Feb 2019 - Edition

#OneLastTime
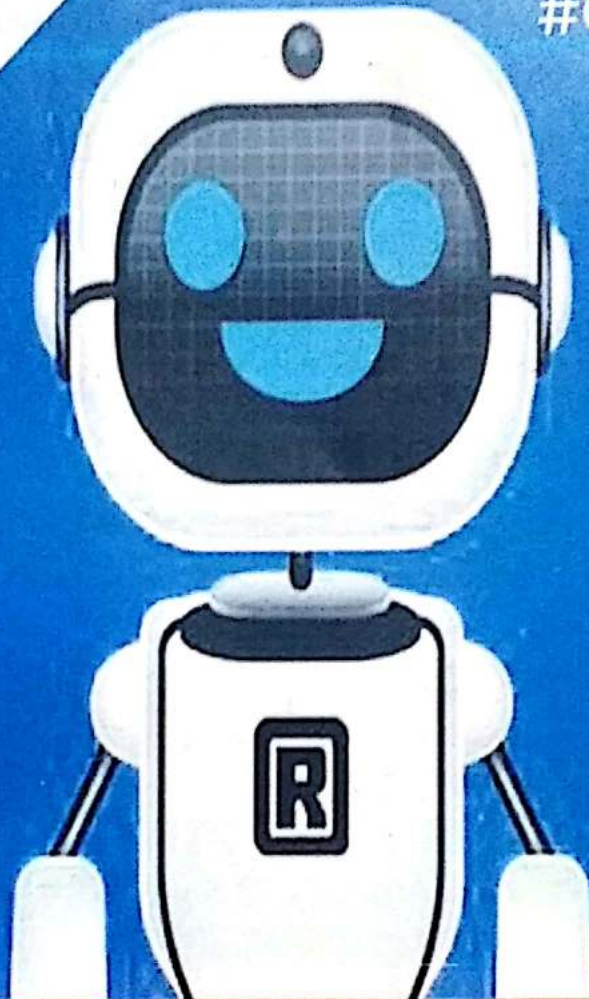
MACHINE LEARNIN

8 SEM

(BE - COMPUTER)

Handcrafted by BackkBenchers Publications

Machine Learning

Marks Distribution:

#	MAY - 16	DEC - 16	MAY - 17	DEC - 17	MAY – 18	DEC - 18
1.	25	15	25	15	25	15
2.	05	05	20	05	20	20
3.	10	20	10	10	15	15
4.	15	15	10	15	10	30
5.	30	30	20	20	10	05
6.	10	10	15	10	15	10
7.	20	15	10	30	25	20
8.	05	20	15	25	10	25
Repeated Marks	-	55	50	70	90	75

CHAP - 1: INTRODUCTION TO MACHINE LEARNING

Q1. What is Machine Learning? Explain how supervised learning is different from unsupervised learning.

Q2. Define Machine Learning? Briefly explain the types of learning.

Ans: [5M | May17 & May18]

MACHINE LEARNING:

1. Machine learning is an application of Artificial Intelligence (AI).
2. It provides ability to systems to automatically learn and improve from experience without being explicitly programmed.
3. Machine learning teaches computer to do what comes naturally to humans learns from experience
4. The primary goal of machine learning is to allow the systems learn automatically with human intervention or assistance and adjust actions accordingly.
5. Real life examples: Google search Engine, Amazon.
6. Machine learning is helpful in
    a. Improving business decision.
    b. Increase productivity.
    c. Detect disease.
    d. Forecast weather.

TYPES OF LEARNING:

I) Supervised Learning:

1. Supervised learning as the name indicates a presence of supervisor as teacher.
2. Basically supervised learning is a learning in which we teach or train the machine using data which is well labelled.
3. After that, machine is provided with new set of examples(data) so that supervised learning algorithm analyses the training data
4. Machine and produces a correct outcome from labelled data.
5. Supervised learning classified into two categories of algorithms:
    a. Classification.
    b. Regression.

II) Unsupervised learning:

1. Unlike supervised learning, no teacher is provided that means no training will be given to the machine.
2. Unsupervised learning is the training of machine using information that is neither classified nor labelled.
3. It allows the algorithm to act on that information without guidance.
4. Unsupervised learning classified into two categories of algorithms:
    a. Clustering
    b. Association

DIFFERENCE BETWEEN SUPERVISED AND UNSUPERVISED LEARNING:

Supervised learning	Unsupervised learning
Uses known and labelled data.	Uses unknown and unlabelled data.
Very complex to develop.	Less complex to develop.
Uses off-line analysis.	Uses real-time analysis.
Number of classes are known.	Number of classes are unknown.
Gives accurate and reliable results.	Gives moderate accurate and reliable results.

Q3.   Applications of Machine Learning algorithms

Q4.   Machine learning applications

Ans: [10M | May16, Dec16 & May17]

APPLICATIONS:

I)   **Virtual Personal Assistants:**

1.   Siri, Alexa, Google Now are some of the popular examples of virtual personal assistants.

2.   As the name suggests, they assist in finding information, when asked over voice.

3.   All you need to do is activate them and ask "What is my schedule for today?", "What are the flights from Germany to London", or similar questions.

II)   **Image Recognition:**

1.   It is one of the most common machine learning applications.

2.   There are many situations where you can classify the object as a digital image.

3.   For digital images, the measurements describe the outputs of each pixel in the image.

4.   In the case of a black and white image, the intensity of each pixel serves as one measurement.

III)   **Speech Recognition:**

1.   Speech recognition (SR) is the translation of spoken words into text.

2.   In speech recognition, a software application recognizes spoken words.

3.   The measurements in this Machine Learning application might be a set of numbers that represent the speech signal.

4.   We can segment the signal into portions that contain distinct words or phonemes.

5.   In each segment, we can represent the speech signal by the intensities or energy in different time-frequency bands.

IV)   **Medical Diagnosis:**

1.   ML provides methods, techniques, and tools that can help in solving diagnostic and prognostic problems in a variety of medical domains.

2.   It is being used for the analysis of the importance of clinical parameters and of their combinations for prognosis.

3.   E.g. prediction of disease progression, for the extraction of medical knowledge for outcomes research.

V)   **Search Engine Result Refining:**

1.   Google and other search engines use machine learning to improve the search results for you.

---

2. Every time you execute a search, the algorithms at the backend keep a watch at how you respond to the results.

VI) Statistical Arbitrage:

1. In finance, statistical arbitrage refers to automated short-term trading strategies that involve a large number of securities.

