Dimensionality Reduction

Ms. Sandhya Harikumar,

Department of Computer Science & Engineering,

Amrita Vishwa Vidyapeetham, Amritapuri

Dimensionality of Dataset

- (X1,X2,….,Xn) : Samples/Rows/Tuples/Instances/ Observations

- (f1,…,fd) : Features/variables for a dataset.

- In statistics, attributes of matrix is referred as the dimensions.

	f1	f2	f3	f4	f5	Y
X1						
X2						
X3						
X4						
X5						
X6						
X7						

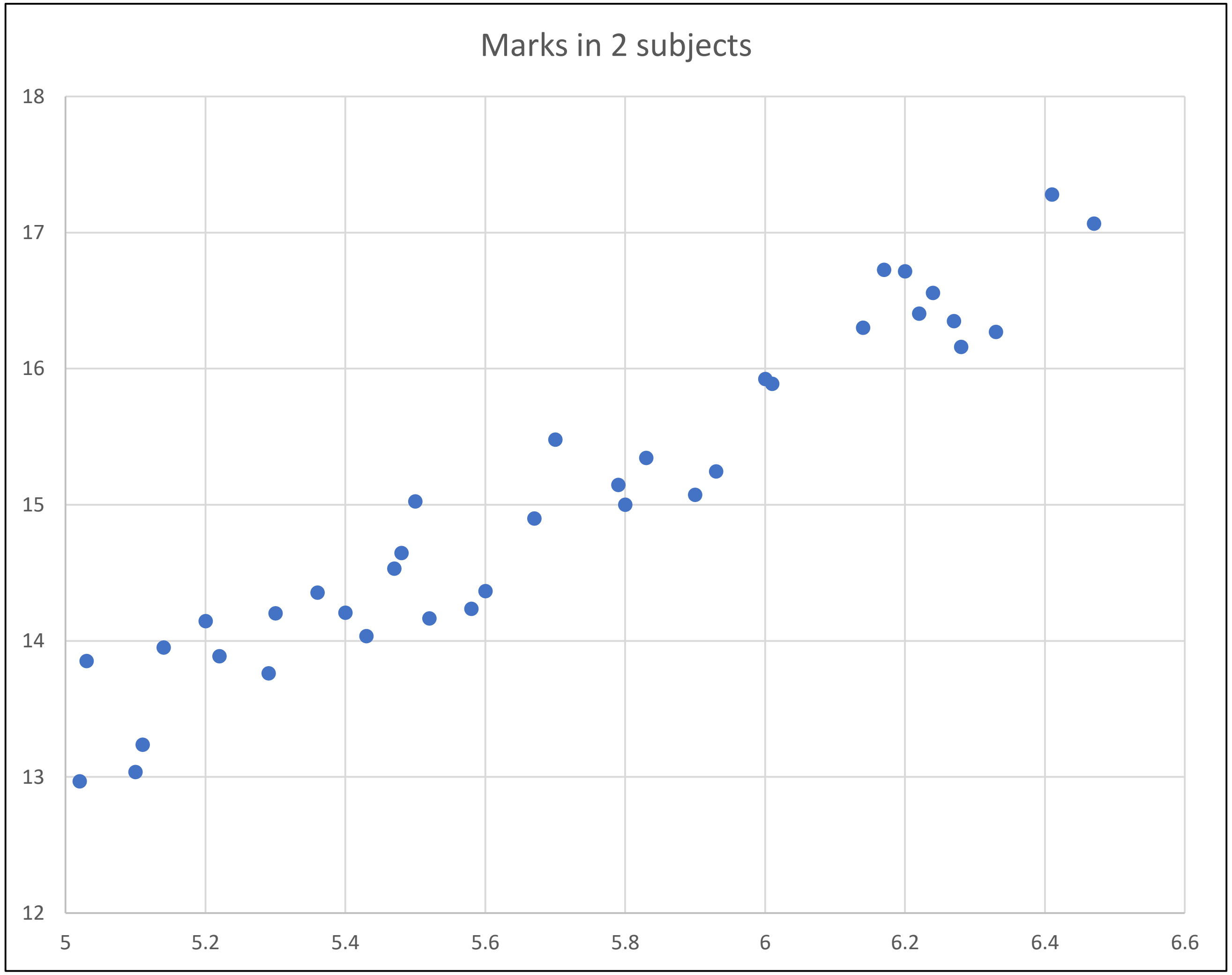
Original versus Observed Dimensionality

- Predict the performance of students
- 2-d data (mark1,mark2)

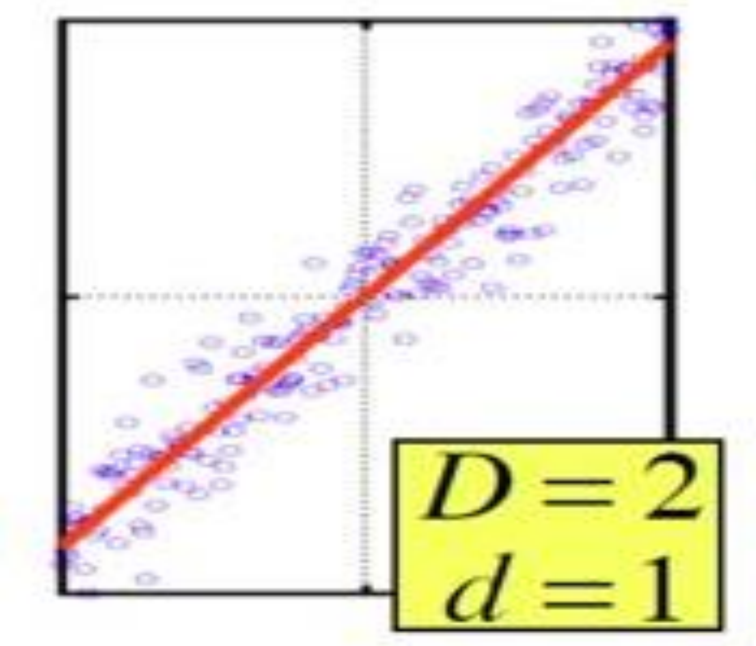
- Correlated

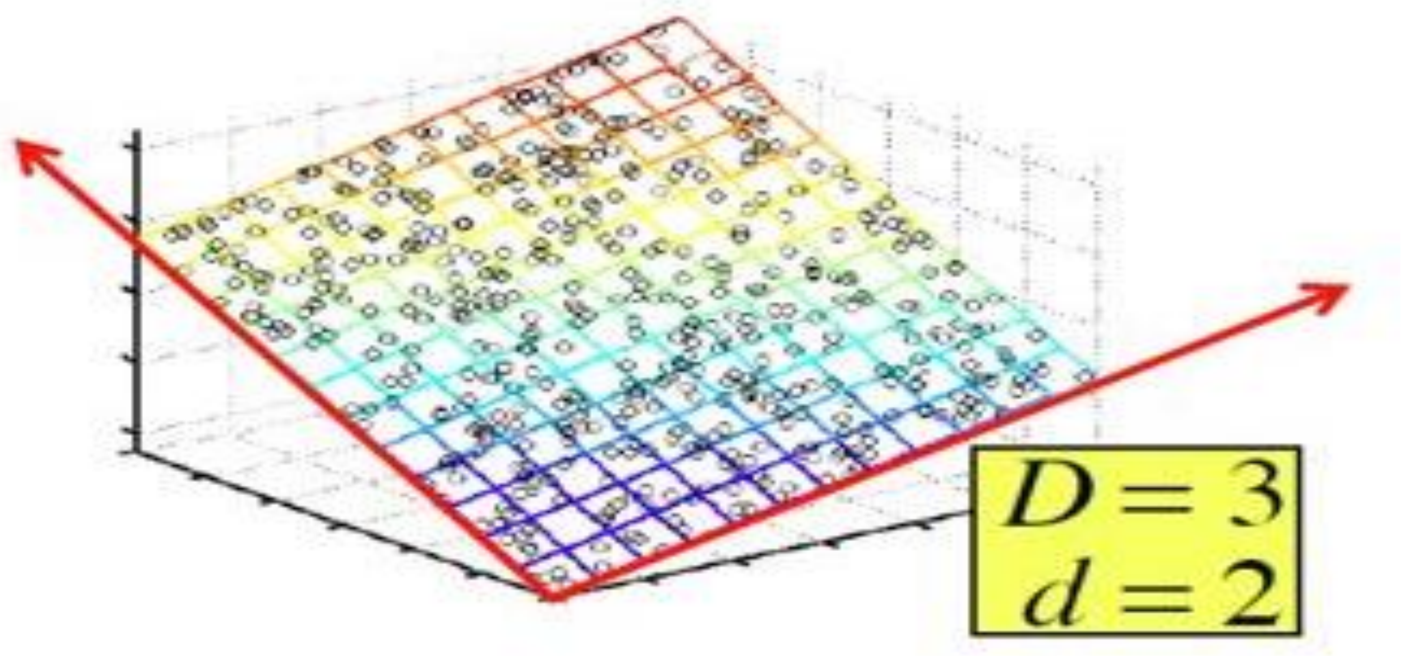
- Do we really require
  2-dimensional ?

REAL – 1-d data
OBSERVED – 2-d data



Dimensionality Reduction





- Dimensionality Reduction : A Technique to reduce the size of data

Why Dimensionality Reduction

High Dimensional Data difficult to handle

- Image , Text Documents, Biological databases

Data Compression

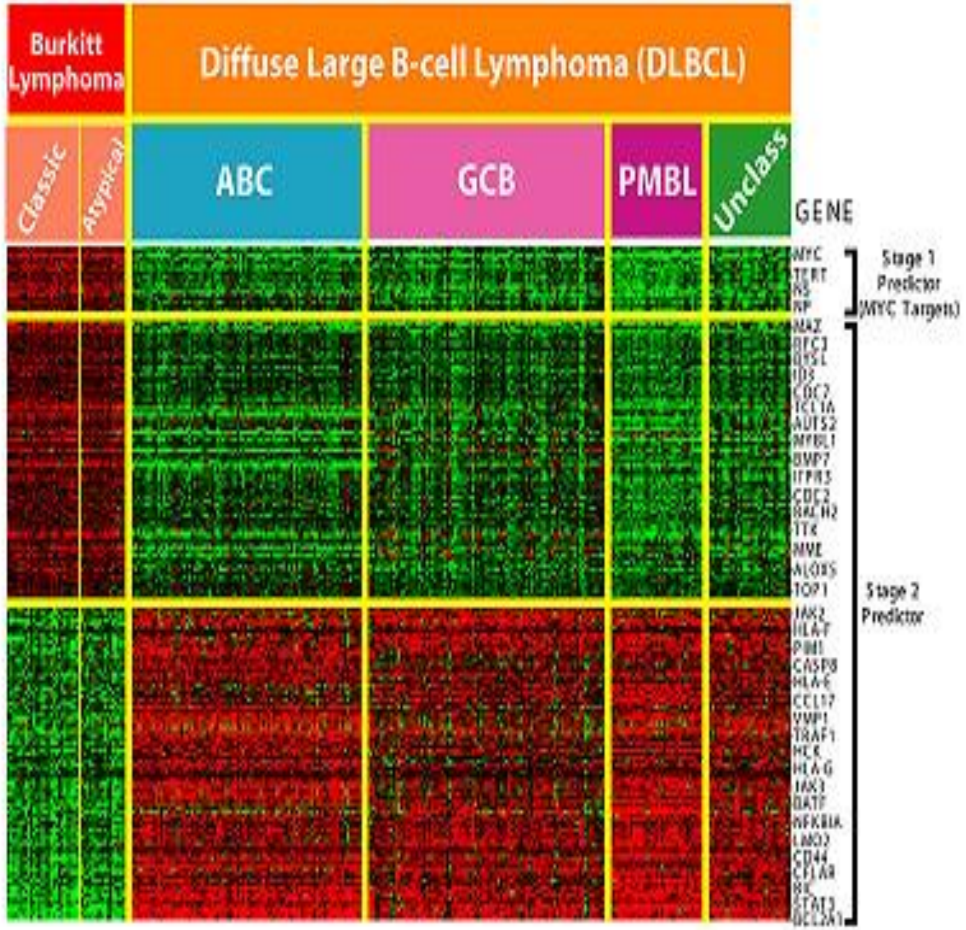
Better visualization

A smaller subspace to keep most of the information about the original data

Preprocessing Step before applying supervised learning algorithm

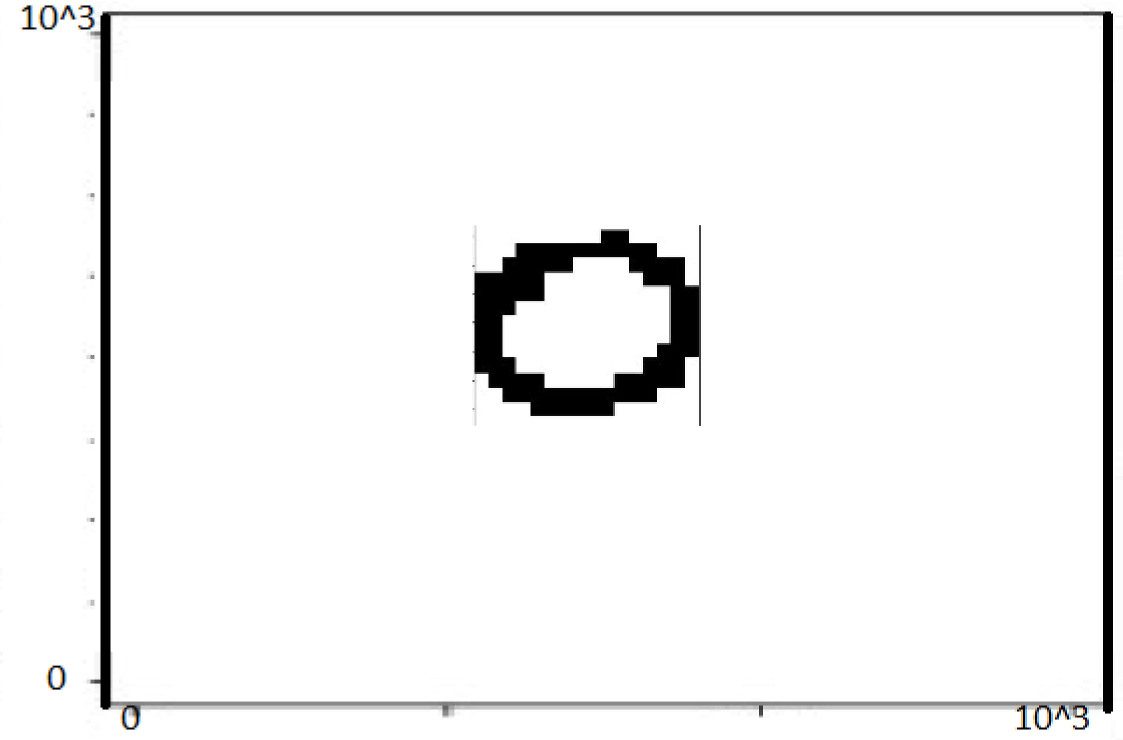
High Dimensional Data

- ➢Number of dimensions are staggeringly high
- ➢Image Data : 10^6 pixels
- ➢Text Data : 10^10 words
- ➢Biological databases : 10^20
- ➢Difficult to handle data

