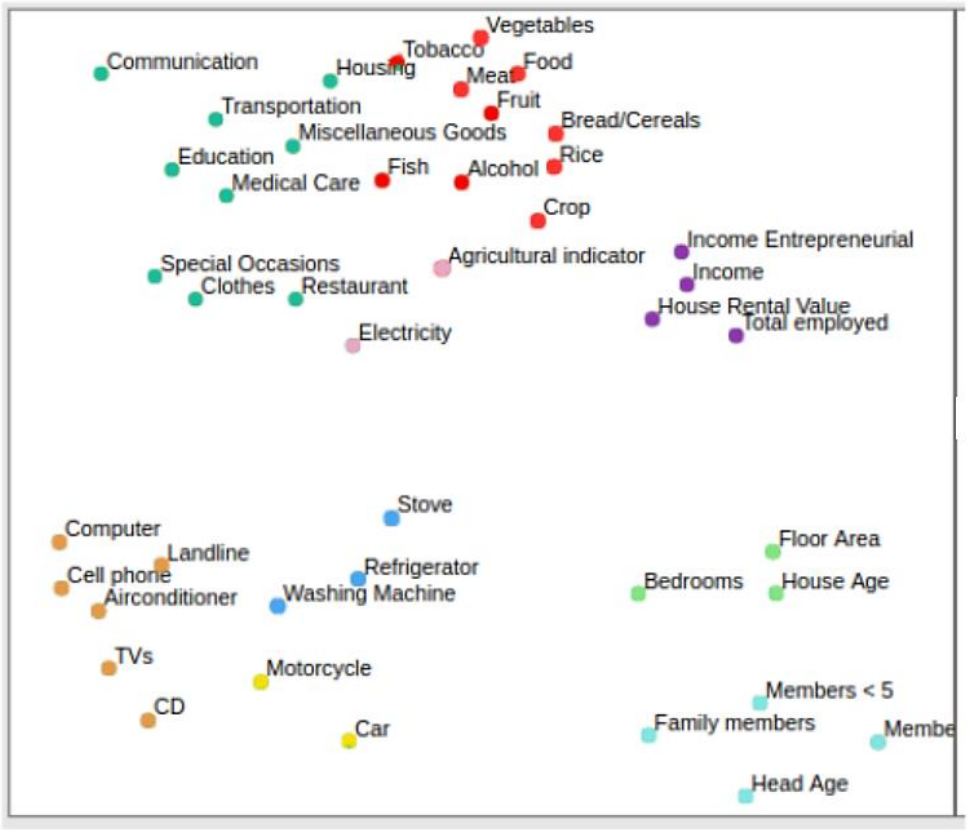
- Food
- Utilities

Ownership

- Electronics
- Transport
- Appliances

Housing

Income



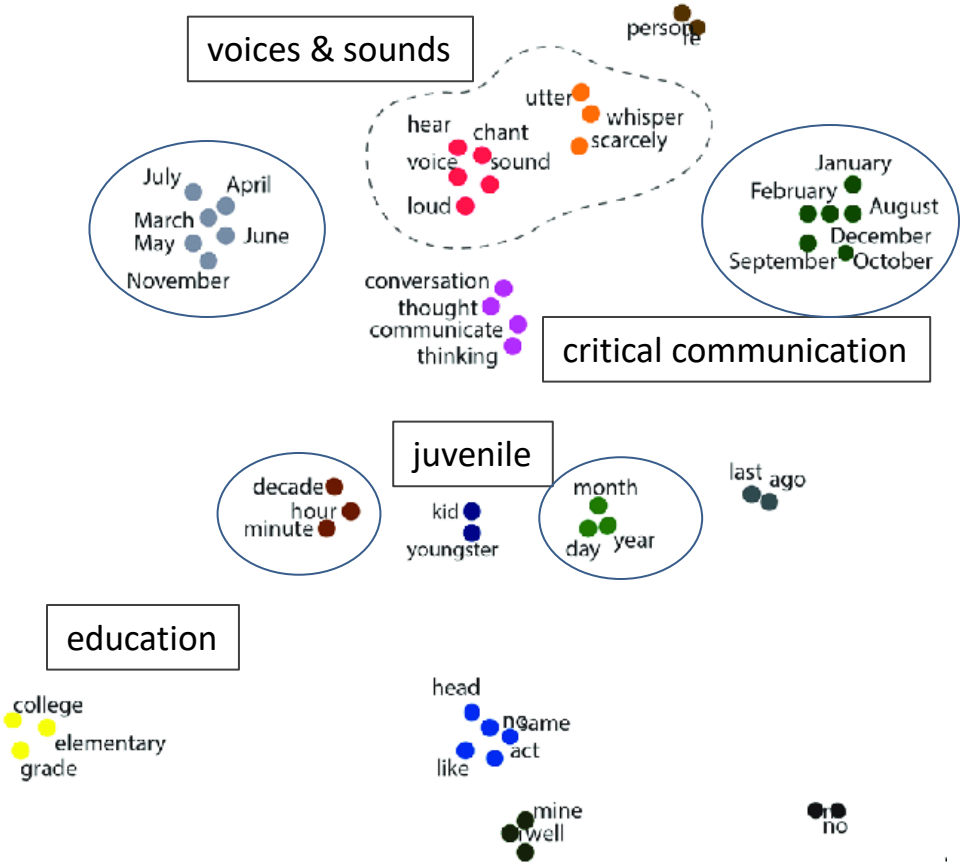
What's a Word Embedding?

A mapping where words associated with similar concepts map into close neighborhoods

Created by algorithms like

- Word2Vec (200-300-D)
- BERT (768-D, 1024-D)
- RoBERTa (768-D, 1024-D)
- Large Language Models (GPT3 has 12,288-D)

Visualized by projecting into 2D

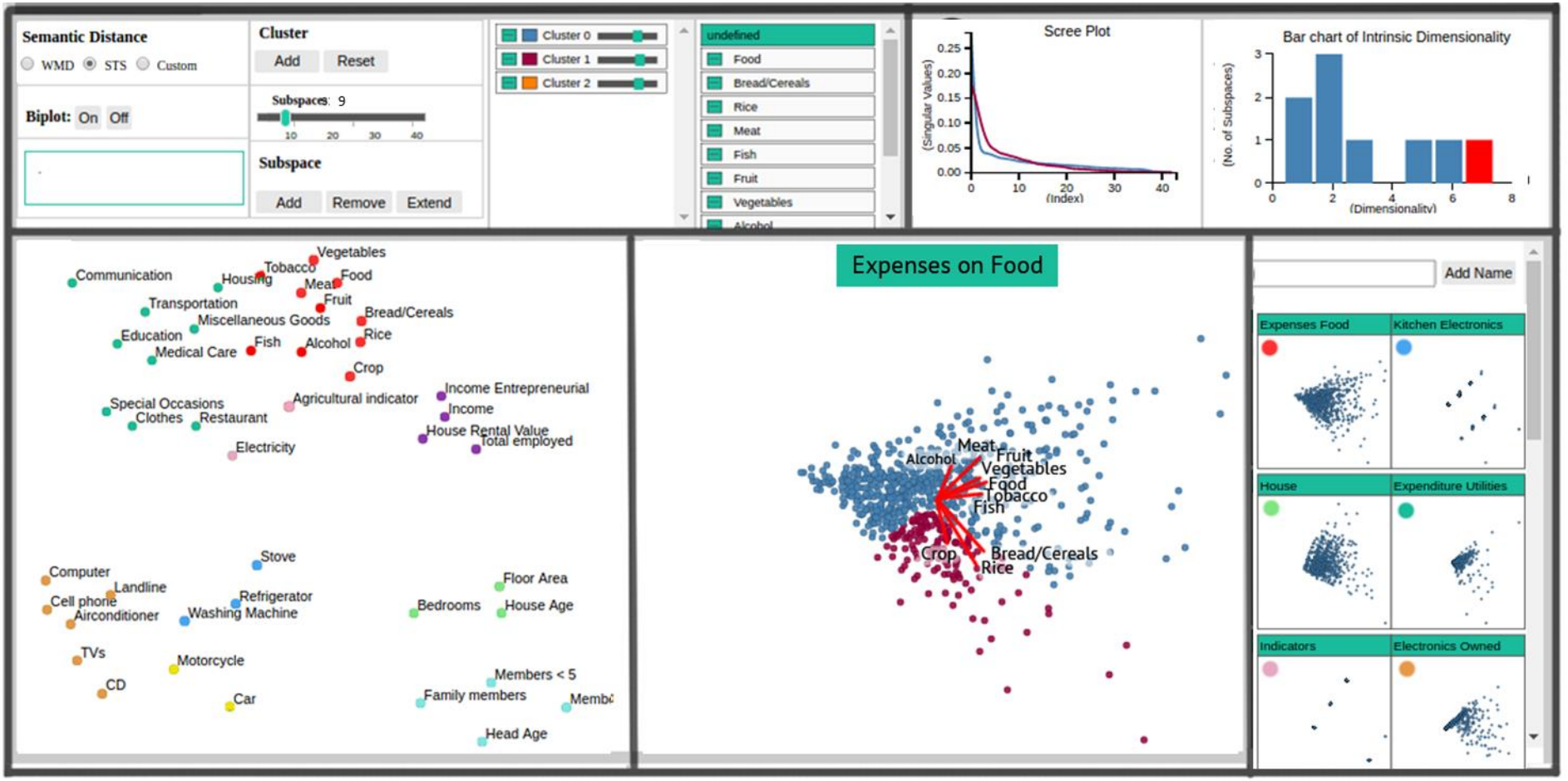


Semantic Subspace Explorer

App for the interactive exploration of semantic subspaces

Published in

- S. Mahmood, K. Mueller, "Interactive Subspace Cluster Analysis Guided by Semantic Attribute Associations," IEEE Trans. on Visualization and Computer Graphics, 30(7): 4197-4210, 2024.