2. In such strategies, the user tries to implement a trading algorithm for a set of securities on the basis of quantities.

3. These measurements can be cast as a classification or estimation problem.

VII) Learning Associations:

1. Learning association is the process of developing insights into various associations between products.

2. A good example is how seemingly unrelated products may reveal an association to one another.

3. When analyzed in relation to buying behaviors of customers.

VIII) Classification:

1. Classification is a process of placing each individual from the population under study in many classes.

2. This is identified as independent variables.

3. Classification helps analysts to use measurements of an object to identify the category to which that object belongs.

4. To establish an efficient rule, analysts use data.

IX) Prediction:

1. Consider the example of a bank computing the probability of any of loan applicants faulting the loan repayment.

2. To compute the probability of the fault, the system will first need to classify the available data in certain groups.

3. It is described by a set of rules prescribed by the analysts.

4. Once we do the classification, as per need we can compute the probability.

X) Extraction:

1. Information Extraction (IE) is another application of machine learning.

2. It is the process of extracting structured information from unstructured data.

3. For example web pages, articles, blogs, business reports, and e-mails.

4. The process of extraction takes input as a set of documents and produces a structured data.

Q5.   Explain the steps required for selecting the right machine learning algorithm

Ans: [10M | May16 & Dec17]

STEPS:

I) Understand Your Data:

1. The type and kind of data we have plays a key role in deciding which algorithm to use.

2. Some algorithms can work with smaller sample sets while others require tons and tons of samples.

3. Certain algorithms work with certain types of data.

4. E.g. Naive Bayes works well with categorical input but is not at all sensitive to missing data

5. It includes:

   a. **Know your data:**

      o Look at Summary statistics and visualizations.

      o Percentiles can help identify the range for most of the data.

      o Averages and medians can describe central tendency.

   b. **Visualize the data:**

      o Box plots can identify outliers.

      o Density plots and histograms show the spread of data.

      o Scatter plots can describe bivariate relationships.

   c. **Clean your data:**

      o Deal with missing value

      o Missing data affects some models more than others.

      o Missing data for certain variables can result in poor predictions.

   d. **Augment your data:**

      o Feature engineering is the process of going from raw data to data that is ready for modelling It can serve multiple purposes:

      o Different models may have different feature engineering requirements.

      o Some have built in feature engineering.

II) Categorize the problem:

This is a two-step process.

1. Categorize by input:

   a. If you have labelled data, it's a supervised learning problem.

   b. If you have unlabeled data and want to find structure, it's an unsupervised learning problem.

   c. If you want to optimize an objective function by interacting with an environment, it's a reinforcement learning problem.

2. Categorize by output:

   a. If the output of your model is a number, it's a regression problem.

   b. If the output of your model is a class, it's a classification problem.

   c. If the output of your model is a set of input groups, it's a clustering problem.

   d. Do you want to detect an anomaly? That's anomaly detection.

III) Understand your constraints:

1. What is your data storage capacity?

   a. Depending on the storage capacity of your system, you might not be able to store gigabytes of classification/regression models or gigabytes of data to cluster.

2. Does the prediction have to be fast? In real time applications:

   a. For example, in autonomous driving, it's important that the classification of road signs be as fast as possible to avoid accidents.

3. **Does the learning have to be fast?**

   a. In some circumstances, training models quickly is necessary; sometimes, you need to rapidly update, on the fly, your model with a different dataset.

IV) Find the available algorithms:

1. Some of the factors affecting the choice of a model are:
2. Whether the model meets the business goals.
3. How much pre-processing the model needs
4. How accurate the model is.
5. How explainable the model is.
6. How fast the model is.
7. How scalable the model is.

V) Try each algorithm, access and compare.

VI) Adjust and combine, optimization techniques.

VII) Choose, operate and continuously measure.

VIII) Repeat.

Q6. What are the steps in designing a machine learning problem? Explain with the checkers problem.

Q7. Explain the steps in developing a machine learning application.

Q8. Explain procedure to design machine learning system.

Ans: [10M | May17, May18 & Dec18]

STEPS FOR DEVELOPING ML APPLICATIONS:

I) Gathering data:

1. This step is very important because the quality of data that you gather will directly determine how good your predictive model will be.
2. We have to collect data from different sources for our ML application training purpose.
3. This includes collecting samples by scraping a website and extracting data from an RSS feed or an API.

II) Preparing the data:

1. Data preparation is where we load our data into a suitable place and prepare it for use in our system for training.
2. The benefit of having this standard format is that you use can mix and matching algorithms and data sources.

III) Choosing a model:

1. There are many models that the data scientists and researcher have created over years

2. Some of them are well suited for image data, other for sequence and some for numerical data.

3. It involves recognizing patterns, identifying outliers and detection of novelty.

IV) Training:

1. In this step, we will use our data to incrementally improve our models ability to predict the data we have inserted.

2. Depending on the algorithm, feed the algorithm good clean data from previous steps and extra knowledge or information.

3. The knowledge extracted is stored in a format that is readily usable by a machine for next steps.

V) Evaluation:

1. Once the training is complete, it's time to check if the model is good for using evaluation.

2. This is where testing datasets comes into play.

3. Evaluation allows us to test our model against data that has never been used for training.

VI) Parameter tuning:

1. Once we are done with evaluation, we want to see if we can further improve our training in any way.

2. We can do this by tuning our parameters.

VII) Prediction:

1. It is a step where we get to answer for some questions.

2. It is the point where the value of machine learning is realized.

CHECKER LEARNING PROBLEM:

1. A computer program that learns to play checkers might improve its performance as measured by its ability to win at the class of tasks involving playing checkers games, through experience obtained by playing games against itself.

2. Choosing a training experience:

   a. The type of training experience 'E' available to a system can have significant impact on success or failure of the learning system.

   b. One key attribute is whether the training experience provides direct or indirect feedback regarding the choices made by the performance system.

   c. Second attribute is the degree to which the learner controls the sequence of training examples.

   d. Another attribute is the degree to which the learner controls the sequence of training examples.

3. **Assumptions:**

   a. Let us assume that our system will train by playing games against itself.

   b. And it is allowed to generate as much training data as time permits.

4. **Issues Related to Experience:**

   a. What type of knowledge/experience should one learn?

   b. How to represent the experience?

   c. What should be the learning mechanism?

