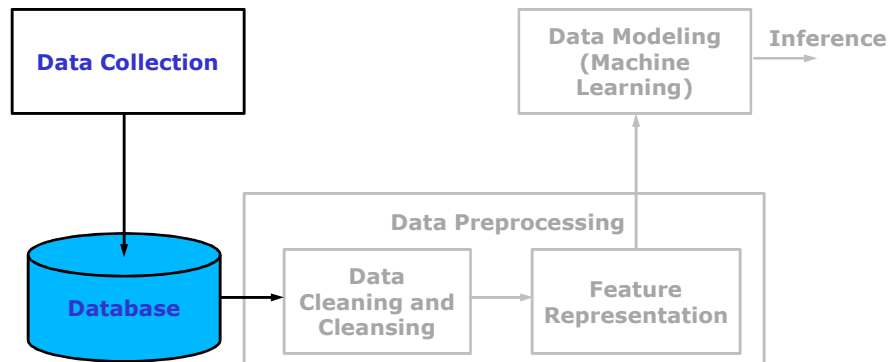


Data Preprocessing

Data Science

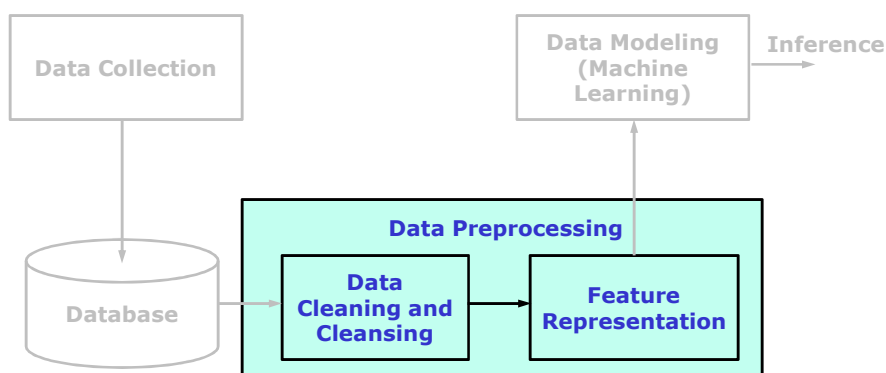
- Multi-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insight from structured and unstructured data
- Central concept is gaining insight from data
- Machine learning uses data to extract knowledge



2

Data Science

- Multi-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insight from structured and unstructured data
- Central concept is gaining insight from data
- Machine learning uses data to extract knowledge



3

Data Preprocessing

- Real world data are tend to be incomplete, noisy and inconsistent due to their huge size and their likely origin from multiple heterogeneous sources
- Preprocessing is important to clean the data
- Low quality data will lead to low quality of analysis results
- If the users believe the data is of low quality (dirty), they are unlikely to trust the results of any data analytics that has been applied to
- Low quality data can cause confusion for analytic procedure using machine learning techniques, resulting in unreliable output
- Incomplete, noisy and inconsistent data are common properties of large real world databases

Tuple (Record)

- A **tuple (record)** is finite ordered list (sequence) of elements, where each element is belong to an attribute

Date/ Time	Temperature (C)/ Humidity (%)	Pressure (Pa)	Rain (Inches)	Light Intensity (lux)	Accelerations (g)	Force (N)	Molsture (%)
2017-09-06 18:44:32	23.00,56.00	617.64	0.01	3	0.52,0.31,-0.80,0.00,0.00,0.00,31.36,-159.01	0.02	81.00
2017-09-06 18:33:32	24.00,58.00	619.47	0.01	12	0.52,0.30,-0.79,0.00,0.00,0.00,31.45,-159.12	0.02	82.00
2017-09-06 18:22:39	24.00,58.00	623.37	0.00	71	0.52,0.31,-0.80,0.00,0.00,0.00,31.35,-158.88	0.02	83.00
2017-09-06 18:11:31	25.00,60.00	627.02	0.05	194	0.51,0.31,-0.80,0.00,0.00,0.00,30.80,-159.00	0.02	81.00

**Tuple
(record)**

- Each row is a tuple

Incomplete Data

- Many tuple (records) have **no recorded value for several attributes**
- Reasons for incomplete data:**
 - User forgot to fill in a field
 - User chose not to fill out the field as it was not considered important at the time of the entry
 - Relevant data may not be recorded due to malfunctioning of equipment
 - Data might have lost while transferring from recorded place
 - Data may not be recorded due to programming error
 - Data might not be recorded due to technology limitations like limited memory

Noisy Data

- Many tuple (records) have incorrect value for several attributes
- Reasons for noisy data:
 - There may be human or computer error occurring in data entry
 - The data collection instruments used may be faulty
 - Error in data transmission
 - There may be technology limitation such as limited buffer size for coordinating synchronised data transfer and consumption

Inconsistent Data

- Data containing discrepancies in stored values for some attributes
- Reasons for inconsistent data:
 - It may result from inconsistencies in name conventions or data codes used or inconsistent formats of input fields such as date
 - Inconsistency in name convention or formats of input fields while integrating
 - Inconsistent data may be due to human or computer error occurring in data entry

Data Preprocessing Techniques

- **Data cleaning:**
 - Applied to identify the missing values, fill in missing values, removing noise and correcting inconsistency in the data
- **Data integration:**
 - It merges data from multiple sources in to a coherent data source
- **Data transformation:**
 - Transforming the entries of data to a common format
 - Techniques like normalization and standardization applied to transform the data to another form to improve the accuracy and efficiency of machine learning (ML) algorithms involving distance measures

Data Preprocessing Techniques

- **Data reduction:**
 - Applied to obtain a reduced representation that is much smaller in volume, yet producing almost same analytical results
 - It can reduce the data size by
 - Aggregation
 - Eliminating irrelevant and redundant features (attributes) through correlation analysis
 - Reducing dimension
- *These techniques are not mutually exclusive; they may work together*

Descriptive Data Summarization (Descriptive Analytics)

- It serves as a foundation for data preprocessing
- It helps us to study the general characteristics of data and identify the presence of noise or outliers
- Data characteristics:
 - Central tendency of data
 - Centre of the data
 - Measuring mean, median and mode
 - Dispersion of data
 - The degree to which numerical data tend to spread
 - Measuring range, quartiles, interquartile range (IQR), the five-number summary and standard deviation

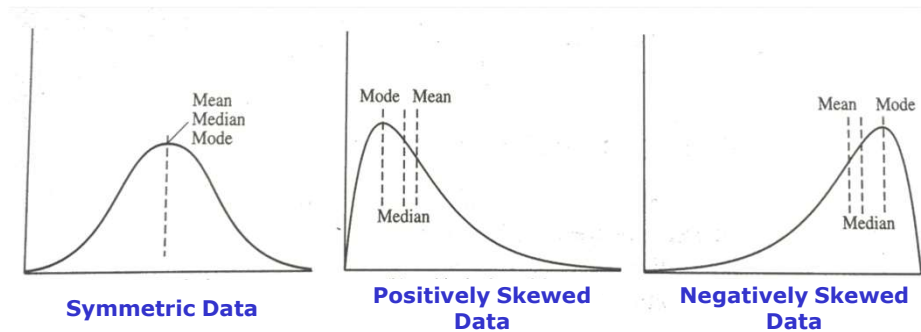
Descriptive Analytics: Measuring Central Tendency

- Mean:
 - Let x_1, x_2, \dots, x_N be a set of N values in an attribute. Mean of this set of values is given by

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i$$
 - Mean is a better measure of central tendency for the symmetric data (symmetrically distributed data)
- Median:
 - For asymmetrically distributed (skewed) data, a better measure of centre of data is median
 - For a given data of N values in sorted order
 - In N is odd, then median is the middle value of the ordered list
 - In N is even, then median is the average of middle two values

Descriptive Analytics: Measuring Central Tendency

- **Mode:** Most frequent value in an attribute in the data



Descriptive Analytics: Measuring Dispersion of Data

- The degree to which numerical data tend to spread
- It is also called as variance
- Common measures:
 - Range
 - The five-number summary (based on quartiles)
 - The inter quartile range (IQR)
 - Standard deviation
- **Range:** The range of a set is the different between the maximum and minimum values

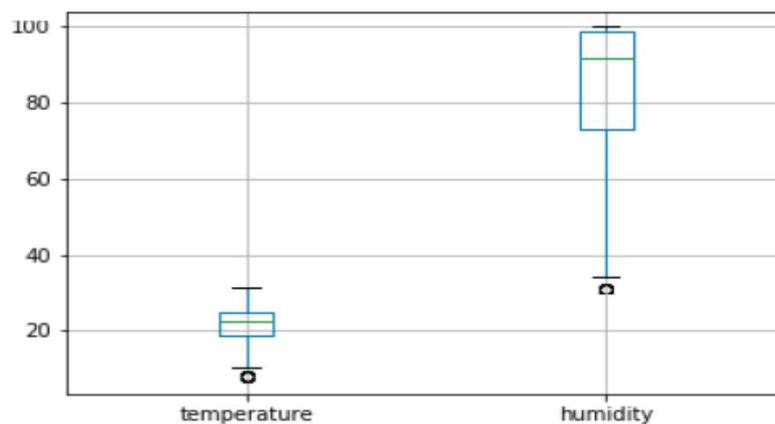
Descriptive Analytics: Measuring Dispersion of Data

- Quartiles:
 - The k^{th} percentile:
 - Let x_1, x_2, \dots, x_N be a set of N values in an attribute
 - The k^{th} percentile of a set of data in numerical order is the value of x_i having the property that k percent of data entries lie at or below x_i
 - Median is the 50th percentile
 - The first quartile (Q1): It is the 25th percentile
 - The third quartile (Q3): It is the 75th percentile
 - The quartiles including median give some indication of centre, spread and shape of distribution
- Inter quartile range (IQR): Distance between the first and third quartile

$$\text{IQR} = \text{Q3} - \text{Q1}$$

Descriptive Analytics: Measuring Dispersion of Data

- The five-number summary of distribution:
 - It consists of minimum value, Q1, median, Q3 and maximum value
 - Box plots are the popular way of visualising distribution



Data Cleaning: Handling Missing Values and Noisy Data

Data Cleaning (Data Cleansing)

- Real world data are tend to be **incomplete**, **noisy** and **inconsistent**
- **Data cleaning** routines attempt to **identify missing values**, **fill in missing values**, **smooth out noise** while identifying outliers and **correct inconsistencies** in the data

” 80 percent of a data scientist's valuable time is spent simply finding, cleansing, and organizing data, leaving only 20 percent to actually perform analysis...

