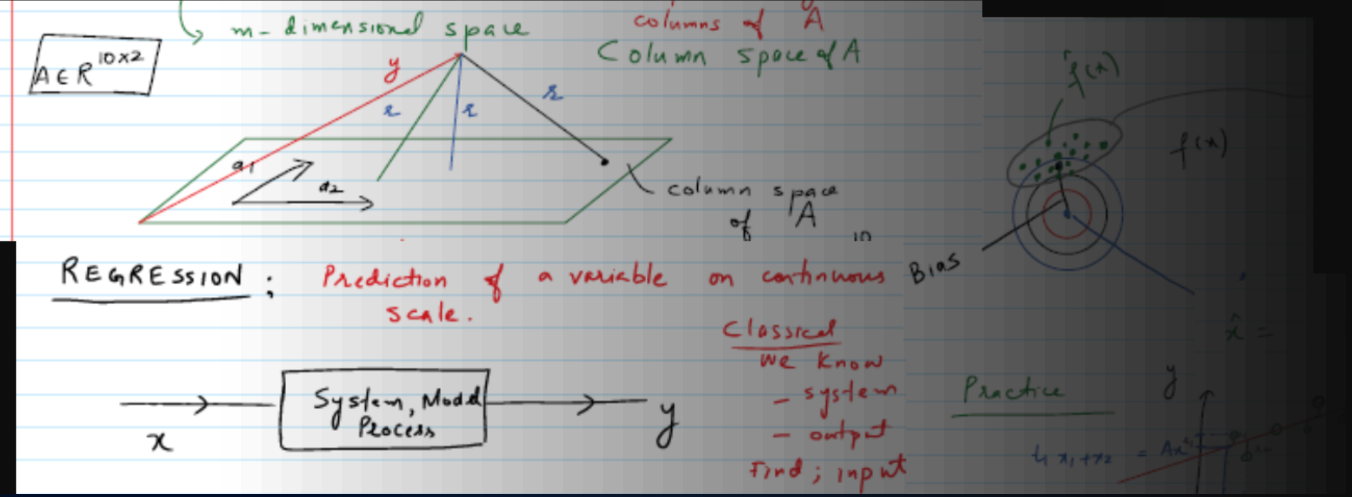


Machine Learning

EE514 – CS535

Dimensionality Reduction: Feature Selection and Feature Extraction (PCA)

Zubair Khalid

School of Science and Engineering
Lahore University of Management Sciences

https://www.zubairkhalid.org/ee514_2021.html

LUMS
A Not-for-Profit University

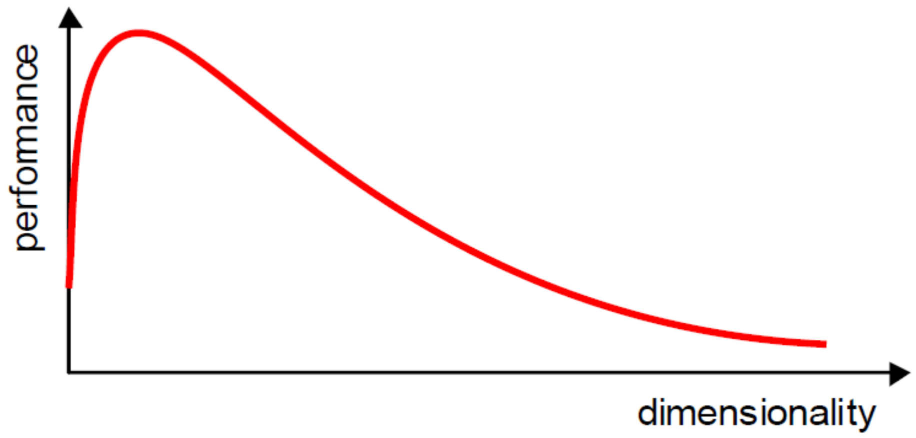
Outline

- Dimensionality Reduction
- Feature Selection
- Feature Extraction – PCA

Dimensionality Reduction

Why?

- *Increasing the number of inputs or features does not always improve accuracy of classification.*
- *Performance of classifier may degrade with the inclusion of irrelevant or redundant features.*
- *Curse of dimensionality; "Intrinsic" dimensionality of the data may be smaller than the actual size of the data.*



Benefits:

- *Improve the classification performance.*
- *Improve learning efficiency and enable faster classification.*
- *Better understanding of the underlying process mapping inputs to output.*

Dimensionality Reduction

Feature Selection and Feature Extraction:

Given a set of features, reduce the number of features such that "the learning ability of the classifier" is maximized.

$$\mathbf{x} = [x_1, x_2, \ldots, x_d]$$

Feature Selection:

Select a subset of the existing features.

$$\mathbf{x} = [x_1, x_2, \ldots, x_d]$$

$$\downarrow$$

$$\mathbf{z} = [x_{i_1}, x_{i_2}, \ldots, x_{i_k}]$$

Feature Extraction:

Transform existing features to obtain a set of new features using some mapping function.

$$\mathbf{x} = [x_1, x_2, \ldots, x_d]$$

$$\downarrow \quad \mathbf{z} = f(\mathbf{x})$$

$$\mathbf{z} = [z_1, z_2, \ldots, z_k]$$

$$k \ll d$$

Dimensionality Reduction

Feature Selection:

Select a subset of the *existing* features.

$$\mathbf{x} = [x_1, x_2, \ldots, x_d]$$

$$\Downarrow$$

$$\mathbf{z} = [x_{i_1}, x_{i_2}, \ldots, x_{i_k}]$$

Select the features in the subset that either improves classification accuracy or maintain same accuracy.