5. **Target Function:**

   a. Choose Move: B → M

   b. Choose Move is a function.

   c. where input B is the set of legal board states and produces M which is the set of legal moves.

   d. M = Choose Move (B)

6. **Representation of Target Function:**

   a. x_1 : the number of white pieces on the board.

   b. x_2 : the number of red pieces on the board.

   c. x_3 : the number of white kings on the board.

   d. x_4 : the number of red kings on the board.

   e. x_5 : the number of white pieces threatened by red (i.e., which can be captured on red's next turn)

   f. x_6: the number of red pieces threatened by white.

   g. $F'(b) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4 + w_5 x_5 + w_6 x_6$

7. The problem of learning a checkers strategy reduces to the problem of learning values for the coefficients w_0 through w_6 in the target function representation.

Q9.   What are the key tasks of Machine Learning

Ans: **[5M | May16 & Dec17]**

CLASSIFICATION:

1. If we have data, say pictures of animals, we can classify them.

2. This animal is a cat, that animal is a dog and so on.

3. A computer can do the same task using a Machine Learning algorithm that's designed for the classification task.

4. In the real world, this is used for tasks like voice classification and object detection.

5. This is a supervised learning task, we give training data to teach the algorithm the classes they belong to.

REGRESSION:

1. Sometimes you want to predict values.

2. What are the sales next month? And what is the salary for a job?

3. Those type of problems are regression problems.

4. The aim is to predict the value of a continuous response variable.

5. This is also a supervised learning task.

CLUSTERING:

1. Clustering is to create groups of data called clusters.

2. Observations are assigned to a group based on the algorithm.

3. This is an unsupervised learning task, clustering happens fully automatically.

4. Imagining have a bunch of documents on your computer, the computer will organize them in clusters based on their content automatically.

FEATURE SELECTION:

1. This task is important because selecting right features would not only help to build models of higher accuracy

2. It also helps in achieving objectives related to building similar models.

3. It also helps in reducing over fitting leads

4. Techniques used for feature selection: filter method, wrapper method

TESTING AND MATCHING:

1. This task related to comparing the data sets

2. Testing and matching methods: Minimum spanning trees, II point correlation

---

Q10. What are the issues in Machine Learning?

Ans: [5M | Dec16 & May18]

ISSUES:

1. In what settings will particular algorithms converge to the desired function, given sufficient training data?

2. What algorithms exist for learning general target functions from specific training examples?

3. Which algorithms perform best for which types of problems and representations?

4. How much training data is sufficient?

5. What is the best way to reduce the learning task to one or more function approximation problems?

6. When and how can prior knowledge held by the learner guide the process of generalizing from examples?

7. Can prior knowledge be helpful even when it is only approximately correct?

8. What is the best strategy for choosing a useful next training experience, and how does the choice of this strategy alter the complexity of the learning problem?

9. How can the learner automatically alter its representation to improve its ability to represent and learn the target function?

Q11. Define well posed learning problem. Hence, define robot driving learning problem.

Ans: [5M | Dec18]

WELL-POSED LEARNING PROBLEMS:

1. A computer program is said to learn from experience 'E' with respect to some class of tasks 'T' and performance measure 'P', if its performance at tasks 'T', as measured by 'P', improves with experience 'E'.

2. It identifies following three features
   a. Class of tasks.
   b. Measure of performance to be improved.
   c. Source of experience.

3. **Examples:**

    a. Learning to classify chemical compounds.

    b. Learning to drive an autonomous vehicle.

    c. Learning to play bridge.

    d. Learning to parse natural language sentences.

ROBOT DRIVING PROBLEM:

1. **Task (T):** Driving on public, 4-lane highway using vision sensors.

2. **Performance measure (P):** Average distance travelled before an error (as judged by human overseer)

3. **Training experience (E):** A sequence of images and steering commands recorded while observing a human driver.

CHAP - 2: LEARNING WITH REGRESSION

Q1. Logistic Regression

Ans:

LOGISTIC REGRESSION:

1. Logistic Regression is one of the basic and popular algorithm to solve a classification problem.

2. It is the go-to method for binary classification problems (problems with two class values).

3. Linear regression algorithms are used to predict/forecast values but logistic regression is used for classification tasks.

4. The term "Logistic" is taken from the Logit function that is used in this method of classification.

5. The logistic function, also called the sigmoid function describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment.

6. It's an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

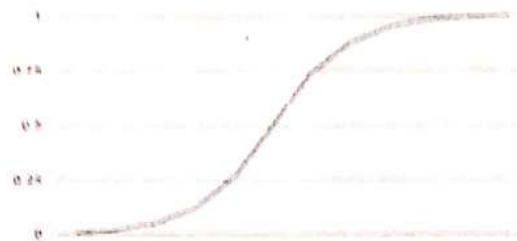7. Figure 2.1 shows the example of logistic function.

Figure 2.1: Logistic Function

SIGMOID FUNCTION (LOGISTIC FUNCTION):

1. Logistic regression algorithm also uses a linear equation with independent predictors to predict a value.

2. The predicted value can be anywhere between negative infinity to positive infinity.

3. We need the output of the algorithm to be class variable I.e. 0-no, 1-yes.

4. Therefore, we are squashing the output of the linear equation into a range of [0, 1].

5. To squash the predicted value between 0 and 1, we use the sigmoid function.

Linear Equation:

$$z = \theta_0 + \theta_1 \cdot x_1 + \theta \cdot x_2 + \cdots$$

Sigmoid Function:

$$g(x) = \frac{1}{1 + e^{-x}}$$

Squashed Output –h:

$$h = g(z) = \frac{1}{1 + e^{-z}}$$

COST FUNCTION:

1. Since we are trying to predict class values, we cannot use the same cost function used in linear regression algorithm.

2. Therefore, we use a logarithmic loss function to calculate the cost for misclassifying.

$$\text{Cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$

CALCULATING GRADIENTS:

1. We take partial derivatives of the cost function with respect to each parameter (theta_0, theta_1, ...) to obtain the gradients.

2. With the help of these gradients, we can update the values of theta_0, theta_1, etc.

3. It does assume linear relationship between the logit of the explanatory variables and the response.

4. Independent variables can be even the power terms or some other nonlinear transformations of the original independent variables.

5. The dependent variable does NOT need to be normally distributed, but it typically assumes a distribution from an exponential family (e.g. binomial, Poisson, multinomial, normal); binary logistic regression assume binomial distribution of the response.

6. The homogeneity of variance does NOT need to be satisfied.

7. Errors need to be independent but NOT normally distributed.

Q2. Explain in brief Linear Regression Technique.