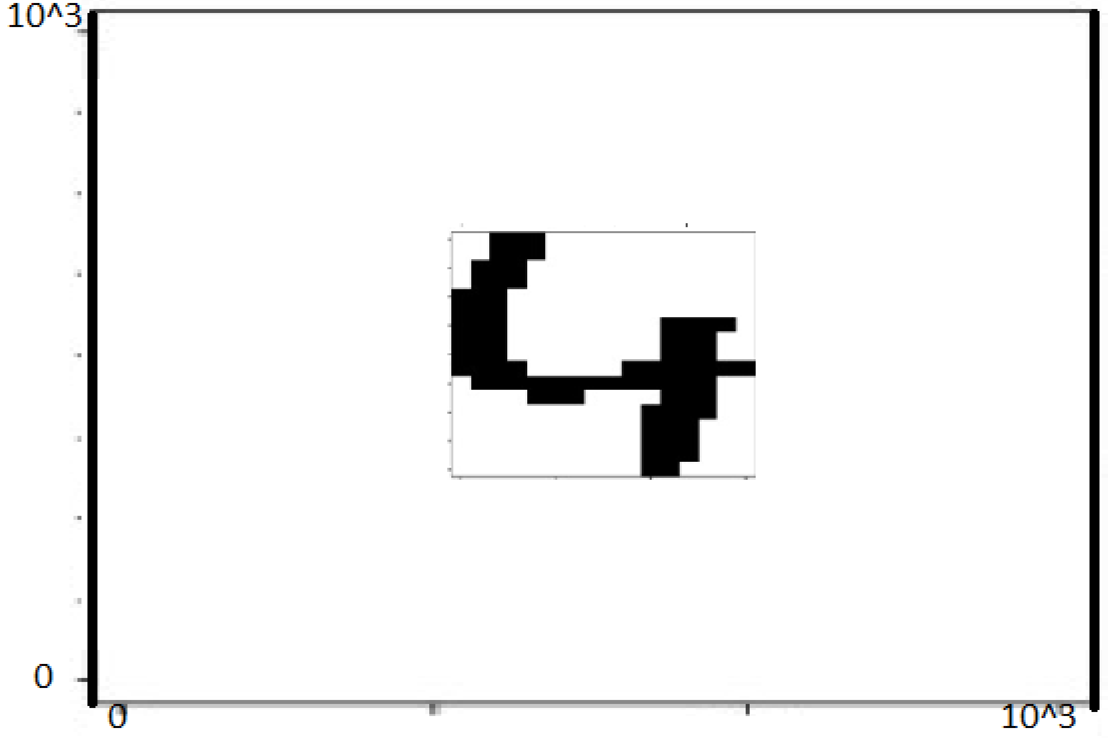


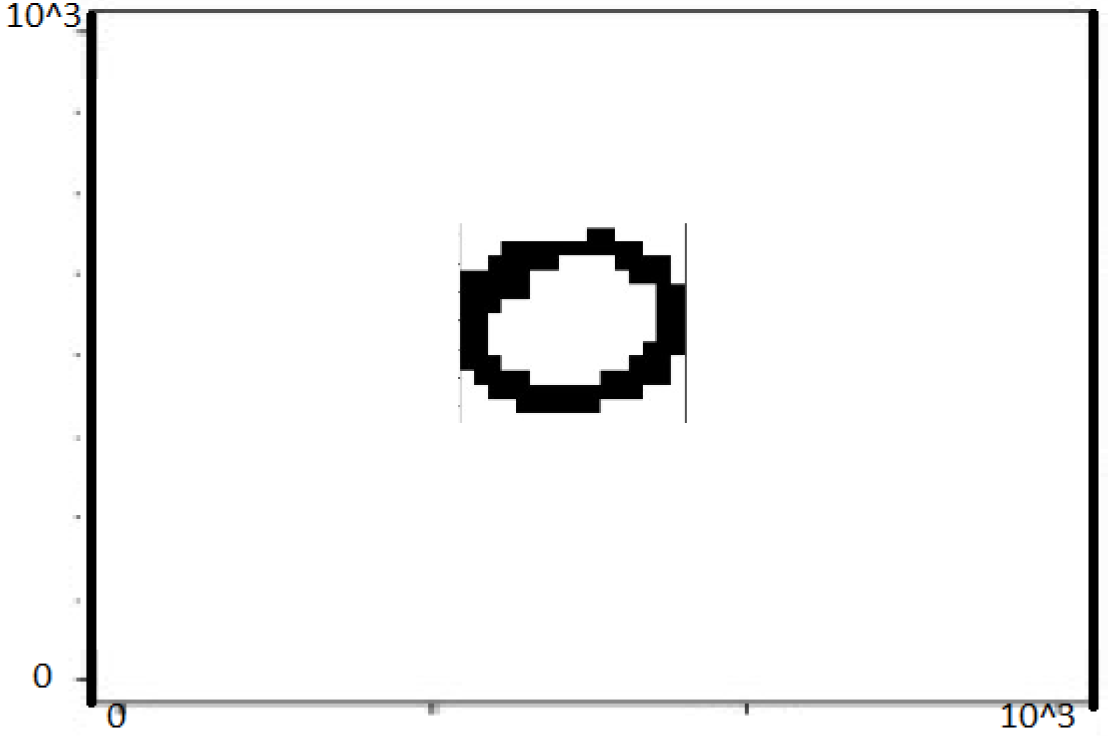
High Dimensional Data : Sparsity

- 1000 x 1000



Both images seem similar overall



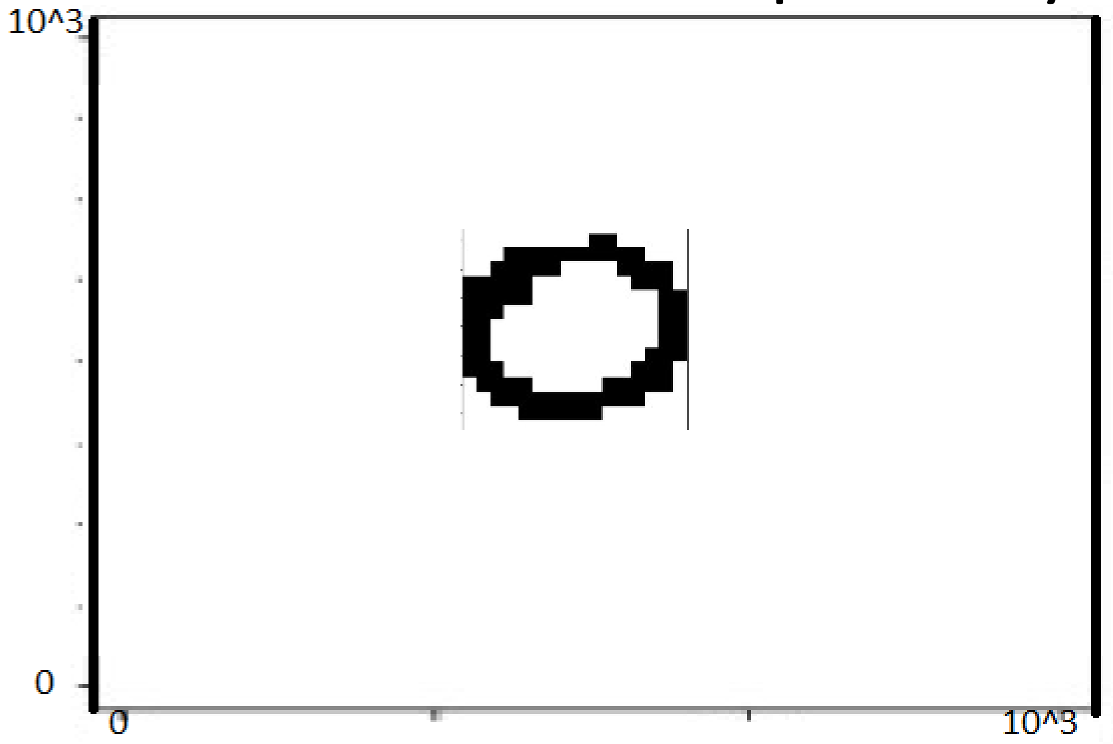


[0 0 0 0 ………1 ….0…..1……0…… …………………………..]

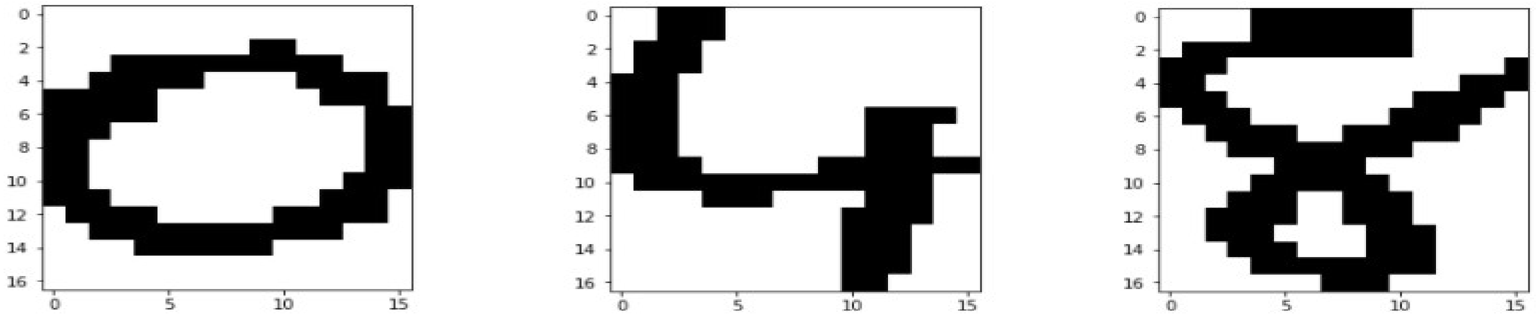
[0 1 0 0 ………1 ….0…..1……0…… …………………………..]

High Dimensional Data : Sparsity

- 1000 x 1000

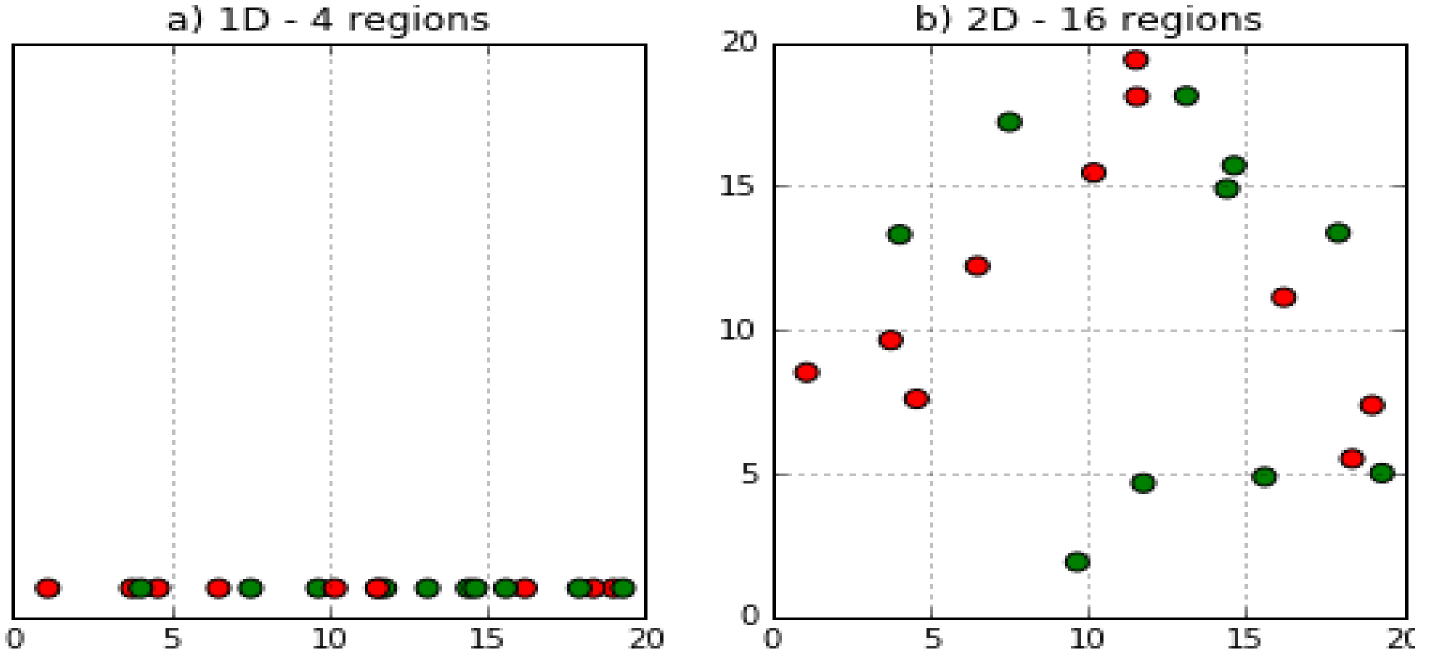


- 16 x 16

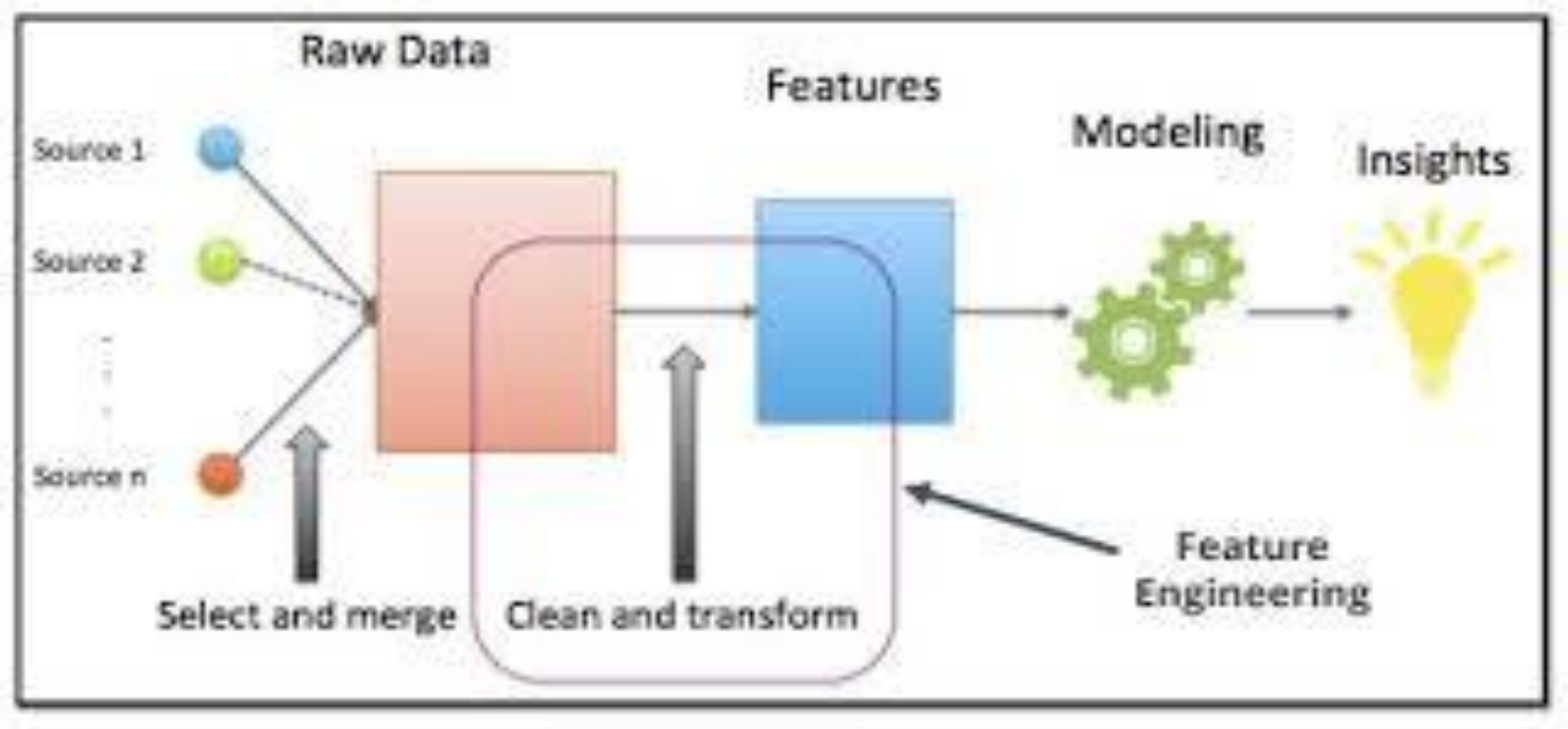


Curse of Dimensionality

- As the dimensions increase the volume of the space increase.
- Requirement of the number of samples increase exponentially to really understand the data.



