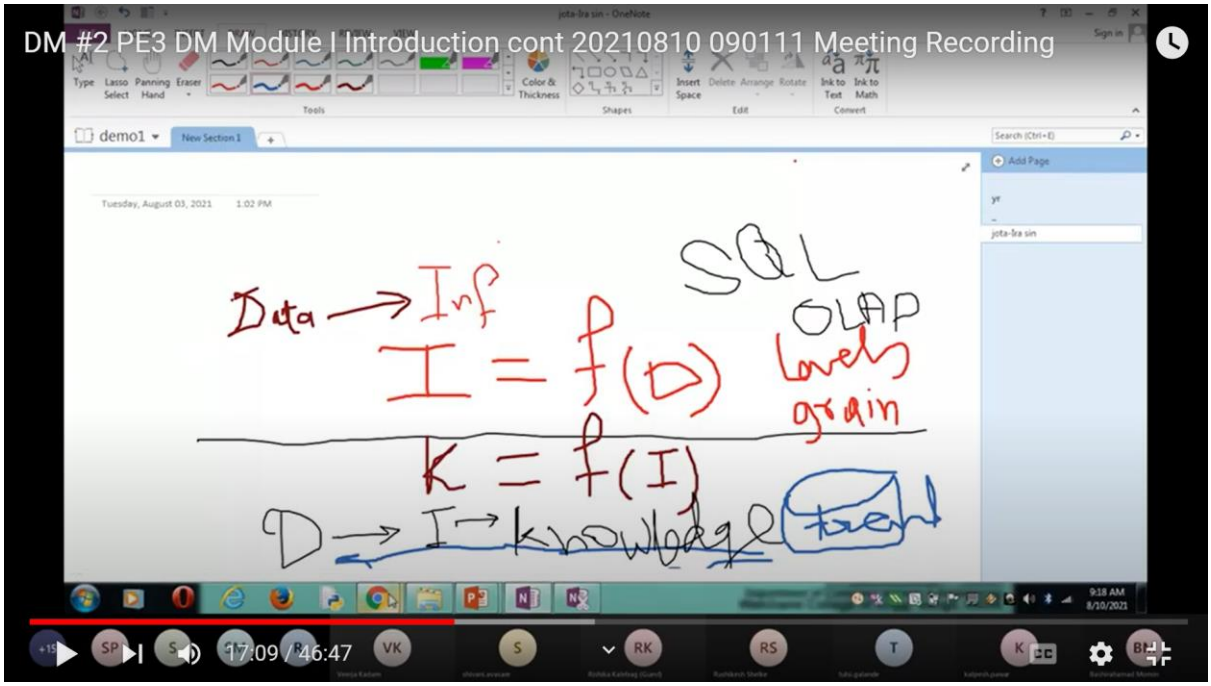
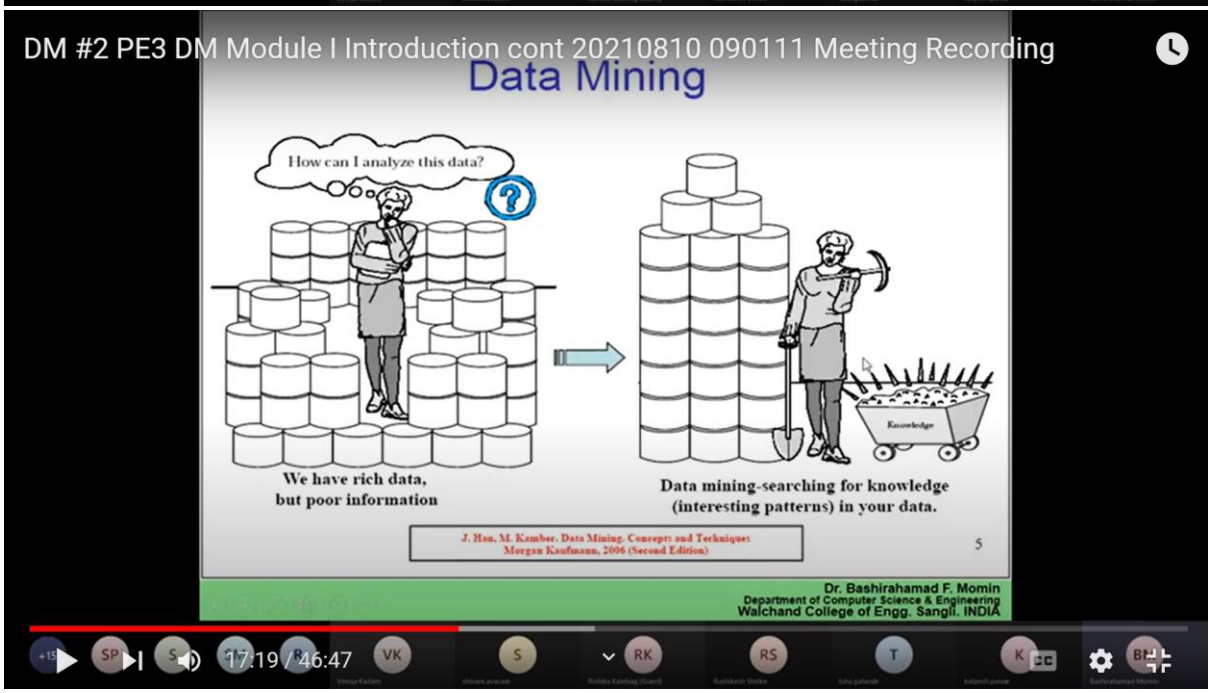

DM #2 PE3 DM Module I Introduction cont 20210810 090111 Meeting Recording
Tuesday, August 03, 2021 1.02 PM
Data → Inf
I = f(D)
K = f(I)
SQL
OLAP
levels
grain
17:09 / 46:47


DM #2 PE3 DM Module I Introduction cont 20210810 090111 Meeting Recording
Data Mining
How can I analyze this data?
We have rich data,
but poor information
Knowledge
Data mining-searching for knowledge
(interesting patterns) in your data.
J. Han, M. Kamber, Data Mining. Concepts and Techniques
Morgan Kaufmann, 2006 (Second Edition)
5
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli. INDIA
17:19 / 46:47

DM #2 PE3 DM Module I Introduction cont 20210810 090111 Meeting Recording

Definition

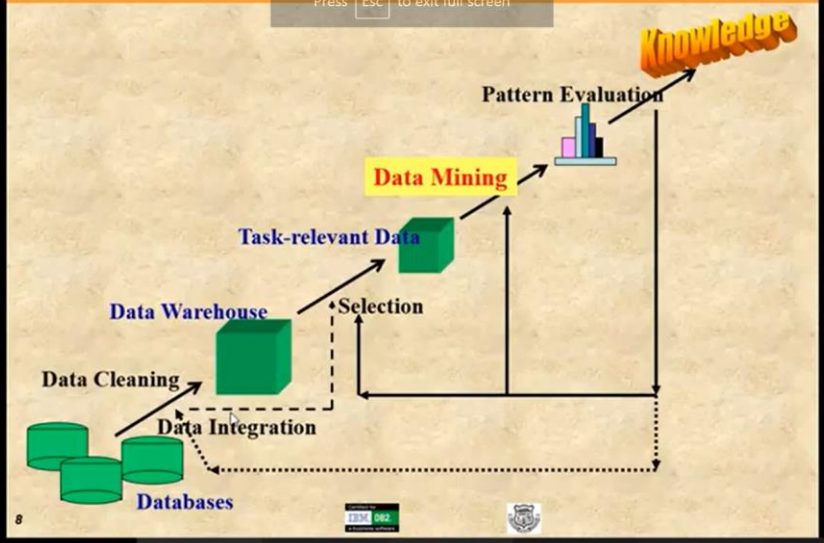
The non-trivial process of analyzing large databases by applying specific algorithm to extract interesting

(hidden, implicit, previously unknown and potentially useful)

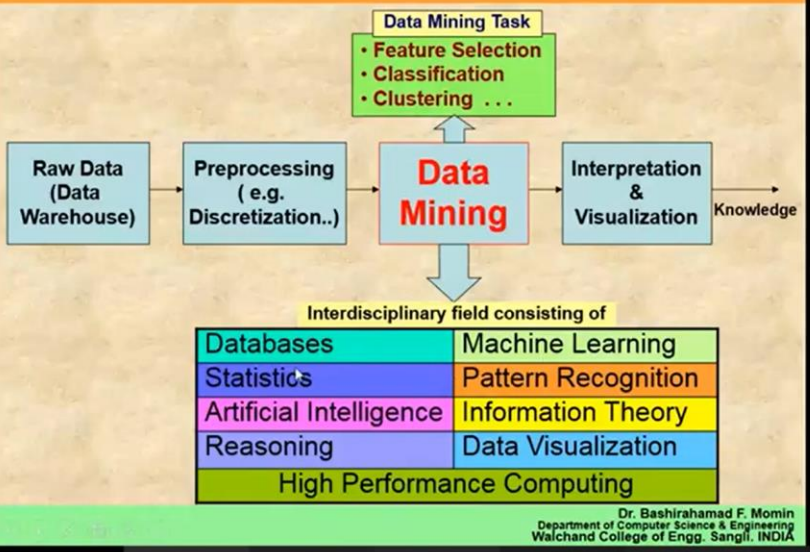
information (knowledge) or patterns from data.

Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli. INDIA

Knowledge Discovery (KDD) Process



Knowledge Discovery in Databases (KDD)

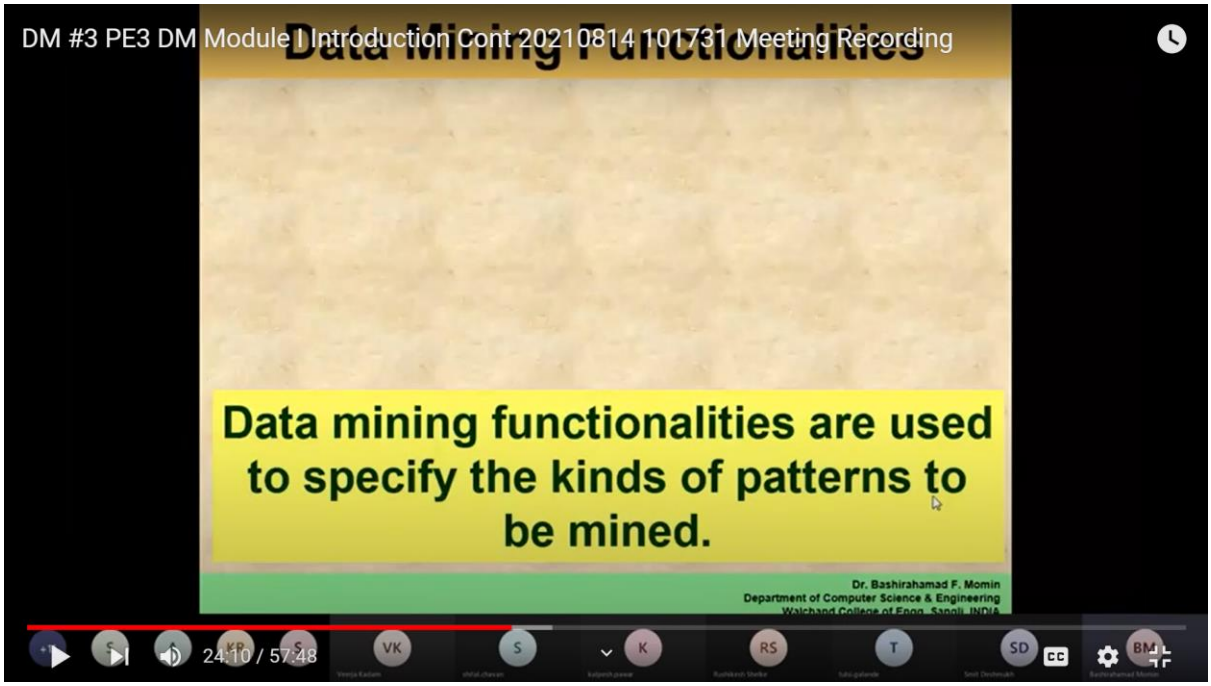


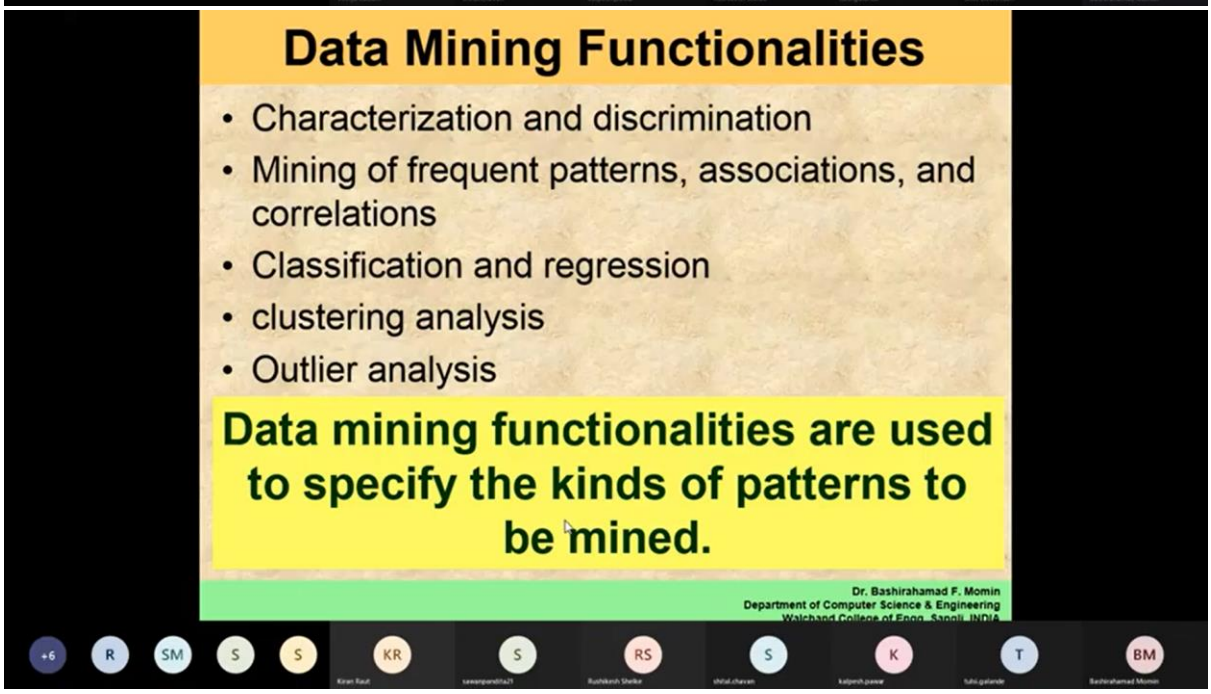
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli. INDIA

Data Mining Domain (Data Types)

- Database-oriented data sets and applications
  - Relational database, data warehouse, transactional database
- Advanced data sets and advanced applications
  - Data streams and sensor data
  - Time-series data, temporal data, sequence data (incl. bio-sequences)
  - Structure data, graphs, social networks and multi-linked data
  - Object-relational databases
  - Heterogeneous databases and legacy databases
  - Spatial data and spatiotemporal data
  - Multimedia database
  - Text databases
  - The World-Wide Web

Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli. INDIA


DM #3 PE3 DM Module I Introduction Cont 20210814 101731 Meeting Recording
Data Mining Functionalities
Data mining functionalities are used to specify the kinds of patterns to be mined.
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
24:10 / 57:48


Data Mining Functionalities
• Characterization and discrimination
• Mining of frequent patterns, associations, and correlations
• Classification and regression
• clustering analysis
• Outlier analysis
Data mining functionalities are used to specify the kinds of patterns to be mined.
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering

Kinds of Patterns

- **Descriptive :**
  - Describes the characteristics of the data in a target data set.
  - It determines, what happened in the past by analyzing stored data.
- **Predictive** :
  - Carry out the induction over the current and past data so that predictions can be made.
  - It determines, what can happen in the future with the help past data analysis.

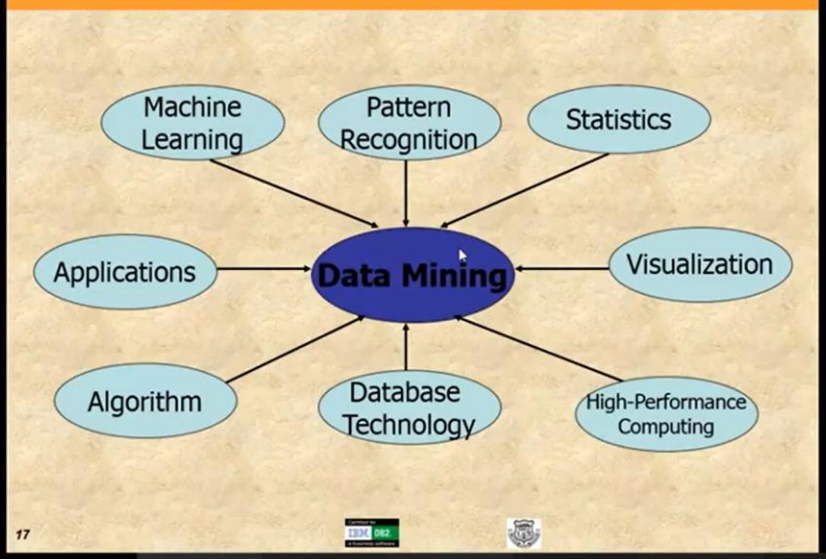
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli, INDIA

Examples

- **Descriptive patterns**
  - Summarizing past events such as sales and operations data or marketing campaigns.
  - Social media usage and engagement data such as Instagram or Facebook likes.
  - Reporting general trends.
  - Collating survey results.
- **Predictive Patterns**
  - Predicting buying behavior in retail. ...
  - Detecting sickness in healthcare. ...
  - Curating content in entertainment. ...
  - Predicting maintenance in manufacturing. ...
  - Detecting fraud in cyber security. ...
  - Predicting employee growth in HR. ...
  - Predicting performance in sports. ...
  - Forecasting patterns in weather.

Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli, INDIA

Data Mining: Inter-disciplinary



17

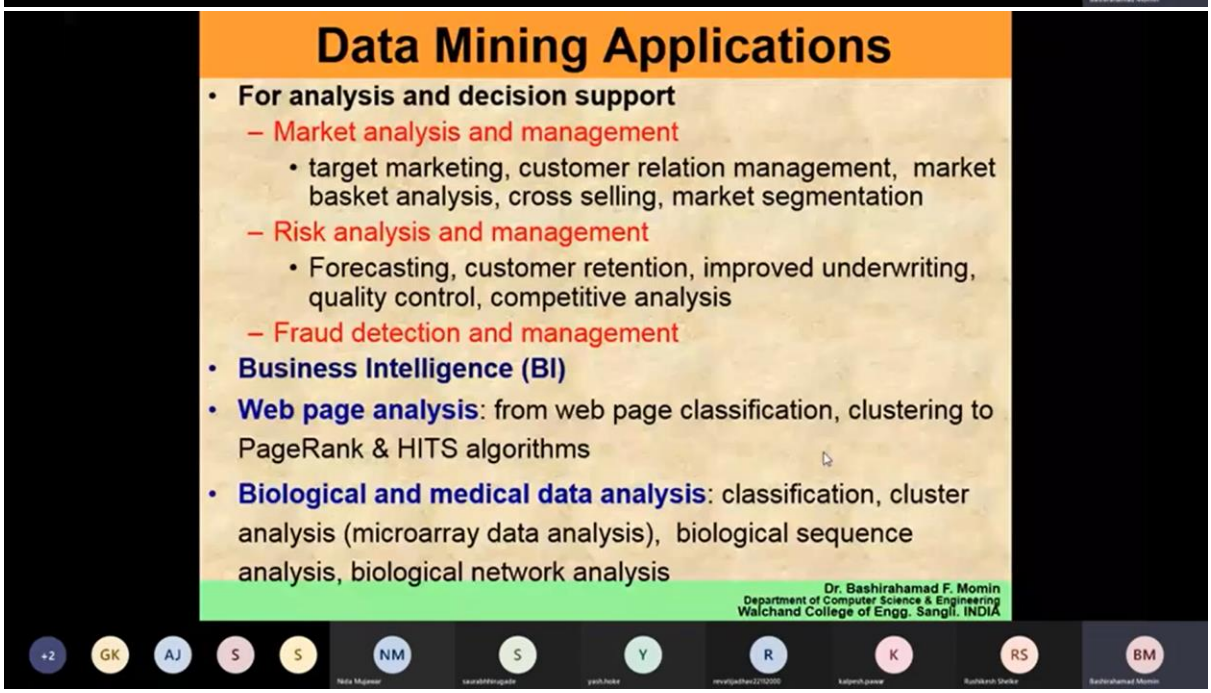
Why Multiple Disciplines?

- **Tremendous amount of data**
  - Algorithms must be highly scalable to handle such as tera-bytes of data
- **High-dimensionality of data**
  - Micro-array may have tens of thousands of dimensions
- **High complexity of data**
  - Data streams and sensor data
  - Time-series data, temporal data, sequence data
  - Structure data, graphs, social networks and multi-linked data
  - Heterogeneous databases and legacy databases
  - Spatial, spatiotemporal, multimedia, text and Web data
  - Software programs, scientific simulations
- **New and sophisticated applications**

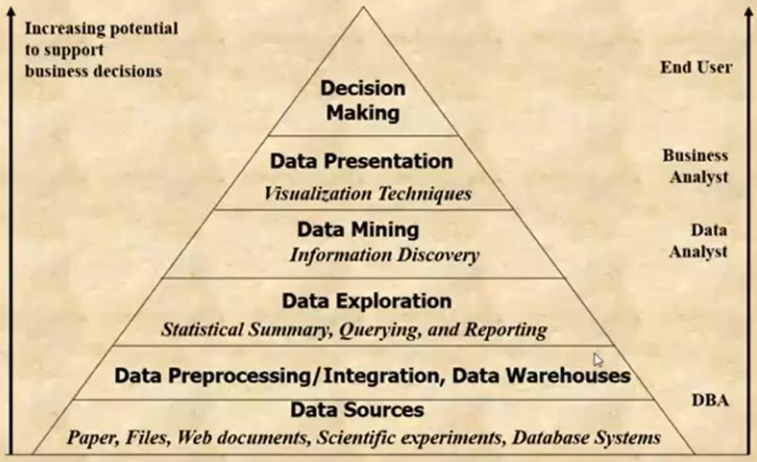
18

Lecture 4:


Which kinds of applications are targeted ?
Where there are data, there are data mining applications.
Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.


Data Mining Applications
For analysis and decision support
Market analysis and management
target marketing, customer relation management, market basket analysis, cross selling, market segmentation
Risk analysis and management
Forecasting, customer retention, improved underwriting, quality control, competitive analysis
Fraud detection and management
Business Intelligence (BI)
Web page analysis: from web page classification, clustering to PageRank & HITS algorithms
Biological and medical data analysis: classification, cluster analysis (microarray data analysis), biological sequence analysis, biological network analysis
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli. INDIA

Data Mining in Business Intelligence



Major Issues in Data Mining

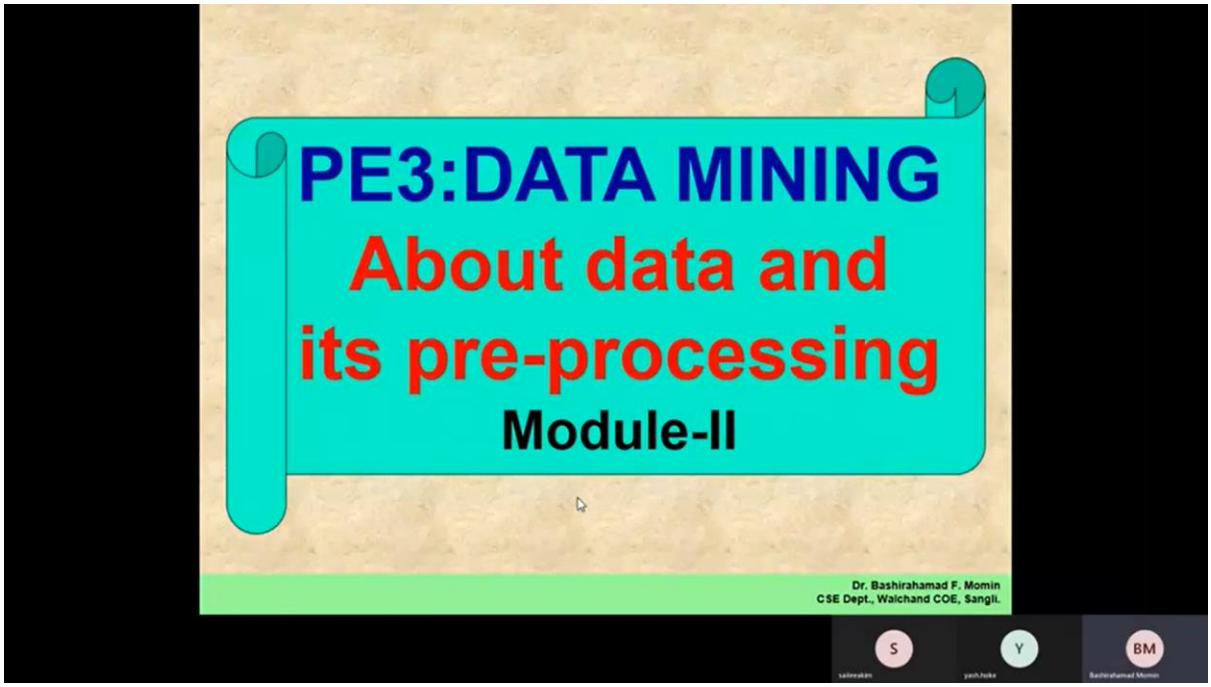
- **Mining Methodology**
  - Mining various and new kinds of knowledge
  - Mining knowledge in multi-dimensional space
  - Data mining: An interdisciplinary effort
  - Boosting the power of discovery in a networked environment
  - Handling noise, uncertainty, and incompleteness of data
  - Pattern evaluation and pattern- or constraint-guided mining
- **User Interaction**
  - Interactive mining
  - Incorporation of background knowledge
  - Presentation and visualization of data mining results

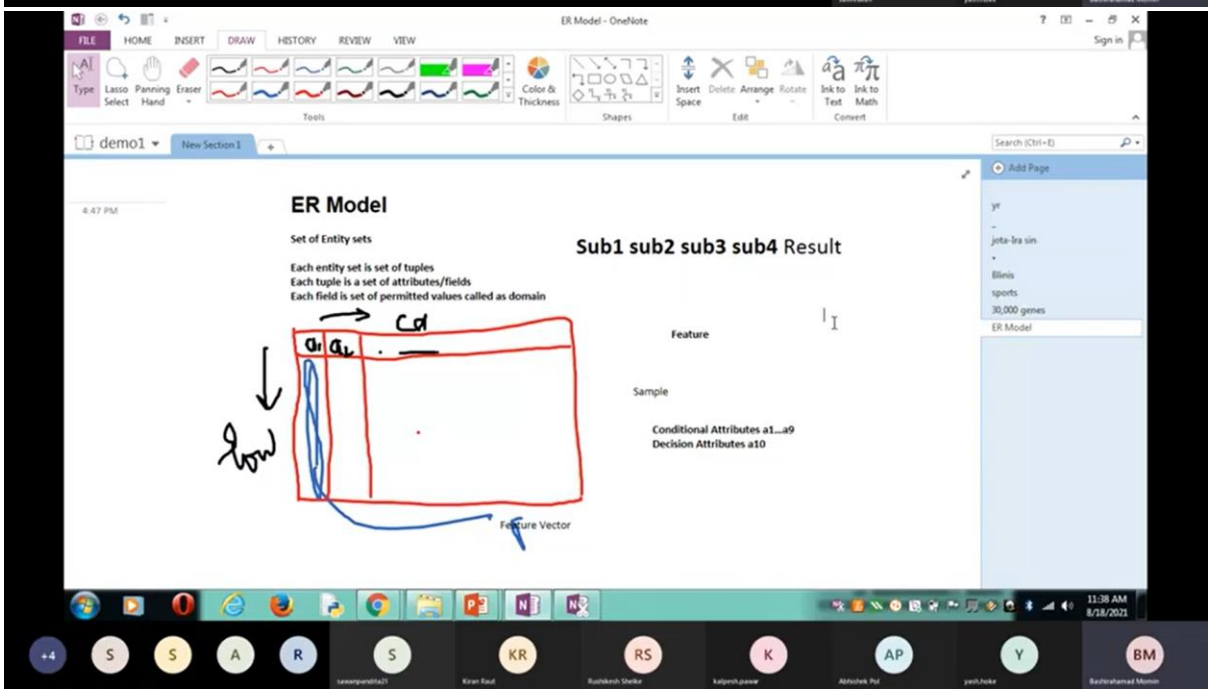
cont... Major Issues in Data Mining

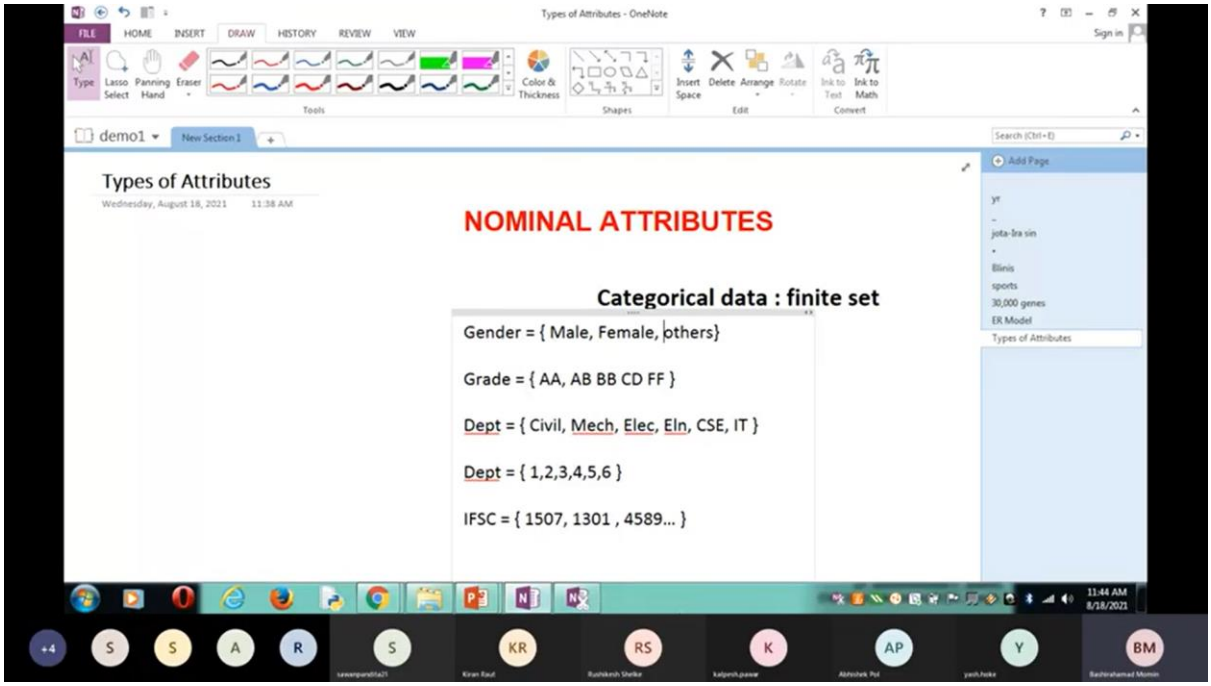
- **Efficiency and Scalability**
  - Efficiency and scalability of data mining algorithms
  - Parallel, distributed, stream, and incremental mining methods
- **Diversity of data types**
  - Handling complex types of data
  - Mining dynamic, networked, and global data repositories
- **Data mining and society**
  - Social impacts of data mining
  - Privacy-preserving data mining
  - Invisible data mining

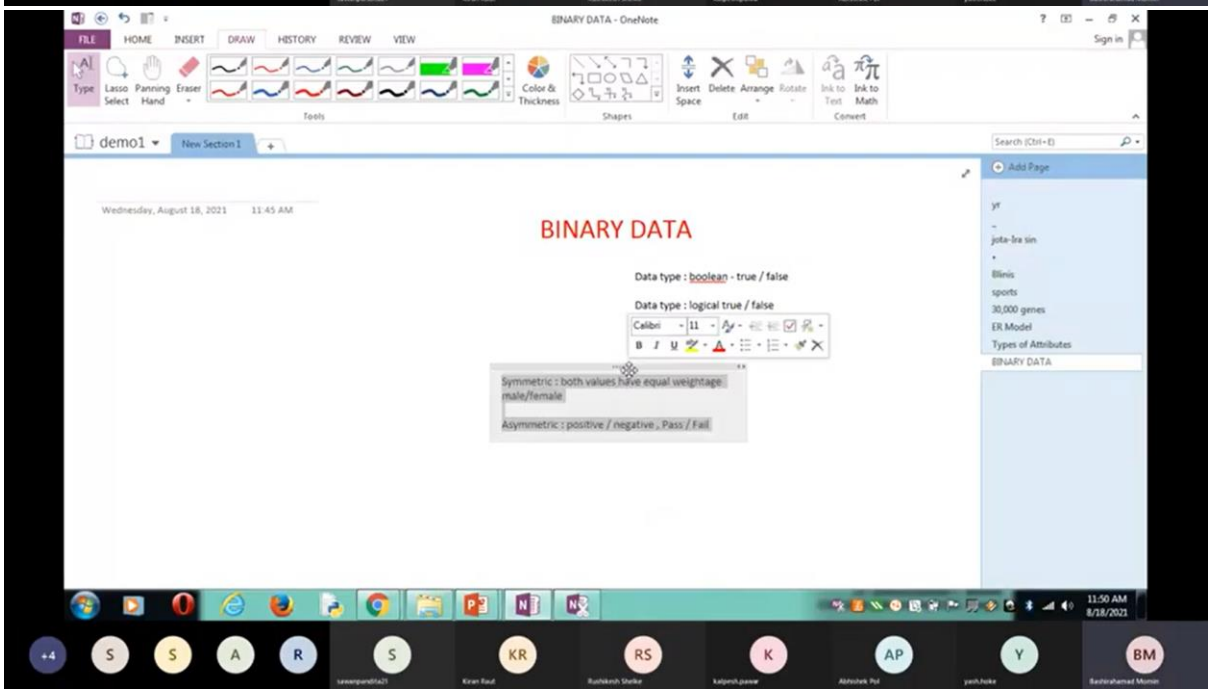
24

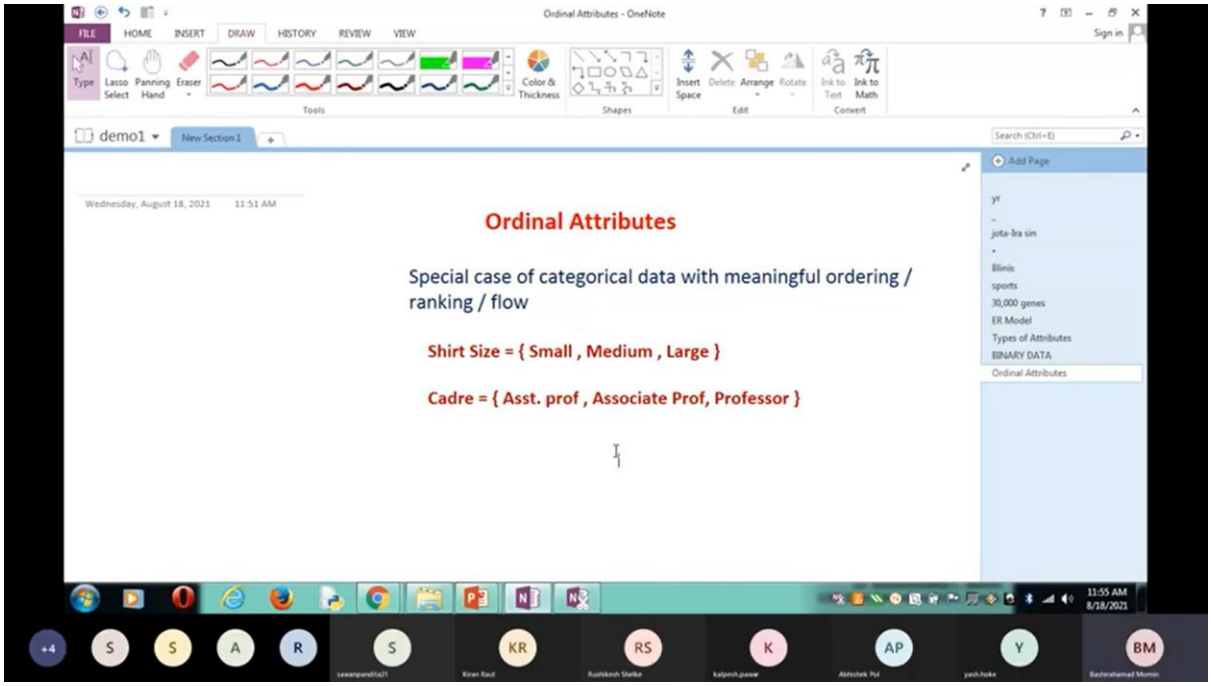
Lecture 5

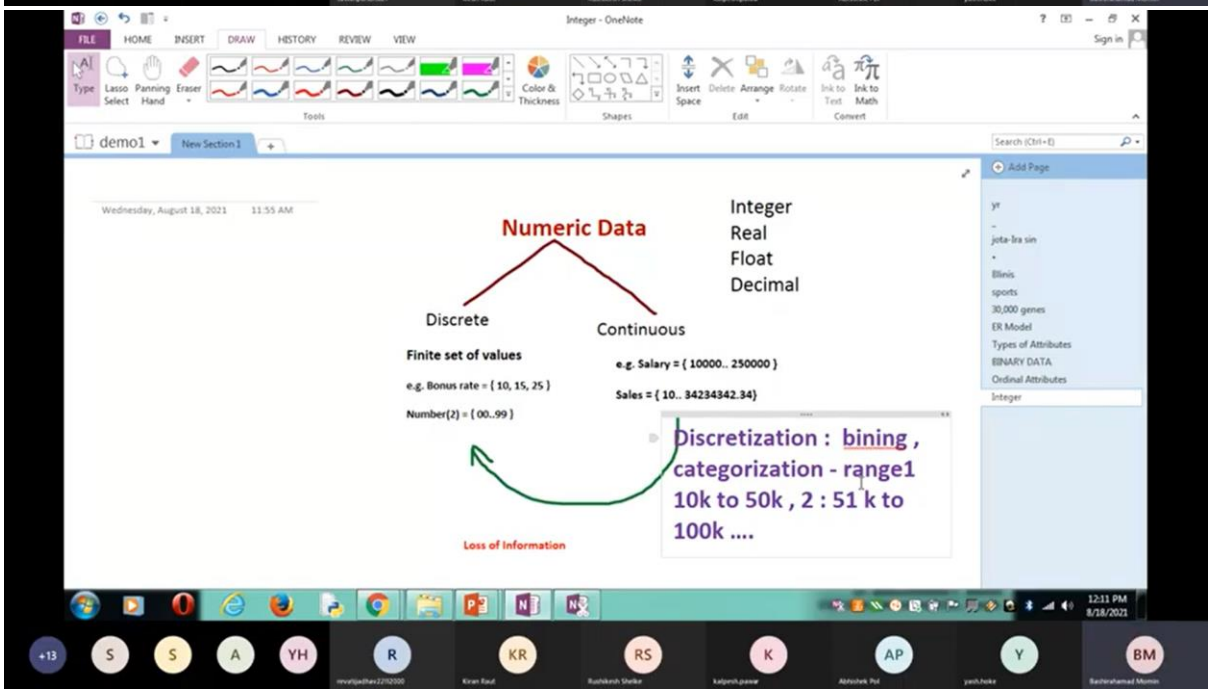

PE3:DATA MINING
About data and
its pre-processing
Module-II
Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.


ER Model - OneNote
ER Model
Set of Entity sets
Each entity set is set of tuples
Each tuple is a set of attributes/fields
Each field is set of permitted values called as domain
Sub1 sub2 sub3 sub4 Result
Feature
Sample
Conditional Attributes a1...a9
Decision Attributes a10
Feature Vector

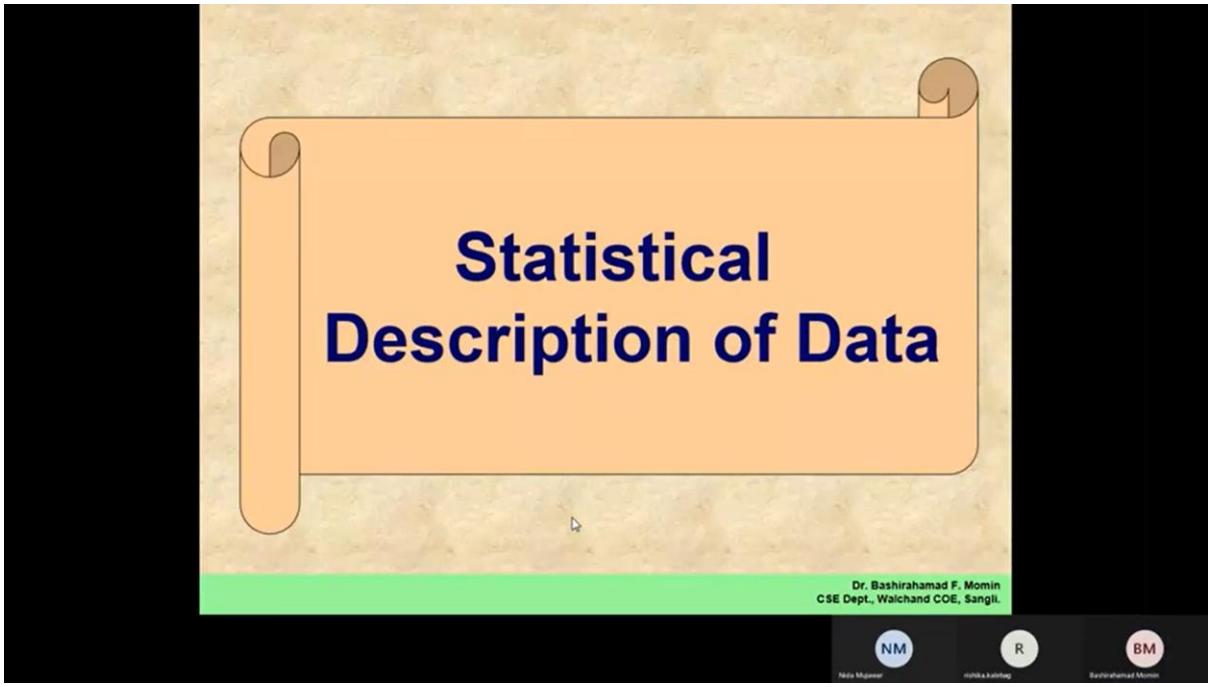

Types of Attributes - OneNote
demo1
New Section 1
Types of Attributes
Wednesday, August 18, 2021 11:38 AM
NOMINAL ATTRIBUTES
Categorical data : finite set
Gender = { Male, Female, others}
Grade = { AA, AB BB CD FF }
Dept = { Civil, Mech, Elec, Eln, CSE, IT }
Dept = { 1,2,3,4,5,6 }
IFSC = { 1507, 1301 , 4589... }
11:44 AM
8/18/2021

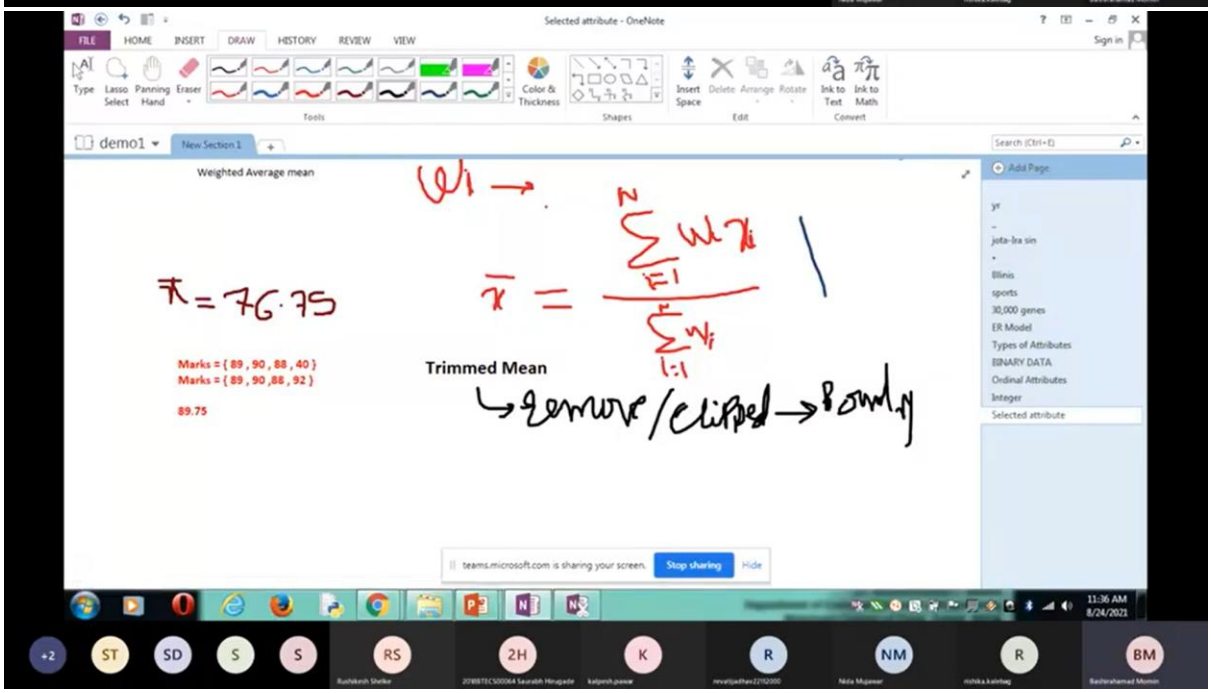

BINARY DATA - OneNote
demo1
New Section 1
Wednesday, August 18, 2021 11:45 AM
BINARY DATA
Data type : boolean - true / false
Data type : logical true / false
Symmetric : both values have equal weightage. male/female
Asymmetric : positive / negative , Pass / Fail
11:50 AM
8/18/2021

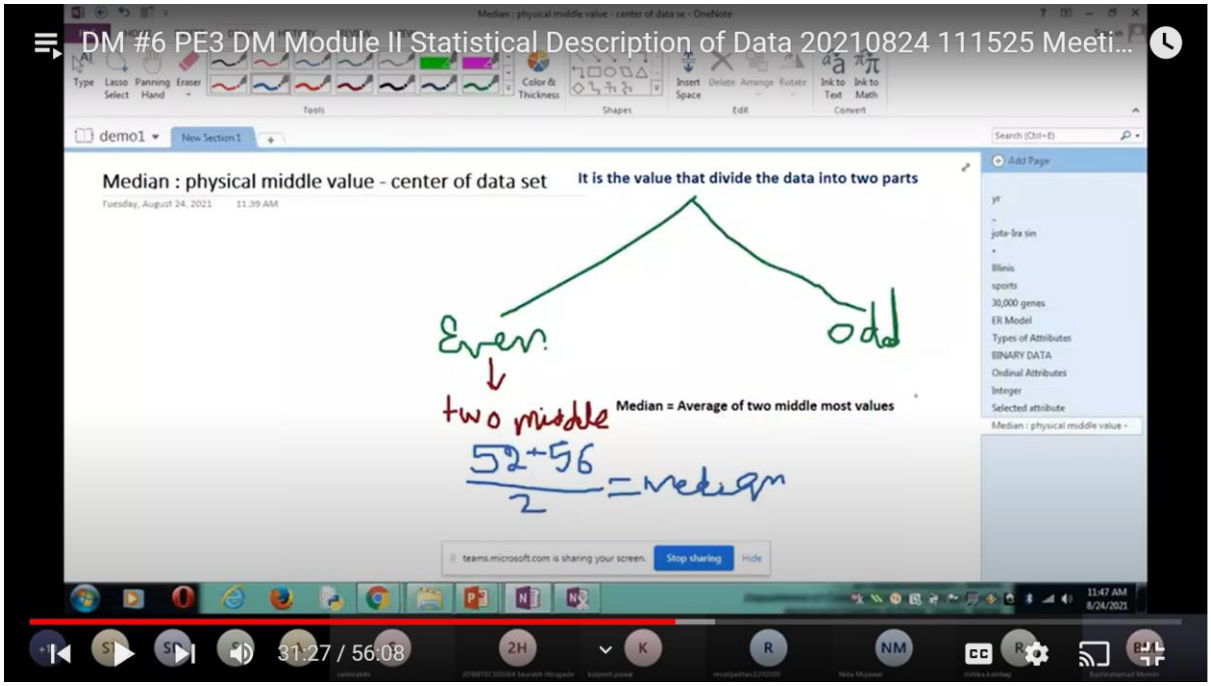

Ordinal Attributes - OneNote
demo1
New Section 1
Wednesday, August 18, 2021 11:51 AM
Ordinal Attributes
Special case of categorical data with meaningful ordering / ranking / flow
Shirt Size = { Small , Medium , Large }
Cadre = { Asst. prof , Associate Prof, Professor }
Types of Attributes
BINARY DATA
Ordinal Attributes
11:55 AM
8/18/2021

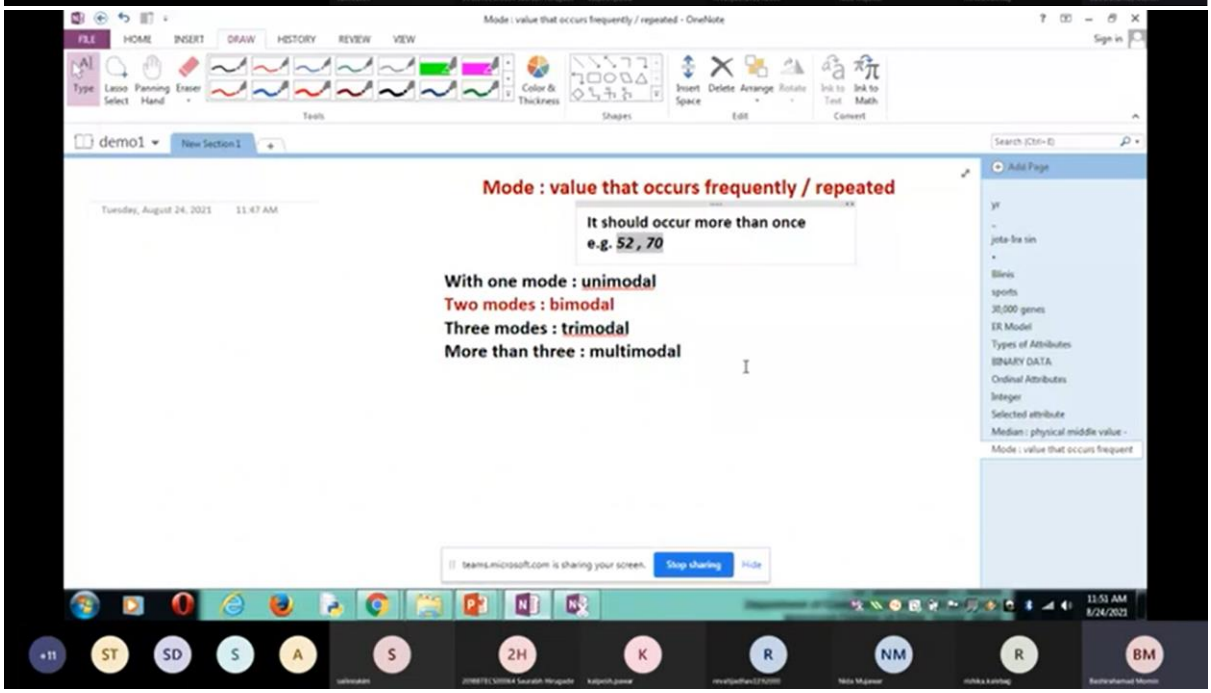

Integer - OneNote
demo1
New Section 1
Wednesday, August 18, 2021 11:55 AM
Numeric Data
Integer
Real
Float
Decimal
Discrete
Continuous
Finite set of values
e.g. Bonus rate = { 10, 15, 25 }
Number(2) = { 00..99 }
e.g. Salary = { 10000.. 250000 }
Sales = { 10.. 34234342.34}
Discretization : bining , categorization - range1 10k to 50k , 2 : 51 k to 100k ....
Loss of Information
Types of Attributes
BINARY DATA
Ordinal Attributes
Integer
12:11 PM
8/18/2021

Lecture 6

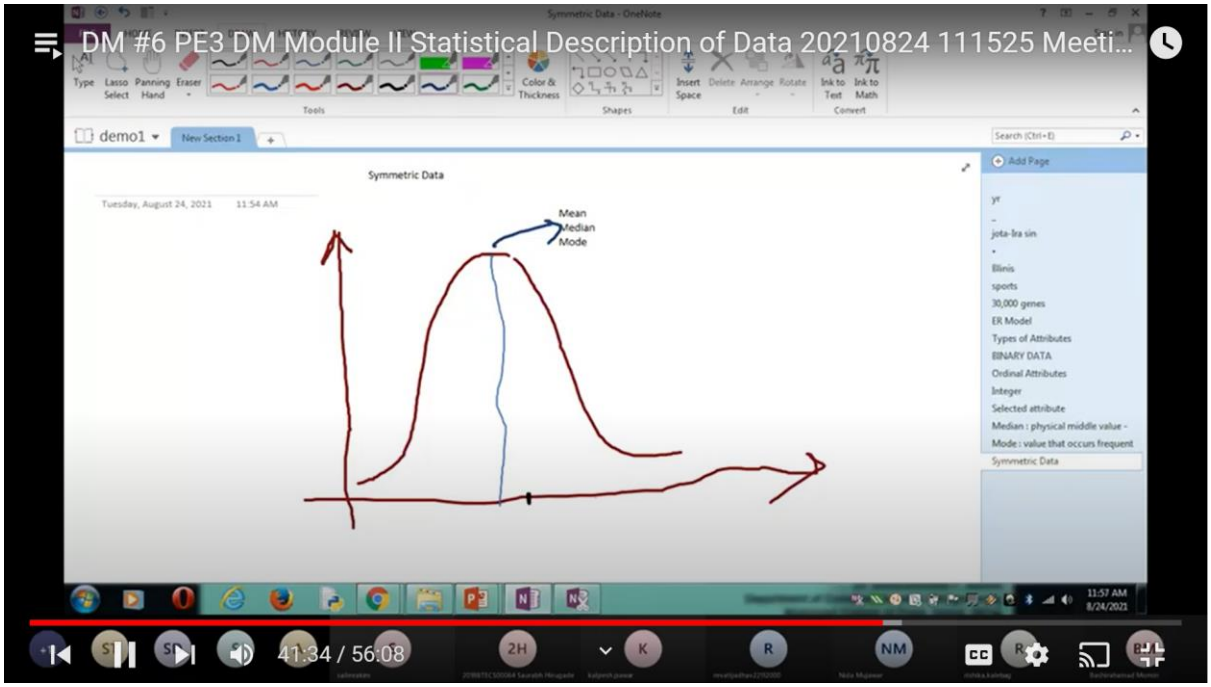

Statistical
Description of Data
Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.