Q3. Explain the concepts behind Linear Regression.

Ans: [5M | May16 & Dec17]

LINEAR REGRESSION:

1. Linear Regression is a machine learning algorithm based on supervised learning.

2. It is a simple machine learning model for regression problems, i.e., when the target variable is a real value.

3. It is used to predict a quantitative response y from the predictor variable x.

4. It is made with an assumption that there's a linear relationship between x and y.

5. This method is mostly used for forecasting and finding out cause and effect relationship between variables.

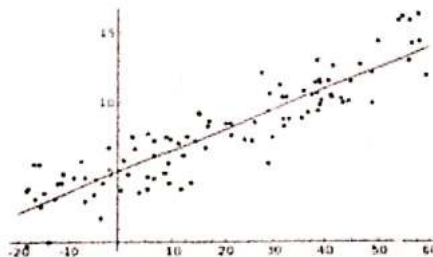6. Figure 2.2 shows the example of linear regression.

Figure 2.2: Linear Regression.

7. The red line in the above graph is referred to as the best fit straight line.

8. Based on the given data points, we try to plot a line that models the points the best.

9. For example, in a simple regression problem (a single x and a single y), the form of the model would be:

$$y = a_0 + a_1{}^* x \dots\dots\dots \text{(Linear Equation)}$$

---

10. The motive of the linear regression algorithm is to find the best values for a0 and a1.

COST FUNCTIONS:

1. The cost function helps us to figure out the best possible values for a0 and a1 which would provide the best fit line for the data points.

2. We convert this search problem into a minimization problem where we would like to minimize the error between the predicted value and the actual value.

3. Minimization and cost function is given below:

$$minimize \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

$$J = \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

4. Cost function(J) of Linear Regression is the Root Mean Squared Error (RMSE) between predicted y value and true y value.

GRADIENT DESCENT:

1. To update a_0 and a_1 values in order to reduce Cost function and achieving the best fit line the model uses Gradient Descent.

2. The idea is to start with random a_0 and a_1 values and then iteratively updating the values, reaching minimum cost.

Q4.   Explain Regression line, Scatter plot, Error in prediction and best fitting line

Ans:

[5M | Dec16]

REGRESSION LINE:

1. The Regression Line is the line that best fits the data, such that the overall distance from the line to the points (variable values) plotted on a graph is the smallest.

2. There are as many numbers of regression lines as variables.

3. Suppose we take two variables, say X and Y, then there will be two regression lines:

4. Regression line of Y on X: This gives the most probable values of Y from the given values of X.
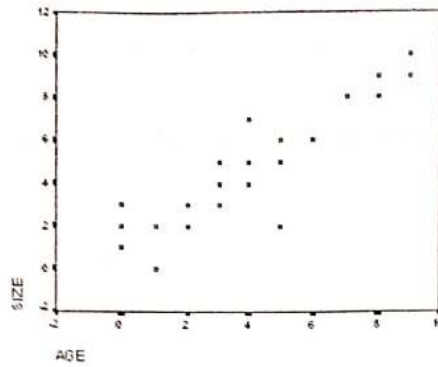
$$Y_e = a + bX$$

5. Regression line of X on Y: This gives the most probable values of X from the given values of Y.

$$X_e = a + bY$$

SCATTER DIAGRAMS:

1. If data is given in pairs then the scatter diagram of the data is just the points plotted on the xy-plane.

2. The scatter plot is used to visually identify relationships between the first and the second entries of paired data.

---

3. **Example:**

4. The scatter plot above represents the age vs. size of a plant.

5. It is clear from the scatter plot that as the plant ages, its size tends to increase.

ERROR IN PREDICTION:

1. The standard error of the estimate is a measure of the accuracy of predictions.

2. The standard error of the estimate is closely related to this quantity and is defined below:

$$\sigma_{est} = \sqrt{\frac{\sum (Y - Y')^2}{N}}$$

3. Where σ_{est} is the standard error of the estimate, Y is an actual score, Y' is a predicted score, and N is the number of pairs of scores.

BEST FITTED LINE:

1. A line of best fit is a straight line that best represents the data on a scatter plot.

2. This line may pass through some of the points, none of the points, or all of the points.

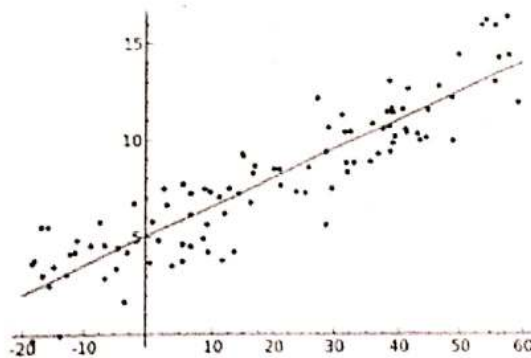3. Figure 2.3 shows the example of best fitted line.

Figure 2.3: Best Fitted Line Example.

4. The red line in the above graph is referred to as the best fit straight line

Q5.   The following table shows the midterm and final exam grades obtained for students in a database course.

Use the method of least squares using regression to predict the final exam grade of a student who received 86 on the midterm exam.

Midterm exam (x)	Final exam (y)
72	84
50	63
81	77
74	78
94	90
86	75
59	49
83	79
65	77
33	52
88	74
81	90

Ans: [10M | May17]

Finding x*y and x² using given data:

x	y	x*y	x²
72	84	6048	5184
50	63	3150	2500
81	77	6237	6561
74	78	5772	5476
94	90	8460	8836
86	75	6450	7396
59	49	2891	3481
83	79	6557	6889
65	77	5005	4225
33	52	1716	1089
88	74	6512	7744
81	90	7290	6561

Here n = 12 (total number of values in either x or y)

Now we have to find $\sum x$, $\sum y$, $\sum(x*y)$ and $\sum x^2$

Where, $\sum x$ = sum of all x values

   $\sum y$ = sum of all y values

   $\sum(x*y)$ = sum of all x*y values

   $\sum x^2$ = sum of all x² values

$\sum x$ = 866

$\sum y$ = 888

$\sum(x*y)$ = 66088

$\sum x^2$ = 65942

Now we have to find a & b.

$$a = \frac{n \cdot \Sigma xy - \Sigma x \Sigma y}{n \cdot \Sigma x^2 - (\Sigma x)^2}$$

Putting values in above equation,

$$a = \frac{(12 \cdot 66088) - (866 \cdot 888)}{(12 \cdot 65942) - (866)^2}$$

a = 0.59

$$b = \frac{\Sigma y - a \cdot \Sigma x}{n}$$

$$b = \frac{888 - (0.59 \cdot 866)}{12}$$

b = 31.42

Estimating final exam grade of a student who received 86 marks = y = a*x + b

Here, a = 0.59

b = 31.42

x = 86

Putting these values in equation of y.

y   = (0.59 * 86) + 31.42

= 82.16 marks

Q6. **The values of independent variable x and dependent value y are given below:**

Find the least square regression line y = ax + b. Estimate the value of y when x is 10.