How many subsets do we have?

How do we choose this subset?

Dimensionality Reduction

Feature Selection:

Example:

$$\mathbf{x} = [x_1, x_2, x_3, x_4, x_5] \quad y$$

x_1	x_2	x_3	x_4	x_5	y
0	0	1	0	1	0
0	1	0	0	1	1
1	0	1	0	1	1
1	1	0	0	1	1
0	0	1	1	0	0
0	1	0	1	0	1
1	0	1	1	0	1
1	1	0	1	0	1

* Source: A tutorial on genomics by Yu (2004).

Data set:

- Five Boolean features
- $y = x_1$ (or) x_2
- x_3 = (not) x_2
- x_4 = (not) x_5

Optimal subset:

$\{x_1, x_2\}$ or $\{x_1, x_3\}$

Optimization in space of all feature subsets would have

$$2^d \text{ possibilities}$$

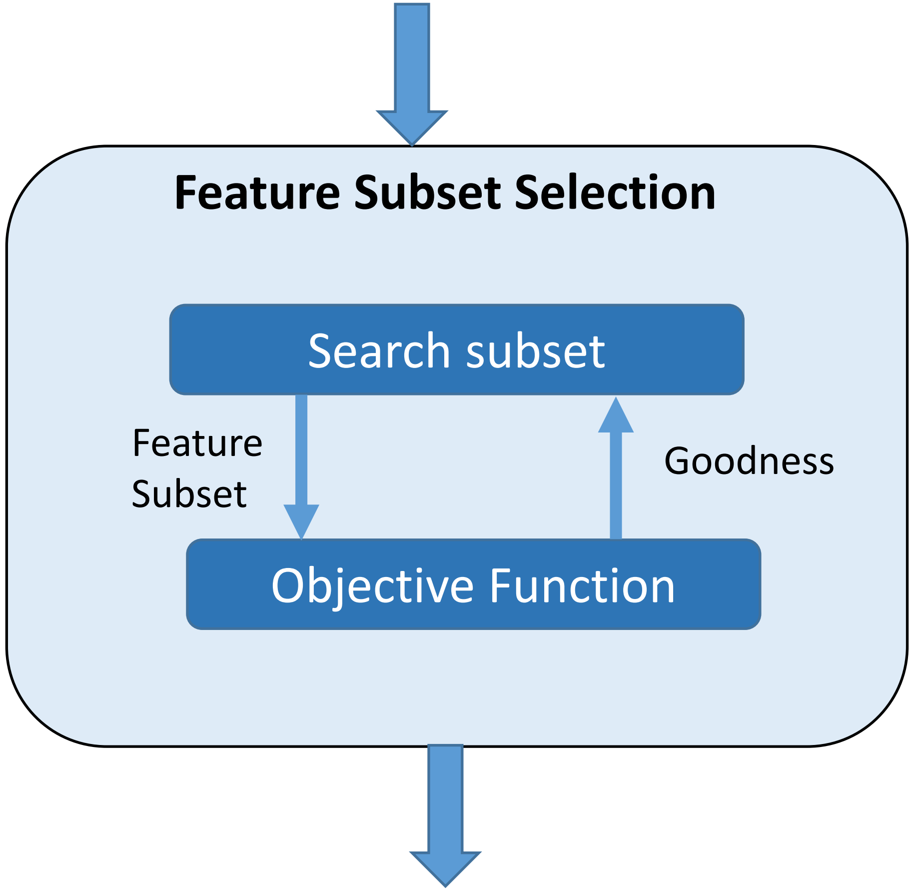
Can't search over all possibilities and therefore we rely on heuristic methods.

Dimensionality Reduction

Feature Selection:

How do we choose this subset?

$$D = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), \dots, (\mathbf{x_n}, y_n)\} \subseteq \mathcal{X}^d \times \mathcal{Y}$$



$$D = \{(\mathbf{z_1}, y_1), (\mathbf{z_2}, y_2), \dots, (\mathbf{z_n}, y_n)\} \subseteq \mathcal{X}^k \times \mathcal{Y}$$

$$\mathbf{z} = [x_{i_1}, x_{i_2}, \dots, x_{i_k}]$$

- *Feature selection can be considered as an optimization problem that involves*
  - *Searching of the space of possible feature subsets*
  - *Choose the subset that is optimal or near-optimal with respect to some objective function*

- ***Filter Methods** (unsupervised method)*
  - *Evaluation is independent of the learning algorithm*
  - *Consider the input only and select the subset that has the most information*