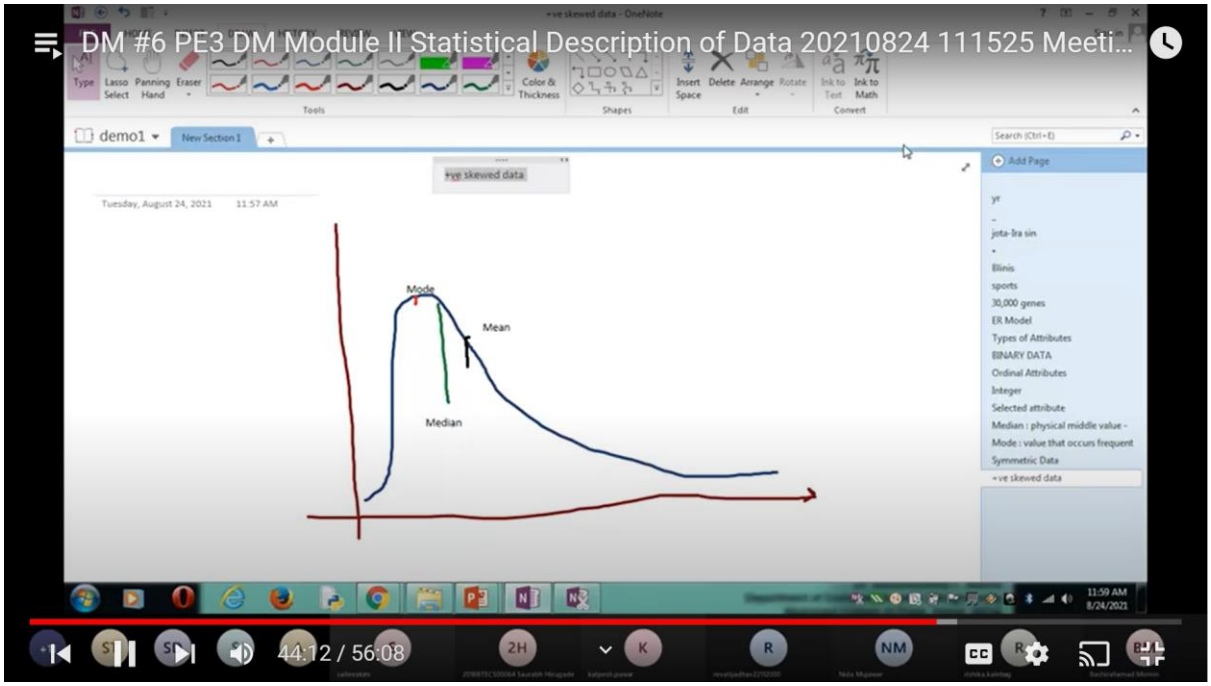

Selected attribute - OneNote
demo1
New Section 1
Weighted Average mean
$\bar{x} = 76.75$
Marks = { 89 , 90 , 88 , 40 }
Marks = { 89 , 90 , 88 , 92 }
89.75
Trimmed Mean
teams.microsoft.com is sharing your screen.
Stop sharing
Hide

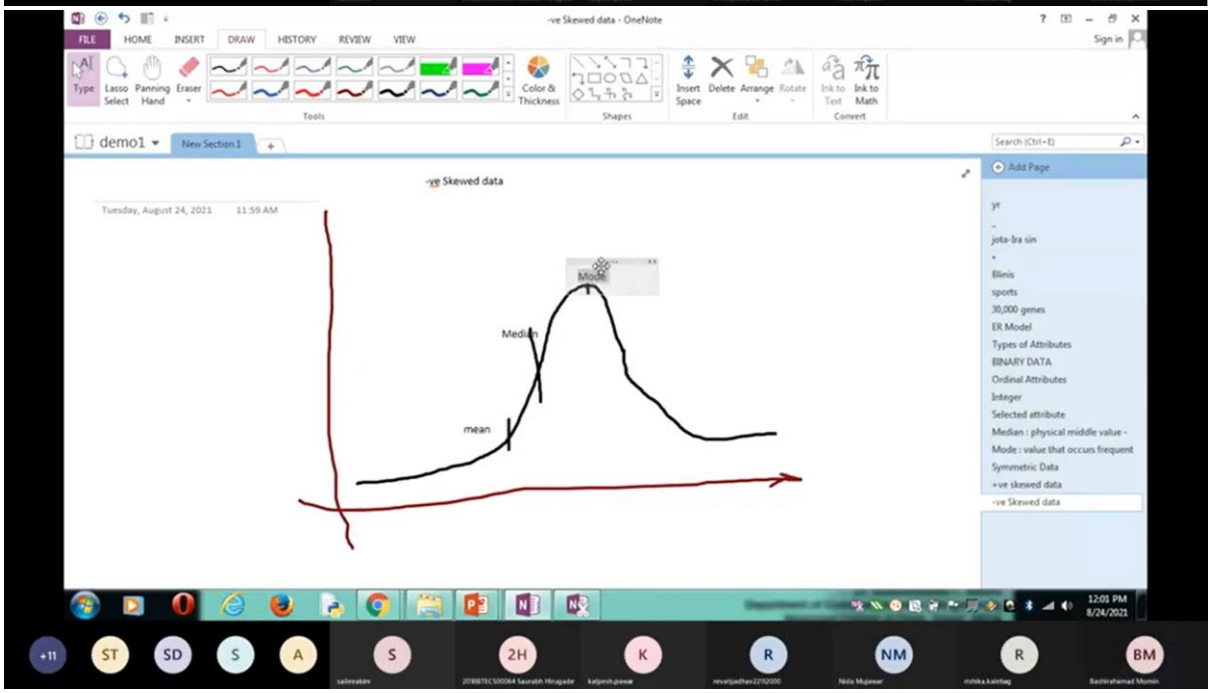

DM #6 PE3 DM Module II Statistical Description of Data 20210824 111525 Meeti...
Median : physical middle value - center of data set
It is the value that divide the data into two parts
Even
odd
two middle
Median = Average of two middle most values
52+56 / 2 = median
31:27 / 56:08


Mode : value that occurs frequently / repeated
It should occur more than once
e.g. 52 , 70
With one mode : unimodal
Two modes : bimodal
Three modes : trimodal
More than three : multimodal

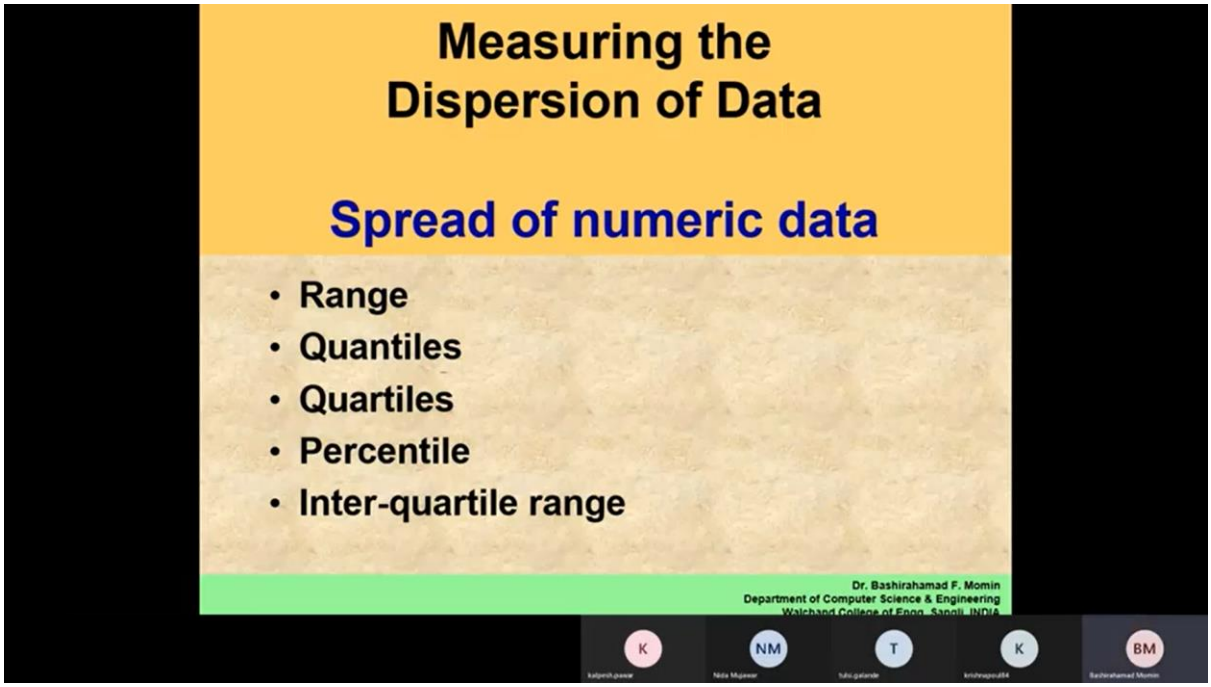
Mid range is the average of largest and smallest values




DM #6 PE3 DM Module II Statistical Description of Data 20210824 111525 Meeti...
+ve skewed data - OneNote
demo1
New Section 1
+ve skewed data
Tuesday, August 24, 2021 11:57 AM
Mode
Mean
Median
Add Page
44:12 / 56:08


-ve Skewed data - OneNote
demo1
New Section 1
-ve Skewed data
Tuesday, August 24, 2021 11:59 AM
Mode
Median
mean
Add Page

Lecture 7


Measuring the Dispersion of Data
Spread of numeric data
• Range
• Quantiles
• Quartiles
• Percentile
• Inter-quartile range
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
K
NM
T
K
BM

Dispersion : Spread of data - OneNote

FILE HOME INSERT DRAW HISTORY REVIEW VIEW

demo1 New Section 1

Dispersion : Spread of data

Wednesday, August 25, 2021 11:14 AM

Range : diff. between largest and smallest

Range : max(X)-min(X)

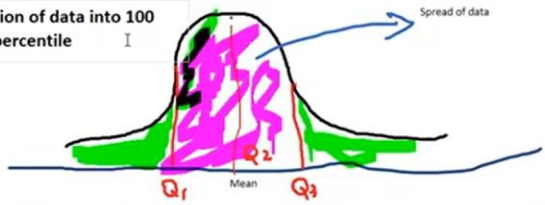
Quartiles : are the data points that split the given data in consecutive sets ... uniform sets

2-quantile : divide the given data sets into lower and upper halves

Median is the data point that divide into two part named as "quartiles"

4-quantile : have 3 data points that divide the data set into four quartiles

100-quantile : division of data into 100 consecutive sets : percentile



Add Page

yr
-
jota-lra sin
•
Blinis
sports
30,000 genes
ER Model
Types of Attributes
BINARY DATA
Ordinal Attributes
Integer
Selected attribute
Median : physical middle value -
Mode : value that occurs frequent
Symmetric Data
+ve skewed data
-ve Skewed data
Python code
Dispersion : Spread of data

11:35 AM 8/25/2021

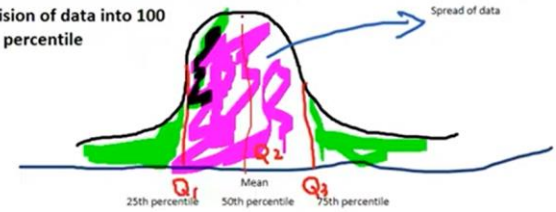
Dispersion : Spread of data - OneNote

FILE HOME INSERT DRAW HISTORY REVIEW VIEW

demo1 New Section 1

4-quantile : have 3 data points that divide the data set into four quartiles

100-quantile : division of data into 100 consecutive sets : percentile



Inter-quartile range IQR = Q3-Q1

Add Page

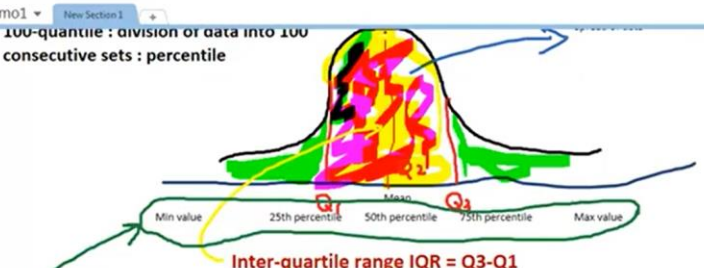
yr
-
jota-lra sin
•
Blinis
sports
30,000 genes
ER Model
Types of Attributes
BINARY DATA
Ordinal Attributes
Integer
Selected attribute
Median : physical middle value -
Mode : value that occurs frequent
Symmetric Data
+ve skewed data
-ve Skewed data
Python code
Dispersion : Spread of data

11:38 AM 8/25/2021

Dispersion : Spread of data - OneNote

demo1 — New Section 1

100-quantile : division of data into 100 consecutive sets : percentile



Inter-quartile range IQR = Q3-Q1

Drawback : don't provide the information about the end points - trails .. extremes

↓

Five number summary : consists of above quartiles + end points

Five number summary = { minimum value , Q1 , median (Q2) , Q3 , maximum value

Dispersion : Spread of data - OneNote

demo1 — New Section 1

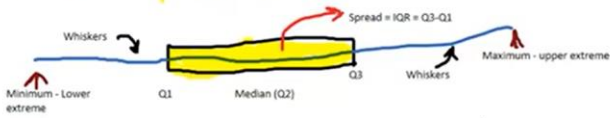
Drawback : don't provide the information about the end points - trails .. extremes

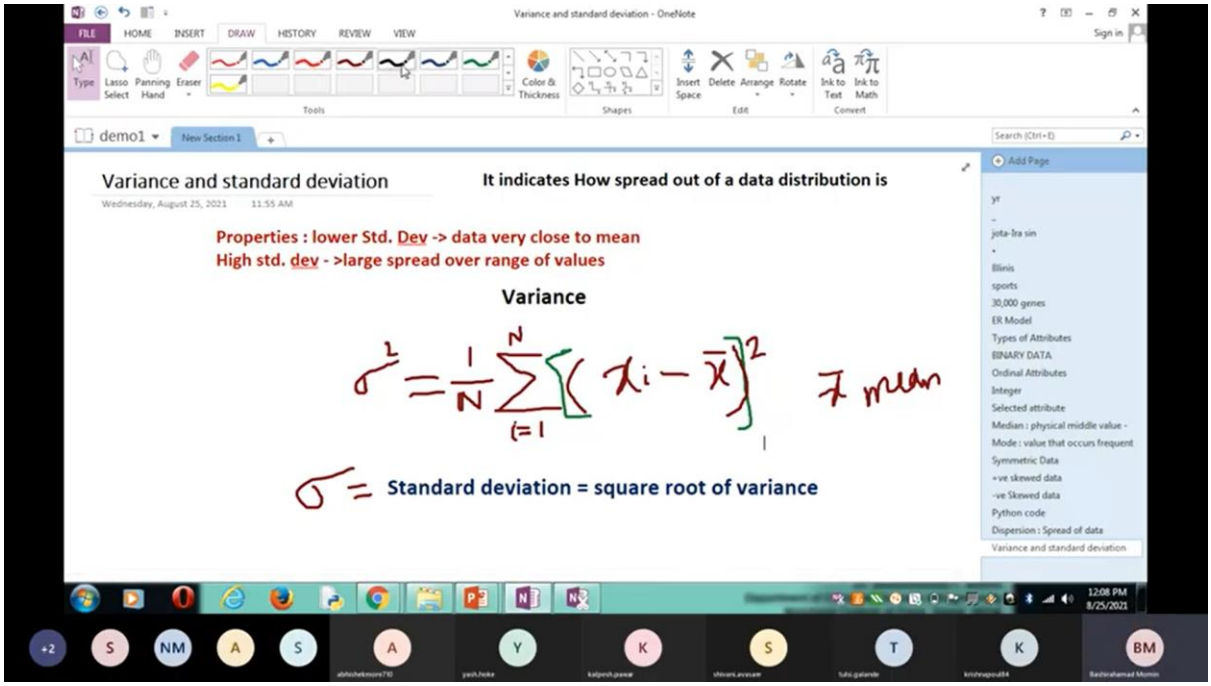
↓

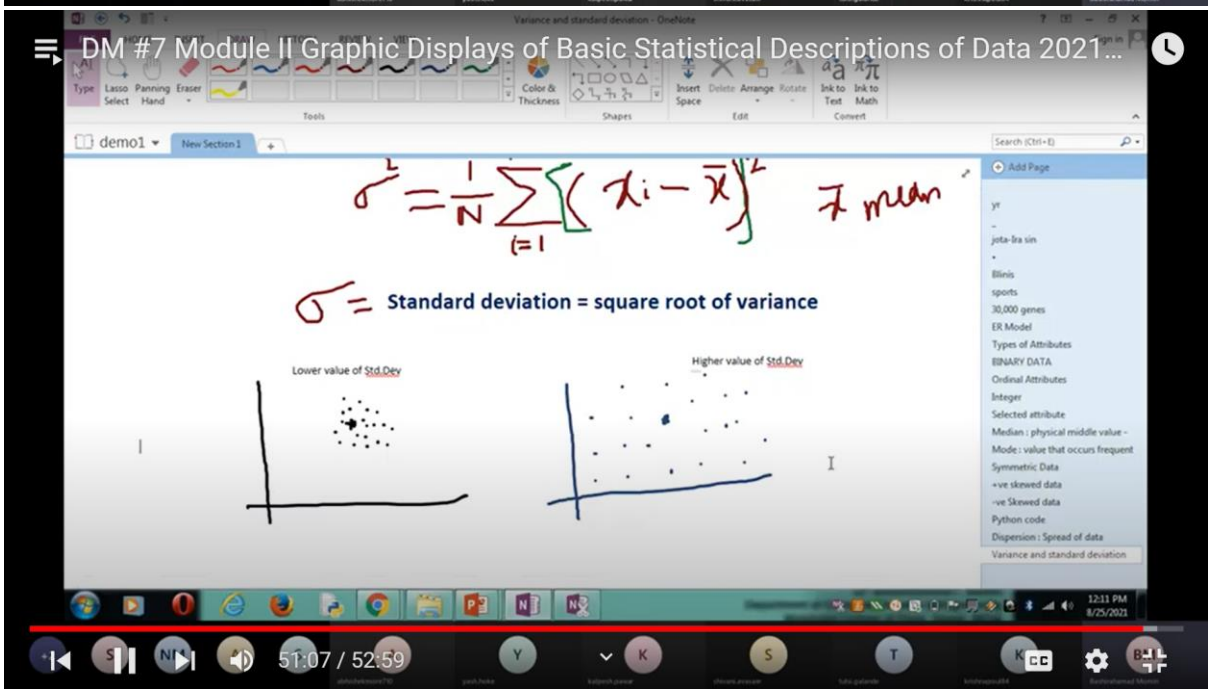
Five number summary : consists of above quartiles + end points

Five number summary = { minimum value , Q1 , median (Q2) , Q3 , maximum value }

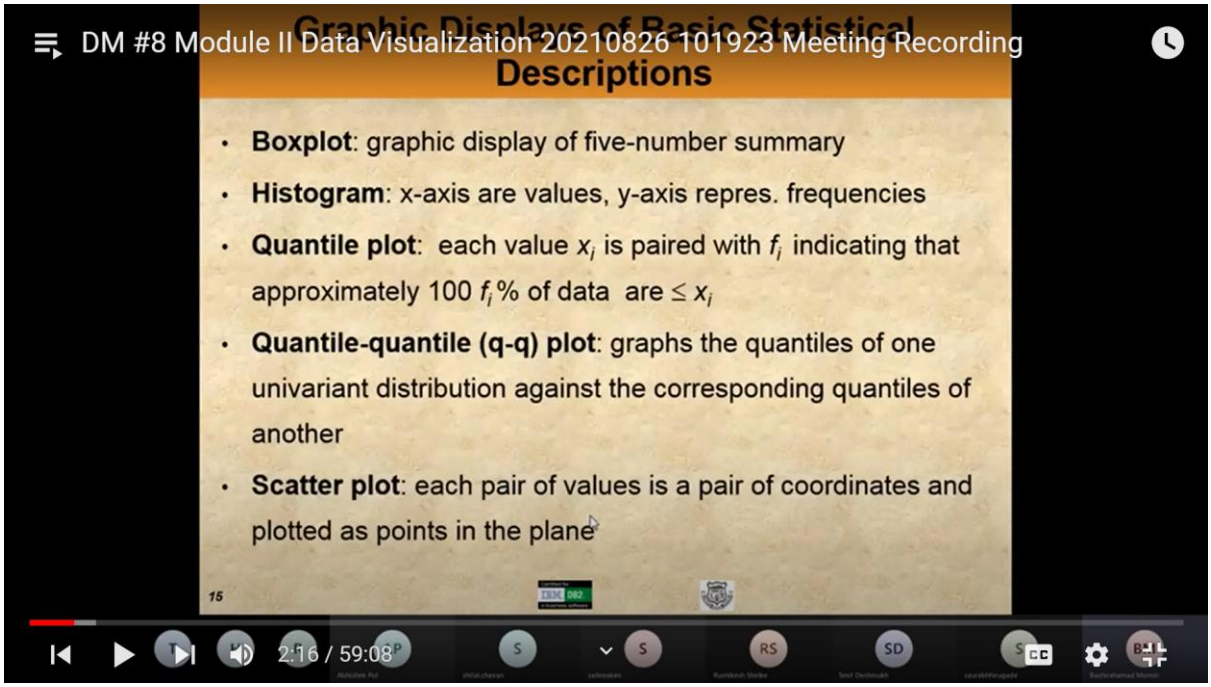
Graphically represented by BOX PLOT

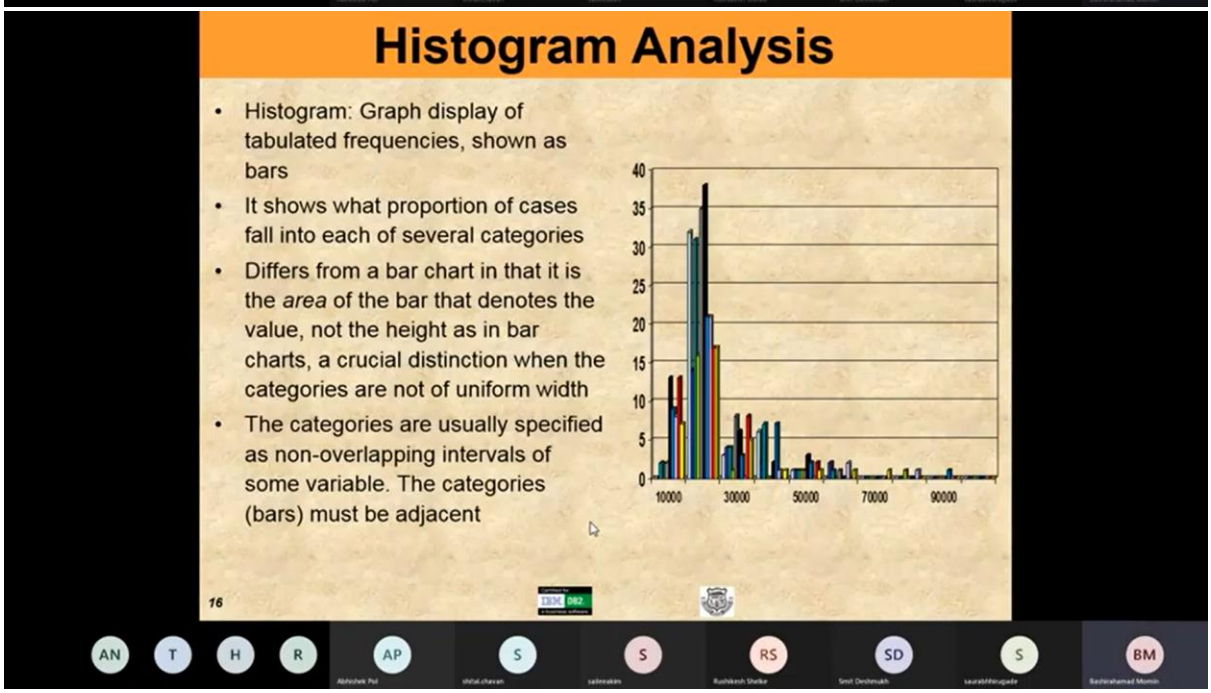



Variance and standard deviation - OneNote
demo1
New Section 1
Variance and standard deviation
It indicates How spread out of a data distribution is
Wednesday, August 25, 2021 11:55 AM
Properties : lower Std. Dev -> data very close to mean
High std. dev - >large spread over range of values
Variance
$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2$
$\bar{x}$ mean
σ = Standard deviation = square root of variance

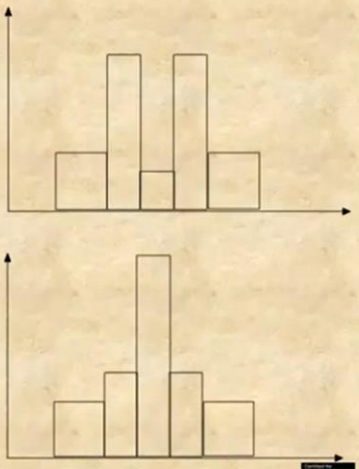

DM #7 Module II Graphic Displays of Basic Statistical Descriptions of Data 2021...
demo1
New Section 1
$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2$
$\bar{x}$ mean
σ = Standard deviation = square root of variance
Lower value of Std.Dev
Higher value of Std.Dev
51:07 / 52:59

Lecture 8:


DM #8 Module II Data Visualization 20210826 101923 Meeting Recording
Graphic Displays of Basic Statistical Descriptions
• Boxplot: graphic display of five-number summary
• Histogram: x-axis are values, y-axis repres. frequencies
• Quantile plot: each value x_i is paired with f_i indicating that approximately 100 f_i % of data are $\leq x_i$
• Quantile-quantile (q-q) plot: graphs the quantiles of one univariant distribution against the corresponding quantiles of another
• Scatter plot: each pair of values is a pair of coordinates and plotted as points in the plane
15
2:16 / 59:08


Histogram Analysis
• Histogram: Graph display of tabulated frequencies, shown as bars
• It shows what proportion of cases fall into each of several categories
• Differs from a bar chart in that it is the area of the bar that denotes the value, not the height as in bar charts, a crucial distinction when the categories are not of uniform width
• The categories are usually specified as non-overlapping intervals of some variable. The categories (bars) must be adjacent
40
35
30
25
20
15
10
5
0
10000
30000
50000
70000
90000
16

Histograms Often Tell More than Boxplots

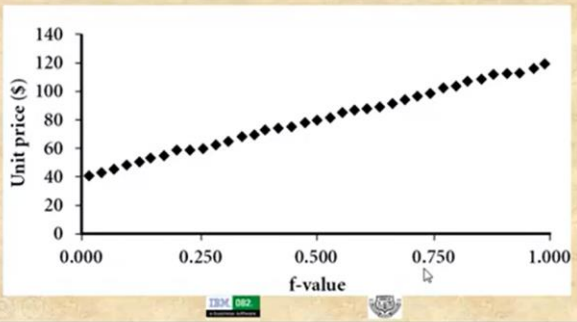


- The two histograms shown in the left may have the same boxplot representation
  - The same values for: min, Q1, median, Q3, max
- But they have rather different data distributions

17

Quantile Plot

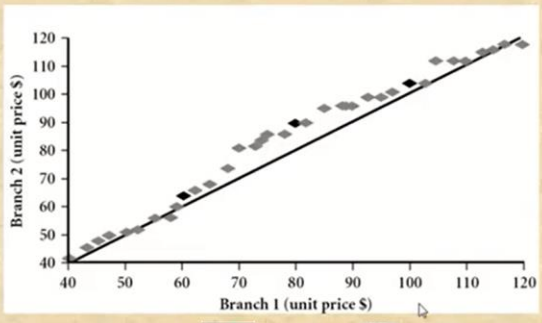
- Displays all of the data (allowing the user to assess both the overall behavior and unusual occurrences)
- Plots **quantile** information
  - For a data x_i data sorted in increasing order, f_i indicates that approximately 100 f_i% of the data are below or equal to the value x_i



18

Quantile-Quantile (Q-Q) Plot

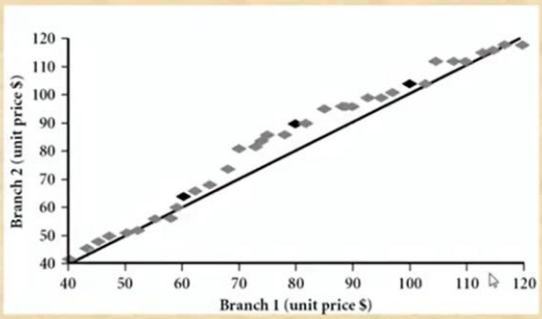
- Graphs the quantiles of one univariate distribution against the corresponding quantiles of another
- View: Is there is a shift in going from one distribution to another?



19

Quantile-Quantile (Q-Q) Plot

- Graphs the quantiles of one univariate distribution against the corresponding quantiles of another
- View: Is there is a shift in going from one distribution to another?



19

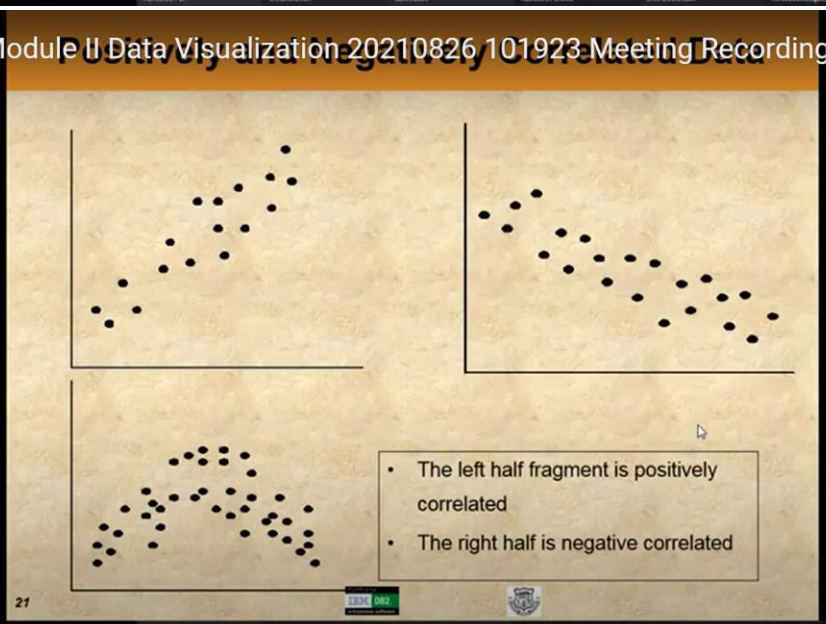
Scatter plot

- Provides a first look at bivariate data to see clusters of points, outliers, etc
- Each pair of values is treated as a pair of coordinates and plotted as points in the plane

20 20

DM #8 Module II Data Visualization 20210826 101923 Meeting Recording

Watch later



21:53 / 59:08

Uncorrelated Data

Data Visualization

Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.

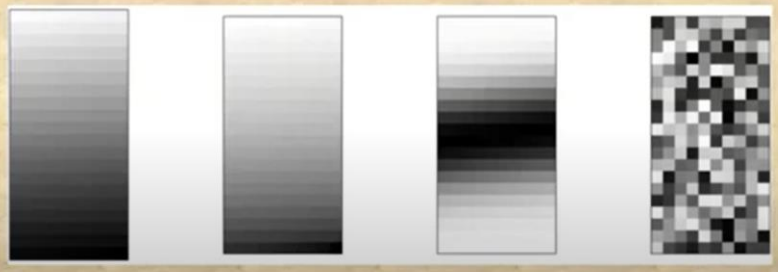
Data Visualization

- Why data visualization?
  - Gain insight into an information space by mapping data onto graphical primitives
  - Provide qualitative overview of large data sets
  - Search for patterns, trends, structure, irregularities, relationships among data
  - Help find interesting regions and suitable parameters for further quantitative analysis
  - Provide a visual proof of computer representations derived
- Categorization of visualization methods:
  - Pixel-oriented visualization techniques
  - Geometric projection visualization techniques
  - Icon-based visualization techniques
  - Hierarchical visualization techniques
  - Visualizing complex data and relations

24

Pixel-Oriented Visualization Techniques

- For a data set of m dimensions, create m windows on the screen, one for each dimension
- The m dimension values of a record are mapped to m pixels at the corresponding positions in the windows



(a) Income (b) Credit Limit (c) transaction volume (d) age

26

Geometric Projection Visualization Techniques

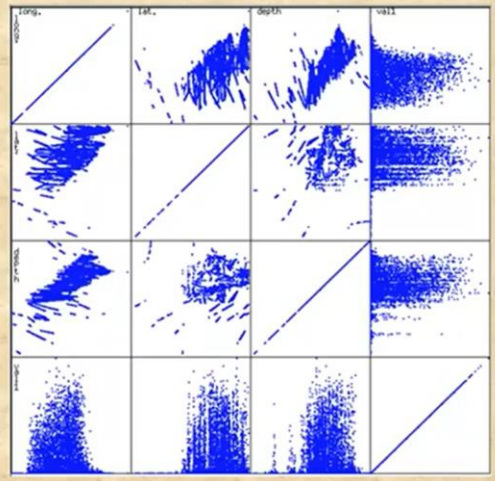
- Visualization of geometric transformations and projections of the data
- Methods
  - Direct visualization
  - Scatterplot and scatterplot matrices
  - Landscapes
  - Projection pursuit technique: Help users find meaningful projections of multidimensional data
  - Prosection views
  - Hyperslice
  - Parallel coordinates

Courtesy of NCSA, University of Illinois at Urbana-Champaign

data courtesy of NCSA, University of Illinois at Urbana-Champaign

27

Scatterplot Matrices



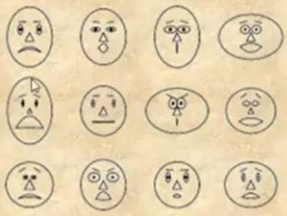
29

Icon-Based Visualization Techniques

- Visualization of the data values as features of icons
- Typical visualization methods
  - Chernoff Faces
  - Stick Figures
- General techniques
  - Shape coding: Use shape to represent certain information encoding
  - Color icons: Use color icons to encode more information
  - Tile bars: Use small icons to represent the relevant feature vectors in document retrieval

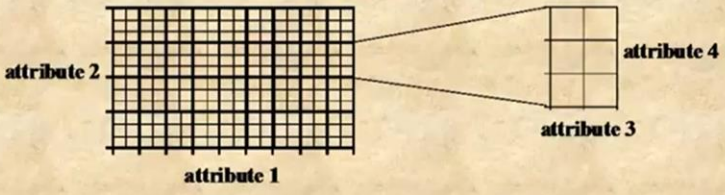
Chernoff Faces

- **A way to display variables on a two-dimensional surface, e.g., let x be eyebrow slant, y be eye size, z be nose length, etc.**
- **The figure shows faces produced using 10 characteristics--head eccentricity, eye size, eye spacing, eye eccentricity, pupil size, eyebrow slant, nose size, mouth shape, mouth size, and mouth opening): Each assigned one of 10 possible values, generated using *Mathematica* (S. Dickson)**



- REFERENCE: Gonick, L. and Smith, W. *The Cartoon Guide to Statistics.* New York: Harper Perennial, p. 212, 1993
- Weisstein, Eric W. "Chernoff Face." From *MathWorld*--A Wolfram Web Resource. mathworld.wolfram.com/ChernoffFace.html

Dimensional Stacking



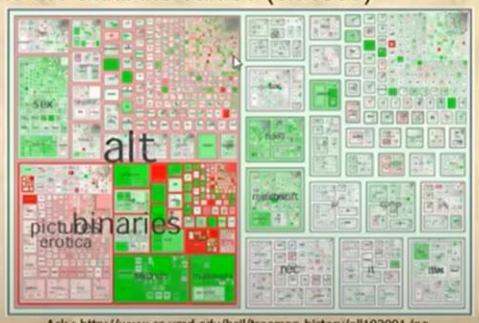
- Partitioning of the n-dimensional attribute space in 2-D subspaces, which are 'stacked' into each other
- Partitioning of the attribute value ranges into classes. The important attributes should be used on the outer levels.
- Adequate for data with ordinal attributes of low cardinality
- But, difficult to display more than nine dimensions

37

Tree-Map

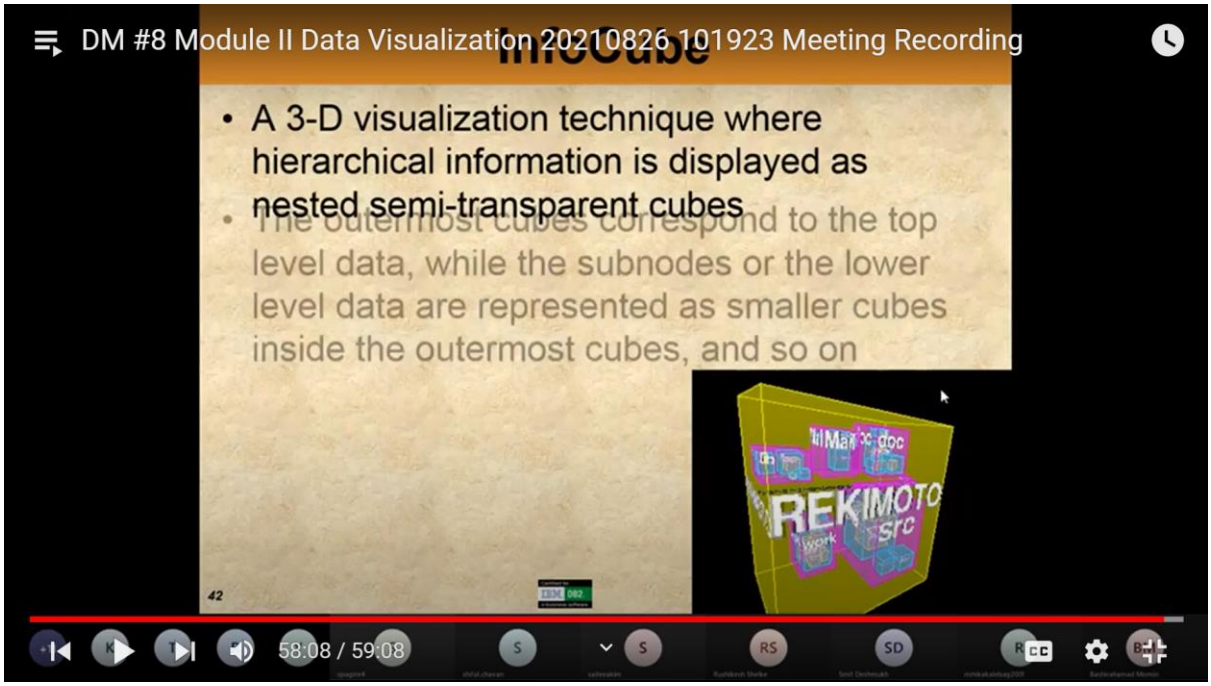
- Screen-filling method which uses a hierarchical partitioning of the screen into regions depending on the attribute values
- The x- and y-dimension of the screen are partitioned alternately according to the attribute values (classes)

MSR Netscan Image



Ack.: http://www.cs.umd.edu/hcil/treemap-history/all102001.jpg

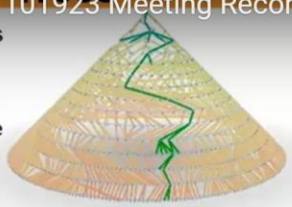
40


DM #8 Module II Data Visualization 20210826 101923 Meeting Recording
InfoCube
• A 3-D visualization technique where hierarchical information is displayed as nested semi-transparent cubes
• The outermost cubes correspond to the top level data, while the subnodes or the lower level data are represented as smaller cubes inside the outermost cubes, and so on
42
REKIMOTO
src
Work
58:08 / 59:08

DM #8 Module II Data Visualization 20210826 101923 Meeting Recording

Three-D Cone Trees

- *3D cone tree* visualization technique works well for up to a thousand nodes or so
- First build a *2D circle tree* that arranges its nodes in concentric circles centered on the root node
- Cannot avoid overlaps when projected to 2D
- G. Robertson, J. Mackinlay, S. Card. "Cone Trees: Animated 3D Visualizations of Hierarchical Information", *ACM SIGCHI'91*
- [illegible] website: Visualize a social network data set that models the way an infection spreads from one person to the next



43

Ack.: http://inadeausoftware.com/articles/visualization

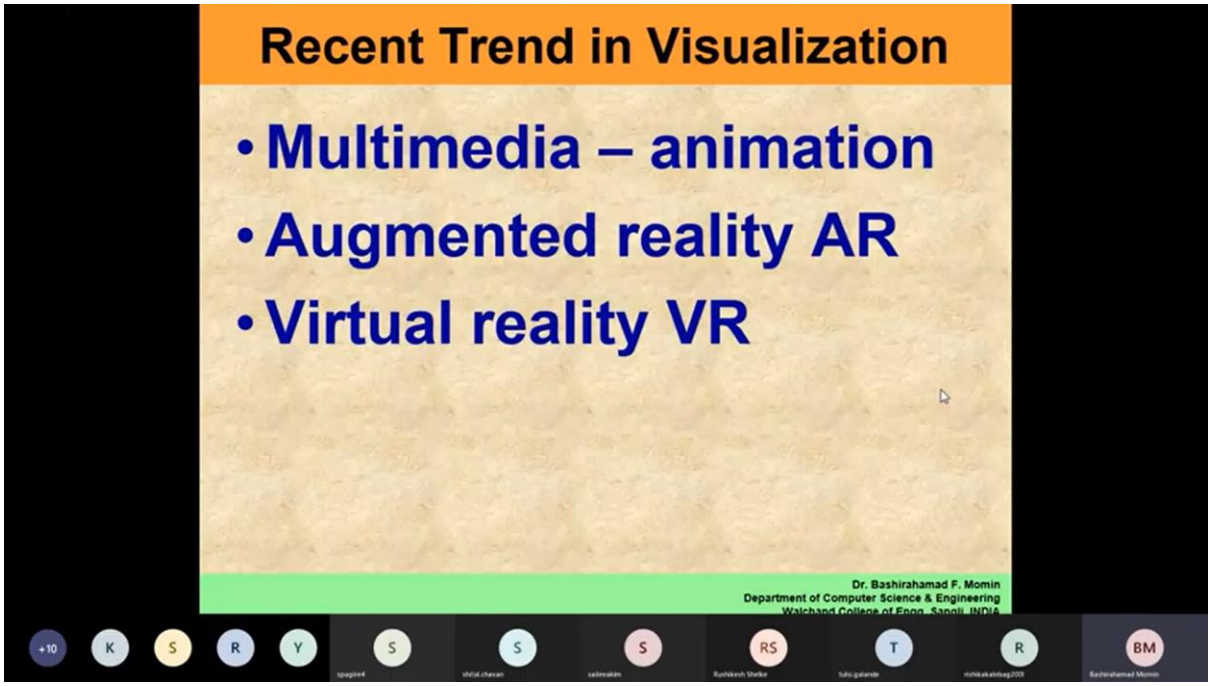
58:13 / 59:08

Visualizing Complex Data and Relations

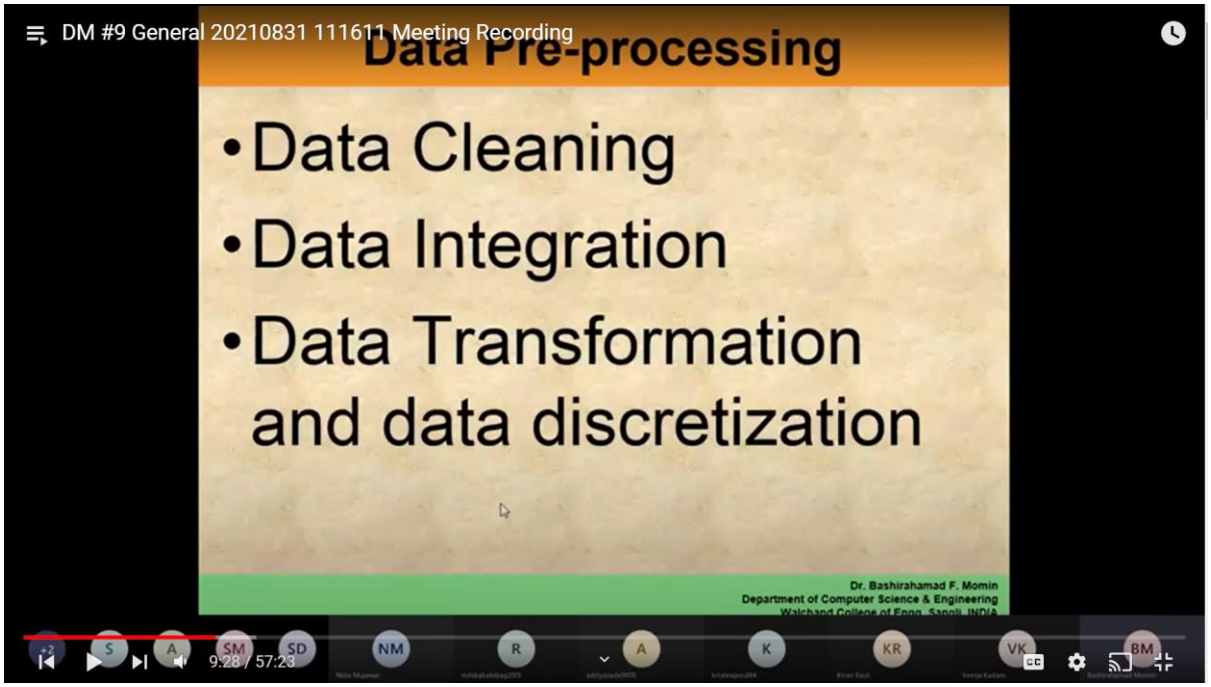
- Visualizing non-numerical data: text and social networks
  - The importance of tag is represented by font size/color
- Besides text data, there are also methods to visualize relationships, such as visualizing social networks

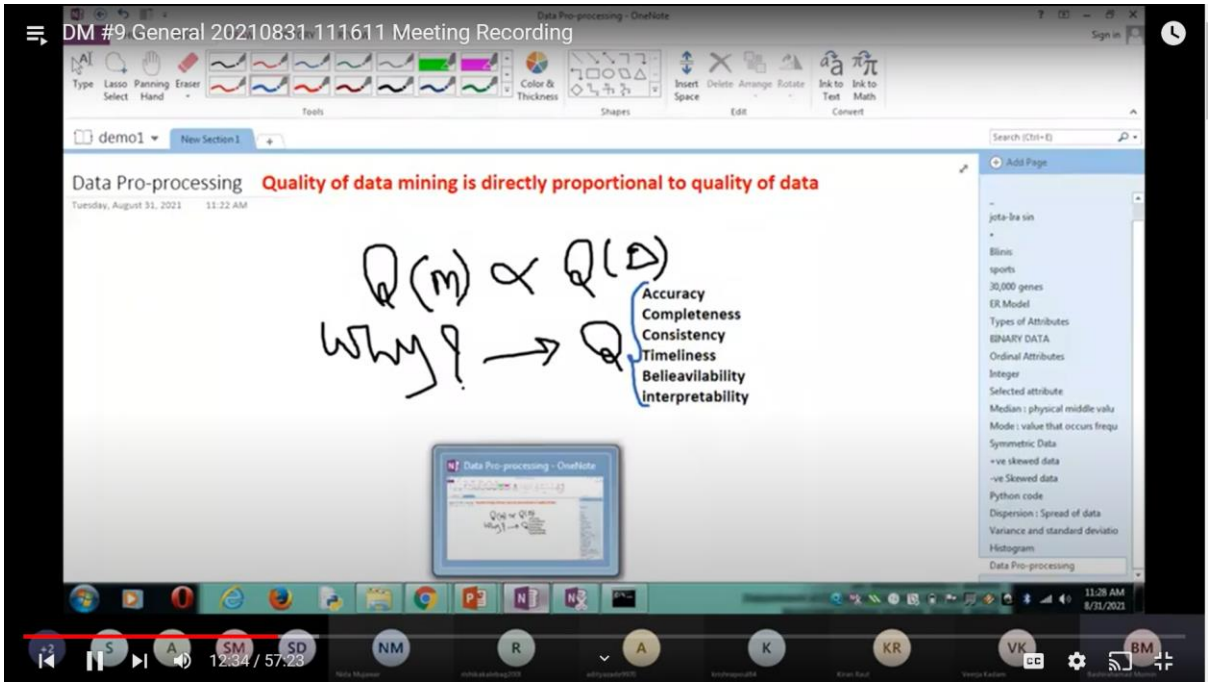
Newsmap: Google News Stories in 2005

44

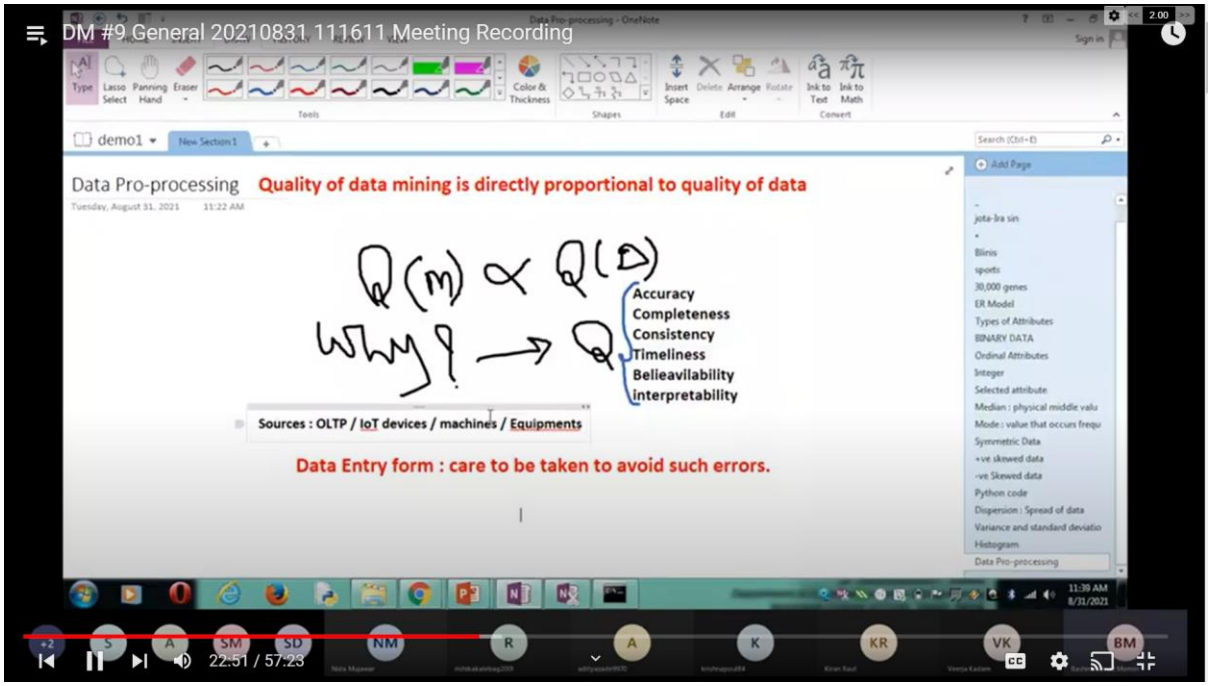

Recent Trend in Visualization
• Multimedia – animation
• Augmented reality AR
• Virtual reality VR
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
+10
K
S
R
Y
S
S
S
RS
T
R
BM

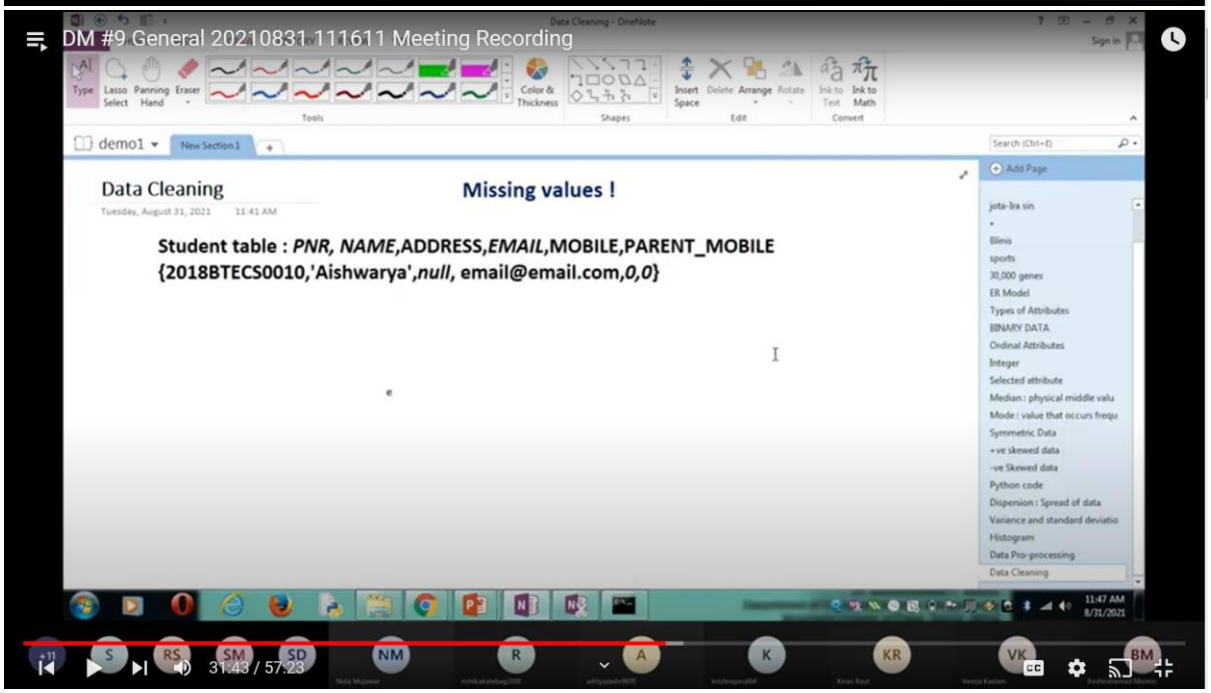
Lecture 9 started

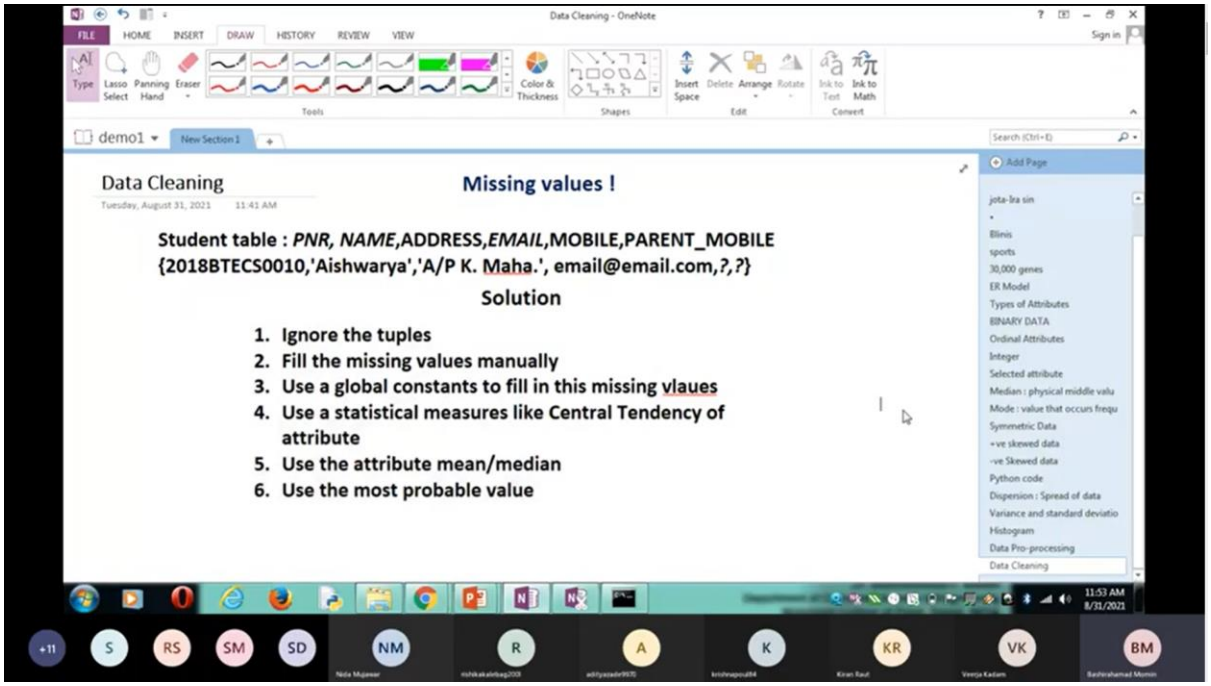




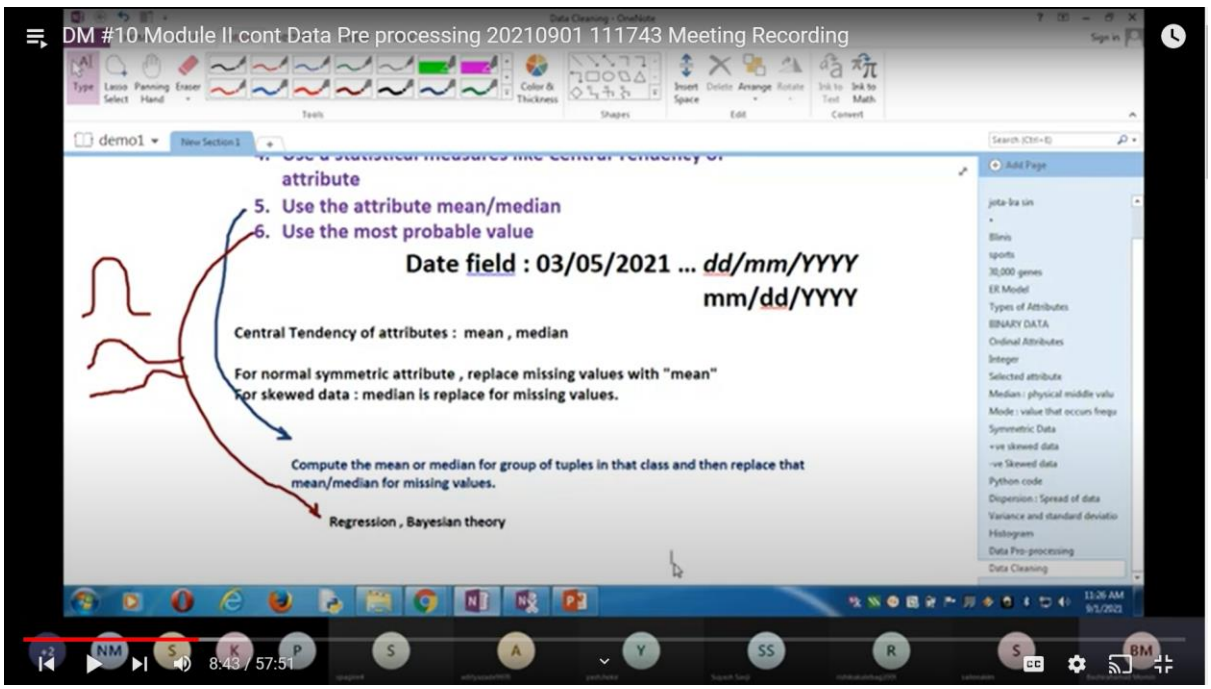
What is the source of data?
OLTP / IoT devices / machines / equipments

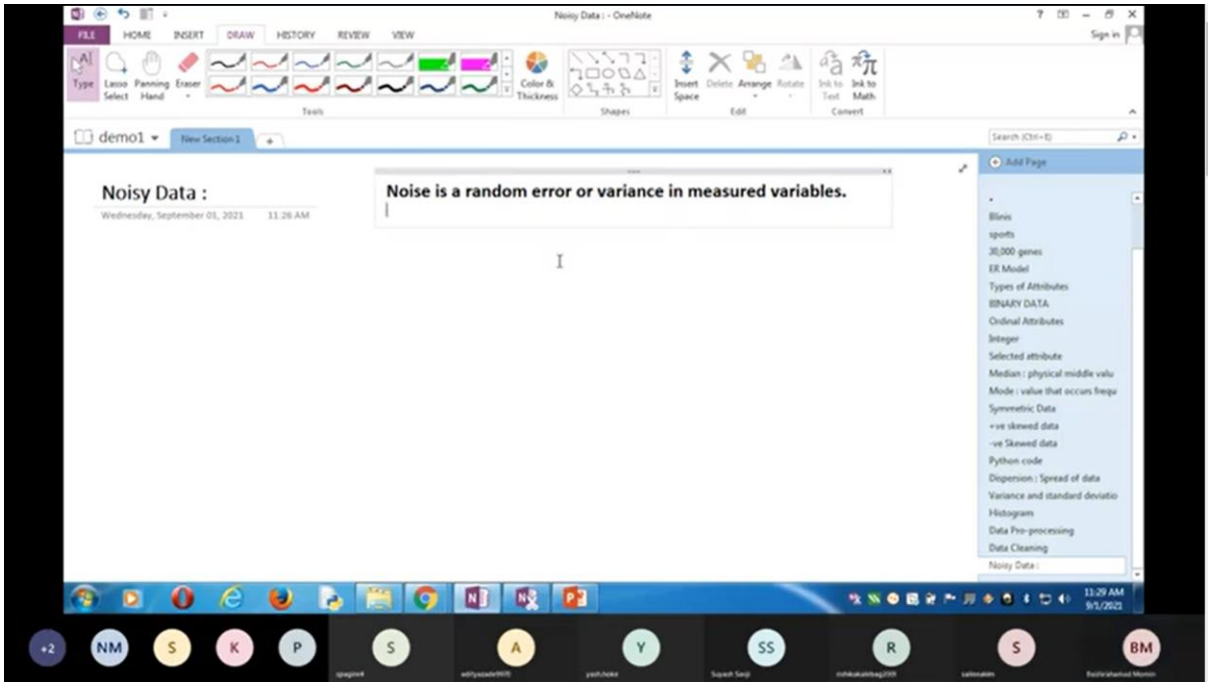

DM #9 General 20210831 111611 Meeting Recording
Data Pro-processing - OneNote
Data Pro-processing
Quality of data mining is directly proportional to quality of data
Tuesday, August 31, 2021 11:22 AM
Q(m) ∝ Q(D)
Why ? → Q
Accuracy
Completeness
Consistency
Timeliness
Belieavilability
interpretability
Sources : OLTP / IoT devices / machines / Equipments
Data Entry form : care to be taken to avoid such errors.
22:51 / 57:23