X	Y
0	2
1	3
2	5
3	4
4	6

Ans: [10M | May!8]

Finding (x*y) and x^2 using given data

x	y	x*y	x²
0	2	0	0
1	3	3	1
2	5	10	4
3	4	12	9
4	6	24	16

Here n = 5 (total number of values in either x or y)

Now we have to find Σx, Σy, $\Sigma(x*y)$ and Σx^2

Where, Σx = sum of all x values

Σy = sum of all y values

$\Sigma(x^*y)$ = sum of all x*y values

Σx^2 = sum of all x^2 values

Σx	= 10
Σy	= 20
$\Sigma(x^*y)$	= 49
Σx^2	= 30

Now we have to find a & b.

$a = \frac{n \cdot \Sigma xy - \Sigma x \Sigma y}{n \cdot \Sigma x^2 - (\Sigma x)^2}$

Putting values in above equation,

$a = \frac{(5 \cdot 49) - (10 \cdot 20)}{(5 \cdot 30) - (10)^2}$

a = 0.9

$b = \frac{\Sigma y - a \cdot \Sigma x}{n}$

$b = \frac{20 - (0.9 \cdot 10)}{5}$

b = 2.2

Estimating final exam grade of a student who received 86 marks =

y = a*x + b

Here, a = 0.9

   b = 2.2

   x = 10

Putting these values in equation of y.

y = (0.9 * 10) + 2.2

y = 11.2

Q7. What is linear regression? Find best fitted line for following example:

i	x	y	ȳ
1	63	127	102.1
2	64	121	126.3
3	66	142	138.5
4	69	157	157.0
5	69	162	157.0
6	71	156	169.2
7	71	169	169.2

8	72	165	175.4
9	73	181	181.5
10	75	208	193.8

Ans: [10M – Dec18]

LINEAR REGRESSION:

Refer Q2.

SUM:

Finding x*y and x^2 using given data

x	y	x*y	x^2
63	127	8001	3969
64	121	7744	4096
66	142	9372	4356
69	157	10833	4761
69	162	11178	4761
71	156	11076	5041
71	169	11999	5041
72	165	11880	5184
73	181	13213	5329
75	208	15600	5625

Here n = 10 (total number of values in either x or y)

Now we have to find $\sum x$, $\sum y$, $\sum(x*y)$ and $\sum x^2$

Where, $\sum x$ = sum of all x values

$\qquad \sum y$ = sum of all y values

$\qquad \sum(x*y)$ = sum of all x*y values

$\qquad \sum x^2$ = sum of all x^2 values

$\sum x \qquad$ = 693

$\sum y \qquad$ = 1588

$\sum(x*y) \quad$ = 110896

$\sum x^2 \qquad$ = 48163

Now we have to find a & b.

$a = \frac{n \cdot \sum xy - \sum x \sum y}{n \cdot \sum x^2 - (\sum x)^2}$

Putting values in above equation,

$$a = \frac{(10 \cdot 110896) - (693 \cdot 1588)}{(10 \cdot 48163) - (693)^2}$$

$a = 6.14$

$$b = \frac{\Sigma y - a \cdot \Sigma x}{n}$$

$$b = \frac{1588 - (6.14 \cdot 693)}{10}$$

$b = -266.71$

Finding best fitting line=
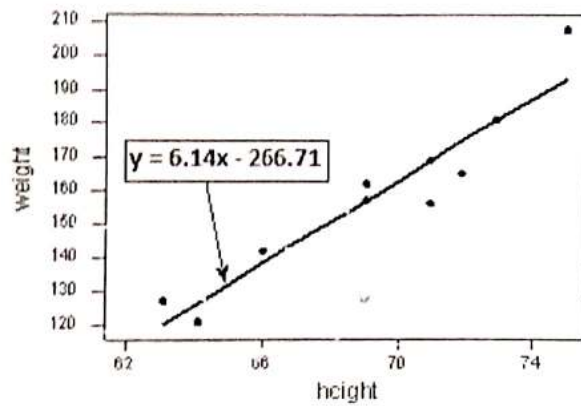
$y = a^* x + b$

here, $a = 6.14$

$b = -266.71$

$x = $ unknown

Putting these values in equation of y.

$y = 6.14 ^* x - 266.71$

Therefore, best fitting line **y = 6.14x – 266.71**

Graph for best fitting line:

CHAP - 3: LEARNING WITH TREES

Q1. What is decision tree? How you will choose best attribute for decision tree classifier? Give suitable example.

Ans: [10M | Dec18]

DECISION TREE:

1. Decision tree is the most powerful and popular tool for classification and prediction.

2. A Decision tree is a **flowchart like tree structure**.

3. Each internal node denotes a test on an attribute.

4. Each branch represents an outcome of the test.

5. Each leaf node (terminal node) holds a class label.

6. **Best attribute** is the attribute that "best" classifies the available training examples.

7. There are two terms one needs to be familiar with in order to define the "best" - entropy and information gain.

8. Entropy is a number that represents how heterogeneous a set of examples is based on their target class.

9. Information gain, on the other hand, shows how much the entropy of a set of examples will decrease if a specific attribute is chosen.

10. Criteria for selecting "best" attribute:

    a. Want to get smallest tree.

    b. Choosing the attribute that produces purest nodes.