IBM Data Analytics

- One of the biggest data cleaning task is handling **missing values**

Data Cleaning: Missing Values

- Many tuple (records) have no recorded value for several attributes
- Identifying missing values:
 - When Pandas library for python is used, it detect the missing values as "NaN" [1]
 - It automatically consider "blank" in the attribute value, "NaN/nan/NAN" in the attribute value, "NA" in the attribute value, "n/a" in the attribute value, "NULL/null" in the attribute value as NaN

[1] https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.read_csv.html

Methods to Handle Missing Values

- Ignore the tuples:
 - This method is effective only when the tuples contain several attributes (> 50% of attributes) with missing value

	Dates	Station Id	Temperature	Humidity	Rain
1	08-07-2018	t10	25.46875	82.1875	6.75
2	09-07-2018			83.14912	
3	10-07-2018	t10	25.17021	85.34043	652.5
4	11-07-2018	t10	24.29851	87.68657	963
5	08-07-2018	t11			
6	09-07-2018	t11	26.8494	61.10241	15
7	10-07-2018	t11	27.88806	75.07463	13583.25
8	11-07-2018	t11	27.35915	76.02113	19768.5
9	23-07-2018	t12	24.39024	94.4065	1071
10	24-07-2018	t12	24.16197	97.66901	438.75
11	25-07-2018				
12	26-07-2018	t12	22.19718	99	864



	Dates	Station Id	Temperature	Humidity	Rain
1	08-07-2018	t10	25.46875	82.1875	6.75
2	10-07-2018	t10	25.17021	85.34043	652.5
3	11-07-2018	t10	24.29851	87.68657	963
4	09-07-2018	t11	26.8494	61.10241	15
5	10-07-2018	t11	27.88806	75.07463	13583.25
6	11-07-2018	t11	27.35915	76.02113	19768.5
7	23-07-2018	t12	24.39024	94.4065	1071
8	24-07-2018	t12	24.16197	97.66901	438.75
9	26-07-2018	t12	22.19718	99	864

Tuples contain several attributes (> 50% of attributes) with missing value

Methods to Handle Missing Values

- Ignore the tuples:
 - This method is effective only when the tuples contain several attributes (> 50% of attributes) with missing value
 - This method is also used when the target variable (class label) is missing

1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	83.14912	
4	10-07-2018	NaN	25.17021	85.34043	652.5
5	11-07-2018	t10	24.29851	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494		15
8	10-07-2018	t11	27.88806	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12		94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	NaN	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864



1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	83.14912	
4	11-07-2018	t10	24.29851	87.68657	963
5	08-07-2018	t11	23.53846	61.92308	3
6	09-07-2018	t11	26.8494		15
7	10-07-2018	t11	27.88806	75.07463	13583
8	11-07-2018	t11	27.35915	76.02113	19769
9	23-07-2018	t12		94.4065	1071
10	24-07-2018	t12	24.16197	97.66901	438.8
11	26-07-2018	t12	22.19718	99	864

Target attribute (StationID) with missing value

Methods to Handle Missing Values

- Fill in the missing values (imputing values) manually:
 - Time consuming
 - Not feasible given a large data set with many missing values
- Use a global constant to fill in missing value (Imputing global constant):
 - Replace all missing attribute values by a same constant
 - Imputed value may not be correct

Methods to Handle Missing Values

- Use attribute mean/median/mode to fill in the missing value (mean/median/mode imputation):
 - Applicable to numeric data
 - Centre of the data won't change
 - However, it does not preserve the relationship with other variables

	Dates	Station Id	Temperature	Humidity	Rain
1					
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	NaN	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	NaN	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	NaN	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864

Methods to Handle Missing Values

- Use attribute mean/median/mode to fill in the missing value (mean/median/mode imputation):
 - Applicable to numeric data
 - Centre of the data won't change
 - However, it does not preserve the relationship with other variables

	Dates	Station Id	Temperature	Humidity	Rain
1					
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	NaN	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	NaN	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	NaN	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864



	Dates	Station Id	Temperature	Humidity	Rain
1					
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	25.1368	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	25.1368	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	25.1368	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864

Methods to Handle Missing Values

- Use attribute mean/median/mode to fill in the missing value (mean/median/mode imputation):
 - Applicable to numeric data
 - Centre of the data won't change
 - However, it does not preserve the relationship with other variables

1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	25.1368	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	25.1368	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	25.1368	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864



1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	85.42	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	25.1368	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	85.42	15
8	10-07-2018	t11	25.1368	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	25.1368	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864

Methods to Handle Missing Values

- Filling with local mean/median/mode:
 - Use attribute mean/median/mode of all samples belonging to a **group (class)** to fill in the missing value
 - Applicable to numeric data
 - Centre of the data of a **group** won't change
 - However, it does not preserve the relationship with other variables

1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	NaN	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	NaN	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	NaN	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864




1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	25.612	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	NaN	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	NaN	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864

Methods to Handle Missing Values

- Filling with local mean/median/mode:
 - Use attribute mean/median/mode of all samples belonging to a **group (class)** to fill in the missing value
 - Applicable to numeric data
 - Centre of the data of a **group** won't change
 - However, it does not preserve the relationship with other variables

1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	25.612	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	NaN	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	NaN	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864




1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	25.612	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	25.916	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	NaN	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864

Methods to Handle Missing Values

- Filling with local mean/median/mode:
 - Use attribute mean/median/mode of all samples belonging to a **group (class)** to fill in the missing value
 - Applicable to numeric data
 - Centre of the data of a **group** won't change
 - However, it does not preserve the relationship with other variables

1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	25.612	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	25.916	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	NaN	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864



1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	25.612	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	25.916	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	23.884	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864

Methods to Handle Missing Values

- Use the values from the previous/next record (with in a group) to fill in missing value (**Padding**)

1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	NaN	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	NaN	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	NaN	15
8	10-07-2018	t11	NaN	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	NaN	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864



1	Dates	Station Id	Temperature	Humidity	Rain
2	08-07-2018	t10	25.46875	82.1875	6.75
3	09-07-2018	t10	26.19298	82.1875	1762
4	10-07-2018	t10	25.17021	85.34043	652.5
5	11-07-2018	t10	25.17021	87.68657	963
6	08-07-2018	t11	23.53846	61.92308	3
7	09-07-2018	t11	26.8494	61.92308	15
8	10-07-2018	t11	26.8494	75.07463	13583
9	11-07-2018	t11	27.35915	76.02113	19769
10	23-07-2018	t12	24.16197	94.4065	1071
11	24-07-2018	t12	24.16197	97.66901	438.8
12	25-07-2018	t12	25.29323	94.84211	13667
13	26-07-2018	t12	22.19718	99	864

- If the data is **categorical** or **text**, one can replace the missing values by **most frequent observations**

Methods to Handle Missing Values

- Use most probable value to fill the missing value:
 - Use **interpolation technique** to predict the missing value
 - Linear interpolation** is achieved by geometrically rendering a straight line between two adjacent points on a graph or plane
 - Interpolation happens column wise
 - Popular strategy
 - It does not preserves the relationship with other variables

1	Dates	Temperature	Humidity	Rain
2	08-07-2018	25.46875	82.1875	6.75
3	09-07-2018	26.19298	83.1491	1761.75
4	10-07-2018	25.17021	85.3404	652.5
5	11-07-2018	NaN	87.6866	963
6	12-07-2018	24.06923	87.6462	254.25
7	13-07-2018	21.20779	95.9481	339.75
8	15-07-2018	23.48571	96.1714	38.25
9	18-07-2018	NaN	98.5897	29.25
10	19-07-2018	25.09346	88.3271	4.5
11	20-07-2018	25.39423	90.4327	112.5
12	21-07-2018	NaN	94.5378	735.75
13	22-07-2018	22.5098	99	607.5
14	23-07-2018	22.904	98	717.75
15	24-07-2018	NaN	99	513
16	25-07-2018	23.18182	98.9697	195.75

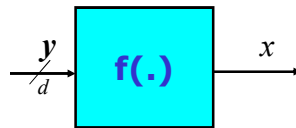


1	Dates	Temperature	Humidity	Rain
2	08-07-2018	25.46875	82.1875	6.75
3	09-07-2018	26.19298	83.1491	1761.75
4	10-07-2018	25.17021	85.3404	652.5
5	11-07-2018	24.2	87.6866	963
6	12-07-2018	24.06923	87.6462	254.25
7	13-07-2018	21.20779	95.9481	339.75
8	15-07-2018	23.48571	96.1714	38.25
9	18-07-2018	21.5	98.5897	29.25
10	19-07-2018	25.09346	88.3271	4.5
11	20-07-2018	25.39423	90.4327	112.5
12	21-07-2018	23.7	94.5378	735.75
13	22-07-2018	22.5098	99	607.5
14	23-07-2018	22.904	98	717.75
15	24-07-2018	21.6	99	513
16	25-07-2018	23.18182	98.9697	195.75

Methods to Handle Missing Values

- Use most probable value to fill the missing value:
 - Use regression techniques to predict the missing value (regression imputation)
 - Let y_1, y_2, \dots, y_d be a set of d attributes
 - Regression (multivariate): The n^{th} value is predicted as