Preprocessing



Visualization : Dimensionality Reduction

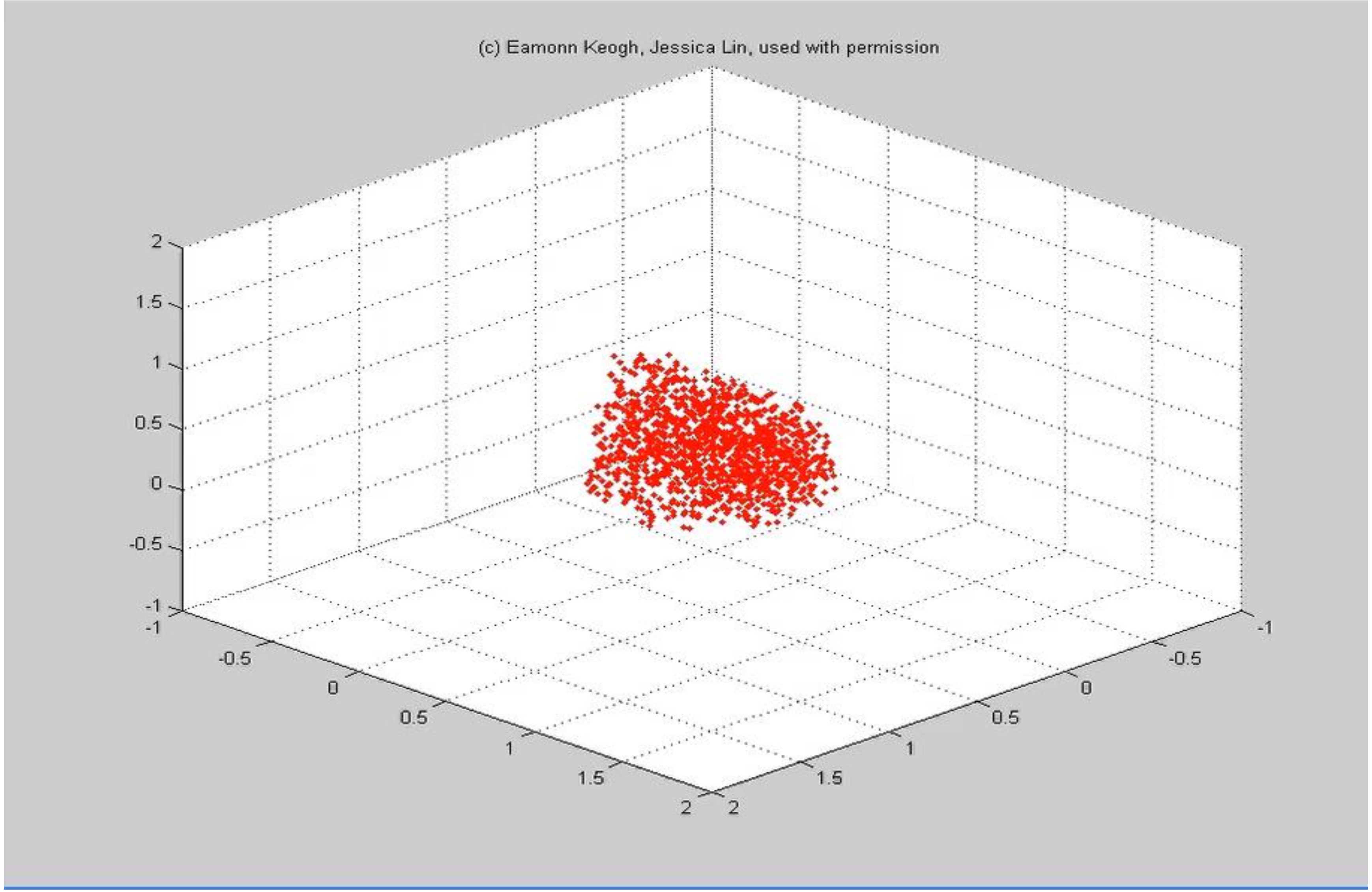
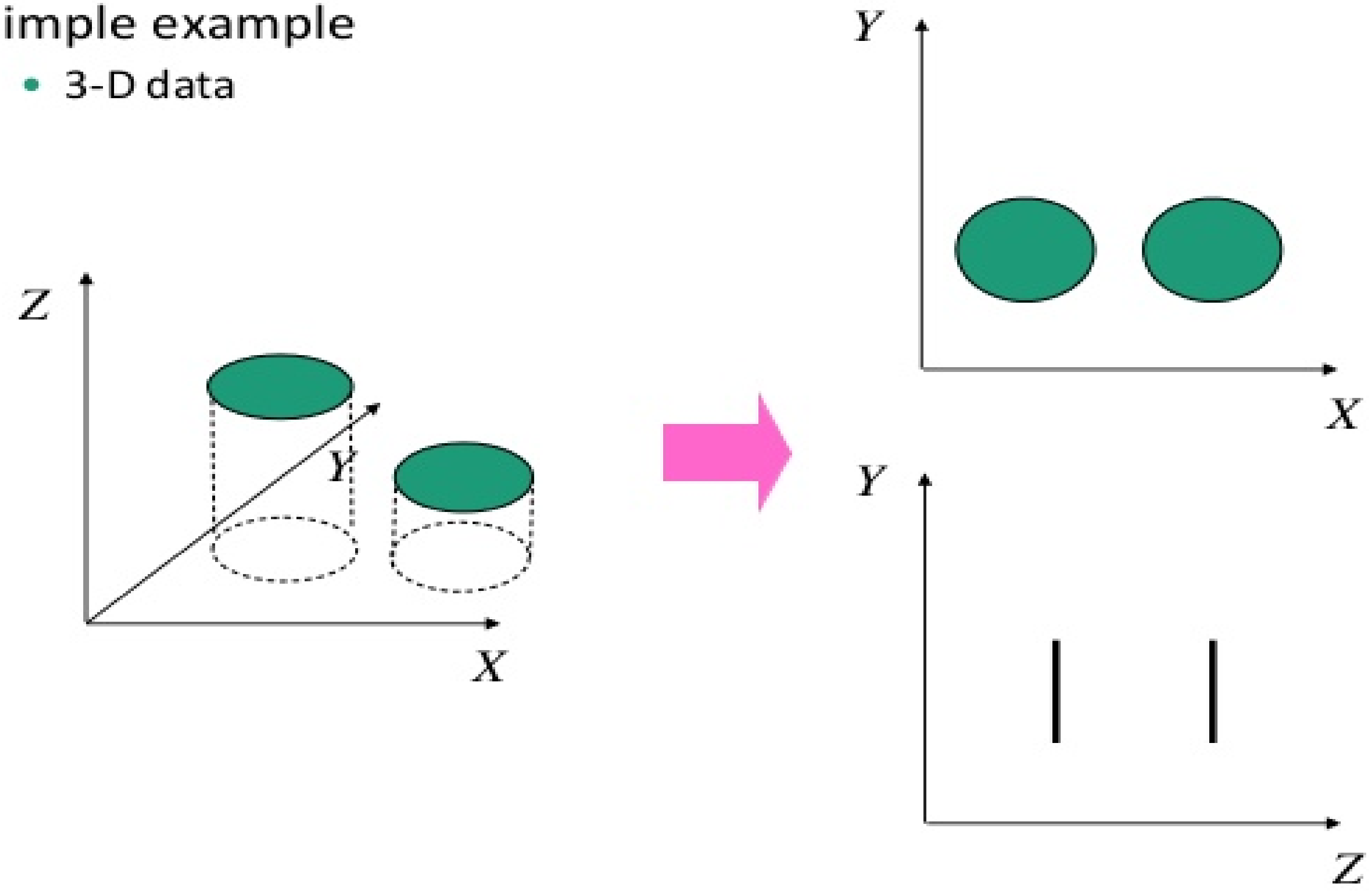


Image Source : https://cs.gmu.edu/~jessica/DimReducDanger.htm

Dimensionality Reduction

- Simple example
    - 3-D data



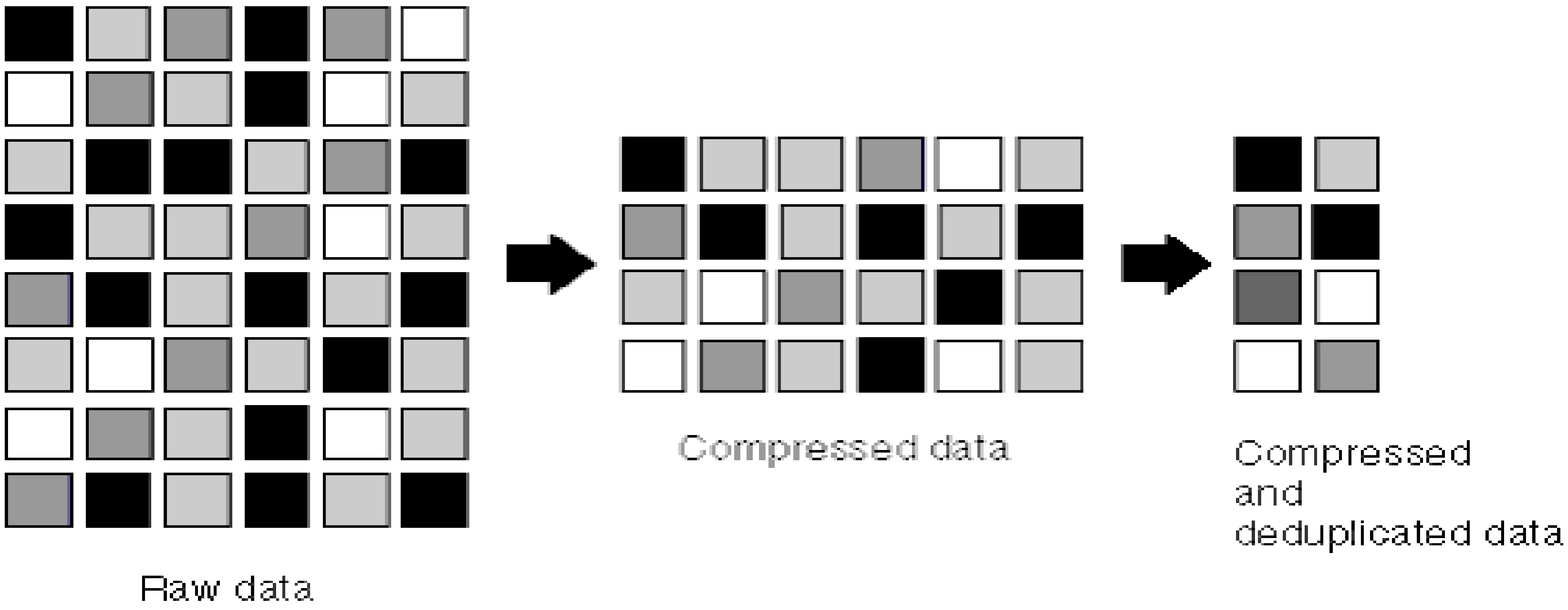
Dimensionality Reduction

day / customer	We 7/10/96	Th 7/11/96	Fr 7/12/96	Sa 7/13/96	Su 7/14/96
ABC Inc.	1	1	1	0	0
DEF Ltd.	2	2	2	0	0
GHI Inc.	1	1	1	0	0
KLM Co.	5	5	5	0	0
Smith	0	0	0	2	2
Johnson	0	0	0	3	3
Thompson	0	0	0	1	1

The above matrix is really “2-dimensional.” All rows can be reconstructed by scaling [1 1 1 0 0] or [0 0 0 1 1]

[5 5 5 0 0] = 5* [1 1 1 0 0] +0* [0 0 0 1 1]

Data compression



Why Reduce Dimensions

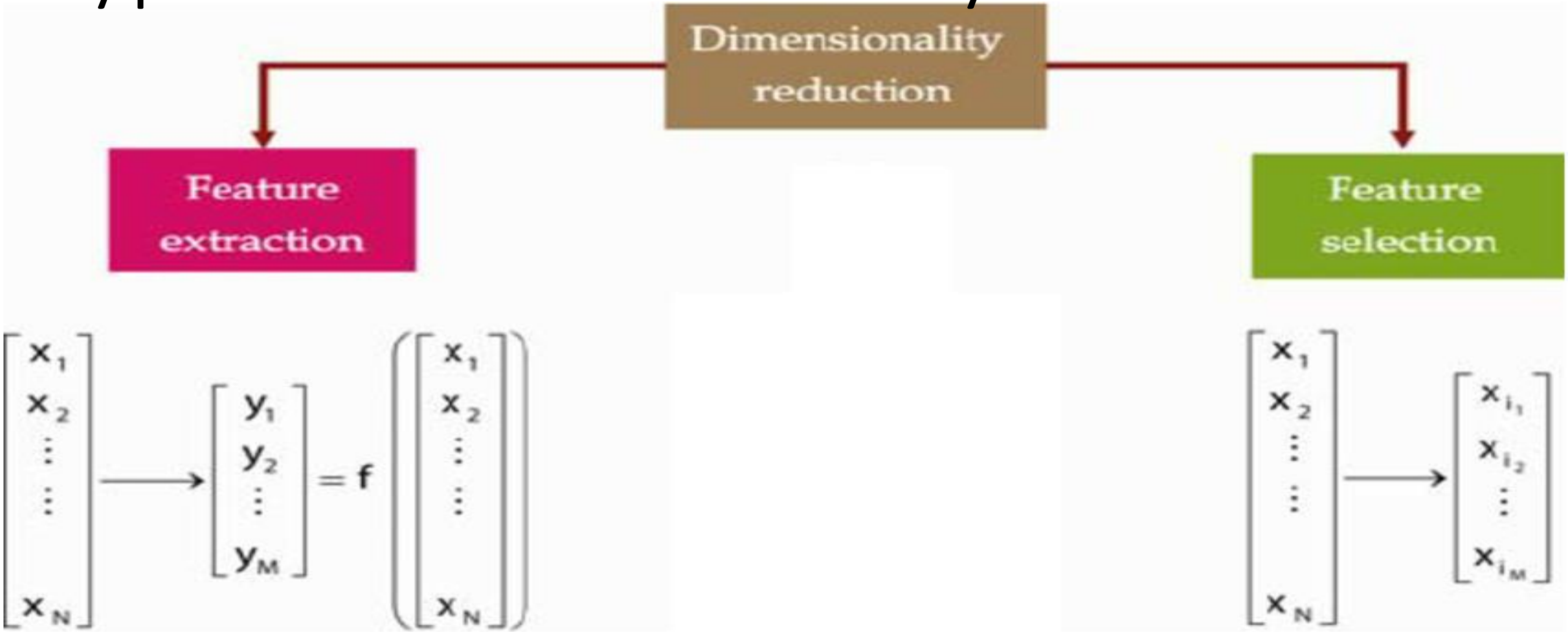
- Discover hidden correlations/topics
  - Example : Words that occur commonly together
- Remove redundant and noisy features
  - Example : Not all words are useful
- Interpretation and visualization
- Less storage space and efficient processing of the data

Applications

- Data Visualization
- Data Compression
- Data Classification
- Trend Analysis
- Factor Analysis
- Noise Reduction

- How many unique “sub-sets” are in the sample?
- How are they similar / different?
- What are the underlying factors that influence the samples?
- Which time / temporal trends are (anti)correlated?
- Which measurements are needed to differentiate?
- How to best present what is “interesting”?
- Which “sub-set” does this new sample rightfully belong?