EXAMPLE:

Q2. Explain procedure to construct decision tree

Ans: [5M | Dec18]

DECISION TREE:

1. Decision tree is the most powerful and popular tool for classification and prediction.

2. A Decision tree is a flowchart like tree structure.

3. Each internal node denotes a test on an attribute.

4. Each branch represents an outcome of the test.

Scanned by CamScanner

5. Each leaf node (terminal node) holds a class label.

6. In decision trees, for predicting a class label for a record we start from the root of the tree.

7. We compare the values of the root attribute with record's attribute.

8. On the basis of comparison, we follow the branch corresponding to that value and jump to the next node.

9. We continue comparing our record's attribute values with other internal nodes of the tree.

10. We do this comparison until we reach a leaf node with predicted class value.

11. As we know how the modelled decision tree can be used to predict the target class of the value.

PSEUDO CODE FOR DECISION TREE ALGORITHM:

1. Place the best attribute of the dataset at the root of the tree.

2. Split the training set into subsets.

3. Subsets should be made in such a way that each subset contains data with the same value for an attribute.

4. Repeat step 1 and step 2 on each subset until you find leaf nodes in all the branches of the tree.
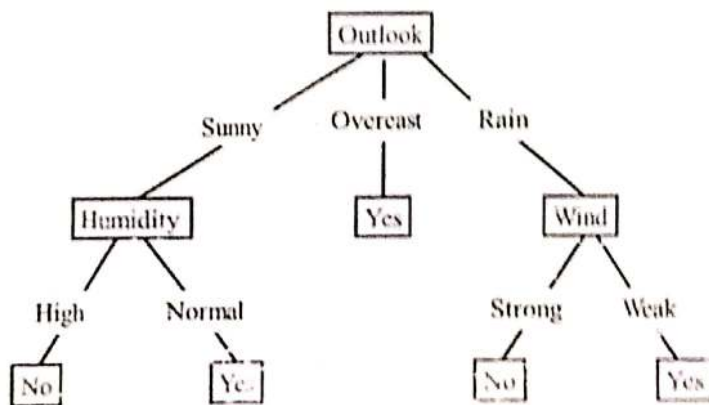
CONSTRUCTION OF DECISION TREE:

Figure 3.1: Example of Decision Tree for Tennis Play

1. Figure 3.1 shows the example of decision tree for tennis play.

2. A tree can be "learned" by splitting the source set into subsets based on an attribute value test.

3. This process is repeated on each derived subset in a recursive manner called recursive partitioning.

4. The recursion is completed when the subset at a node all has the same value of the target variable, or when splitting no longer adds value to the predictions.

5. The construction of decision tree classifier does not require any domain knowledge or parameter setting.

6. Therefore is appropriate for exploratory knowledge discovery.

7. Decision trees can handle high dimensional data.

8. In general decision tree classifier has good accuracy.

9. Decision tree induction is a typical inductive approach to learn knowledge on classification

Scanned by CamScanner

Q3. **What are the issues in decision tree induction?**

Ans: [10M | Dec16 & May18]

ISSUES IN DECISION TREE INDUCTION:

I) **Instability:**

1. The reliability of the information in the decision tree depends on feeding the precise internal and external information at the onset.

2. Even a small change in input data can at times, cause large changes in the tree.

3. Following things will require reconstructing the tree:

   a. Changing variables.

   b. Excluding duplication information.

   c. Altering the sequence midway.

II) **Analysis Limitations:**

1. Among the major disadvantages of a decision tree analysis is its inherent limitations.

2. The major limitations include:

   a. Inadequacy in applying regression and predicting continuous values.

   b. Possibility of spurious relationships.

   c. Unsuitability for estimation of tasks to predict values of a continuous attribute.

   d. Difficulty in representing functions such as parity or exponential size.

III) **Over fitting:**

1. Over fitting happens when learning algorithm continues to develop hypothesis

2. It reduce training set error at the cost of an increased test set error

3. How to avoid over fitting-

   a. **Pre-pruning:** It stops growing the tree very early, before it classifies the training set.

   b. **Post-pruning:** It allows tree to perfectly classify the training set and then prune the tree.

IV) **Attributes with many values:**

1. If attributes have a lot values, then the Gain could select any value for processing.

2. This reduces the accuracy for classification.

V) **Handling with costs:**

1. Strong quantitative and analytical knowledge required to build complex decision trees.

2. This raises the possibility of having to train people to complete a complex decision tree analysis.

3. The costs involved in such training makes decision tree analysis an expensive option.

VI) **Unwieldy:**

1. Decision trees, while providing easy to view illustrations, can also be unwieldy.

2. Even data that is perfectly divided into classes and uses only simple threshold tests may require a large decision tree.

3. Large trees are not intelligible, and pose presentation difficulties.

4. Drawing decision trees manually usually require several re-draws owing to space constraints at some sections

5. There is no fool proof way to predict the number of branches or spears that emit from decisions or sub-decisions.

VII) Incorporating continuous valued attributes:

1. The attributes which have continuous values can't have a proper class prediction

2. For example, AGE or Temperature can have any values

3. There is no solution for it until a range is defined in decision tree itself.

VIII) Handling examples with missing attributes values:

1. It is possible to have missing values in training set.

2. To avoid this, most common value among examples can be selected for tuple in consideration.

IX) Unable to determine depth of decision tree:

1. If the training set does not have an end value i.e. the set is given to be continuous.

2. This can lead to an infinite decision tree building.

X) Complexity:

1. Among the major decision tree disadvantages are its complexity.

2. Decision trees are easy to use compared to other decision-making models.

3. Preparing decision trees, especially large ones with many branches, are complex and time-consuming affairs.

4. Computing probabilities of different possible branches, determining the best split of each node.

Q4. For the given data determine the entropy after classification using each attribute for classification separately and find which attribute is best as decision attribute for the root by finding information gain with respect to entropy of Temperature as reference attribute.

Sr. No	Temperature	Wind	Humidity
1	Hot	Weak	High
2	Hot	Strong	High
3	Mild	Weak	Normal
4	Cool	Strong	High
5	Cool	Weak	Normal
6	Mild	Strong	Normal
7	Mild	Weak	High
8	Hot	Strong	High
9	Mild	Weak	Normal
10	Hot	Strong	Normal

Ans:

[10M | May16]

First we have to find entropy of all attributes,

1. **Temperature:**

    There are three distinct values in Temperature which are Hot, Mild and Cool.

    As there are three distinct values in reference attribute, Total information gain will be I(p, n, r).

    Here, p = total count of Hot = 4

    n = total count of Mild = 4

    r = total count of cool = 2

    s = p + n + r = 4 + 4 + 2 = 10

    Therefore,

    $$I(p, n, r) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s} - \frac{r}{s}\log_2\frac{r}{s}$$

Scanned by CamScanner

$$= -\frac{4}{10}\log_2\frac{4}{10} - \frac{4}{10}\log_2\frac{4}{10} - \frac{2}{10}\log_2\frac{2}{10}$$

I(p, n, r)   = 1.522 .... using calculator

2. **Wind:**

There are two distinct values in Wind which are Strong and Weak.