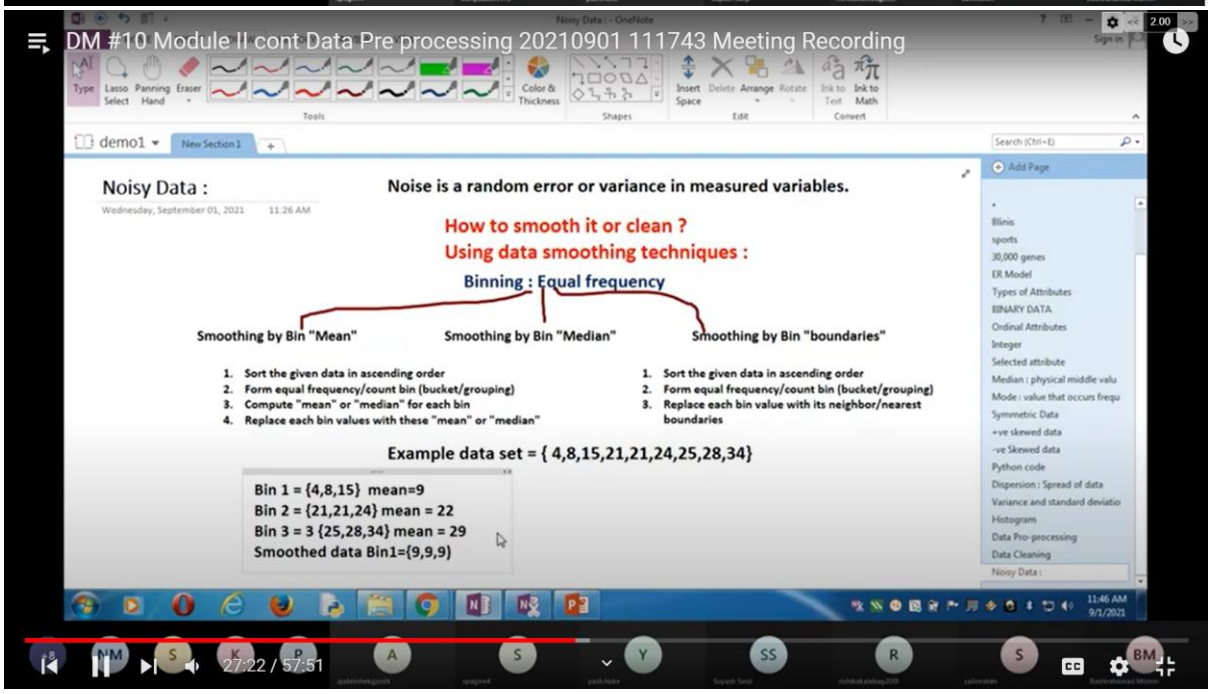

DM #9 General 20210831 111611 Meeting Recording
Data Cleaning - OneNote
Data Cleaning
Missing values !
Tuesday, August 31, 2021 11:41 AM
Student table : PNR, NAME,ADDRESS,EMAIL,MOBILE,PARENT_MOBILE
{2018BTECS0010,'Aishwarya',null, email@email.com,0,0}
31:43 / 57:23

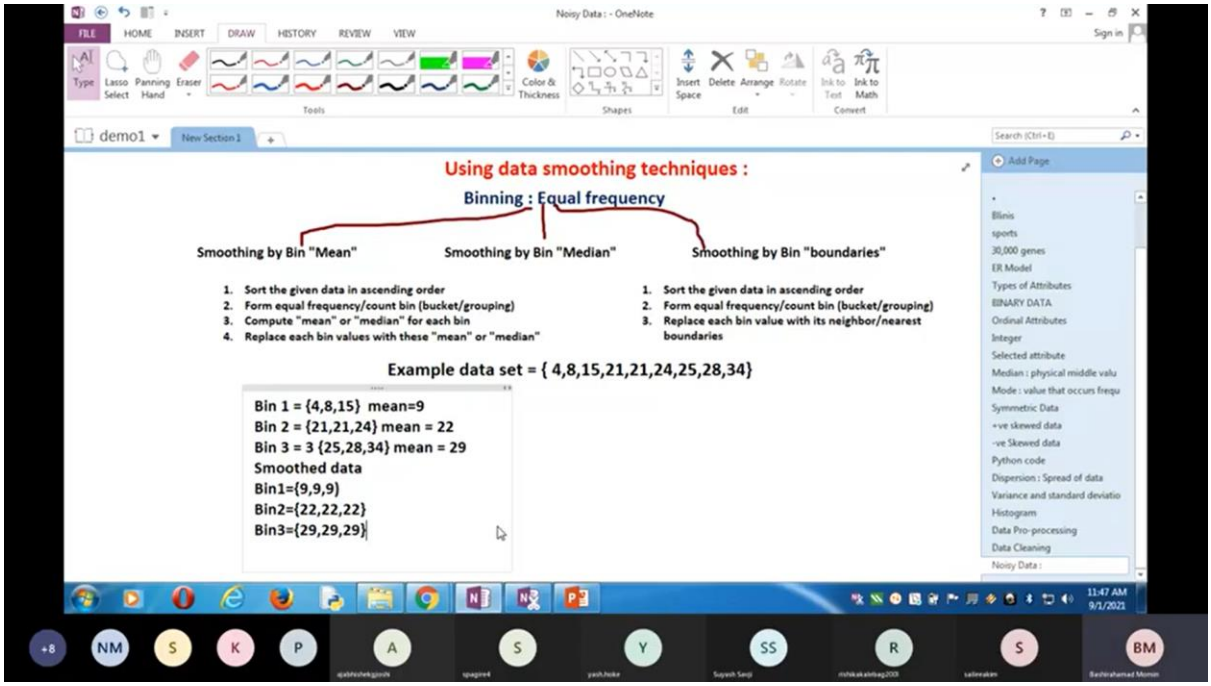

Data Cleaning - OneNote
demo1
Data Cleaning
Missing values !
Student table : PNR, NAME,ADDRESS,EMAIL,MOBILE,PARENT_MOBILE
{2018BTECS0010,'Aishwarya','A/P K. Maha.', email@email.com,?,?}
Solution
1. Ignore the tuples
2. Fill the missing values manually
3. Use a global constants to fill in this missing vlaues
4. Use a statistical measures like Central Tendency of attribute
5. Use the attribute mean/median
6. Use the most probable value

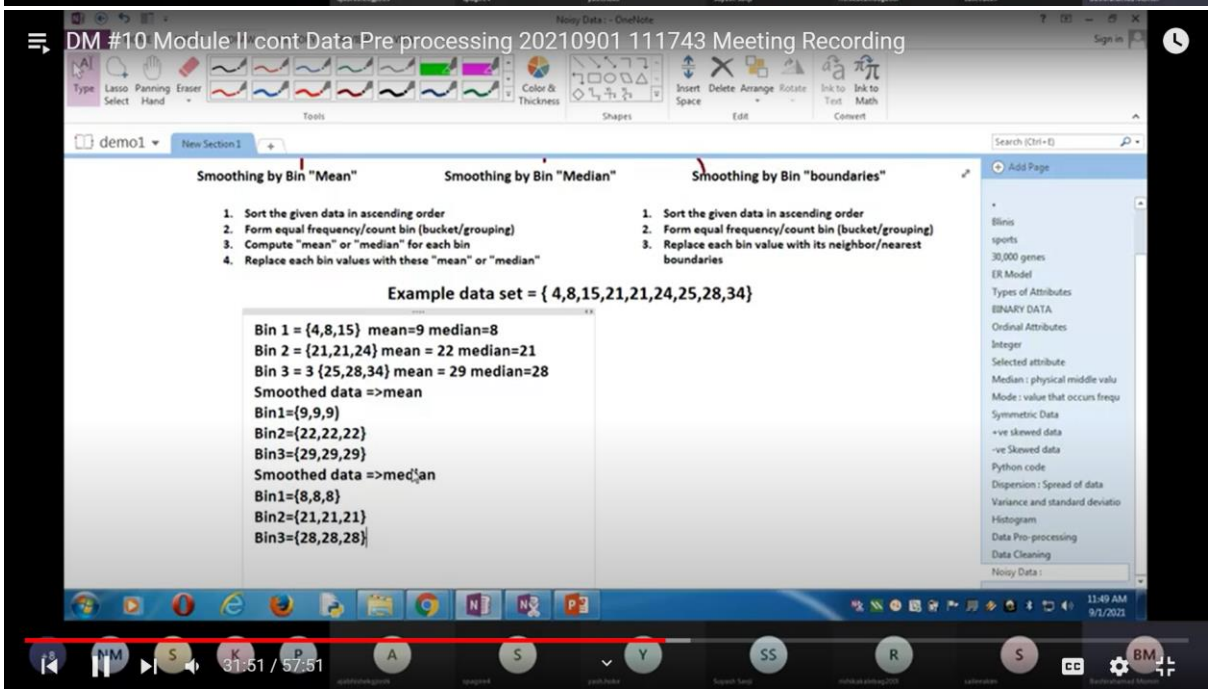
Lecture 10

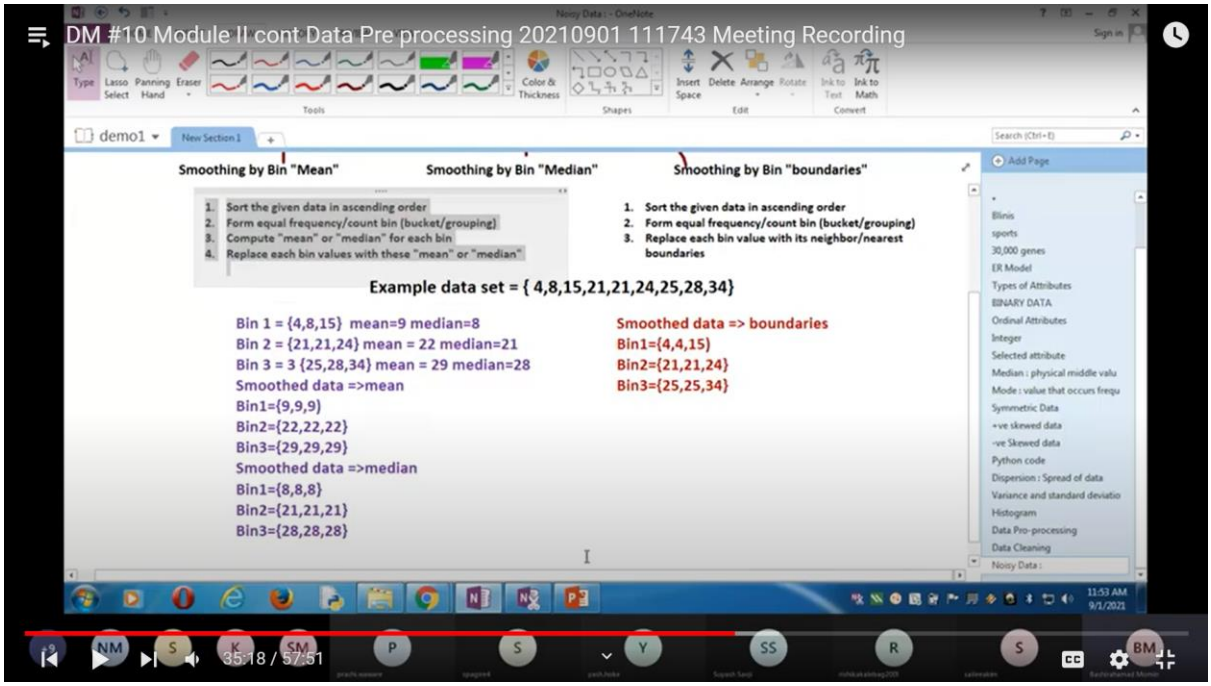

DM #10 Module II cont Data Pre processing 20210901 111743 Meeting Recording
attribute
5. Use the attribute mean/median
6. Use the most probable value
Date field : 03/05/2021 ... dd/mm/YYYY
mm/dd/YYYY
Central Tendency of attributes : mean , median
For normal symmetric attribute , replace missing values with "mean"
For skewed data : median is replace for missing values.
Compute the mean or median for group of tuples in that class and then replace that mean/median for missing values.
Regression , Bayesian theory
8:43 / 57:51


Noisy Data : - OneNote
demo1
New Section 1
Noisy Data :
Wednesday, September 01, 2021 11:26 AM
Noise is a random error or variance in measured variables.


DM #10 Module II cont Data Pre processing 20210901 111743 Meeting Recording
Noisy Data :
Wednesday, September 01, 2021 11:26 AM
Noise is a random error or variance in measured variables.
How to smooth it or clean ?
Using data smoothing techniques :
Binning : Equal frequency
Smoothing by Bin "Mean"
Smoothing by Bin "Median"
Smoothing by Bin "boundaries"
1. Sort the given data in ascending order
2. Form equal frequency/count bin (bucket/grouping)
3. Compute "mean" or "median" for each bin
4. Replace each bin values with these "mean" or "median"
1. Sort the given data in ascending order
2. Form equal frequency/count bin (bucket/grouping)
3. Replace each bin value with its neighbor/nearest boundaries
Example data set = { 4,8,15,21,21,24,25,28,34}
Bin 1 = {4,8,15} mean=9
Bin 2 = {21,21,24} mean = 22
Bin 3 = 3 {25,28,34} mean = 29
Smoothed data Bin1={9,9,9}
27:22 / 57:51

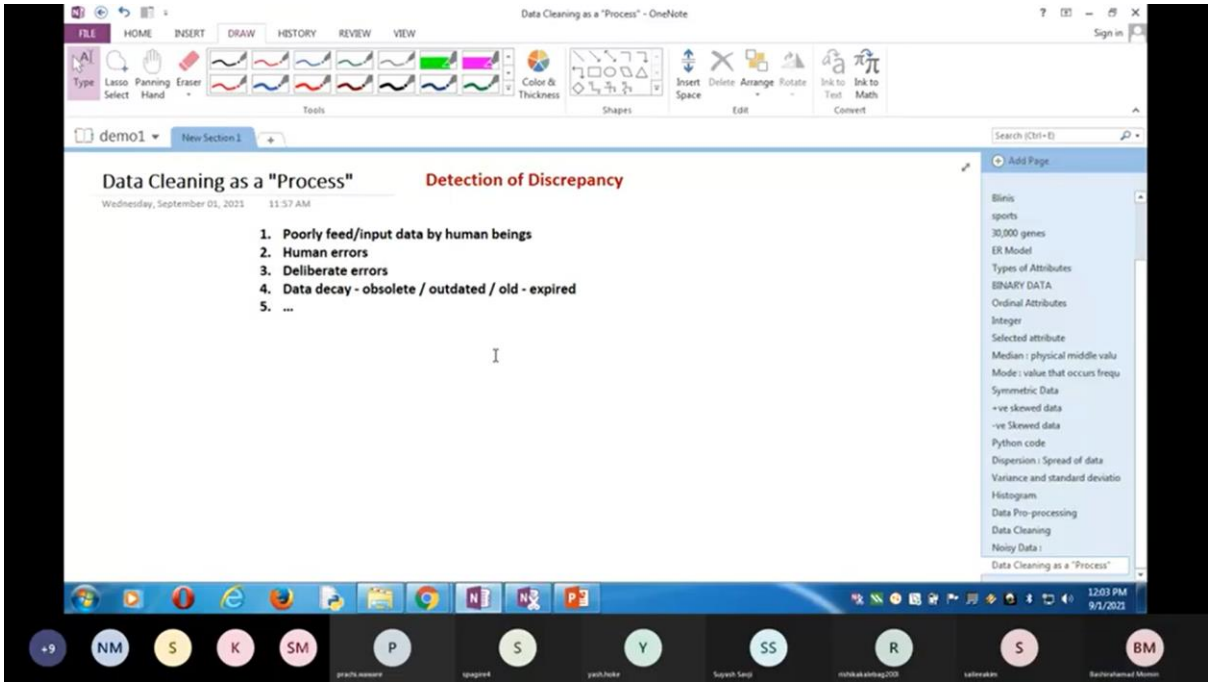

Noisy Data : - OneNote
demo1
New Section 1
Using data smoothing techniques :
Binning : Equal frequency
Smoothing by Bin "Mean"
Smoothing by Bin "Median"
Smoothing by Bin "boundaries"
1. Sort the given data in ascending order
2. Form equal frequency/count bin (bucket/grouping)
3. Compute "mean" or "median" for each bin
4. Replace each bin values with these "mean" or "median"
1. Sort the given data in ascending order
2. Form equal frequency/count bin (bucket/grouping)
3. Replace each bin value with its neighbor/nearest boundaries
Example data set = { 4,8,15,21,21,24,25,28,34}
Bin 1 = {4,8,15} mean=9
Bin 2 = {21,21,24} mean = 22
Bin 3 = 3 {25,28,34} mean = 29
Smoothed data
Bin1={9,9,9)
Bin2={22,22,22}
Bin3={29,29,29}
11:47 AM
9/1/2021

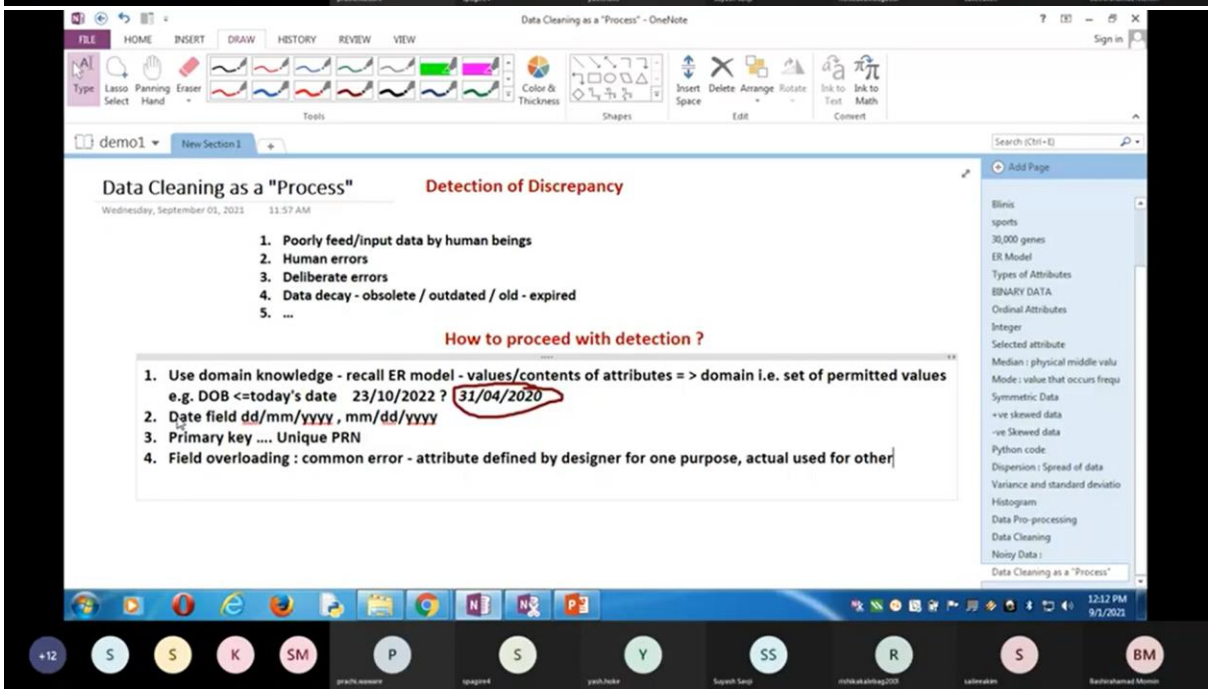

DM #10 Module II cont Data Pre processing 20210901 111743 Meeting Recording
Noisy Data : - OneNote
demo1
New Section 1
Smoothing by Bin "Mean"
Smoothing by Bin "Median"
Smoothing by Bin "boundaries"
1. Sort the given data in ascending order
2. Form equal frequency/count bin (bucket/grouping)
3. Compute "mean" or "median" for each bin
4. Replace each bin values with these "mean" or "median"
1. Sort the given data in ascending order
2. Form equal frequency/count bin (bucket/grouping)
3. Replace each bin value with its neighbor/nearest boundaries
Example data set = { 4,8,15,21,21,24,25,28,34}
Bin 1 = {4,8,15} mean=9 median=8
Bin 2 = {21,21,24} mean = 22 median=21
Bin 3 = 3 {25,28,34} mean = 29 median=28
Smoothed data =>mean
Bin1={9,9,9)
Bin2={22,22,22}
Bin3={29,29,29}
Smoothed data =>median
Bin1={8,8,8}
Bin2={21,21,21}
Bin3={28,28,28}
11:49 AM
9/1/2021
31:51 / 57:51



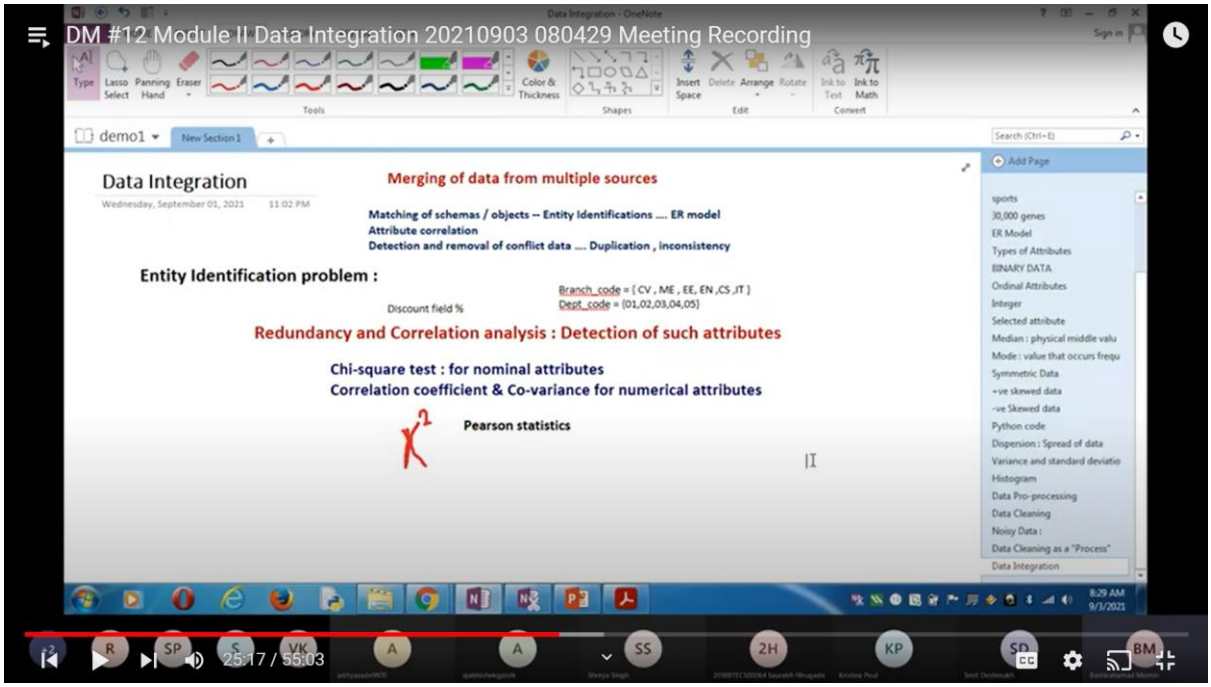
How to smooth the data?
Binning
Regression: Linear/ Multi linear

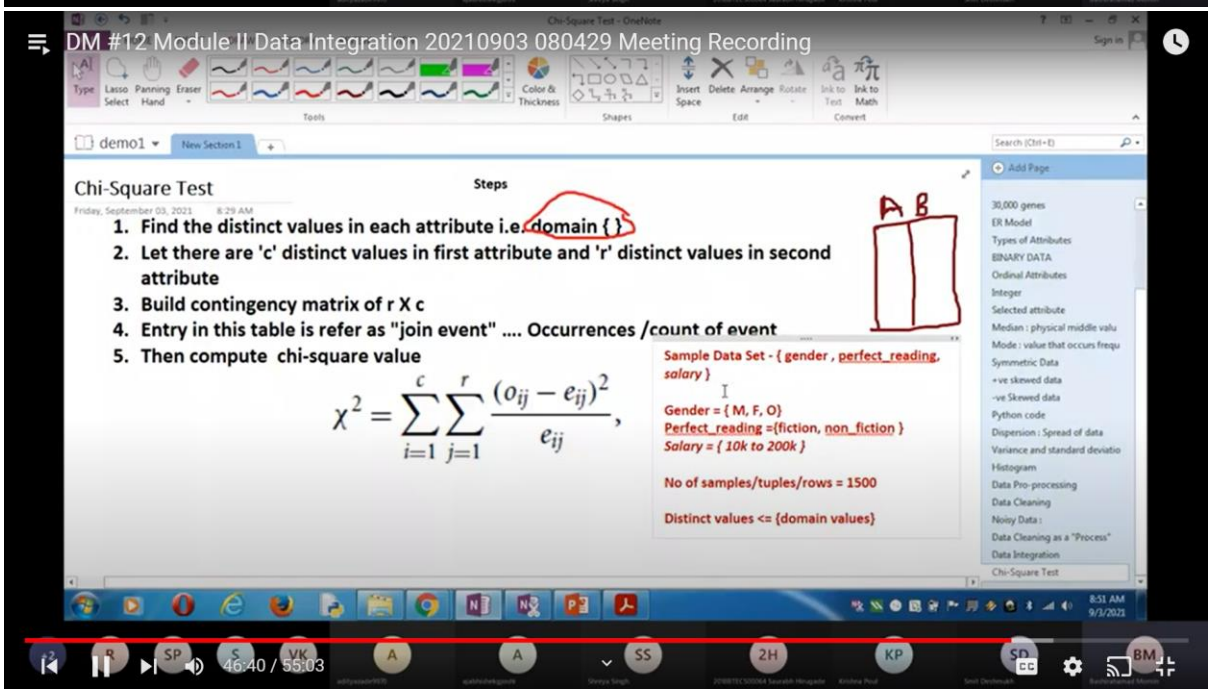
Data Cleaning as a “process”
Detection of Discrepancy
Whenever there is an human intervention, there is a possibility of an error


Data Cleaning as a "Process" - OneNote
demo1
New Section 1
Data Cleaning as a "Process"
Detection of Discrepancy
Wednesday, September 01, 2021 11:57 AM
1. Poorly feed/input data by human beings
2. Human errors
3. Deliberate errors
4. Data decay - obsolete / outdated / old - expired
5. ...

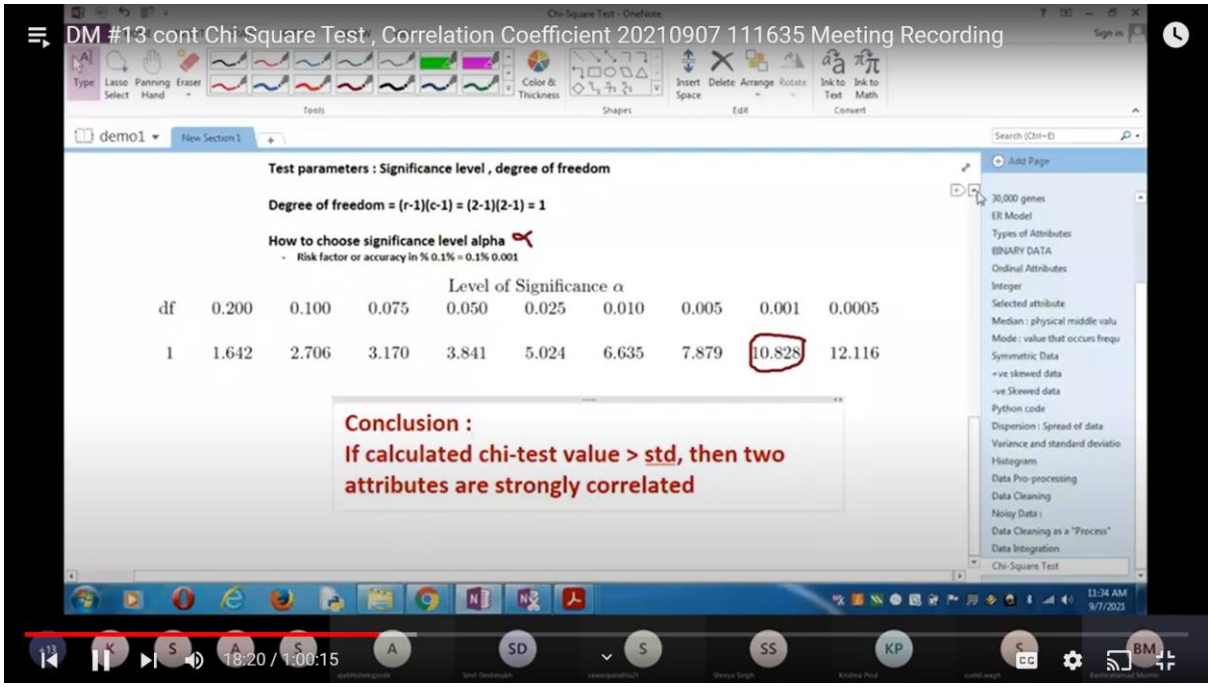

Data Cleaning as a "Process" - OneNote
demo1
New Section 1
Data Cleaning as a "Process"
Detection of Discrepancy
Wednesday, September 01, 2021 11:57 AM
1. Poorly feed/input data by human beings
2. Human errors
3. Deliberate errors
4. Data decay - obsolete / outdated / old - expired
5. ...
How to proceed with detection ?
1. Use domain knowledge - recall ER model - values/contents of attributes = > domain i.e. set of permitted values
e.g. DOB <=today's date 23/10/2022 ? 31/04/2020
2. Date field dd/mm/yyyy , mm/dd/yyyy
3. Primary key .... Unique PRN
4. Field overloading : common error - attribute defined by designer for one purpose, actual used for other

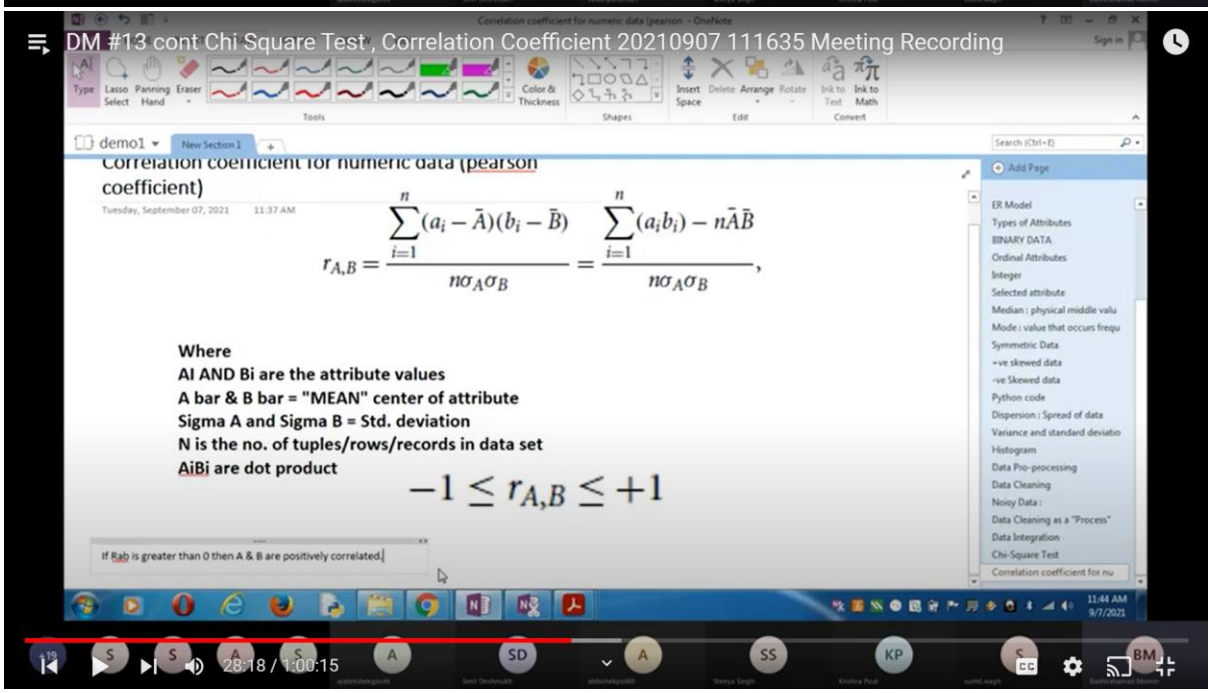
Lecture 12

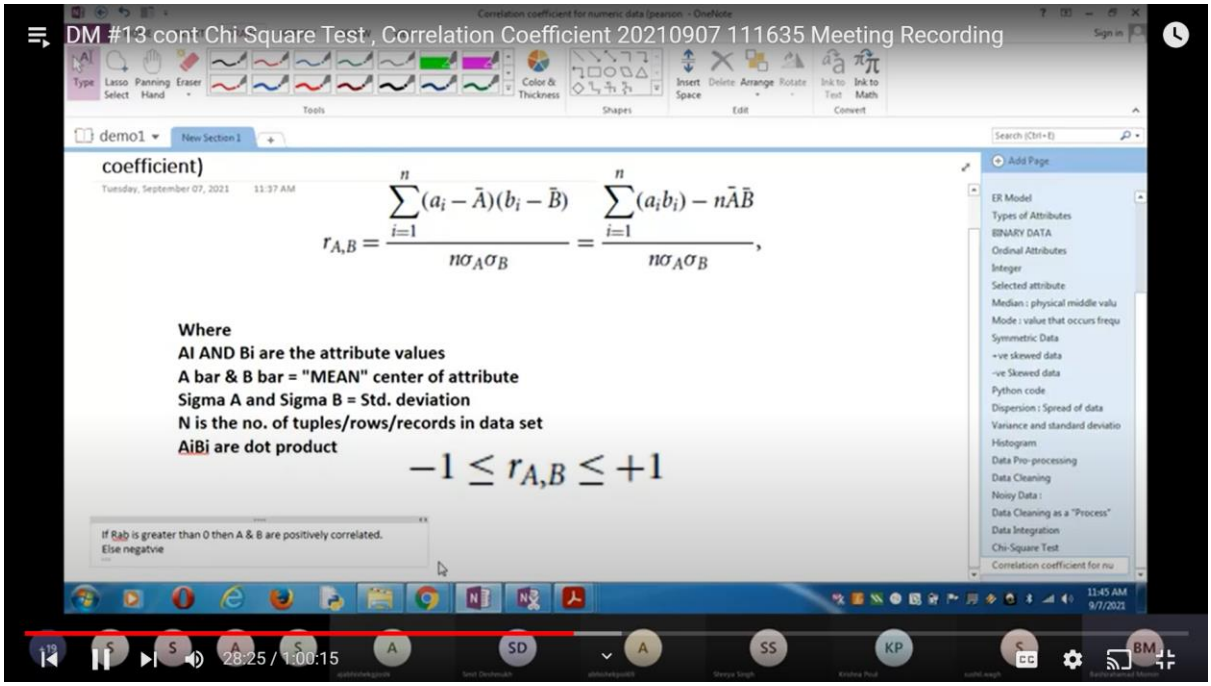

DM #12 Module II Data Integration 20210903 080429 Meeting Recording
Data Integration
Merging of data from multiple sources
Matching of schemas / objects -- Entity Identifications .... ER model
Attribute correlation
Detection and removal of conflict data .... Duplication , inconsistency
Entity Identification problem :
Branch_code = { CV , ME , EE, EN ,CS ,IT }
Dept_code = {01,02,03,04,05}
Discount field %
Redundancy and Correlation analysis : Detection of such attributes
Chi-square test : for nominal attributes
Correlation coefficient & Co-variance for numerical attributes
Pearson statistics
25:17 / 55:03

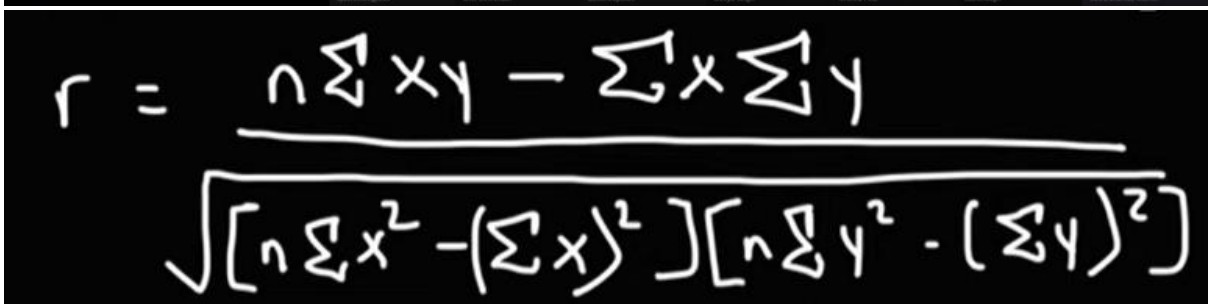

DM #12 Module II Data Integration 20210903 080429 Meeting Recording
Chi-Square Test
Steps
1. Find the distinct values in each attribute i.e. domain { }
2. Let there are 'c' distinct values in first attribute and 'r' distinct values in second attribute
3. Build contingency matrix of r X c
4. Entry in this table is refer as "join event" .... Occurrences /count of event
5. Then compute chi-square value
$$\chi^2 = \sum_{i=1}^{c}\sum_{j=1}^{r}\frac{(o_{ij}-e_{ij})^2}{e_{ij}},$$
A B
Sample Data Set - { gender , perfect_reading, salary }
Gender = { M, F, O}
Perfect_reading ={fiction, non_fiction }
Salary = { 10k to 200k }
No of samples/tuples/rows = 1500
Distinct values <= {domain values}
46:40 / 55:03

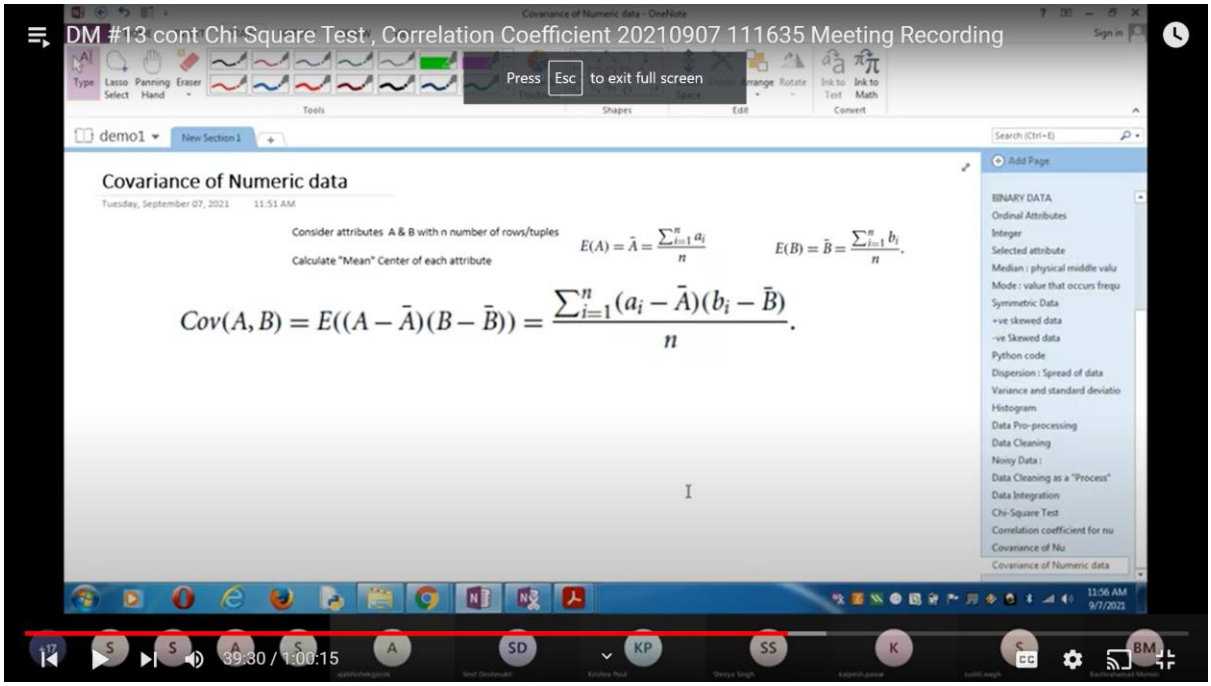
Lecture 13

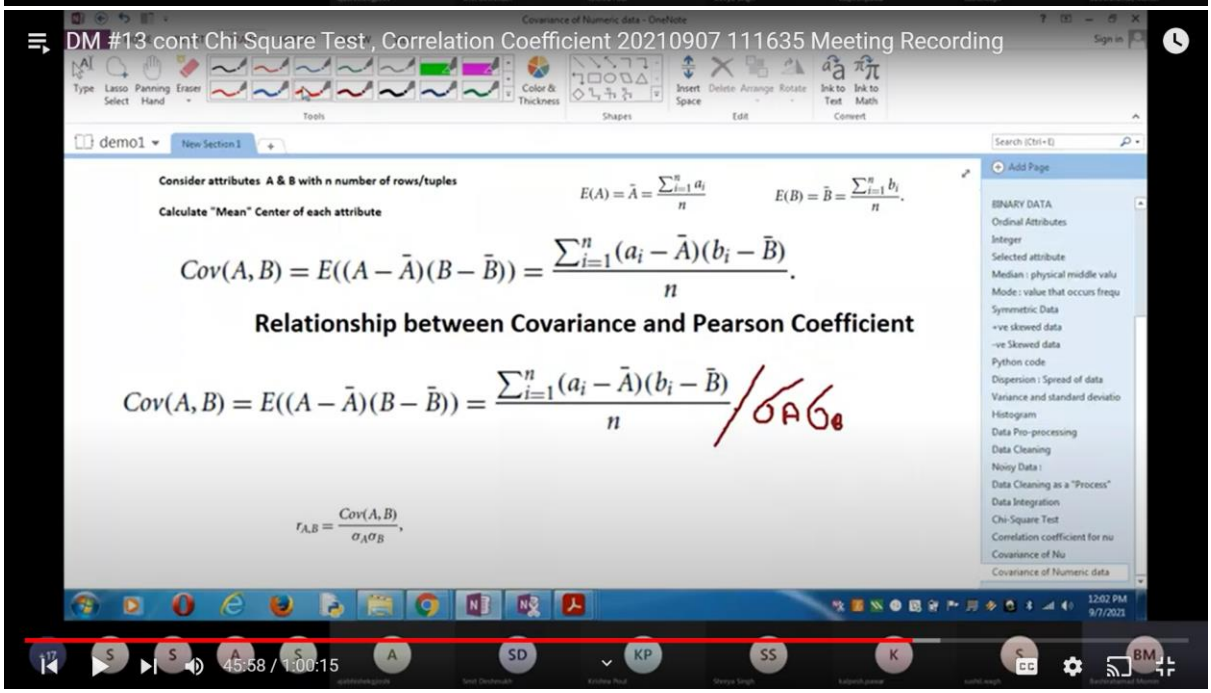

DM #13 cont Chi Square Test , Correlation Coefficient 20210907 111635 Meeting Recording
Test parameters : Significance level , degree of freedom
Degree of freedom = (r-1)(c-1) = (2-1)(2-1) = 1
How to choose significance level alpha
Risk factor or accuracy in % 0.1% = 0.1% 0.001
Level of Significance α
df 0.200 0.100 0.075 0.050 0.025 0.010 0.005 0.001 0.0005
1 1.642 2.706 3.170 3.841 5.024 6.635 7.879 10.828 12.116
Conclusion :
If calculated chi-test value > std, then two attributes are strongly correlated
18:20 / 1:00:15

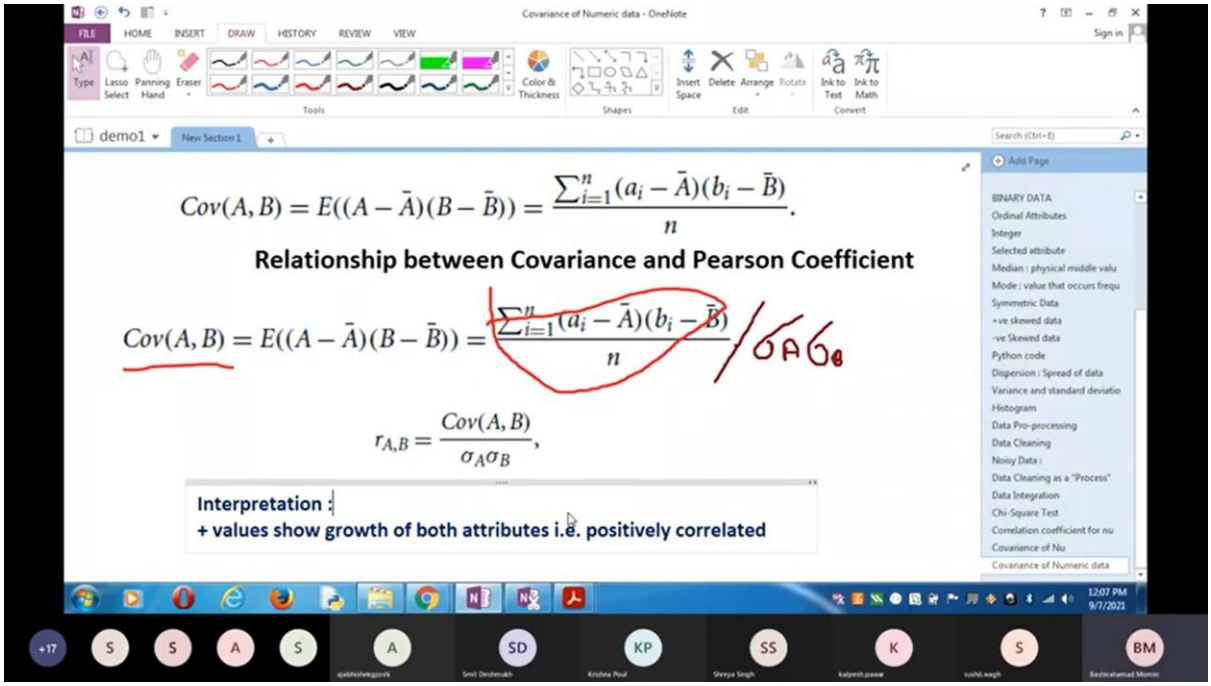

DM #13 cont Chi Square Test , Correlation Coefficient 20210907 111635 Meeting Recording
Correlation coefficient for numeric data (pearson coefficient)
$$r_{A,B} = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{n\sigma_A\sigma_B} = \frac{\sum_{i=1}^{n}(a_i b_i) - n\bar{A}\bar{B}}{n\sigma_A\sigma_B},$$
Where
AI AND Bi are the attribute values
A bar & B bar = "MEAN" center of attribute
Sigma A and Sigma B = Std. deviation
N is the no. of tuples/rows/records in data set
AiBi are dot product
$$-1 \leq r_{A,B} \leq +1$$
If Rab is greater than 0 then A & B are positively correlated.
28:18 / 1:00:15

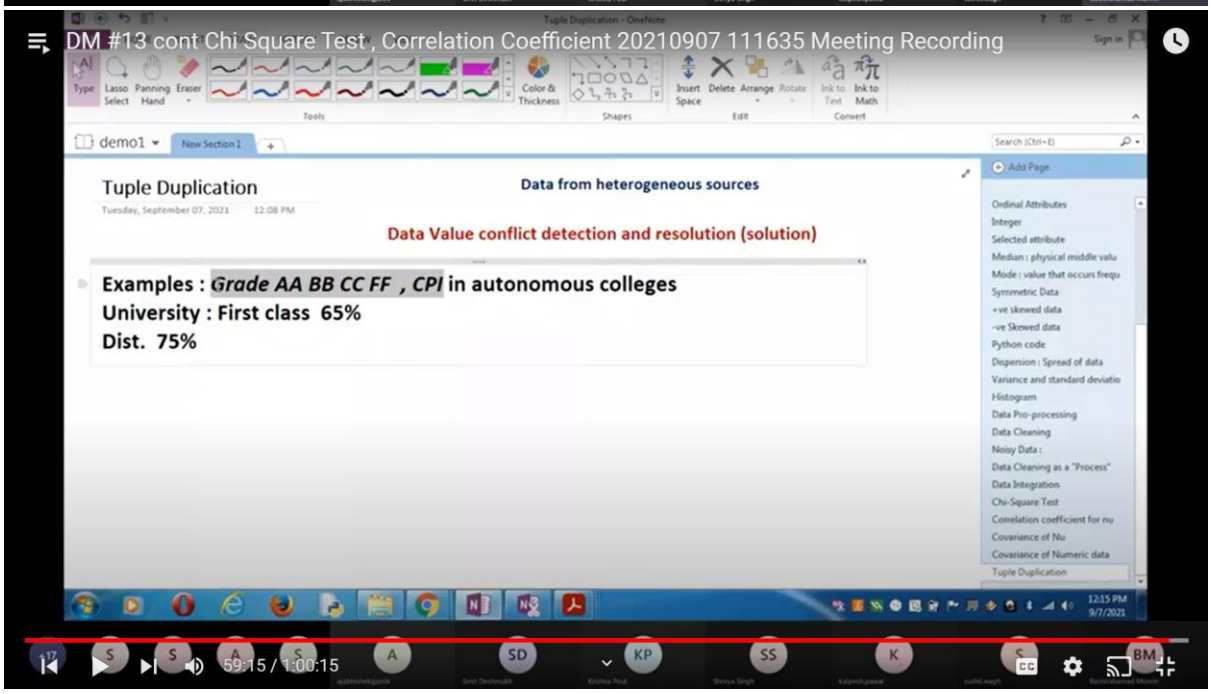

DM #13 cont Chi Square Test , Correlation Coefficient 20210907 111635 Meeting Recording
coefficient)
Tuesday, September 07, 2021 11:37 AM
$$r_{A,B} = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{n\sigma_A\sigma_B} = \frac{\sum_{i=1}^{n}(a_i b_i) - n\bar{A}\bar{B}}{n\sigma_A\sigma_B},$$
Where
AI AND Bi are the attribute values
A bar & B bar = "MEAN" center of attribute
Sigma A and Sigma B = Std. deviation
N is the no. of tuples/rows/records in data set
AiBi are dot product
$$-1 \leq r_{A,B} \leq +1$$
If Rab is greater than 0 then A & B are positively correlated.
Else negatvie
28:25 / 1:00:15


$$r = \frac{n\Sigma xy - \Sigma x \Sigma y}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}}$$

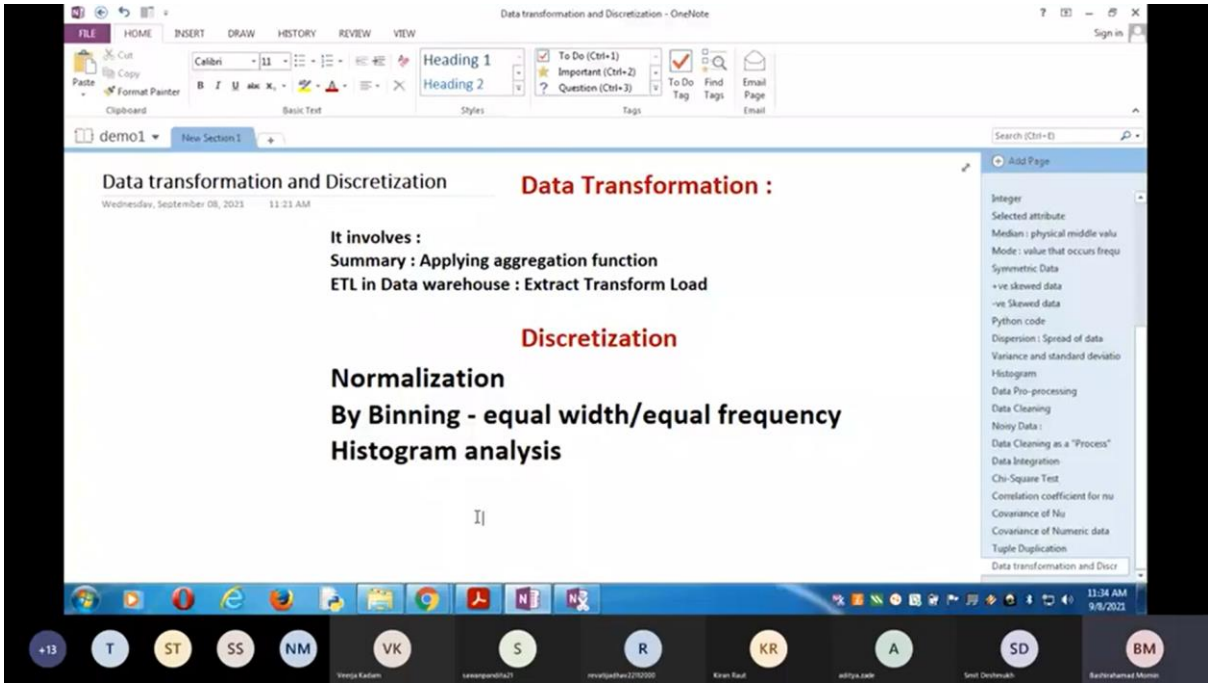

DM #13 cont Chi Square Test , Correlation Coefficient 20210907 111635 Meeting Recording
Press Esc to exit full screen
Covariance of Numeric data - OneNote
demo1
New Section 1
Covariance of Numeric data
Tuesday, September 07, 2021 11:51 AM
Consider attributes A & B with n number of rows/tuples
Calculate "Mean" Center of each attribute
$E(A) = \bar{A} = \frac{\sum_{i=1}^{n} a_i}{n}$
$E(B) = \bar{B} = \frac{\sum_{i=1}^{n} b_i}{n}$.
$Cov(A, B) = E((A - \bar{A})(B - \bar{B})) = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{n}$.
BINARY DATA
Ordinal Attributes
Integer
Selected attribute
Median : physical middle valu
Mode : value that occurs frequ
Symmetric Data
+ve skewed data
-ve Skewed data
Python code
Dispersion : Spread of data
Variance and standard deviatio
Histogram
Data Pro-processing
Data Cleaning
Noisy Data :
Data Cleaning as a "Process"
Data Integration
Chi-Square Test
Correlation coefficient for nu
Covariance of Nu
Covariance of Numeric data
11:56 AM
9/7/2021
39:30 / 1:00:15

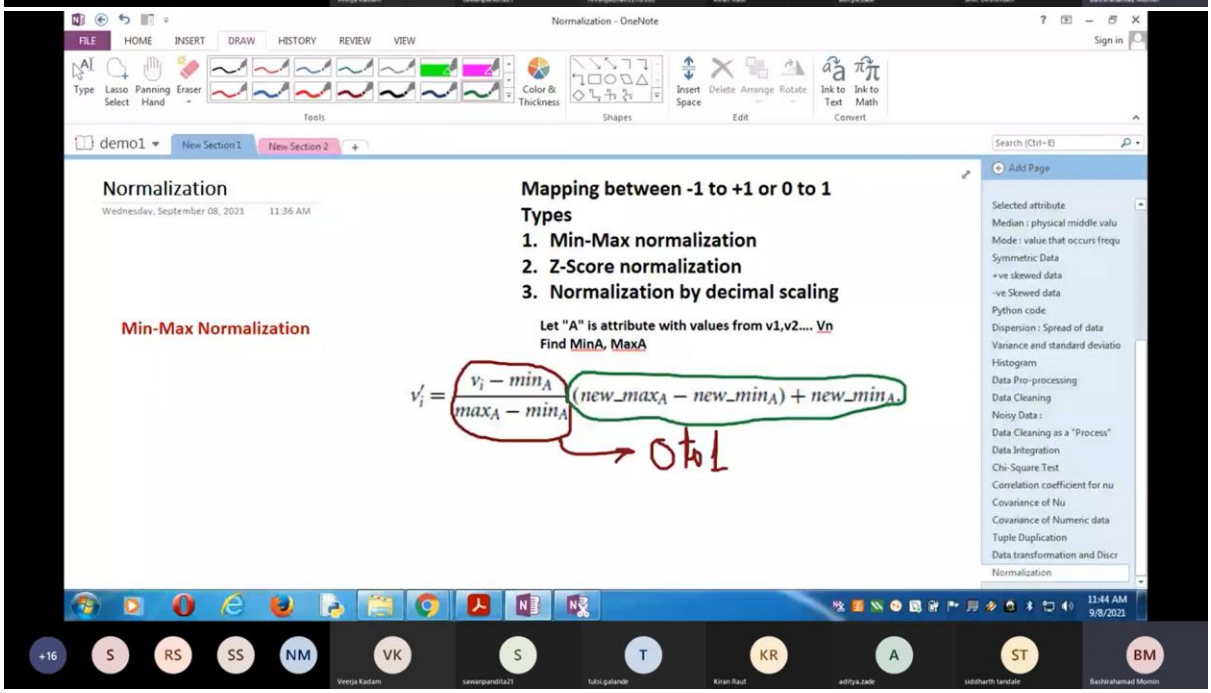

DM #13 cont Chi Square Test , Correlation Coefficient 20210907 111635 Meeting Recording
Covariance of Numeric data - OneNote
demo1
New Section 1
Consider attributes A & B with n number of rows/tuples
Calculate "Mean" Center of each attribute
$E(A) = \bar{A} = \frac{\sum_{i=1}^{n} a_i}{n}$
$E(B) = \bar{B} = \frac{\sum_{i=1}^{n} b_i}{n}$.
$Cov(A, B) = E((A - \bar{A})(B - \bar{B})) = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{n}$.
Relationship between Covariance and Pearson Coefficient
$Cov(A, B) = E((A - \bar{A})(B - \bar{B})) = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{n}$
$\sigma_A \sigma_B$
$r_{A,B} = \frac{Cov(A, B)}{\sigma_A \sigma_B}$,
12:02 PM
9/7/2021
45:58 / 1:00:15

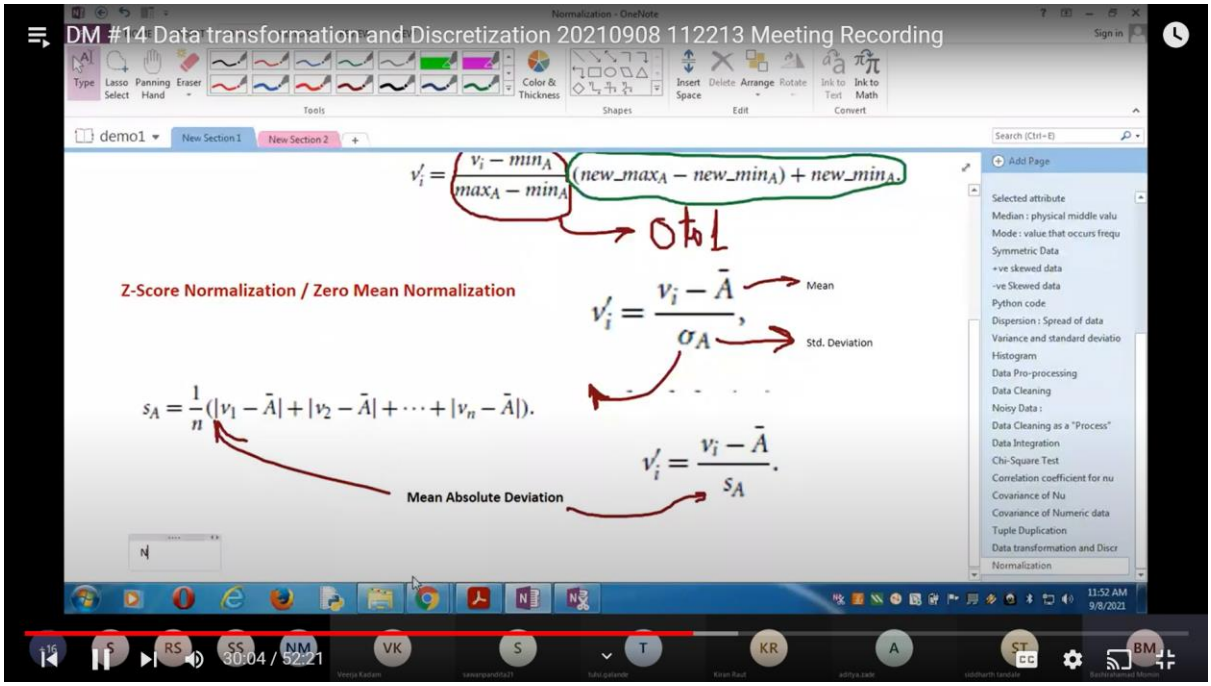

Covariance of Numeric data - OneNote
demo1
New Section 1
$Cov(A,B) = E((A-\bar{A})(B-\bar{B})) = \frac{\sum_{i=1}^{n}(a_i-\bar{A})(b_i-\bar{B})}{n}$.
Relationship between Covariance and Pearson Coefficient
$Cov(A,B) = E((A-\bar{A})(B-\bar{B})) = \frac{\sum_{i=1}^{n}(a_i-\bar{A})(b_i-\bar{B})}{n} / \sigma_A \sigma_B$
$r_{A,B} = \frac{Cov(A,B)}{\sigma_A \sigma_B}$,
Interpretation :
+ values show growth of both attributes i.e. positively correlated