Types of Dimensionality Reduction



- Principal Component Analysis
- Singular Value Decomposition
- Linear Discriminant Analysis

- Correlation
- Wrapper
- Filter

Takeaways

- Dimensionality reduction refers to reducing the size of data
- Curse of Dimensionality
- Motivation for Dimensionality Reduction
- Applications
- Types of Dimensionality Reduction

How to find the ‘best’ low dimension space that conveys maximum useful information.

Principal Component Analysis

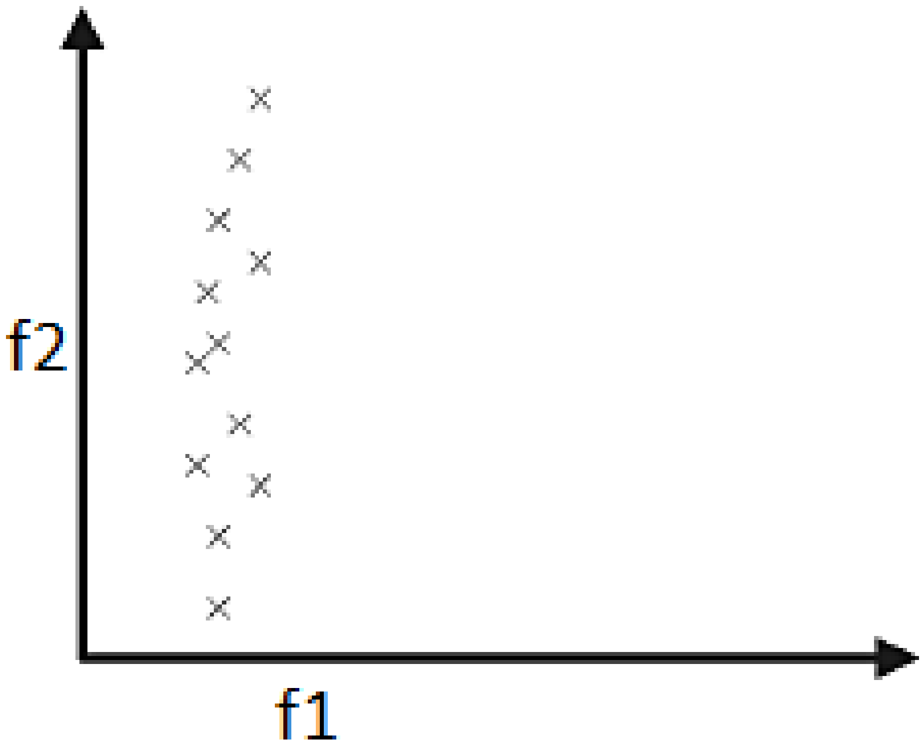
- Given dataset $\mathrm{X} = \mathbb{R}^{n\times d}$, reduce from d-dimension to p-dimension.

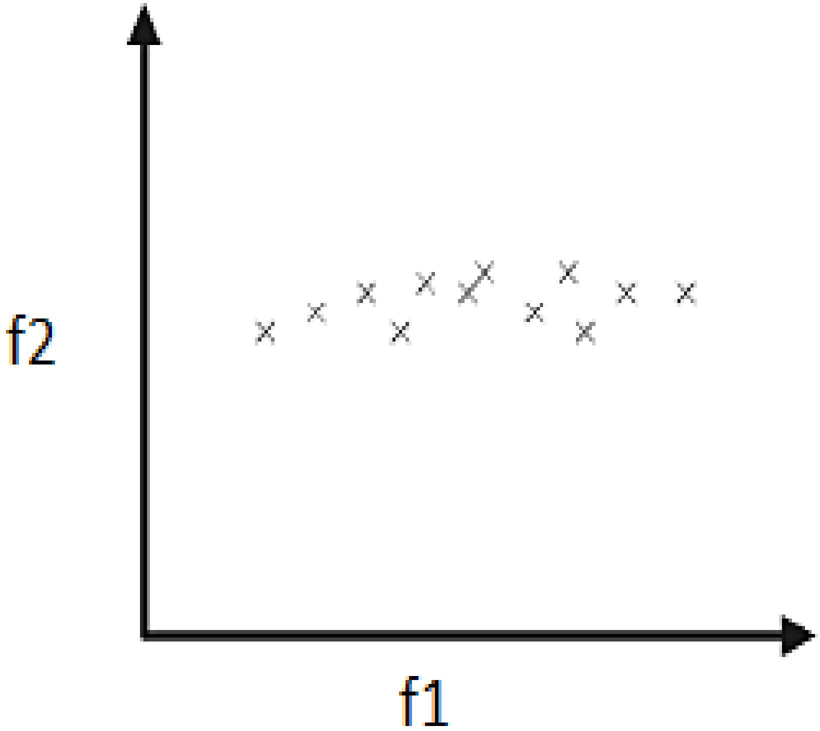
$$\mathrm{X}' = \mathbb{R}^{n\times p},$$

without much loss of information.

- Feature extraction
- d-dim -> p-dim data

High Variance is More information

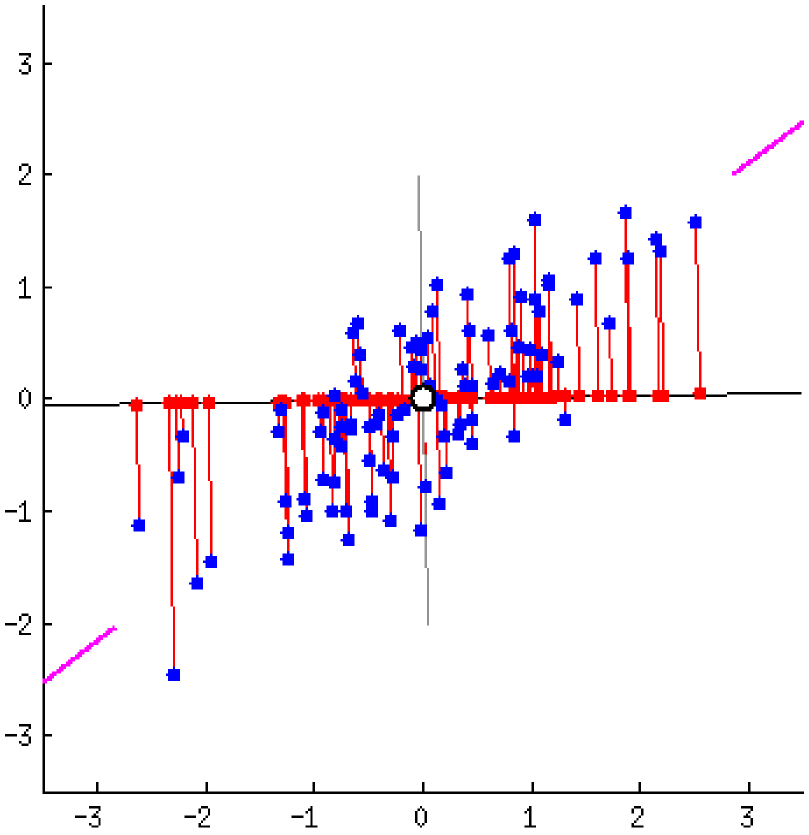




- ➢High spread in f2
- ➢f1 can be dropped

High spread in f1
f2 can be dropped

Axis with maximum variance retains the information the most

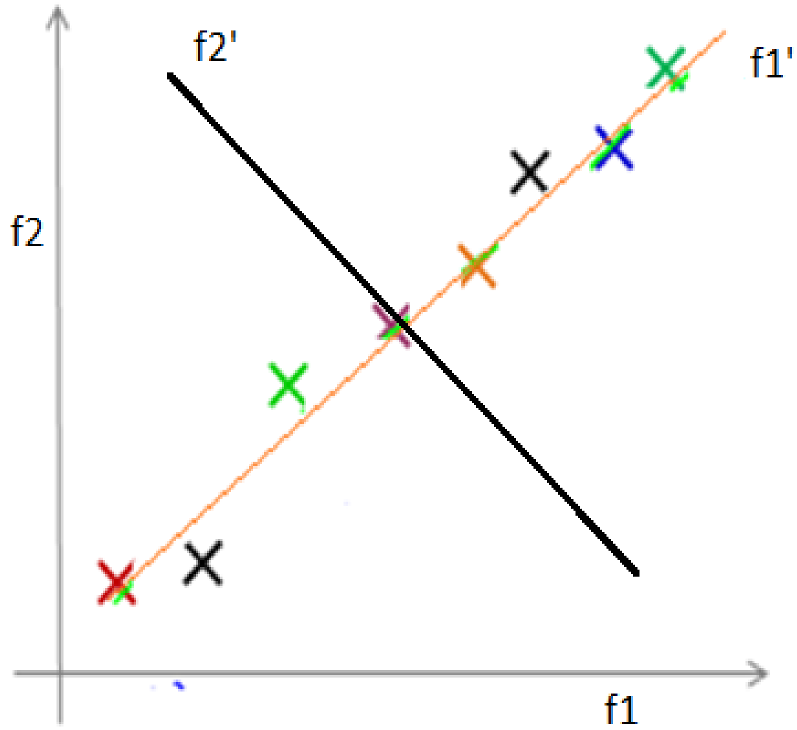


f1’ perpendicular f2’

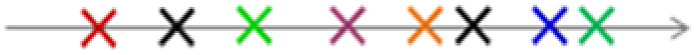
- f1’ retained
- f2’ dropped
- Project xi’s onto f1’

- Objective : Find an axis f1’ such that the variance of xi projected onto f1’ is maximized.

X



X’



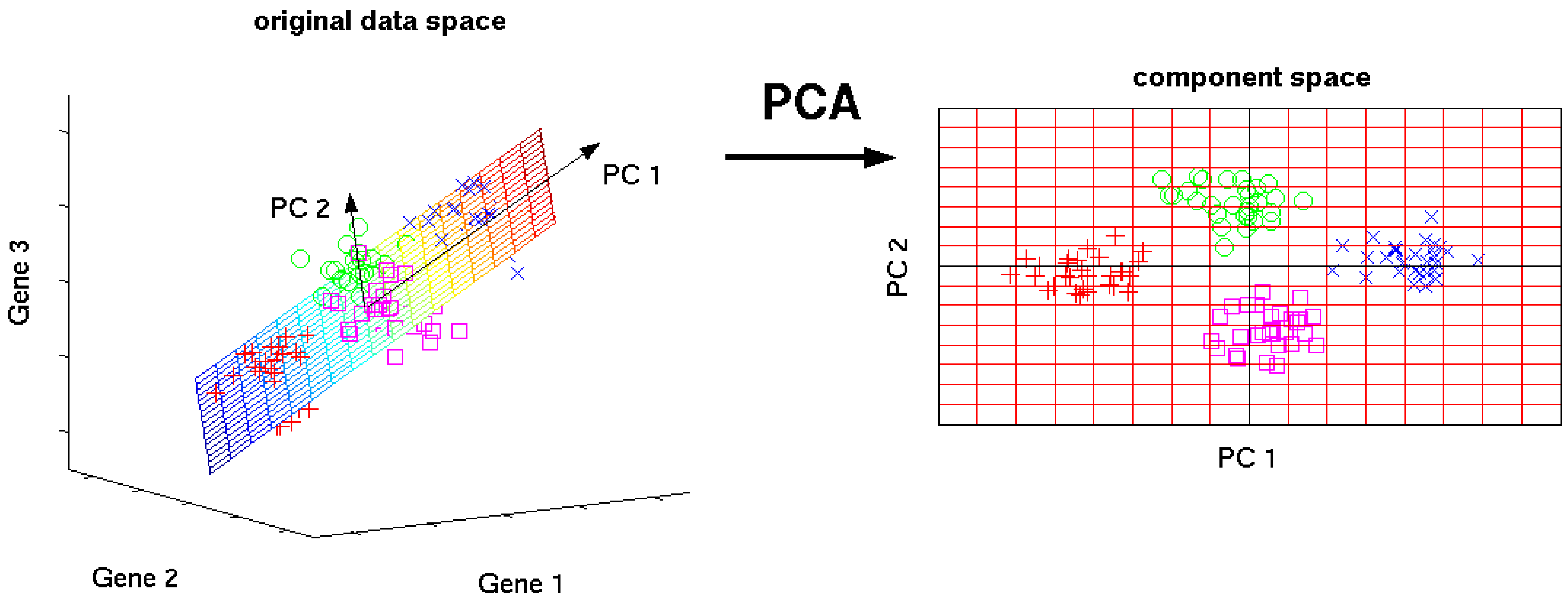
Objective of PCA

- Project data onto a lower dimensional Linear Space such that the variance of the projected data is maximized.