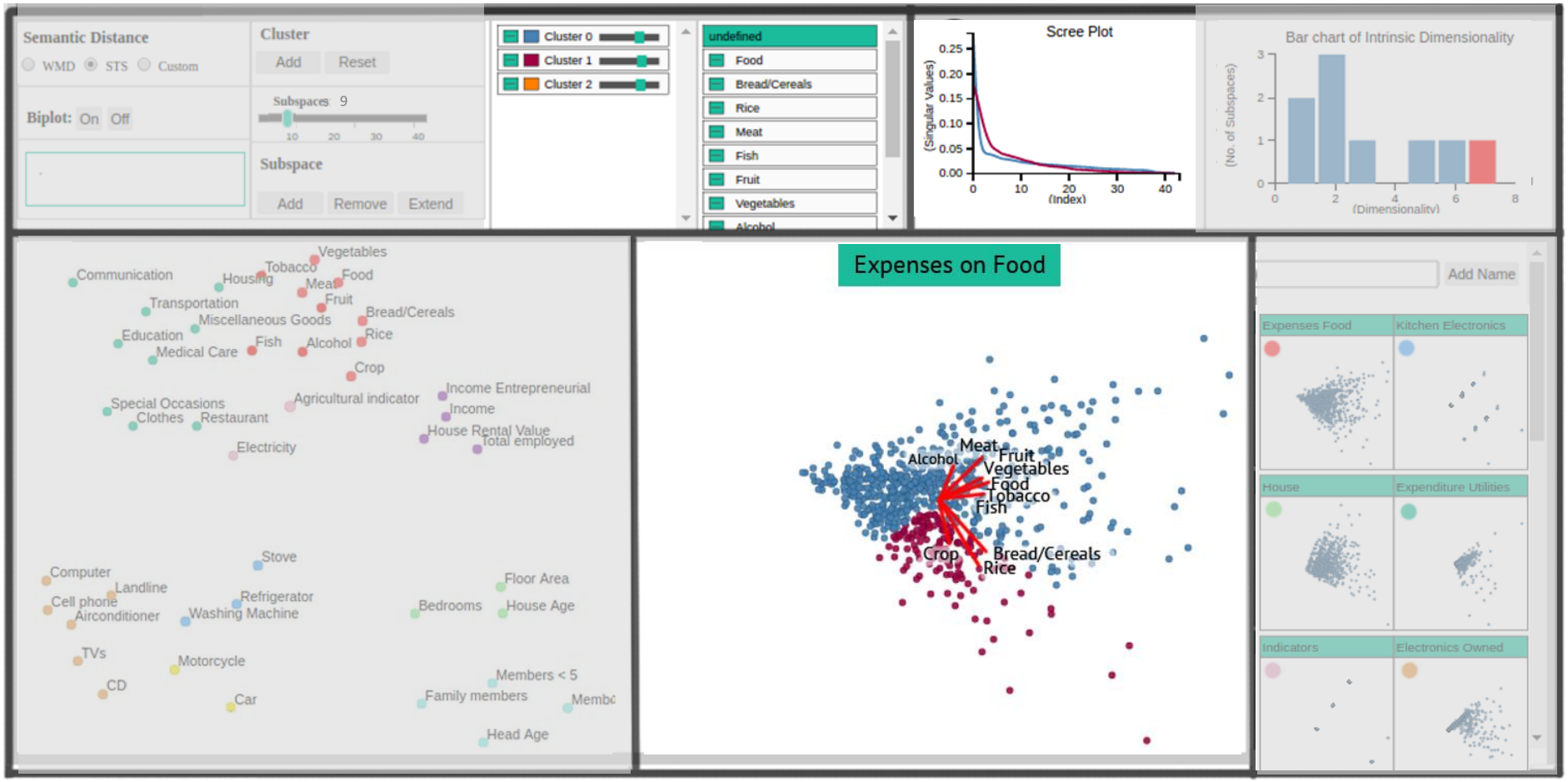


Full Interface – shown here is the Filipino Family Income & Expenditure dataset (60-D)

Select number of subspaces

Semantic space overview

Grid with labeled semantic subspaces



Subspace biplot (shows all households in the context of the attribute axes of the selected subspace)

Visualize the 'Expenses on Food Subspace' and Highlight Interesting Groups

Biplot reveals two dimension clusters

- Crop, Bread, Rice (basic staples)
- Other (more luxurious) foods

Distinguish these for further visual analysis

- Color these "Basic Staples" household points in magenta

Next, explore how what Services the "Basic Staple" households spend money on

- project data into the "Expenses on Services" subspace

Project Into The 'Expenses on Services' Subspace and Do Further Highlighting

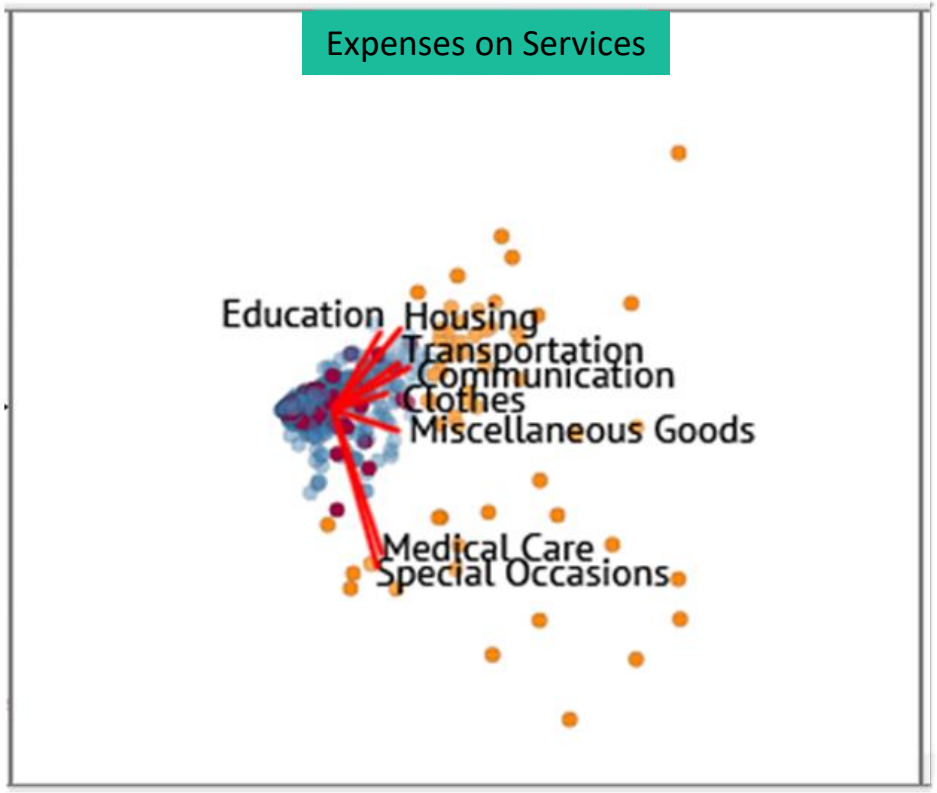
Make blue clusters less opaque to focus on the magenta "Basic Staples" households

- "Basic staples" households do not spend much on services
- Medical Care is different from other services, similar to Special Occasions

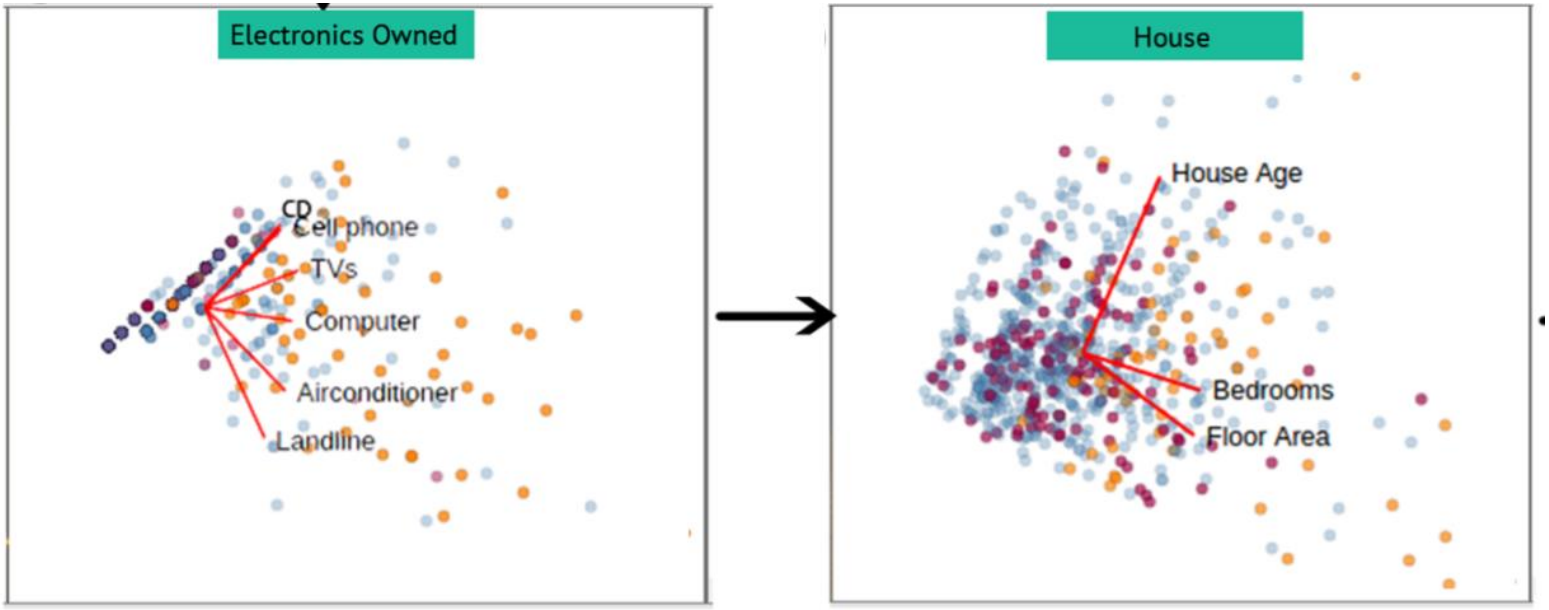
Next, lets see what Electronics they own and what kind of Houses they have