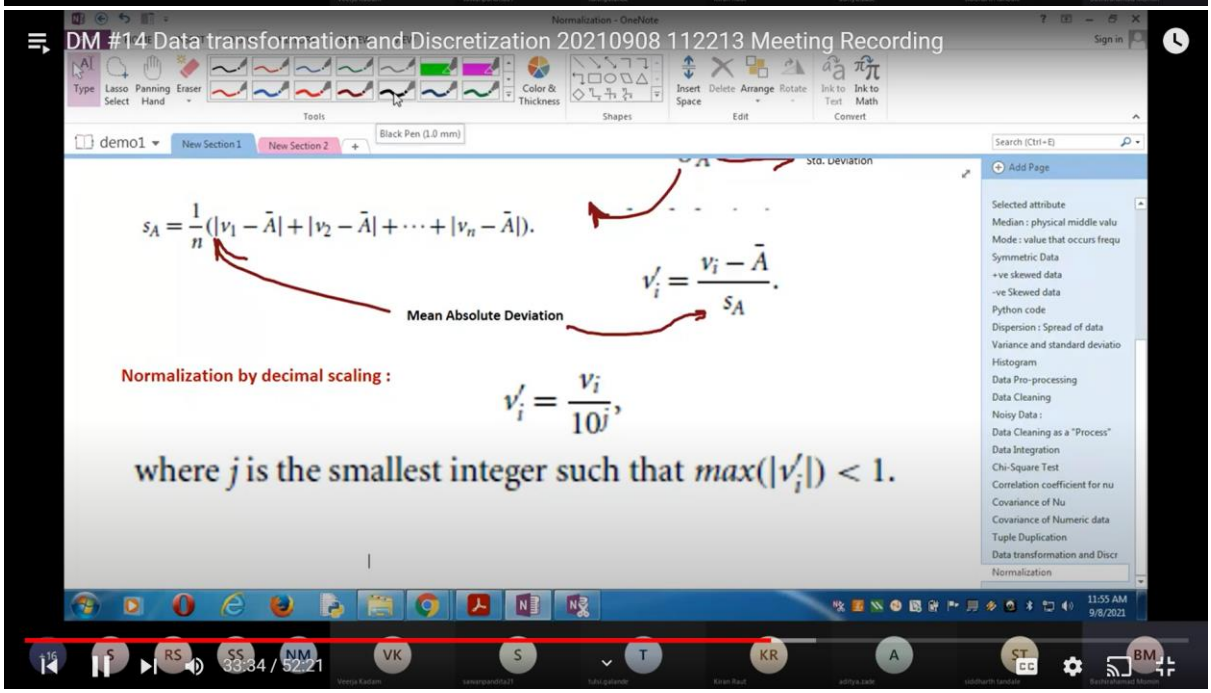

DM #13 cont Chi Square Test , Correlation Coefficient 20210907 111635 Meeting Recording
Tuple Duplication - OneNote
demo1
New Section 1
Tuple Duplication
Tuesday, September 07, 2021 12:08 PM
Data from heterogeneous sources
Data Value conflict detection and resolution (solution)
Examples : Grade AA BB CC FF , CPI in autonomous colleges
University : First class 65%
Dist. 75%
59:15 / 1:00:15

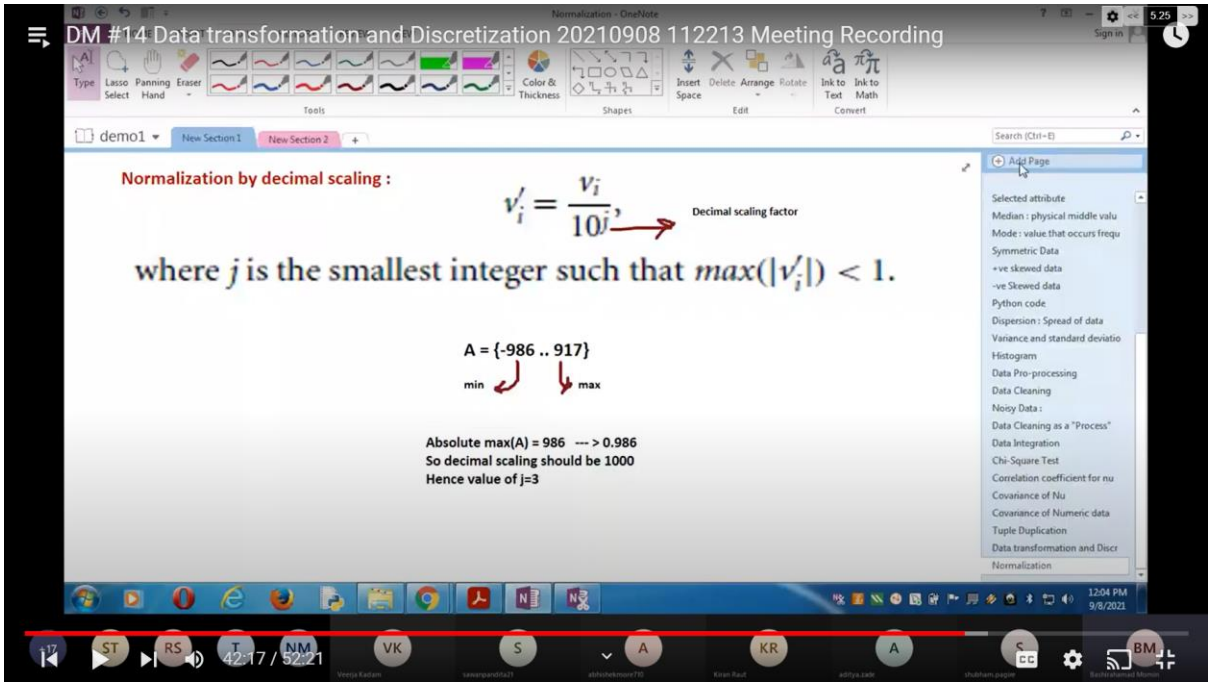
Lecture 14

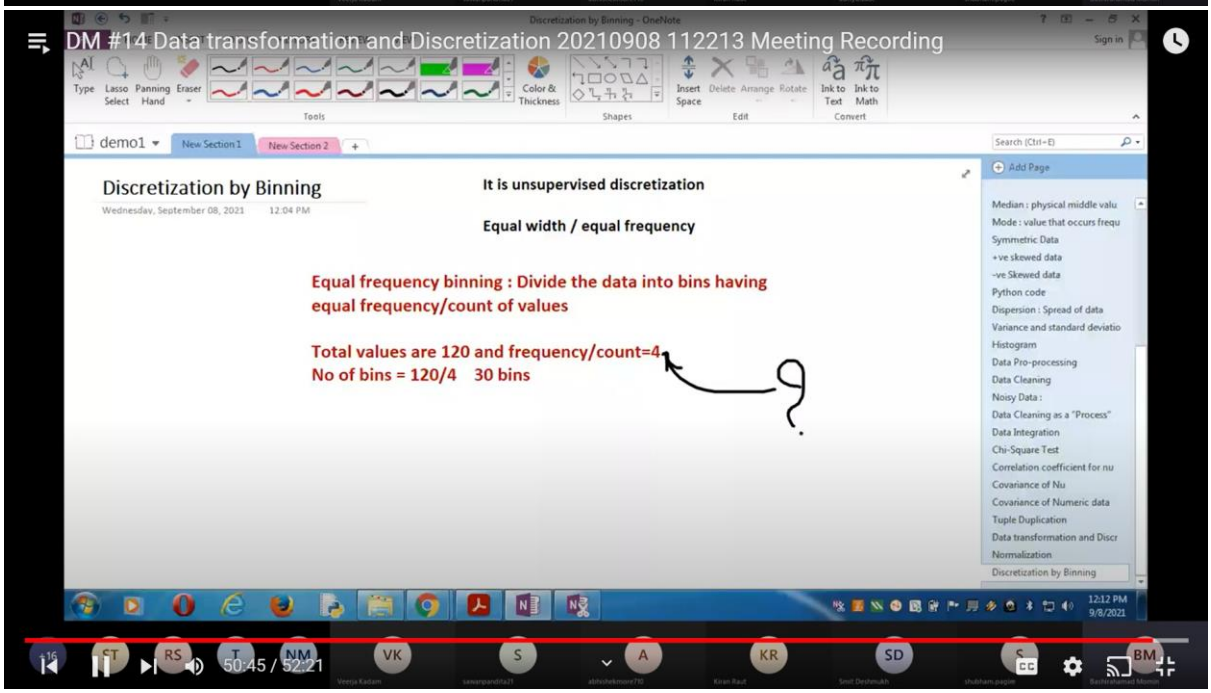

Data transformation and Discretization - OneNote
Data transformation and Discretization
Wednesday, September 08, 2021 11:21 AM
Data Transformation :
It involves :
Summary : Applying aggregation function
ETL in Data warehouse : Extract Transform Load
Discretization
Normalization
By Binning - equal width/equal frequency
Histogram analysis

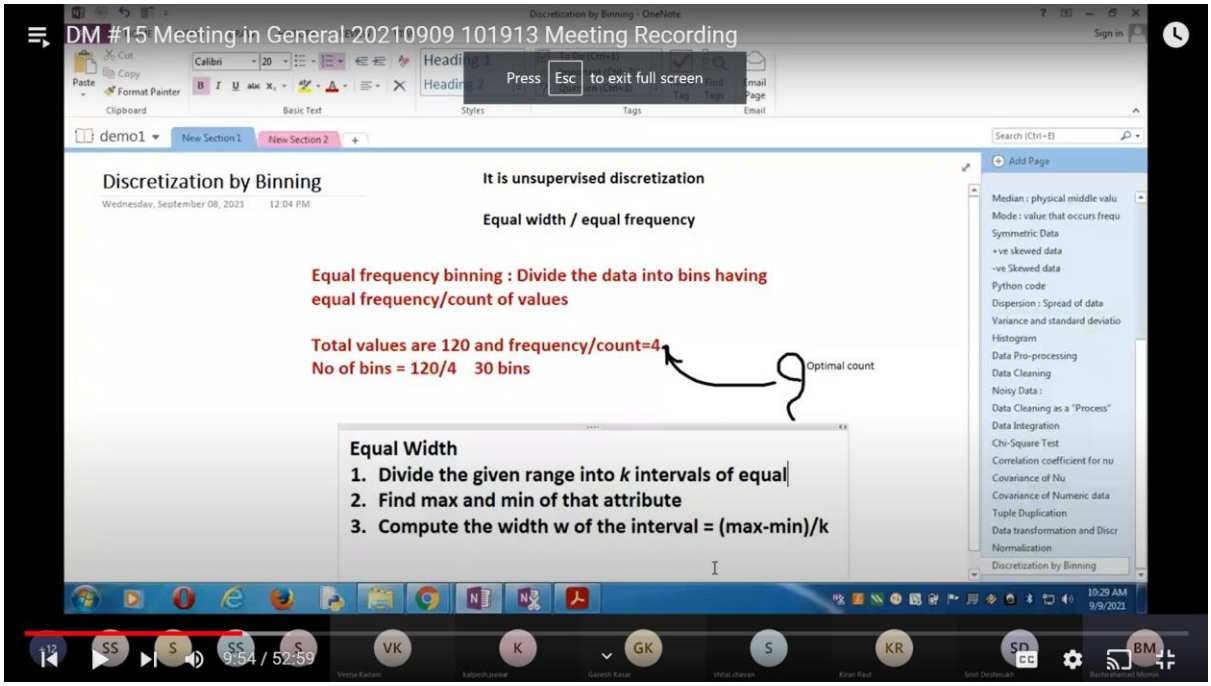

Normalization - OneNote
Normalization
Wednesday, September 08, 2021 11:36 AM
Mapping between -1 to +1 or 0 to 1
Types
1. Min-Max normalization
2. Z-Score normalization
3. Normalization by decimal scaling
Min-Max Normalization
Let "A" is attribute with values from v1,v2.... Vn
Find MinA, MaxA
$v_i' = \frac{v_i - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$
0 to 1

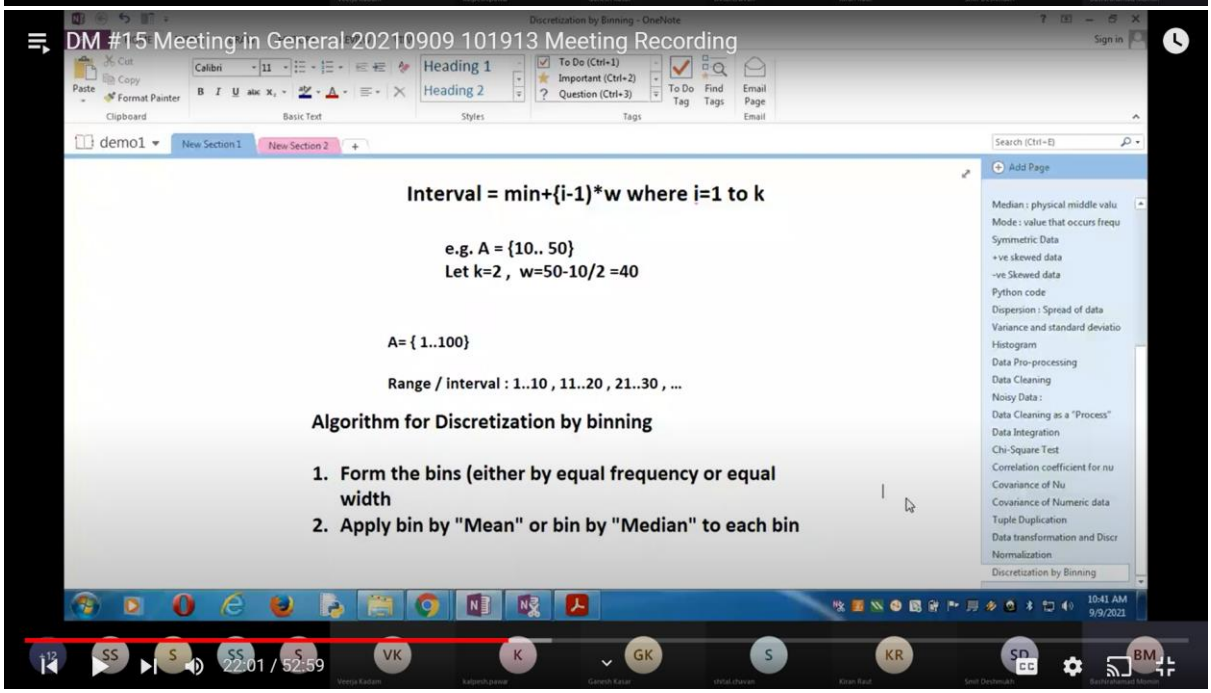

DM #14 Data transformation and Discretization 20210908 112213 Meeting Recording
Normalization - OneNote
demo1
New Section 1
New Section 2
$v'_i = \frac{v_i - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A.$
0 to 1
Z-Score Normalization / Zero Mean Normalization
$v'_i = \frac{v_i - \bar{A}}{\sigma_A}$,
Mean
Std. Deviation
$s_A = \frac{1}{n}(|v_1 - \bar{A}| + |v_2 - \bar{A}| + \cdots + |v_n - \bar{A}|).$
Mean Absolute Deviation
$v'_i = \frac{v_i - \bar{A}}{s_A}$.
Selected attribute
Median : physical middle valu
Mode : value that occurs frequ
Symmetric Data
+ve skewed data
-ve Skewed data
Python code
Dispersion : Spread of data
Variance and standard deviatio
Histogram
Data Pro-processing
Data Cleaning
Noisy Data :
Data Cleaning as a "Process"
Data Integration
Chi-Square Test
Correlation coefficient for nu
Covariance of Nu
Covariance of Numeric data
Tuple Duplication
Data transformation and Discr
Normalization
30:04 / 52:21

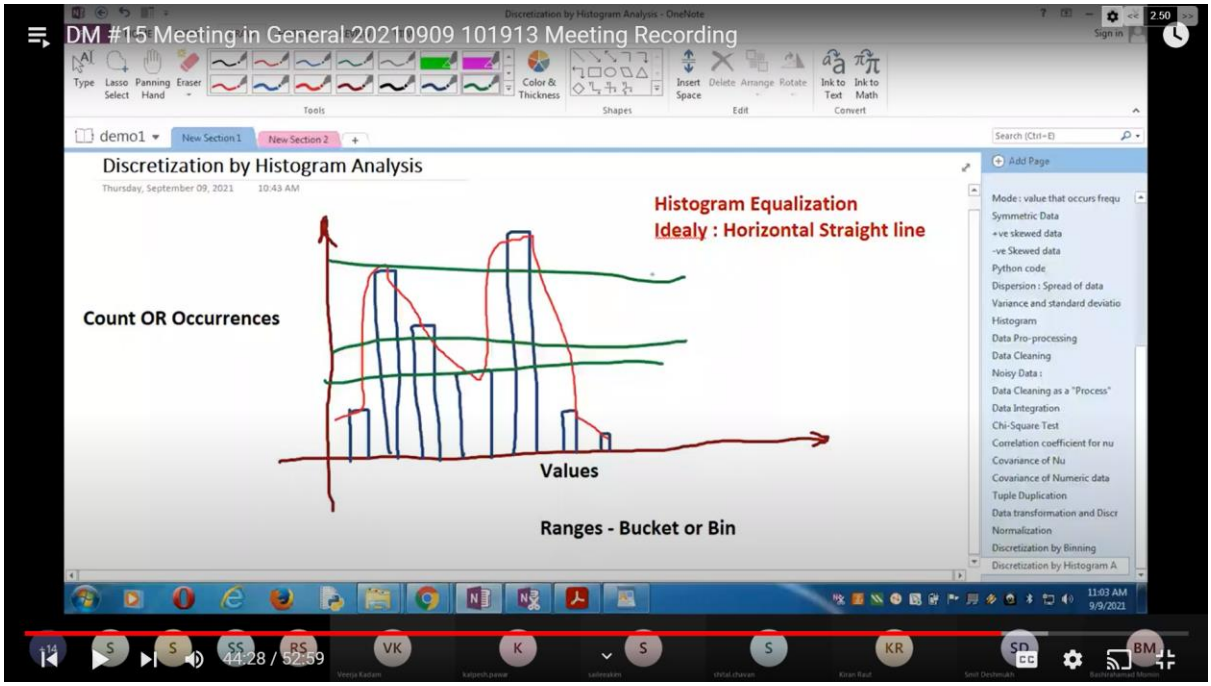

DM #14 Data transformation and Discretization 20210908 112213 Meeting Recording
Normalization - OneNote
demo1
New Section 1
New Section 2
Black Pen (1.0 mm)
Std. Deviation
$s_A = \frac{1}{n}(|v_1 - \bar{A}| + |v_2 - \bar{A}| + \cdots + |v_n - \bar{A}|).$
Mean Absolute Deviation
$v'_i = \frac{v_i - \bar{A}}{s_A}$.
Normalization by decimal scaling :
$v'_i = \frac{v_i}{10^j}$,
where j is the smallest integer such that $max(|v'_i|) < 1$.
Selected attribute
Median : physical middle valu
Mode : value that occurs frequ
Symmetric Data
+ve skewed data
-ve Skewed data
Python code
Dispersion : Spread of data
Variance and standard deviatio
Histogram
Data Pro-processing
Data Cleaning
Noisy Data :
Data Cleaning as a "Process"
Data Integration
Chi-Square Test
Correlation coefficient for nu
Covariance of Nu
Covariance of Numeric data
Tuple Duplication
Data transformation and Discr
Normalization
33:34 / 52:21

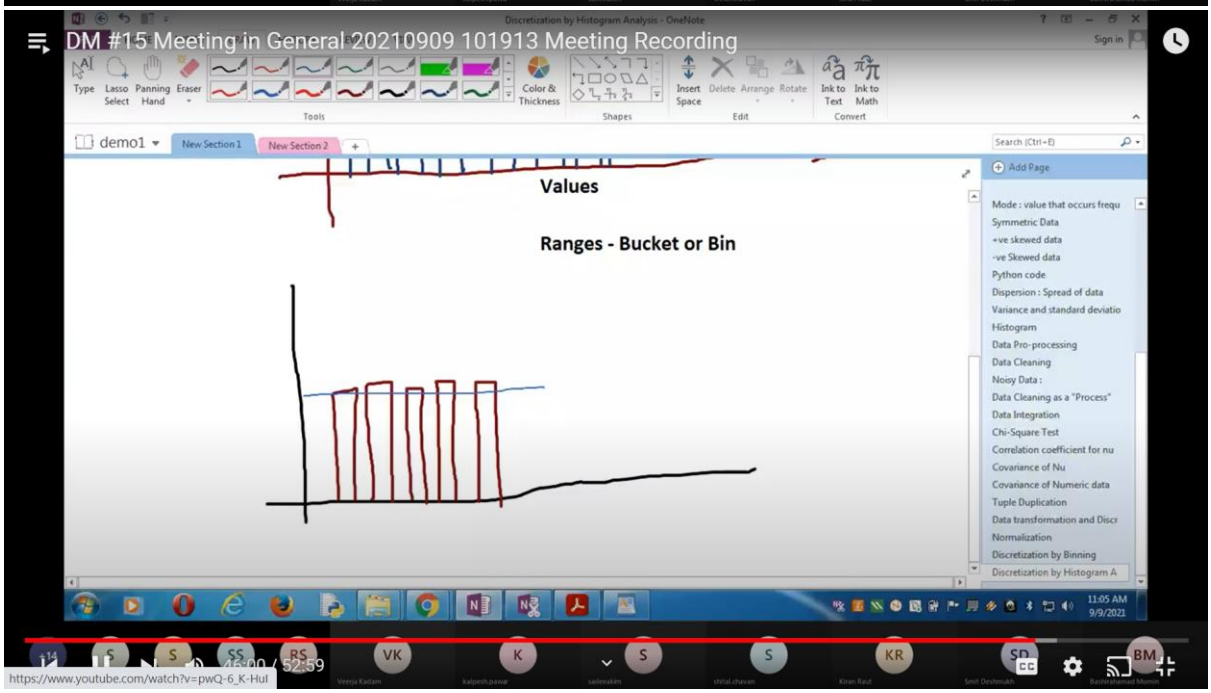

DM #14 Data transformation and Discretization 20210908 112213 Meeting Recording
Normalization - OneNote
demo1
New Section 1
New Section 2
Normalization by decimal scaling :
$v'_i = \frac{v_i}{10^j}$,
Decimal scaling factor
where j is the smallest integer such that $max(|v'_i|) < 1$.
A = {-986 .. 917}
min
max
Absolute max(A) = 986 ---> 0.986
So decimal scaling should be 1000
Hence value of j=3
Add Page
Selected attribute
Median : physical middle valu
Mode : value that occurs frequ
Symmetric Data
+ve skewed data
-ve Skewed data
Python code
Dispersion : Spread of data
Variance and standard deviatio
Histogram
Data Pro-processing
Data Cleaning
Noisy Data :
Data Cleaning as a "Process"
Data Integration
Chi-Square Test
Correlation coefficient for nu
Covariance of Nu
Covariance of Numeric data
Tuple Duplication
Data transformation and Discr
Normalization
12:04 PM
9/8/2021
42:17 / 52:21


DM #14 Data transformation and Discretization 20210908 112213 Meeting Recording
Discretization by Binning - OneNote
demo1
New Section 1
New Section 2
Discretization by Binning
Wednesday, September 08, 2021 12:04 PM
It is unsupervised discretization
Equal width / equal frequency
Equal frequency binning : Divide the data into bins having equal frequency/count of values
Total values are 120 and frequency/count=4
No of bins = 120/4 30 bins
Add Page
Median : physical middle valu
Mode : value that occurs frequ
Symmetric Data
+ve skewed data
-ve Skewed data
Python code
Dispersion : Spread of data
Variance and standard deviatio
Histogram
Data Pro-processing
Data Cleaning
Noisy Data :
Data Cleaning as a "Process"
Data Integration
Chi-Square Test
Correlation coefficient for nu
Covariance of Nu
Covariance of Numeric data
Tuple Duplication
Data transformation and Discr
Normalization
Discretization by Binning
12:12 PM
9/8/2021
50:45 / 52:21

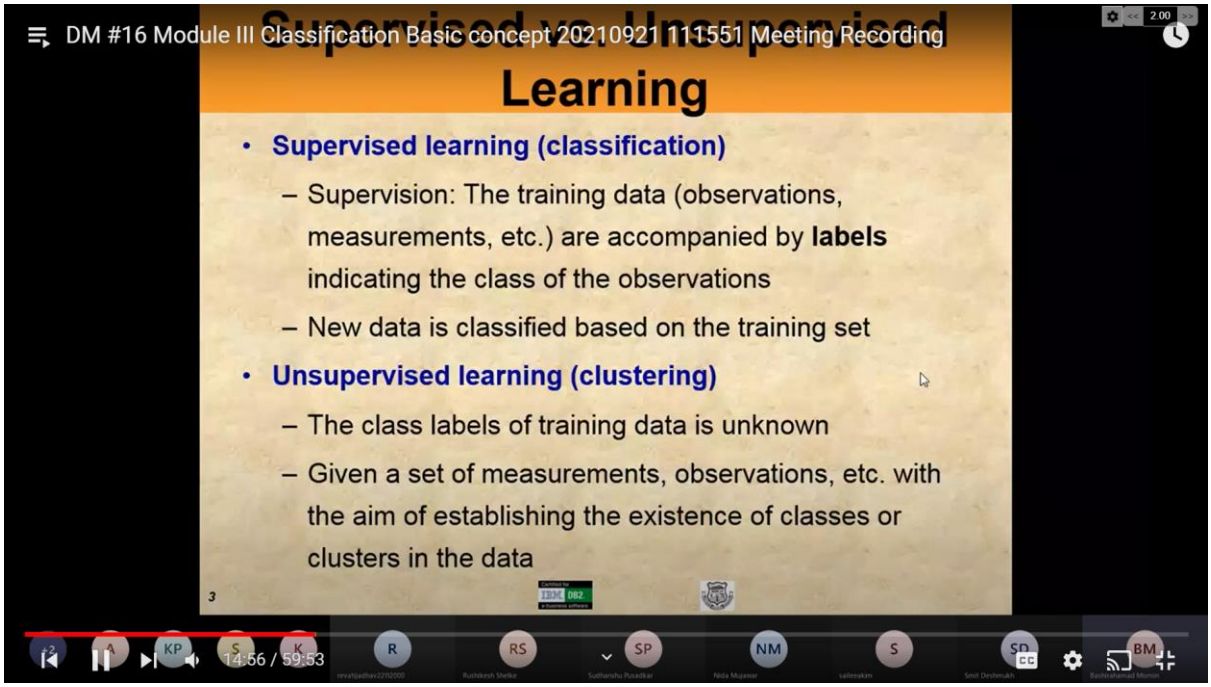

DM #15 Meeting in General-20210909 101913 Meeting Recording
Press Esc to exit full screen
Discretization by Binning
Wednesday, September 08, 2021 12:04 PM
It is unsupervised discretization
Equal width / equal frequency
Equal frequency binning : Divide the data into bins having equal frequency/count of values
Total values are 120 and frequency/count=4
No of bins = 120/4 30 bins
Optimal count
Equal Width
1. Divide the given range into k intervals of equal
2. Find max and min of that attribute
3. Compute the width w of the interval = (max-min)/k
9:54 / 52:59

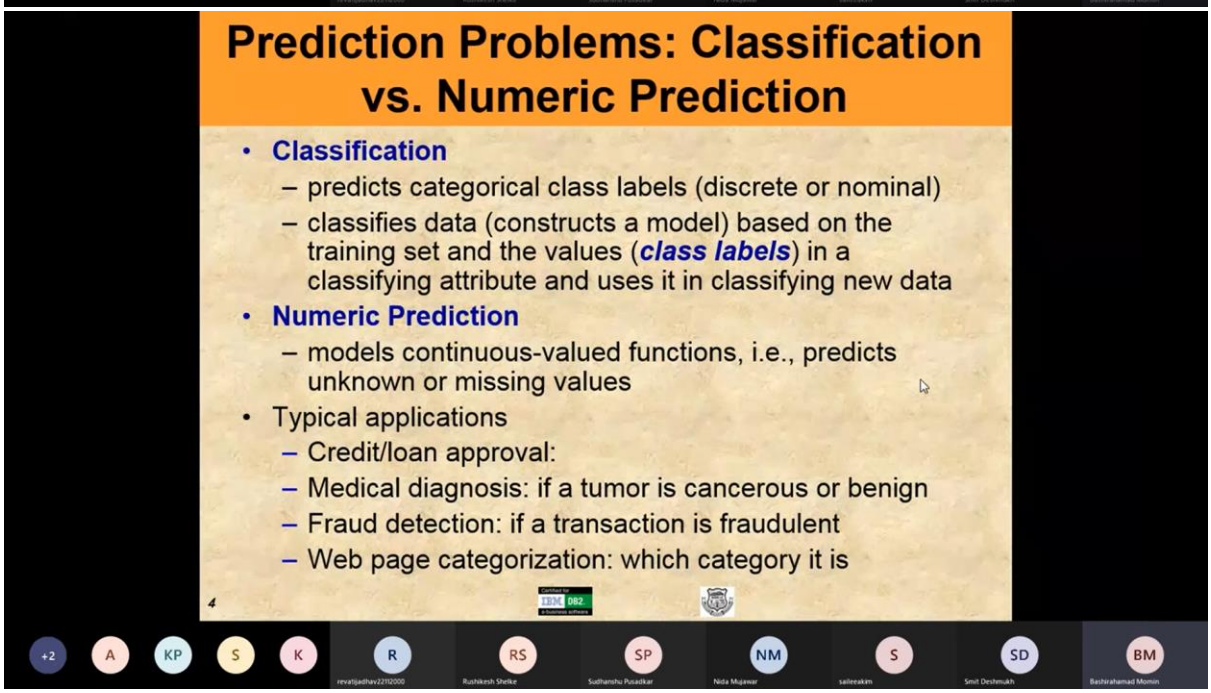

DM #15 Meeting in General-20210909 101913 Meeting Recording
Interval = min+{i-1)*w where i=1 to k
e.g. A = {10.. 50}
Let k=2 , w=50-10/2 =40
A= { 1..100}
Range / interval : 1..10 , 11..20 , 21..30 , ...
Algorithm for Discretization by binning
1. Form the bins (either by equal frequency or equal width
2. Apply bin by "Mean" or bin by "Median" to each bin
22:01 / 52:59


DM #15 Meeting in General-20210909 101913 Meeting Recording
Discretization by Histogram Analysis
Thursday, September 09, 2021 10:43 AM
Histogram Equalization
Idealy : Horizontal Straight line
Count OR Occurrences
Values
Ranges - Bucket or Bin
44:28 / 52:59


DM #15 Meeting in General-20210909 101913 Meeting Recording
Values
Ranges - Bucket or Bin
45:00 / 52:59
https://www.youtube.com/watch?v=pwQ-6_K-Hul

Lecture 16


DM #16 Module III Classification Basic concept 20210921 111551 Meeting Recording
Supervised vs. Unsupervised Learning
• Supervised learning (classification)
– Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations
– New data is classified based on the training set
• Unsupervised learning (clustering)
– The class labels of training data is unknown
– Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data
3
14:56 / 59:53


Prediction Problems: Classification vs. Numeric Prediction
• Classification
– predicts categorical class labels (discrete or nominal)
– classifies data (constructs a model) based on the training set and the values (class labels) in a classifying attribute and uses it in classifying new data
• Numeric Prediction
– models continuous-valued functions, i.e., predicts unknown or missing values
• Typical applications
– Credit/loan approval:
– Medical diagnosis: if a tumor is cancerous or benign
– Fraud detection: if a transaction is fraudulent
– Web page categorization: which category it is
4

Classification - Definition

- Method of categorizing or assigning class labels to a pattern set under supervision of teacher : *Supervised Learning.*
- The decision boundaries are generated to discriminate between patterns of different classes.
- Prediction of class from set of samples
- Different methods :
  - *Decision Trees*
  - *Probabilistic or generative models*
  - *Nearest Neighbor classifier*
  - *Artificial Neural Network (ANN)*

5

Classification Problem

- Given a database $D=\{t_1,t_2,\ldots,t_n\}$ and a set of classes $C=\{C_1,\ldots,C_m\}$, the ***Classification Problem*** is to define a mapping $f:D\rightarrow C$ where each t_i is assigned to one class.
- Actually divides D into ***equivalence*** *classes*.
- ***Prediction*** is similar, but may be viewed as having infinite number of classes.

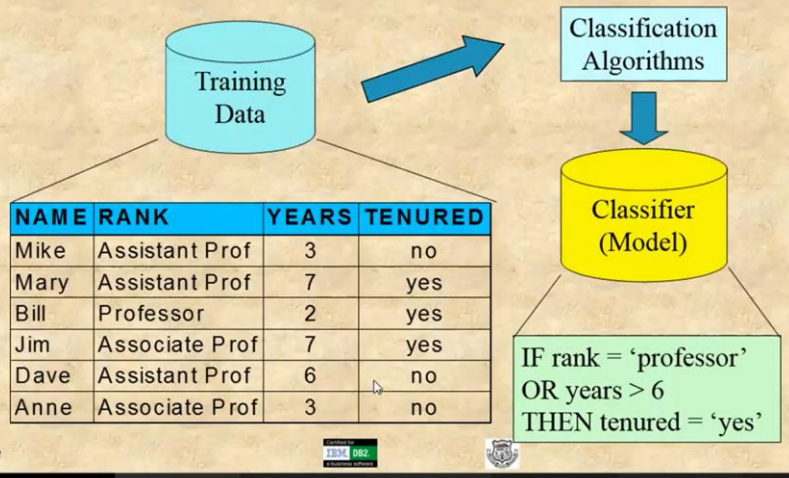
6

Classification—A Two-Step Process

- **Model construction**: describing a set of predetermined classes
  - Each tuple/sample is assumed to belong to a predefined class, as determined by the **class label attribute**
  - The set of tuples used for model construction is **training set**
  - The model is represented as classification rules, decision trees, or mathematical formulae
- **Model usage**: for classifying future or unknown objects
  - **Estimate accuracy** of the model
    - The known label of test sample is compared with the classified result from the model
    - Accuracy rate is the percentage of test set samples that are correctly classified by the model
    - Test set is independent of training set, otherwise over-fitting will occur
  - If the accuracy is acceptable, use the model to **classify data** tuples whose class labels are not known

7

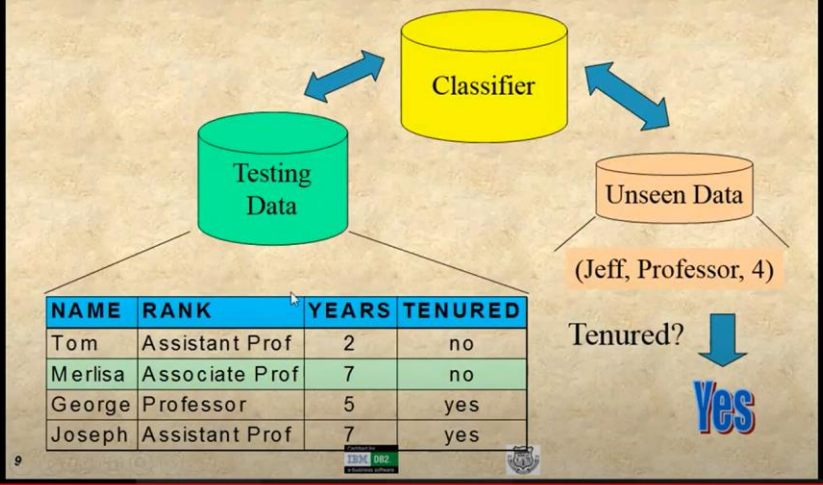
Process (1): Model Construction



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

8

DM #16 Module III Classification Basic concept 20210921 111551 Meeting Recor...



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

9

52:52 / 59:53

Classification Examples

- Teachers classify students' grades as A, B, C, D, or F.
- Identify mushrooms as poisonous or edible.
- Predict when a river will flood.
- Identify individuals with credit risks.
- Speech recognition
- Pattern recognition

10

Classification Ex: Letter Recognition

View letters as constructed from 5 components:

Featues : Strokes , Tees , joint , end points etc

 Letter A

 Letter B

 Letter C

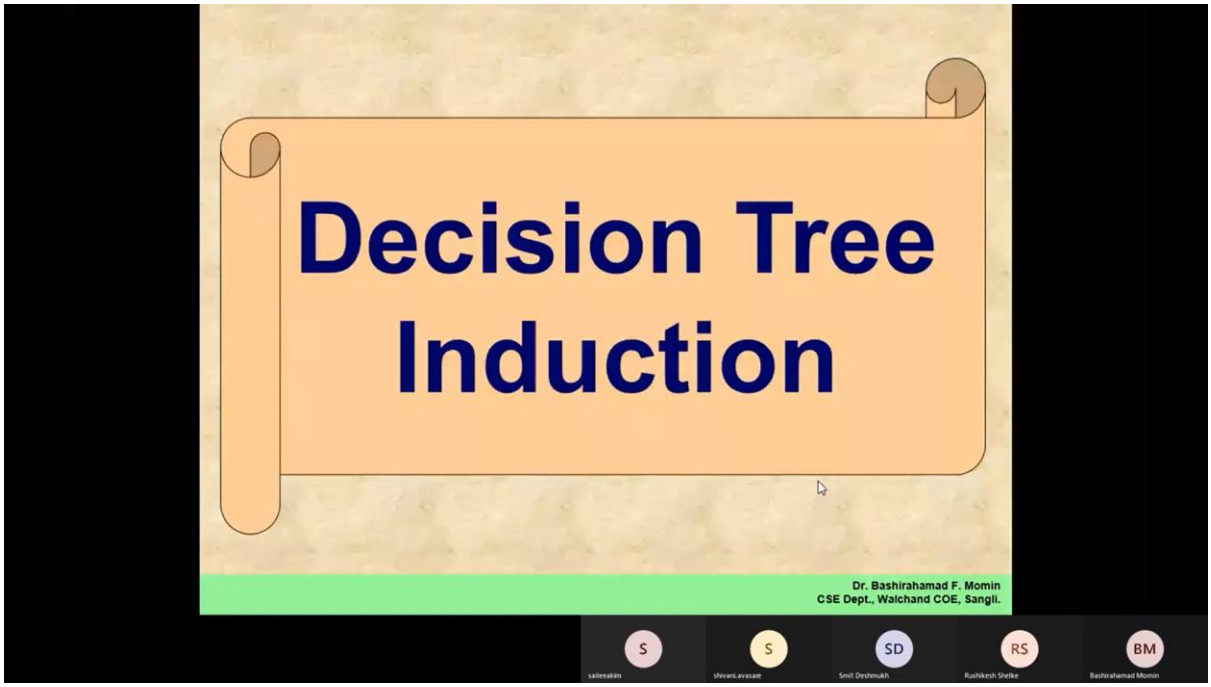
 Letter D

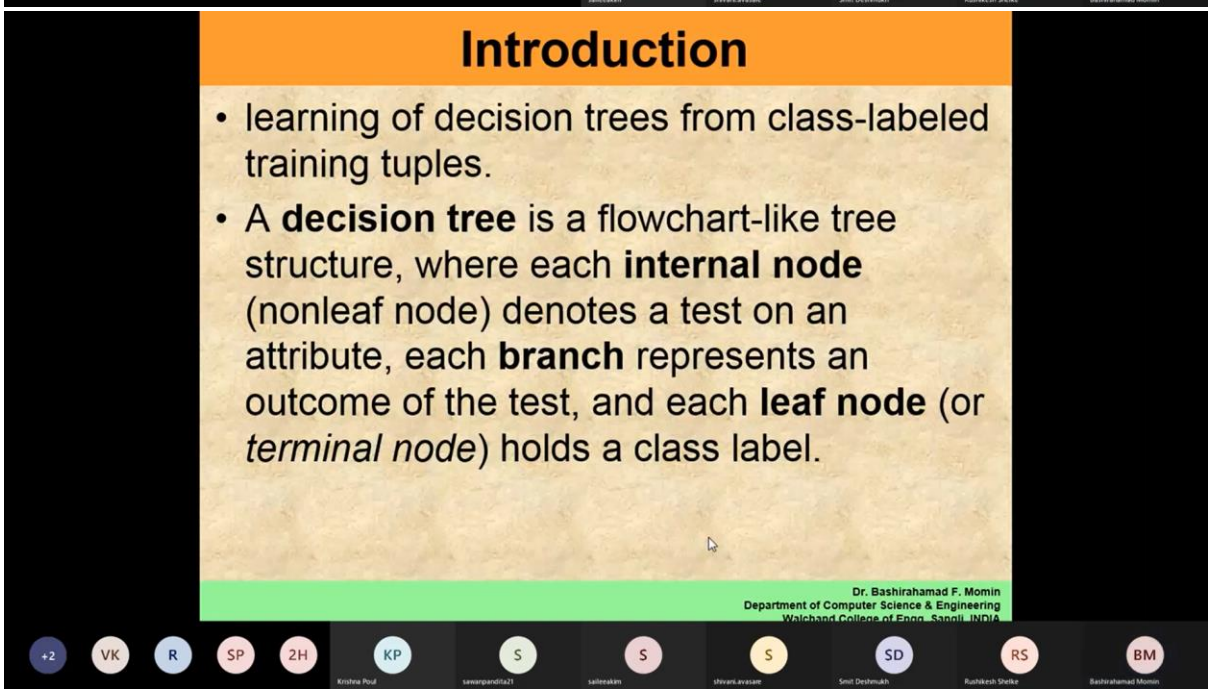
 Letter E

 Letter F

11

Lecture 17


Decision Tree Induction
Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.


Introduction
• learning of decision trees from class-labeled training tuples.
• A **decision tree** is a flowchart-like tree structure, where each **internal node** (nonleaf node) denotes a test on an attribute, each **branch** represents an outcome of the test, and each **leaf node** (or *terminal node*) holds a class label.
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli. INDIA

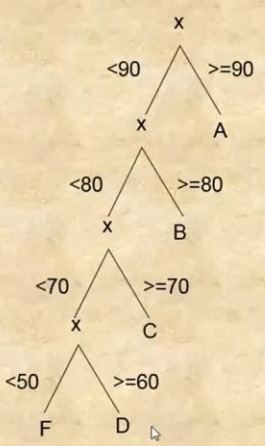
...Cont Introduction

- Internal nodes are denoted by ***rectangles***, and leaf nodes are denoted by ***ovals***.
- Some decision tree algorithms produce only ***binary*** trees (where each internal node branches to exactly two other nodes), whereas others can produce ***nonbinary*** trees.

Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli, INDIA

Decision Tree for Grading

- If x >= 90 then grade =A.
- If 80<=x<90 then grade =B.
- If 70<=x<80 then grade =C.
- If 60<=x<70 then grade =D.
- If x<50 then grade =F.

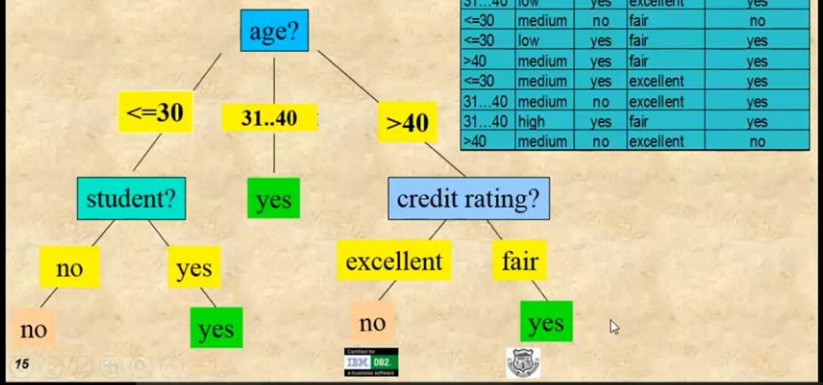


+2 VK R SP 2H KP S S S SD RS BM
Krishna Poul sawanpandita21 saileeakim shivani.avasare Smit Deshmukh Rushikesh Shelke Bashirahamad Momin

Decision Tree Induction: An Example

- Training data set: Buys_computer
- Resulting tree:

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no



15

How are the decision trees used for classification ?

- Given a tuple, ***X***, for which the associated class label is unknown, the attribute values of the tuple are tested against the decision tree.
- A path is traced from the root to a leaf node, which holds the class prediction for that tuple – ***tree traversal***.
- Decision trees can easily be converted to **classification rules**
- #rules = #path from root to leaf node

Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg, Sangli, INDIA

Why are decision tree classifiers so popular?

- The construction of decision tree classifiers does not require any domain knowledge or parameter setting.
- appropriate for exploratory knowledge discovery.
- can handle multidimensional data.
- representation of acquired knowledge in tree form is intuitive and easy to assimilate/understand by humans
- decision tree classifiers have good accuracy.

Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli, INDIA

How to build Decision Tree ?

Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.

History

- During the late 1970s and early 1980s, J. Ross Quinlan, a researcher in machine learning, developed a decision tree algorithm known as **ID3** (Iterative Dichotomiser).
- Quinlan later presented **C4.5** (a successor of ID3), which became a benchmark to which newer supervised learning algorithms are often compared.
- In 1984, a group of statisticians (L. Breiman, J. Friedman, R. Olshen, and C. Stone) published the book *Classification and Regression Trees* (**CART**), which described the generation of binary decision trees.
- ID3 and CART were invented independently of one another at
- around the same time, yet follow a similar approach for learning decision trees from
- training tuples.
- **ID3 , C4.5 & CART** algorithms becomes de-facto standards for decision tree induction

Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli, INDIA

DM #17 Module III Decision Tree Induction 20210922 111519 Meeting Recording

1] Constructing/building the model : Needs to train by training set
2] Evaluation / testing : test data set
Division : 80 20 or 70 30

30:10 / 1:00:22

Greedy Algorithm

- **ID3**, **C4.5**, and **CART** adopt a greedy (i.e., nonbacktracking) approach.
- decision trees are constructed in a top-down recursive divide-and-conquer manner.
- It starts with a training set of tuples and their associated class labels.
- The training set is recursively partitioned into smaller subsets as the tree is being built.

Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
Walchand College of Engg. Sangli, INDIA

Algorithm for Decision Tree Induction

- Basic algorithm (a greedy algorithm)
  - Tree is constructed in a **top-down recursive divide-and-conquer manner**
  - At start, all the training examples are at the root
  - Attributes are categorical (if continuous-valued, they are discretized in advance)
  - Examples are partitioned recursively based on selected attributes
  - Test attributes are selected on the basis of a heuristic or statistical measure – **Entropy , Gain Ratio & Gini Index**)

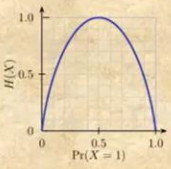
Algorithm for Decision Tree Induction

- Basic algorithm (a greedy algorithm)
  - Tree is constructed in a **top-down recursive divide-and-conquer manner**
  - At start, all the training examples are at the root
  - Attributes are categorical (if continuous-valued, they are discretized in advance)
  - Examples are partitioned recursively based on selected attributes
  - Test attributes are selected on the basis of a heuristic or statistical measure – **Entropy , Gain Ratio & Gini Index**)
- Conditions for stopping partitioning
  - All samples for a given node belong to the same class
  - There are no remaining attributes for further partitioning – **majority voting** is employed for classifying the leaf
  - There are no samples left

22

Entropy

- Entropy (Information Theory)
  - A measure of uncertainty associated with a random variable
  - Calculation: For a discrete random variable Y taking m distinct values $\{y_1, \ldots, y_m\}$,
    - $H(Y) = -\sum_{i=1}^{m} p_i \log(p_i)$, where $p_i = P(Y = y_i)$
  - Interpretation:
    - Higher entropy => higher uncertainty
    - Lower entropy => lower uncertainty
- Conditional Entropy
  - $H(Y|X) = \sum_x p(x) H(Y|X = x)$



m = 2

23

Gini Index (CART, IBM IntelligentMiner)

- If a data set D contains examples from n classes, gini index, $gini(D)$ is defined as

$$gini(D) = 1 - \sum_{j=1}^{n} p_j^2$$

where p_j is the relative frequency of class j in D

- If a data set D is split on A into two subsets D_1 and D_2, the *gini* index $gini(D)$ is defined as

$$gini_A(D) = \frac{|D_1|}{|D|} gini(D_1) + \frac{|D_2|}{|D|} gini(D_2)$$

- Reduction in Impurity:

- The attribute provides the smallest $gini_{split}(D)$ (or the largest reduction in impurity) is chosen to split the node (*need to enumerate all the possible splitting points for each attribute*)

26

Attribute Selection using Information Gain (ID3/C4.5)

- Select the attribute with the highest information gain
- Let p_i be the probability that an arbitrary tuple in D belongs to class C_i, estimated by $|C_{i,D}|/|D|$
- **Expected information** (entropy) needed to classify a tuple in D:

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

- **Information** needed (after using A to split D into v partitions) to classify D:

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j)$$

- **Information gained** by branching on attribute A

$$Gain(A) = Info(D) - Info_A(D)$$

24

ENTROPY AND INFORMATION GAIN

Day	Outlook	Temp	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

$$Values(Wind) = Weak, Strong$$

$$S = [9+, 5-]$$

$$S_{Weak} \leftarrow [6+, 2-]$$

$$S_{Strong} \leftarrow [3+, 3-]$$

$$Gain(S, Wind) = Entropy(S) - \sum_{v \in \{Weak, Strong\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$= Entropy(S) - (8/14) Entropy(S_{Weak}) - (6/14) Entropy(S_{Strong})$$

$$= 0.940 - (8/14)0.811 - (6/14)1.00$$

$$= 0.048$$

Subscribe to Mahesh Huddar Visit: vtupulse.com

Gain Ratio for Attribute Selection (C4.5)

- Information gain measure is biased towards attributes with a large number of values
- C4.5 (a successor of ID3) uses gain ratio to overcome the problem (normalization to information gain)

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

  - GainRatio(A) = Gain(A)/SplitInfo(A)
- Ex.

$$SplitInfo_{income}(D) = -\frac{4}{14} \times \log_2\left(\frac{4}{14}\right) - \frac{6}{14} \times \log_2\left(\frac{6}{14}\right) - \frac{4}{14} \times \log_2\left(\frac{4}{14}\right) = 1.557$$

  - gain_ratio(income) = 0.029/1.557 = 0.019
- The attribute with the maximum gain ratio is selected as the splitting attribute

25

DM #17 Module III Decision Tree Induction 20210922 111519 Meeting Recording

Gain Ratio for Attribute Selection (C4.5)

- Information gain measure is biased towards attributes with a large number of values
- C4.5 (a successor of ID3) uses gain ratio to overcome the problem (normalization to information gain)

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

  - GainRatio(A) = Gain(A)/SplitInfo(A)
- Ex.

$$SplitInfo_{income}(D) = -\frac{4}{14} \times \log_2\left(\frac{4}{14}\right) - \frac{6}{14} \times \log_2\left(\frac{6}{14}\right) - \frac{4}{14} \times \log_2\left(\frac{4}{14}\right) = 1.557$$

  - gain_ratio(income) = 0.029/1.557 = 0.019
- The attribute with the maximum gain ratio is selected as the splitting attribute

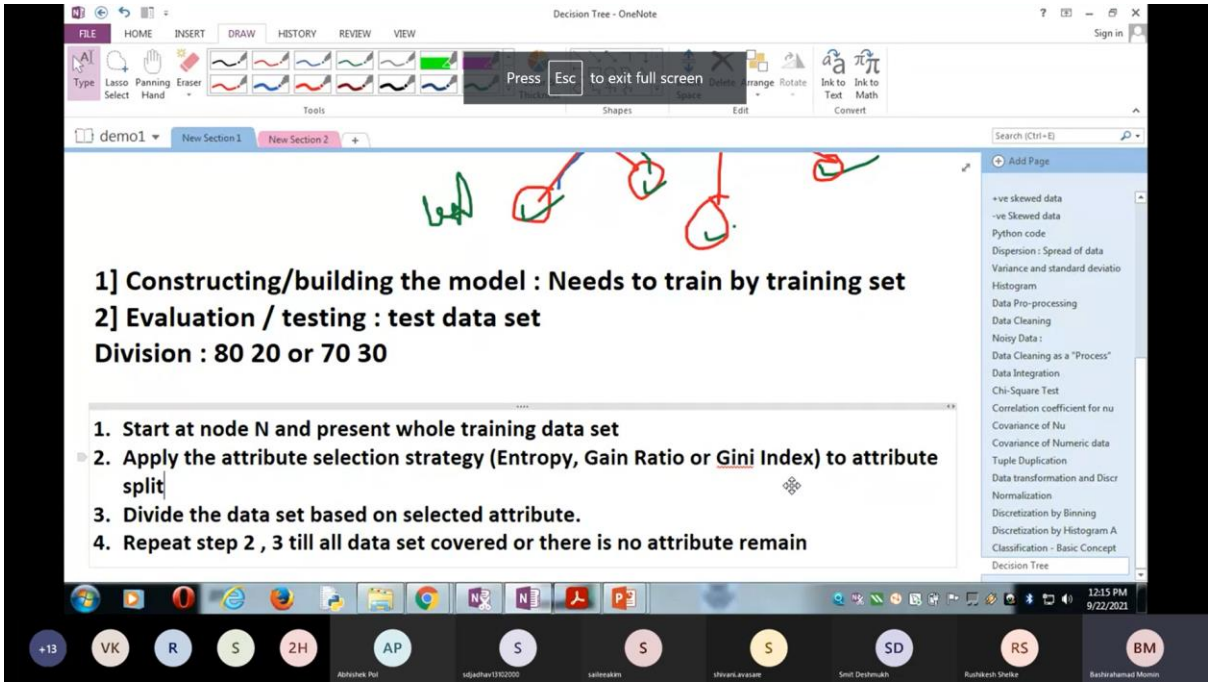
25

45:12 / 1:00:22

Comparing Attribute Selection Measures

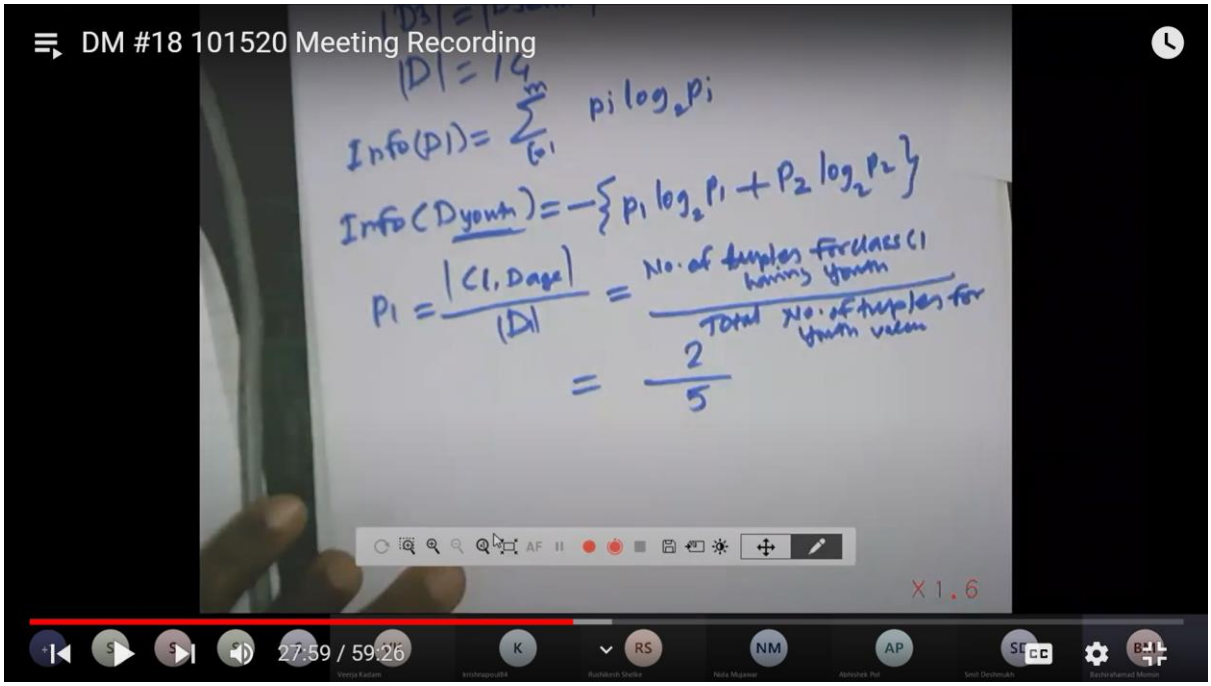
- The three measures, in general, return good results but
  - **Information gain**:
    - biased towards multivalued attributes
  - **Gain ratio**:
    - tends to prefer unbalanced splits in which one partition is much smaller than the others
  - **Gini index**:
    - biased to multivalued attributes
    - has difficulty when # of classes is large
    - tends to favor tests that result in equal-sized partitions and purity in both partitions