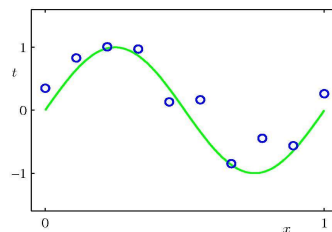
$$x_n = f(y_{n1}, y_{n2}, \dots, y_{nd})$$



- Linear regression (multivariate): $x_n = w_1 y_{n1} + w_2 y_{n2} + \dots + w_d y_{nd}$
- Popular strategy
- It uses the most information from the present data to predict the missing values
- It preserves the relationship with other variables

Data Cleaning: Smoothing the Noisy Data

- Noise is a random error or variance in a measured variable
- Due to noise, many tuple (records) have incorrect value for several attributes
- Mostly data is full of noise
- Smooth out the data to remove noise
- Data smoothing allows important patterns to stand out
- The idea is to sharpen the patterns (values) in the data and highlight trends the data is pointing to



- Methods for data smoothing:
 - Binning
 - Regression

Binning Methods for Data Smoothing

- Binning method smooth a sorted data value of a noisy attribute by consulting its neighbourhood i.e., the values around it
- It perform local smoothing as this method consult the neighbourhood of values
- The sorted values are partitioned into (almost) equal-frequency bins
- Smoothing by bin means:
 - Each value in a bin is replaced by the mean value of the bin
- Smoothing by bin medians:
 - Each value in a bin is replaced by the median value of the bin

Binning Methods for Data Smoothing

- Smoothing by bin boundaries:
 - The minimum and maximum values in a given bin are identified as bin boundaries
 - Each bin value is then replaced by the closest boundary value
- Larger the width, the greater the effect of the smoothing
- Example:
 - Noisy data for price (in Rs) : 8, 15, 34, 24, 4, 21, 28, 21, 25
 - Sorted data for price (in Rs) : 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into bins: Smoothing by bin means: Smoothing by bin Boundaries:

Bin1: 4, 8, 15

Bin1: 9, 9, 9

Bin1: 4, 4, 15

Bin2: 21, 21, 24

Bin2: 22, 22, 22

Bin2: 21, 21, 24

Bin3: 25, 28, 34

Bin3: 29, 29, 29

Bin3: 25, 25, 34

Data Integration

Data Integration

- **Data integration** is the process of combining the data from multiple sources into a coherent data store
- These sources may include multiple databases or flat files
- **Issues to consider during data integration:**
 - Schema integration (entity matching)
 - Data value conflict
 - Redundancy

Schema Integration (Entity Matching)

- **Entity**: Each entity in real-world problem is the attribute in the database
- Addresses the question of
 - “*how can equivalent real-world entities from multiple sources be matched up?*”
 - “*how can data analysts be sure that they are same?*”
- **Attribute name conflict** across the multiple sources of data
 - Example: `customer_id`, `customer_num`, `cust_num`
- **Entity identification problem**:
 - Metadata is associated with each attribute
 - Metadata include:
 - Name,
 - Meaning
 - Data type
 - Range of values permitted

Data Value Conflict

- **Issue**: Detection and resolution of data value conflicts
- For the same real-world entity, attribute values from different sources may differ
- This may be due to difference in representation, scaling, or encoding
- Example:
 - “weight” attribute may be stored in **metric unit (gram, kilogram, etc.)** in one system, **British imperial unit (pound, ounce, etc.)** in another system
 - In a database for hotel chain in different countries:
 - “price of room” attribute may be stored with **price value in different currencies**
 - Categorical data: “gender” may be stored with **male and female** or **M and F**

Redundancy

- Major issue to be addressed
- Sources of redundancy:
 - An attribute may be redundant, if it can be derived from another attribute or set of attributes
 - **Example:** Attribute "Total Marks"
 - **Inconsistency in the attribute naming** can also cause redundancy in resulting data sets
- Two types of redundancies:
 - **Redundancy between the attributes**
 - **Redundancy at the tuple level**
 - Duplication of tuples

Redundancy Between Attributes

- Two attributed may be related or dependent
- Detected by the **correlation analysis**
- **Correlation analysis** measures how strongly one attribute implies (related) to other, based on available data
- Correlation analysis for **numerical attributes**:
 - Compute **correlation coefficient** between two attributes A and B (known as **Pearson's product moment coefficient** or **Pearson's correlation coefficient**)

Redundancy Between Numerical Attributes

- Pearson's correlation coefficient ($\rho_{A,B}$):

$$\rho_{A,B} = \frac{1}{N} \frac{\sum_{i=1}^N (a_i - \mu_A)(b_i - \mu_B)}{\sigma_A \sigma_B}$$

- N : number of tuples
 - a_i and b_i : respective values of A and B in tuple i
 - μ_A and μ_B : respective mean values of A and B
 - σ_A and σ_B : respective standard deviation of A and B
- Note: $-1 \leq \rho_{A,B} \leq +1$

Redundancy Between Numerical Attributes: Pearson's correlation coefficient

- If $\rho_{A,B}$ is greater than 0, then attributes A and B are positively correlated
 - The values of A increases as the values of B increases or vice versa
 - The higher the value, the stronger the correlation
 - A higher correlation value may indicate that A (or B) may be removed as a redundancy
- If $\rho_{A,B}$ is equal to 0, then attributes A and B have no correlation between them (may be independent)
- If $\rho_{A,B}$ is less than 0, then attributes A and B are negatively correlated
 - The values of A increases as the values of B decreases or vice versa
 - Each attribute discourages the other

Redundancy Between Numerical Attributes: Pearson's correlation coefficient

- Scatter plots can also be use to view correlation between the numerical attributes

Redundancy Between Categorical (Discrete) Attributes

- Correlation relationship between two categorical attributes A and B can be discovered by χ^2 (chi-square) test
- Steps in χ^2 (chi-square) test
 - Identify the two categorical attributes
 - Null hypothesis: Two attributes are independent (not related)
 - Complete the contingency matrix (table) with observed and expected frequencies (count)
 - Calculate the observed χ^2 value based on contingency matrix
 - Use the standard χ^2 table compare if the observed χ^2 value to critical χ^2 value for the problem's degree of freedom and confidence (significance i.e. p-value) level
 - If the observed χ^2 value < critical χ^2 value then the attributes are not related (null-hypothesis is true)

Redundancy Between Categorical (Discrete) Attributes

- Correlation relationship between two categorical attributes A and B can be discovered by χ^2 (chi-square) test
- Suppose attribute A has n_A distinct value ($a_1, a_2, \dots, a_i, \dots, a_{n_A}$)
- Suppose attribute B has n_B distinct value ($b_1, b_2, \dots, b_j, \dots, b_{n_B}$)
- The data tuples described by attributes A and B can be shown as a contingency table

- Contingency table has

- n_A distinct values of A making up the rows
- n_B distinct values of B making up the columns

		b_1	b_2	...	b_{n_B}
		1	2	...	n_B
a_1	1			...	
a_2	2			...	
				...	
a_{n_A}	n_A			...	

(A_i, B_j) denote event that i^{th} distinct value of A and j^{th} distinct value of B taken on jointly

Redundancy Between Categorical (Discrete) Attributes

- The observed χ^2 (chi-square) value (Pearson χ^2 statistics) is computed as

$$\chi^2 = \sum_{i=1}^{n_A} \sum_{j=1}^{n_B} \frac{(o_{ij} - e_{ij})^2}{e_{ij}}$$

- o_{ij} : observed frequency (actual count) of joint event (A_i, B_j)
- e_{ij} : expected frequency (count) of joint event (A_i, B_j)
- Expected frequency (e_{ij}) is computed as

$$e_{ij} = \frac{\text{count}(A = a_i) \times \text{count}(B = b_j)}{N}$$

- N : number of tuples
- $\text{Count}(A = a_i)$: The number of tuple having value a_i for A
- $\text{Count}(B = b_j)$: The number of tuple having value b_j for B

Redundancy Between Categorical (Discrete) Attributes

- The χ^2 statistic tests the hypothesis that A and B are independent (Null hypothesis)
- The test is based on the significance level (p-value), with $(n_A-1)*(n_B-1)$ degree of freedom
- If the hypothesis can be rejected, then we say that A and B are statistically related or associated for the given data set

Redundancy Between Categorical Attributes: Illustration

- A group of 1500 people are surveyed
- The gender of each person is noted
- Each person is polled as to whether their preferred type of reading material was fiction or non-fiction
- This leads to two attributes gender and preferred_reading
 - gender takes two distinct values male and female
 - preferred_reading takes two distinct values fiction and non-fiction
- Size of the contingency matrix is 2 x 2

	male (b_1)	female (b_2)	Total
fiction (a_1)	250 (o_{11})	200 (o_{12})	450
non-fiction (a_2)	50 (o_{21})	1000 (o_{22})	1050
Total	300	1200	1500

Redundancy Between Categorical Attributes: Illustration

	male (b_1)	female (b_2)	Total
fiction (a_1)	250 (o_{11}) 90 (e_{11})	200 (o_{12}) 360 (e_{11})	450
non-fiction (a_2)	50 (o_{21}) 210 (e_{11})	1000 (o_{22}) 840 (e_{11})	1050
Total	300	1200	1500

- The numbers in blue are the expected frequencies (count)
- The χ^2 value is computed as

$$\chi^2 = \frac{(o_{11} - e_{11})^2}{e_{11}} + \frac{(o_{12} - e_{12})^2}{e_{12}} + \frac{(o_{21} - e_{21})^2}{e_{21}} + \frac{(o_{22} - e_{22})^2}{e_{22}}$$

$$\chi^2 = \frac{(250 - 90)^2}{90} + \frac{(200 - 360)^2}{360} + \frac{(50 - 210)^2}{210} + \frac{(1000 - 840)^2}{840}$$

$$\chi^2 = 507.93$$

Redundancy Between Categorical Attributes: Illustration

- For 2 x 2 contingency table, the degree of freedom is $(2-1)*(2-1) = 1$
- Obtain the χ^2 value for 0.05 significance i.e. $p=0.05$ (95% chance or confidence) with 1 degree of freedom
 - χ^2 value is 7.879 (Taken from the table of χ^2 distribution)
- Computed χ^2 value for given population is 507.93
- The computed value is above the 7.879
 - We reject the hypothesis that gender and preferred_reading are independent
- Conclusion:** The two attributes (gender and preferred_reading) are strongly correlated for the given group of people