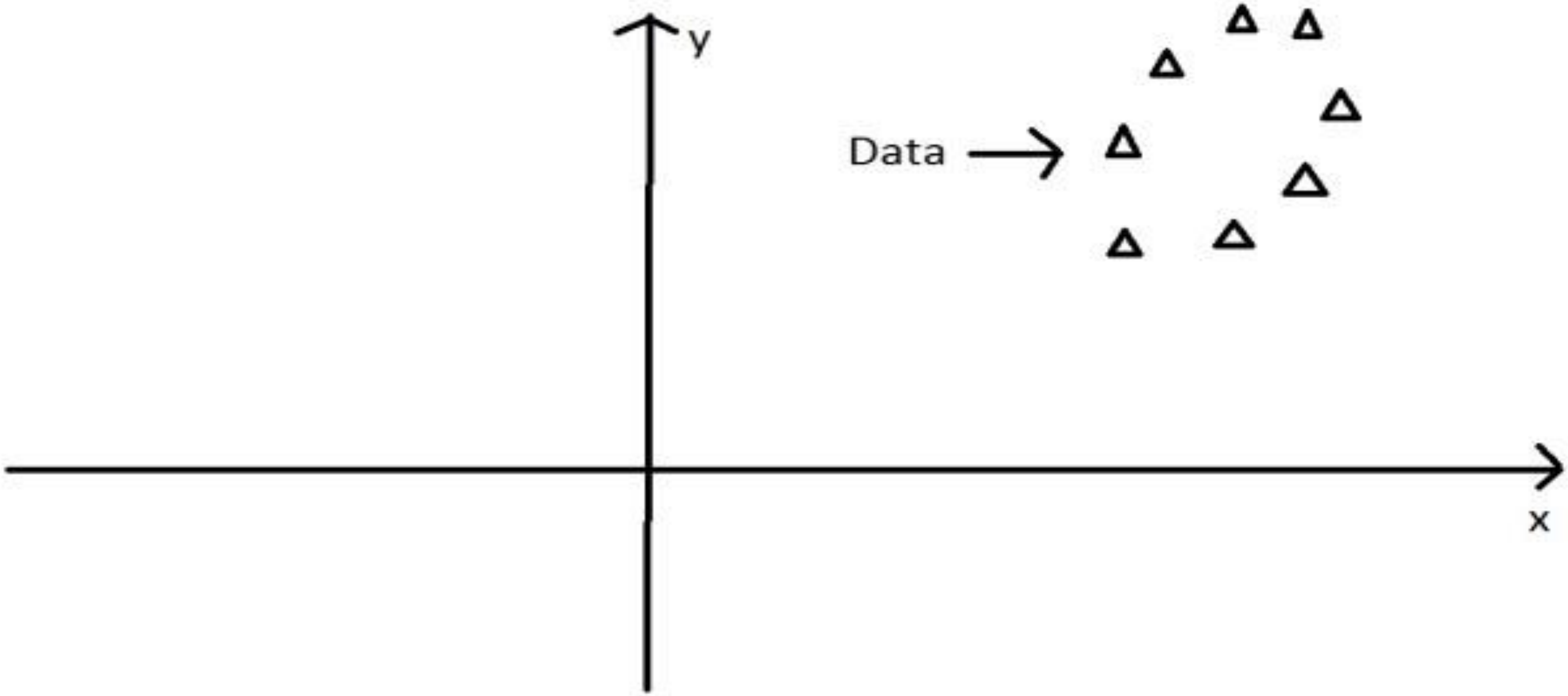
$$Max_{u_1} \frac{1}{N} \sum_{i=1}^{N} (u_1^T x_i)^2$$

- $u_1^T x_i$ is the projection of x_i onto u_1

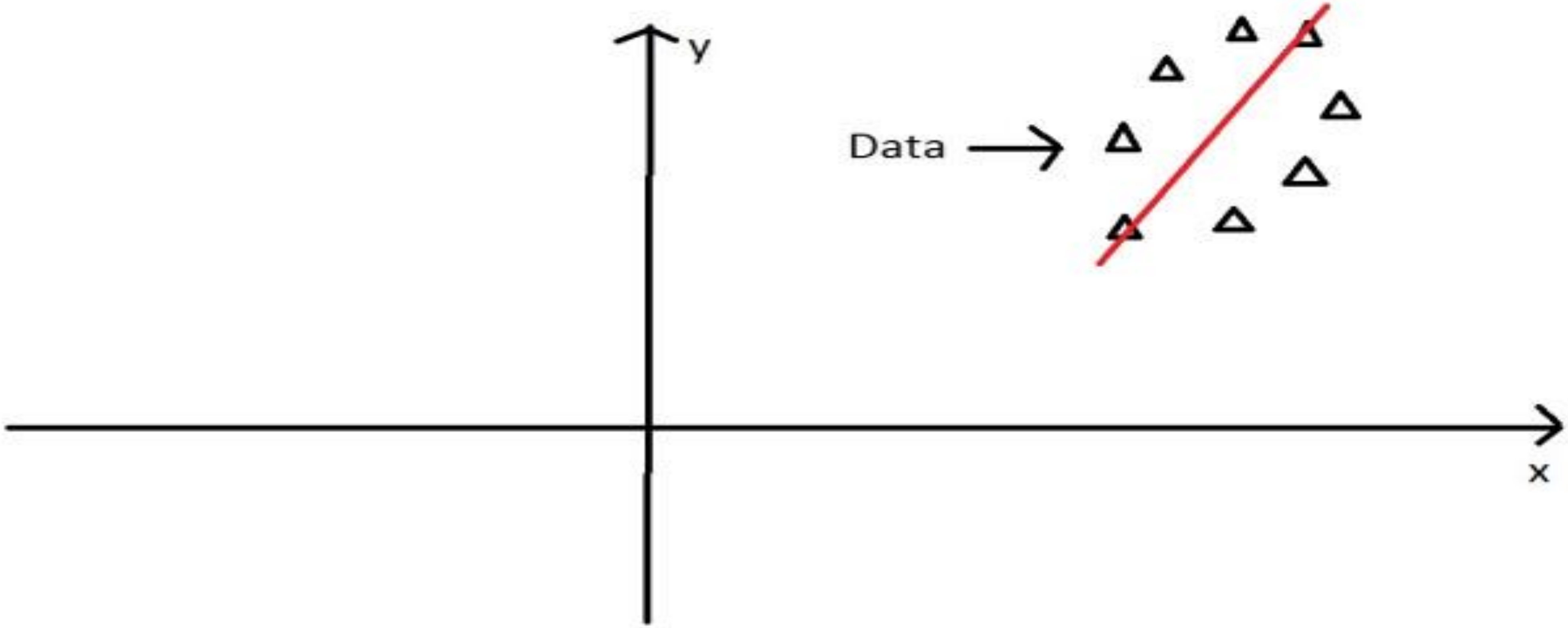
Projection onto 2d-space



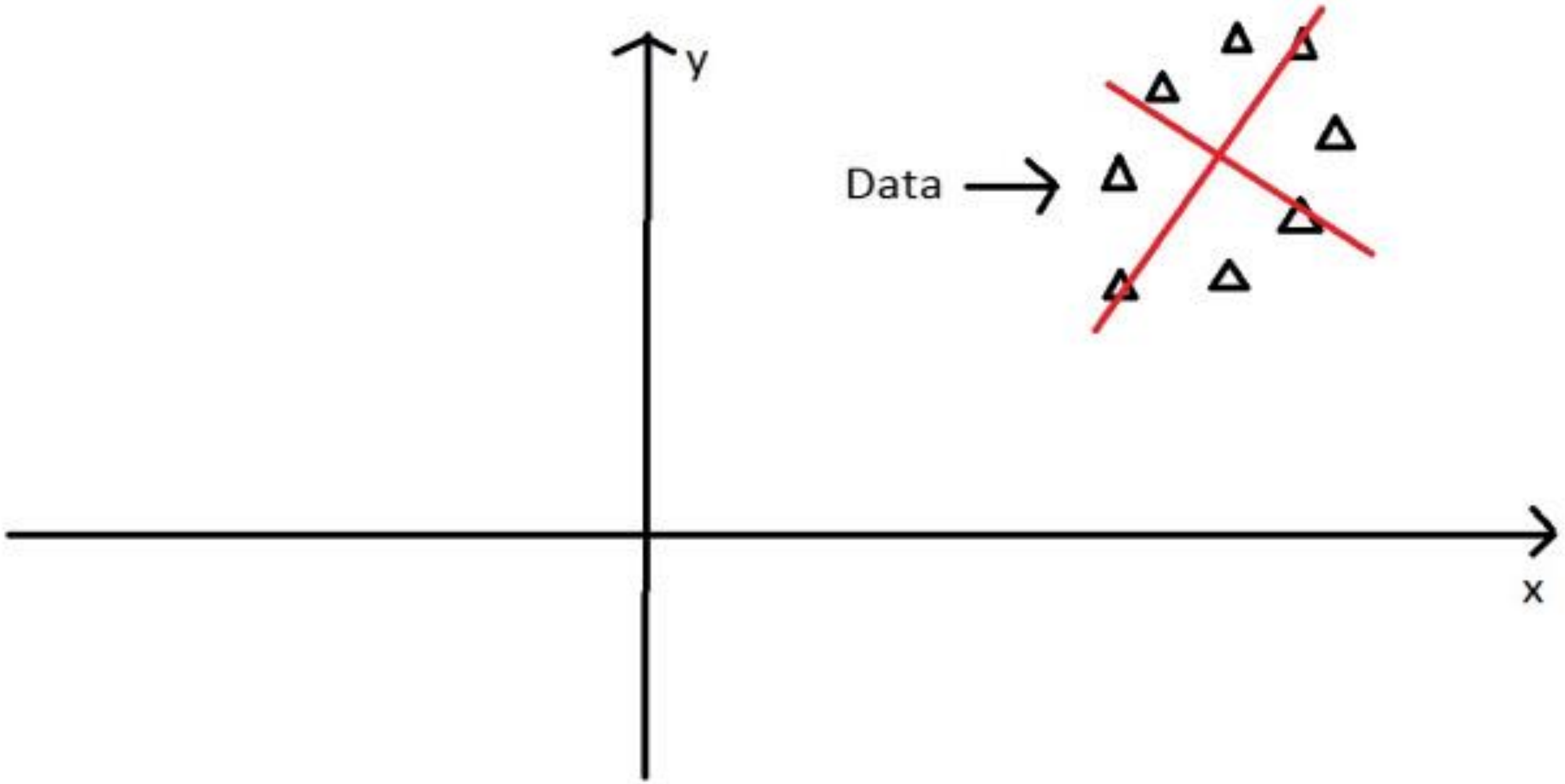
Original space



Direction of maximum variance
First Principal Component

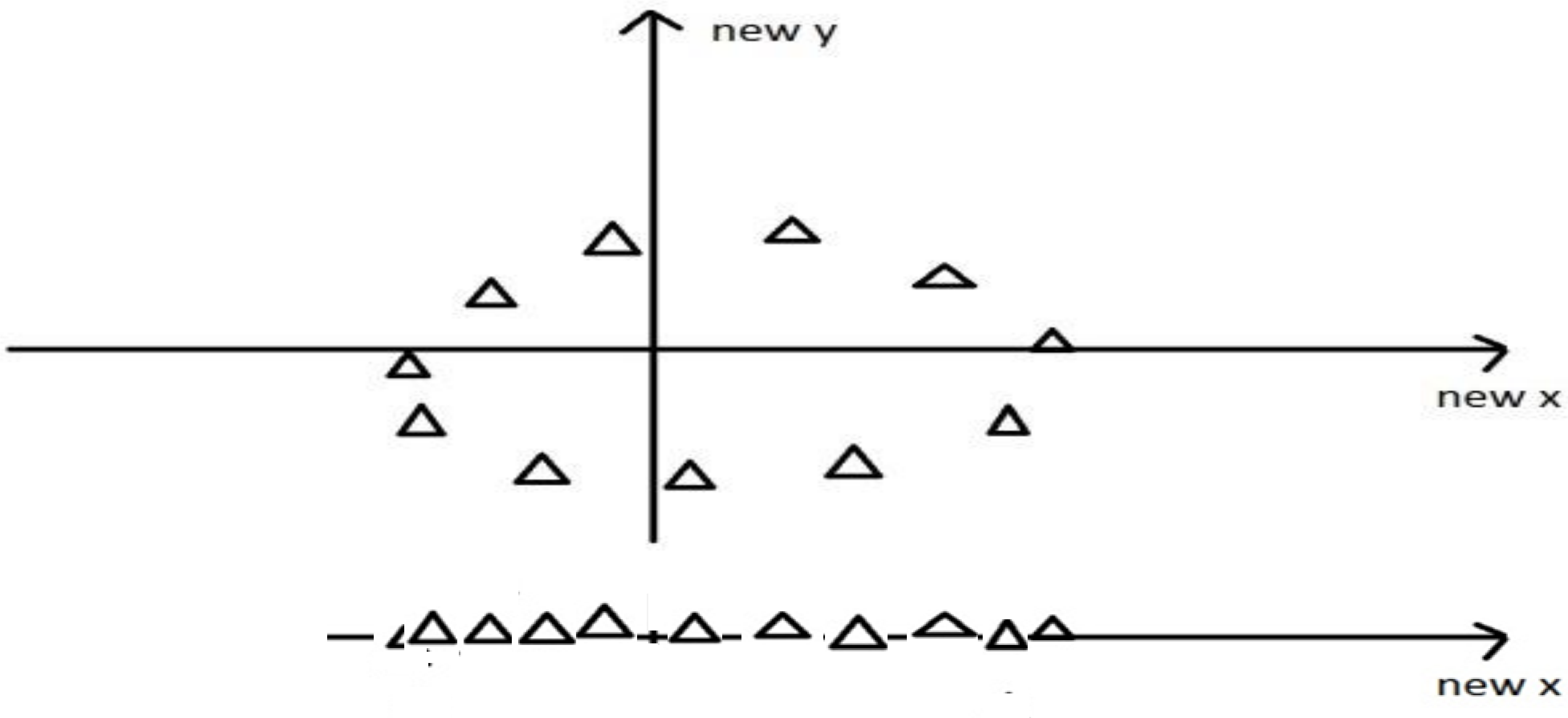


Second maximum variance
Second Principal Component



New space : Transformed Space

Linear combination of original features



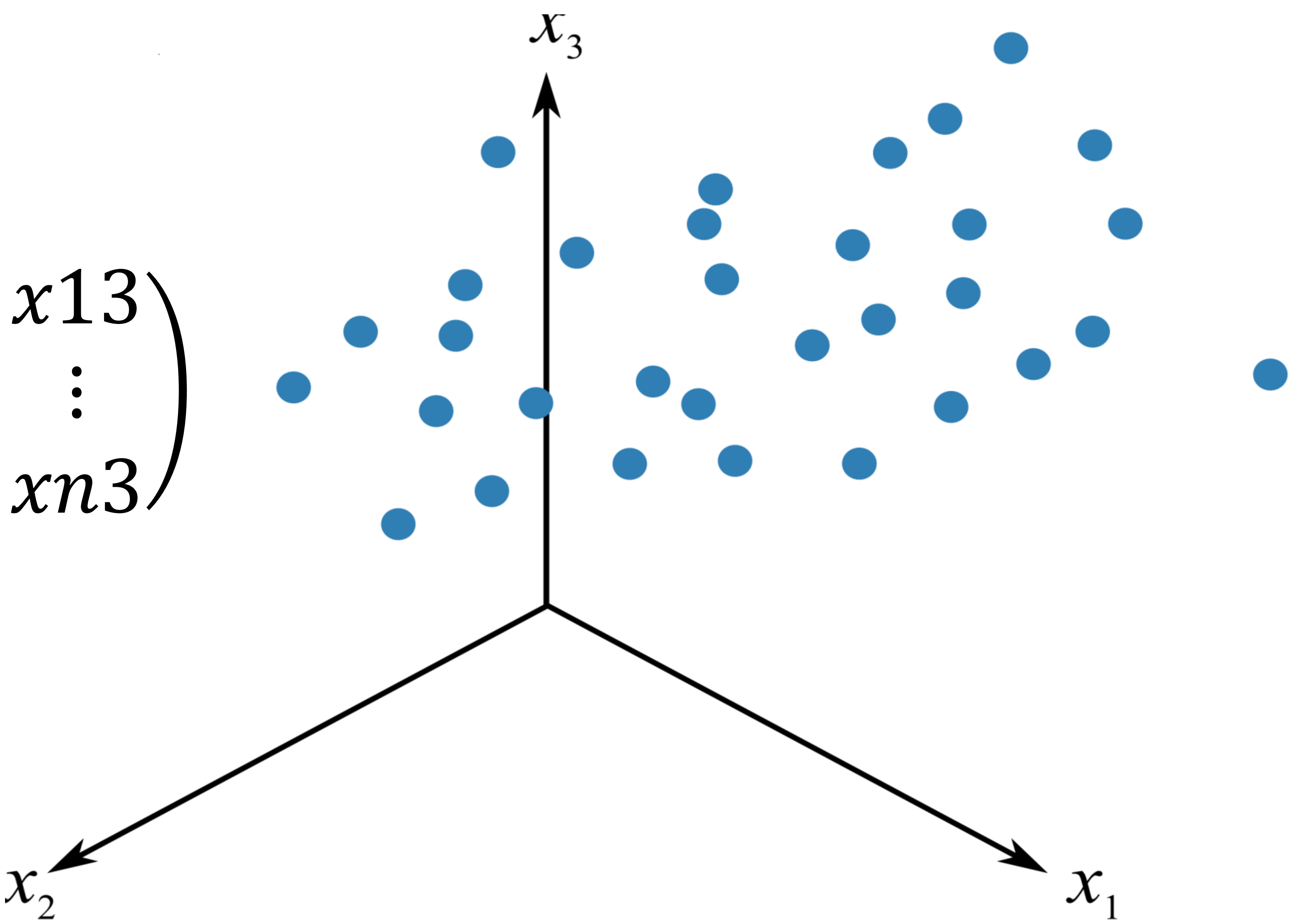
Takeaways

- PCA identifies axes/features in decreasing order of variance.
- Orthogonal axes
- The first PC is the best axis with maximum variance
- Projection onto a subset of axes leads to reduction in the dimensionality of original feature space
- Data transformed in different directions.
- The directions are obtained by some linear combination of the original features.