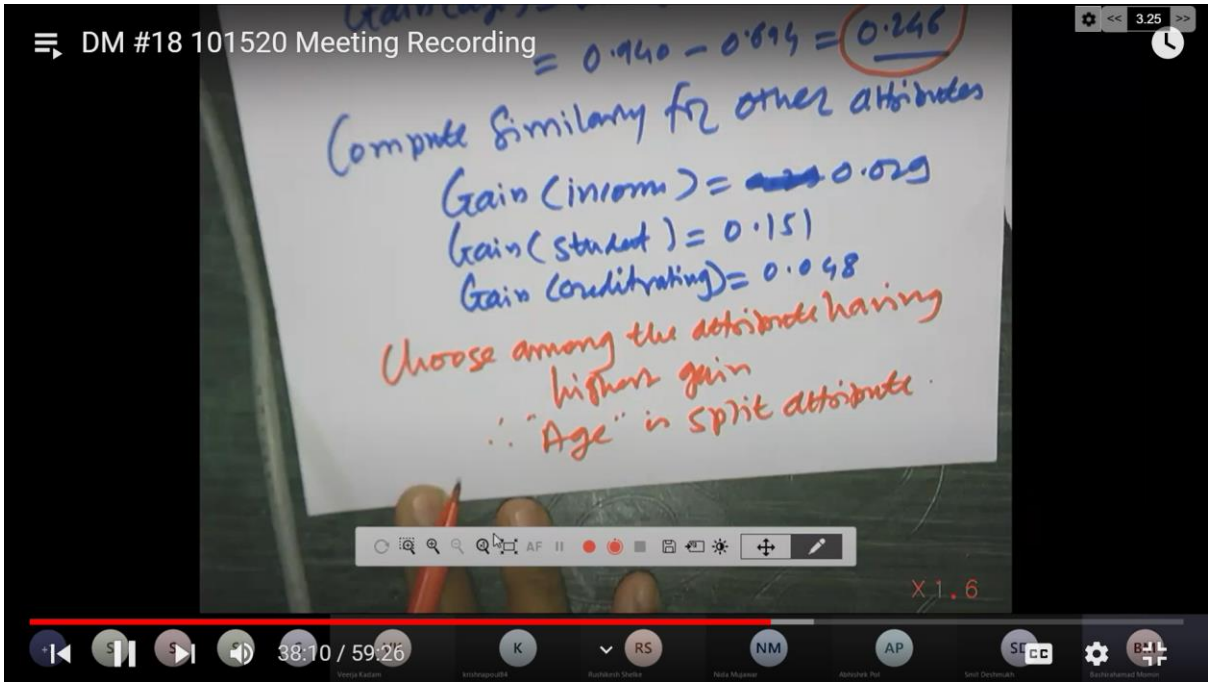
27

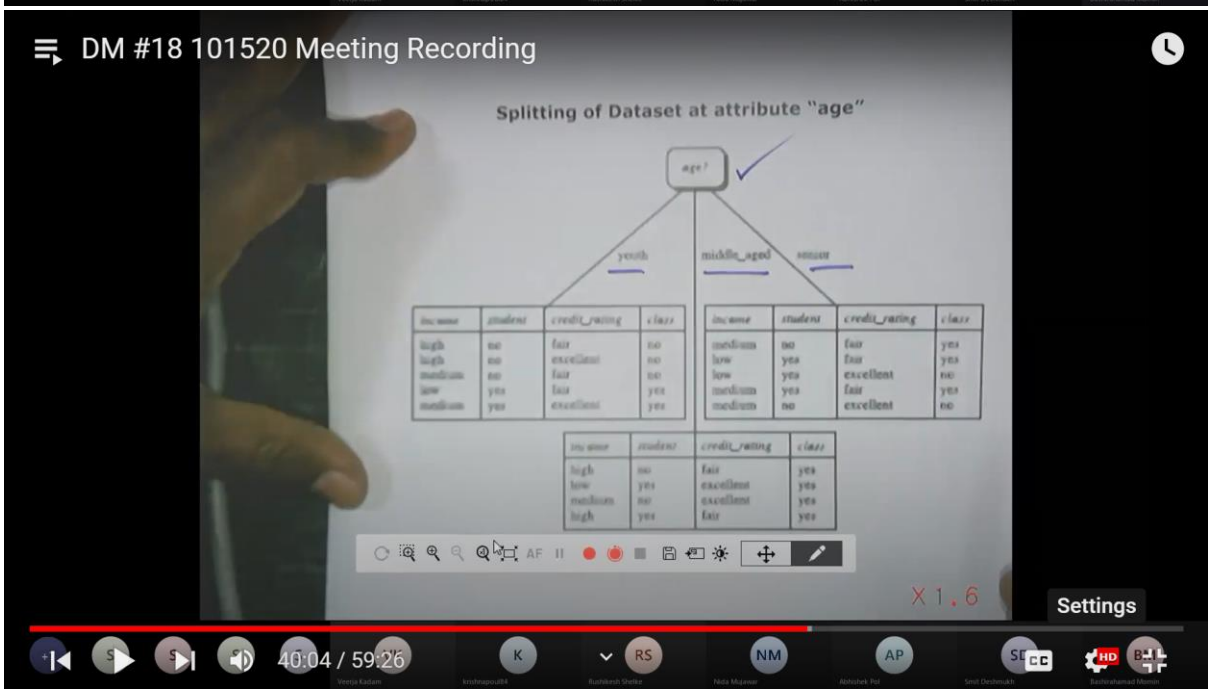

Decision Tree - OneNote
Press Esc to exit full screen
demo1
New Section 1
New Section 2
1] Constructing/building the model : Needs to train by training set
2] Evaluation / testing : test data set
Division : 80 20 or 70 30
1. Start at node N and present whole training data set
2. Apply the attribute selection strategy (Entropy, Gain Ratio or Gini Index) to attribute split
3. Divide the data set based on selected attribute.
4. Repeat step 2 , 3 till all data set covered or there is no attribute remain
+ve skewed data
-ve Skewed data
Python code
Dispersion : Spread of data
Variance and standard deviatio
Histogram
Data Pro-processing
Data Cleaning
Noisy Data :
Data Cleaning as a "Process"
Data Integration
Chi-Square Test
Correlation coefficient for nu
Covariance of Nu
Covariance of Numeric data
Tuple Duplication
Data transformation and Discr
Normalization
Discretization by Binning
Discretization by Histogram A
Classification - Basic Concept
Decision Tree
12:15 PM
9/22/2021

Lecture 18

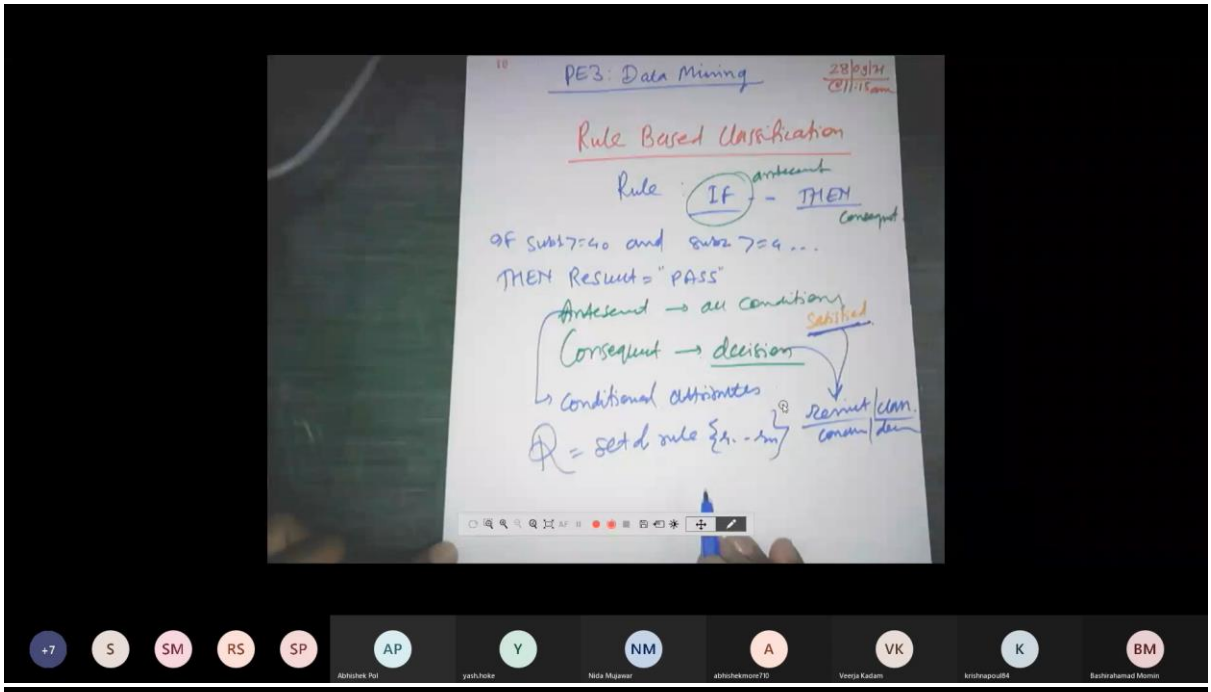

DM #18 101520 Meeting Recording
Training Data Set
RID | age | Income | student | credit_rating | Class: buys_computer
1 youth high no fair no
2 youth high no excellent no
3 middle_aged high no fair yes
4 senior medium no fair yes
5 senior low yes fair yes
6 senior low yes excellent no
7 middle_aged low yes excellent yes
8 youth medium no fair no
9 youth low yes fair yes
10 senior medium yes fair yes
11 youth medium yes excellent yes
12 middle_aged medium no excellent yes
13 middle_aged high yes fair yes
14 senior medium no excellent no
Dr. Bashirahamad F. Momin
Department of Computer Science & Engineering
4:14 / 59:26

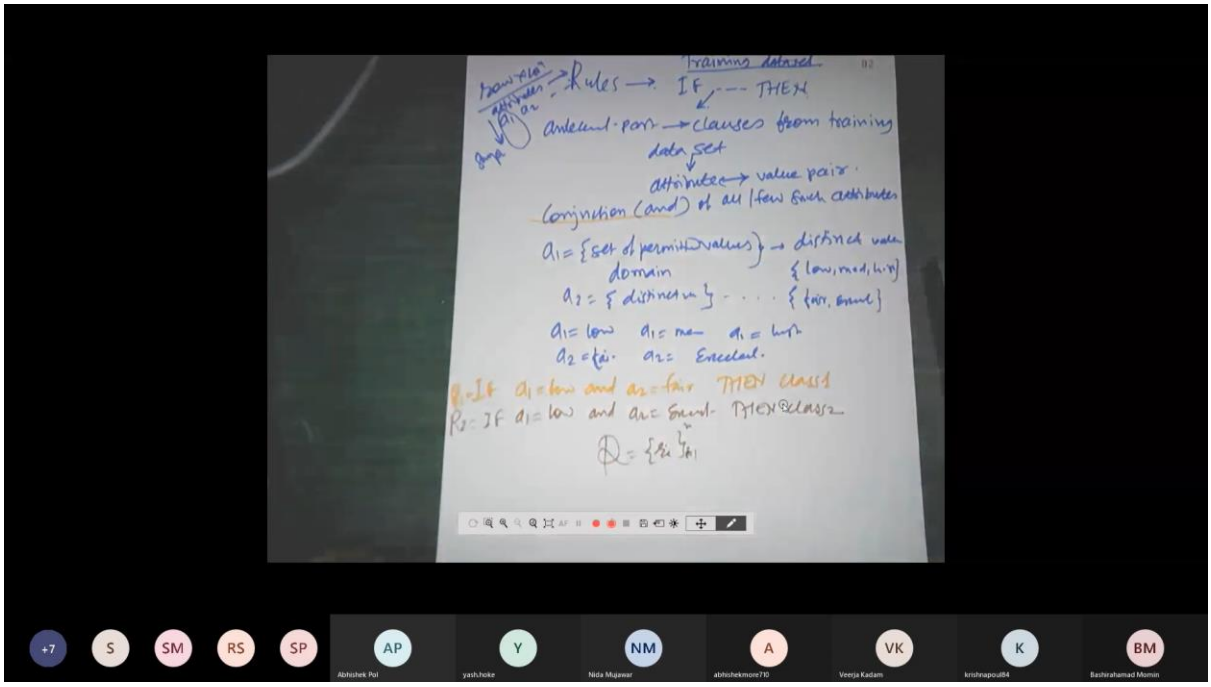

DM #18 101520 Meeting Recording
|D| = 14
Info(D) = $\sum_{i=1}^{m} p_i \log_2 p_i$
Info(D_youth) = $-\{p_1 \log_2 p_1 + p_2 \log_2 p_2\}$
$p_1 = \frac{|C_1, D_{age}|}{|D|}$ = No. of tuples for class C1 having youth / Total No. of tuples for youth value
$= \frac{2}{5}$
X 1.6
27:59 / 59:26


DM #18 101520 Meeting Recording
= 0.940 − 0.694 = 0.246
Compute similarity for other attributes
Gain (income) = 0.029
Gain (student) = 0.151
Gain (credit rating) = 0.048
Choose among the attribute having higher gain
∴ "Age" is split attribute
X 1.6
38:10 / 59:26

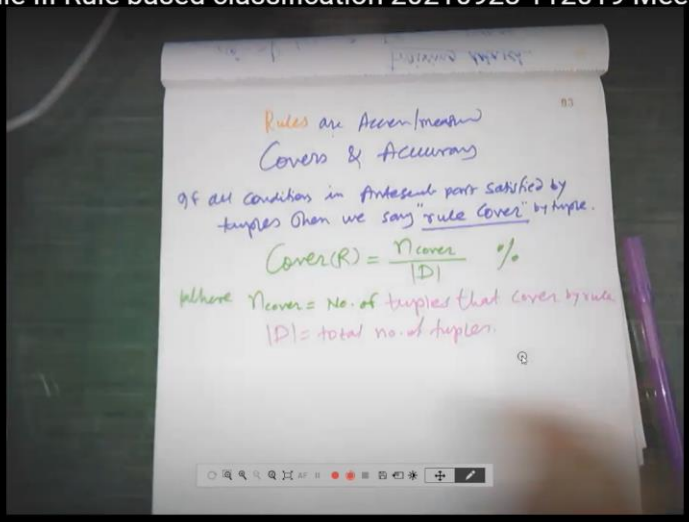

DM #18 101520 Meeting Recording
Splitting of Dataset at attribute "age"
age?
youth
middle_aged
senior
X 1.6
Settings
40:04 / 59:26

Lecture 19

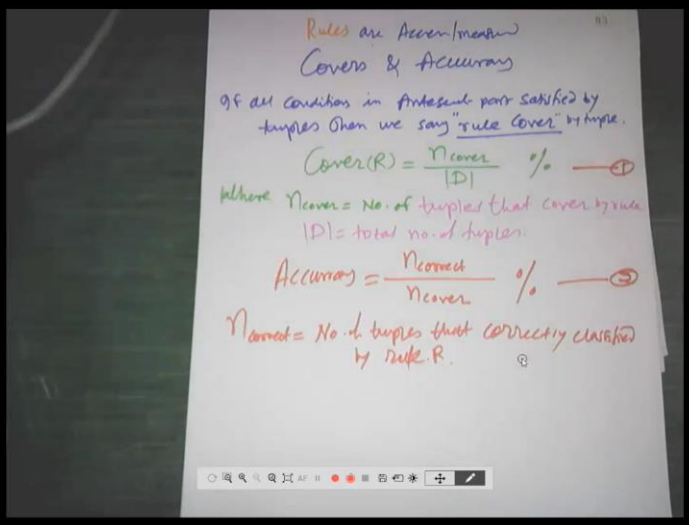

PE3: Data Mining
Rule Based Classification
Rule : IF - THEN
antecedent
Consequent
IF sub1 >= 40 and sub2 >= 40 ...
THEN Result = "PASS"
Antecedent → all conditions
Satisfied
Consequent → decision
Conditional attributes
R = set of rule {r1 ... rn}
result/class
consequent

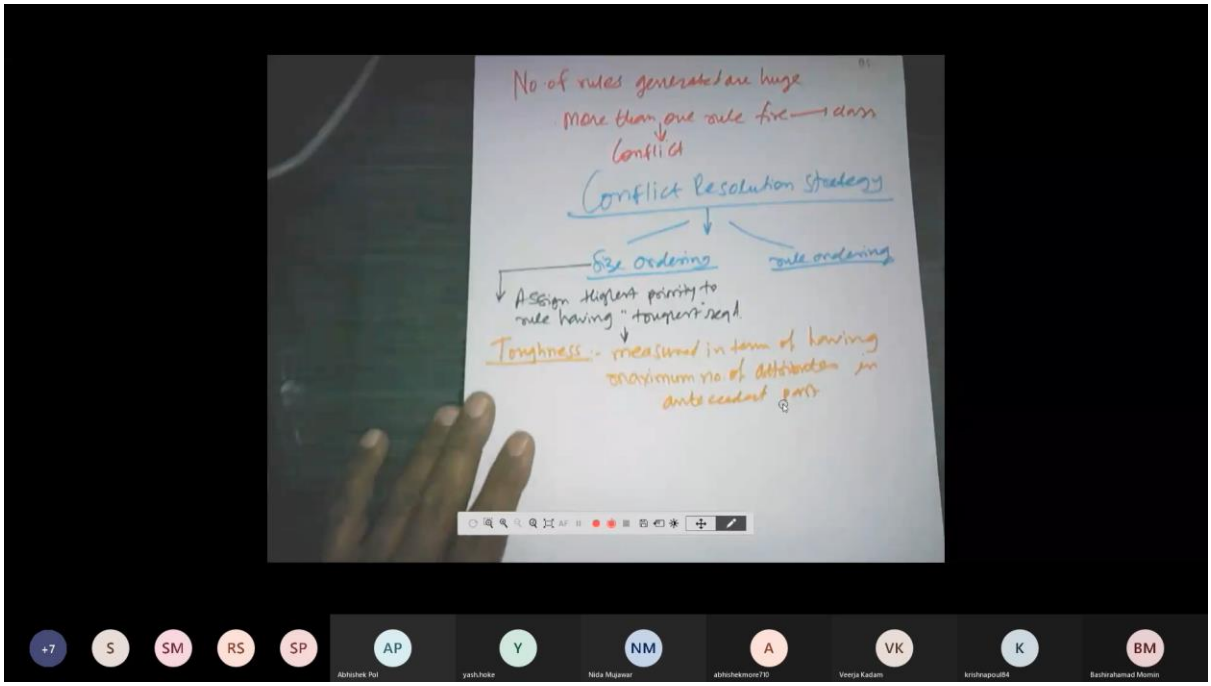

training dataset
Rules → IF --- THEN
antecedent part → clauses from training data set
attributes → value pair
Conjunction (and) of all few such attributes
a1 = {set of permitted values} → distinct value domain
{low, med, high}
a2 = {distinct values} ... {fair, Excel}
a1 = low a1 = med a1 = high
a2 = fair a2 = Excellent
R1 = If a1 = low and a2 = fair THEN class1
R2 = IF a1 = low and a2 = Excel THEN class2
R = {r1 ... rn}

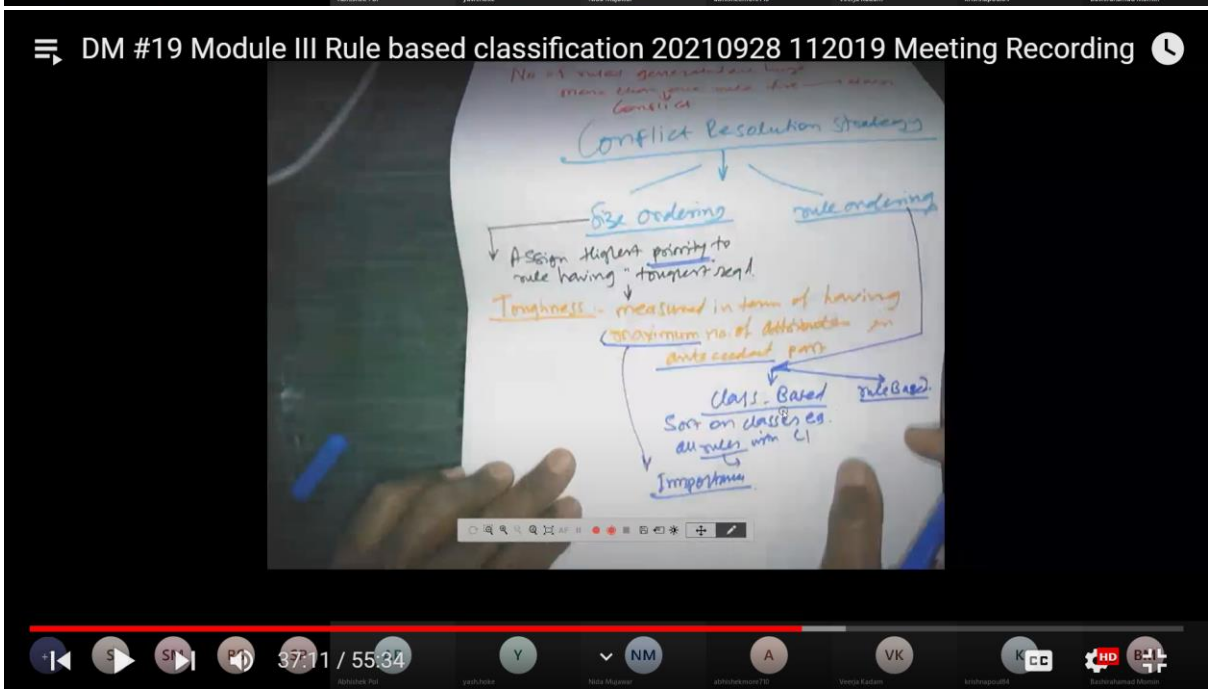
DM #19 Module III Rule based classification 20210928 112019 Meeting Recording

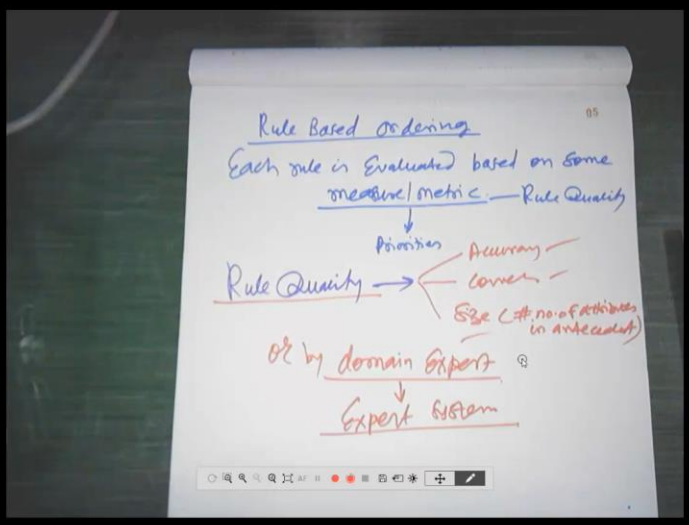


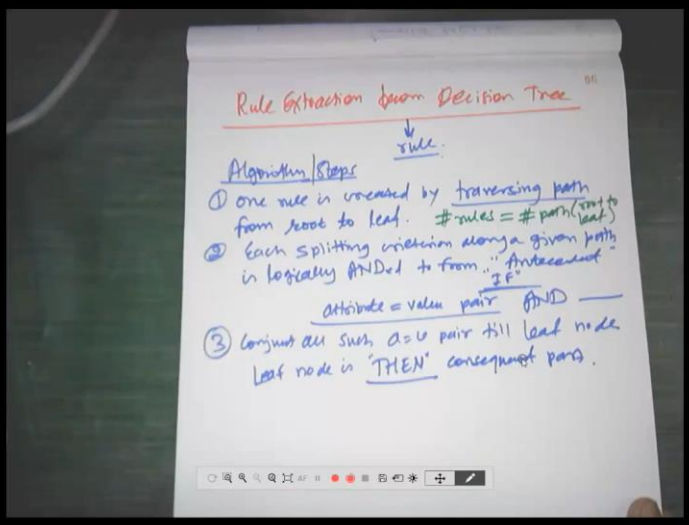
28:45 / 55:34



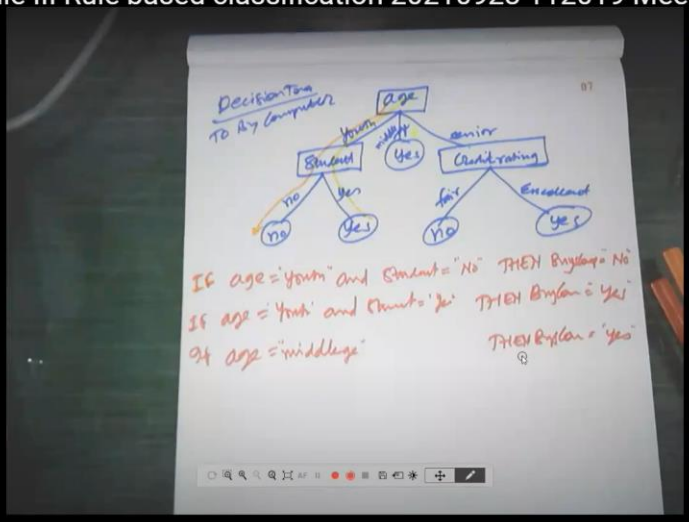

No of rules generated are huge
more than one rule fire → class
Conflict
Conflict Resolution Strategy
Size ordering
rule ordering
Assign highest priority to
rule having "toughest" lhs.
Toughness - measured in term of having
maximum no. of attributes in
antecedent part.


DM #19 Module III Rule based classification 20210928 112019 Meeting Recording
Conflict Resolution Strategy
Size ordering
rule ordering
Assign highest priority to
rule having "toughest" lhs.
Toughness - measured in term of having
maximum no. of attributes in
antecedent part.
Class - Based
Rule Based
Sort on classes eg.
all rules with C1
Importance
37:11 / 55:34



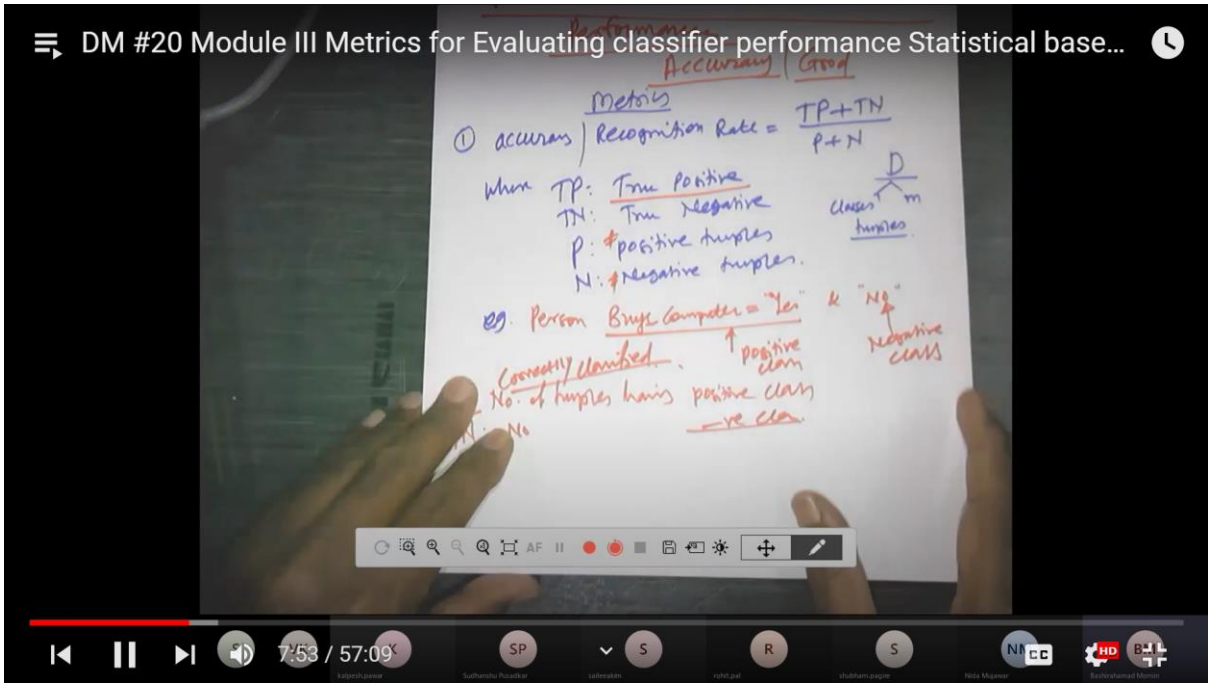


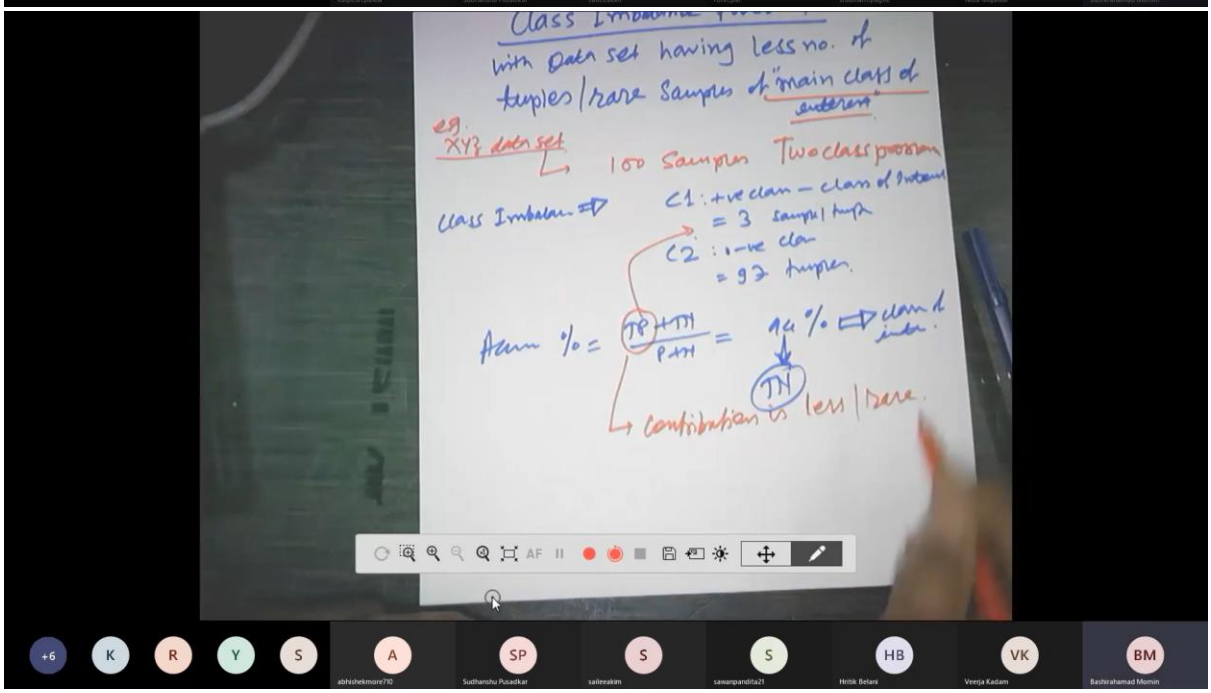
DM #19 Module III Rule based classification 20210928 112019 Meeting Recording

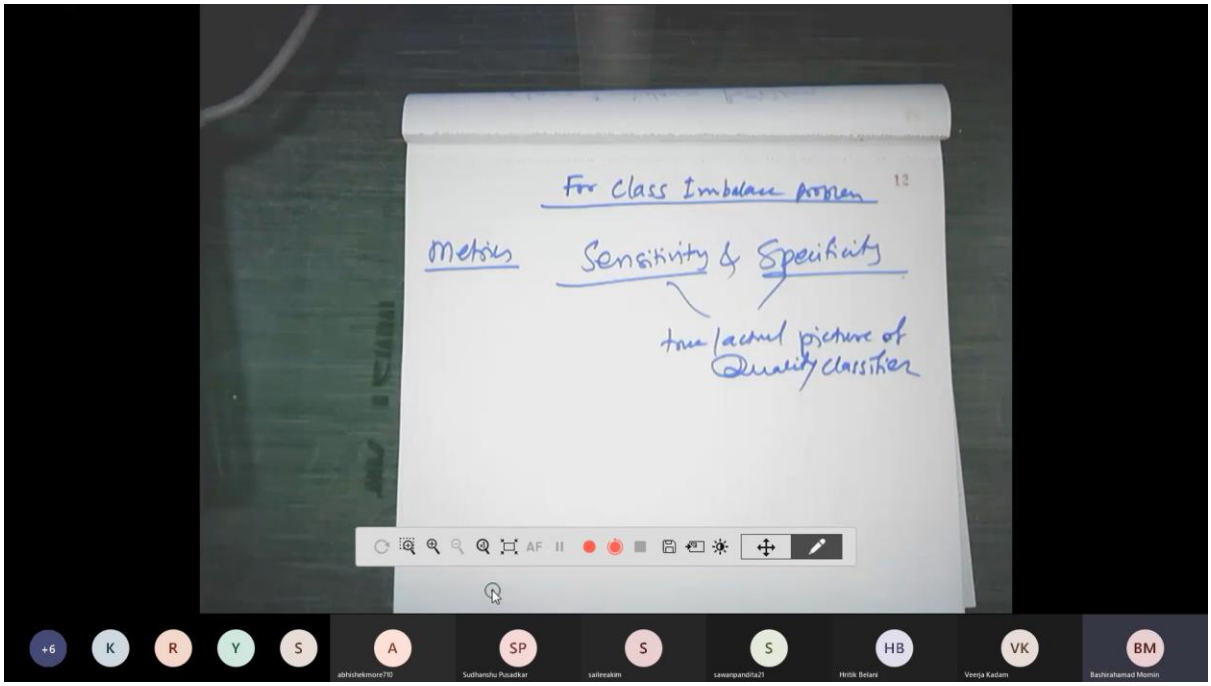


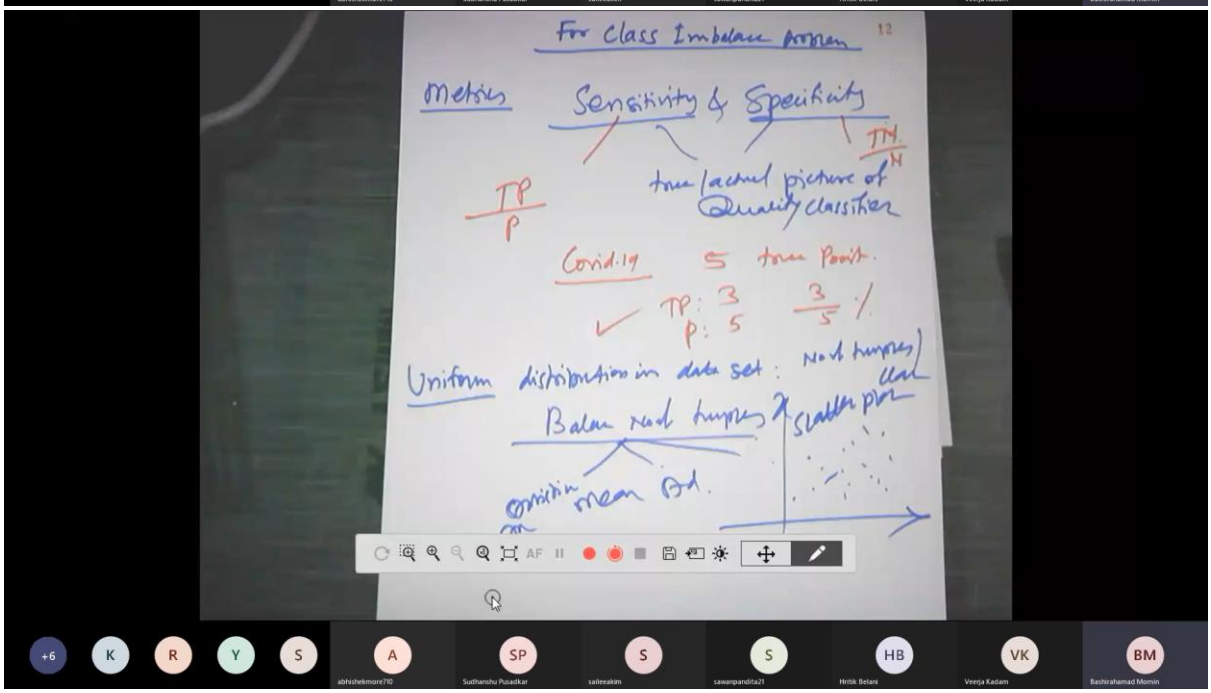
54:40 / 55:34

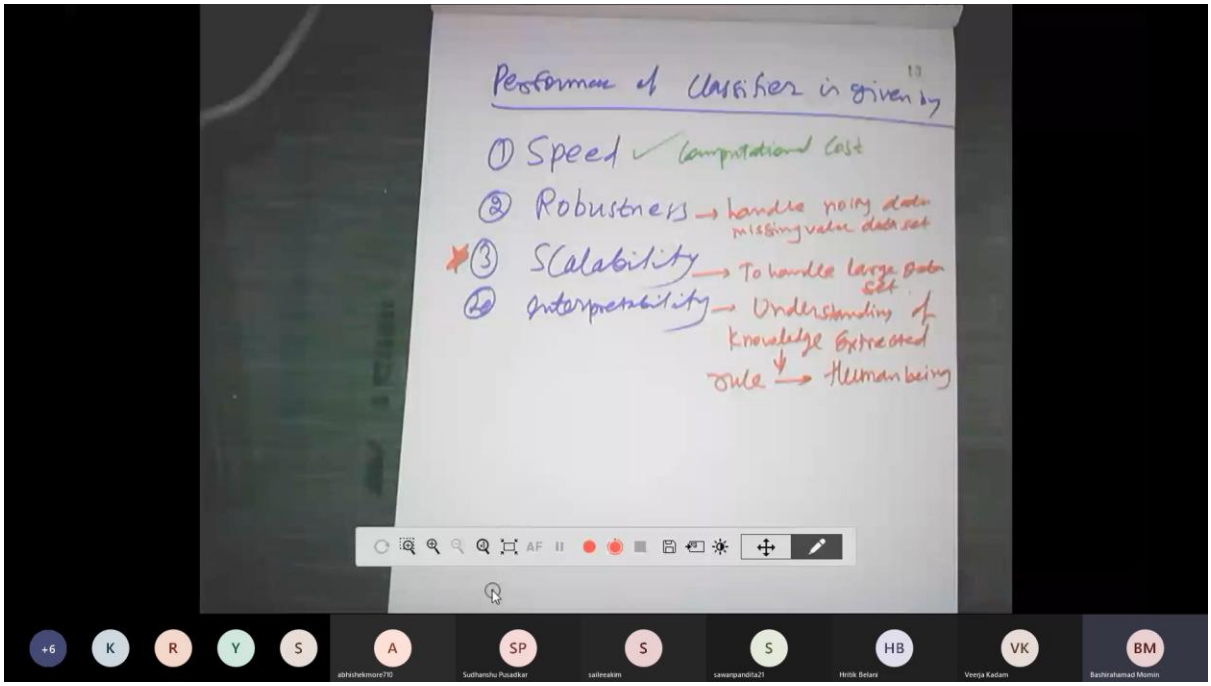
Lecture 20


DM #20 Module III Metrics for Evaluating classifier performance Statistical base...
Metrics
accuracy / Recognition Rate = TP+TN / P+N
where TP: True Positive
TN: True Negative
P: # positive tuples
N: # negative tuples
eg. Person Buys computer = "Yes" & "No"
positive class
negative class
Correctly classified
No. of tuples having positive class
-ve class
7:53 / 57:09

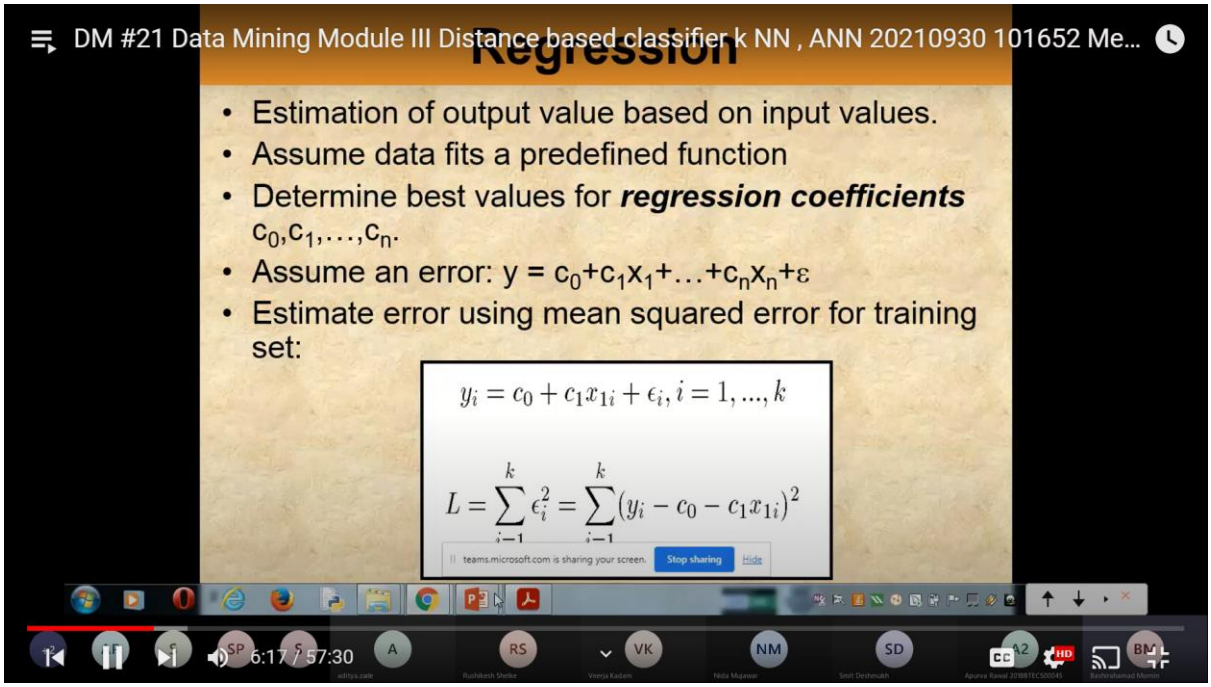

with Data set having less no. of tuples / rare samples of "main class of interest"
eg.
XYZ data set
100 Samples Two class problem
Class Imbalance
C1: +ve class = class of interest
= 3 sample tuple
C2: -ve class
= 97 tuples
Accuracy % = TP+TN / P+N = 14 % ⇒ class imbalance
TN
Contribution is less / rare.

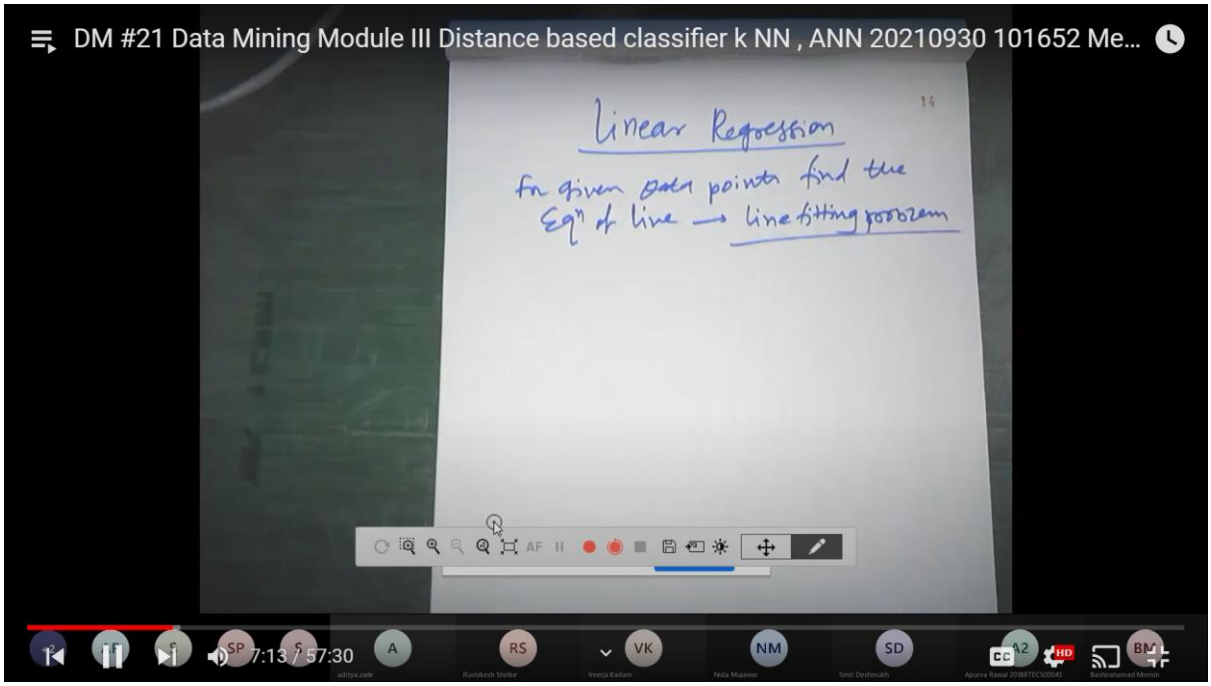

For Class Imbalance problem
Metrics
Sensitivity & Specificity
true/actual picture of Quality classifier

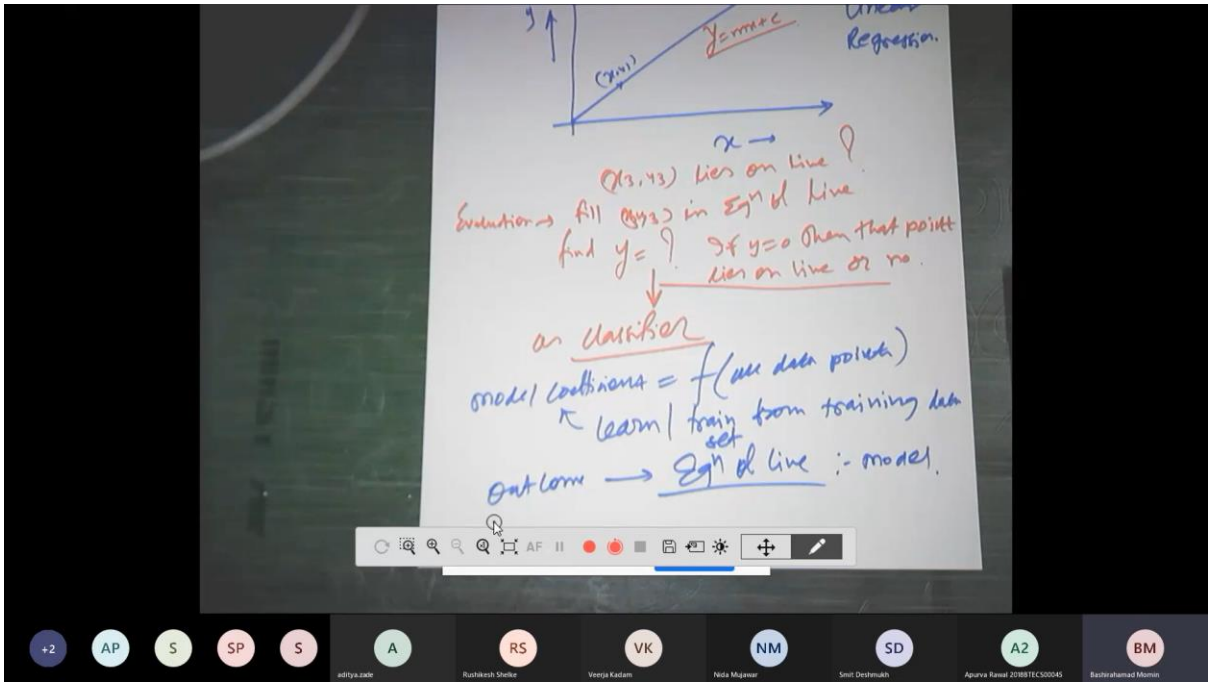

For Class Imbalance problem
Metrics
Sensitivity & Specificity
TN
TP
P
true/actual picture of Quality classifier
Covid-19
5
true Positive
TP: 3
P: 5
3/5 %
Uniform distribution in data set:
Balance real tuples
mean
Std.
scatter plot

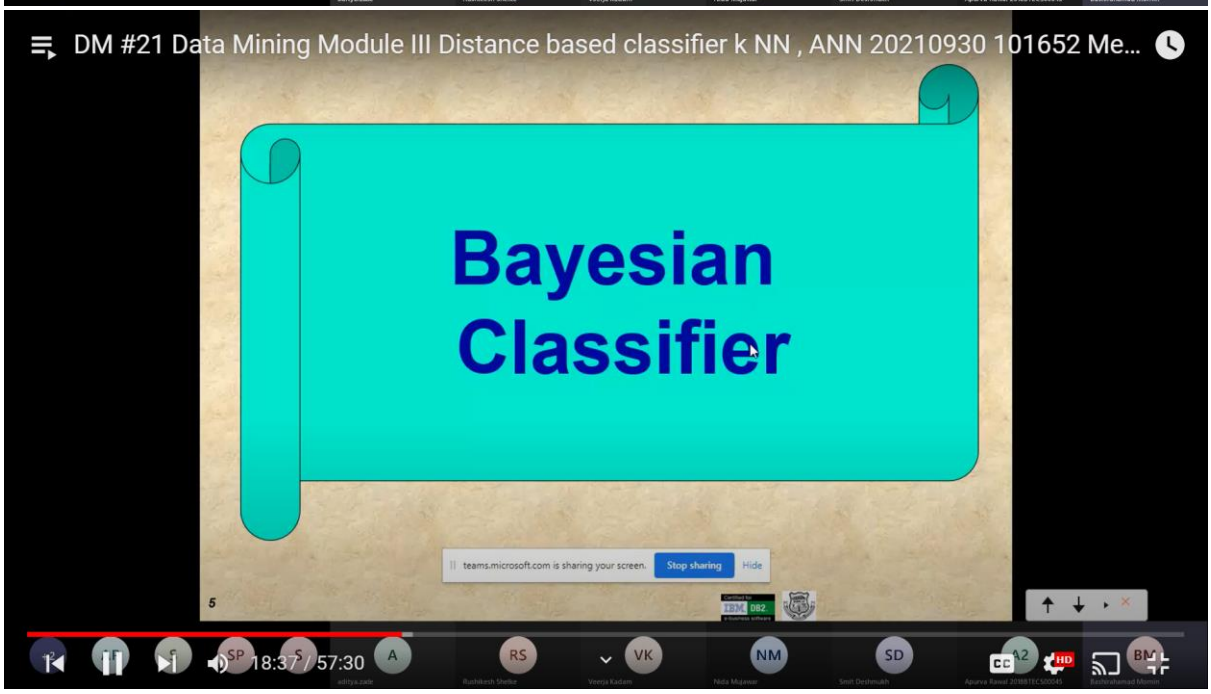

Performance of classifier is given by
1) Speed → computational cost
2) Robustness → handle noisy data / missing value data set
3) Scalability → To handle large data set
4) Interpretability → Understanding of knowledge extracted
rule → Human being
+6
K
R
Y
S
A
abhishekmore710
SP
Sudhanshu Pusadkar
S
saileeakim
S
sawanpandita21
HB
Hritik Belani
VK
Veerja Kadam
BM
Bashirahamad Momin

Lecture 21


DM #21 Data Mining Module III Distance based classifier k NN , ANN 20210930 101652 Me...
Regression
• Estimation of output value based on input values.
• Assume data fits a predefined function
• Determine best values for regression coefficients $c_0,c_1,\ldots,c_n$.
• Assume an error: $y = c_0+c_1x_1+\ldots+c_nx_n+\varepsilon$
• Estimate error using mean squared error for training set:
$y_i = c_0 + c_1x_{1i} + \epsilon_i, i = 1, \ldots, k$
$L = \sum_{i-1}^{k} \epsilon_i^2 = \sum_{i-1}^{k} (y_i - c_0 - c_1x_{1i})^2$
teams.microsoft.com is sharing your screen.
Stop sharing
Hide
6:17 / 57:30


DM #21 Data Mining Module III Distance based classifier k NN , ANN 20210930 101652 Me...
Linear Regression
for given data points find the Eqn of line → line fitting problem
7:13 / 57:30


y = mx + c
Regression.
(x1, y1)
x →
Q(3, 43) lies on line ?
Evaluation → fill (3, 43) in Eqn of line
find y = ? If y=0 then that point lies on line or no.
an classifier
model coefficients = f(our data points)
learn / train from training data set
Outcome → Eqn of line :- model.


DM #21 Data Mining Module III Distance based classifier k NN , ANN 20210930 101652 Me...
Bayesian Classifier
teams.microsoft.com is sharing your screen.
Stop sharing
Hide
5
18:37 / 57:30

Bayesian Classification: Why?

- A statistical classifier: performs *probabilistic prediction, i.e.*, predicts class membership probabilities
- Foundation: Based on Bayes' Theorem.
- Performance: A simple Bayesian classifier, *naïve Bayesian classifier*, has comparable performance with decision tree and selected neural network classifiers
- Incremental: Each training example can incrementally increase/decrease the probability that a hypothesis is correct — prior knowledge can be combined with observed data
- Standard: Even when Bayesian methods are computationally intractable, they can provide a standard of optimal decision making against which other methods can be measured

Lecture 23

Classification Using Distance

- Place items in class to which they are "closest".
- Must determine distance between an item and a class.
- Classes represented by
  - ***Centroid:*** Central value.
  - ***Medoid:*** Representative point.
  - Individual points
- Algorithm: *k*-NN

16

k-NN Algorithm

```
Input:
   D       //Training data
   K       //Number of neighbors
   t       //Input tuple to classify
Output:
   c       //Class to which t is assigned
KNN Algorithm:
           //Algorithm to classify tuple using KNN
   N = ∅;
                //Find set of neighbors, N, for t
   foreach d ∈ D do
      if | N |≤ K then
         N = N ∪ d;
      else
         if ∃ u ∈ N such that sim(t, u) ≥ sim(t, d) then
            begin
               N = N − u;
               N = N ∪ d;
            end
            //Find class for classification
   c = class to which the most u ∈ N are classified;
```

Nearest Neighbor (NN) Classifier

- Based on learning by analogy
- Better for data set which are :
  - Non-parametric in nature
  - Multivariate dependencies
- Closeness measure
  - Euclidean Distance (ED)
  - Cosine Coefficient (CC)
  - Pearson Coefficient etc. …

Nearest Neighbor Classifier (*1-NN*)

Unknown pattern

$U_1, \cdots\cdots, Un$

- Find the nearest* known sample
- Assign class label of nearest sample to unknown pattern

Known (training) Samples

Samples		Class Label
V_1	$X_1, \cdots\cdots, X_n$	C_1
V_m		C_k

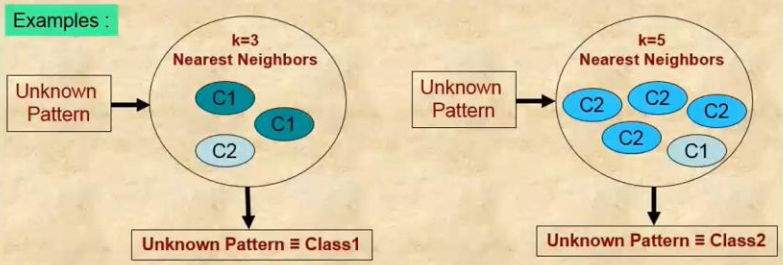
*Distance metric : Euclidean Distance

$$ED(U,X) = \sqrt{\sum_{i=1}^{n} (U_i - X_i)^2}$$

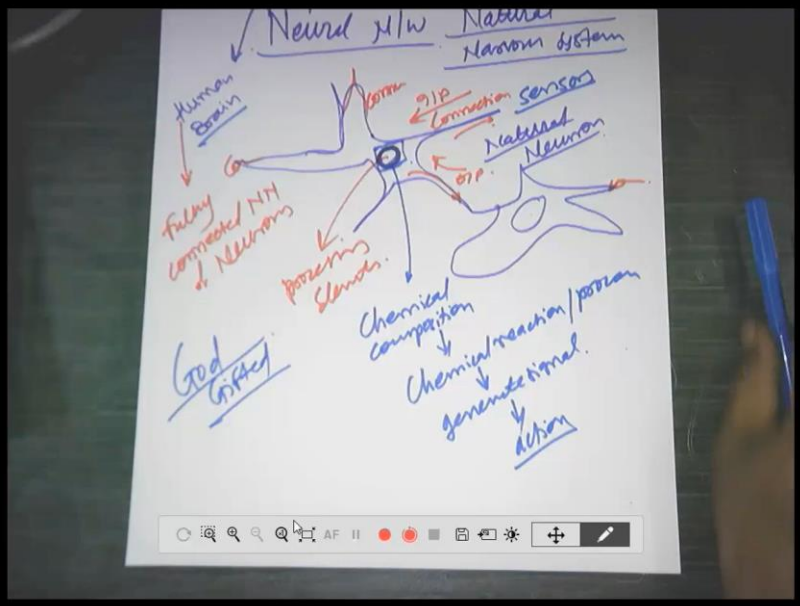
k-Nearest Neighbor (*k*-NN) Classifier

- Choose “k” >=3 (odd number)
- Find “k” nearest neighbors using Euclidean Distance.
- Majority voting : select class label assign to maximum of “k”-neighbors.
- Assign selected class label to unknown pattern.