- project the brushed data into these subspaces

But first color "high spenders" in yellow



Project into Other Subspaces for Further Insight



"High spender" households own more electronics and live in larger houses
"Basic staples" households own few electronics (left) and smaller houses (right)
There is no distinction in terms of house age (right)

Make Final Conclusions

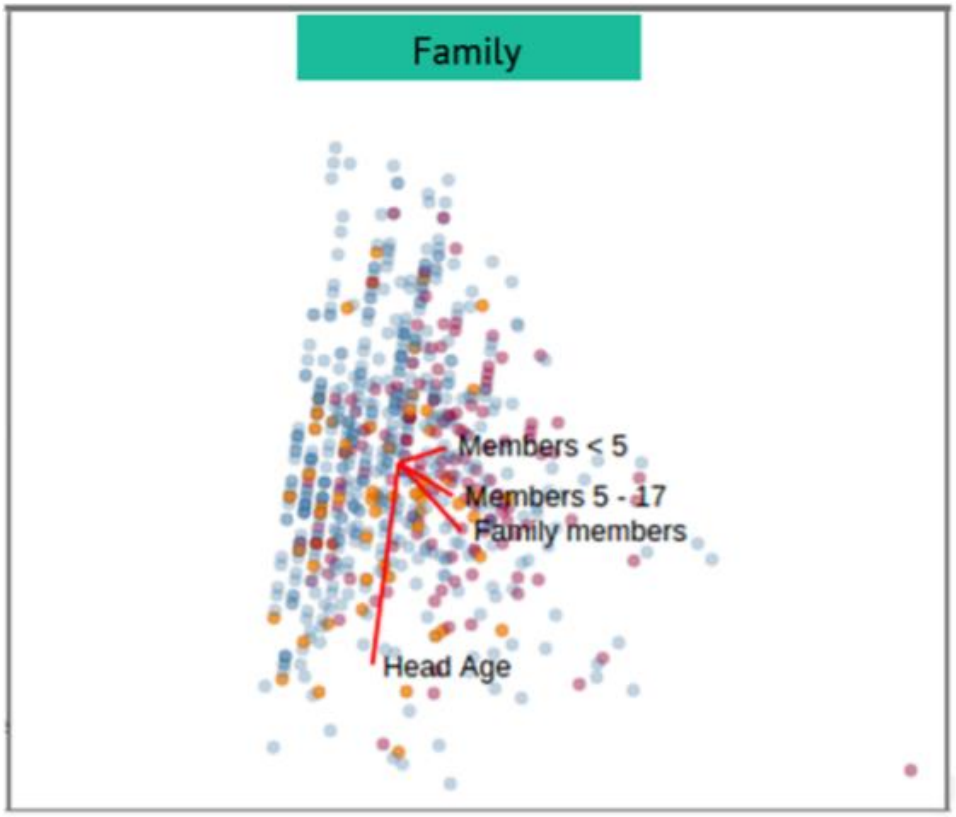
Projecting into the “Family” subspace

“Basic staples” households are more evenly distributed

“High spenders” seem to be older families

The overall finding is thus:

- In Filipino families with less economic resources, Rice, Bread/Cereal and Crops make up a major portion of the food consumption.



Next Theme

3D

2D

Dimension Reduction

Measure of Attribute Similarity

Are there attributes that "go together"?

Can you name a few?

Feature Vector (1)

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

Feature Vector (2)

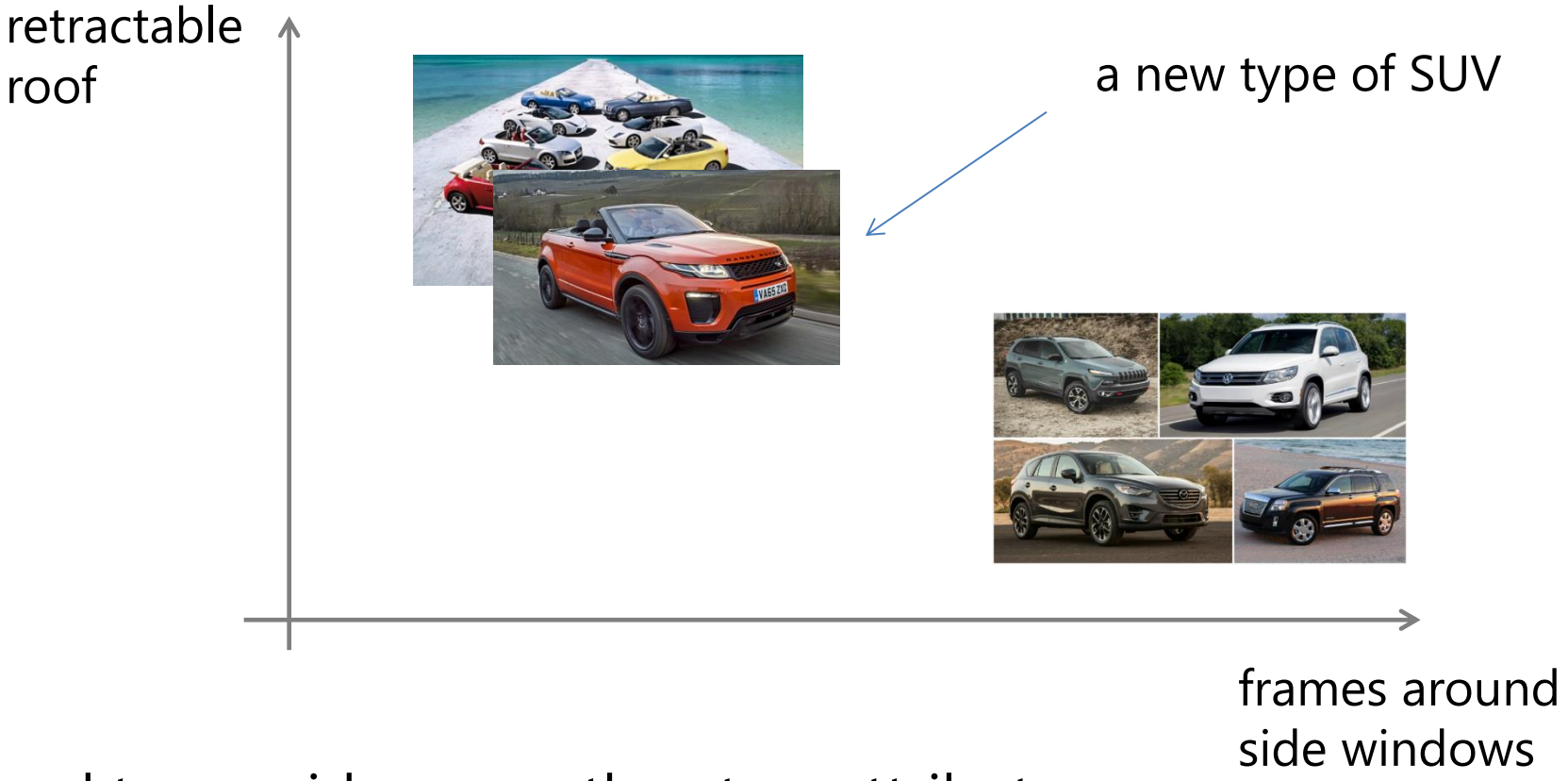
Which attributes are not so useful?

- 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

Attribute Space



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

Attribute Space

retractable roof

why can empty feature spaces be interesting or useful?

new class: the convertible SUV

height

New classes are constantly evolving over time

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

How Many Data Do We Need?

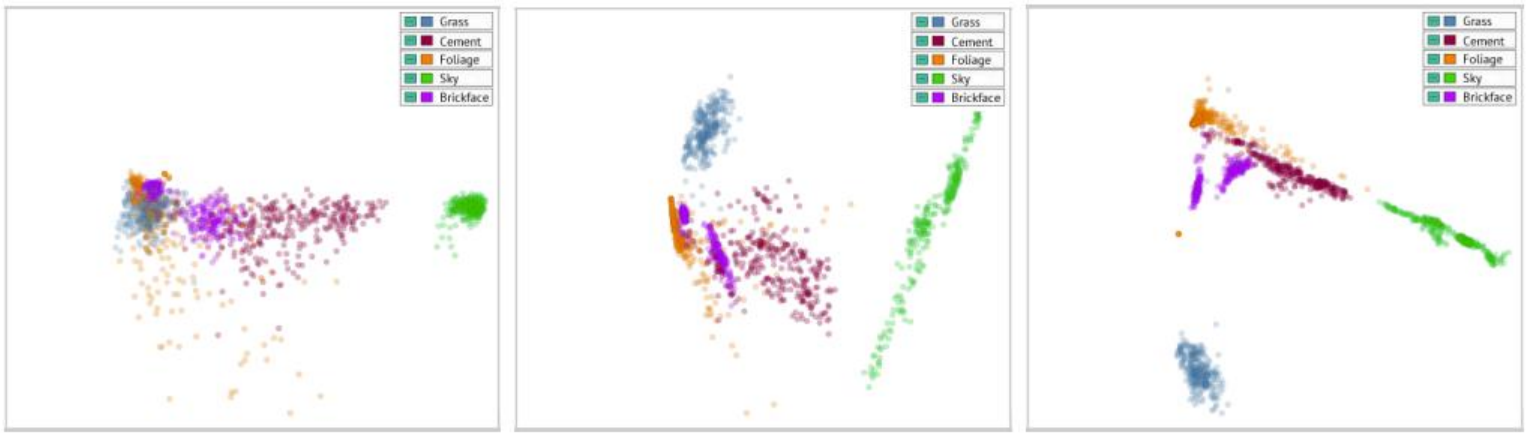
The more data (examples) the better

- increases the chances to discover the rare specimen

- but some attributes are useless
- we can cull them away
- perform attribute reduction or *dimension reduction*

How Many Attributes Do We Need?