Geometric Rationale of PCA

3-d data

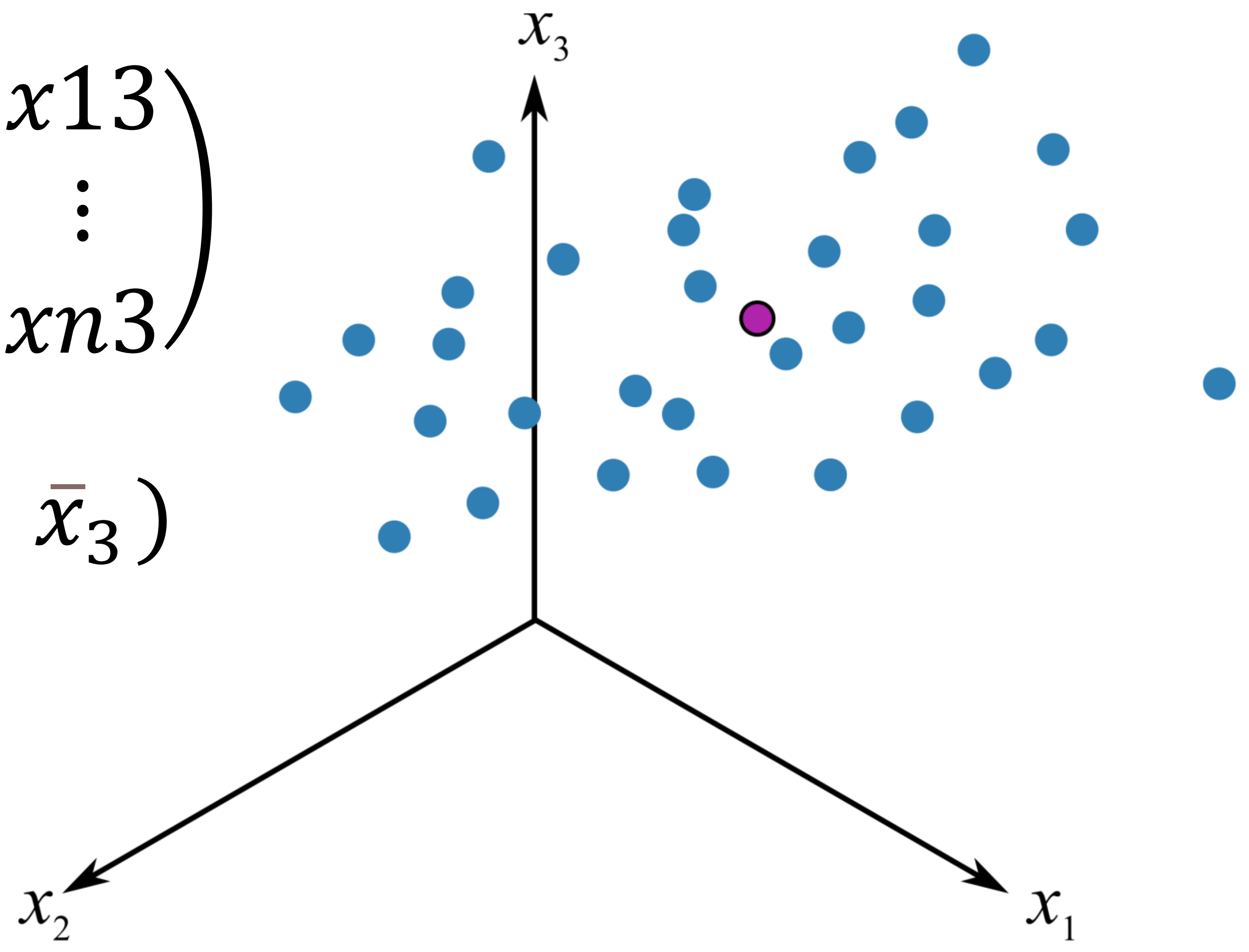
$$\begin{pmatrix} x11 & \cdots & x13 \\ \vdots & \ddots & \vdots \\ xn1 & \cdots & xn3 \end{pmatrix}$$



Step 1 : Centroid of data

$$\begin{pmatrix} x11 & x12 & x13 \\ \vdots & \ddots & \vdots \\ xn1 & xn2 & xn3 \end{pmatrix}$$

Mean vector = $(\bar{x}_1, \quad \bar{x}_2, \quad \bar{x}_3)$



Variance

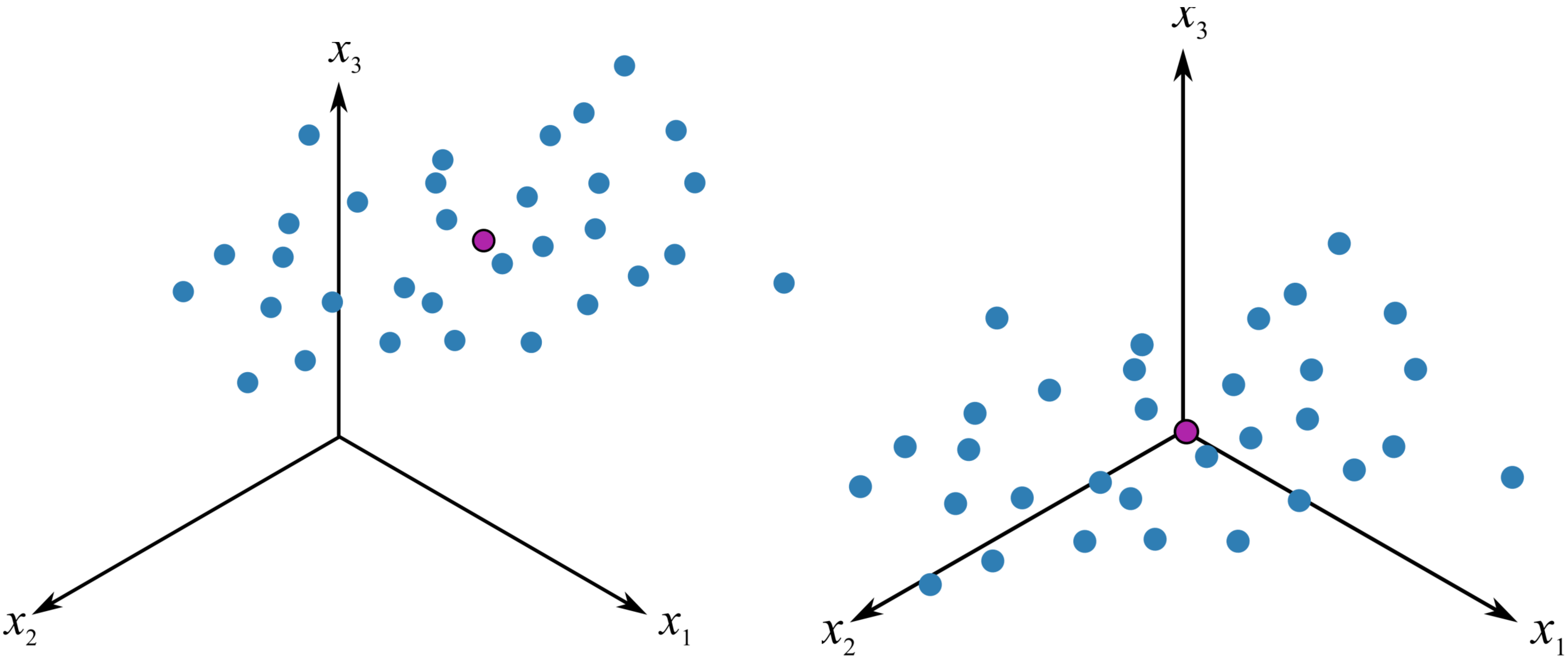
- The variance of each variable is the average squared deviation of its *n* values around the mean of that variable.

$$V_i = \frac{1}{n-1}\sum_{m=1}^{n}\left(X_{im} - \bar{X}_i\right)^2 \qquad SD_i = \sqrt{V_i}$$

- Features with high variances will dominate the principal components

- These problems are generally avoided by standardizing each variable to unit variance and zero mean.

Step 2 : Mean-centered data/Standardization



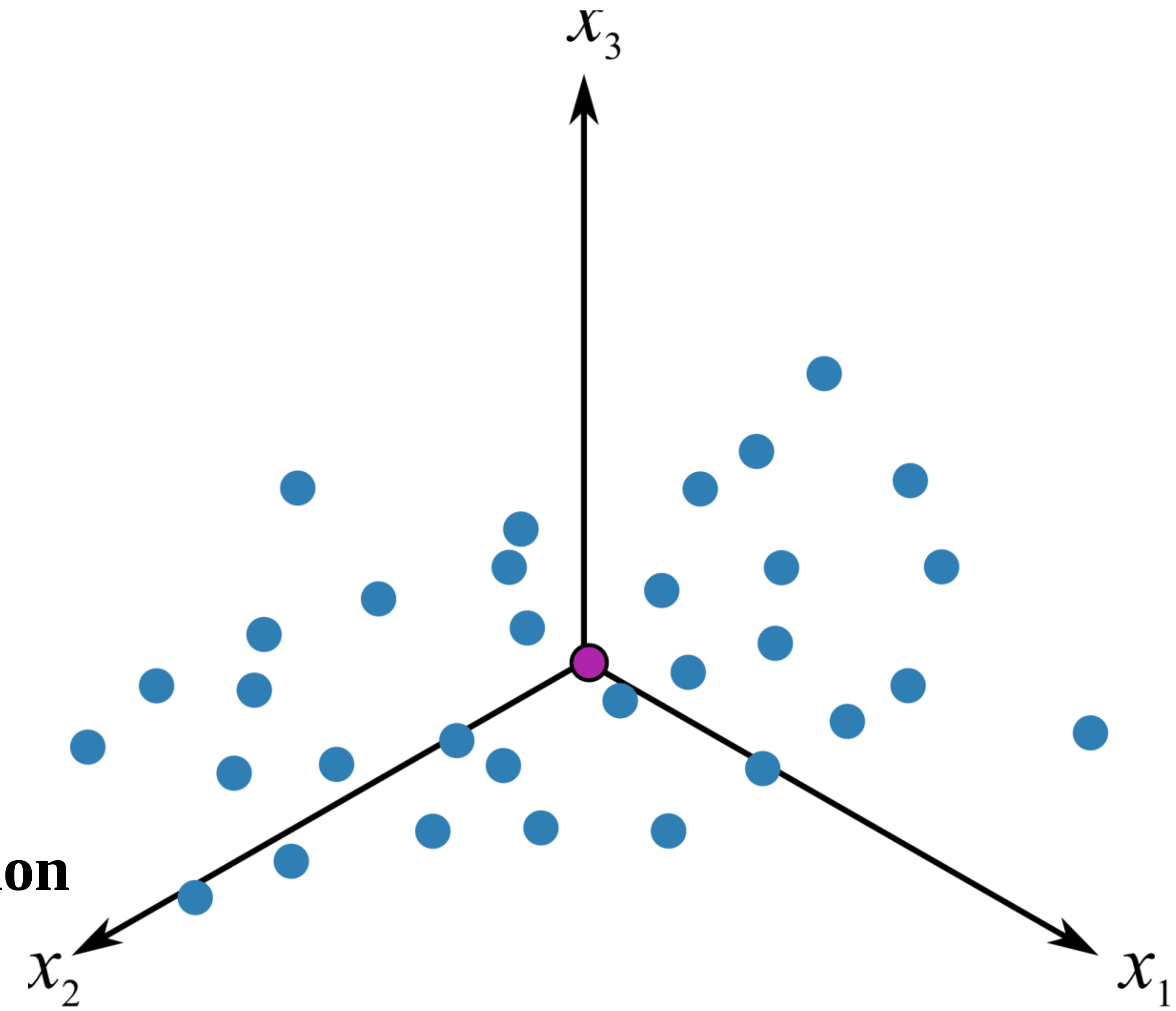
Mean-centered data/Standardization

- Move data to center of coordinate system
- Removes arbitrary bias
- Also scale the data to unit-variance

Mean variable *i*

$$X'_{im} = \frac{\left(X_{im} - \bar{X}_i\right)}{\mathrm{SD}_i}$$

Standard deviation of variable *i*



Step 3 : Find Covariance

➢ $\text{cov}(X_i, X_j) = \frac{1}{n-1}\sum_{\text{k}=1}^{n}(X_{ik} - \bar{X}_i)(X_{jk} - \bar{X}_j)$

➢ For d-dimensional data : dxd matrix

$$\begin{pmatrix} cov(X_1, X_1) & cov(X_1, X_2) \ldots . & cov(X_1, X_d) \\ \vdots & \ddots & \vdots \\ cov(X_d, X_1) & cov(X_d, X_2) & cov(X_d, X_d) \end{pmatrix}$$

➢ Sign of the covariance is important.

- If positive then : the two variables increase or decrease together (correlated)
- if negative then : One increases when the other decreases (Inversely correlated)

Covariance *vs* Correlation

- Covariances between the standardized variables are correlations
- After standardization, each variable has a mean of 0 and a variance of 1.000
- Correlations can be also calculated from the variances and covariances:

Correlation between variables *i* and *j*

$$r_{ij} = \frac{C_{ij}}{\sqrt{V_i V_j}}$$

Covariance of variables *i* and *j*

Variance of variable *i*

Variance of variable *j*

Covariance

- In matrix notation, Covariance is computed as

$$\mathbf{S} = \mathbf{X}'\mathbf{X}$$

- where X is the *n* x d data matrix, with each feature mean-centered (also standardized by SD if using correlations).
- Square, symmetric matrix
- Diagonals are the variances, off-diagonals are the covariances.