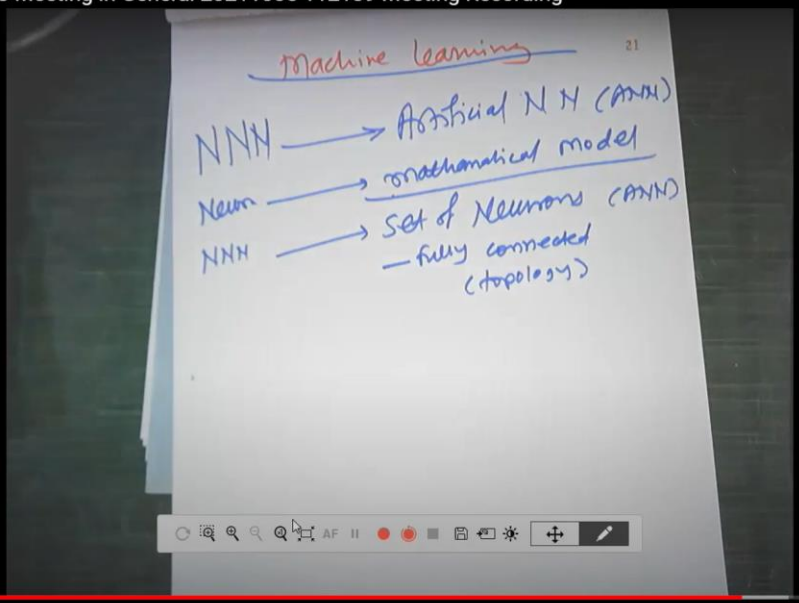
Examples :



20

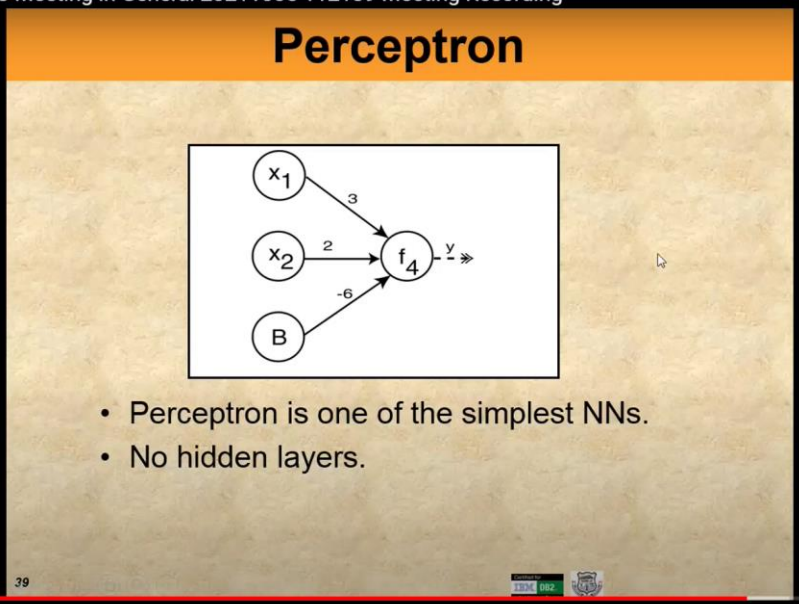


DM #23 Meeting in General 20211005 112139 Meeting Recording

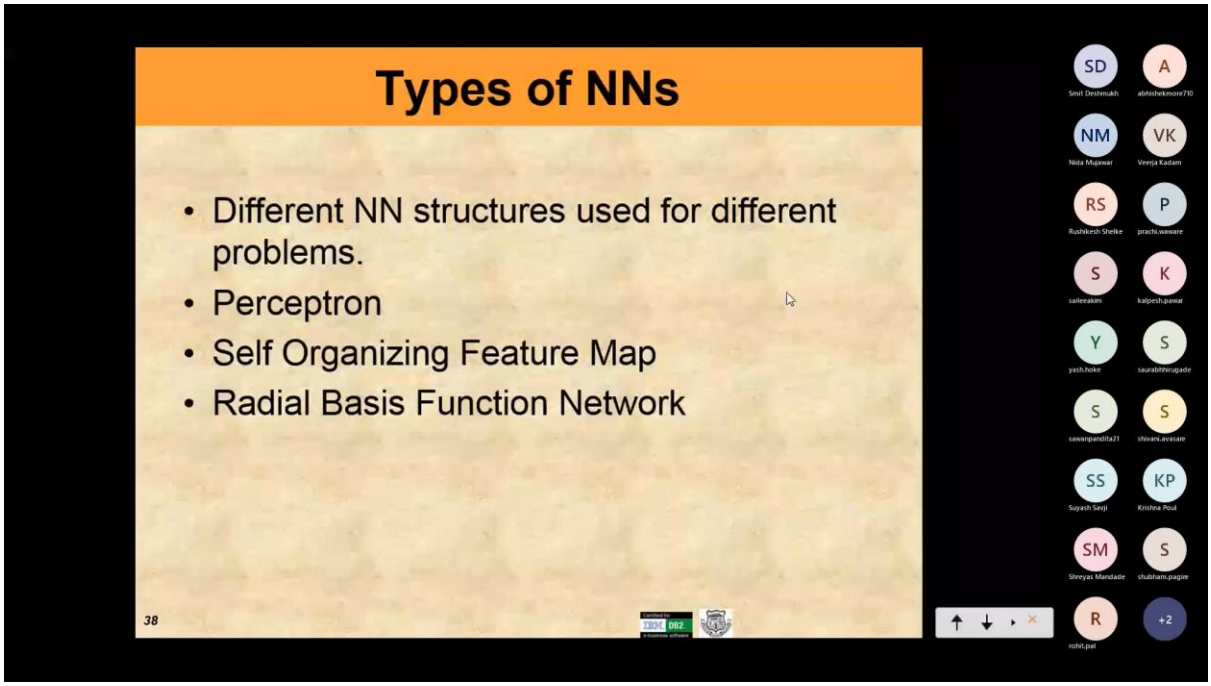


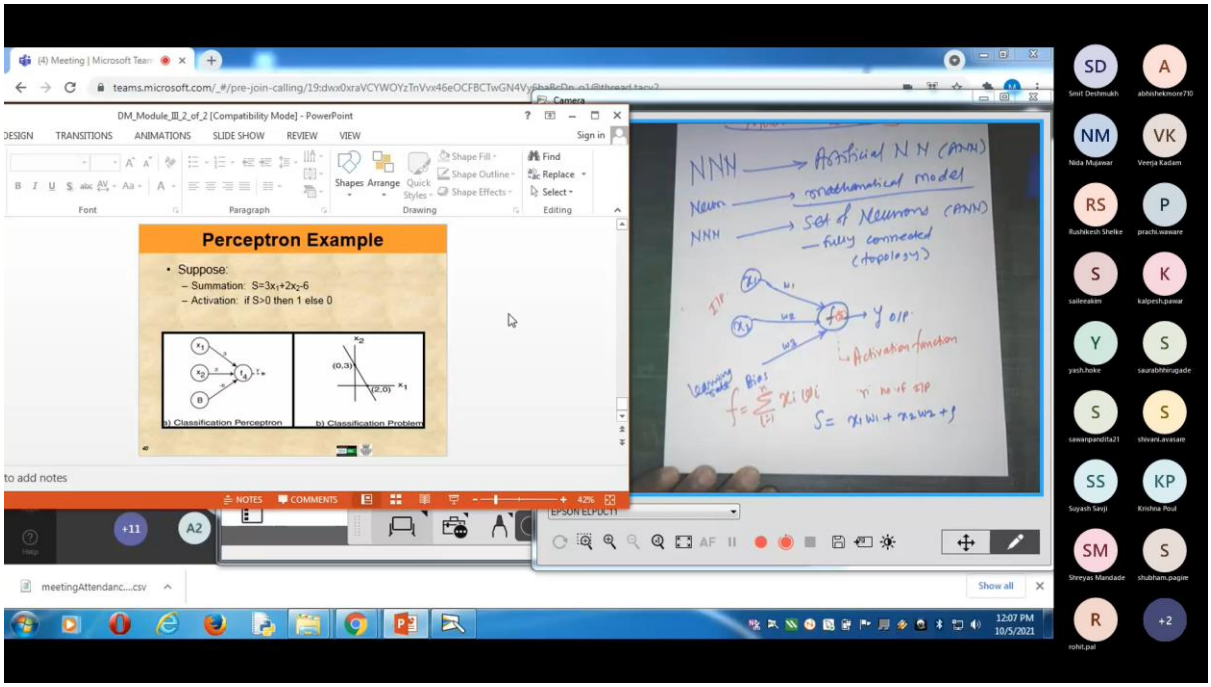
39:08 / 53:54

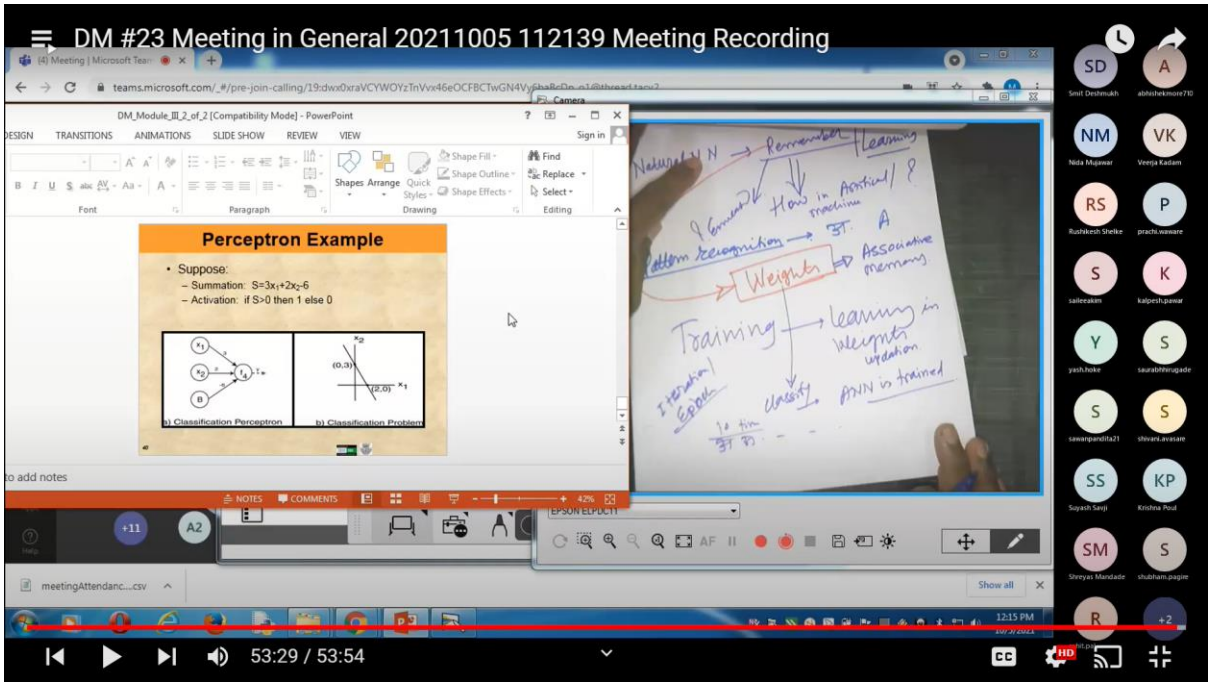
DM #23 Meeting in General 20211005 112139 Meeting Recording

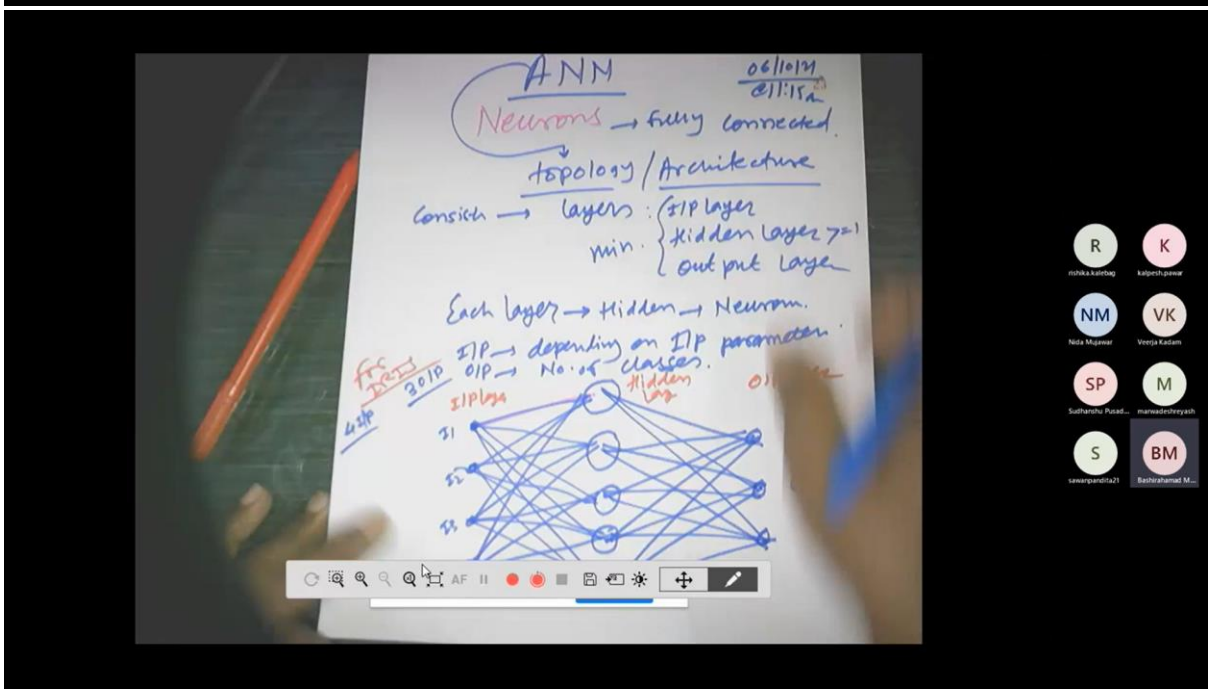


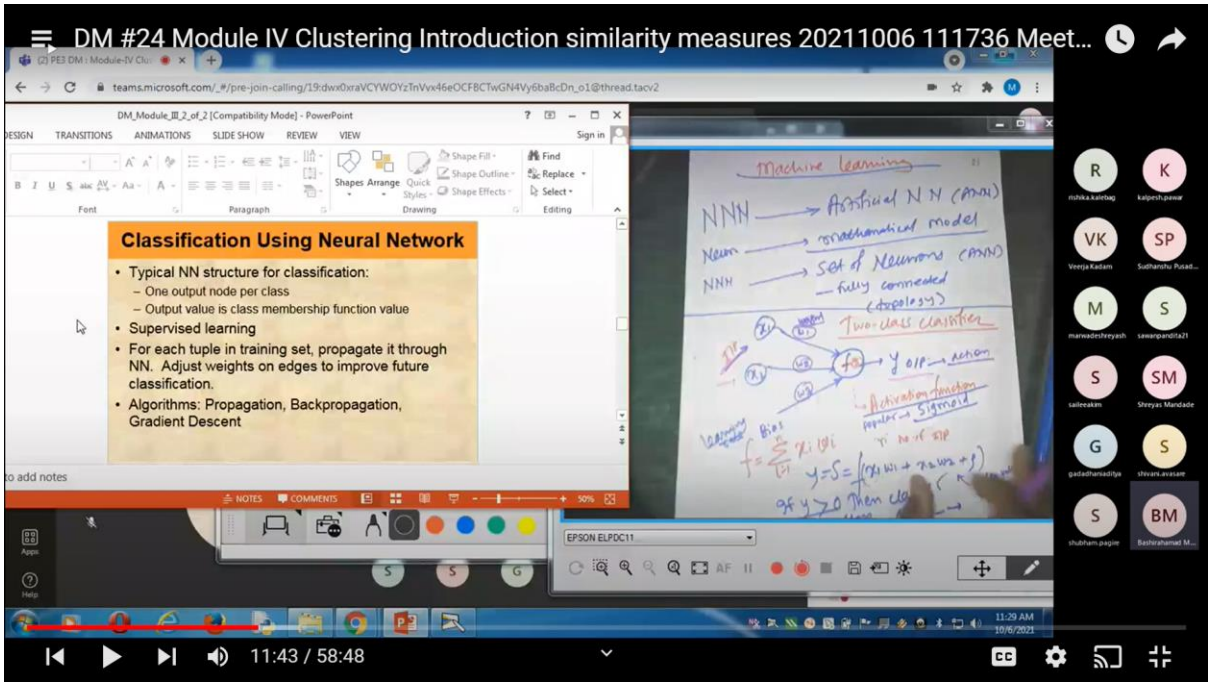
39:22 / 53:54


Types of NNs
• Different NN structures used for different problems.
• Perceptron
• Self Organizing Feature Map
• Radial Basis Function Network
38


Perceptron Example
• Suppose:
– Summation: S=3x1+2x2-6
– Activation: if S>0 then 1 else 0
a) Classification Perceptron
b) Classification Problem
NNN → Artificial N N (ANN)
Neuron → mathematical model
NNN → Set of Neurons (ANN)
fully connected
(topology)
Activation function


DM #23 Meeting in General 20211005 112139 Meeting Recording
Perceptron Example
Suppose:
Summation: S=3x1+2x2-6
Activation: if S>0 then 1 else 0
Weights
Training
ANN is trained
53:29 / 53:54


ANN
Neurons → fully connected
topology / Architecture
Consists → layers : I/P layer, Hidden layer >=1, Output layer
Each layer → Hidden → Neurons
I/P → depending on I/P parameters
O/P → No. of classes
Hidden Layer



NN Issues

- Number of source nodes
- Number of hidden layers
- Training data
- Number of sinks
- Interconnections
- Weights
- Activation Functions
- Learning Technique
- When to stop learning (convergence)

NN Learning

- Adjust weights to perform better with the associated test data.
- ***Supervised:*** Use feedback from knowledge of correct classification.
- ***Unsupervised:*** No knowledge of correct classification needed.

NN Supervised Learning

```
Input:
   N     //Starting Neural Network
   X     //Input tuple from Training Set
   D     //Output tuple desired
Output:
   N     //Improved Neural Network
SupLearn Algorithm:
            //Simplistic algorithm to illustrate approach to NN learning
   Propagate X through N producing output Y;
   Calculate error by comparing D to Y;
   Update weights on arcs in N to reduce error;
```

Supervised Learning

- Possible error values assuming output from node i is y_i but should be d_i:

$$|y_i - d_i|$$

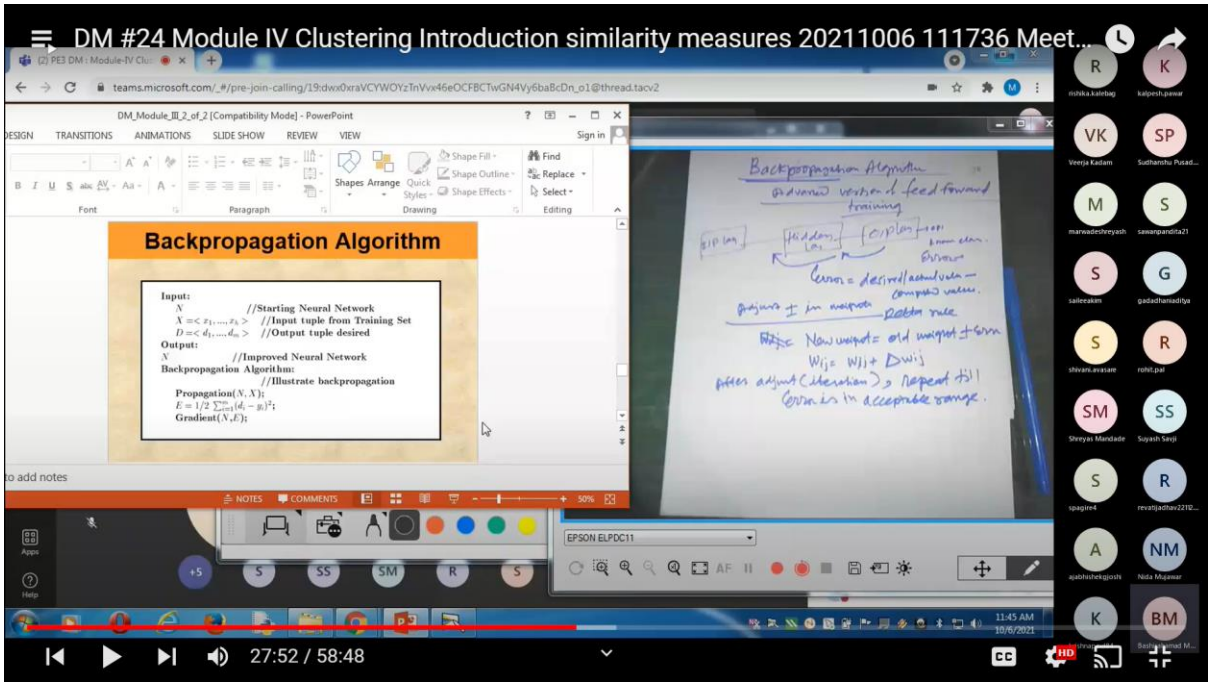
$$\frac{(y_i - d_i)^2}{2}$$

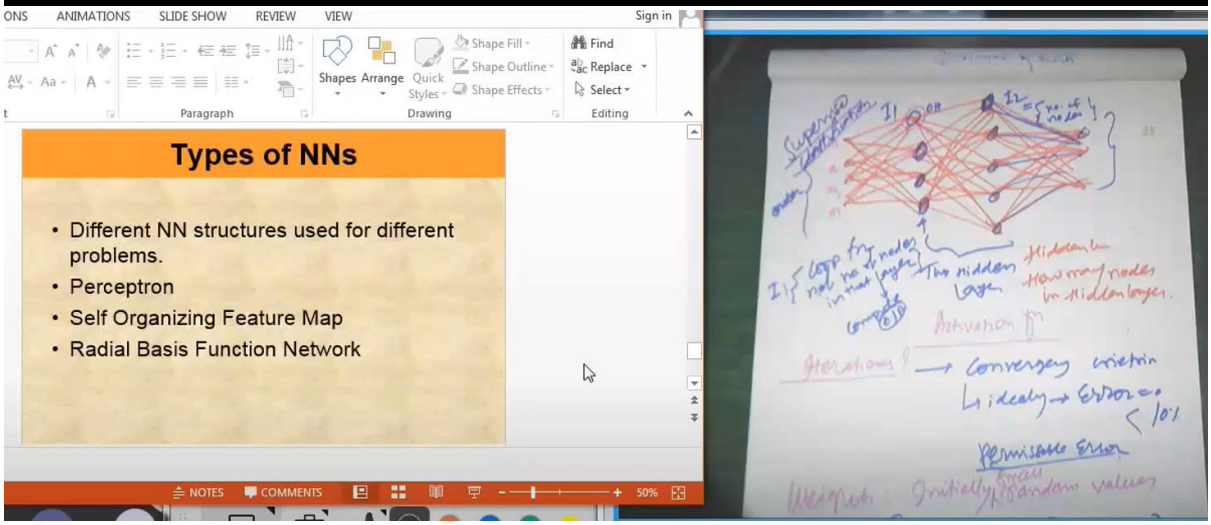
$$\sum_{i=1}^{m} \frac{(y_i - d_i)^2}{m}$$

- Change weights on arcs based on estimated error

NN Backpropagation

- Propagate changes to weights backward from output layer to input layer.
- ***Delta Rule:*** $\triangle w_{ij} = c\, x_{ij} (d_j - y_j)$
- ***Gradient Descent:*** technique to modify the weights in the graph.


DM #24 Module IV Clustering Introduction similarity measures 20211006 111736 Meet...
Backpropagation Algorithm
Input:
N //Starting Neural Network
X =< x1, ..., xk > //Input tuple from Training Set
D =< d1, ..., dm > //Output tuple desired
Output:
N //Improved Neural Network
Backpropagation Algorithm:
//Illustrate backpropagation
Propagation(N, X);
E = 1/2 Σ(di − yi)2;
Gradient(N,E);
27:52 / 58:48


Types of NNs
• Different NN structures used for different problems.
• Perceptron
• Self Organizing Feature Map
• Radial Basis Function Network

Lecture 24

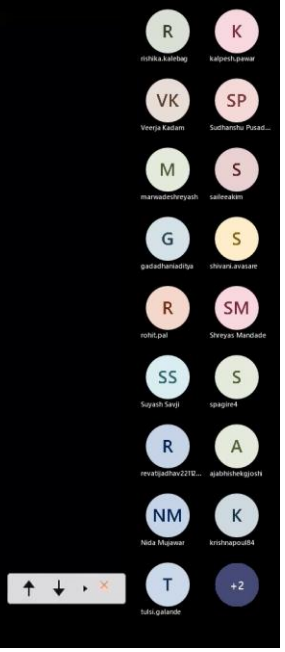
Clustering - Introduction

- **Unlike classification, the class label of each training sample is not known : *Unsupervised Learning.***
- **It is the process of grouping a set of training samples or examples into similar classes.**
- **It groups similar objects based on their distance, connectivity, relative density or some specific characteristics.**
- **A *cluster* is therefore a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters**
- **Different algorithms / techniques :**
  - Hierarchical Algorithms
  - Partitional Algorithms
  - Clustering Large Databases

Clustering Problem

- Given a database $D=\{t_1,t_2,\ldots,t_n\}$ of tuples and an integer value k, the ***Clustering Problem*** is to define a mapping $f:D\rightarrow\{1,..,k\}$ where each t_i is assigned to one cluster K_j, $1<=j<=k$.
- A ***Cluster***, K_j, contains precisely those tuples mapped to it.
- Unlike classification problem, clusters are not known a priori.

4

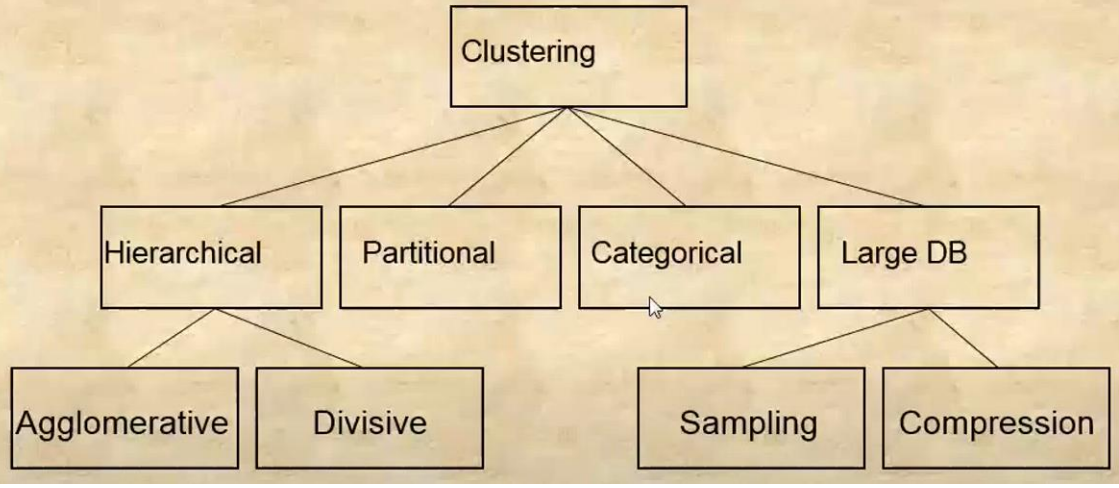


Types of Clustering

- ***Hierarchical*** – Nested set of clusters created.
- ***Partitional*** – One set of clusters created.
- ***Incremental*** – Each element handled one at a time.
- ***Simultaneous*** – All elements handled together.
- ***Overlapping/Non-overlapping***

5

Clustering Approaches



Clustering Examples

- ***Segment*** customer database based on similar buying patterns.
- Group houses in a town into neighborhoods based on similar features.
- Identify new plant species
- Identify similar Web usage patterns

Clustering Issues

- Outlier handling
- Dynamic data
- Interpreting results
- Evaluating results
- Number of clusters
- Data to be used
- Scalability

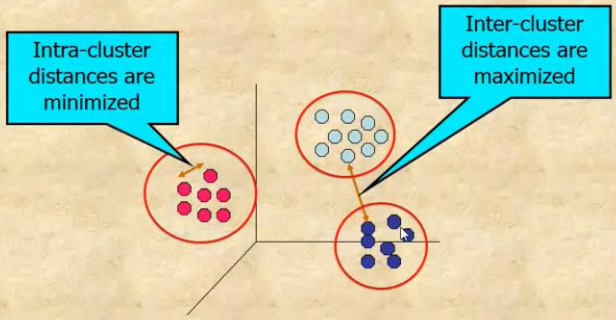
Clustering Example

Income	Age	Children	Marital Status	Education
$25,000	35	3	Single	High School
$15,000	25	1	Married	High School
$20,000	40	0	Single	High School
$30,000	20	0	Divorced	High School
$20,000	25	3	Divorced	College
$70,000	60	0	Married	College
$90,000	30	0	Married	Graduate School
$200,000	45	5	Married	Graduate School
$100,000	50	2	Divorced	College

9

Similarity and Distance Measures

Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups



11

Dissimilarity / Similarity metric

Similarity is expressed in terms of a distance function, typically metric : *d(i, j)*

- Distances are normally used to measure the similarity or dissimilarity between two data objects
- Some popular ones include: *Minkowski distance*:

$$d(i,j)=\sqrt[q]{(|x_{i1}-x_{j1}|^q+|x_{i2}-x_{j2}|^q+...+|x_{ip}-x_{jp}|^q)}$$

where $i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ are two p-dimensional data objects, and q is a positive integer

- If $q = 1$, d is Manhattan distance

$$d(i,j)=|x_{i1}-x_{j1}|+|x_{i2}-x_{j2}|+...+|x_{ip}-x_{jp}|$$

Cont... Dissimilarity / Similarity metric

- *If* $q = 2$, d is **Euclidean distance**:

$$d(i,j)=\sqrt{(|x_{i1}-x_{j1}|^2+|x_{i2}-x_{j2}|^2+...+|x_{ip}-x_{jp}|^2)}$$

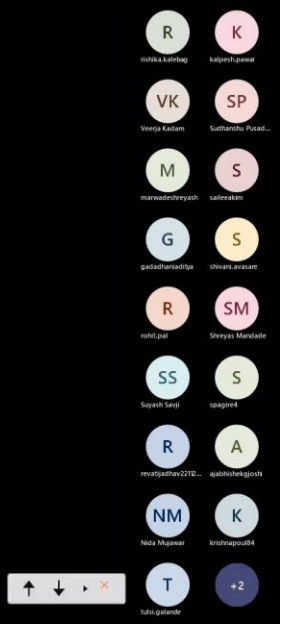
  - Properties
    - $d(i,j) \geq 0$
    - $d(i,i) = 0$
    - $d(i,j) = d(j,i)$
    - $d(i,j) \leq d(i,k) + d(k,j)$
- Also, one can use weighted distance, parametric Pearson product moment correlation, or other dissimilarity measures

13

The Distance between Clusters

- **Single link**: smallest distance between an element in one cluster and an element in the other, i.e., $dis(K_i, K_j) = min(t_{ip}, t_{jq})$
- **Complete link**: largest distance between an element in one cluster and an element in the other, i.e., $dis(K_i, K_j) = max(t_{ip}, t_{jq})$
- **Average**: avg distance between an element in one cluster and an element in the other, i.e., $dis(K_i, K_j) = avg(t_{ip}, t_{jq})$
- **Centroid**: distance between the centroids of two clusters, i.e., $dis(K_i, K_j) = dis(C_i, C_j)$
- **Medoid**: distance between the medoids of two clusters, i.e., $dis(K_i, K_j) = dis(M_i, M_j)$
  - **Medoid: one chosen, centrally located object in the cluster**

14



Centroid, Radius and Diameter of a Cluster

- **Centroid**: the "middle" of a cluster

$$C_m = \frac{\sum_{i=1}^{N} (t_{ip})}{N}$$

- **Radius**: square root of average distance from any point of the cluster to its centroid

$$R_m = \sqrt{\frac{\sum_{i=1}^{N} (t_{ip} - c_m)^2}{N}}$$

- **Diameter**: square root of average mean squared distance between all pairs of points in the cluster

$$D_m = \sqrt{\frac{\sum_{i=1}^{N} \sum_{i=1}^{N} (t_{ip} - t_{iq})^2}{N(N-1)}}$$

15

Clustering

Centroid

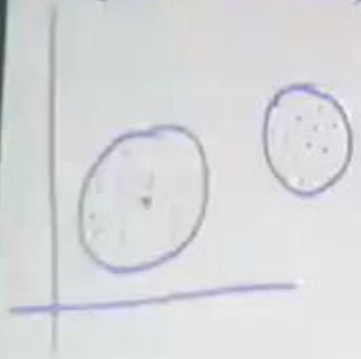
medoid

Clusters → mathematically

often computed scaler value

Center, 'middle' Need not to be physical point

most centrally located PHYSICAL POINT



Partitioning Algorithms: Basic Concept

- **Partitioning method**: Construct a partition of a database ***D*** of ***n*** objects into a set of ***k*** clusters, s.t., min sum of squared distance

$$\Sigma_{m=1}^{k}\Sigma_{t_{mi}\in Km}(C_m - t_{mi})^2$$

- Given a *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion
  - Global optimal: exhaustively enumerate all partitions
  - Heuristic methods: *k-means* and *k-medoids* algorithms
  - ***k-means*** (MacQueen'67): Each cluster is represented by the center of the cluster
  - ***k-medoids*** or **PAM (Partition Around Medoids)** (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

k-Means

- Initial set of clusters randomly chosen.
- Iteratively, items are moved among sets of clusters until the desired set is reached.
- High degree of similarity among elements in a cluster is obtained.
- Given a cluster $K_i=\{t_{i1},t_{i2},\ldots,t_{im}\}$, the ***cluster mean*** is $m_i = (1/m)(t_{i1} + \ldots + t_{im})$

Partitioning Clustering

Distance metric

↓

Similarity / Closeness / nearest.

K-mean

↓

Center

↓

Centroid

(avg) $\frac{1}{n}\sum_{i=1}^{n} x_i$

K-medoid

↓

center

↓

medoid

↓

physically existing located point.

K-means

1. Randomly choose k-point as k-clusters where k given by user
2. merge/assign new points closest to k
3. Recompute centeroid.
4. Repeat step 2 & 3 till convergence

K-NN Means

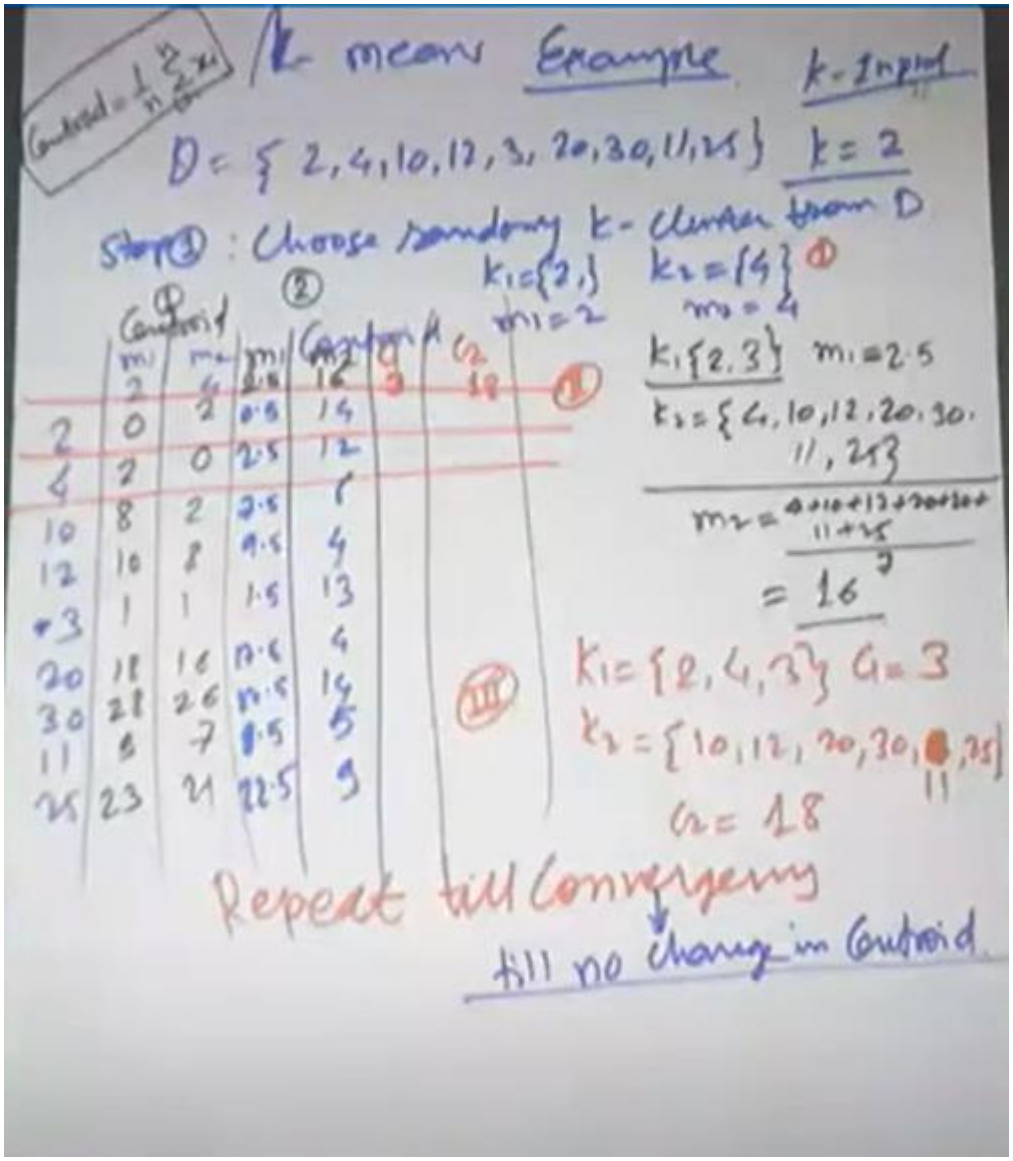
Convergence till no change in centroid.

k-Means Example

- Given: {2,4,10,12,3,20,30,11,25}, k=2
- Randomly assign means: $m_1=2, m_2=4$

1. $K_1=\{2,3\}$, $K_2=\{4,10,12,20,30,11,25\}$, $m_1=2.5, m_2=16$
2. $K_1=\{2,3,4\}$, $K_2=\{10,12,20,30,11,25\}$, $m_1=3, m_2=18$
3. $K_1=\{2,3,4,10\}$, $K_2=\{12,20,30,11,25\}$, $m_1=4.75, m_2=19.6$
4. $K_1=\{2,3,4,10,11,12\}$, $K_2=\{20,30,25\}$, $m_1=7, m_2=25$

- Stop as the clusters when these means are the same.


Centroid = 1/n Σ xi
k- means Example
k-Input
D = { 2, 4, 10, 12, 3, 20, 30, 11, 25 } k = 2
Step ① : Choose randomly k- clusters from D
k1 = {2,} k2 = {4} ①
m1 = 2 m2 = 4
②
Centroid
m1 m2 m1 m2 C1 C2
2 4 2 12 3 18 ①
2 0 2 14
4 2 0 12
10 8 2 10
12 10 8 4
3 1 1 13
20 18 16 4
30 28 26 14
11 9 7 5
25 23 21 9
k1 {2.3} m1 = 2.5
k2 = {4, 10, 12, 20, 30, 11, 25}
m2 = (4+10+12+20+30+11+25)/7
= 16
③
k1 = {2, 4, 3} C1 = 3
k2 = {10, 12, 20, 30, 11, 25}
C2 = 18
Repeat till Convergency
till no change in Centroid.

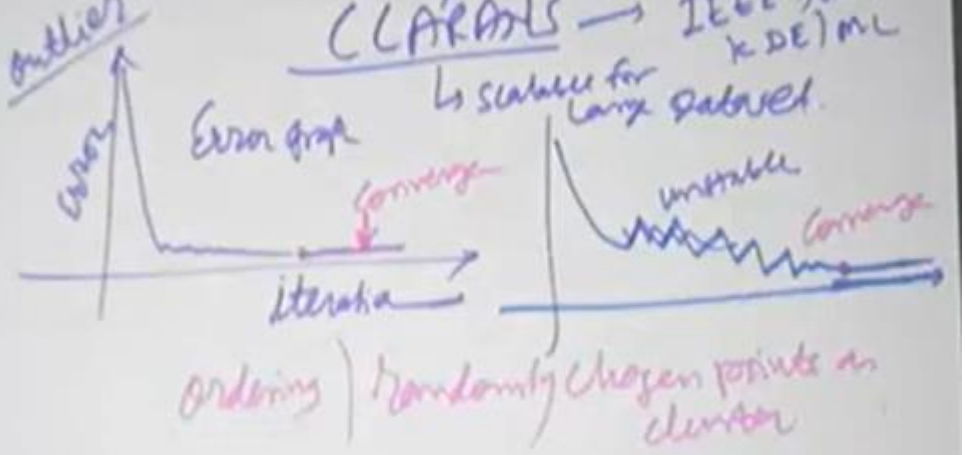
Partitioning Around Medoids (PAM) (k-Medoids)

- Handles outliers well.
- Ordering of input does not impact results.
- Does not scale well.
- Each cluster represented by one item, called the ***medoid.***
- Initial set of *k* medoids randomly chosen.

PAM

Partitioning Around medoid

- K-medoid → physical
- CLARANS → IEEE from KDE/ML
  - Is scalable for large dataset

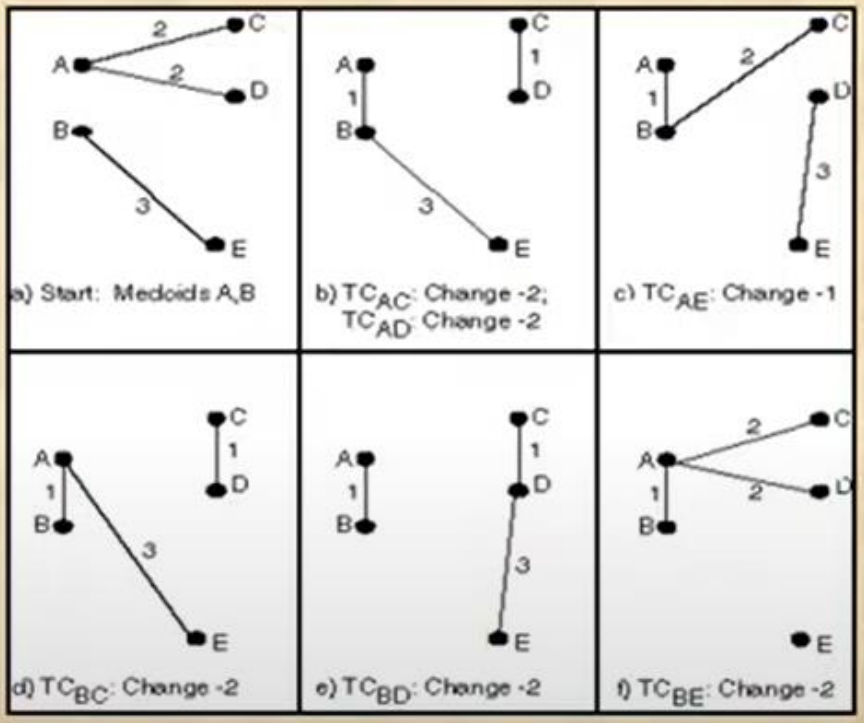


ordering / Randomly chosen points as cluster

PAM Algorithm

- Use real object to represent the cluster
  1. Select ***k*** representative objects arbitrarily
  2. For each pair of non-selected object ***h*** and selected object ***i***, calculate the total swapping cost TC_{ih}
  3. For each pair of ***i*** and ***h***,
     a) If $TC_{ih} < 0$, ***i*** is replaced by ***h***
     b) Then assign each non-selected object to the most similar representative object
  4. repeat steps 2-3 until there is no change

PAM



PAM Cost Calculation

- At each step in algorithm, medoids are changed if the overall cost is improved.
- C_{jih} – cost change for an item t_j associated with swapping medoid t_i with non-medoid t_h.

1. $t_j \in K_i$, but $\exists$ another medoid t_m *where* $dis(t_j, t_m) \leq dis(t_j, t_h)$
2. $t_j \in K_i$, but $dis(t_j, t_h) \leq dis(t_j, t_m) \forall$ *other medoids* t_m;
3. $t_i \in K_m$, $\notin K_i$, and $dis(t_j, t_m) \leq dis(t_j, t_h)$; and
4. $t_j \in K_m$, $\notin K_i$, but $dis(t_j, t_h) \leq dis(t_j, t_m)$.

PAM Algorithm Listing

```
Input:
   D = {t1, t2, ..., tn}   // Set of elements
   A     // Adjacency matrix showing distance between elements.
   k     // Number of desired clusters.
Output:
   K     // Set of clusters.
PAM Algorithm:
   arbitrarily select k medoids from D;
   repeat
      for each th not a medoid do
         for each medoid ti do
            calculate TCih;
      find i, h where TCih is the smallest;
      if TCih < 0 then
         replace medoid ti with th;
   until TCih ≥ 0;
   for each ti ∈ D do
      assign ti to Kj where dis(ti, tj) is the smallest over all medoids;
```

Swapping cost = Sum of distance of each non-selected point to selected (medoid) $= \sum dist_{pai}$.

Example

$D = \{2, 4, 10, 12, 3, 20, 30, 11, 25\}$

medoid: $k_1 = 2$ $k_2 = 4$

$nCr = \frac{n!}{r!(n-r)!}$

Choose unselected point 3

Pair with unselected {2, 3}

pair with remaining {3, 4} (2, 10), (2, 12)

Compute cost of new pair & old pairs.

keep pair having smallest cost / discard else.

Repeat above swapping till no change in Medoid

Lecture 26

Clustering Large Databases

- Most clustering algorithms assume a large data structure which is memory resident.
- Clustering may be performed first on a sample of the database then applied to the entire database.
- Algorithms
  - BIRCH
  - DBSCAN
  - CURE

Desired Features for Large Databases

- One scan (or less) of DB
- Online
- Suspendable, stoppable, resumable
- Incremental
- Work with limited main memory
- Different techniques to scan (e.g. sampling)
- Process each tuple once

BIRCH : Balanced Iterative Reducing and Clustering using Hierarchies

- Incremental, hierarchical, one scan
- Save clustering information in a tree
- Each entry in the tree contains information about one cluster
- New nodes inserted in closest entry in tree

Clustering Feature

- CT Triple: $(N, \overrightarrow{LS}, SS)$
  - N: Number of points in cluster
  - $\overrightarrow{LS}$: Sum of points in the cluster
  - SS: Sum of squares of points in the cluster
- CF Tree
  - Balanced search tree
  - Node has CF triple for each child
  - Leaf node represents cluster and has CF value for each subcluster in it.
  - Subcluster has maximum diameter

Clustering large Databases

BIRCH

DBSCAN

BIRCH

Incremental & hierarchical / Single Scan

→ Representing TREE → CF-tree

↓

Tree structure

Clustering Feature (CF): triplet

$CF = (N, LS, SS)$

Where N: No. of data points in D.

LS: Linear Sum of points

SS: Sum of Squared points

CF-tree consists nodes represented by CF

Statistics from CF

CF = (N, LS, SS)

Let D consisted N point

$$LS = \sum_{i=1}^{N} x_i \qquad SS = \sum_{i=1}^{N} (x_i)^2$$

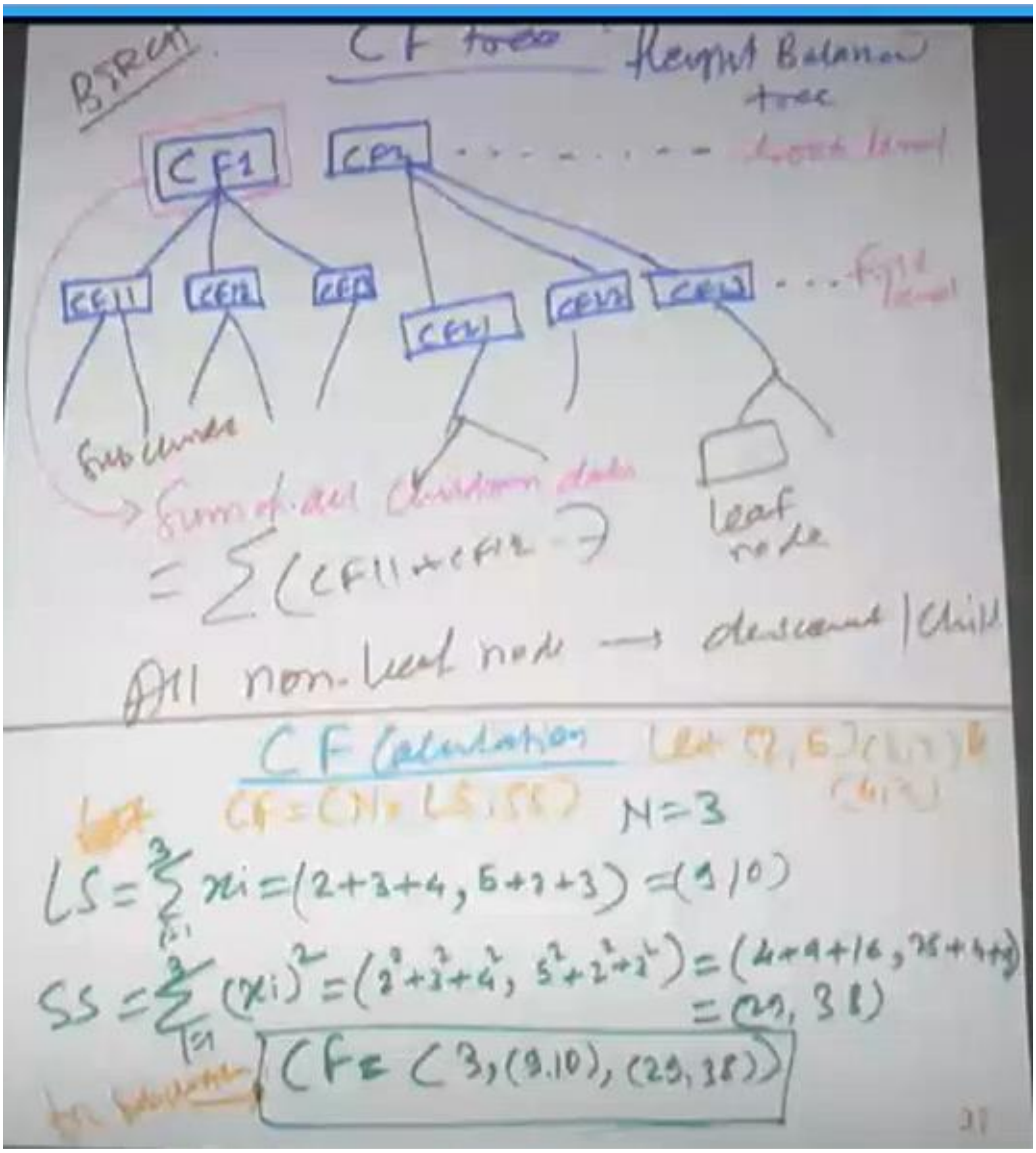
Cluster Centroid $x_0 = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{LS}{n}$

$$\text{Radius} = \sqrt{\frac{\sum_{i=1}^{N} (x_i - x_0)^2}{N}} = \sqrt{\frac{nSS - 2LS^2 + nLS}{n^2}}$$

$$\text{Diameter } D = \sqrt{\frac{\sum_{i=1}^{N}\sum_{i=1}^{N} (x_i - x_j)^2}{n(n-1)}} = \sqrt{\frac{2nSS - 2LS^2}{n(n-1)}}$$

R: Average distance from member objects to centroid.

D: Average pairwise distance within cluster


BIRCH
CF tree
Height Balance tree
CF1
CF11
CF12
CF13
CF21
CF22
CF23
sub cluster
Sum of all children data
= Σ (CF11 + CF12 ...)
leaf node
All non-leaf node → descendant | child
CF Calculation
CF = (N, LS, SS)
N = 3
LS = Σ xi = (2+3+4, 5+2+3) = (9, 10)
SS = Σ (xi)^2 = (2^2+3^2+4^2, 5^2+2^2+3^2) = (4+9+16, 25+4+9)
= (29, 38)
CF = <3, (9,10), (29,38)>

BIRCH Algorithm → B+ tree

Two parameters: i) Branching factor 'B'
ii) Threshold 'T'

B = Max. no. of children per non leaf node

T = max. Diameter of subcluster stored at leaf node of tree

CF tree
&
B+ tree

Two phases

(I) Scan the database to build initial tree

(II) Apply any other clustering algorithm to cluster each leaf node

CF 1 →

non leaf node

Split ... B & T

BIRCH Algorithm

Input:
$D = \{t_1, t_2, ..., t_n\}$ // Set of elements
T // Threshold for CF tree construction.
Output:
K // Set of clusters.
BIRCH Clustering Algorithm:

```
for each t_i ∈ D do
    determine correct leaf node for t_i insertion;
    if threshold condition is not violated then
        add t_i to cluster and update CF triples;
    else
        if room to insert t_i then
            insert t_i as single cluster and update CF triples;
        else
```


Improve Clusters

1. Create initial CF tree using Algorithm. If there is insufficient memory to construct the CF tree with a given threshold, the threshold value is increased, and a new smaller CF tree is constructed.

2. Apply another global clustering approach applied to the leaf nodes in the CF tree. Here each leaf node is treated as a single point for clustering.

3. The last phase (which is optional) reclusters all points by placing them in the cluster which has the closest centroid.

A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise

DBSCAN

Martin Ester, Hans-Peter Kriegel, Jiirg Sander, Xiaowei Xu

Institute for Computer Science, University of Munich
Oettingenstr. 67, D-80538 Miinchen, Germany
{ester I kriegel I sander I xwxu } @informatik.uni-muenchen

KDD-96 Proceedings. pp 226-231

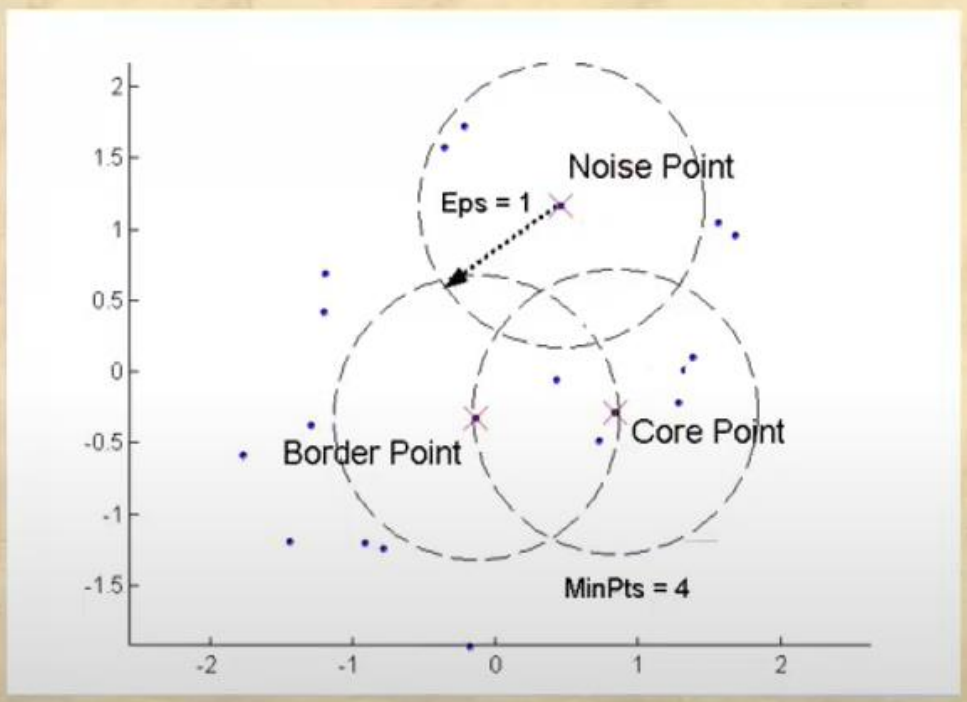
DBscan Algorithm

Density-Based Spatial Clustering of Applications with Noise

DBSCAN is a density-based algorithm.

- ***Eps*** : Maximum radius of the neighbourhood
- ***MinPts*** : Minimum number of points in an Eps-neighbourhood of that point
- Density = number of points within a specified radius (***Eps***)
- A point is a *core point* if it has more than a specified number of points (***MinPts***) within ***Eps***
  - These are points that are at the interior of a cluster
- A *border point* has fewer than ***MinPts*** within ***Eps***, but is in the neighborhood of a core point
- A *noise point* is any point that is not a *core point* or a *border point*.

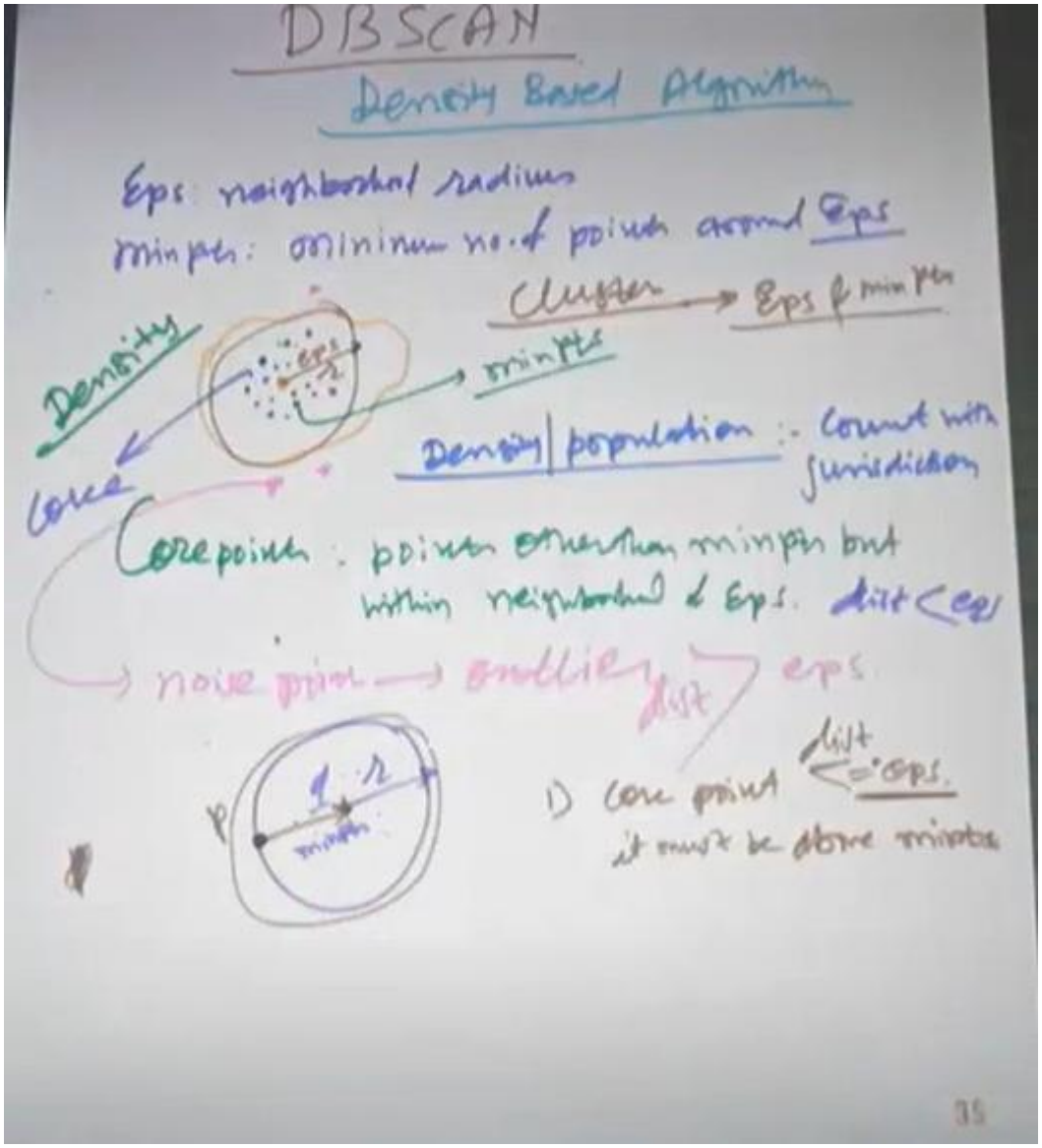
DBSCAN: *Core*, *Border*, and *Noise Points*



DBscan : other parameters

- ***Directly density-reachable***: A point ***p*** is directly density-reachable from a point ***q*** w.r.t. *Eps*, *MinPts* if
  - *p* belongs to $N_{Eps}(q)$
  - ***$N_{Eps}(q)$: { p belongs to D | dist(p,q) <=Eps }***
  - core point condition:

    $|N_{Eps}(q)| >= MinPts$



DBscan Algorithm : Generic Steps

- Arbitrary select a point *p*
- Retrieve all points *density-reachable* from *p* w.r.t. ***Eps*** and ***MinPts***.
- If *p* is a *core point*, a cluster is formed.
- If *p* is a *border point*, no points are density-reachable from *p* and DBSCAN visits the next point of the database.
- Continue the process until all of the points have been processed.

DBSCAN Algorithm

Algorithm: DBSCAN: a density-based clustering algorithm.

Input:

D: a data set containing *n* objects,

: the radius parameter, and

MinPts: the neighborhood density threshold.

Output: A set of density-based clusters.

Cont..

Method:

(1) mark all objects as unvisited;
(2) **do**
(3) randomly select an unvisited object ***p***;
(4) mark ***p*** as visited;
(5) **if** the -neighborhood of ***p*** has at least *MinPts* objects
(6) create a new cluster *C*, and add ***p*** to *C*;
(7) let *N* be the set of objects in the -neighborhood of ***p***;
(8) **for** each point ***p***0 in *N*
(9) if ***p***0 is unvisited
(10) mark ***p***0 as visited;
(11) if the -neighborhood of ***p***0 has at least *MinPts* points, add those points to *N*;
(12) if ***p***0 is not yet a member of any cluster, add ***p***0 to *C*;
(13) **end for**
(14) output *C*;
(15) **else** mark ***p*** as noise;
(16) **until** no object is unvisited;

DBSCAN Algorithm

I/P : D of n objects, ϵ, eps, minPts.

O/P : Clusters formed.

1. marks all points as unvisited
2. Randomly select unvisited 'p'
3. mark as visited
4. If neighborhood of p → minPts
   find no. of points around 'p' with eps
   of that no. in minPts, create new cluster C
   & add 'p' to C
   → N : No. of points around p with eps
   Then for each point p' ∈ N
   Repeat 3 & 4
   Else, mark p as noise or outlier
5. repeat to 2 if points are still remain unvisited
6. output cluster 'k'
7. End

How to select

Selection of eps & minPts → attacks DBSCAN

tips

minPts = Dimensional data + 1

Eg for IRIS : 4

∴ minPts = 4 + 1 = 5

Radius eps : Maximum distance among pairs of data points

Whole DBSCAN cluster depends on selection (optimal) of eps & minPts

Lecture 27

Evaluation of Clustering

19/01/21 @10:15

To determine how clusters have formed?