Data Transformation

Data Transformation

- The data are transformed or consolidated into the forms appropriate of data modelling
- Data Transformation involve
 - **Smoothing:**
 - Used for removing noise
 - Techniques: Binning, Regression, Clustering
 - **Aggregation:**
 - Summery or aggregation operation are applied to the data
 - Analysis of data at multiple granularity
 - Example: Daily sales data, Monthly sales data (aggregated on daily data)
 - **Attribute construction (feature construction):**
 - New attributes are constructed from the raw-data to help mining process
 - **Normalization and standardization**

Attribute Normalization

- In the context of machine learning, it is termed as **feature normalization**
- An attribute is normalised by **scaling its value** so that they **fall within a small specified range** (for example 0.0 to 1.0)
- Normalization is particularly useful for classification algorithms involving distance measurements and clustering
- For distance based approaches, normalization **helps prevent attributes with large ranges from overweighting attributes with smaller ranges**

53

Illustration

x_1	x_2
Price	Score for Sale
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2

min: 2300.00 2

max: 73567.00 8

y_1	y_2
23000.00	6.5

$$\text{Euclidean Distance (ED)} = \sum_{i=1}^d (x_i - y_i)^2$$

$$\text{ED1} = (23500.00 - 23000.00)^2 + (8 - 6.5)^2$$

$$\text{ED1} = \mathbf{250002.25}$$

Illustration

x_1	x_2
Price	Score for Sale
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2

y_1	y_2
23000.00	6.5

$$\text{Euclidian Distance (ED)} = \sum_{i=1}^d (x_i - y_i)^2$$

$$\text{ED1} = (23500.00 - 23000.00)^2 + (8 - 6.5)^2$$

$$\text{ED1} = \mathbf{250002.25}$$

$$\text{ED1} = (23500.00 - 23000.00)^2 + (6 - 6.5)^2$$

$$\text{ED1} = \mathbf{250000.25}$$

min: 2300.00 2

max: 73567.00 8

Illustration

x_1	x_2
Price	Score for Sale
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2

y_1	y_2
23000.00	6.5

$$\text{Euclidian Distance (ED)} = \sum_{i=1}^d (x_i - y_i)^2$$

$$\text{ED1} = (23500.00 - 23000.00)^2 + (8 - 6.5)^2$$

$$\text{ED1} = \mathbf{250002.25}$$

$$\text{ED1} = (23500.00 - 23000.00)^2 + (6 - 6.5)^2$$

$$\text{ED1} = \mathbf{250000.25}$$

$$\text{ED3} = (22879.00 - 23000.00)^2 + (2 - 6.5)^2$$

$$\text{ED3} = \mathbf{14661.25}$$



min: 2300.00 2

max: 73567.00 8

Attribute Normalization: Min-Max Normalization

- It performs a **linear transformation** on the original data
- The transformed data is the **scaled version of the original data** so that they **fall within a small specified range**
- Each numeric attributes in a data are normalised separately
- **Steps:**
 - Compute **minimum** (mn_A) and **maximum** (mx_A) values of an attribute A
 - Specify the **new minimum** (new_mn_A) and **new maximum** range (new_mx_A)
 - **Min-Max normalization** maps a value, x of attribute A to \hat{x} in the specified range by computing

$$\hat{x} = \frac{x - mn_A}{mx_A - mn_A} (new_mx_A - new_mn_A) + new_mn_A$$

57

Attribute Normalization: Min-Max Normalization

- When **new minimum** (new_mn_A) and **new maximum** range (new_mx_A) is 0 and 1 respectively, then the data is scaled to 0.0 to 1.0 range
 - **Min-Max normalization** maps a value, x of attribute A to \hat{x} in the 0.0 to 1.0 range by computing

$$\hat{x} = \frac{x - mn_A}{mx_A - mn_A}$$

58

Attribute Normalization: Min-Max Normalization

- Min-Max normalization preserves the relationship among the original data values
- It is useful when data has varying ranges among attributes
- It is useful when machine learning (ML) algorithms we are using does not make any assumption about distribution of data
- It is useful when the actual minimum and maximum values for the attribute is known
- **Disadvantage:** "out-of-bound" error if a future input case for normalization falls outside the original range of attribute A
 - This situation arises when the actual minimum and maximum of attribute A is unknown

59

Illustration of Min-Max Normalization

	Temperature	Humidity	Rain		Temperature	Humidity	Rain
1							
2	25.46875	82.1875	6.75		0.85472	0.00000	0.00128
3	26.19298	83.14912	1762		1.00000	0.05720	1.00000
4	25.17021	85.34043	653		0.79484	0.18753	0.36876
5	24.29851	87.68657	963		0.61998	0.32708	0.54545
6	24.06923	87.64615	254		0.57399	0.32468	0.14213
7	21.20779	95.94805	340		0.00000	0.81847	0.19078
8	23.48571	96.17143	38.3		0.45694	0.83176	0.01921
9	21.79487	98.58974	29.3		0.11776	0.97560	0.01408
10	25.09346	88.3271	4.5		0.77944	0.36518	0.00000
11	25.39423	90.43269	113		0.83978	0.49042	0.06146
12	23.89076	94.53782	736		0.53819	0.73459	0.41613
13	22.5098	99	608		0.26118	1.00000	0.34315
14	22.904	98	718		0.34025	0.94052	0.40589
15	21.72464	99	513		0.10368	1.00000	0.28937

min: 21.20779 82.187 4.5

max: 26.19298 99 1762

0.000 0.000 0.000

1.000 1.000 1.000

Illustration of Min-Max Normalization

Price	Score for Sale
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2



Price	Credit for Sale
0.2975	1.0000
0.2975	0.6667
0.2888	0.0000
0.0000	0.3333
0.4543	0.5000
0.1878	1.0000
0.2336	0.6667
0.0905	0.0000
0.4938	0.3333
1.0000	0.0000
0.7084	0.3333
0.0084	0.1667
0.0599	0.1667
0.2624	0.0000

min: 2300.00 2

max: 73567.00 8

0.000 0.000

1.000 1.000

Illustration of Min-Max Normalization

Price	Score for Sale
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2

23000.00	6.5
----------	-----



0.2905	0.75
--------	------

min: 2300.00 2

max: 73567.00 8

Illustration

x_1	x_2
Price	Credit for Sale
0.2975	1.0000
0.2975	0.6667
0.2888	0.0000
0.0000	0.3333
0.4543	0.5000
0.1878	1.0000
0.2336	0.6667
0.0905	0.0000
0.4938	0.3333
1.0000	0.0000
0.7084	0.3333
0.0084	0.1667
0.0599	0.1667
0.2624	0.0000

y_1	y_2
0.2905	0.75

$$\text{Euclidean Distance (ED)} = \sum_{i=1}^d (x_i - y_i)^2$$

$$\text{ED1} = (0.2975 - 0.2905)^2 + (1 - 0.75)^2$$

$$\text{ED1} = \mathbf{0.06255}$$

min: 0.00 0.00

max: 1.00 1.00

Illustration

x_1	x_2
Price	Credit for Sale
0.2975	1.0000
0.2975	0.6667
0.2888	0.0000
0.0000	0.3333
0.4543	0.5000
0.1878	1.0000
0.2336	0.6667
0.0905	0.0000
0.4938	0.3333
1.0000	0.0000
0.7084	0.3333
0.0084	0.1667
0.0599	0.1667
0.2624	0.0000

y_1	y_2
0.2905	0.75

$$\text{Euclidean Distance (ED)} = \sum_{i=1}^d (x_i - y_i)^2$$

$$\text{ED1} = (0.2975 - 0.2905)^2 + (1 - 0.75)^2$$

$$\text{ED1} = \mathbf{0.06255}$$

$$\text{ED2} = (0.2975 - 0.2905)^2 + (0.6667 - 0.75)^2$$

$$\text{ED2} = \mathbf{0.00699}$$

min: 0.00 0.00

max: 1.00 1.00

Illustration

x_1	x_2
Price	Credit for Sale
0.2975	1.0000
0.2975	0.6667
0.2888	0.0000
0.0000	0.3333
0.4543	0.5000
0.1878	1.0000
0.2336	0.6667
0.0905	0.0000
0.4938	0.3333
1.0000	0.0000
0.7084	0.3333
0.0084	0.1667
0.0599	0.1667
0.2624	0.0000

$\min:$ 0.00 0.00
 $\max:$ 1.00 1.00

y_1	y_2
0.2905	0.75

$$\text{Euclidean Distance (ED)} = \sum_{i=1}^d (x_i - y_i)^2$$

$$\text{ED1} = (0.2975 - 0.2905)^2 + (1.0 - 0.75)^2$$

$$\text{ED1} = \mathbf{0.06255}$$

$$\text{ED2} = (0.2975 - 0.2905)^2 + (0.6667 - 0.75)^2$$

$$\text{ED2} = \mathbf{0.00699}$$

$$\text{ED3} = (0.2888 - 0.2905)^2 + (0.0 - 0.75)^2$$

$$\text{ED2} = \mathbf{0.56250}$$



Data Standardization (z-score Normalization)

- The process of rescaling one or more attributes so that the transformed data have **0 mean** and **unit variance** i.e. **standard deviation of 1**
- Standardization assumes that data has a **Gaussian distribution**
 - This assumption does not strictly have to be true, but this technique is more effective if your attribute distribution is Gaussian
- In this process, values of an attribute, A, are normalised based on the **mean** and **standard deviation** of A
- A value, x , of attribute A is normalised to \hat{x} by computing

$$\hat{x} = \frac{x - \mu_A}{\sigma_A}$$

• μ_A : mean of attribute A

• σ_A : standard deviation of attribute A

Data Standardization (z-score Normalization)