As there are two distinct values in reference attribute, Total information gain will be I(p, n).

Here, p = total count of Strong = 5

n = total count of Weak = 5

s = p + n = 5 + 5 = 10

Therefore,

I(p, n) $= -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

$$= -\frac{5}{10}\log_2\frac{5}{10} - \frac{5}{10}\log_2\frac{5}{10}$$

I(p, n)   = 1 .... as value of p and n are same, the answer will be 1.

3. **Humidity:**

There are two distinct values in Humidity which are High and Normal.

As there are two distinct values in reference attribute, Total information gain will be I(p, n).

Here, p = total count of High = 5

n = total count of Normal = 5

s = p + n = 5 + 5 = 10

Therefore,

I(p, n) $= -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

$$= -\frac{5}{10}\log_2\frac{5}{10} - \frac{5}{10}\log_2\frac{5}{10}$$

I(p, n)   = 1 .... as value of p and n are same, the answer will be 1.

Now we will find best root node using Temperature as reference attribute.

Here, reference attribute is Temperature.

 a. There are three distinct values in Temperature which are Hot, Mild and Cool.

 b. Here we will find Total Information Gain for whole data using reference attribute.

 c. As there are three distinct values in reference attribute, Total information gain will be I(p, n, r).

 d. Here,   p = total count of Hot = 4

         n = total count of Mild = 4

         r = total count of cool = 2

         s = p + n + r = 4 + 4 + 2 = 10

Therefore,

I(p, n, r) $= -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s} - \frac{r}{s}\log_2\frac{r}{s}$

Scanned by CamScanner

$$= -\frac{4}{10}\log_2\frac{4}{10} - \frac{4}{10}\log_2\frac{4}{10} - \frac{2}{10}\log_2\frac{2}{10}$$

$$I(p, n, r) \qquad = 1.522 \quad .... \text{ using calculator}$$

Now we will find Information Gain, Entropy and Gain of other attributes except reference attribute.

1. **Wind:**

Wind attribute have two distinct values which are weak and strong.

We will find information gain of these distinct values as following

    I. Weak =

      p = no of Hot values related to weak = 1

      n = no of Mild values related to weak = 3

      r = no of Cool values related to weak = 1

      $s = p + n + r = 1 + 3 + 1 = 5$

      Therefore,

      $I(weak) = I(p, n, r) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s} - \frac{r}{s}\log_2\frac{r}{s}$

      $= -\frac{1}{5}\log_2\frac{1}{5} - \frac{3}{5}\log_2\frac{3}{5} - \frac{1}{5}\log_2\frac{1}{5}$

      $I(weak) = I(p, n, r) = 1.371 \quad .... \text{ using calculator}$

    II. Strong =

      p = no of Hot values related to strong = 3

      n = no of Mild values related to strong = 1

      r = no of Cool values related to strong = 1

      $s = p + n + r = 3 + 1 + 1 = 5$

      Therefore,

      $I(weak) = I(p, n, r) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s} - \frac{r}{s}\log_2\frac{r}{s}$

      $= -\frac{3}{5}\log_2\frac{3}{5} - \frac{1}{5}\log_2\frac{1}{5} - \frac{1}{5}\log_2\frac{1}{5}$

      $I(weak) = I(p, n, r) = 1.371 \quad .... \text{ using calculator}$

Therefore,

Wind			
Distinct values from Wind	(total related values of Hot) p_i	(total related values of Mild) n_i	(total related values of Cool) r_i	Information Gain of value $I(p_i, n_i, r_i)$
Weak	1	3	1	1.371
Strong	3	1	1	1.371

Now we will find Entropy of Wind as following,

Entropy of Wind = $\sum_{i=1}^{k} \frac{p_i + n_i + r_i}{p+n+r} \times I(p_i, n_i, r_i)$

Here,   p + n + r = total count of Hot, Mild and Cold from reference attribute = 10

   $p_{i+n_i+r_i}$ = total count of related values from above table for distinct values in Wind attribute

   $I(p_i, n_i, r_i)$ = Information gain of particular distinct value of attribute

Entropy of Wind $= \frac{p_i+n_i+r_i \text{ for weak}}{p+n+r} \times I(p_i, n_i, r_i) + \frac{p_i+n_i+r_i \text{ for strong}}{p+n+r} \times I(p_i, n_i, r_i)$

   $= \frac{1+3+1}{10} \times 1.371 + \frac{3+1+1}{10} \times 1.371$

Entropy of wind   = 1.371

Gain of wind = Total Information Gain – Entropy of wind = 1.522 – 1.371 = **0.151**

2. **Humidity:**

Humidity attribute have two distinct values which are High and Normal.

We will find information gain of these distinct values as following

i.   **High =**

   p_i = no of Hot values related to High = 3

   n_i = no of Mild values related to High = 1

   r_i = no of Cool values related to High = 1

   $s_i = p_i + n_i + r_i = 3 + 1 + 1 = 5$

   Therefore,

   $I(High) = I(p, n, r) = -\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s} - \frac{r}{s} \log_2 \frac{r}{s}$

   $= -\frac{3}{5} \log_2 \frac{3}{5} - \frac{1}{5} \log_2 \frac{1}{5} - \frac{1}{5} \log_2 \frac{1}{5}$

   $I(High) = I(p, n, r) = 1.371$ .... using calculator

II.   **Normal =**

   p_i = no of Hot values related to Normal = 1

   n_i = no of Mild values related to Normal = 3

   r_i = no of Cool values related to Normal = 1

   $s_i = p_i + n_i + r_i = 1 + 3 + 1 = 5$

   Therefore,

   I(Normal) $= I(p, n, r) = -\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s} - \frac{r}{s} \log_2 \frac{r}{s}$

   $= -\frac{1}{5} \log_2 \frac{1}{5} - \frac{3}{5} \log_2 \frac{3}{5} - \frac{1}{5} \log_2 \frac{1}{5}$

   I(weak) $= I(p, n, r) = 1.371$ .... using calculator

Therefore,

Humidity				
Distinct values from Humidity	(total related values of Hot) p_i	(total related values of Mild) n_i	(total related values of Cool) r_i	Information Gain of valu $I(p_i, n_i, r_i)$
High	3	1	1	1.371
Normal	1	3	1	1.371