- ***Wrapper Methods** (supervised method)*
  - *evaluation is carried out using model selection the machine learning algorithm*
  - *Train on selected subset and estimate error on validation dataset*

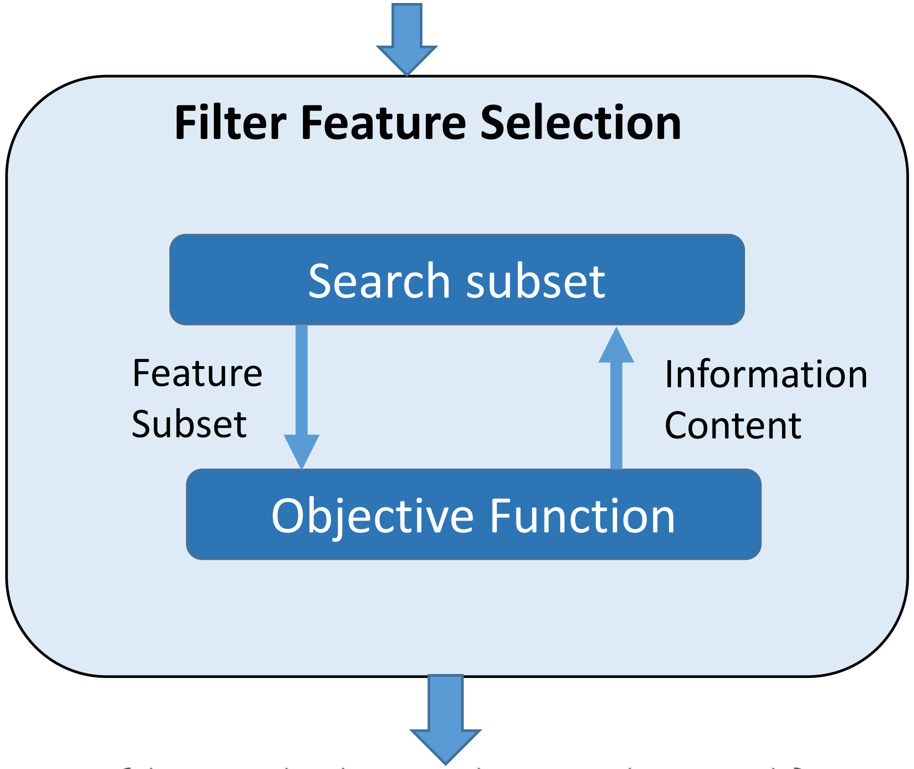
Dimensionality Reduction

Feature Selection:

How do we choose this subset?

Filter Methods

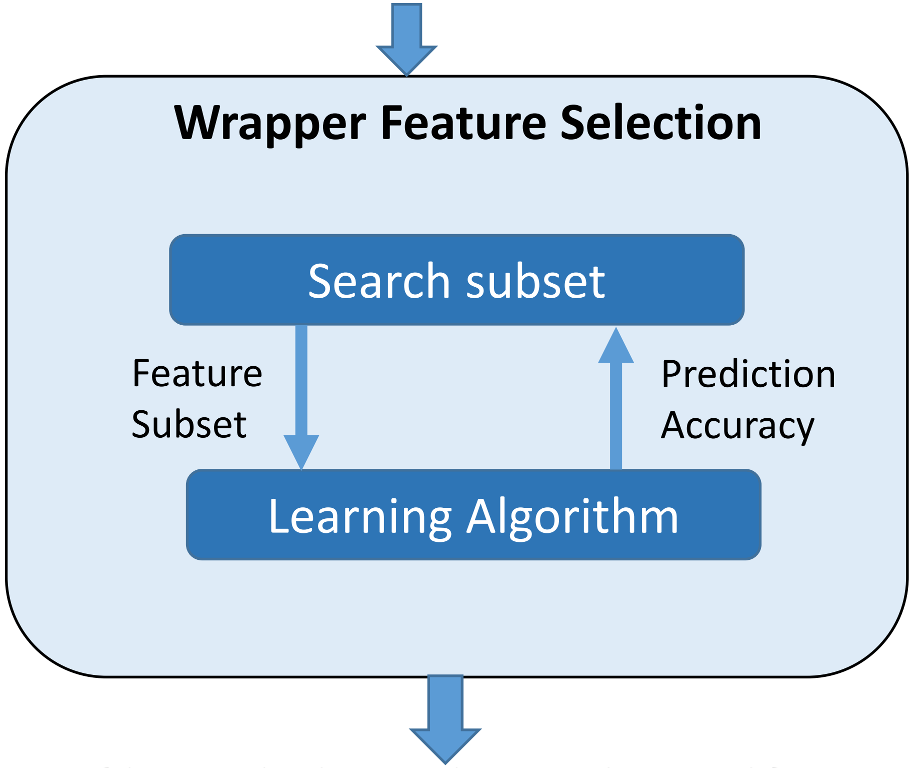
$$D = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), \dots, (\mathbf{x_n}, y_n)\} \subseteq \mathcal{X}^d \times \mathcal{Y}$$



$$D = \{(\mathbf{z_1}, y_1), (\mathbf{z_2}, y_2), \dots, (\mathbf{z_n}, y_n)\} \subseteq \mathcal{X}^k \times \mathcal{Y}$$