	X_1	X_2
X_1	6.6707	3.4170
X_2	3.4170	6.2384

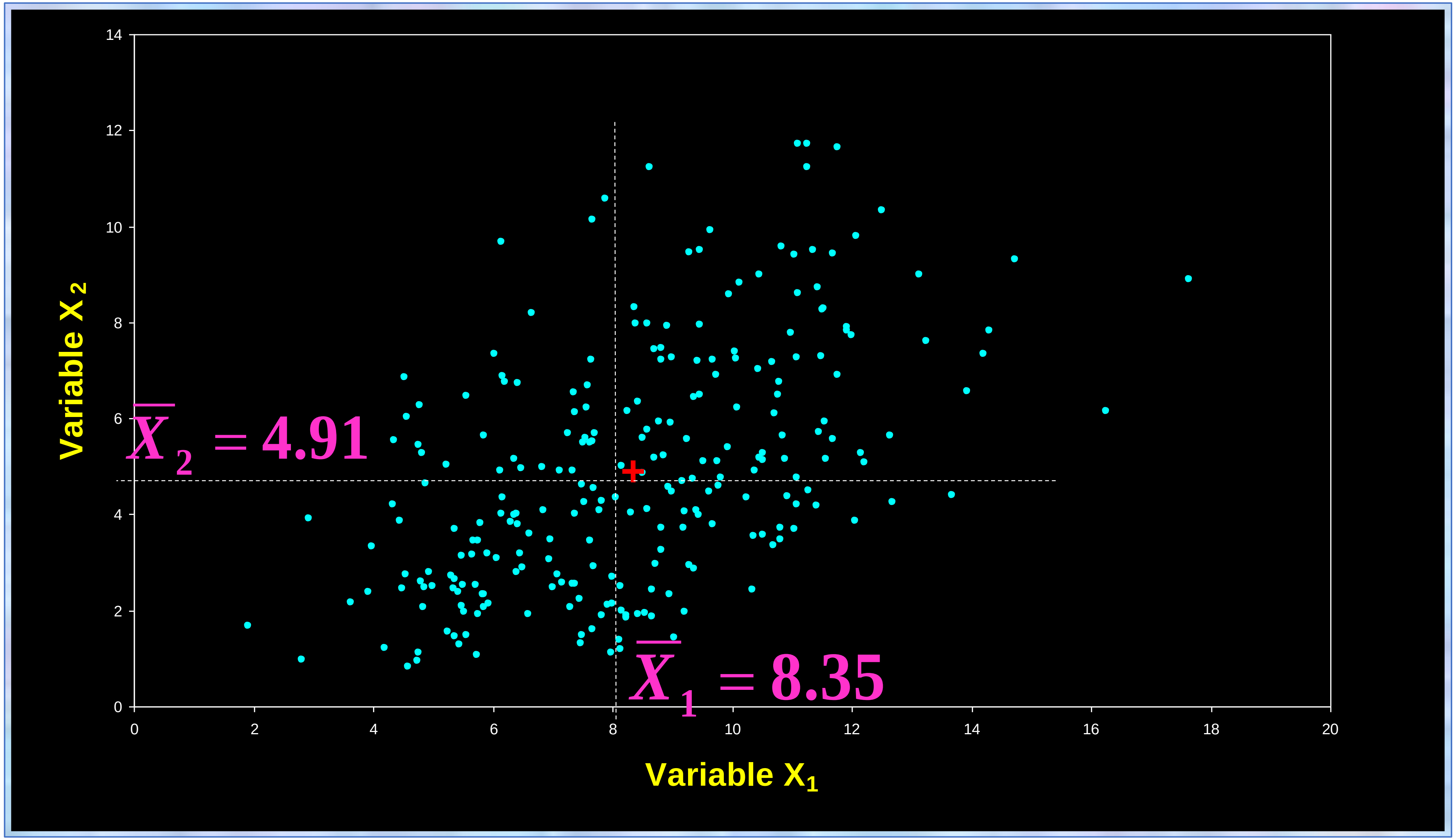
Variance-covariance Matrix

	X_1	X_2
X_1	1.0000	0.5297
X_2	0.5297	1.0000

Correlation Matrix

2D Example of PCA

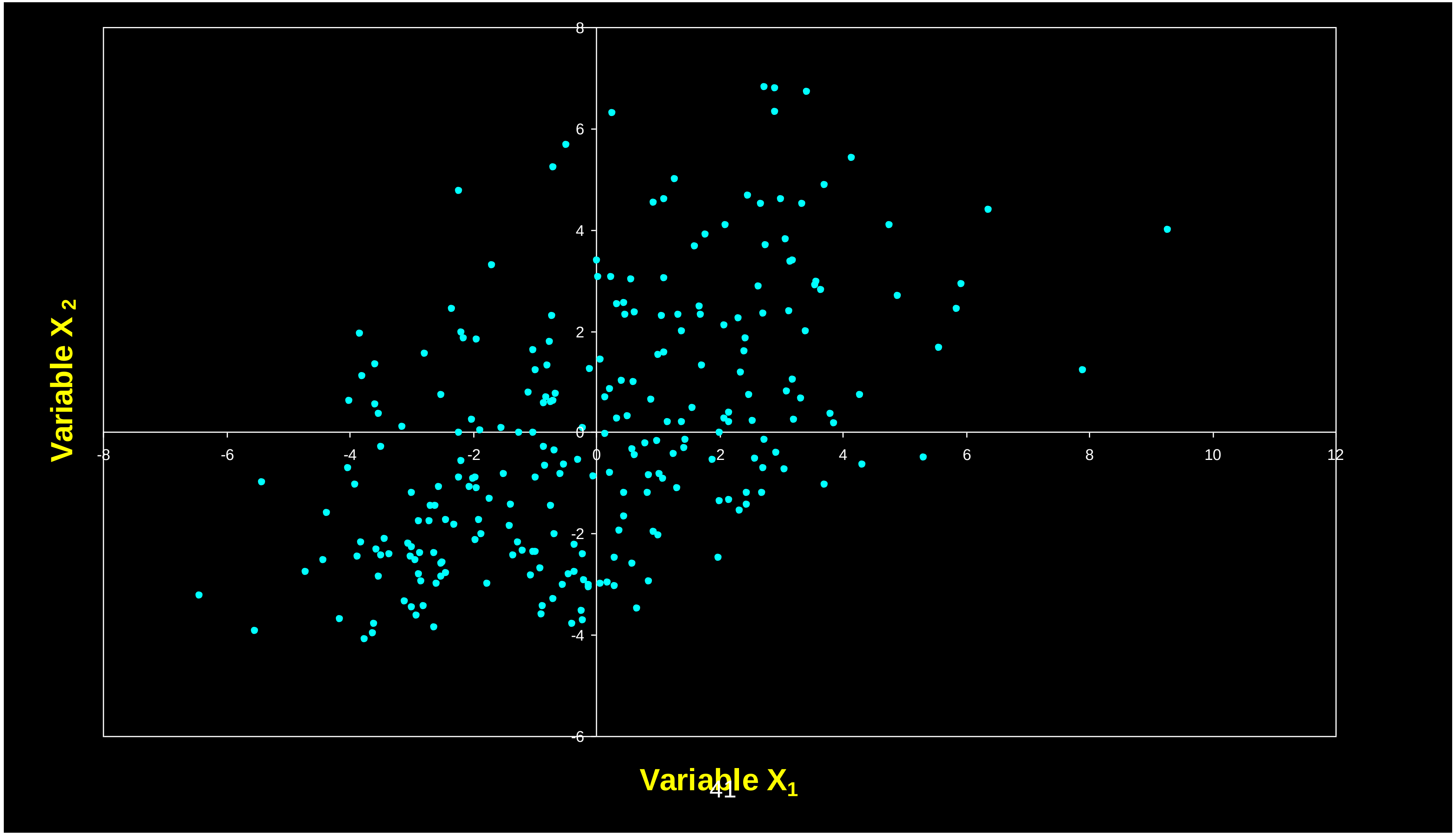
- Variables X_1 and X_2 have positive covariance & each has a similar variance.



$$V_1 = 6.67 \qquad V_2 = 6.24 \qquad C_{1,2} = 3.42$$

Configuration is Centered

- Each variable is adjusted to a mean of zero (by subtracting the mean from each value).



Trace

- Sum of the diagonals of the variance-covariance matrix is called the trace
- Trace represents the total variance in the data
- It is the mean squared Euclidean distance between each object and the centroid in *d*-dimensional space.

	X_1	X_2
X_1	6.6707	3.4170
X_2	3.4170	6.2384

Trace = 12.9091

	X_1	X_2
X_1	1.0000	0.5297
X_2	0.5297	1.0000

Trace = 2.0000

Step 4 : Compute Eigen vectors and Eigen values of Covariance Matrix S

- Finding the principal axes involves eigen analysis of the covariance matrix (S)
- The eigenvalues (latent roots) of S are solutions (λ) to the characteristic equation

$$\left|\mathbf{S} - \lambda \mathbf{I}\right| = 0$$

Eigen Vectors and Eigen Values

- The eigen vector : Direction of axis

- The eigenvalues, λ_1, λ_2, ... λ_d are the variances of the coordinates on each axis

- PC1 : The eigen vector corresponding to highest λ value

	f_1	f_2
f_1	6.6707	3.4170
f_2	3.4170	6.2384

$$\lambda_1 = 9.8783$$
$$\lambda_2 = 3.0308$$

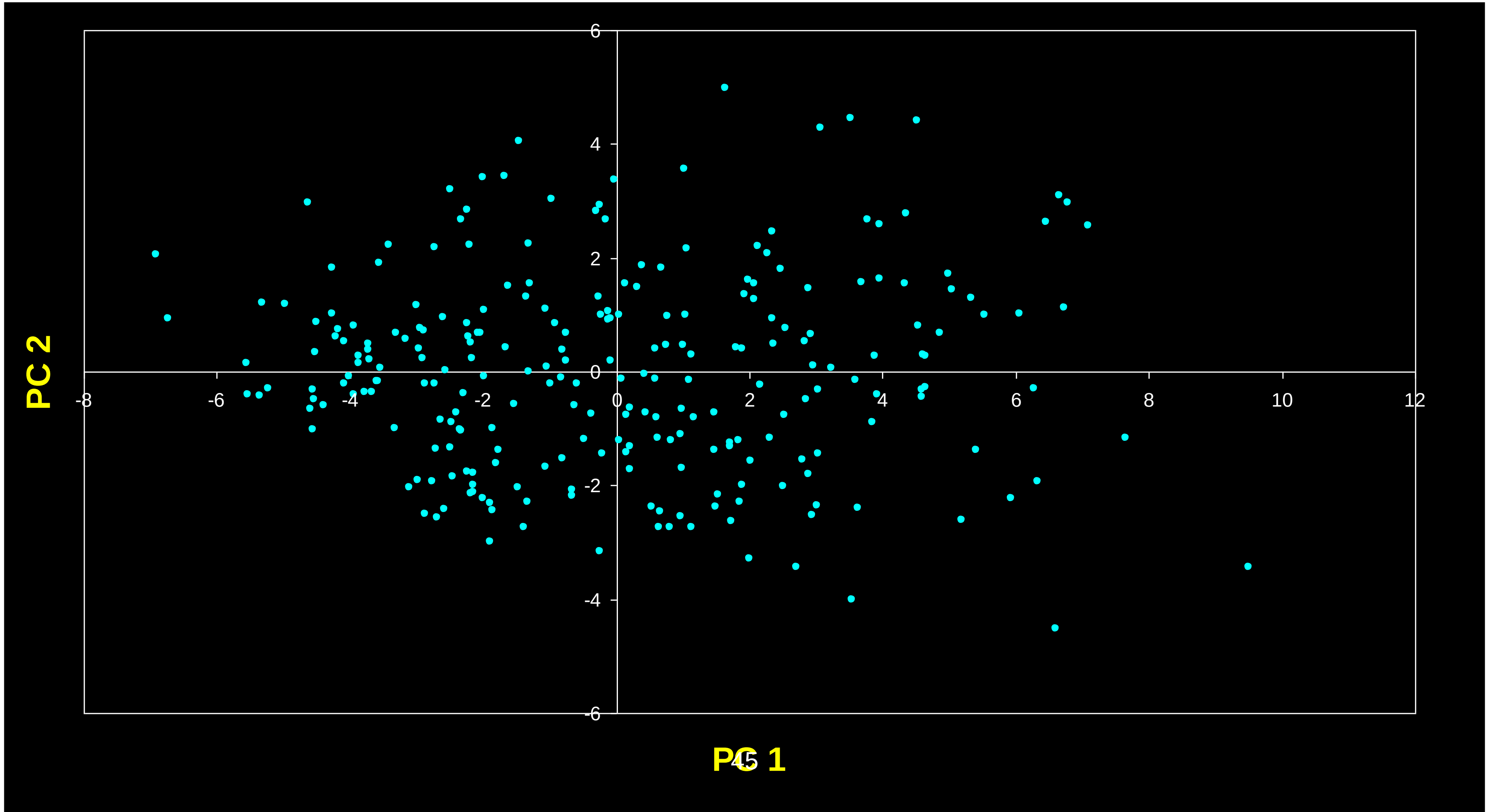
	u_1	u_2
f_1	0.7291	-0.6844
f_2	0.6844	0.7291

Trace = 12.9091

Note: $\lambda_1 + \lambda_2 = 12.9091$

Principal Components are Computed

- PC 1 has the highest possible variance (9.88)
- PC 2 has a variance of 3.03
- PC 1 and PC 2 have zero covariance.



Eigen vectors as principal components

- Each eigenvector consists of d values which represent the "contribution" of each variable to the principal component axis
- Eigenvectors are uncorrelated (orthogonal)

Eigenvectors

	u_1	u_2
f_1	0.7291	-0.6844
f_2	0.6844	0.7291

u1.u2 = 0.7291*(-0.6844) + 0.6844*0.7291 = 0

Transformed space using PCs

$$\begin{pmatrix} x11 & x12 \\ x21 & x22 \end{pmatrix} \begin{pmatrix} 0.7291 & -0.6844 \\ 0.6844 & 0.7291 \end{pmatrix} = Z^*$$

nxd dxd nxd

- f1’ = 0.7291*x11 + 0.6844*x12
- f2’ = -0.6844*x11 + 0.7291*x12

Step 5 : Dimensionality Reduction using PCs

$$\begin{pmatrix} x11 & x12 \\ x21 & x22 \end{pmatrix} \begin{pmatrix} 0.7291 & \cancel{-0.6844} \\ 0.6844 & \cancel{0.7291} \end{pmatrix}$$

$$\underset{\text{nxd}}{\begin{pmatrix} x11 & x12 \\ x21 & x22 \end{pmatrix}} \underset{\text{dxp}}{\begin{pmatrix} 0.7291 \\ 0.6844 \end{pmatrix}} = \underset{\text{nxp}}{Z^*}$$

Transformed Feature Space

- Coordinates of each object *i* on the k^{th} principal axis, known as the scores on PC *k*, are computed as

$$z_{ik} = u_{1k}x_{i1} + u_{2k}x_{i2} + \cdots + u_{dk}x_{id}$$

- where Z is the *n* x *k* matrix of PC scores,
- X is the *n* x *d* centered data matrix and
- U is the d x *k* matrix of eigenvectors.