- This method of normalization is useful
 - when the actual minimum and maximum of attribute A are unknown
 - when there are outliers that dominates the Min-Max normalization
 - when data has Gaussian distribution (symmetric distribution)
- This method of normalization is useful when the ML algorithms make any assumptions of Gaussian distribution

Illustration of Data Standardization (z-score Normalization)

	Temperature	Humidity	Rain		Temperature	Humidity	Rain
1							
2	25.46875	82.1875	6.75		1.05444	-1.57673	-0.97166
3	26.19298	83.14912	1762		1.51216	-1.41995	2.62269
4	25.17021	85.34043	653		0.86576	-1.06268	0.35088
5	24.29851	87.68657	963		0.31484	-0.68016	0.98680
6	24.06923	87.64615	254		0.16993	-0.68675	-0.46476
7	21.20779	95.94805	340		-1.63853	0.66679	-0.28965
8	23.48571	96.17143	38.3		-0.19886	0.70321	-0.90714
9	21.79487	98.58974	29.3		-1.26749	1.09749	-0.92558
10	25.09346	88.3271	4.5		0.81726	-0.57573	-0.97627
11	25.39423	90.43269	113		1.00735	-0.23244	-0.75508
12	23.89076	94.53782	736		0.05714	0.43686	0.52138
13	22.5098	99	608		-0.81564	1.16438	0.25871
14	22.904	98	718		-0.56650	1.00134	0.48451
15	21.72464	99	513		-1.31187	1.16438	0.06517

μ : 23.80035 91.86 481

σ : 1.58225 6.13 488

0.000 0.000 0.000

1 1 1

Data Reduction

Data Reduction

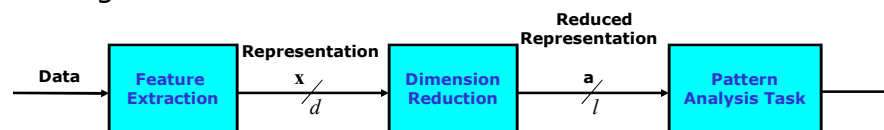
- Data reduction techniques are applied to obtain a reduced representation of the dataset that is much smaller in volume, yet closely maintain the integrity of the original data
- The mining on the reduced dataset should produce the same or almost same analytical results
- Different strategies:
 - Attribute subset selection (feature selection):
 - Irrelevant, weakly relevant or redundant attributes (dimensions) are detected and removed
 - Dimensionality reduction:
 - Encoding mechanisms are used to reduce the dataset size

Attribute (Feature) Subset Section

- In the context of machine learning, it is termed as **feature subset selection**
- Irrelevant or redundant features are detected using **correlation analysis**
- Two strategies:
 - **First strategy:**
 - Perform the **correlation analysis between every pair of attributes**
 - Drop one among the two attributes when they are highly correlated
 - **Second strategy:**
 - Perform the **correlation analysis between each attribute and target attribute**
 - Drop the attributes that are less correlated with target attribute.

Dimensionality Reduction

- Data encoding or transformations are applied so as to obtain a **reduced** or **compressed** representation of the original data



- If the original data can be reconstructed from **compressed data without any loss of information**, the data reduction is called **lossless**
- If **only an approximation of the original data** can be reconstructed from compressed data, then the data reduction is called **lossy**
- One of the popular and effective methods of lossy dimensionality reduction is **principal component analysis (PCA)**

Tuple (Data Vector) – Attribute (Dimension)

Temperature	Humidity	Pressure	Rain	Moisture
25.47	82.19	1036.35	6.75	0.00
26.19	83.15	1037.60	1761.75	5.69
25.17	85.34	1037.89	652.50	6.85
24.30	87.69	1036.86	963.00	6.04
24.07	87.65	1027.83	254.25	31.24
21.21	95.95	1006.92	339.75	100.00
23.49	96.17	1006.57	38.25	93.20
21.79	98.59	1009.42	29.25	5.77
25.09	88.33	991.65	4.50	4.29
25.39	90.43	1009.66	112.50	3.62
23.89	94.54	1009.27	735.75	3.76
22.51	99.00	1009.80	607.50	4.03
22.90	98.00	1009.90	717.75	3.83
21.72	99.00	996.29	513.00	3.04
23.18	98.97	800.00	195.75	3.00
21.24	99.00	1009.21	474.75	3.05
21.63	99.00	1008.89	409.50	3.00
20.91	99.00	1008.89	1161.00	3.20
23.67	97.80	1009.38	0.00	2.04
24.53	92.90	1008.66	0.00	1.80

- A tuple (one row) is referred as a **vector**
- Attribute is referred as **dimension**
- In this example:
 - Number of vectors = number of rows = **20**
 - Dimension of a vector = number of attributes = **5**
 - Size of data matrix is **20x5**

Tuple (Data Vector)

73

Principal Component Analysis (PCA)

- Suppose data to be reduced consist of N tuples (or **data vectors**) described by d -attributes (d - dimensions)

$$\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N, \mathbf{x}_n \in \mathbb{R}^d$$

- Let \mathbf{q}_i , where $i = 1, 2, \dots, d$ be the d **orthonormal vectors** in the d -dimensional space, $\mathbf{q}_i \in \mathbb{R}^d$
 - These are unit vectors that each point in a direction perpendicular to the others

$$\mathbf{q}_i^T \mathbf{q}_j = 0 \quad \forall i \neq j$$

$$\mathbf{q}_i^T \mathbf{q}_i = 1$$

- PCA searches for l **orthonormal vectors** that can best be used to represent the data, where $l < d$

Principal Component Analysis (PCA)

- These orthonormal vectors are also called as **direction of projection**
- The original data (each of the tuples (data vectors), \mathbf{x}_n) is then projected onto each of the l **orthonormal vectors** get the **principal components**

$$a_{ni} = \mathbf{q}_i^T \mathbf{x}_n \quad \forall i = 1, 2, \dots, l$$

– a_{ni} is an i^{th} **principal component** of \mathbf{x}_n

- This transform each of the d – dimensional vectors (i.e. tuples) to l – dimensional vectors

$$\mathbf{x}_n = \begin{bmatrix} x_{n1} \\ x_{n2} \\ \dots \\ x_{nd} \end{bmatrix} \rightarrow \mathbf{a}_n = \begin{bmatrix} a_{n1} \\ a_{n2} \\ \dots \\ a_{nl} \end{bmatrix}$$

Principal Component Analysis (PCA)

- Thus the original data is **projected onto much smaller space**, resulting in **dimensionality reduction**
- It combines the essence of attributes by creating an alternative, smaller set of variables (attributes)
- It is possible to **reconstruct the good approximation of original data**, \mathbf{x}_n , as linear combination of the direction of projection, \mathbf{q}_i , and the principal components, a_{ni}

$$\hat{\mathbf{x}}_n = \sum_{i=1}^l a_{ni} \mathbf{q}_i$$

– $\hat{\mathbf{x}}_n$ is approximation of original tuple \mathbf{x}_n

- The **Euclidian distance** between the original and approximated tuples give the **error** in reconstruction

$$\text{Error} = \|\mathbf{x}_n - \hat{\mathbf{x}}_n\| = \sqrt{\sum_{i=1}^d (x_{ni} - \hat{x}_{ni})^2}$$

PCA: Basic Procedure

- **Given:** Data with N samples, $\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N, \mathbf{x}_n \in \mathbb{R}^d$
- Remove mean for each attribute (dimension) in data samples (tuples)
- Then construct a data matrix \mathbf{X} using the mean subtracted samples, $\mathbf{X} \in \mathbb{R}^{N \times d}$
 - Each row of the matrix \mathbf{X} corresponds to 1 sample (tuple or a data vector)
- Compute a correlation matrix $\mathbf{C} = \mathbf{X}^T \mathbf{X}$
- Perform the **eigen analysis** of correlation matrix \mathbf{C}

$$\mathbf{C}\mathbf{q}_i = \lambda_i \mathbf{q}_i \quad \forall i = 1, 2, \dots, d$$

- As correlation matrix is symmetric matrix,
 - Each eigenvalues λ_i are distinct and non-negative.
 - Eigenvectors \mathbf{q}_i corresponding to each eigenvalues are orthonormal vectors
 - Eigenvalues indicate the **variance or strength** of eigenvectors

77

PCA: Basic Procedure

- Project the \mathbf{x}_n onto each of the directions (eigenvectors) to get the **principal components**

$$a_{ni} = \mathbf{q}_i^T \mathbf{x}_n \quad \forall i = 1, 2, \dots, d$$
 - a_{ni} is an i^{th} **principal component** of \mathbf{x}_n
- Thus, each training example \mathbf{x}_n is transformed to a new representation \mathbf{a}_n by projecting on to d -orthonormal basis (eigenvectors)

$$\mathbf{x}_n = \begin{bmatrix} x_{n1} \\ x_{n2} \\ \dots \\ x_{nd} \end{bmatrix} \rightarrow \mathbf{a}_n = \begin{bmatrix} a_{n1} \\ a_{n2} \\ \dots \\ a_{nd} \end{bmatrix}$$

- It is possible to **reconstruct the original data**, \mathbf{x}_n , without error as linear combination of the direction of projection, \mathbf{q}_i , and the principal components, a_{ni}

$$\mathbf{x}_n = \sum_{i=1}^d a_{ni} \mathbf{q}_i$$

78

PCA for Dimension Reduction

- In general, we are interested in representing the data using fewer dimensions **such that the data has high variance along these dimensions**
- **Idea:** Select l out of d orthonormal basis vectors (eigenvectors) that contain high variance of data (i.e. more information content)
- Rank order the eigenvalues (λ_i 's) such that