Wrapper Methods

$$D = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), \dots, (\mathbf{x_n}, y_n)\} \subseteq \mathcal{X}^d \times \mathcal{Y}$$



$$D = \{(\mathbf{z_1}, y_1), (\mathbf{z_2}, y_2), \dots, (\mathbf{z_n}, y_n)\} \subseteq \mathcal{X}^k \times \mathcal{Y}$$

Dimensionality Reduction

Feature Selection:

Filters Method:

- ***Univariate Methods***
  - *Treats each feature independently of other features*

- *Calculate score of each feature against the label using the following metrics:*
  - *Pearson correlation coefficient*
  - *Mutual Information*
  - *F-score*
  - *Chi-square*
  - *Signal-to-noise ratio (SNR), etc.*

- *Rank features with respect to the score*

- *Select the top k-ranked features (k is selected by the user)*

Dimensionality Reduction

Feature Selection:

Filters Method – Ranking Metrics:

- ***Pearson correlation coefficient (measure of linear dependence)***

  Denote feature values by a vector $\mathbf{a} \in \mathbf{R}^n$ (Note n is the number of points).

  Denote labels by a vector $\mathbf{y} = [y_1, y_2, \ldots, y_n]$.

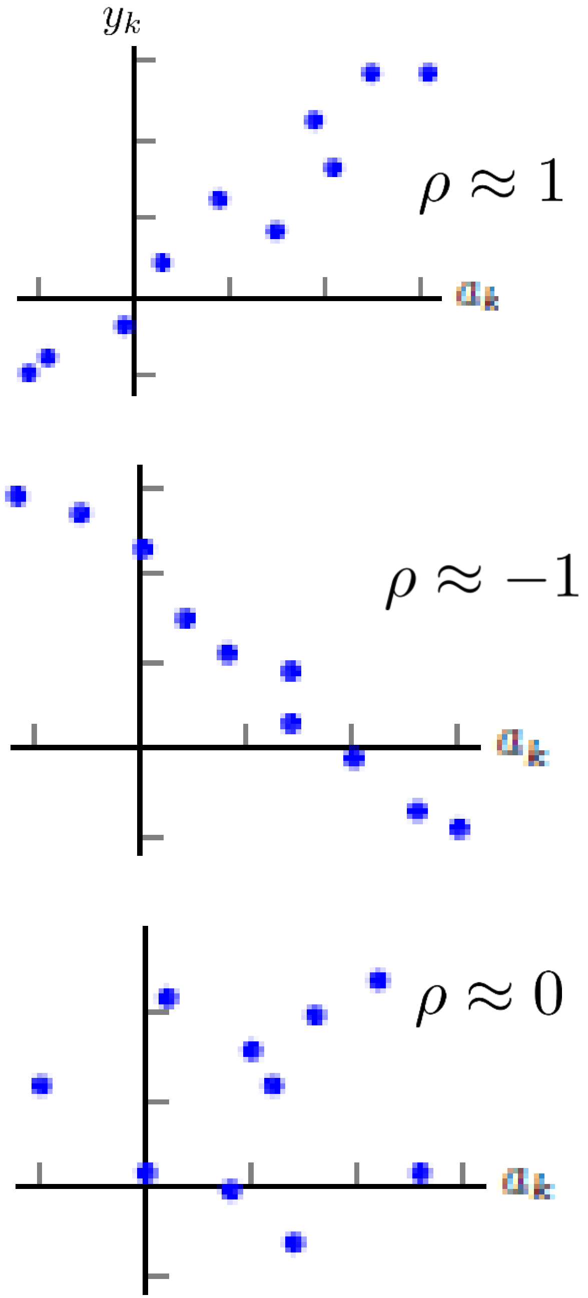
  Define Pearson correlation coefficient as

$$\rho = \frac{\tilde{\mathbf{a}}^T \tilde{\mathbf{y}}}{\|\tilde{\mathbf{a}}\|_2 \|\tilde{\mathbf{y}}\|_2}, \quad |\rho| \leq 1$$

Here

$$\tilde{\mathbf{a}} = \mathbf{a} - \text{avg}(\mathbf{a})\mathbf{1}$$

is a demeaned vector and is obtined by subtracting mean of a vector from it.



- ***Signal-to-noise ratio (SNR)***

$$\text{SNR} = \frac{\text{avg}(\mathbf{a}) - \text{avg}(\mathbf{y})}{\text{std}(\mathbf{a}) - \text{std}(\mathbf{y})},$$

where std denotes the standard deviation of the vector.