Assessment

1. Assessing cluster tendancy
2. Determine the no. of Good/optimal Clusters K
3. measuring the cluster Quality

1. Clustering Tendancy

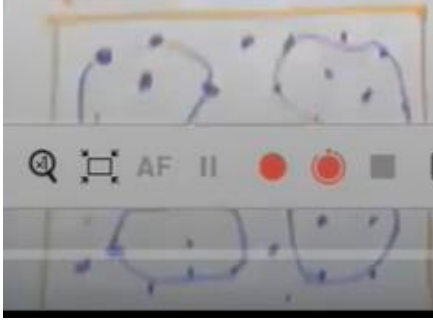
for given Data set ⇒ form clusters

↳ eligible to form clusters

↳ Characteristics

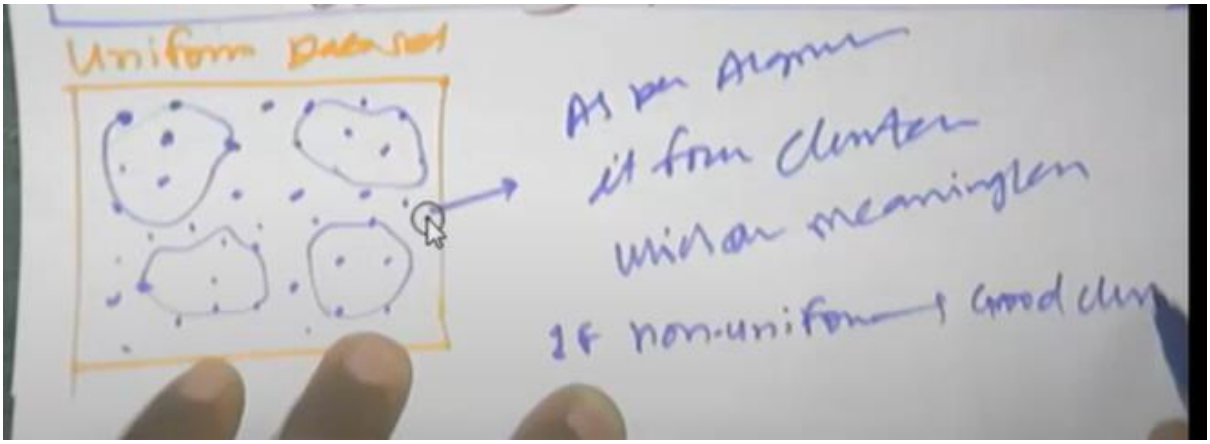
For Good clustering ⇒ Dataset ⇒ non-uniform

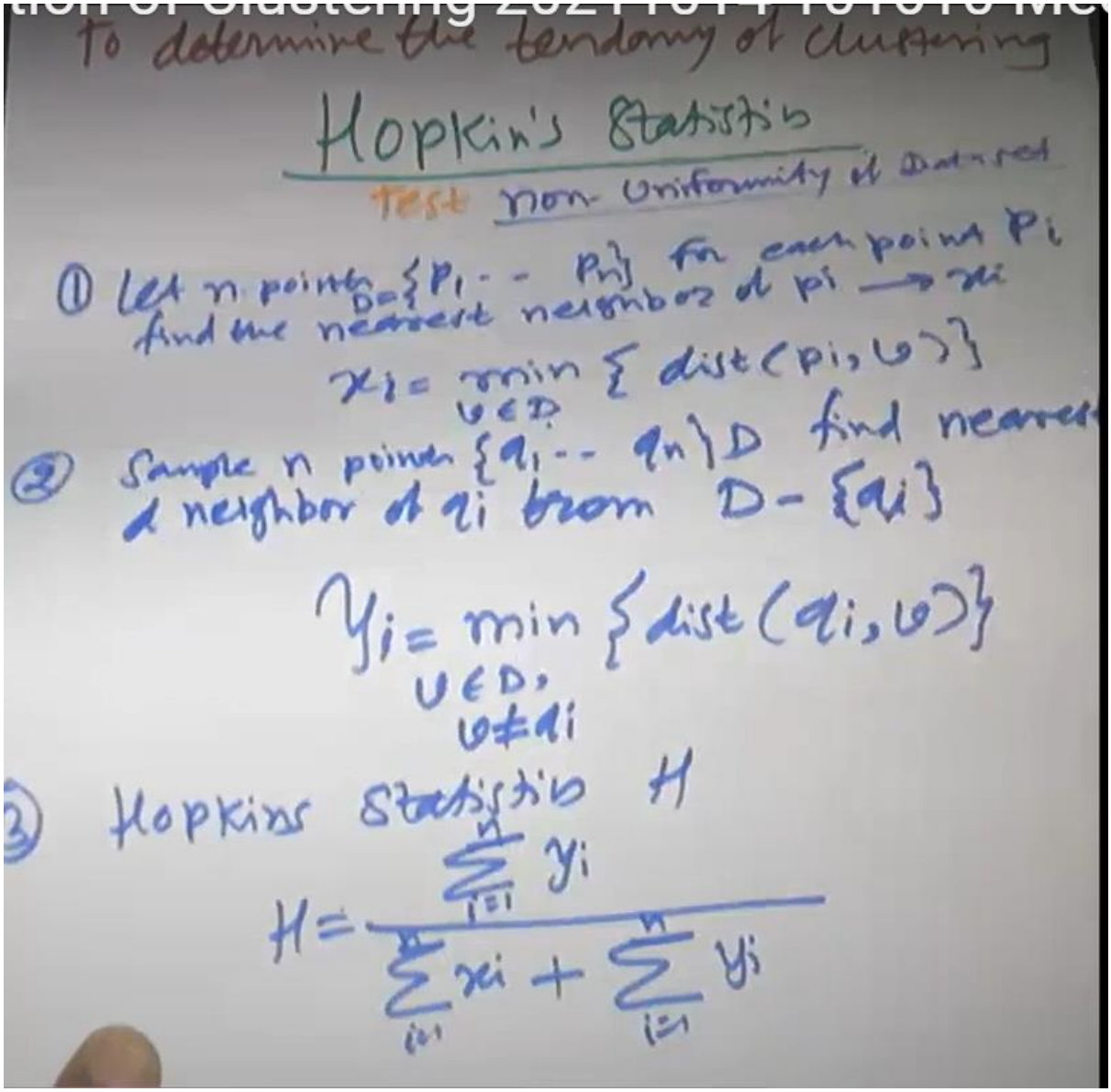
Uniform Dataset



As per Algorithm it form clusters which are meaningless

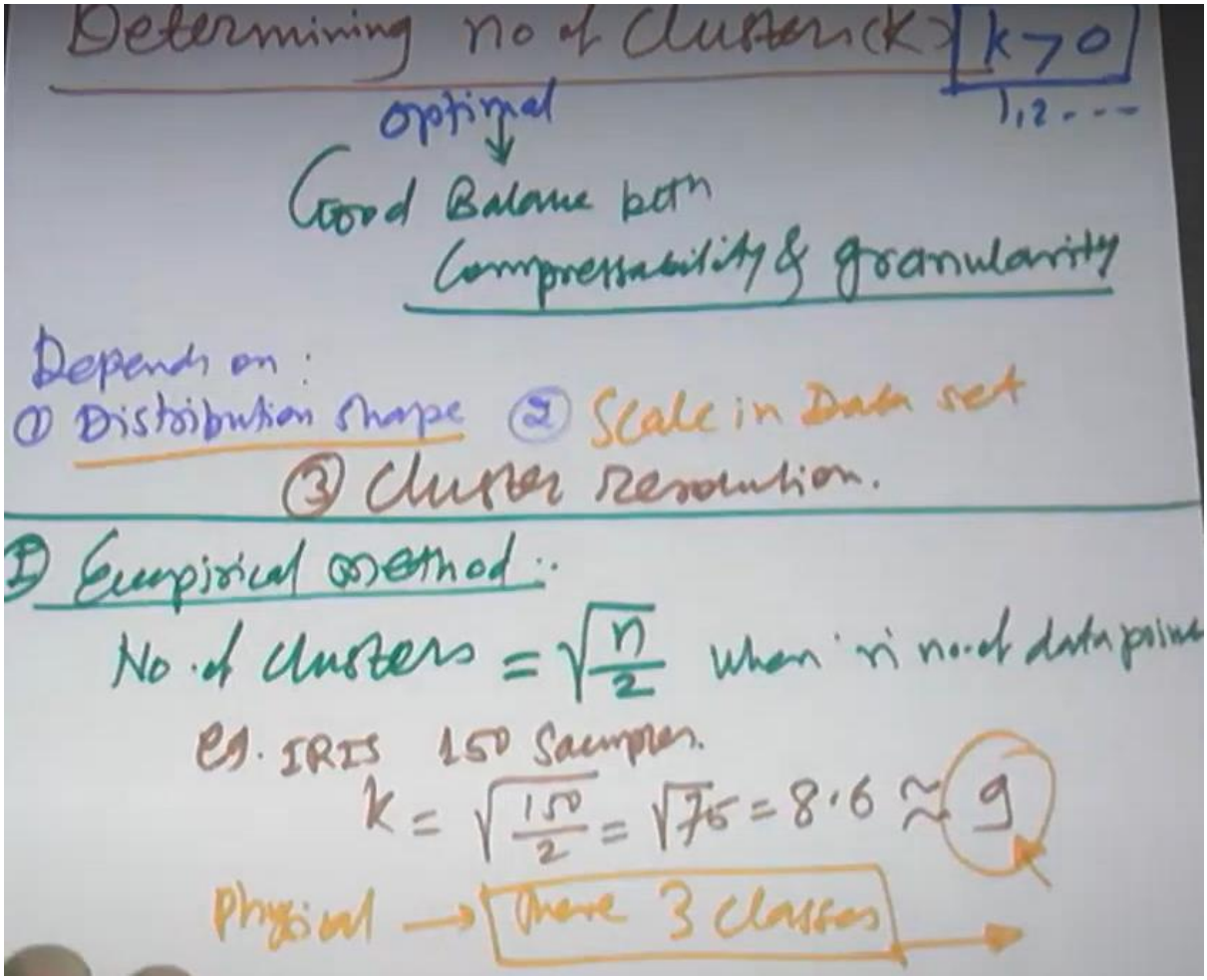
If non-uniform → Good clusters





H=0.5 for uniform dataset and

H=0 for non-uniform dataset



(11) Elbow method

Increasing no of cluster → reduce the sum of "within-cluster variance.

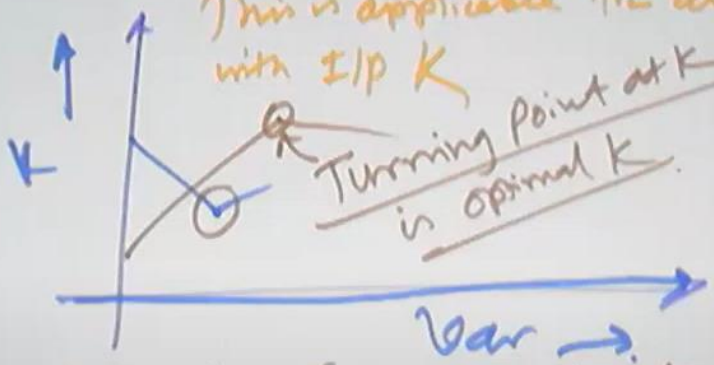
"within cluster" - Variance

Find the sum of within cluster variance for k

Var(k)

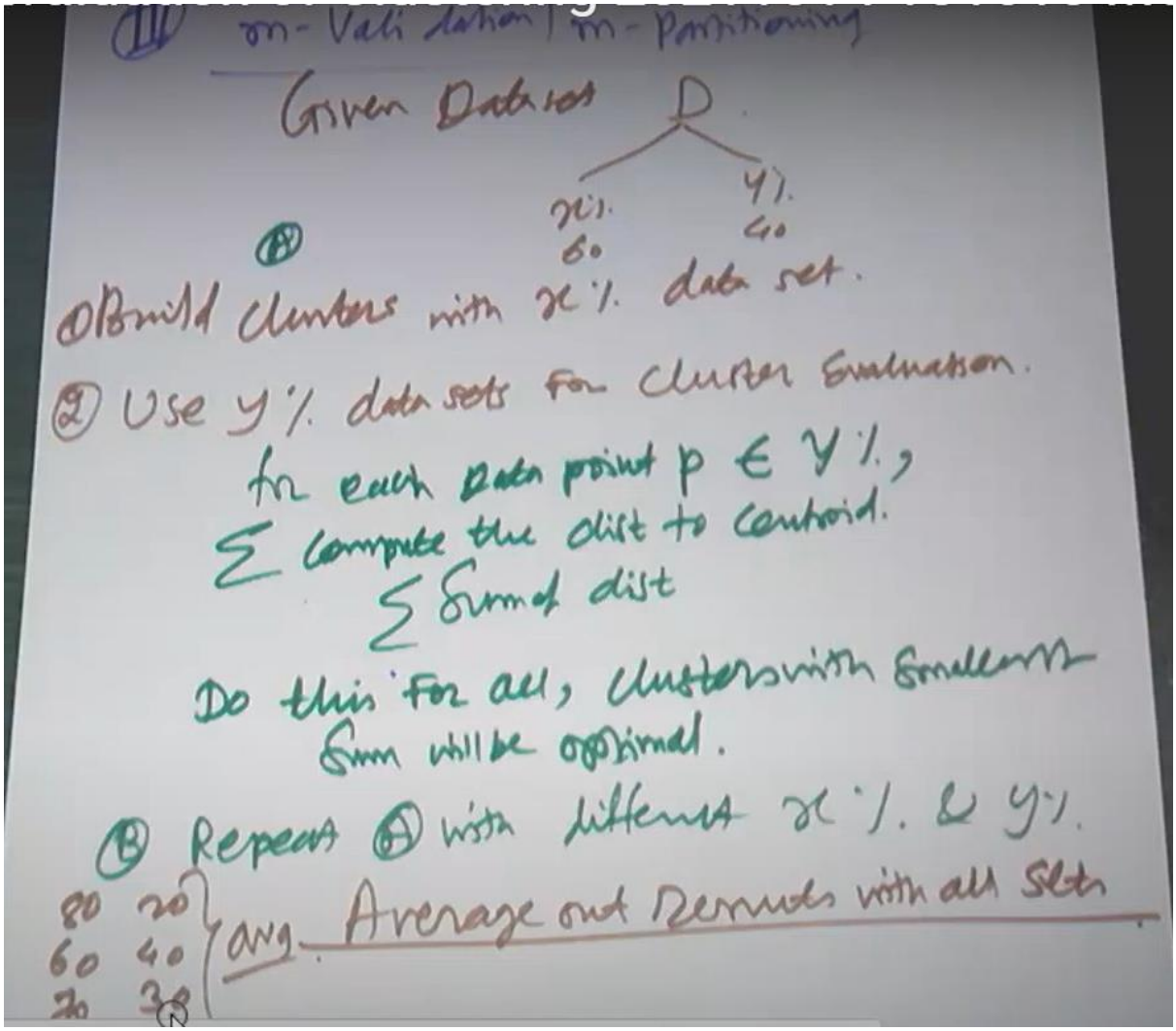
k=1 to m

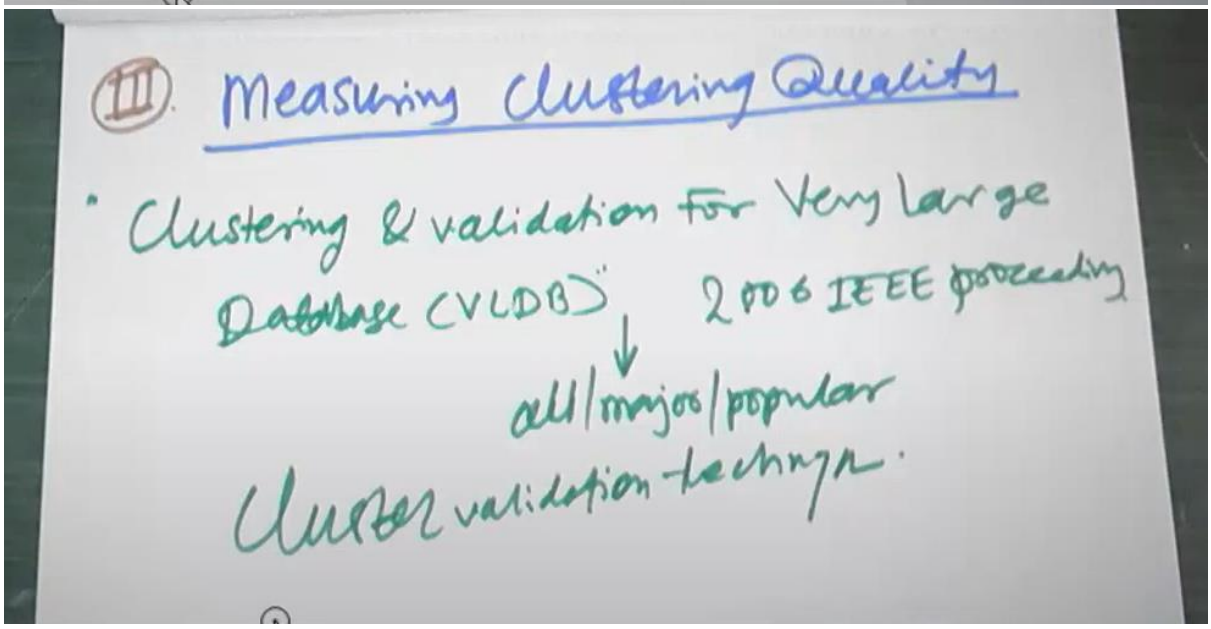
This is applicable for algorithms with I/p K



① Compute variance for each cluster.

② plot Var → K → check turning point in graph → Turning point @ k will be optimal





Sir has published an IEEE paper for the above mentioned topic.

(III) Measuring Clustering Quality

- Clustering & validation for Very Large Database (VLDB) 2006 IEEE proceeding
  - ↓
  - all/major/popular
  - Cluster validation techniques.

dist > high — Inter Cluster Similarity → low

dist < small — intra Cluster Similarity → high

Inter Cluster distance → high

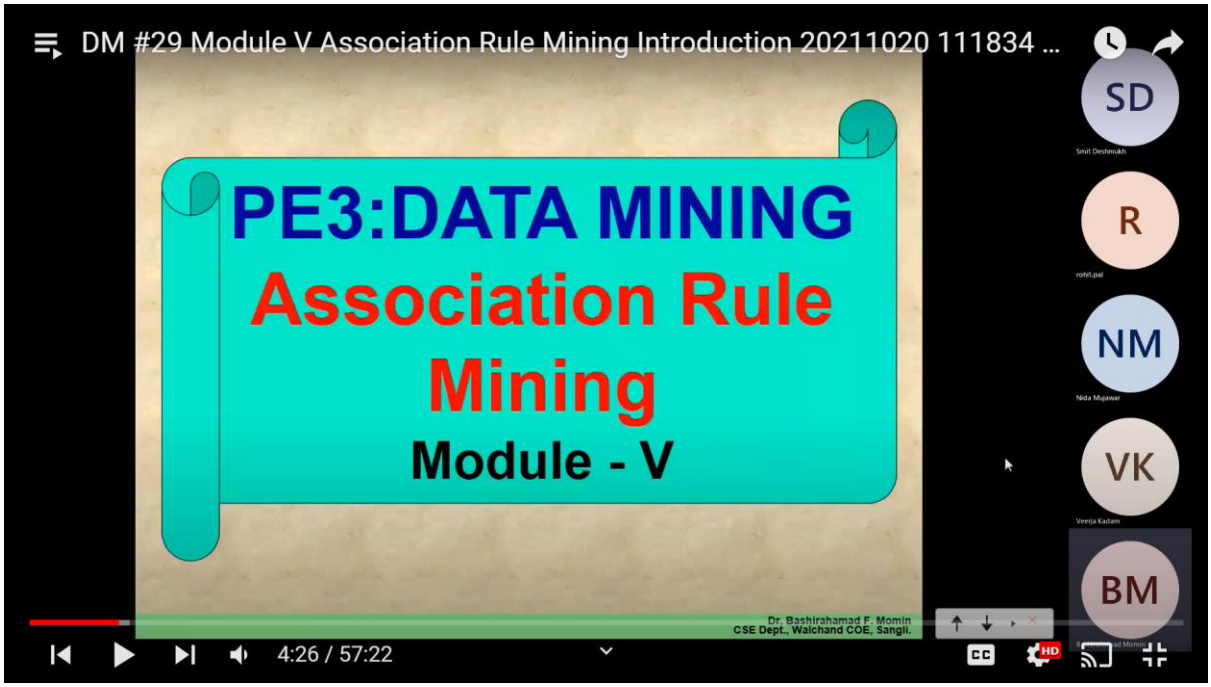
Intra cluster dista → small

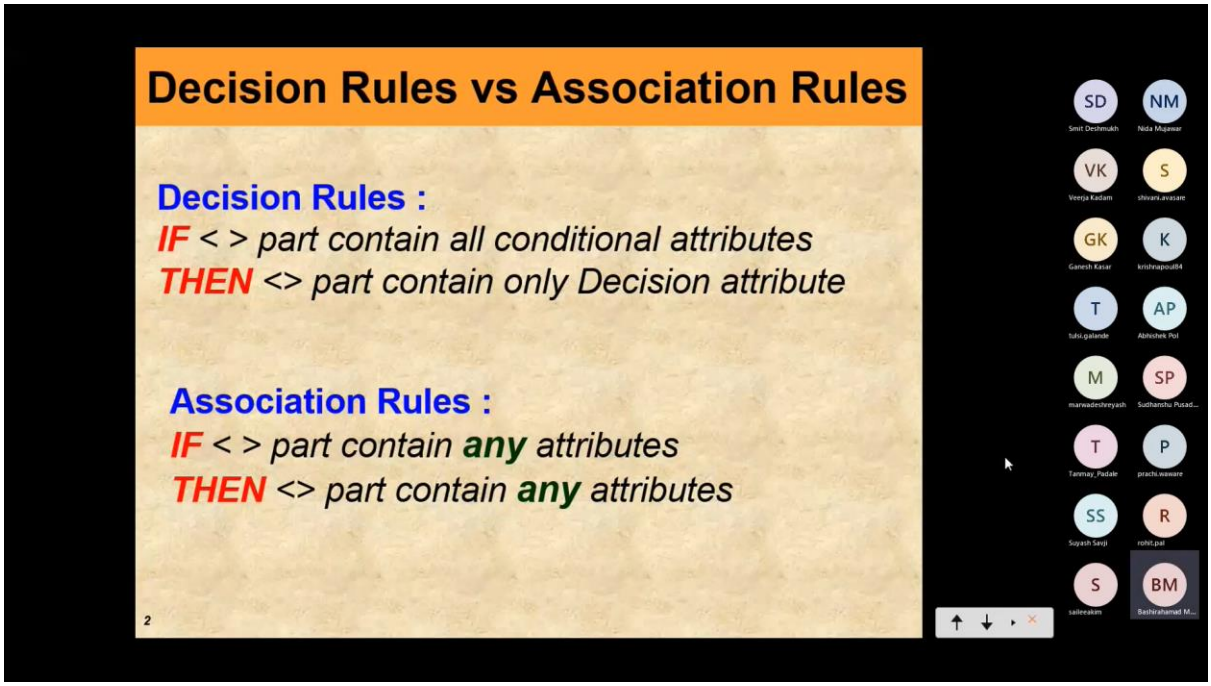
Soralers Distance → most similar

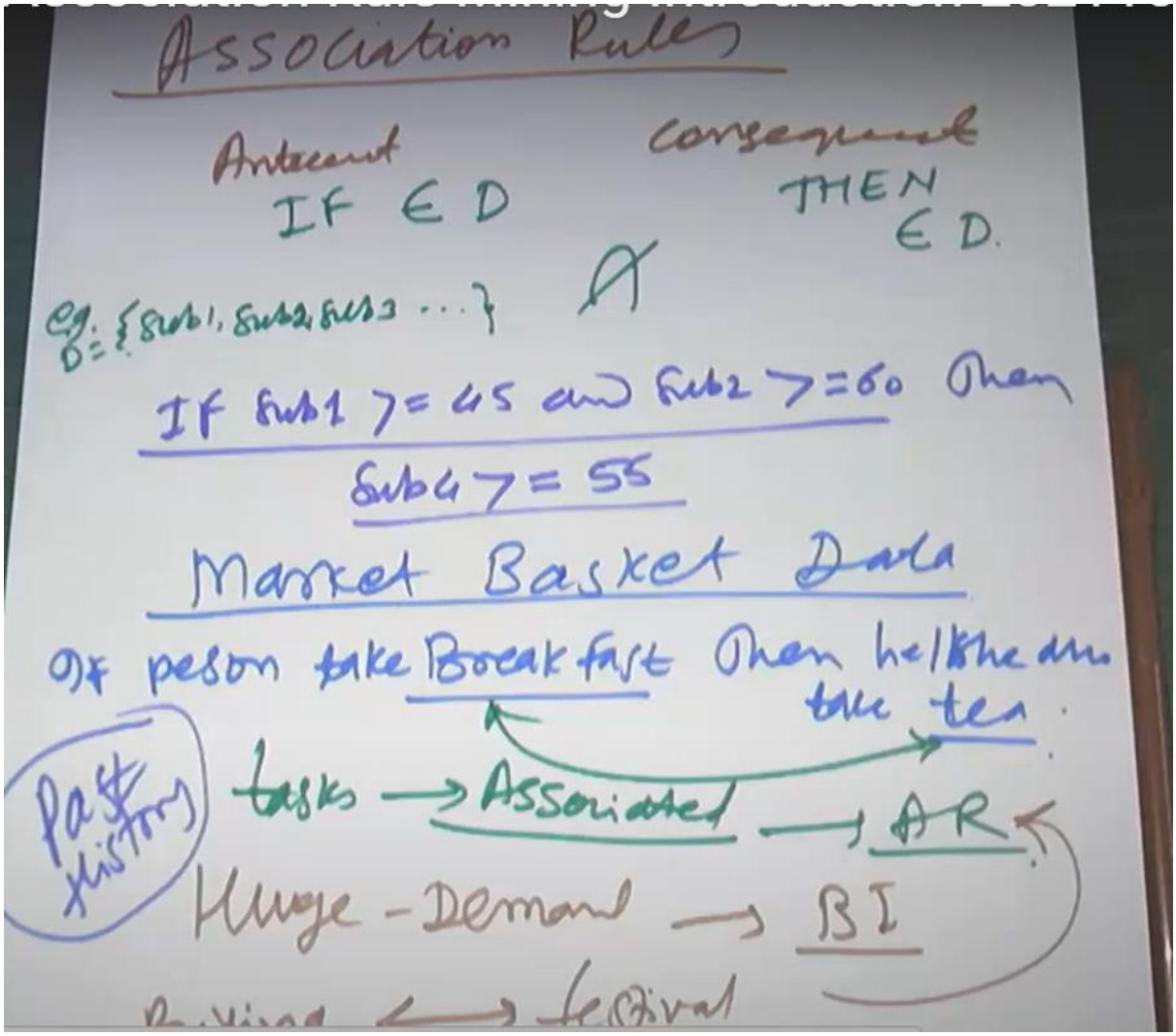
NN → Soralers distance

→ Silhouette's Statistics

Lecture 29


DM #29 Module V Association Rule Mining Introduction 20211020 111834 ...
PE3:DATA MINING
Association Rule Mining
Module - V
Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.
4:26 / 57:22
SD
R
NM
VK
BM


Decision Rules vs Association Rules
Decision Rules :
IF < > part contain all conditional attributes
THEN <> part contain only Decision attribute
Association Rules :
IF < > part contain any attributes
THEN <> part contain any attributes
2



Definitions of various terms

- ***Set of items:*** $I=\{I_1,I_2,\ldots,I_m\}$
- ***Transactions:*** $D=\{t_1,t_2,\ldots,t_n\}$, $t_j \subseteq I$
- ***Itemset:*** $\{I_{i1},I_{i2},\ldots,I_{ik}\} \subseteq I$
- ***Support of an itemset:*** Percentage of transactions which contain that itemset.
- ***Large (Frequent) itemset:*** **Itemset** whose number of occurrences is above a threshold.

Itemset

For ‘n’ items in data set, there will be $2^n - 1$ subset

An *itemset* is any subset of the set of all items

Association Rules Example

Transaction	Items
t_1	Bread,Jelly,PeanutButter
t_2	Bread,PeanutButter
t_3	Bread,Milk,PeanutButter
t_4	Beer,Bread
t_5	Beer,Milk

I = { Beer, Bread, Jelly, Milk, PeanutButter}

Support of {Bread,PeanutButter} is 60%

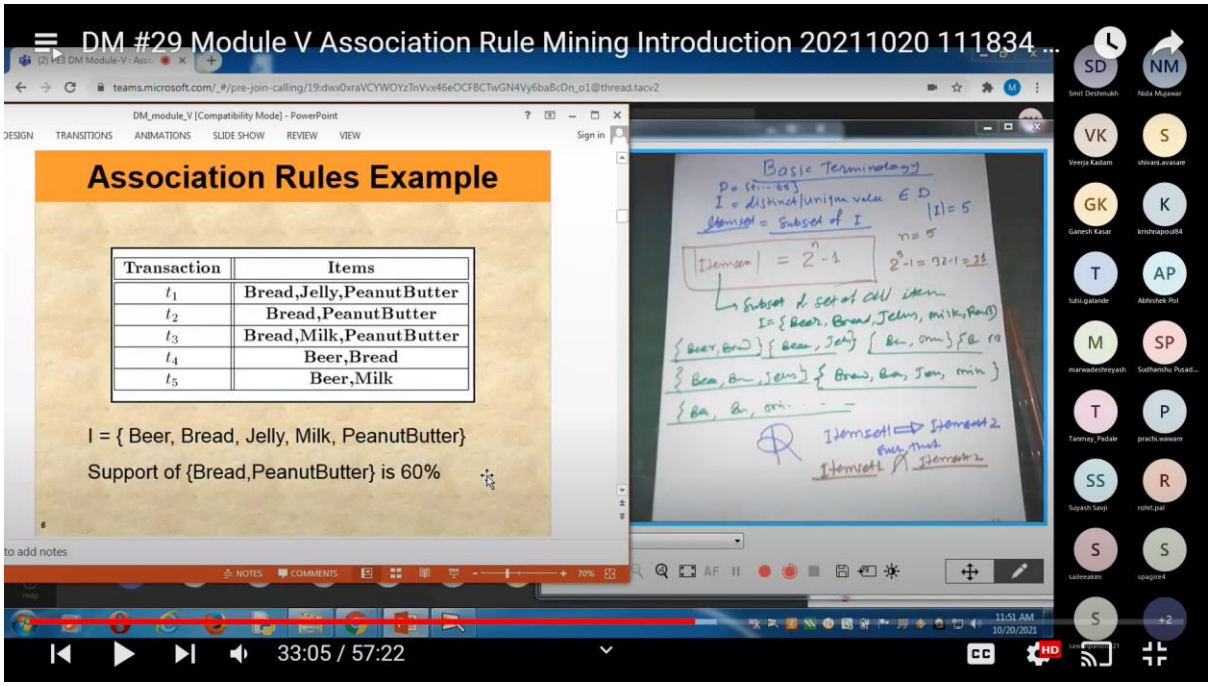
Example: Market Basket Data

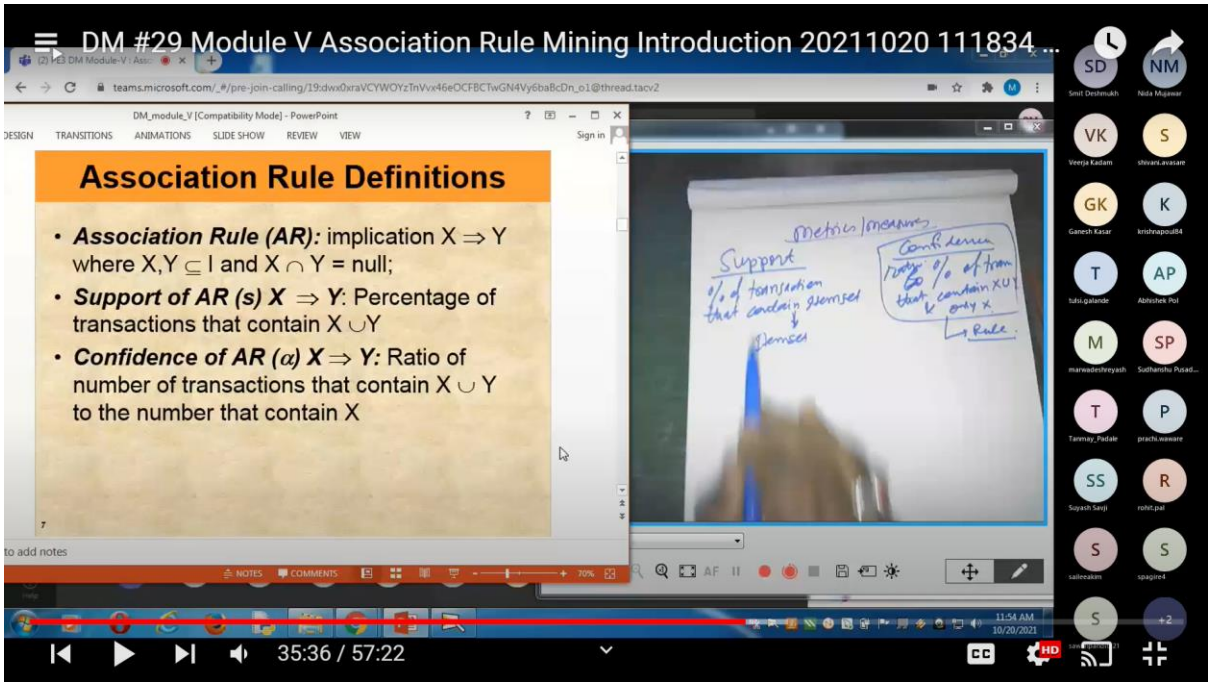
- Items frequently purchased together:

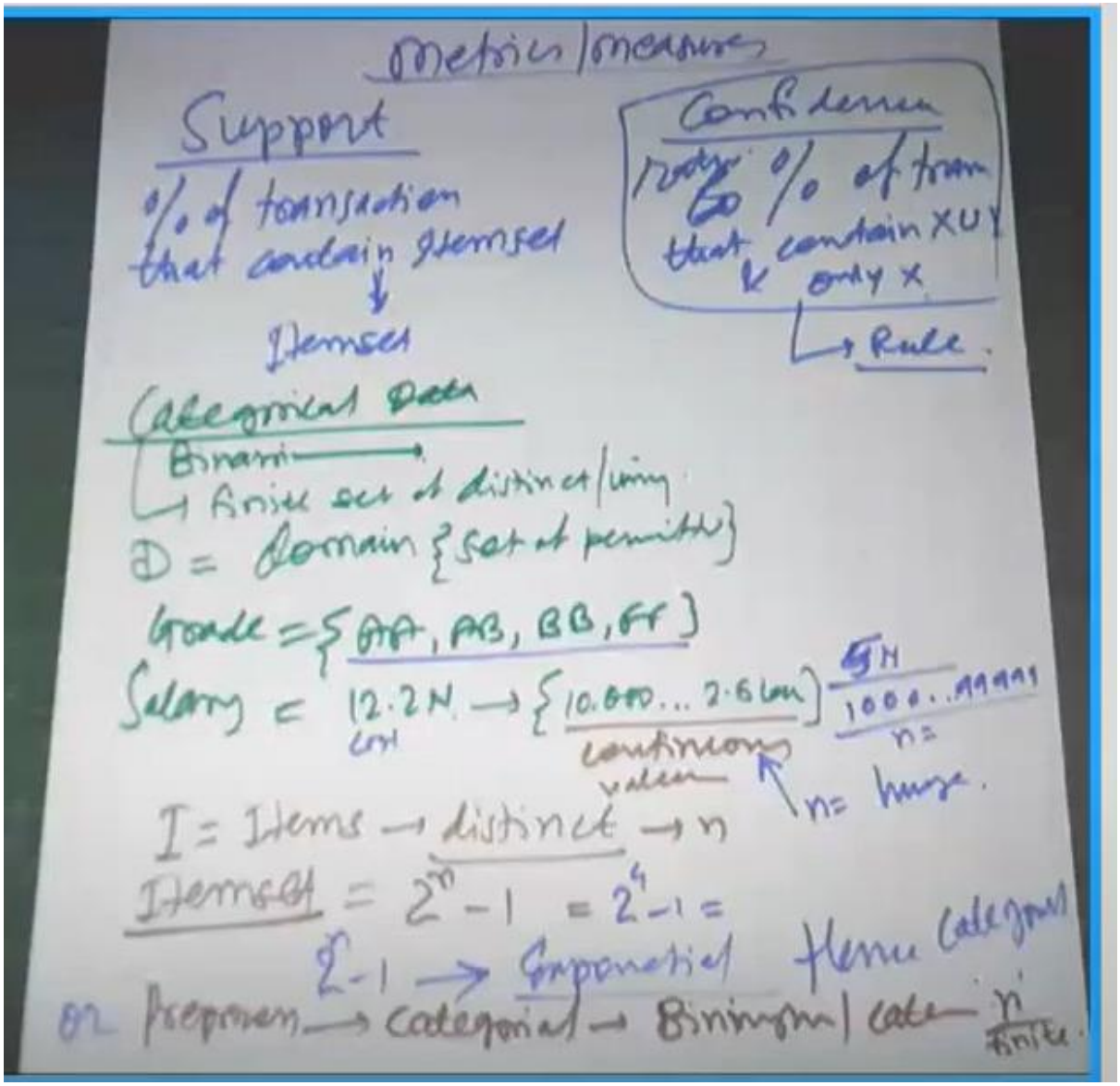
  Bread ⇒PeanutButter

- Uses:
  - Placement
  - Advertising
  - Sales
  - Coupons
- Objective: increase sales and reduce costs

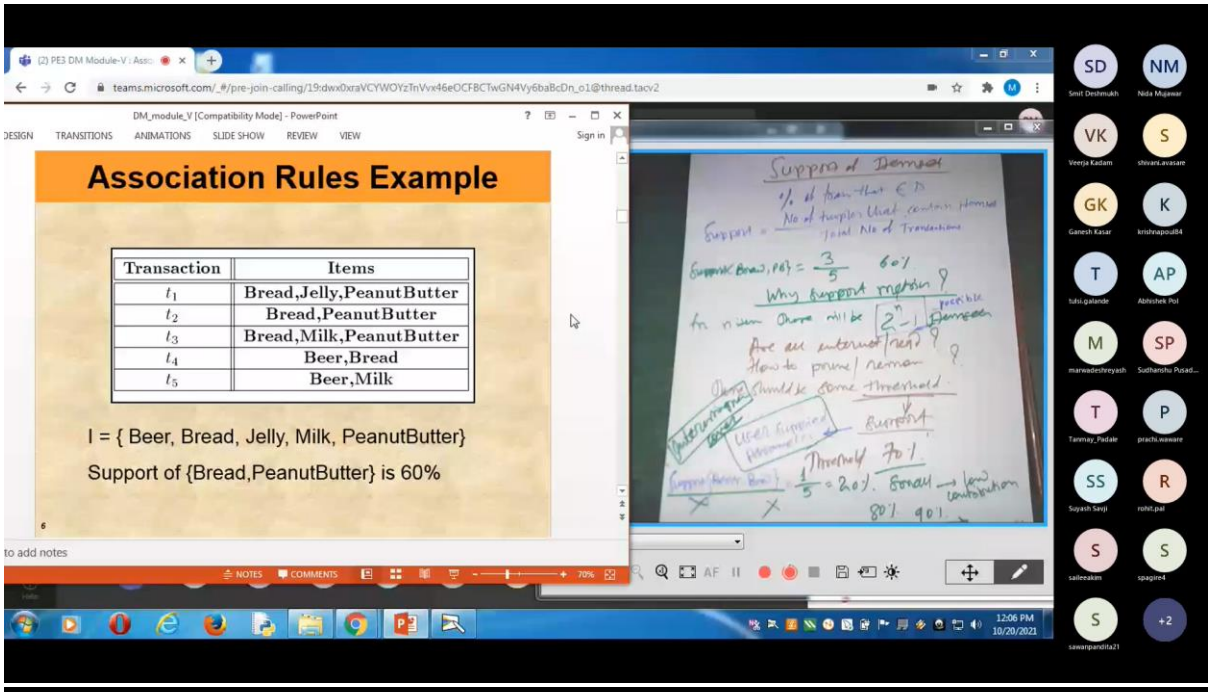
5

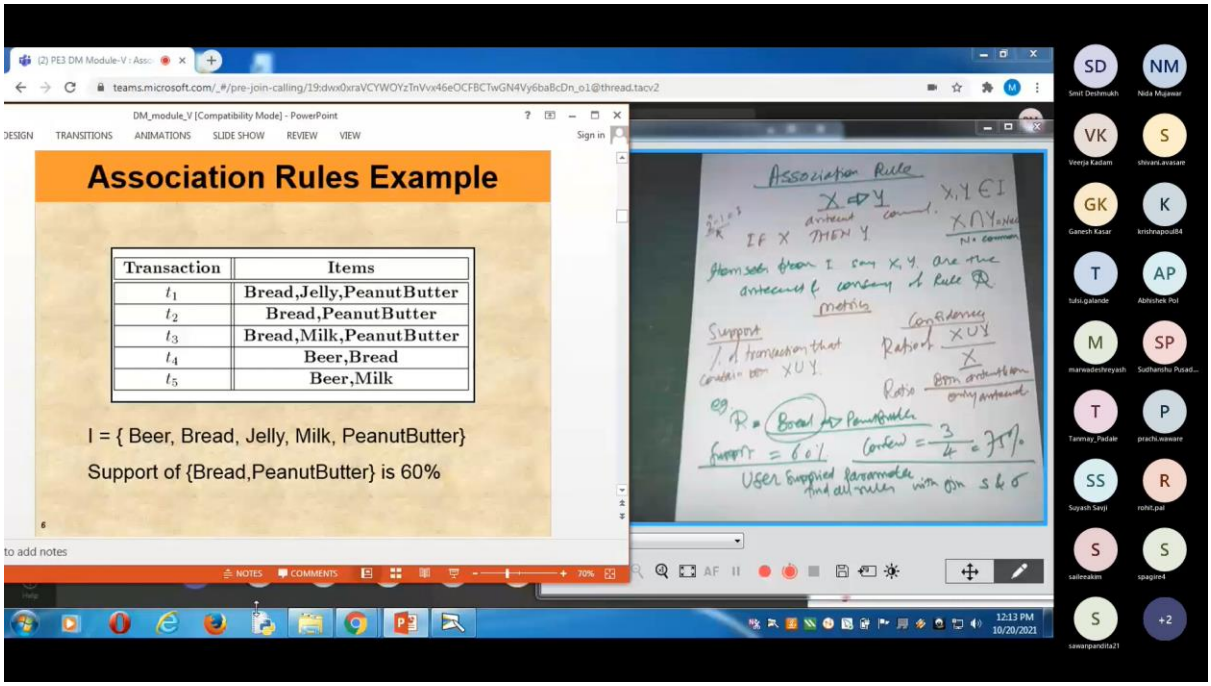


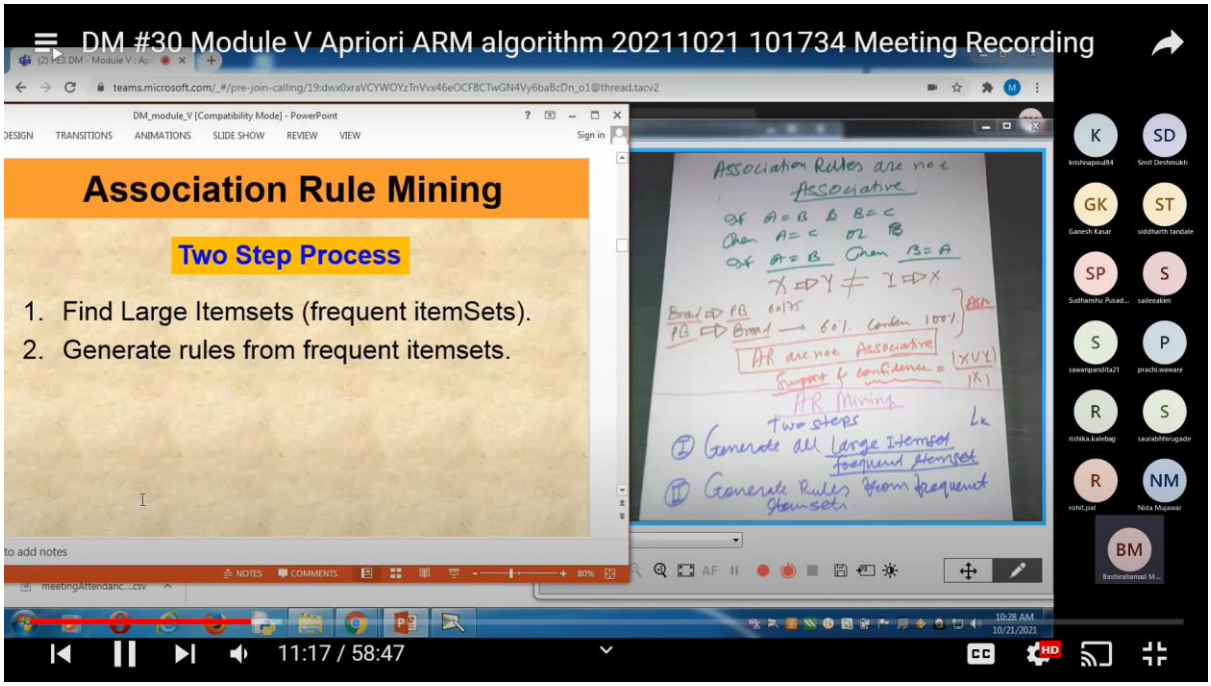

DM #29 Module V Association Rule Mining Introduction 20211020 111834 ...
Association Rule Definitions
• Association Rule (AR): implication X ⇒ Y where X,Y ⊆ I and X ∩ Y = null;
• Support of AR (s) X ⇒ Y: Percentage of transactions that contain X ∪Y
• Confidence of AR (α) X ⇒ Y: Ratio of number of transactions that contain X ∪ Y to the number that contain X
Metrics/measures
Support
% of transaction that contain itemset
Itemset
Confidence
Rule
35:36 / 57:22

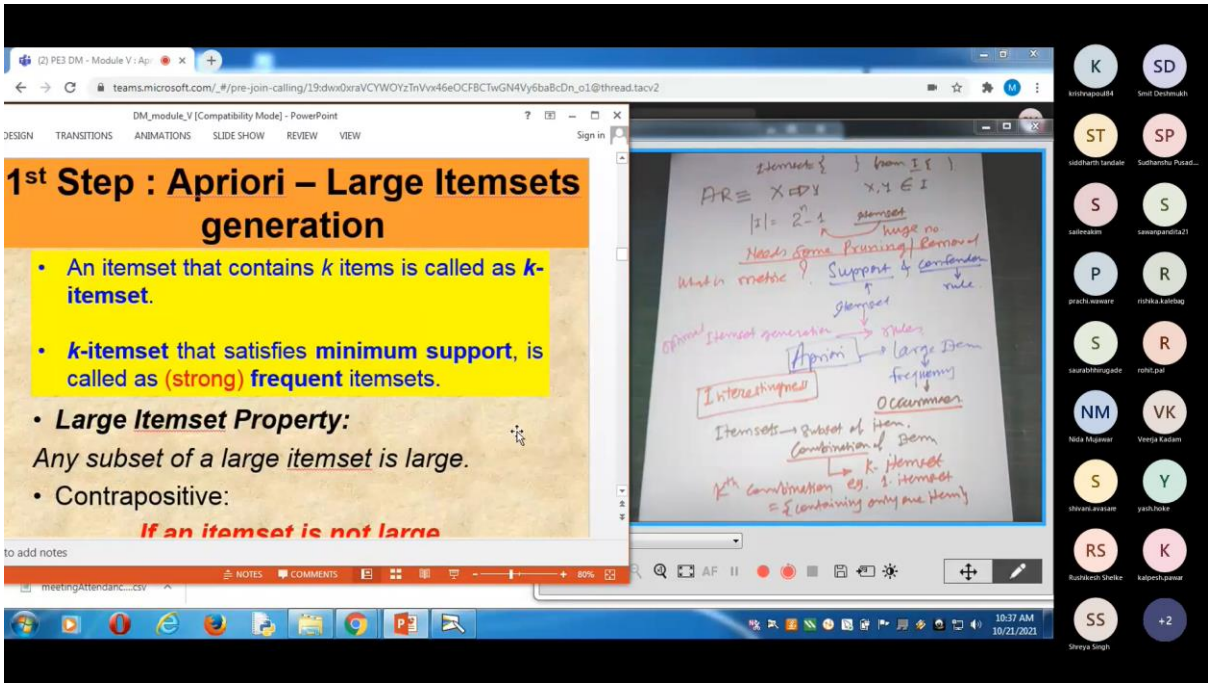

Metrics/measures
Support
% of transaction that contain itemset
Itemset
Confidence
Rule
Categorical Data
Binary
Domain {set of permitted}
Grade = {AA, AB, BB, FF}
Salary ∈ 12.2N → {10,000 ... 2.6 Lac}
continuous values
n = huge
I = Items → distinct → n
Itemset = 2^n - 1 = 2^4 - 1 =
2^-1 → Exponential Hence Categorical
Preprocess → categorical → Binning/ Categorize
finite

Lecture 30


Association Rules Example
Transaction | Items
t_1 | Bread,Jelly,PeanutButter
t_2 | Bread,PeanutButter
t_3 | Bread,Milk,PeanutButter
t_4 | Beer,Bread
t_5 | Beer,Milk
I = { Beer, Bread, Jelly, Milk, PeanutButter}
Support of {Bread,PeanutButter} is 60%


Association Rules Example
Transaction | Items
t_1 | Bread,Jelly,PeanutButter
t_2 | Bread,PeanutButter
t_3 | Bread,Milk,PeanutButter
t_4 | Beer,Bread
t_5 | Beer,Milk
I = { Beer, Bread, Jelly, Milk, PeanutButter}
Support of {Bread,PeanutButter} is 60%


DM #30 Module V Apriori ARM algorithm 20211021 101734 Meeting Recording
Association Rule Mining
Two Step Process
1. Find Large Itemsets (frequent itemSets).
2. Generate rules from frequent itemsets.
11:17 / 58:47


1st Step : Apriori – Large Itemsets generation
An itemset that contains k items is called as k-itemset.
k-itemset that satisfies minimum support, is called as (strong) frequent itemsets.
Large Itemset Property:
Any subset of a large itemset is large.
Contrapositive:
If an itemset is not large

1st Step : Apriori – Large Itemsets generation

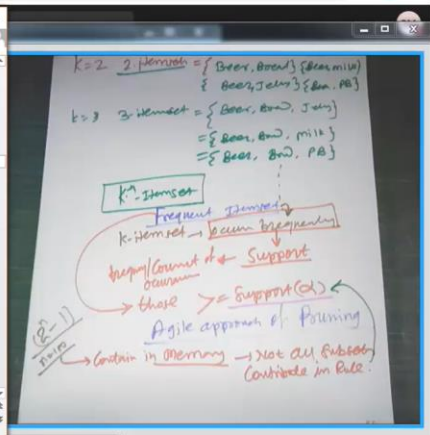
- An itemset that contains *k* items is called as ***k*-itemset**.
- ***k*-itemset** that satisfies **minimum support**, is called as (strong) **frequent** itemsets.
- ***Large Itemset Property:***

Any subset of a large itemset is large.

- Contrapositive:

If an itemset is not large, none of its supersets are large.

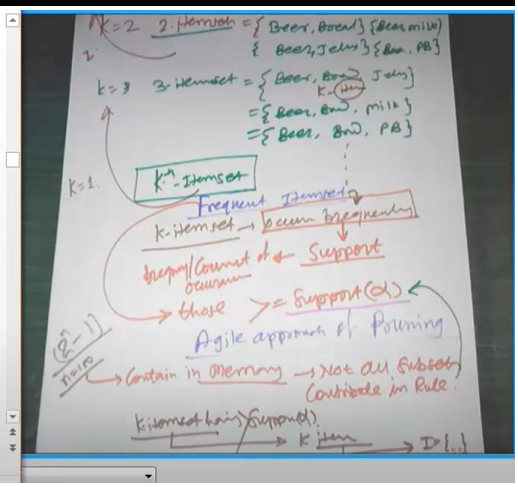
11



Apriori Algorithm

1. C_1 = Itemsets of size one in I;
2. Determine all large itemsets of size 1, L_1;
3. i = 1;
4. Repeat
5. i = i + 1;
6. C_i = Apriori-Gen(L_{i-1});
7. Count C_i to determine L_i;
8. until no more large itemsets found;

13

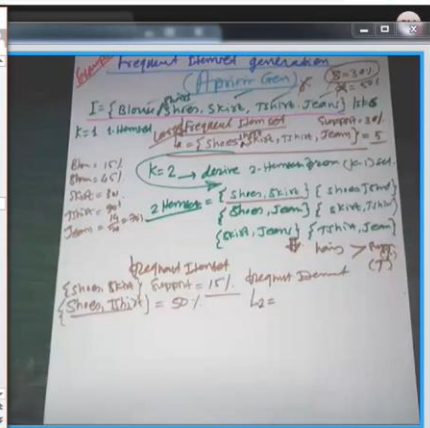


Apriori-Gen Example (cont'd)

s=30% α = 50%

Scan	Candidates	Large Itemsets
1	{Blouse},{Jeans},{Shoes}, {Shorts},{Skirt},{TShirt}	{Jeans},{Shoes},{Shorts} {Skirt},{Tshirt}
2	{Jeans,Shoes},{Jeans,Shorts},{Jeans,Skirt}, {Jeans,TShirt},{Shoes,Shorts},{Shoes,Skirt}, {Shoes,TShirt},{Shorts,Skirt},{Shorts,TShirt}, {Skirt,TShirt}	{Jeans,Shoes},{Jeans,Shorts}, {Jeans,TShirt},{Shoes,Shorts}, {Shoes,TShirt},{Shorts,TShirt}, {Skirt,TShirt}
3	{Jeans,Shoes,Shorts},{Jeans,Shoes,TShirt}, {Jeans,Shorts,TShirt},{Jeans,Skirt,TShirt}, {Shoes,Shorts,TShirt},{Shoes,Skirt,TShirt}, {Shorts,Skirt,TShirt}	{Jeans,Shoes,Shorts}, {Jeans,Shoes,TShirt}, {Jeans,Shorts,TShirt}, {Shoes,Shorts,TShirt}
4	{Jeans,Shoes,Shorts,TShirt}	{Jeans,Shoes,Shorts,TShirt}
5	∅	∅

16



Lecture 32:

2nd Step : Algorithm to Generate Association Rules

Input:
- D //Database of transactions
- I //Items
- L //Large itemsets
- s //Support
- α //Confidence

Output:
- R //Association Rules satisfying s and α

ARGen Algorithm:

$R = \emptyset$;
for each $l \in L$ do
  for each $x \subset l$ such that $x \neq \emptyset$ and $x \neq l$ do
    if $\frac{support(l)}{support(x)} \geq \alpha$ then
      $R = R \cup \{x \Rightarrow (l - x)\}$;

18

Association Rule Generation Apriori Algorithm

If there are *n* items

Then maximum no. of rules generated

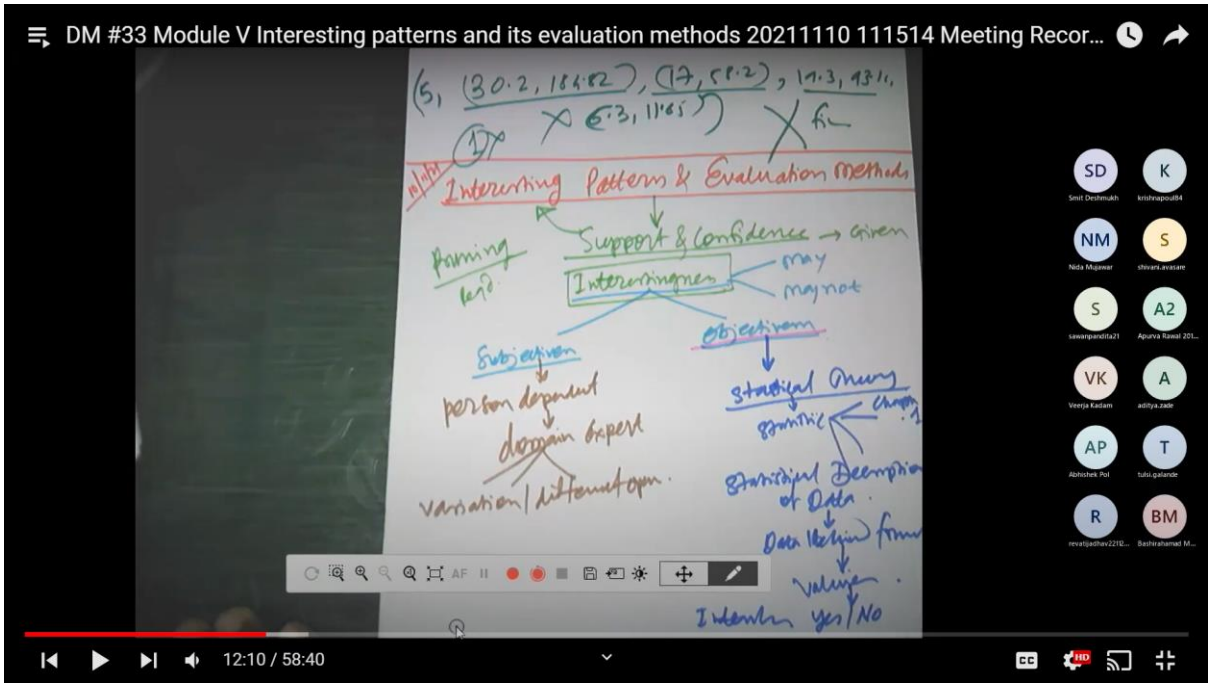
$R = 3^n - 2^{n+1}+1$

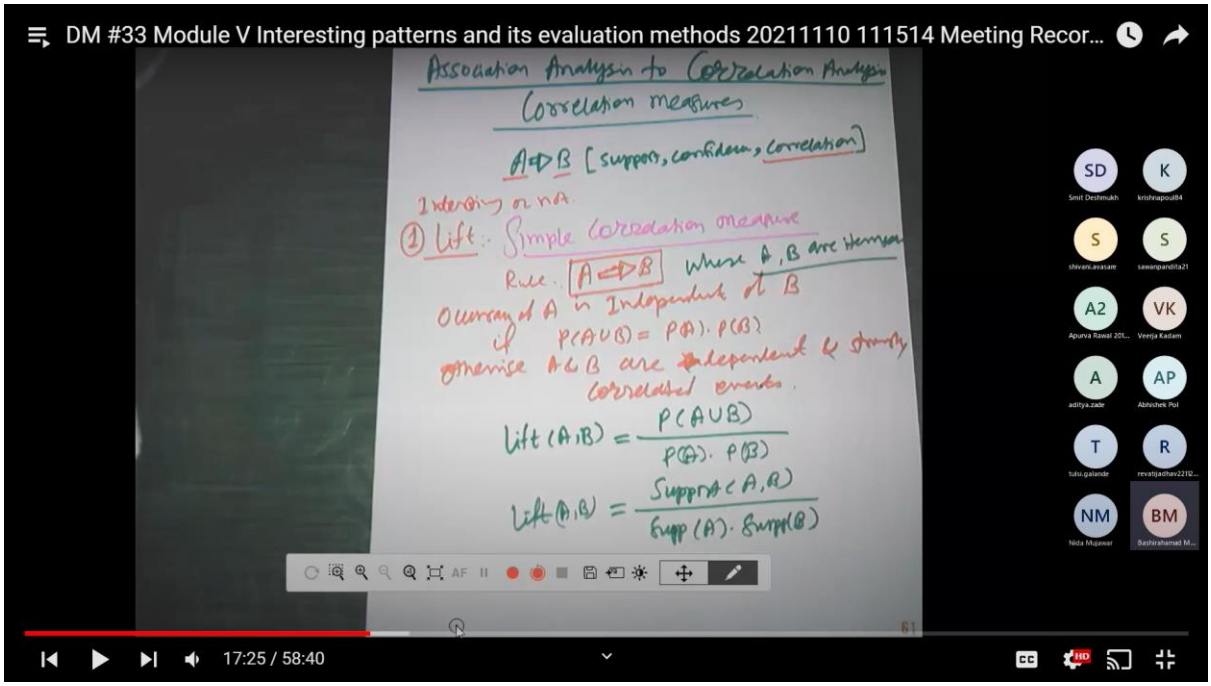
Huge no. of rules generated, hence built-in rule pruning in Apriori

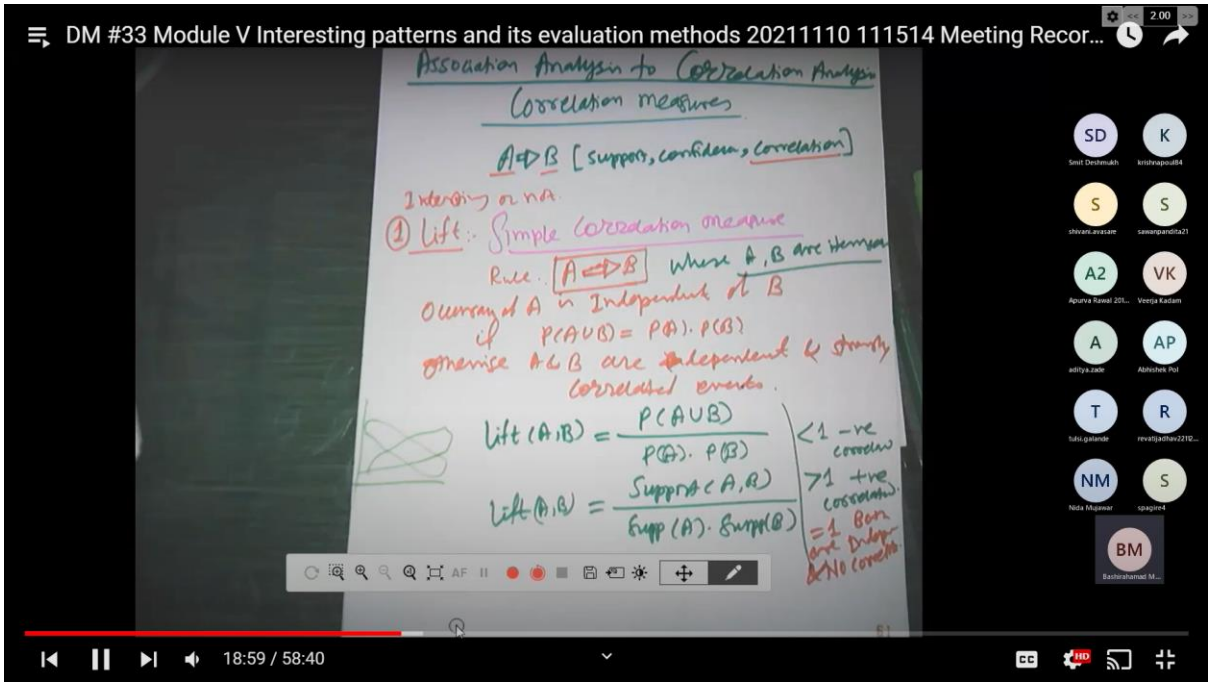
Generate the strong rules from strong frequent / large item set, satisfying *minimum support and confidence*

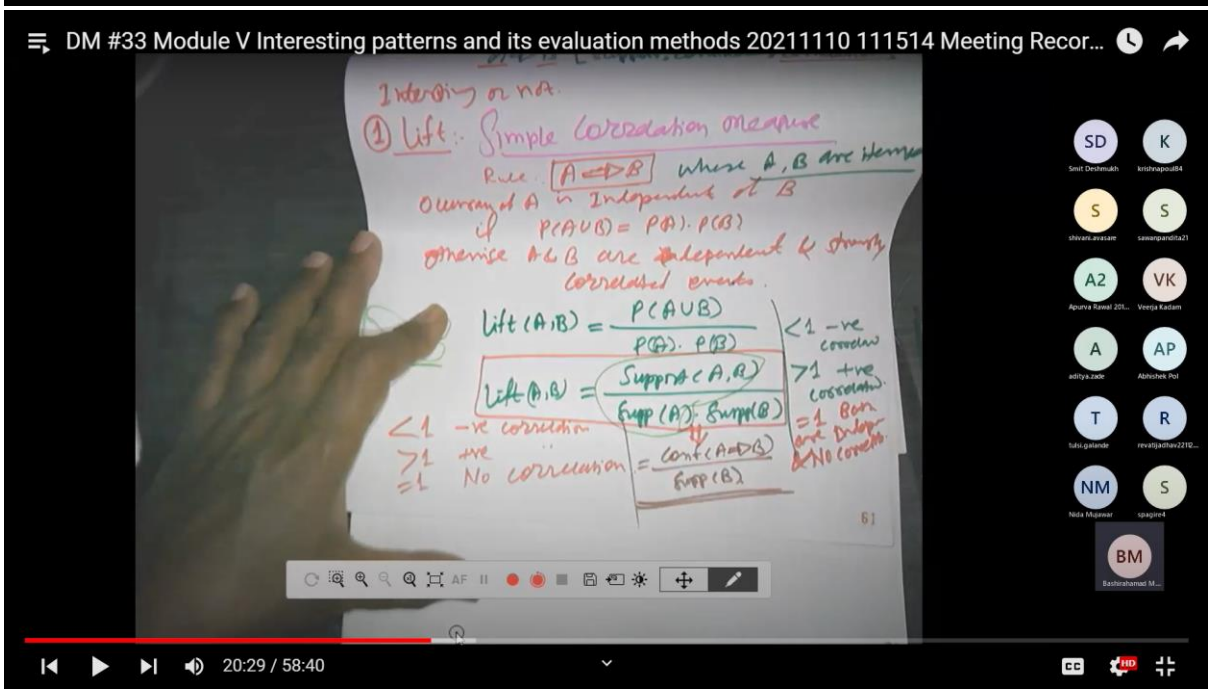
Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.