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$$
- Based on the **Definition 1**, consider the l ($l \ll d$) eigenvectors corresponding to l significant eigenvalues
 - **Definition 1:** Let $\lambda_1, \lambda_2, \dots, \lambda_d$ be the eigenvalues of an $d \times d$ matrix A . λ_1 is called the dominant (significant) eigenvalue of A if $|\lambda_1| \geq |\lambda_i|, i = 1, 2, \dots, d$

79

PCA for Dimension Reduction

- Project the \mathbf{x}_n onto each of the l directions (eigenvectors) to get reduced dimensional representation

$$a_{ni} = \mathbf{q}_i^T \mathbf{x}_n \quad \forall i = 1, 2, \dots, l$$

- Thus, each training example \mathbf{x}_n is transformed to a new reduced dimensional representation \mathbf{a}_n by projecting on to l -orthonormal basis vectors (eigenvectors)

$$\mathbf{x}_n = \begin{bmatrix} x_{n1} \\ x_{n2} \\ \dots \\ x_{nd} \end{bmatrix} \rightarrow \mathbf{a}_n = \begin{bmatrix} a_{n1} \\ a_{n2} \\ \dots \\ a_{nl} \end{bmatrix}$$

- The eigenvalue λ_i correspond to the variance of projected data

80

PCA for Dimension Reduction

- Since the strongest l directions are considered for obtaining reduced dimensional representation, it should be possible to reconstruct a good approximation of the original data
- An **approximation of original data**, \mathbf{x}_n , is obtained as linear combination of the direction of projection (strongest eigenvectors), \mathbf{q}_i , and the principal components, a_i

$$\hat{\mathbf{x}}_n = \sum_{i=1}^l a_i \mathbf{q}_i$$

– $\hat{\mathbf{x}}_n$ is approximation of original tuple \mathbf{x}_n

81

Illustration: PCA

Temperature	Humidity	Pressure	Rain	Moisture
25.47	82.19	1036.35	6.75	0.00
26.19	83.15	1037.60	1761.75	5.69
25.17	85.34	1037.89	652.50	6.85
24.30	87.69	1036.86	963.00	6.04
24.07	87.65	1027.83	254.25	31.24
21.21	95.95	1006.92	339.75	100.00
23.49	96.17	1006.57	38.25	93.20
21.79	98.59	1009.42	29.25	5.77
25.09	88.33	991.65	4.50	4.29
25.39	90.43	1009.66	112.50	3.62
23.89	94.54	1009.27	735.75	3.76
22.51	99.00	1009.80	607.50	4.03
22.90	98.00	1009.90	717.75	3.83
21.72	99.00	996.29	513.00	3.04
23.18	98.97	800.00	195.75	3.00
21.24	99.00	1009.21	474.75	3.05
21.63	99.00	1008.89	409.50	3.00
20.91	99.00	1008.89	1161.00	3.20
23.67	97.80	1009.38	0.00	2.04
24.53	92.90	1008.66	0.00	1.80

- **Atmospheric Data:**

– N = Number tuples (data vectors) = 20

– d = Number of attributes (dimension) = 5

- **Mean of each dimension:**

23.42 93.63 1003.55 448.88 14.4

82

Illustration: PCA

Temperature	Humidity	Pressure	Rain	Moisture
2.05	-11.45	32.80	-442.13	-14.37
2.77	-10.49	34.05	1312.88	-8.68
1.75	-8.29	34.34	203.63	-7.52
0.88	-5.95	33.31	514.13	-8.33
0.65	-5.99	24.28	-194.63	16.87
-2.21	2.31	3.37	-109.13	85.63
0.07	2.54	3.02	-410.63	78.83
-1.62	4.96	5.86	-419.63	-8.60
1.68	-5.31	-11.90	-444.38	-10.08
1.98	-3.20	6.11	-336.38	-10.76
0.47	0.90	5.72	286.88	-10.61
-0.91	5.37	6.24	158.63	-10.34
-0.51	4.37	6.34	268.88	-10.54
-1.69	5.37	-7.26	64.13	-11.33
-0.24	5.34	-203.55	-253.13	-11.37
-2.18	5.37	5.65	25.88	-11.32
-1.79	5.37	5.34	-39.38	-11.37
-2.51	5.37	5.34	712.13	-11.18
0.25	4.17	5.83	-448.88	-12.34
1.11	-0.73	5.11	-448.88	-12.57

- Step1: Subtract mean from each attribute

83

Illustration: PCA

- Step2: Compute correlation matrix from the data matrix

50.17	-156.00	268.87	314.10	-183.33
-156.00	666.50	-2224.20	-8746.24	252.92
268.87	-2224.20	47093.53	102982.84	1521.49
314.10	-8746.24	102982.84	4090333.01	-46138.70
-183.33	252.92	1521.49	-46138.70	15811.30

84

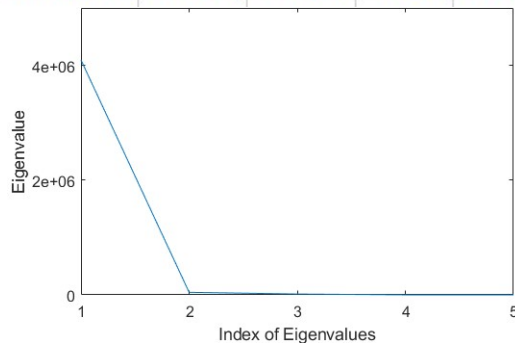
Illustration: PCA

Eigen Values

4093494.12	44809.05	15054.24	587.14	9.95
------------	----------	----------	--------	------

Eigen Vectors

-7.90E-05	0.00559	-0.01372	0.2496	0.96824
0.00215066	-0.04478	0.02318	-0.967	0.24986
-0.0254375	0.99457	-0.08919	-0.0469	0.00509
-0.99961022	-0.02438	0.01358	-0.0007	0.00042
0.01130117	0.09055	0.99556	0.0218	0.00797



- Step4: Perform Eigen analysis on correlation matrix

- Get eigenvalues and eigenvectors

- Step5: Sort the eigenvalues in descending order

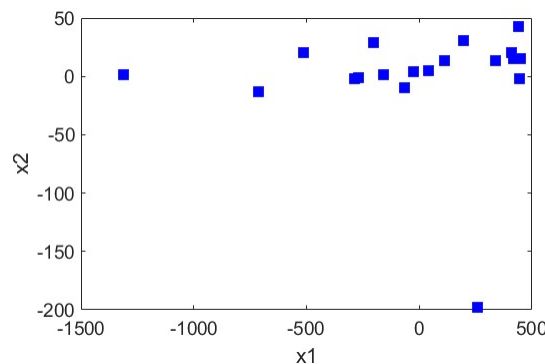
- Step6: Arrange the eigenvectors in the descending order of their corresponding eigenvalues

85

Illustration: PCA

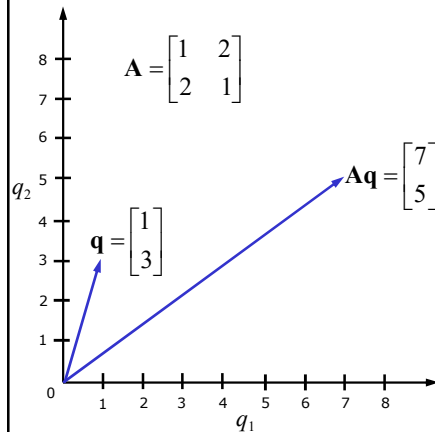
x_1	x_2
440.94	42.62
-1313.36	1.55
-204.53	28.89
-514.88	20.11
194.11	30.69
109.97	13.65
411.29	20.04
419.23	15.05
444.38	-1.67
335.96	13.46
-287.03	-2.30
-158.83	1.16
-269.05	-1.40
-64.04	-10.06
258.09	-197.54
-26.13	3.71
39.11	4.99
-712.10	-13.32
448.43	15.44
448.43	14.93

- Step7: Consider the two leading (significant) eigenvalues and their corresponding eigenvectors
- Step8: Project the mean subtracted data matrix onto the selected two eigenvectors corresponding to leading eigenvalues



86

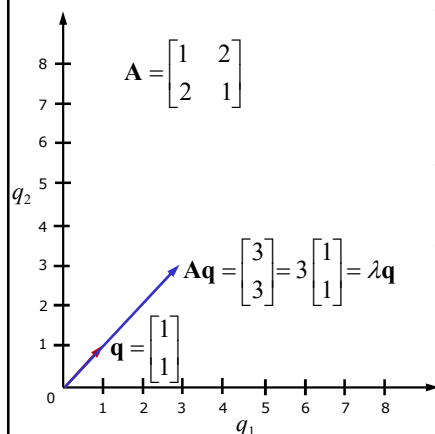
Eigenvalues and Eigenvectors



- What happens when a vector is multiplied with a matrix?
- The vector gets transformed into a new vector
 - Direction changes
- The vector may also get scaled (elongated or shortened) in the process

87

Eigenvalues and Eigenvectors



- For a given square symmetric matrix A , there exist special vectors which do not change direction when multiplied
- These vectors are called eigenvectors
- More formally,

$$A\mathbf{q} = \lambda \mathbf{q}$$
 - λ is eigenvalue
 - Eigenvalue indicate the magnitude of the eigenvector
- The vector will only get scaled but will not change its direction
- So what is so special about eigenvalues and eigenvectors?