PCA steps

Input : X matrix of size n x d ; n samples, d features

Step 1 : Mean of each feature value

Step 2 : Mean centering of X

$$X'_{im} = \frac{\left(X_{im} - \bar{X}_i\right)}{\mathrm{SD}_i}$$

Mean variable *i*

Standard deviation of variable *i*

Step 3 : Compute Covariance $\mathbf{S} = \mathbf{X}'\mathbf{X}$

Step 4 : Find eigen vectors, eigen values from S

Arrange eigen vectors in descending order of eigen values

$$\lambda_1 < \lambda_2 < \ldots < \lambda_d$$

Step 5 : Retain the first p eigen vectors : **U** matrix with n x p size

X.U gives the p-dimensional data

Advantages

- Removes correlation amongst the features in original data space
- Principal Components are independent of one another. There is no correlation among them.
- Most effective transformation of existing attributes through a linear transformation technique
- Dimensionality Reduction
- Preprocessing data
- Reduces Overfitting

Limitations

- Independent variables become less interpretable
- Data standardization is must before PCA

References

- https://www.statisticshowto.com/dimensionality/
- https://builtin.com/data-science/step-step-explanation-principal-component-analysis
- http://docs.netapp.com/ontap-9/index.jsp?topic=%2Fcom.netapp.doc.onc-sm-help-930%2FGUID-B0C5894F-6D20-4210-A031-D5CD39C7A029.html
- https://medium.com/@bishikh90/geometrical-and-mathematical-interpretation-principal-component-analysis-52f39a924b40
- https://learnche.org/pid/latent-variable-modelling/principal-component-analysis/geometric-explanation-of-pca

PCA Steps

- Step 1 Get some data
- Step2 Subtract the mean – produces a data set whose mean is zero

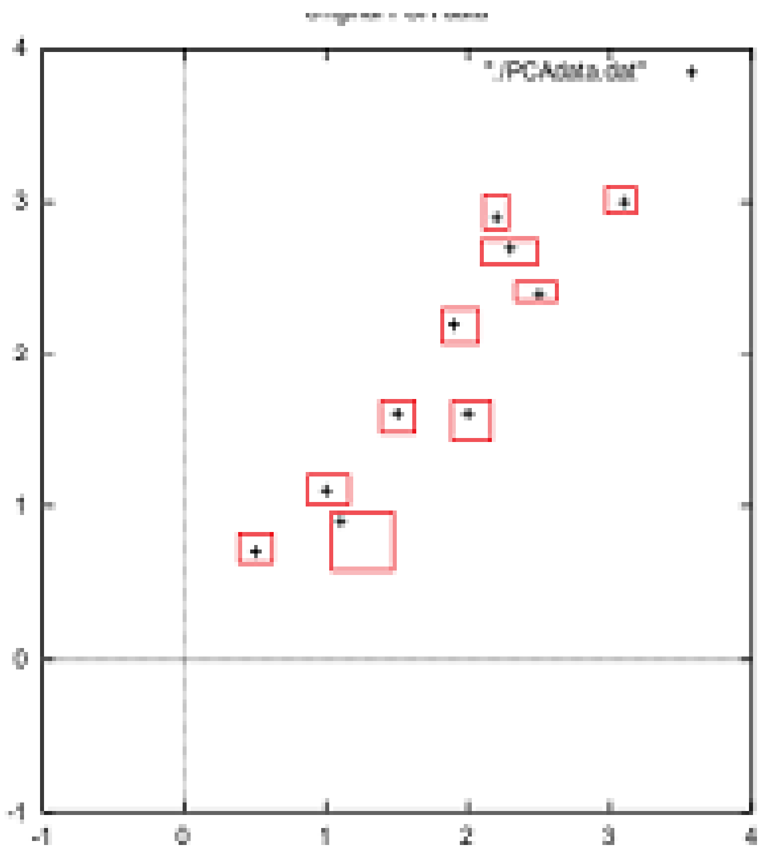
Data =

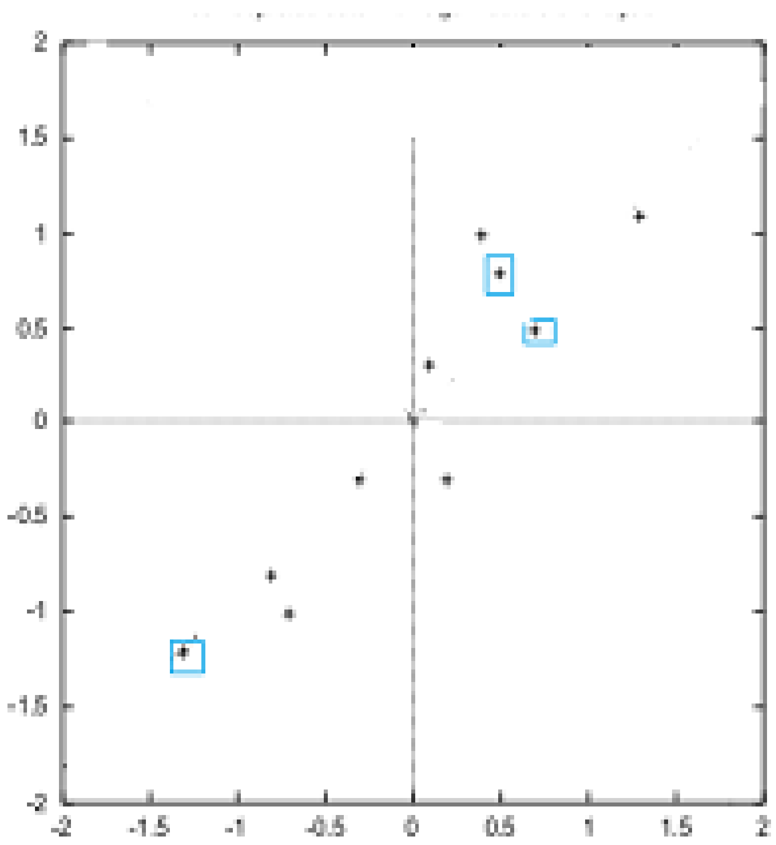
x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

DataAdjust =

x	y
.69	.49
-1.31	-1.21
.39	.99
.09	.29
1.29	1.09
.49	.79
.19	-.31
-.81	-.81
-.31	-.31
-.71	-1.01

PCA Steps





Mean adjusted data

PCA Steps

- Step3: Calculate the covariance matrix
- non-diagonal elements in this covariance matrix are both the variable increase together

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

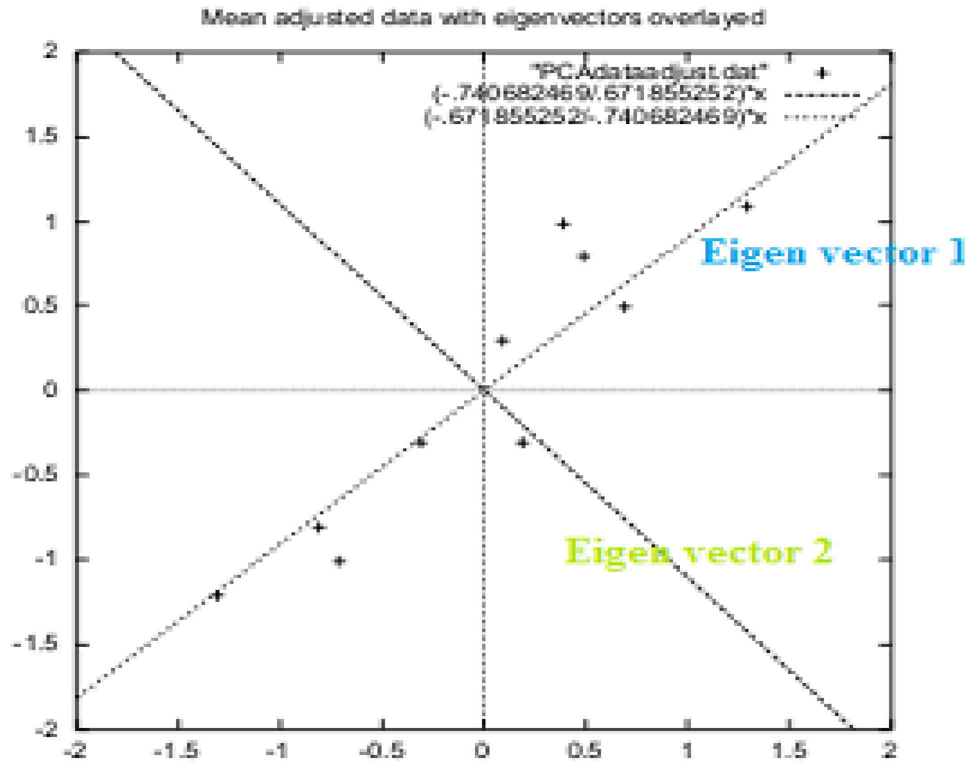
PCA Steps

- Step4:Calculate the eigen vectors and eigen values of the covariance matrix

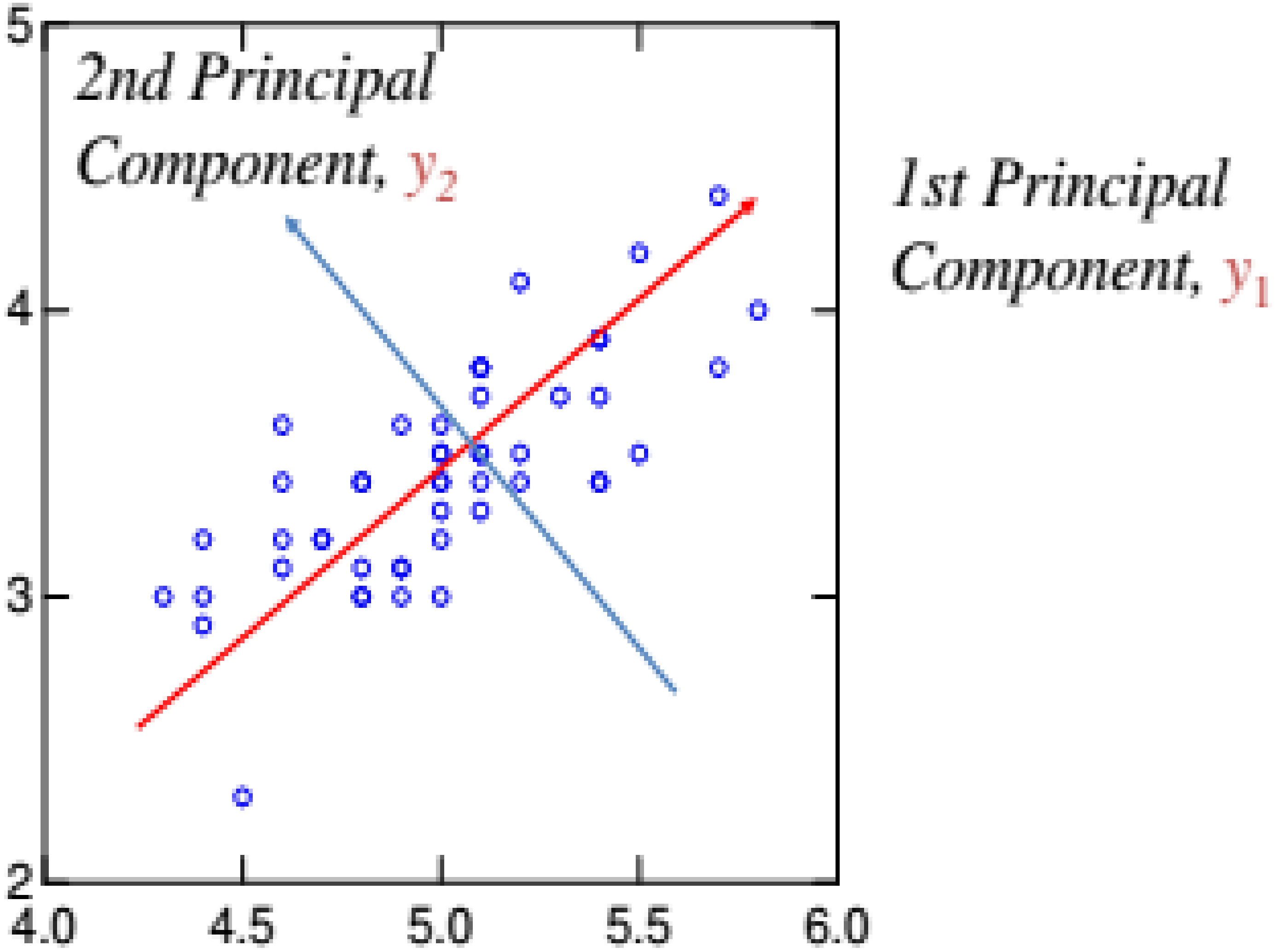
$$eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$eigenvectors = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$

PCA Steps



line of best fit


2nd Principal
Component, y_2
1st Principal
Component, y_1
5
4
3
2
4.0
4.5
5.0
5.5
6.0

PCA Steps

- Step5:Choosing components and forming a feature vector
- Eigen vector with the highest eigen value is the principle compone
- Order by eigen value, highest to lowest gives the components in order of significance

$$FeatureVector = (eig_1\ eig_2\ eig_3\ ....\ eig_n)$$

PCA Steps

- Step6:Deriving the new dataset

$$FinalData = RowFeatureVector \times RowDataAcjust,$$

Transformed Data=

x	y
-.827970186	-.175115307
1.77758033	.142857227
-.992197494	.384374989
-.274210416	.130417207
-1.67580142	-.209498461
-.912949103	.175282444
.0991094375	-.349824698
1.14457216	.0464172582
.438046137	.0177646297
1.22382056	-.162675287

Reconstruction of original Data

X

-.827970186

1.77758033

-.992197494

-.274210416

-1.67580142

-.912949103

.0991094375

1.14457216

.438046137

1.22382056

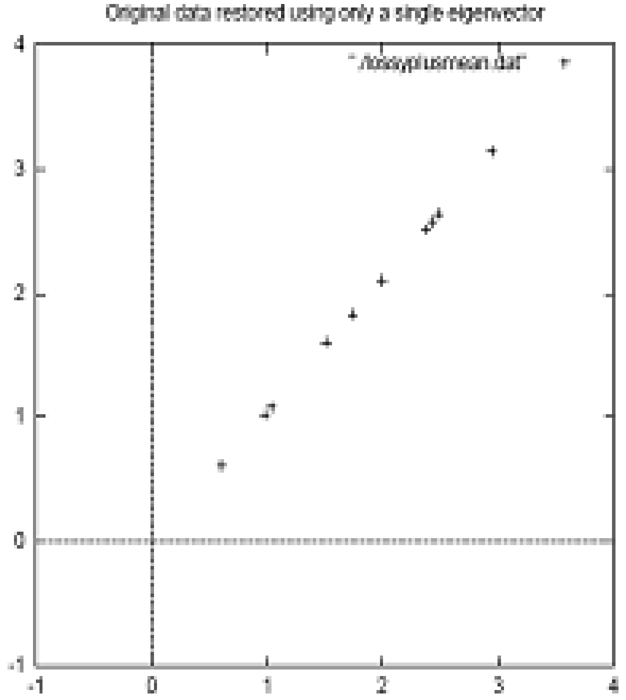


Figure 3.5: The reconstruction from the data that was derived using only a single eigenvector