Dimensionality Reduction

Feature Selection:

Wrappers Method:

- **Forward Search Feature Subset Selection Algorithm (Super intuitive)**
  - Start with empty set as feature subset
  - Try adding one feature from the remaining features to the subset
  - Estimate classification or regression error for adding each feature
  - Add feature to the subset that gives max improvement

- **Backward Search Feature Subset Selection Algorithm (Super intuitive)**
  - Start with full feature set as subset
  - Try removing one feature from the subset
  - Estimate classification or regression error for removing each feature
  - Remove/drop the feature that gives minimal impact on error or reduces the error

Outline

- Dimensionality Reduction
- Feature Selection
- Feature Extraction - PCA

Dimensionality Reduction

Feature Extraction:

Transform existing features to obtain a set of new features using some mapping function.

$$\mathbf{x} = [x_1, x_2, \ldots, x_d]$$

$$\mathbf{z} = f(\mathbf{x})$$

$$\mathbf{z} = [z_1, z_2, \ldots, z_k]$$

- The mapping function $\mathbf{z}=f(\mathbf{x})$ can be **linear** or non-linear.

- Can be interpreted as projection or mapping of the data in the higher dimensional space to the lower dimensional space.

- Mathematically, we want to find an optimum mapping $\mathbf{z}=f(\mathbf{x})$ that preserves the desired information as much as possible.

Dimensionality Reduction

Feature Extraction:

Idea:

- Finding optimum mapping is equivalent to optimizing an **objective** function.
- We use different objective functions in different methods;
  - Minimize Information Loss: Mapping that represent the data as accurately as possible in the lower-dimensional space, e.g., Principal Components Analysis (PCA).
  - Maximize Discriminatory Information: Mapping that best discriminates the data in the lower-dimensional space, e.g., Linear Discriminant Analysis (LDA).
- Here we focus on PCA, that is, a linear mapping.
- Why Linear: Simpler to Compute and Analytically Tractable.

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

- Given features in d-dimensional space

- Project into lower dimensional space using the following linear transformation

$$\mathbf{z} = \mathbf{W}^T \mathbf{x}$$

- For example (can you tell me size of matrix W for the following cases),
  - find best planar approximation to 4D data
  - find best planar approximation to 100D data

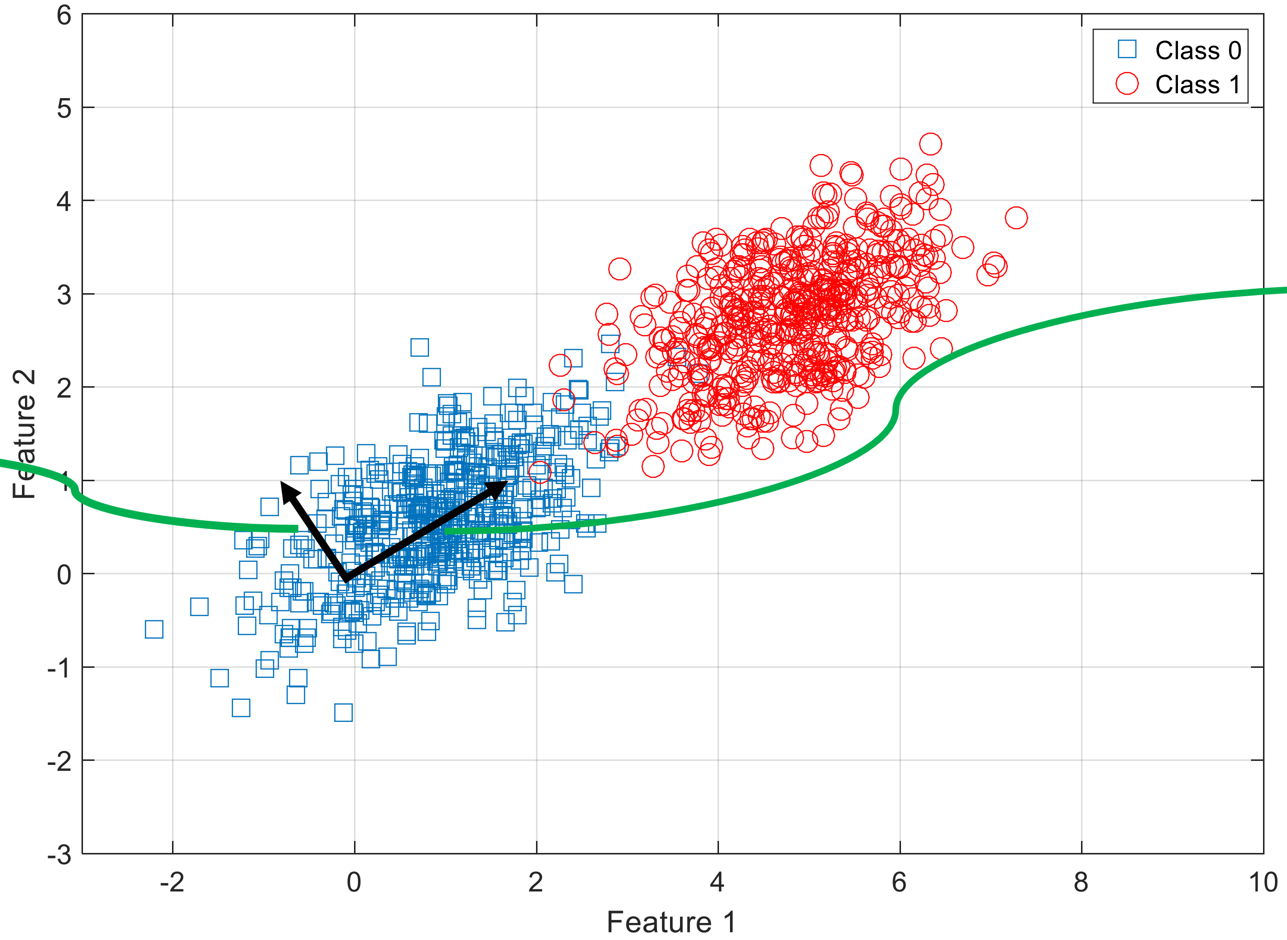
- We want to find this mapping while preserving as much information as possible, and ensuring
  - **Objective 1:** the features after mapping are uncorrelated; cannot be reduced further
  - **Objective 2:** the features after mapping have large variance