Too many attributes can lead to obliteration of data patterns



(a) Full space (b) Subspace (c) Extended subspace

PCA projections of the Image Segmentation dataset generated from

(a) the full 16D dataspace comprised of all feature dimensions
(b) the 3D Raw Color semantic subspace
(c) the 5D extended Raw Color semantic subspace.

The points are colored by their image class

Only (b) and (c) can separate the image classes well

Dimensionality Reduction

By axis rotation

- determine a more efficient basis
- Principal Component Analysis (PCA)
- Singular value decomposition (SVD)
- Latent semantic analysis (LSA)

By type transformation

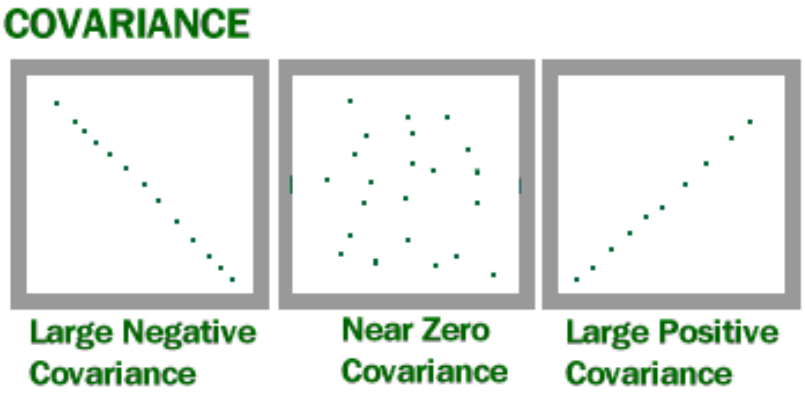
- determine a more efficient data type
- Fourier analysis and Wavelets for grids
- Multidimensional scaling (MSD) for graphs
- Locally Linear Embedding
- Isomap
- Self Organizing Maps (SOM)
- Linear Discriminant Analysis (LDA)

Principal Component Analysis (PCA)

Some Theory is Needed

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 → the *covariance matrix*
- for two variables X_1 and X_2

$$\mathrm{Var}[X] = \begin{bmatrix} \mathrm{Var}[X_1] & \mathrm{Cov}[X_1, X_2] \\ \mathrm{Cov}[X_2, X_1] & \mathrm{Var}[X_2] \end{bmatrix}$$

Formulae

Covariance cov(X,Y)

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

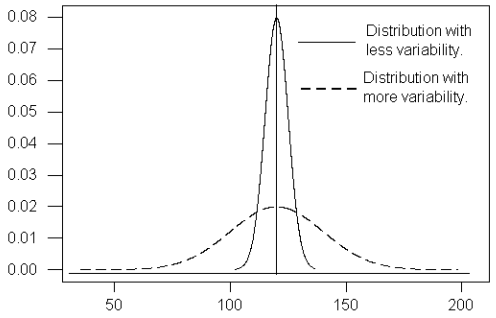
$$COV(X,Y) = \frac{\sum_{i=1}^{n}\left(X_i - \bar{X}\right)\left(Y_i - \bar{Y}\right)}{n-1}$$

Pearson's correlation r

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

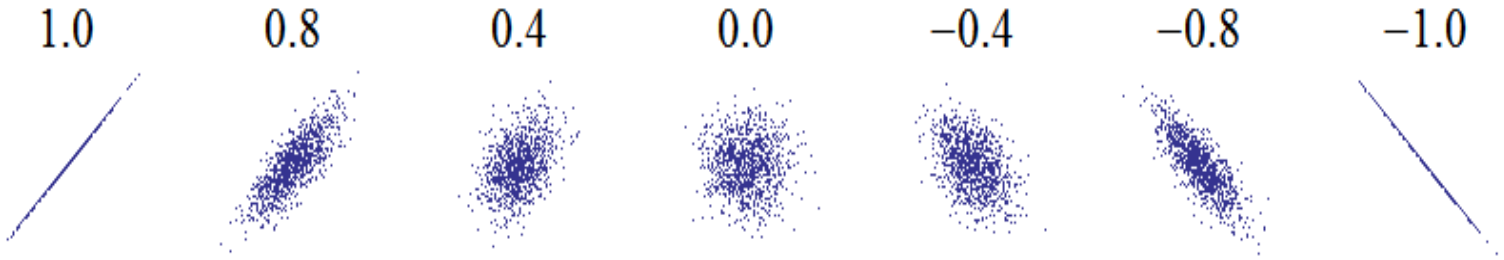
$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{x})^2}}$$

individual variances for attributes X and Y



Correlation Patterns

Correlation rates between -1 and 1:



Important to note:

- correlation is defined for linear relationships
- visualization can help
- none of these point distributions have correlations:

Covariance Matrix

Analytical: $Cov(X,Y) = E[(X - \mu_x)(Y - \mu_y)]$

Samples: $\sigma_{xy} = \mathrm{cov}_{xy} = \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$

An n-D dataset has *n* variables $x_1, x_2, \ldots x_n$

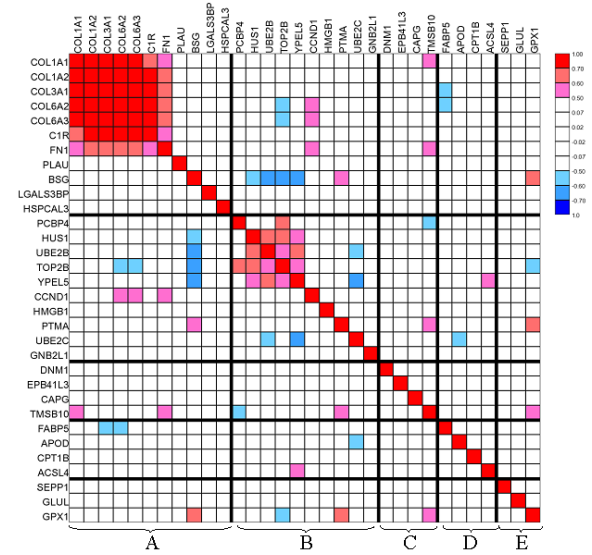
- define pairwise covariance among all of these variables
- construct a covariance matrix

$$\Sigma = \mathrm{Cov}(\mathbf{X}) = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix}$$

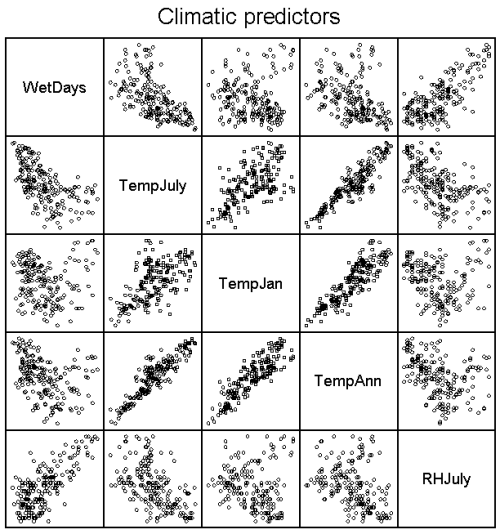
- a correlation matrix would just list the correlations instead

Correlation Matrix

	MO	FP	MP	IM	IC	FM	FE	FI	SPC	DSC	DST
MO	1.00										
FP	0.31[a]	1.00									
MP	0.32[a]	0.71[a]	1.00								
IM	0.36[a]	0.12[c]	0.14[c]	1.00							
IC	0.39[a]	0.18[b]	0.21[a]	0.62[a]	1.00						
FM	0.26[a]	0.21[a]	0.14[c]	0.30[a]	0.27[a]	1.00					
FE	0.47[a]	0.21[a]	0.18[b]	0.38[a]	0.28[a]	0.24[a]	1.00				
FI	0.53[a]	0.26[a]	0.22[a]	0.36[a]	0.37[a]	0.29[a]	0.47[a]	1.00			
SPC	0.32[a]	0.22[a]	0.31[a]	0.51[a]	0.47[a]	0.32[a]	0.37[a]	0.35[a]	1.00		
DSC	−0.12[c]	0.03[c]	0.05[c]	0.17[b]	0.08[c]	0.18[b]	−0.05[c]	0.06[c]	0.01[c]	1.00	
DST	−0.02[c]	−0.01[c]	0.05[c]	0.24[a]	0.14[c]	0.05[c]	−0.05[c]	0.05[c]	0.05[c]	0.56[a]	1.00
DM	0.05[c]	0.144	0.136[c]	0.199[a]	0.169[b]	0.247[a]	0.08[c]	0.11[c]	0.14[c]	0.46[a]	0.71[a]



just value

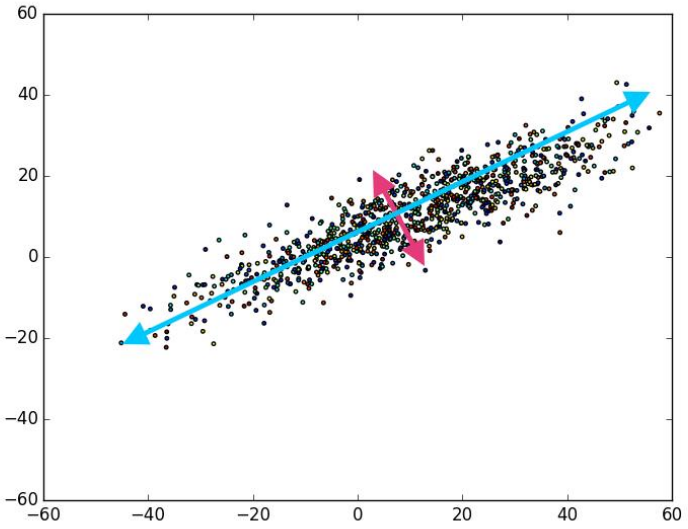


distribution (scatterplot matrix)