88

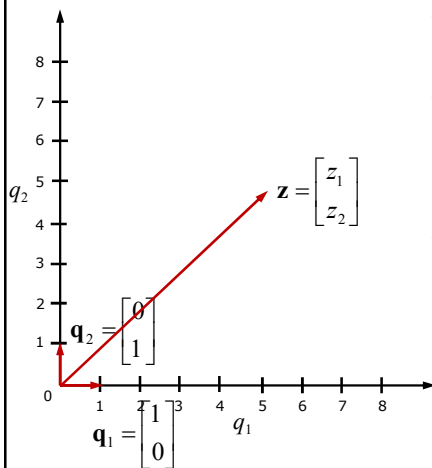
Linear Algebra: Basic Definitions

- **Basis:** A set of vectors $\in \mathbb{R}^d$ is called a **basis**, if
 - those vectors are **linearly independent** and
 - every vector $\in \mathbb{R}^d$ can be expressed as a linear combination of these basis vectors
- **Linearly independent vectors:**
 - A set of d vectors $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_d$ is linearly independent if no vector in the set can be expressed as a linear combination of the remaining $d-1$ vectors
 - In other words, the only solution to

$$c_1\mathbf{q}_1 + c_2\mathbf{q}_2 + \dots + c_d\mathbf{q}_d = \mathbf{0} \text{ is } c_1 = c_2 = \dots = c_d = 0$$
 - Here c_i are scalars

89

Linear Algebra: Basic Definitions



- For example consider the space \mathbb{R}^2
- Consider the vectors:

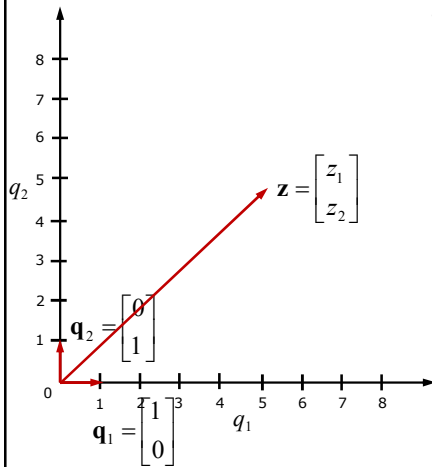
$$\mathbf{q}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \mathbf{q}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
- Any vector $\begin{bmatrix} z_1 & z_2 \end{bmatrix}^T$ can be expressed as a **linear combination of these two vectors**

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = z_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + z_2 \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
- Further, \mathbf{q}_1 and \mathbf{q}_2 are linearly independent
 - The only solution to

$$c_1\mathbf{q}_1 + c_2\mathbf{q}_2 = \mathbf{0} \text{ is } c_1 = c_2 = 0$$

90

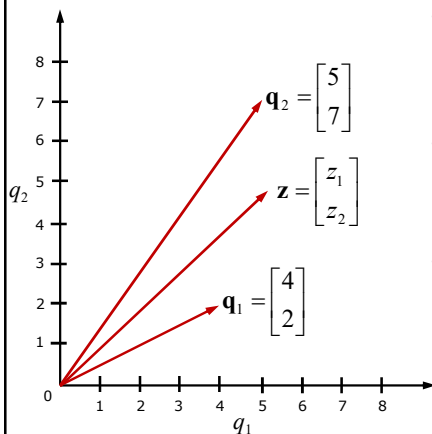
Linear Algebra: Basic Definitions



- It turns out that \mathbf{q}_1 and \mathbf{q}_2 are unit vectors in the direction of the co-ordinate axes
- And indeed we are used to representing all vectors in \mathbb{R}^2 as a linear combination of these two vectors

91

Linear Algebra: Basic Definitions



- We could have chosen any 2 linearly independent vectors in \mathbb{R}^2 as the basis vectors
- For example, consider the linearly independent vectors $[4 \ 2]^T$ and $[5 \ 7]^T$
- Any vector $\mathbf{z} = [z_1 \ z_2]^T$ can be expressed as a linear combination of these two vectors
$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \lambda_1 \begin{bmatrix} 4 \\ 2 \end{bmatrix} + \lambda_2 \begin{bmatrix} 5 \\ 7 \end{bmatrix}$$

$$\mathbf{z} = \lambda_1 \mathbf{q}_1 + \lambda_2 \mathbf{q}_2$$

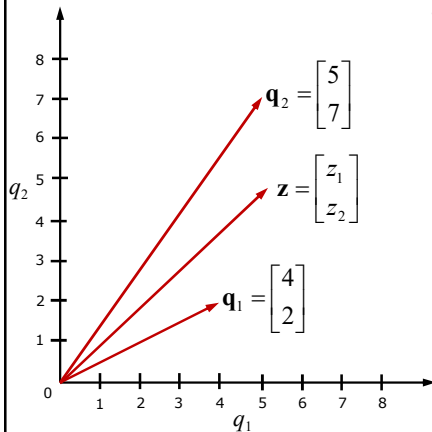
$$z_1 = 4\lambda_1 + 5\lambda_2$$

$$z_2 = 2\lambda_1 + 7\lambda_2$$

- We can find λ_1 and λ_2 by solving a system of linear equations

92

Linear Algebra: Basic Definitions



- In general, given a set of linearly independent vectors

$$\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_d \in \mathbb{R}^d$$

- we can express any vector $\mathbf{z} \in \mathbb{R}^d$ as a linear combination of these vectors

$$\mathbf{z} = \lambda_1 \mathbf{q}_1 + \lambda_2 \mathbf{q}_2 + \dots + \lambda_d \mathbf{q}_d$$

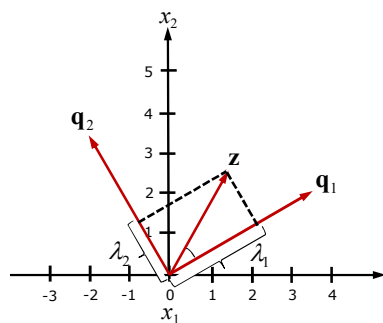
$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_d \end{bmatrix} = \lambda_1 \begin{bmatrix} q_{11} \\ q_{12} \\ \vdots \\ q_{1d} \end{bmatrix} + \lambda_2 \begin{bmatrix} q_{21} \\ q_{22} \\ \vdots \\ q_{2d} \end{bmatrix} + \dots + \lambda_d \begin{bmatrix} q_{d1} \\ q_{d2} \\ \vdots \\ q_{dd} \end{bmatrix}$$

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_d \end{bmatrix} = \begin{bmatrix} q_{11} & q_{21} & \dots & q_{d1} \\ q_{12} & q_{22} & \dots & q_{d2} \\ \dots & \dots & \dots & \dots \\ q_{1d} & q_{2d} & \dots & q_{dd} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_d \end{bmatrix}$$

$$\mathbf{z} = \mathbf{Q} \boldsymbol{\lambda}$$

93

Linear Algebra: Basic Definitions



- Let us see if we have **orthonormal basis**

$$\mathbf{q}_i^T \mathbf{q}_i = 1 \text{ and } \mathbf{q}_i^T \mathbf{q}_j = 0 \forall i \neq j$$

- We can express any vector $\mathbf{z} \in \mathbb{R}^d$ as a linear combination of these vectors

$$\mathbf{z} = \lambda_1 \mathbf{q}_1 + \lambda_2 \mathbf{q}_2 + \dots + \lambda_d \mathbf{q}_d$$

- Multiply \mathbf{q}_1 to both sides

$$\mathbf{q}_1^T \mathbf{z} = \lambda_1 \mathbf{q}_1^T \mathbf{q}_1 + \lambda_2 \mathbf{q}_1^T \mathbf{q}_2 + \dots + \lambda_d \mathbf{q}_1^T \mathbf{q}_d$$

$$\mathbf{q}_1^T \mathbf{z} = \lambda_1$$

- Similarly, $\lambda_2 = \mathbf{q}_2^T \mathbf{z}$

...

$$\lambda_d = \mathbf{q}_d^T \mathbf{z}$$

- An **orthogonal basis** is the most convenient basis that one can hope for

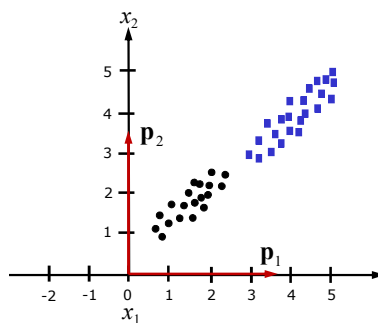
94

Eigenvalues and Eigenvectors

- What does any of this have to do with **eigenvectors**?
- **Eigenvectors can form a basis**
- **Theorem 1**: The eigenvectors of a matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$ having distinct eigenvalues are **linearly independent**
- **Theorem 2**: The eigenvectors of a square symmetric matrix are **orthogonal**
- **Definition 1**: Let $\lambda_1, \lambda_2, \dots, \lambda_d$ be the eigenvalues of an $d \times d$ matrix \mathbf{A} . λ_1 is called the dominant (significant) eigenvalue of \mathbf{A} if $|\lambda_1| \geq |\lambda_i|, i = 1, 2, \dots, d$
- We will put all of this to use for principal component analysis

95

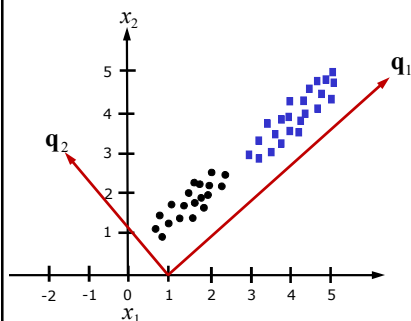
Principal Component Analysis (PCA)



- Each point (vector) here is represented using a linear combination of the x_1 and x_2 axes
- In other words we are using \mathbf{p}_1 and \mathbf{p}_2 as the basis

96

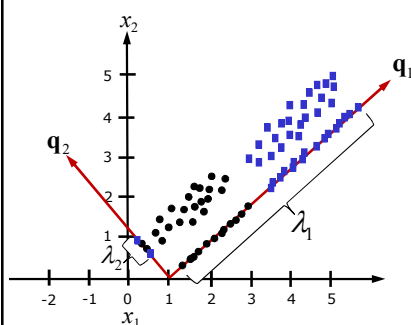
Principal Component Analysis (PCA)



- Lets consider **orthonormal vectors** \mathbf{q}_1 and \mathbf{q}_2 as a basis instead of \mathbf{p}_1 and \mathbf{p}_2 as the basis
- We observe that all the points have a very small component in the direction of \mathbf{q}_2 (almost noise)

97

Principal Component Analysis (PCA)