LUMS
A Not-for-Profit University

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Geometric Intuition:



Second Principal Component

Most contribution of each class lies in this direction

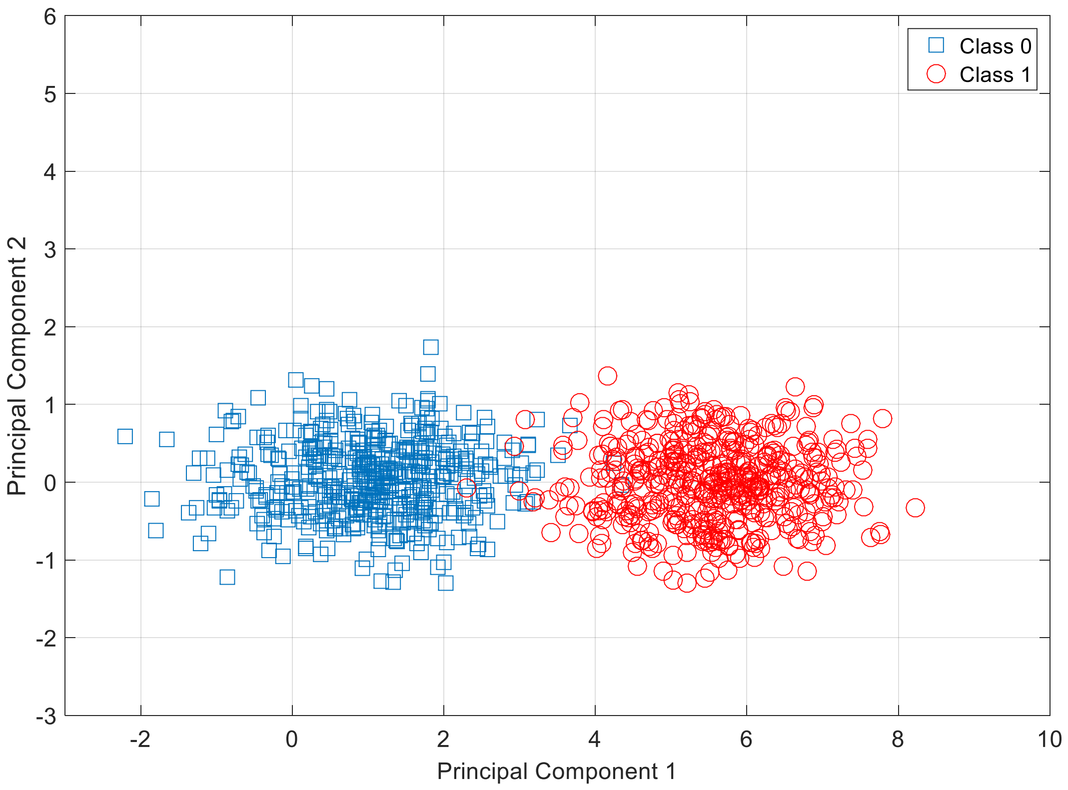
First Principal Component

Toy Illustration in two dimensions

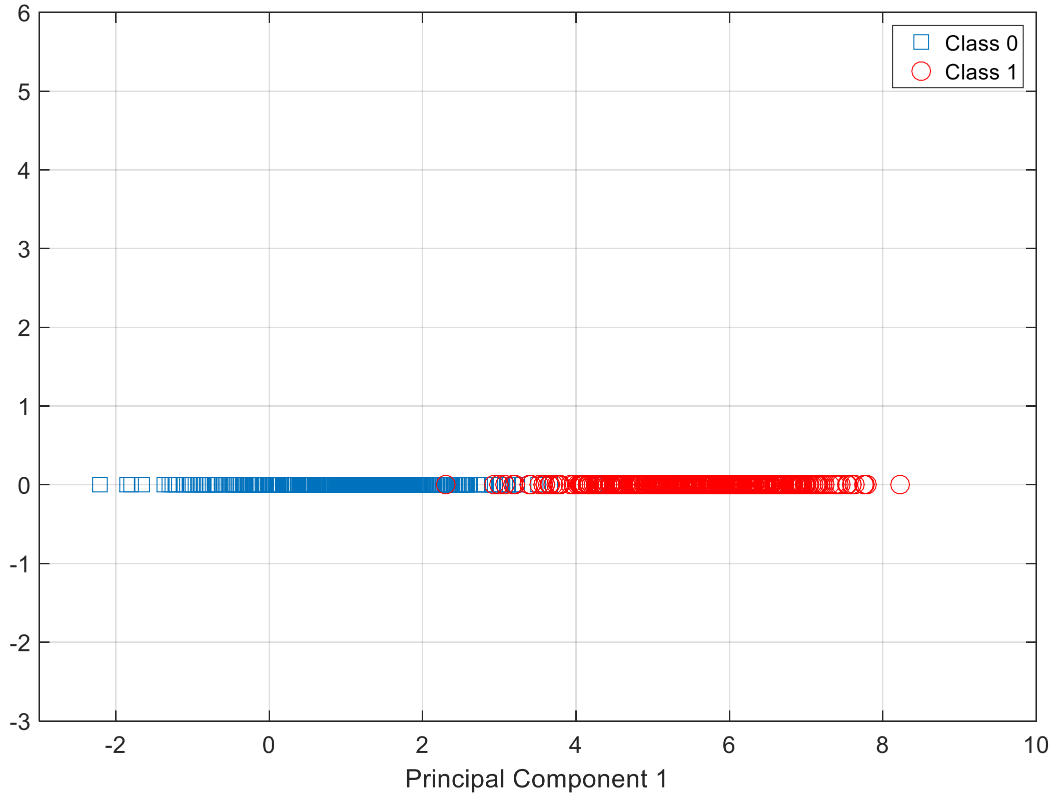
Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Geometric Intuition:



Change of coordinates: Linear combinations of features



Ignoring the Second Component/Feature

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Mathematical Formulation:

We have n feature vectors of the form $\mathbf{x} \in \mathbf{R}^d$.

Note d represents the number of features.

In PCA, we want to represent $\mathbf{x}$ in a new space of lower dimensionality using only k basis vectors $(k < N)$, that is,

$$\hat{\mathbf{x}} = \sum_{i=1}^{k} z_i \mathbf{v}_i$$

such that

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2$$

is minimized.

Here $\mathbf{v}_i \in \mathbf{R}^d$ for $i = 1, 2, \ldots, k$ represent the k number of orthogonal vectors that form the basis, referred to as principal components, of the subspace of dimensionality$=k$.

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Mathematical Formulation:

How do we find the basis vectors $\mathbf{v}_i \in \mathbf{R}^d$ for $i = 1, 2, \ldots, k$?

Steps to find Principal Components:

We have n feature vectors $\mathbf{x}_i \in \mathbf{R}^d, \quad i = 1, 2, \ldots, n$.

Step 1: Compute Sample Mean:

Sample mean (note summtion over the number of feature vectors n)

$$\overline{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i$$

Step 2: Subtract Sample Mean:

Subtract sample mean from each feature vector $\mathbf{x}_i$ to obtain $\mathbf{s}_i$, that is,