Principal Component Analysis

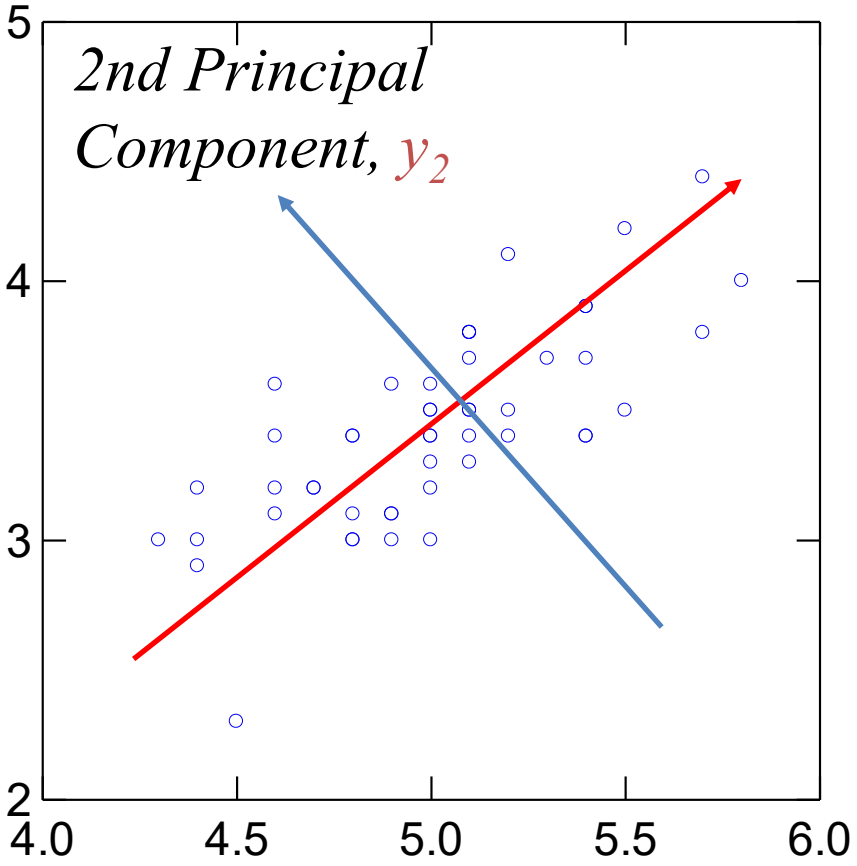
Ultimate goal:

- 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



PCA – How To Do

Find the principal components (factors) of a distribution

First characterize the distribution by

- covariance matrix Cov
- correlation matrix Corr
- lets call it C

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

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

Q: matrix with Eigenvectors

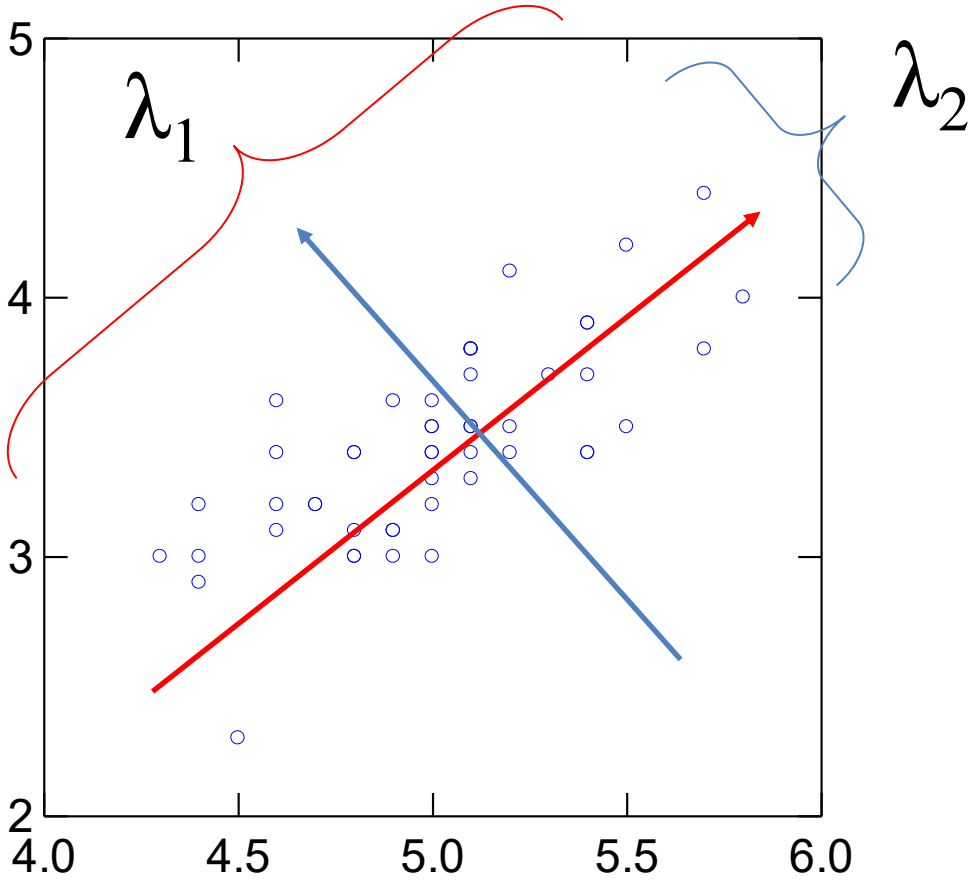
Λ: diagonal matrix with Eigenvalues λ

- now order the Eigenvectors in terms of their Eigenvalues λ

Eigenvectors and Eigenvalues

λ_1, λ_2 are the Eigenvalues

- encode the length (and therefore significance) of the Eigenvectors



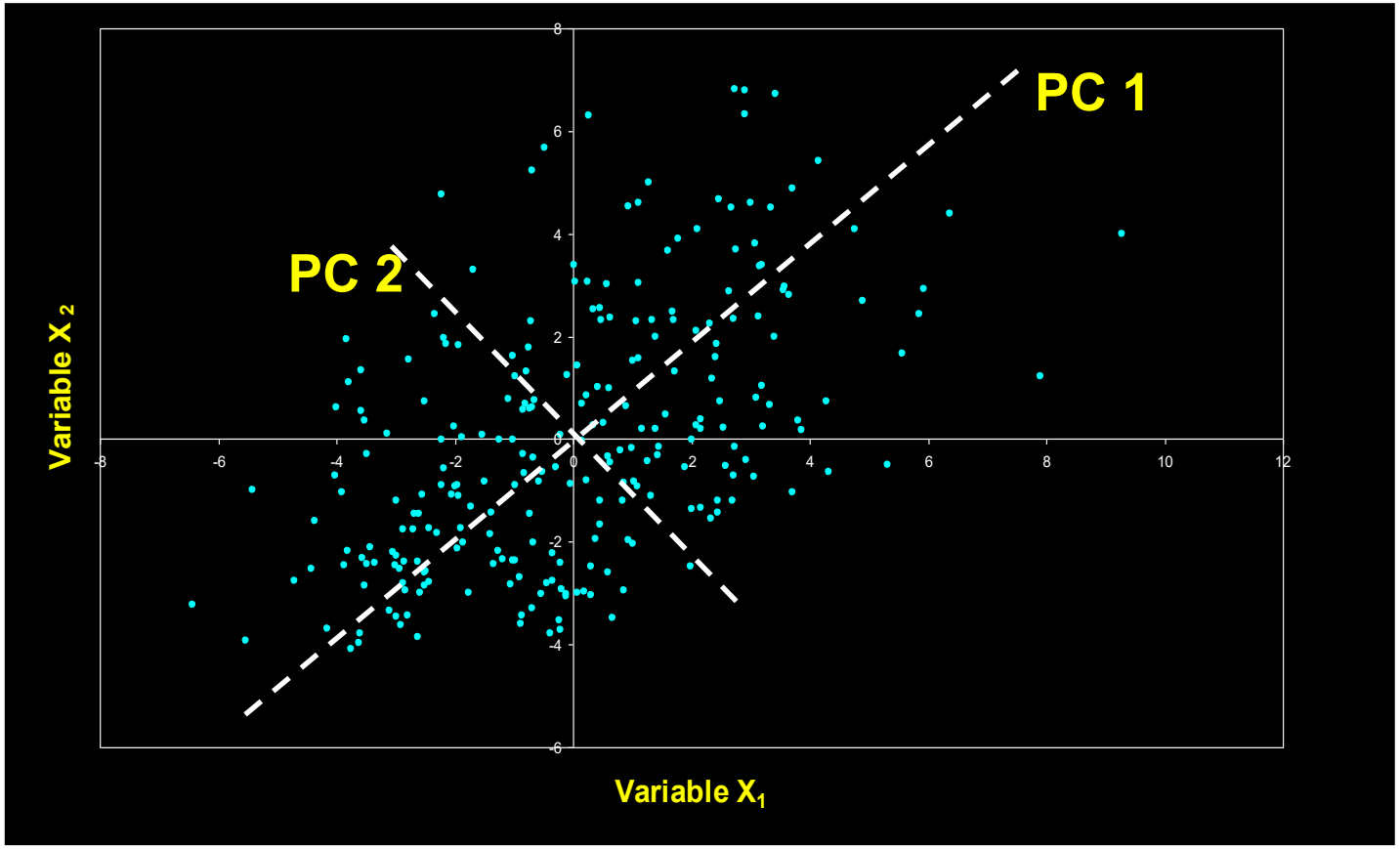
Covariance vs. Correlation

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

EXAMPLE

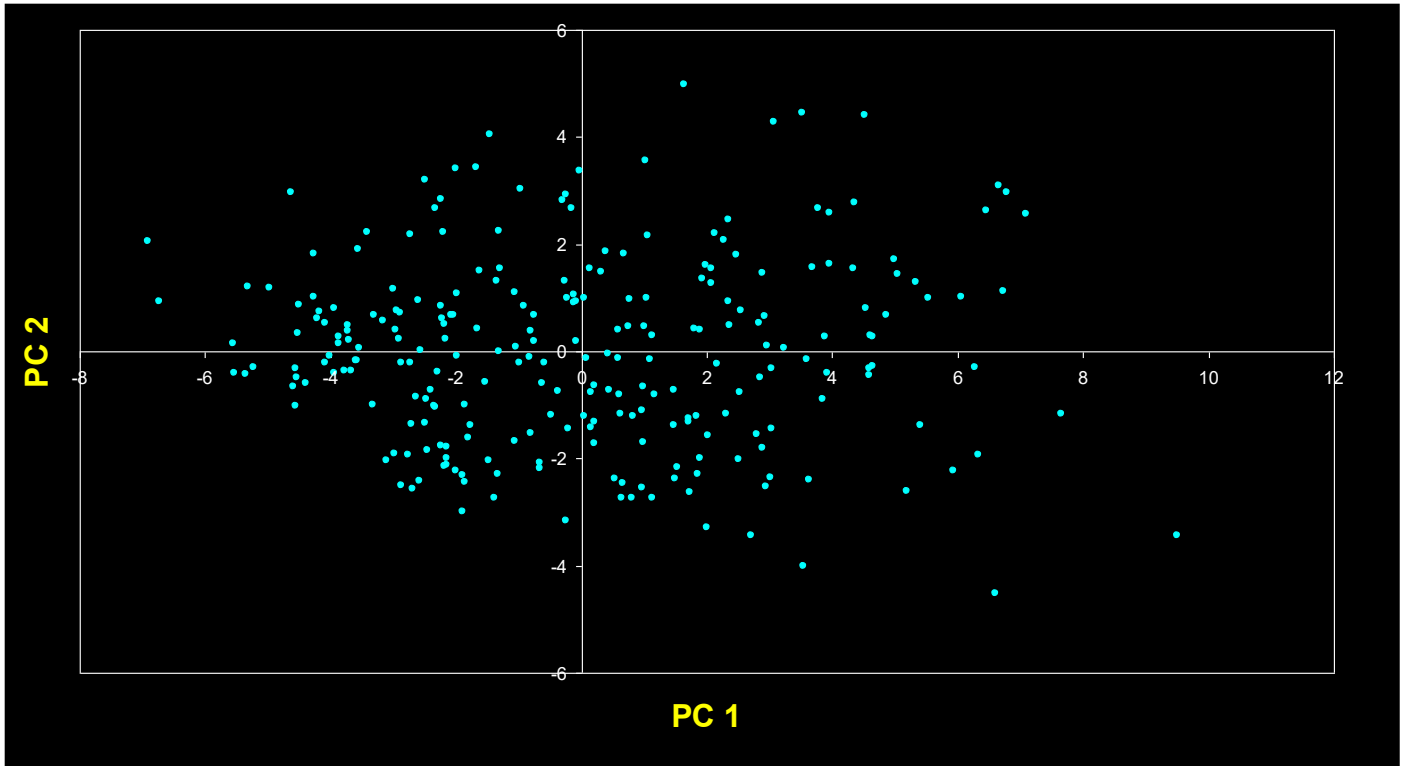
Before PCA



Example

After PCA

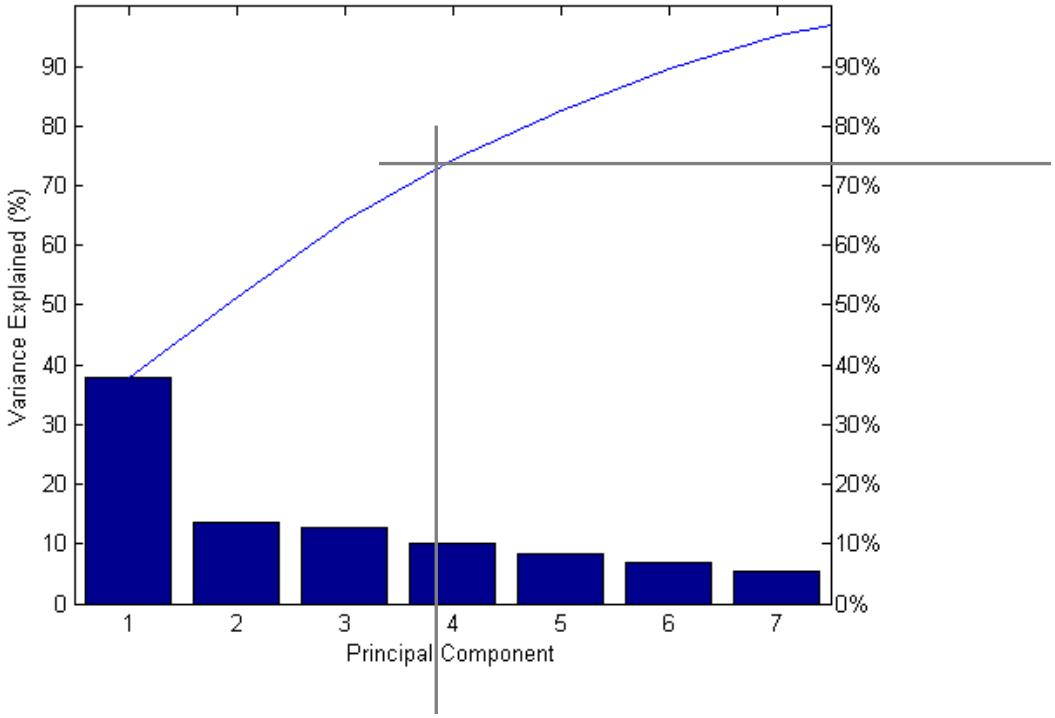
- λ_1 = 9.8783 λ_2 = 3.0308 Trace = 12.9091
- PC 1 displays ("explains") 9.8783/12.9091 = 76.5% of total variance



Dimension Reduction

Create a *scree plot*

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



possible threshold (explain 75% of data variance)

keep top 3 principal components → reduce dimensions by a factor of 4/7 = 57%

PCA Applied To Faces

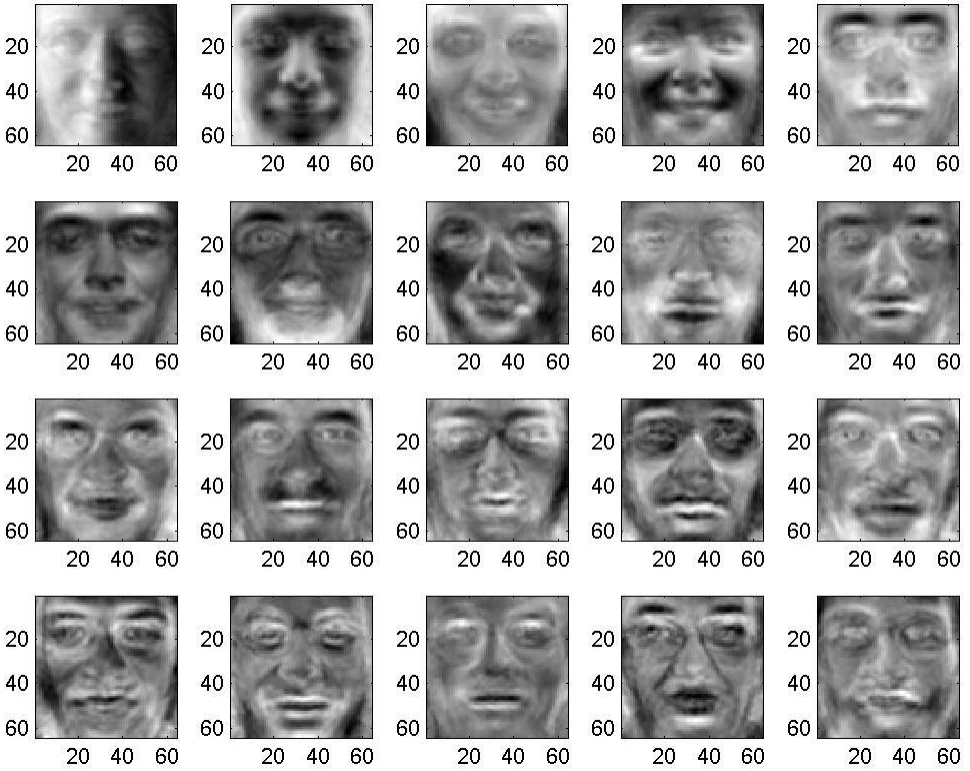
Some familiar faces…

PCA Applied To Faces

We can reconstruct each face as a linear combination of "basis" faces, or Eigenfaces [M. Turk and A. Pentland (1991)]