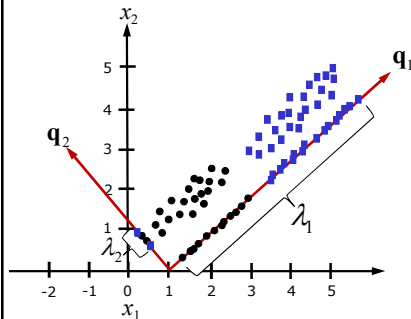


- Lets consider **orthonormal vectors** \mathbf{q}_1 and \mathbf{q}_2 as a basis instead of \mathbf{p}_1 and \mathbf{p}_2 as the basis
- We observe that all the points have a very small component in the direction of \mathbf{q}_2 (almost noise)

- Now the same data can be represented in 1-dimension in the direction of \mathbf{q}_1 by making a smarter choice for the basis
- Why do we not care about \mathbf{q}_2 ?
 - Variance in the data in this direction is very small
 - All data points have almost the same value in the \mathbf{q}_2 direction

98

Principal Component Analysis (PCA)



- If we were to build a classifier on top of this data then \mathbf{q}_2 would not contribute to the classifier

– The points are not distinguishable along this direction

- In general, we are interested in representing the data using fewer dimensions **such that**
 - the data has high variance along these dimensions
 - the dimensions are linearly independent (uncorrelated)

99

PCA: Basic Procedure

- **Given:** Data with N samples, $\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N, \mathbf{x}_n \in \mathbb{R}^d$
- 1. Remove mean for each attribute (dimension) in data samples (tuples)
- 2. Then construct a data matrix \mathbf{X} using the mean subtracted samples, $\mathbf{X} \in \mathbb{R}^{N \times d}$
 - Each row of the matrix \mathbf{X} corresponds to 1 sample (tuple)
- 3. Compute a correlation matrix $\mathbf{C} = \mathbf{X}^T \mathbf{X}$
- 4. Perform the **eigen analysis** of correlation matrix \mathbf{C}

$$\mathbf{C}\mathbf{q}_i = \lambda_i \mathbf{q}_i \quad \forall i = 1, 2, \dots, d$$

- As correlation matrix is **symmetric matrix**,
 - Each eigenvalues λ_i are **distinct and non-negative**
 - Eigenvectors \mathbf{q}_i corresponding to each eigenvalues are **orthonormal vectors**
 - Eigenvalues indicate the **variance or strength** of eigenvectors

100

PCA for Dimension Reduction

- In general, we are interested in representing the data using fewer dimensions **such that the data has high variance along these dimensions**

- Rank order the eigenvalues (λ_i 's) (sorted order) such that

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$$

- Consider the l ($l \ll d$) eigenvectors corresponding to l significant eigenvalues
- Project the \mathbf{x}_n onto each of the l directions (eigenvectors) to get reduced dimensional representation

$$a_{ni} = \mathbf{q}_i^T \mathbf{x}_n \quad \forall i = 1, 2, \dots, l$$

101

PCA for Dimension Reduction

- Thus, each training example \mathbf{x}_n is transformed to a new reduced dimensional representation \mathbf{a}_n by projecting on to l -orthonormal basis

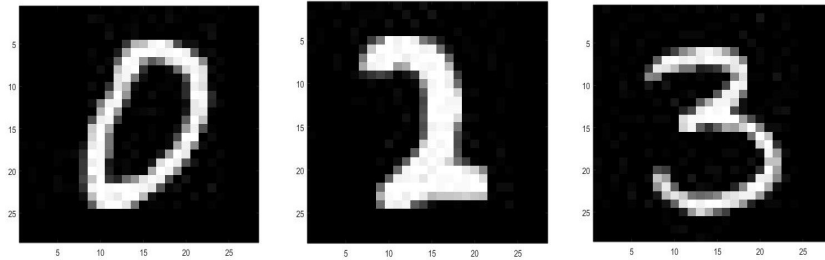
$$\mathbf{x}_n = \begin{bmatrix} x_{n1} \\ x_{n2} \\ \dots \\ x_{nd} \end{bmatrix} \rightarrow \mathbf{a}_n = \begin{bmatrix} a_{n1} \\ a_{n2} \\ \dots \\ a_{nl} \end{bmatrix}$$

- The new reduced representation \mathbf{a}_n is uncorrelated
- The eigenvalue λ_i correspond to the **variance of projected data**

102

Illustration: PCA

- Handwritten Digit Image [1]:
 - Size of each image: 28 x 28
 - Dimension after linearizing: 784
 - Total number of training examples: 5000 (500 per class)



[1] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Intelligent Signal Processing*, 306-351, IEEE Press, 2001

103

Illustration: PCA

- Handwritten Digit Image:
 - All 784 Eigenvalues

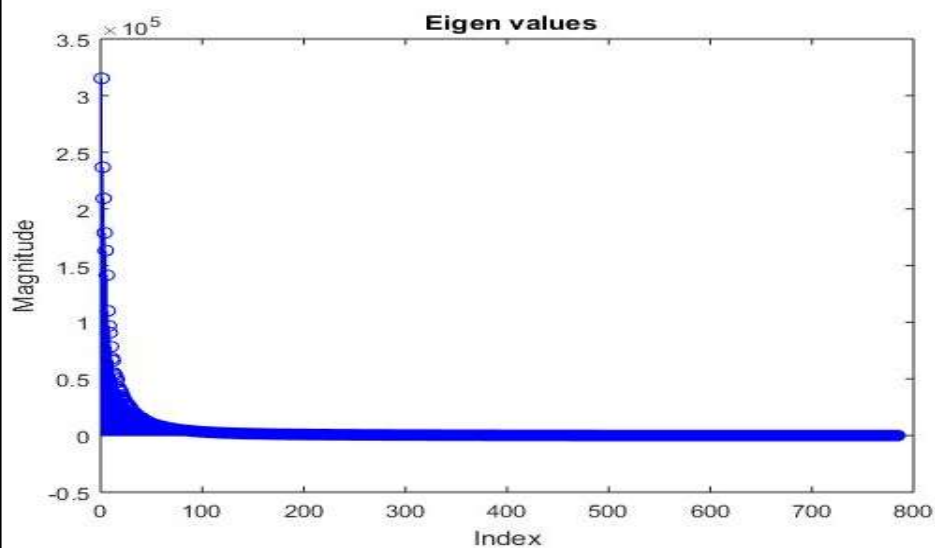


Illustration: PCA

- Handwritten Digit Image:
 - Leading 100 Eigenvalues

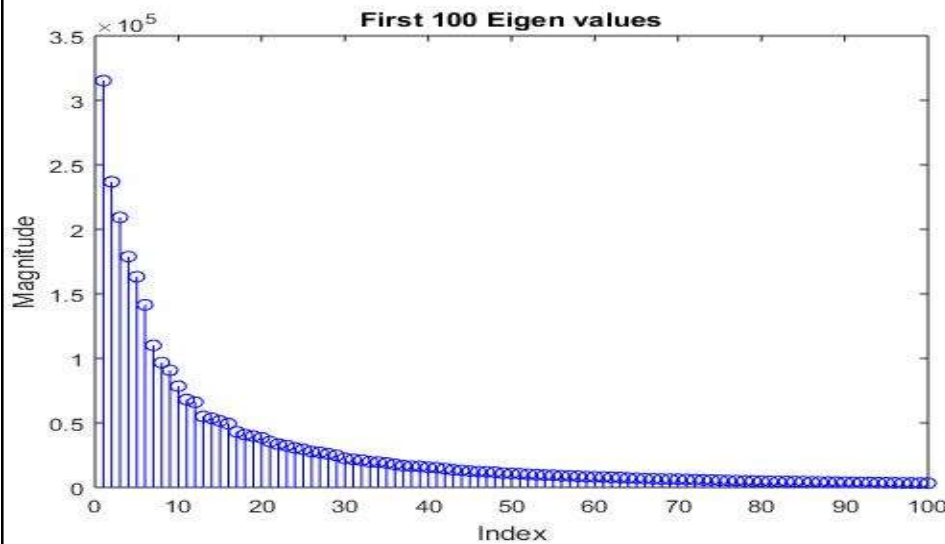


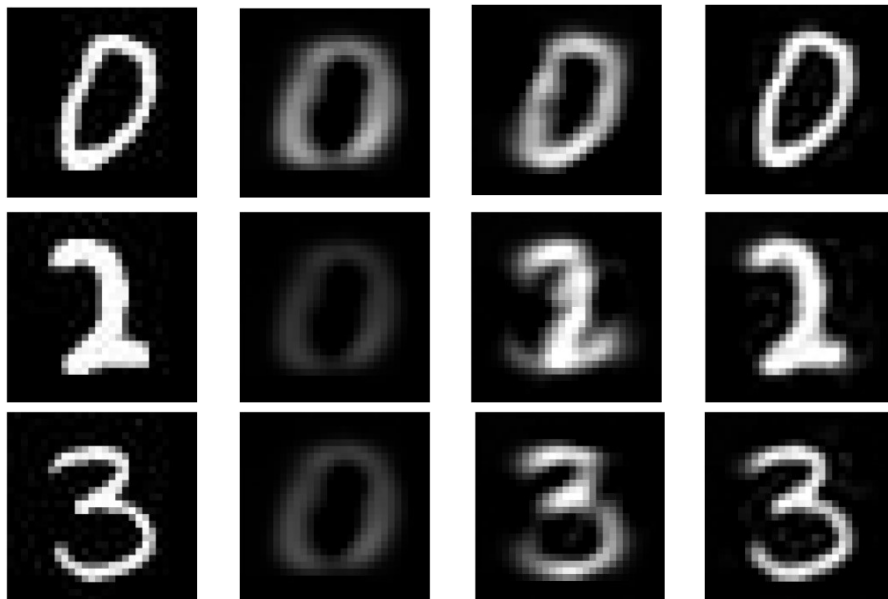
Illustration: PCA-Reconstructed Images

Original Image

$l=1$

$l=20$

$l=100$



106