$$\mathbf{s}_i = \mathbf{x}_i - \overline{\mathbf{x}}$$

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Mathematical Formulation:

Step 3: Calculate the Covariance Matrix:

Now we have n feature vectors $\mathbf{s}_i \in \mathbf{R}^d, \quad i = 1, 2, \ldots, n.$

Calculate the Covariance Matrix as follows

$$\Sigma = \frac{1}{n}\sum_{i=1}^{n} \mathbf{s}_i \mathbf{s}_i^T$$

This can also be expressed as

$$\Sigma = \frac{1}{n}\mathbf{S}\mathbf{S}^T$$

where

$$\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n]$$

What is special about these vectors?

Zero mean; taken along all feature vectors

How do you interpret the entries of the matrix? Spend some time and try to understand this!

For two vectors $\mathbf{f}, \mathbf{g} \in \mathbf{R}^n$, covariance is defined as

$$\sigma_{\mathbf{fg}} = \frac{1}{n}\sum_{i}^{n} (f_i - \text{avg}(\mathbf{f}))\,(g_i - \text{avg}(\mathbf{g}))$$

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Special about the Covariance Matrix:

The covarince matrix is symmetric, that is, $\Sigma^T = \Sigma$. (super easy to show)

The covarince matrix is positive semi-definite. (again, super easy)

Size of Σ is $d \times d$.

Step 4: Carry out Eigenvalue Decomposition of Covariance Matrix:

Carry out eigenvalue decomposition of the covarince matrix as

$$\Sigma = \mathbf{V}\mathbf{D}\mathbf{V}^T$$

Here the matrix $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d]$ contains d orthogonal eigenvectors $\mathbf{v}_i \in \mathbf{R}^d$, referred to as principal components, that serve as the basis of $\mathbf{R}^d$.

Here the matrix $\mathbf{D}$ is a diagonal matrix with eigenvalues denoted by $\lambda_1, \lambda_2, \ldots, \lambda_d$.