Average Face

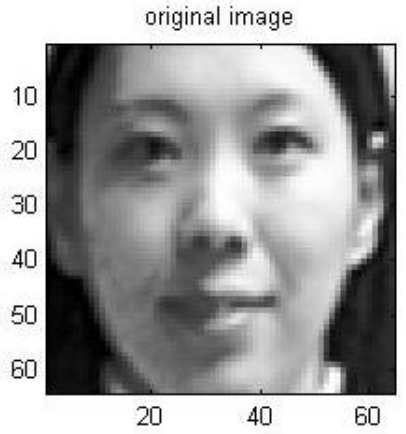
+

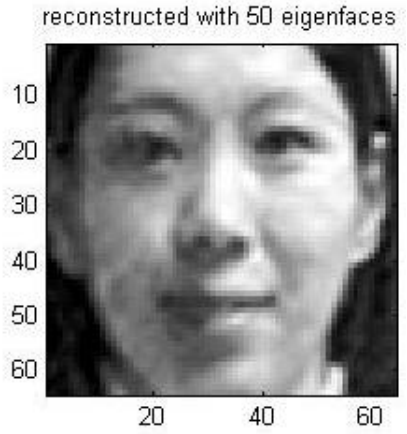


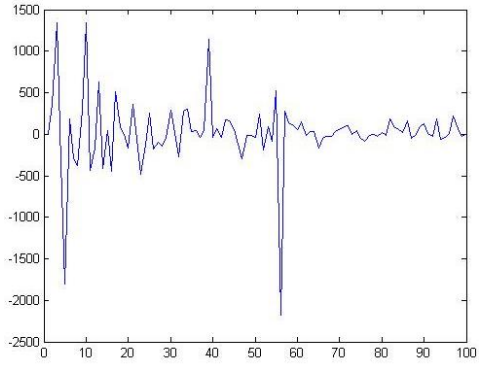
Eigenfaces

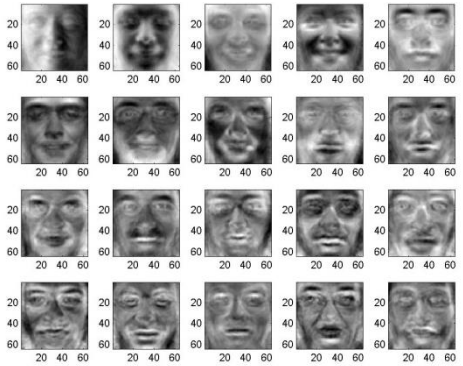
Reconstruction Using PCA

- 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









A More Challenging Example

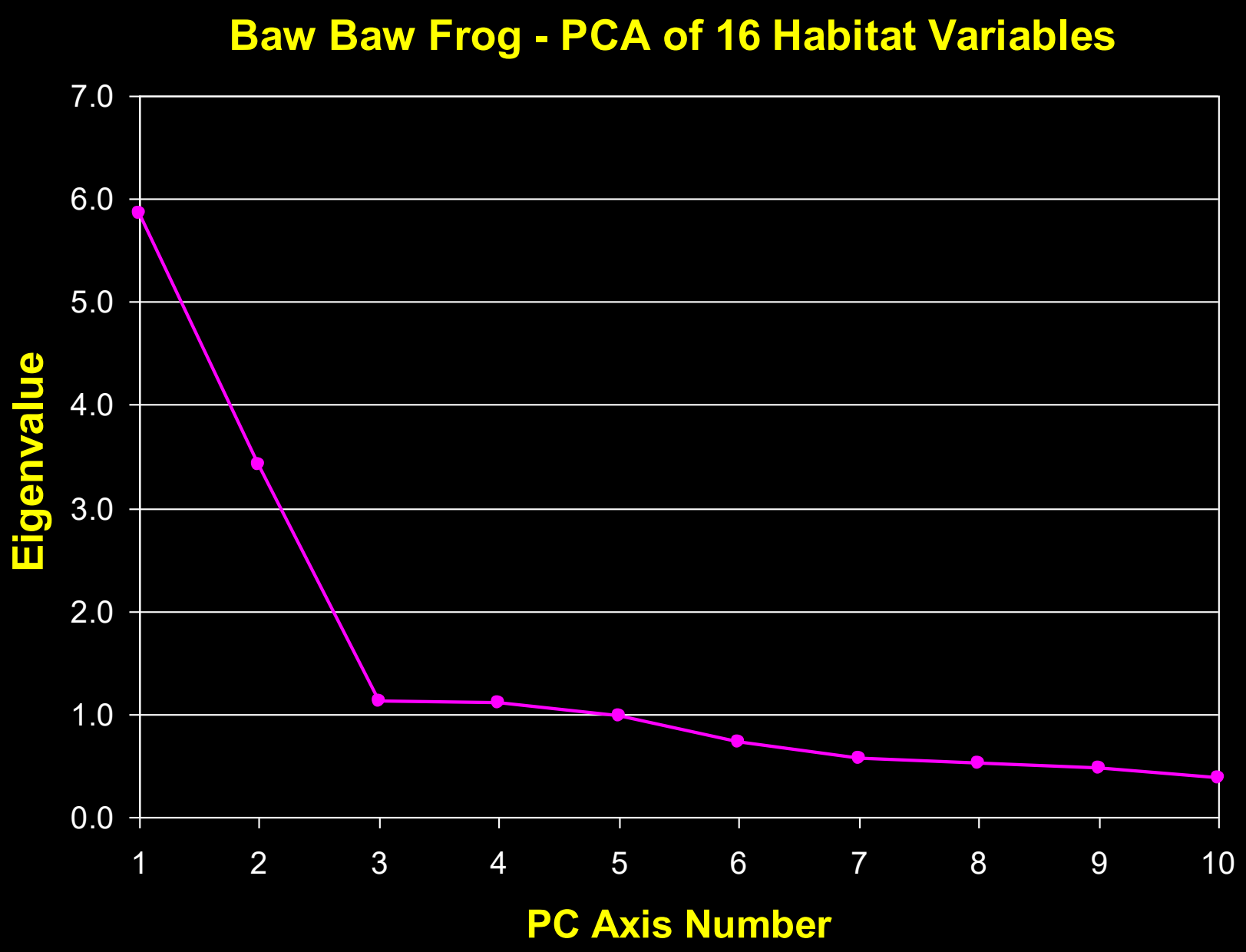
- Data from research on habitat definition in the endangered Baw Baw frog
- 16 environmental and structural variables measured at each of 124 sites
- Correlation matrix used because variables have different units

Eigenvalues

Axis	Eigenvalue	% of Variance	Cumulative % of Variance
1	5.855	36.60	36.60
2	3.420	21.38	57.97
3	1.122	7.01	64.98
4	1.116	6.97	71.95
5	0.982	6.14	78.09
6	0.725	4.53	82.62
7	0.563	3.52	86.14
8	0.529	3.31	89.45
9	0.476	2.98	92.42
10	0.375	2.35	94.77

How Many Axes Are Needed?

- Does the $(k+1)^{th}$ principal axis represent more variance than would be expected by chance?
- Several tests and rules have been proposed
- A common "rule of thumb" when PCA is based on correlations is that axes with eigenvalues > 1 are worth interpreting
- In our example 4 Eigenvectors fit this criterion (we shall keep 3 for simplicity)


Baw Baw Frog - PCA of 16 Habitat Variables
Eigenvalue
7.0
6.0
5.0
4.0
3.0
2.0
1.0
0.0
1
2
3
4
5
6
7
8
9
10
PC Axis Number

Interpreting Eigenvectors

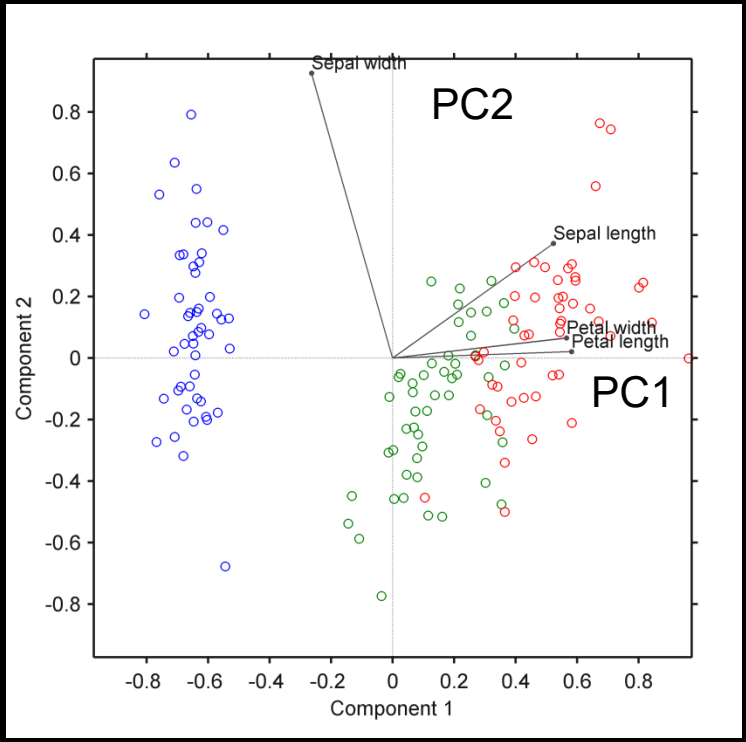
- Correlations between variables and the principal axes are known as **loadings**
- Each element of the eigenvectors represents the contribution of a given variable to a component
- The loadings of variables on the first three PCs are shown here

	PC 1	PC 2	PC 3
Altitude	0.3842	0.0659	-0.1177
pH	-0.1159	0.1696	-0.5578
Cond	-0.2729	-0.1200	0.3636
TempSurf	0.0538	-0.2800	0.2621
Relief	-0.0765	0.3855	-0.1462
maxERht	0.0248	0.4879	0.2426
avERht	0.0599	0.4568	0.2497
%ER	0.0789	0.4223	0.2278
%VEG	0.3305	-0.2087	-0.0276
%LIT	-0.3053	0.1226	0.1145
%LOG	-0.3144	0.0402	-0.1067
%W	-0.0886	-0.0654	-0.1171
H1Moss	0.1364	-0.1262	0.4761
DistSWH	-0.3787	0.0101	0.0042
DistSW	-0.3494	-0.1283	0.1166
DistMF	0.3899	0.0586	-0.0175

What's a "Loading"?