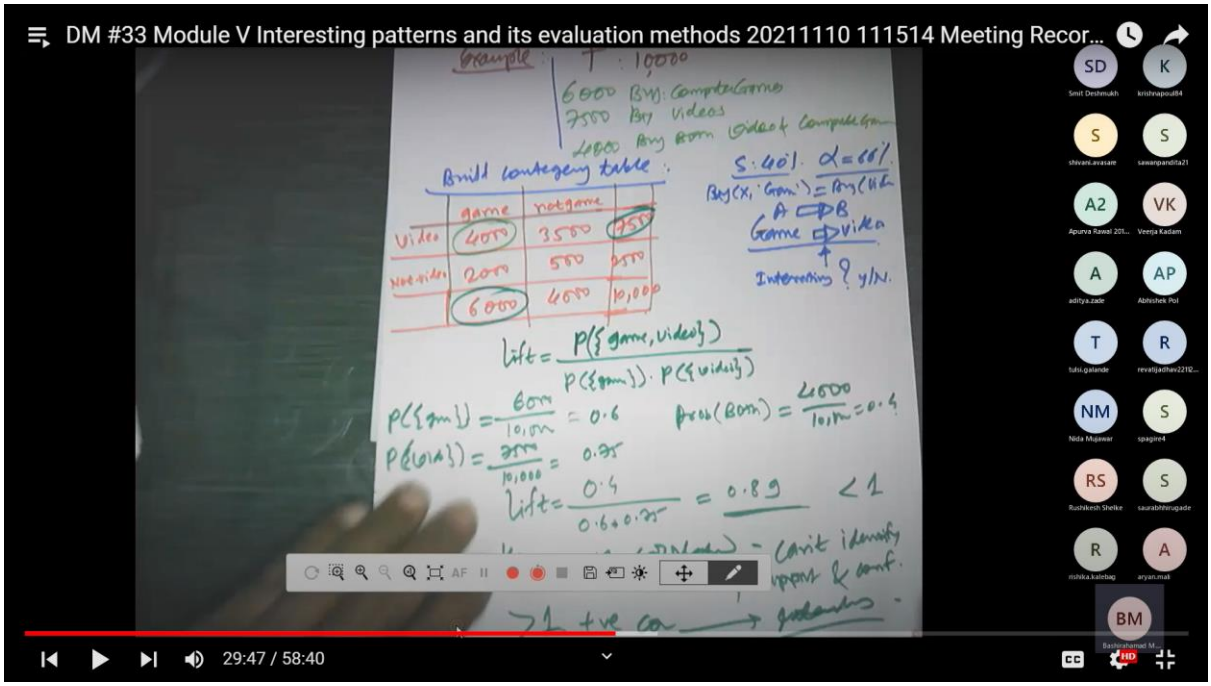
Lecture 33:

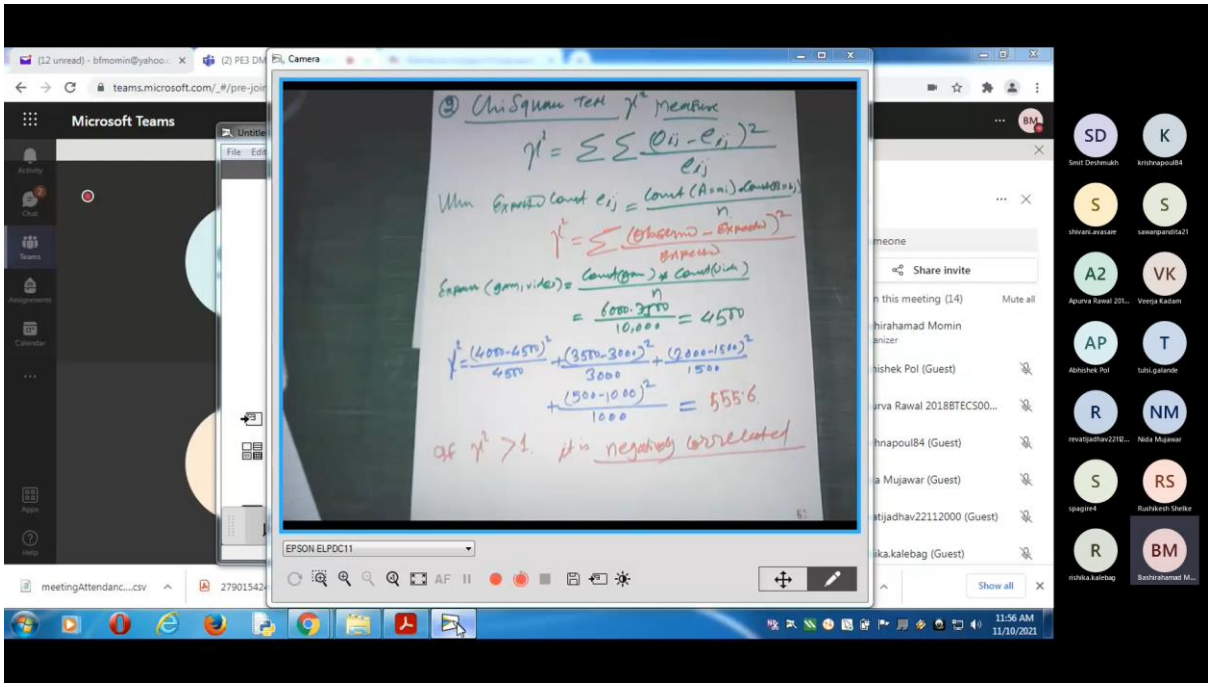

DM #33 Module V Interesting patterns and its evaluation methods 20211110 111514 Meeting Recor...
Interesting Patterns & Evaluation methods
Support & Confidence → Given
Interestingness
may
may not
Subjective
person dependent
domain expert
variation / different opn.
objectivism
statistical theory
Statistical Decision of Data
Data
Validity
Yes / No
12:10 / 58:40

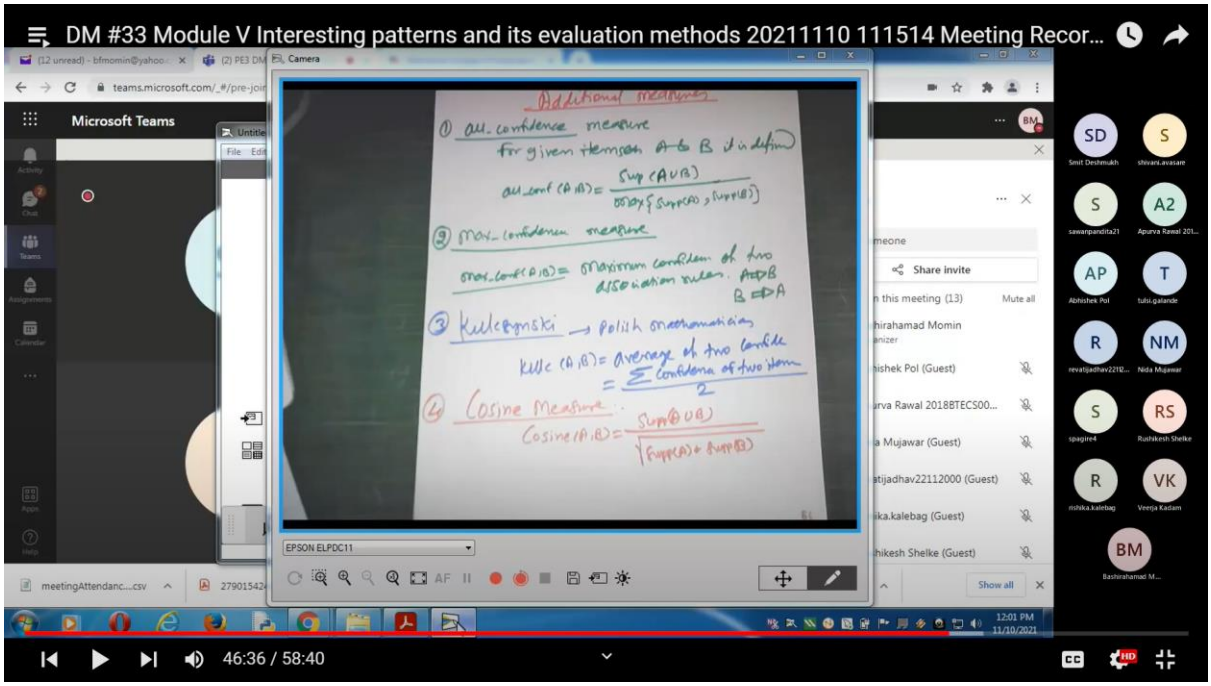

DM #33 Module V Interesting patterns and its evaluation methods 20211110 111514 Meeting Recor...
Association Analysis to Correlation Analysis
Correlation measures
A ⇒ B [support, confidence, correlation]
Interesting or not
1. Lift : Simple correlation measure
Rule A ⇒ B where A, B are Itemsets
Occurrence of A is Independent of B
if P(A ∪ B) = P(A) · P(B)
otherwise A & B are dependent & strongly correlated events.
Lift(A,B) = P(A ∪ B) / P(A) · P(B)
Lift(A,B) = Supp(A,B) / Supp(A) · Supp(B)
17:25 / 58:40

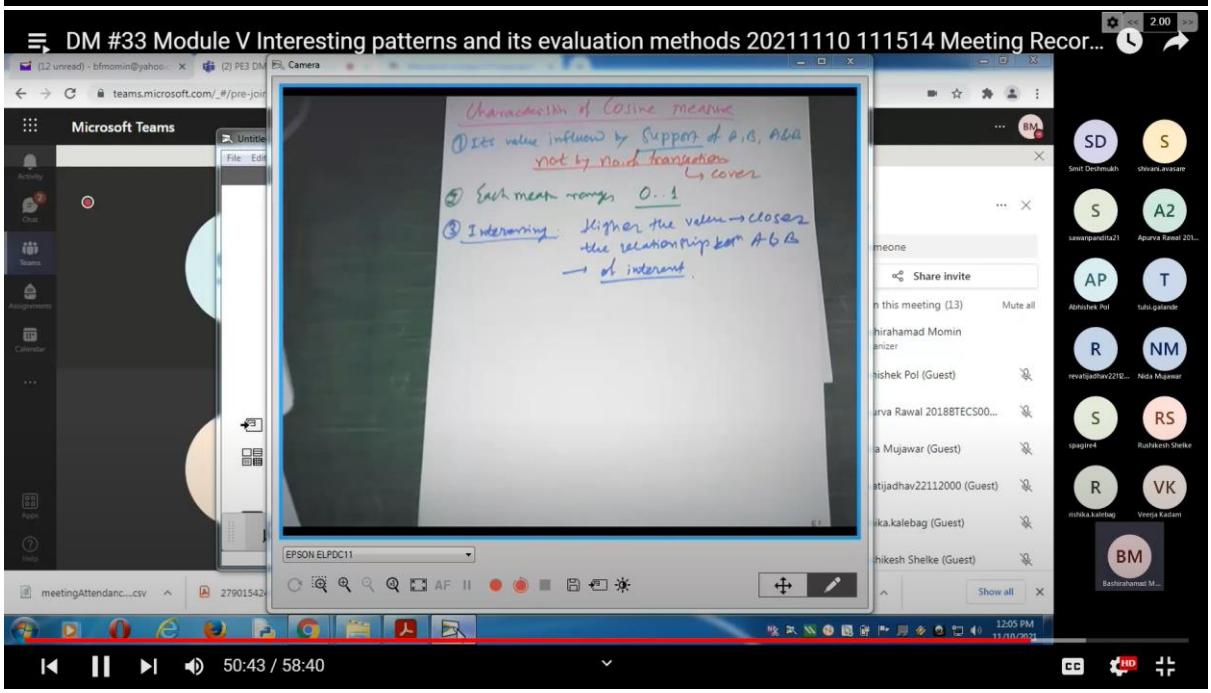

DM #33 Module V Interesting patterns and its evaluation methods 20211110 111514 Meeting Recor...
Association Analysis to Correlation Analysis
Correlation measures
A⇒B [support, confidence, correlation]
Interesting or not.
1) Lift: Simple correlation measure
Rule: A⇒B where A, B are itemsets
Occurrence of A is Independent of B
if P(A∪B) = P(A). P(B)
otherwise A & B are dependent & strongly correlated events.
Lift(A,B) = P(A∪B) / P(A). P(B)
<1 -ve correlation
>1 +ve correlation
=1 Both are Independent & No correlation
Lift(A,B) = Support(A,B) / Supp(A). Supp(B)
18:59 / 58:40


DM #33 Module V Interesting patterns and its evaluation methods 20211110 111514 Meeting Recor...
Interesting or not.
1) Lift: Simple correlation measure
Rule: A⇒B where A, B are itemsets
Occurrence of A is Independent of B
if P(A∪B) = P(A). P(B)
otherwise A & B are dependent & strongly correlated events.
Lift(A,B) = P(A∪B) / P(A). P(B)
<1 -ve correlation
>1 +ve correlation
=1 Both are Independent & No correlation
Lift(A,B) = Support(A,B) / Supp(A). Supp(B)
<1 -ve correlation
>1 +ve
=1 No correlation = Conf(A⇒B) / Supp(B)
20:29 / 58:40


DM #33 Module V Interesting patterns and its evaluation methods 20211110 111514 Meeting Recor...
Build contegency table
game
netgame
Video
6000
3500
9500
Non-video
2000
500
2500
8000
4000
10,000
Lift = P({game, video}) / P({game}) · P({video})
P({game}) = 6000/10,000 = 0.6
P({video}) = 3500/10,000 = 0.35
Lift = 0.4 / 0.6 · 0.35 = 0.89 < 1
Interesting ? y/N
29:47 / 58:40


Microsoft Teams
Camera
Chi Square Test χ² measure
χ² = Σ Σ (Oij − eij)² / eij
eij = count(A=ai) count(B=bj) / n
χ² = Σ (Observed − Expected)² / Expected
Expected(game, video) = count(game) * count(video) / n
= 6000 · 3500 / 10,000 = 4500
χ² = (4000−4500)²/4500 + (3500−3000)²/3000 + (2000−1500)²/1500 + (500−1000)²/1000 = 555.6
If χ² > 1 it is negatively correlated
EPSON ELPDC11
Share invite
meetingAttendanc....csv
11:56 AM
11/10/2021


DM #33 Module V Interesting patterns and its evaluation methods 20211110 111514 Meeting Recor...
Additional measures
1) all_confidence measure
for given itemset A & B it is defined
all_conf(A,B) = sup(A∪B) / max{sup(A), sup(B)}
2) max_confidence measure
max_conf(A,B) = maximum confidence of two association rules. A⇒B & B⇒A
3) Kulczynski → Polish mathematician
Kulc(A,B) = average of two confidence
= (confidence of two items) / 2
4) Cosine Measure
Cosine(A,B) = sup(A∪B) / √(sup(A) × sup(B))
Microsoft Teams
Share invite
46:36 / 58:40


DM #33 Module V Interesting patterns and its evaluation methods 20211110 111514 Meeting Recor...
2.00
Characteristics of Cosine measure
1) Its value influenced by support of A, B, A∪B
not by no. of transactions
↳ cover
2) Each measure ranges 0..1
3) Interesting: Higher the value → closer the relationship between A & B
→ of interest
Microsoft Teams
Share invite
50:43 / 58:40

Lecture 34

Web Data Mining

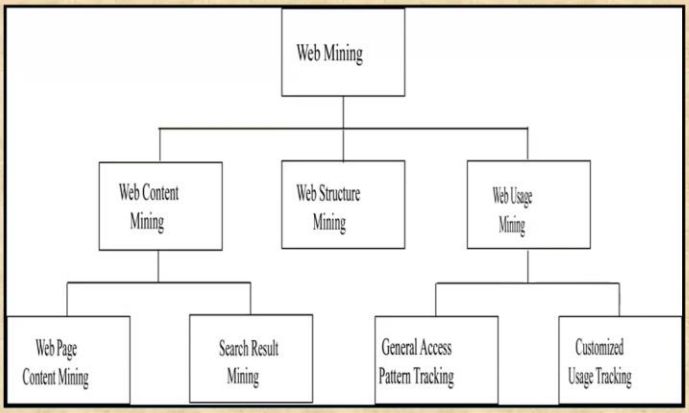
Use of data mining techniques to automatically discover interesting and potentially useful information from Web documents and services.

Web mining may be divided into three categories:

1. **Web content mining**
2. **Web structure mining**
3. **Web usage mining**

3

Web Mining Taxonomy



4

Web Data

- Web pages
- Intra-page structures
- Inter-page structures
- Usage data
- Supplemental data
  - Profiles
  - Registration information
  - Cookies

5

Web details

- More than 20 billion pages in 2008
- Many more documents in databases accessible from the Web
- More than 4m servers
- A total of perhaps 100 terabytes
- More than a million pages are added daily
- Several hundred gigabytes change every month
- Hyperlinks for navigation, endorsement, citation, criticism or plain whim

Web data is *BIG Data*

6

Web Index Size in Pages

Total Web is estimated to be about 56 B pages.

Search Engine	Number of Pages in Billions
Google	16
MSN Search	7
Yahoo	50
Ask	4.2

Source

http://www.worldwidewebsize.com/

7

Web Representation by Graph

- Web is a graph – vertices and edges (V,E)
- Directed graph – directed edges (p,q)
- Undirected graph - undirected edges (p,q)
- Strongly connected component - a set of nodes such that for any (u,v) there is a path from u to v

8

Graph terminology

- Breadth first search - layer 1 consists of all nodes that are pointed by the root, layer k consists of all nodes that are pointed by nodes on level k-1
- Diameter of a graph - maximum over all ordered pairs (u,v) of the shortest path from u to v

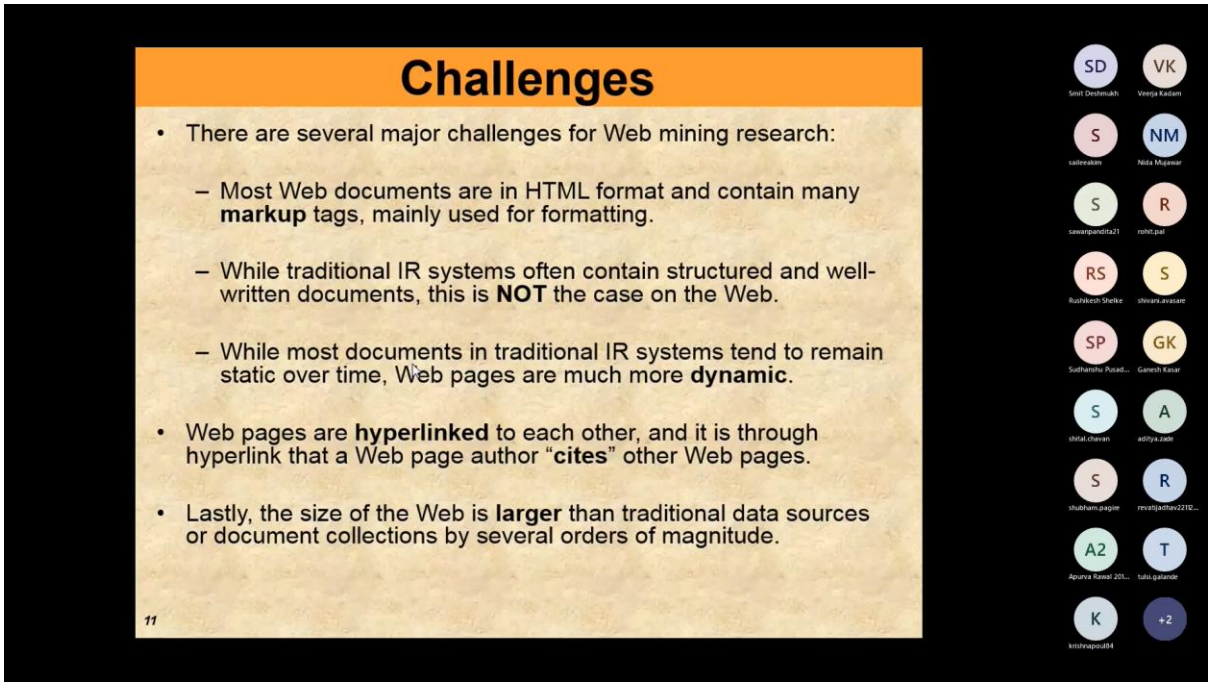
9

Web size

- In-degree is number of links to a node
- Out-degree is the number of links from a node
- Fraction of pages with i in-links is proportional to $1/i^{2.1}$
- With i out-links, it is $1/i^{2.72}$

i	In-links	Out-links
2	23%	15%
3	10%	5%
4	5%	2%
5	3%	1%

10


Challenges
• There are several major challenges for Web mining research:
– Most Web documents are in HTML format and contain many **markup** tags, mainly used for formatting.
– While traditional IR systems often contain structured and well-written documents, this is **NOT** the case on the Web.
– While most documents in traditional IR systems tend to remain static over time, Web pages are much more **dynamic**.
• Web pages are **hyperlinked** to each other, and it is through hyperlink that a Web page author "**cites**" other Web pages.
• Lastly, the size of the Web is **larger** than traditional data sources or document collections by several orders of magnitude.
11


DM #34 Module VI Web Mining Introduction 20211111 101726 Meeting Recording
Web Content Mining
12
32:47 / 55:53

Web Content mining

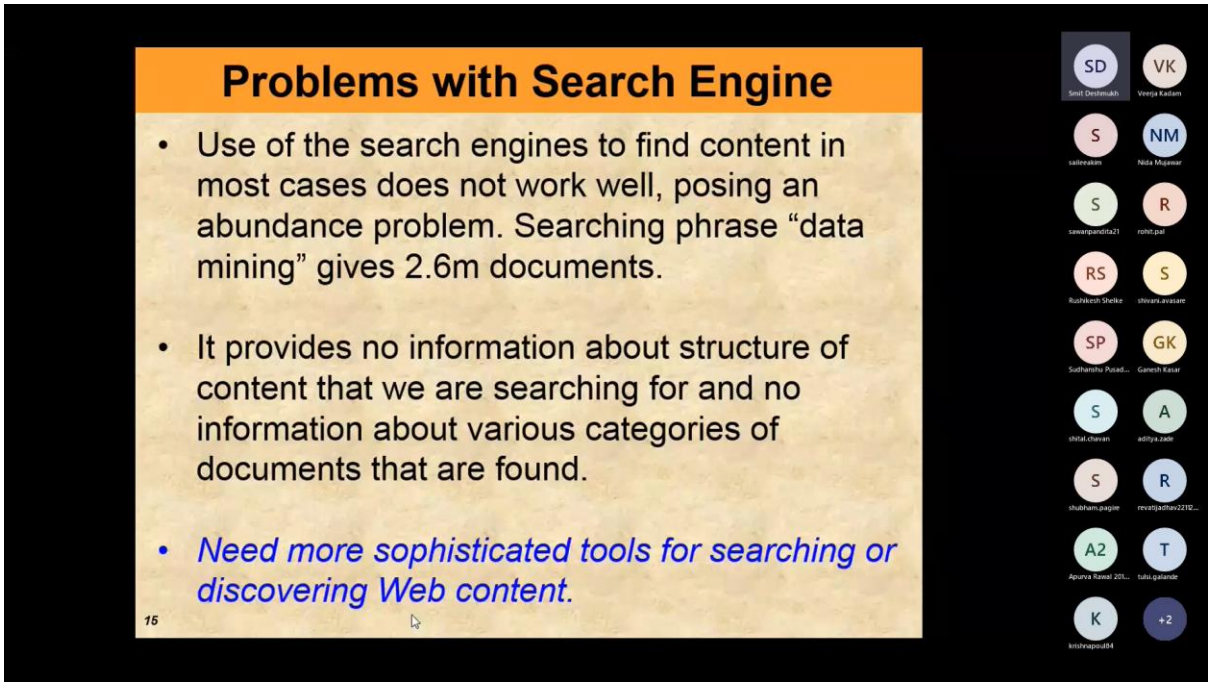
- Discovering useful information from contents of Web pages.
- Web content is very rich consisting of textual, image, audio, video etc and metadata as well as hyperlinks.
- The data may be unstructured (free text) or structured (data from a database) or semi-structured (html) although much of the Web is unstructured.

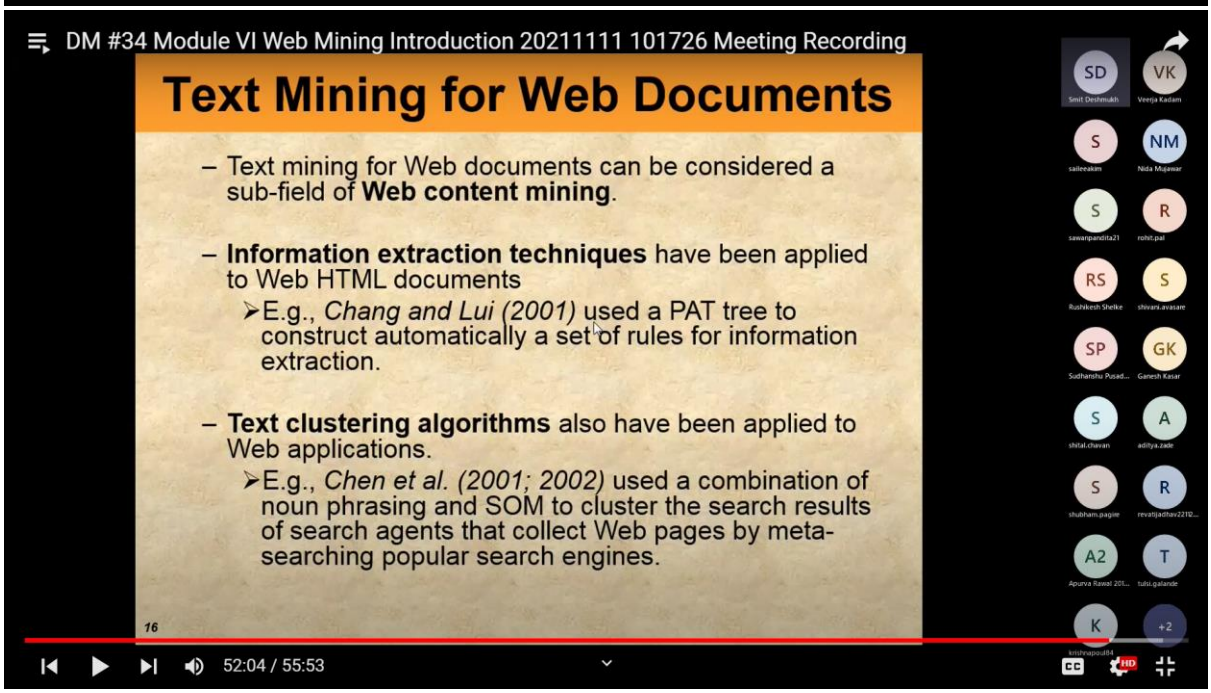
13

Web Content Mining

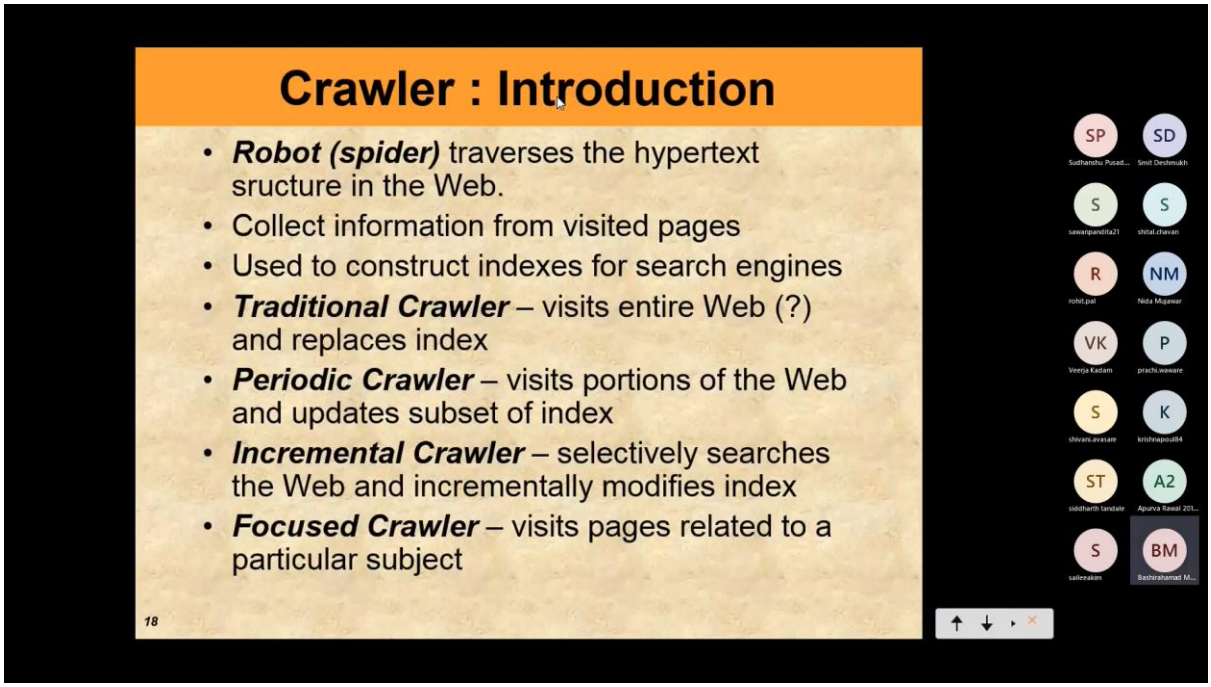
- Extends work of basic search engines
- Search Engines
  - IR application
  - Keyword based
  - Similarity between query and document
  - Crawlers
  - Indexing
  - Profiles
  - Link analysis
- **Two approaches** :
  - agent-based
    - » software agents perform the content mining
  - database oriented
    - » view the Web data as belonging to a database

14


Problems with Search Engine
• Use of the search engines to find content in most cases does not work well, posing an abundance problem. Searching phrase “data mining” gives 2.6m documents.
• It provides no information about structure of content that we are searching for and no information about various categories of documents that are found.
• Need more sophisticated tools for searching or discovering Web content.
15


DM #34 Module VI Web Mining Introduction 20211111 101726 Meeting Recording
Text Mining for Web Documents
– Text mining for Web documents can be considered a sub-field of **Web content mining**.
– **Information extraction techniques** have been applied to Web HTML documents
➢E.g., *Chang and Lui (2001)* used a PAT tree to construct automatically a set of rules for information extraction.
– **Text clustering algorithms** also have been applied to Web applications.
➢E.g., *Chen et al. (2001; 2002)* used a combination of noun phrasing and SOM to cluster the search results of search agents that collect Web pages by meta-searching popular search engines.
16
52:04 / 55:53

Lecture 35

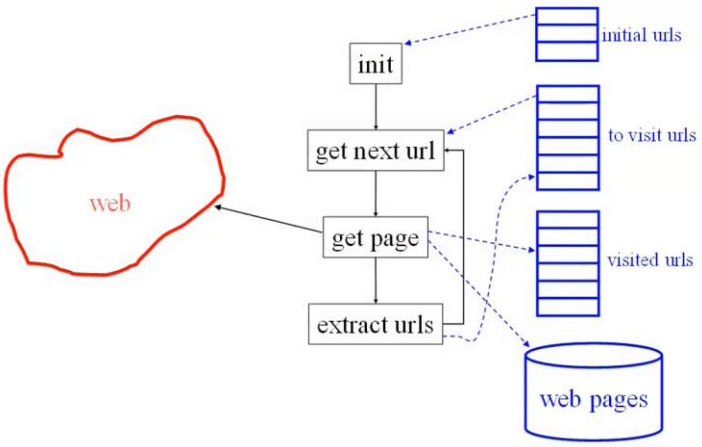

Crawler : Introduction
• Robot (spider) traverses the hypertext sructure in the Web.
• Collect information from visited pages
• Used to construct indexes for search engines
• Traditional Crawler – visits entire Web (?) and replaces index
• Periodic Crawler – visits portions of the Web and updates subset of index
• Incremental Crawler – selectively searches the Web and incrementally modifies index
• Focused Crawler – visits pages related to a particular subject
18
SP
SD
S
S
R
NM
VK
P
S
K
ST
A2
S
BM

How it works ?

- It starts with a list of URLs to visit, called the **seeds**.. As the crawler visits these URLs, it identifies all the hyperlinks in the page and adds them to the list of visited URLs, called the **crawl frontier**.
- URLs from the frontier are recursively visited according to a set of policies.

Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.

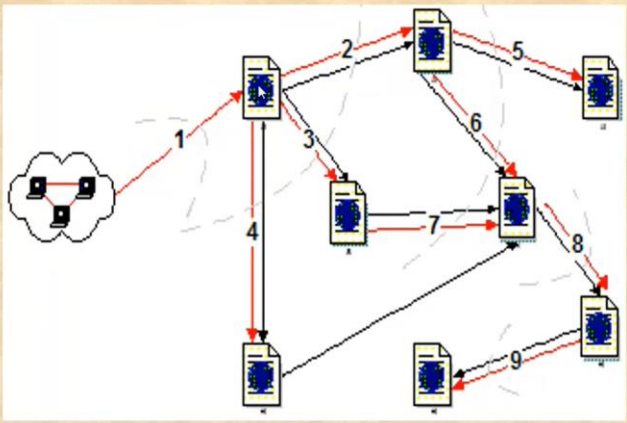
What is a Crawler?



Algorithms

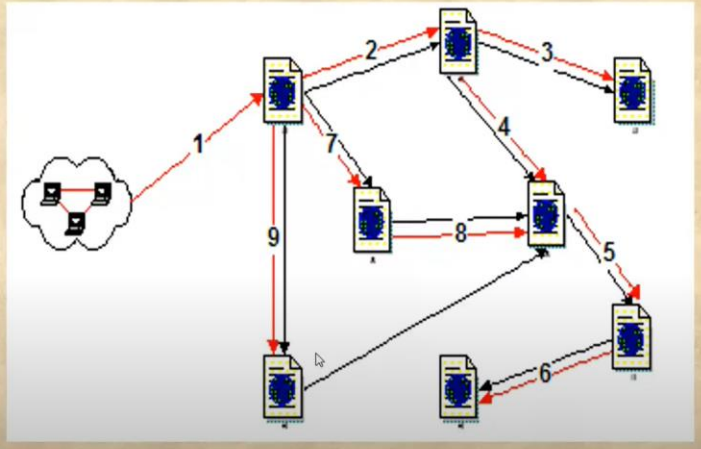
- Build the web graph
- Apply breadth first / depth first search algorithm

Breadth First Crawler



DM #35 Meeting in General 20211115 102328 Meeting Recording

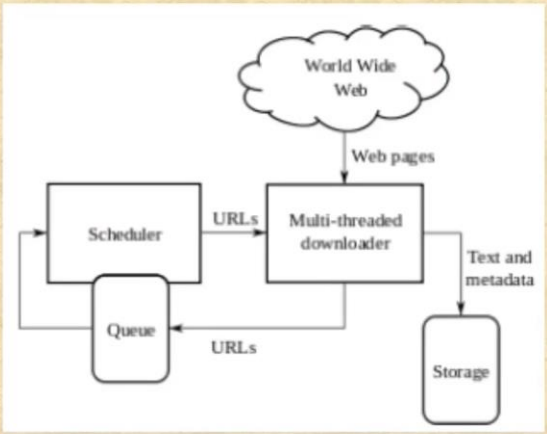
Depth First Crawler



Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.

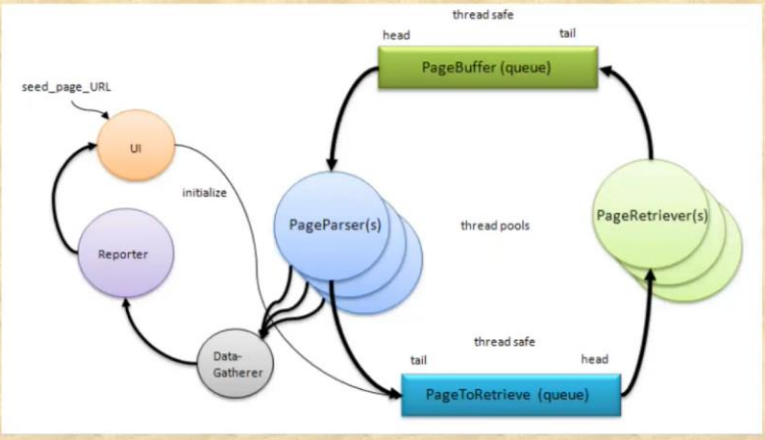
18:21 / 49:29

Multi-threaded Crawler



Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.

Multi-threaded Crawler



Dr. Bashirahamad F. Momin
CSE Dept., Walchand COE, Sangli.

Focused Crawler

- Only visit links from a page if that page is determined to be relevant.
- Classifier is static after learning phase.
- Components:
  - Classifier which assigns relevance score to each page based on crawl topic.
  - Distiller to identify *hub pages.*
  - Crawler visits pages to based on crawler and distiller scores.

DM #35 Meeting in General 20211115 102328 Meeting Recording

Focused Crawler

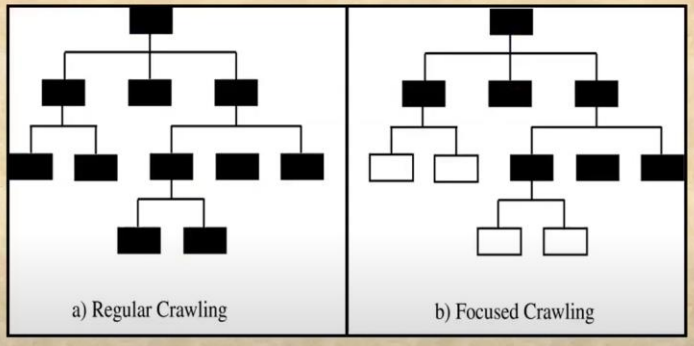
- Classifier to related documents to topics
- Classifier also determines how useful outgoing links are
- ***Hub Pages*** contain links to many relevant pages. Must be visited even if not high relevance score.

27

24:44 / 49:29

DM #35 Meeting in General 20211115 102328 Meeting Recording

Focused Crawler



a) Regular Crawling

b) Focused Crawling

28

24:46 / 49:29

DM #35 Meeting in General 20211115 102328 Meeting Recording

Context Focused Crawler

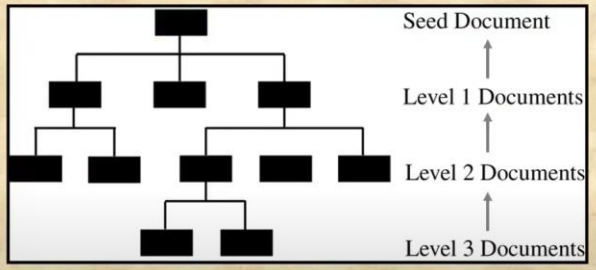
- Context Graph:
  - Context graph created for each seed document .
  - Root is the seed document.
  - Nodes at each level show documents with links to documents at next higher level.
  - Updated during crawl itself .
- Approach:
  1. Construct context graph and classifies using seed documents as training data.
  2. Perform crawling using classifiers and context graph created.

29

24:48 / 49:29

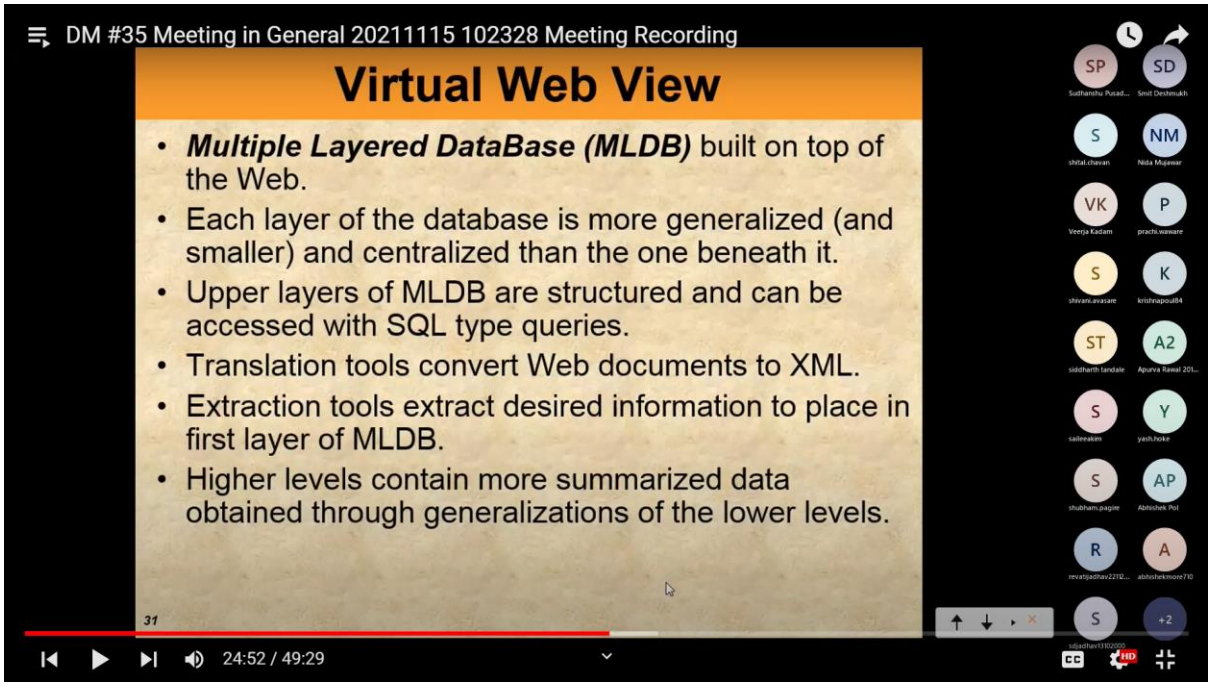
DM #35 Meeting in General 20211115 102328 Meeting Recording

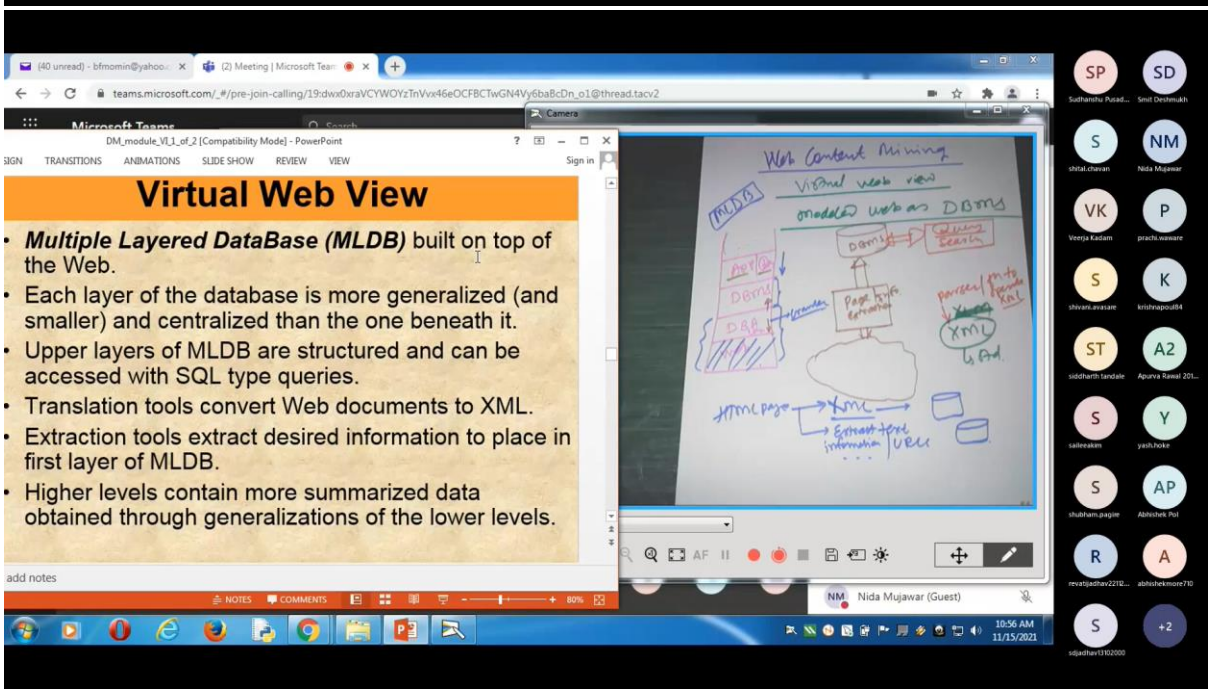
Context Graph



30

24:50 / 49:29



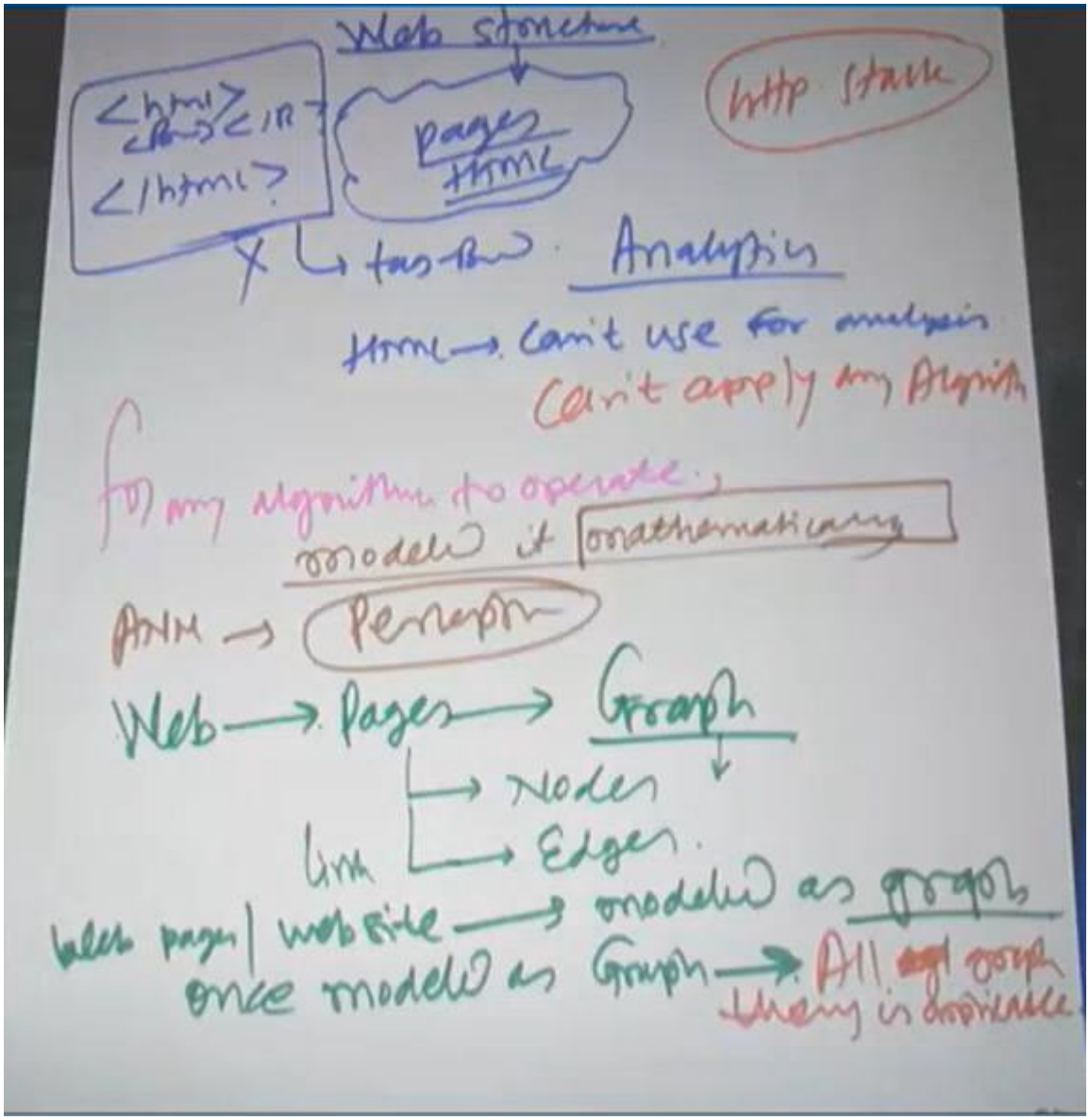


Personalization

- Web access or contents tuned to better fit the desires of each user.
- Manual techniques identify user's preferences based on profiles or demographics.
- ***Collaborative filtering*** identifies preferences based on ratings from similar users.
- ***Content based filtering*** retrieves pages based on similarity between pages and user profiles.

Web Structure Mining

- Mine structure (links, graph) of the Web
- Web as graph
  - Pages = nodes, hyperlinks = edges
- Techniques
  - PageRank
  - CLEVER *(modified as HITS)*
- Create a model of the Web organization.
- May be combined with content mining to more effectively retrieve important pages.


Web structure
<html>
</html>
pages
html
http stack
Analysis
Html → can't use for analysis
can't apply any Algorithm
for any algorithm to operate,
models it mathematically
ANN → Perceptron
Web → Pages → Graph
→ Nodes
Link → Edges.
Web page / website → modelled as graphs
once modelled as Graph → All algorithms
theory is available

Lecture 36:Not watched completely

CLEVER

- Identify authoritative and hub pages.
- ***Authoritative Pages*** :
  - Highly important pages.
  - Best source for requested information.
- ***Hub Pages*** :
  - Contain links to highly important pages.

37

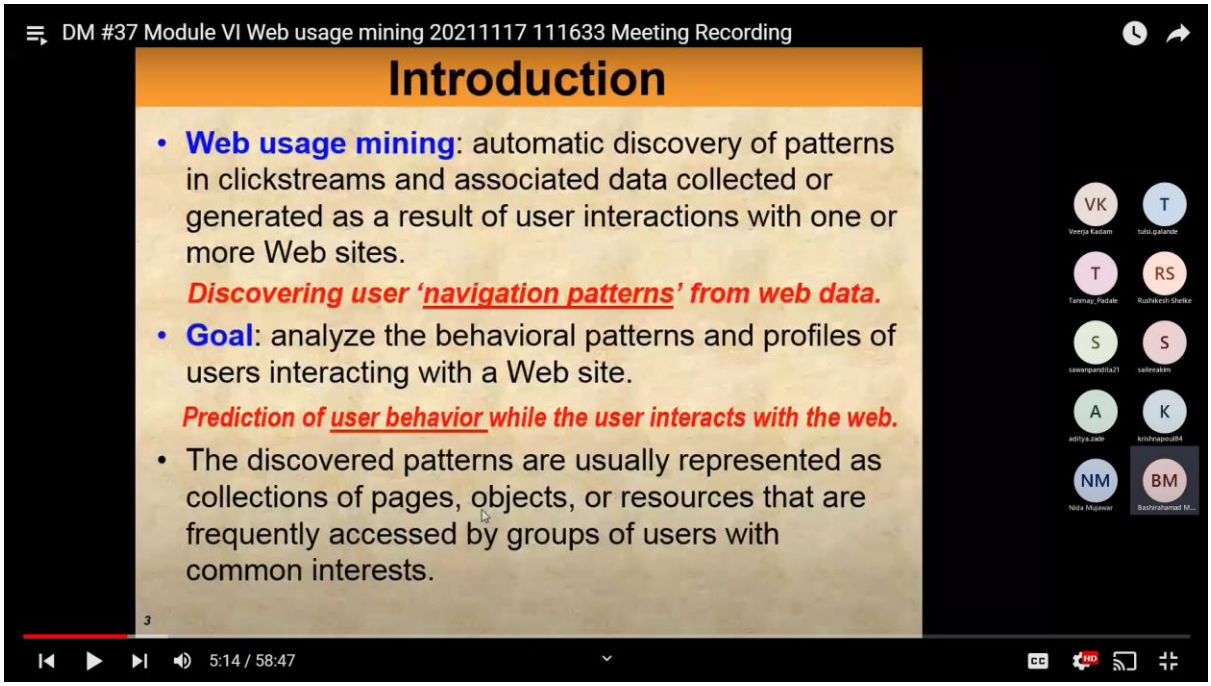
Hyperlink-Induces Topic Search (HITS)

- Based on a set of keywords, find set of relevant pages – R.
- Identify hub and authority pages for these.
  - Expand R to a base set, B, of pages linked to or from R.
  - Calculate weights for authorities and hubs.
- Pages with highest ranks in R are returned.

38

Lecture 37:


Web Usage Mining
2


DM #37 Module VI Web usage mining 20211117 111633 Meeting Recording
Introduction
• Web usage mining: automatic discovery of patterns in clickstreams and associated data collected or generated as a result of user interactions with one or more Web sites.
Discovering user 'navigation patterns' from web data.
• Goal: analyze the behavioral patterns and profiles of users interacting with a Web site.
Prediction of user behavior while the user interacts with the web.
• The discovered patterns are usually represented as collections of pages, objects, or resources that are frequently accessed by groups of users with common interests.
3
5:14 / 58:47

Web Usage Mining Applications

- Personalization
- Improve structure of a site's Web pages
- Aid in caching and prediction of future page references
- Improve design of individual pages
- Improve effectiveness of e-commerce (sales and advertising)

4

Sources of Data for Usage Mining

- Automatically generated data stored in server *access* logs, *referrer* logs, *agent* logs, and client-side *cookies*.
- user profiles.
- metadata: page attributes, content attributes, usage data.

5