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Step 5: Dimensionality Reduction

We wanted to find the basis vectors $\mathbf{v}_i \in \mathbf{R}^d$ for $i = 1, 2, \ldots, k$.

We have $\mathbf{v}_i \in \mathbf{R}^d$ for $i = 1, 2, \ldots, d$.

- Q: How to select k out of d?

- A: Simple, select the ones corresponding to k largest eigenvalues.

Construct the maapping matrix of size $d \times k$ as

$$\mathbf{W} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k]$$

to reduce the dimensionality of the feature space from $\mathbf{R}^d$ to $\mathbf{R}^k$ as

$$\mathbf{z} = \mathbf{W}^T \mathbf{x}$$

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Using $\mathbf{z}$, we can go back to $\mathbf{R}^d$ to obtain apprxoimation of $\mathbf{x}$ as

$$\hat{\mathbf{x}} = \sum_{i=1}^{k} z_i \mathbf{v}_i = \mathbf{W}\mathbf{z}$$

Connection with the Objectives:

- ***Objective 1***: the features after mapping are uncorrelated; cannot be reduced further
  - Enabled by orthogonality of the principal components
- ***Objective 2***: the features after mapping have large variance
  - We have used covariance matrix to define the mapping and used eigenvectors with largest eigenvalues, that is, those dimensions capturing the variations in the data.
  - **PCA maps the data along the directions where we have most of the variations in the data.**

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

How do we choose k?

- It depends on the amount of information, that is variance, we want to preserve in the mapping process.

- We can define a variable T to quantify this preservation of information

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{d} \lambda_i} > T$$

- T=1, when k=d; No reduction.
- T=0.8, interpreted as that 80% variation in the data has been preserved.

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Example: $d = 2,\ n = 10,\ k = 1$

Step 1: Compute sample mean:

$$\bar{\mathbf{x}} = [1.81, 1.91]$$

	x_1	x_2
$\mathbf{x}_1$	2.5000	2.4000
$\mathbf{x}_2$	0.5000	0.7000
	2.2000	2.9000
	1.9000	2.2000
	3.1000	3.0000
	2.3000	2.7000
	2.0000	1.6000
	1.0000	1.1000
	1.5000	1.6000
	1.1000	0.9000

Step 2: Subtract Sample Mean:

$$\mathbf{s}_i = \mathbf{x}_i - \bar{\mathbf{x}}$$

	s_1	s_2
$\mathbf{s}_1$	0.6900	0.4900
$\mathbf{s}_2$	-1.3100	-1.2100
	0.3900	0.9900
	0.0900	0.2900
	1.2900	1.0900
	0.4900	0.7900
	0.1900	-0.3100
	-0.8100	-0.8100
	-0.3100	-0.3100
	-0.7100	-1.0100

Step 3: Calculate the Covariance Matrix:

$$\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n]$$

$$\Sigma = \frac{1}{n}\sum_{i=1}^{n} \mathbf{s}_i \mathbf{s}_i^T = \frac{1}{n}\mathbf{S}\mathbf{S}^T$$

$$\Sigma = \begin{bmatrix} 0.5549 & 0.5539 \\ 0.5539 & 0.6449 \end{bmatrix}$$

We have divided by n. Some authors divide by n-1. It won't change the principal components

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Example:

Step 4: Carry out Eigenvalue Decomposition of Covariance Matrix:

$$\Sigma = \mathbf{V}\mathbf{D}\mathbf{V}^T \qquad \mathbf{V} = \begin{bmatrix} -0.7352 & 0.6779 \\ 0.6779 & 0.7352 \end{bmatrix} \qquad \mathbf{D} = \begin{bmatrix} 0.0442 & 0 \\ 0 & 1.1556 \end{bmatrix}$$

Step 5: Dimensionality Reduction

Use $\mathbf{W} = [\mathbf{v}_2]$ (associated with the largest eigenvalue) to reduce the dimensionality of the feature space from $\mathbf{R}^2$ to $\mathbf{R}$ as

$$\mathbf{z} = \mathbf{W}^T\mathbf{x}$$

$\mathbf{z}$
3.4591
0.8536
3.6233
2.9054
4.3069
3.5441
2.5320
1.4866
2.1931
1.4073

Dimensionality Reduction

Feature Extraction - Principal Component Analysis:

Practical Considerations and Limitations:

- Data should be normalized before using PCA for dimensionality reduction.
- Usually, we normalize every feature by subtracting mean of that feature followed by dividing with standard deviation of the feature.
- The covariance matrix of the reduced feature is projection along orthogonal components (directions) and therefore features are uncorrelated to each other. In other words, PCA decorrelates the features.
- **Limitation:**
  - PCA does not consider the separation of data with respect to class label and therefore we do not have a guarantee the mapping of the data along dimensions of maximum variance results in the new features good enough for class discrimination.

  Solution: Linear Discriminant Analysis (LDA) - Find mapping directions along which the classes are best separated.

Feedback: Questions or Comments?

Email: zubair.khalid@lums.edu.pk