- The amount of weight a data dimension has on a principal component
  - petal length/width have a high loading on PC1
  - sepal width has a high loading on PC2

BiPlot



- Another observation
  - projection into PC basis can also bring out clusters better
  - since spread is maximized

Significance of Variables

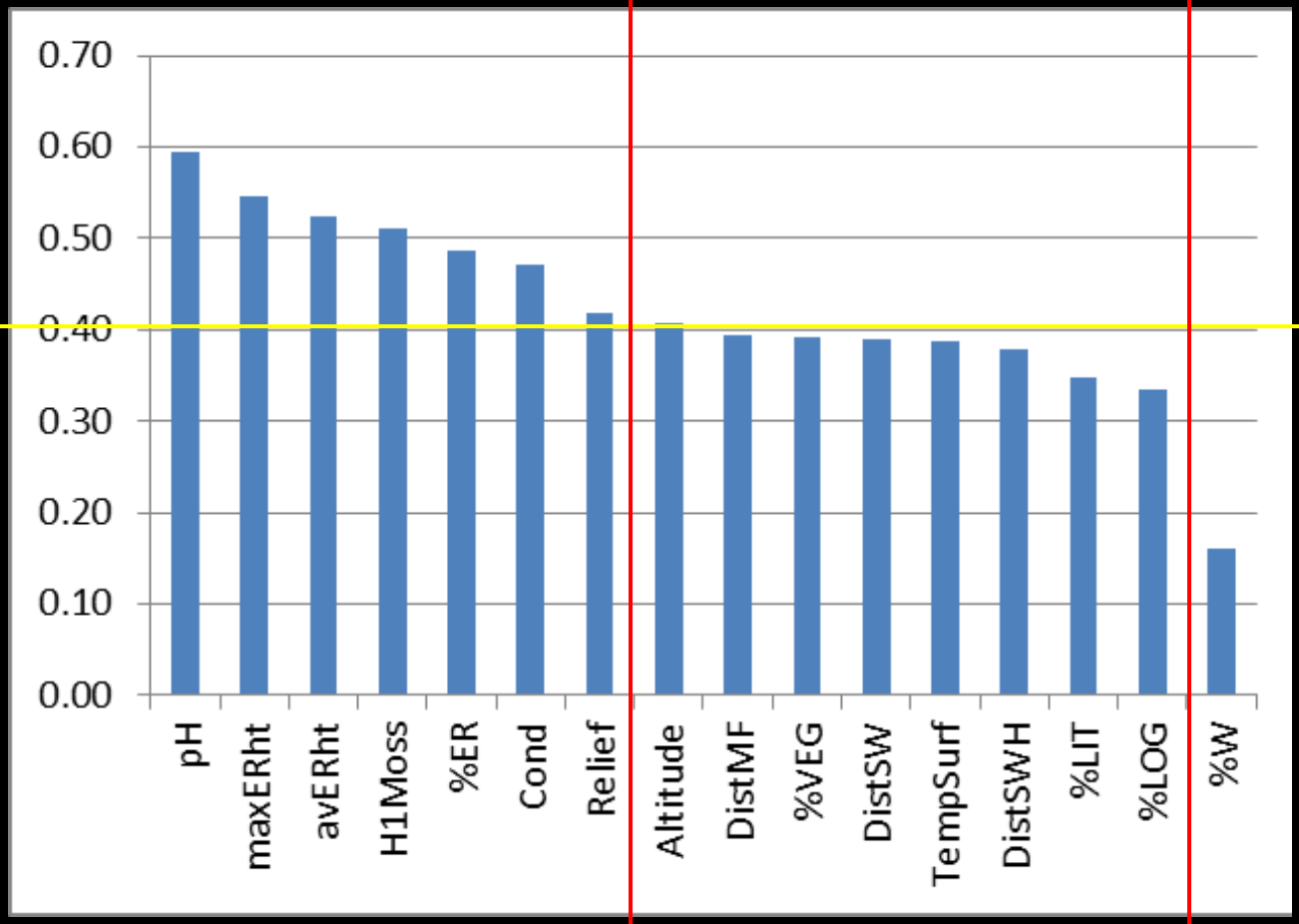
- We can compute the significance of the variables as the sum of squared loadings on to the most significant Eigenvectors we selected (3 in our example)
- The next slide shows the table of the last slide expanded with these squared loadings
- We can then sort the table by the squared loadings and make a scree plot
- The most significant variables are those above some chosen cutoff, for example 0.4 (marked in yellow in the table)

Significance of Variables

	PC 1	PC 2	PC 3	sum of squared loadings
Altitude	0.3842	0.0659	-0.1177	0.41
pH	-0.1159	0.1696	-0.5578	0.59
Cond	-0.2729	-0.1200	0.3636	0.47
TempSurf	0.0538	-0.2800	0.2621	0.39
Relief	-0.0765	0.3855	-0.1462	0.42
maxERht	0.0248	0.4879	0.2426	0.55
avERht	0.0599	0.4568	0.2497	0.52
%ER	0.0789	0.4223	0.2278	0.49
%VEG	0.3305	-0.2087	-0.0276	0.39
%LIT	-0.3053	0.1226	0.1145	0.35
%LOG	-0.3144	0.0402	-0.1067	0.33
%W	-0.0886	-0.0654	-0.1171	0.16
H1Moss	0.1364	-0.1262	0.4761	0.51
DistSWH	-0.3787	0.0101	0.0042	0.38
DistSW	-0.3494	-0.1283	0.1166	0.39
DistMF	0.3899	0.0586	-0.0175	0.39

Significance of Variables

- Scree plot



chosen significance threshold

variables considered significant

more aggressive reduction of variables

only eliminate very weak variables

Summary

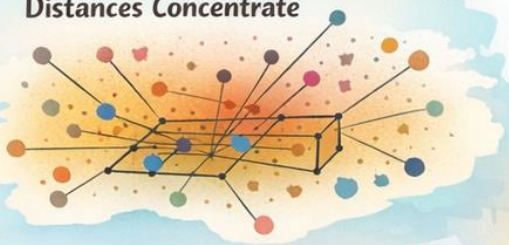
High-Dimensional Space & Dimension Reduction

Klaus Mueller - Stony Brook University

The Curse of Dimensionality

Sparse & Hard to Grasp

Distances Concentrate



SUBSPACES

Finding Structure in the Data

High-Dimensional Data

Feature Vectors in $\mathbf{R}^n$ (n = many dimensions)

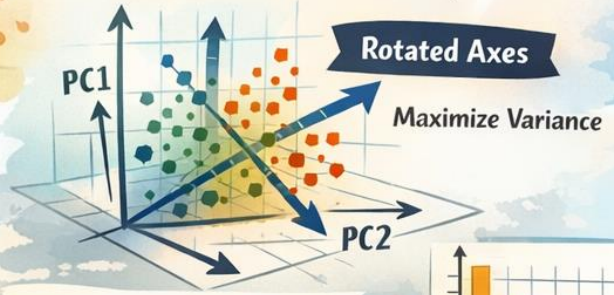
1	2	9	4	5	6	7
2	9	2	3	6	5	7
1	2	3	1	9	5	7

Data Points

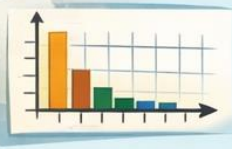
Need to Find Meaningful Patterns!

PCA

Principal Component Analysis



Eigenvalues & Scree Plot



Semantic Subspaces

Grouped by Attributes or Meaning

Two Paths: Semantic Grouping OR Algebraic Rotation

Why High-D Data Break Intuition

High-D data consists of feature vectors in $\mathbb{R}^n$, not objects we can directly "see"

- As dimensionality increases:
  - space becomes sparse
  - distances concentrate (points are all equally far apart)
  - indexing and memory costs grow exponentially
- As a result, distance alone becomes meaningless unless structure exists
- Structure lives in subspaces, not in the full dimensional space

→ Understanding high-dimensional data means finding the right subspaces

Subspace Discovery and Dimension Reduction

Semantic / Attribute-Driven

- Subspaces defined by meaningful groups of attributes
- Guided by domain knowledge or semantic similarity
- Interpretable by construction
  (e.g., expenditures on food, services, housing)

Algebraic / Variance-Driven (PCA)

- PCA rotates the coordinate system to find efficient axes
- Each principal component:
  - is a linear subspace
  - captures decreasing amounts of variance
- Eigenvalues rank subspaces by importance
- Loadings connect components back to original attributes
- PCA is a systematic decomposition of high-dimensional space into ordered linear subspaces.