Now we will find Entropy by Humidity as following,

Entropy of Humidity = $\sum_{i=1}^{k} \frac{p_i + n_i + r_i}{p+n+r} \times I(p_i, n_i, r_i)$

Here, p + n + r = total count of Hot, Mild and Cold from reference attribute = 10

$p_i + n_i + r_i$ = total count of related values from above table for distinct values in Humidity attribute

$I(p_i, n_i, r_i)$ = Information gain of particular distinct value of attribute

Entropy of Humidity = $\frac{p_i + n_i + r_i \, for \, weak}{p+n+r} \times I(p_i, n_i, r_i) + \frac{p_i + n_i + r_i \, for \, strong}{p+n+r} \times I(p_i, n_i, r_i)$

Entropy of Humidity = $\frac{1+3+1}{10} \times 1.371 + \frac{3+1+1}{10} \times 1.371$

Entropy of Humidity = 1.371

Gain of wind = Total Information Gain – Entropy of wind = 1.522 – 1.371 = **0.151**

Gain of wind = Gain of humidity = 0.151

Here both values are same so we can take any one attribute as root node.

If they were different then we would have selected biggest value from it.

Q5. Create a decision tree for the attribute "class" using the respective values:

Eye Colour	Married	Sex	Hair Length	Class
Brown	Yes	Male	Long	Football
Blue	Yes	Male	Short	Football
Brown	Yes	Male	Long	Football
Brown	No	Female	Long	Netball
Brown	No	Female	Long	Netball
Blue	No	Male	Long	Football
Brown	No	Female	Long	Netball
Brown	No	Male	Short	Football
Brown	Yes	Female	Short	Netball
Brown	No	Female	Long	Netball
Blue	No	Male	Long	Football
Blue	No	Male	Short	Football

Ans:

[10M | Dec16]

Finding total information gain I(p, n) using class attribute.

There are two distinct values in Class which are Football and Netball.

Here, p = total count of Football = 7

----

n = total count of Netball = 5

s = p + n = 7 + 5 = 12

Therefore,

$I(p, n)$ $\qquad = -\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s}$

$\qquad = -\frac{7}{12} \log_2 \frac{7}{12} - \frac{5}{12} \log_2 \frac{5}{12}$

$I(p, n) = 0.980$ .... using calculator

Now we will find Information Gain, Entropy and Gain of other attributes except reference attribute

1. **Eye Colour:**

Wind attribute have two distinct values which are Brown and Blue. We will find information gain of these distinct values as following

   I.   **Brown =**

      p_i = no of Football values related to weak = 3

      n_i = no of Netball values related to weak = 5

      $s_i = p_i + n_i = 3 + 5 = 8$

      Therefore,

      $I(Brown)$ $\qquad = I(p, n) = -\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s}$

      $\qquad\qquad\qquad = -\frac{3}{8} \log_2 \frac{3}{8} - \frac{5}{8} \log_2 \frac{5}{8}$

      $I(Brown) = I(p, n) = 0.955$ .... using calculator

   II.   **Blue =**

      p_i = no of Football values related to Blue = 4

      n_i = no of Netball values related to Blue = 0

      $s_i = p_i + n_i = 4 + 0 = 4$

      Therefore,

      $I(Blue) = I(p, n) = -\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s}$

      $\qquad\qquad\qquad = -\frac{4}{4} \log_2 \frac{4}{4} - \frac{0}{4} \log_2 \frac{0}{4}$

      $\qquad\qquad\qquad = 0$ .... If anyone value is 0 then the answer will be 0 for Information gain

Therefore,

Eye Colour			
Distinct values from Eye Colour	(total related values of Football) p_i	(total related values of Netball) n_i	Information Gain of value $I(p_i, n_i)$
Brown	3	5	0.955
Blue	4	0	0

Now we will find Entropy of Eye Colour as following,

Entropy of Eye Colour $= \sum_{i=1}^{k} \frac{p_i + n_i}{p+n} \times I(p_i, n_i)$

Here, $p + n$ = total count of Football and Netball from class attribute = 12

$p_i + n_i$ = total count of related values from above table for distinct values in Eye Colour attribute

$I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Eye Colour $= \frac{p_i + n_i \; for \; Brown}{p+n} \times I(p_i, n_i) \; of \; Brown + \frac{p_i + n_i \; for \; Blue}{p+n} \times I(p_i, n_i) \; of \; Blue$

$$= \frac{3+5}{12} \times 0.955 + \frac{4+0}{12} \times 0$$

Entropy of Eye Colour $= 0.637$

Gain of Eye Colour = Total Information Gain – Entropy of Eye Colour = 0.980 – 0.637 = **0.343**

2. Married:

Married attribute have two distinct values which are Yes and No. We will find information gain of these distinct values as following

I. Yes =

p_i = no of Football values related to yes = 3

n_i = no of Netball values related to yes = 1

$s_i = p_i + n_i = 3 + 1 = 4$

Therefore,

$I(Yes) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

$$= -\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4}$$

$I(Yes) = I(p, n) = 0.812$ .... using calculator

II. No =

p_i = no of Football values related to No = 4

n_i = no of Netball values related to No = 4

$s_i = p_i + n_i = 4 + 4 = 8$

Therefore,

$I(No) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

$$= -\frac{4}{8}\log_2\frac{4}{8} - \frac{4}{8}\log_2\frac{4}{8}$$

$= 1$ .... As both value are 4 which is same so answer will be 1

Therefore,

Married			
Distinct values from Married	(total related values of Football) p_i	(total related values of Netball) n_i	Information Gain of value $I(p_i, n_i)$
Yes	3	1	0.812
No	4	4	1

Now we will find Entropy of Married as following,

Entropy of Married $= \sum_{i=1}^{k} \frac{p_i + n_i}{p+n} \times I(p_i, n_i)$

Here,   p + n = total count of Football and Netball from class attribute = 12

$p_i + n_i$ = total count of related values from above table for distinct values in Married attribute

$I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Married $= \frac{p_i + n_i \, for \, Yes}{p+n} \times I(p_i, n_i) \, of \, Yes + \frac{p_i + n_i \, for \, No}{p+n} \times I(p_i, n_i) \, of \, No$

$$= \frac{3+1}{12} \times 0.812 + \frac{4+4}{12} \times 1$$

Entropy of Married = 0.938

Gain of Married = Total Information Gain – Entropy of Married = 0.980 – 0.938 = **0.042**

3. Sex:

Wind attribute have two distinct values which are Male and Female. We will find information gain of these distinct values as following

I. **Male =**

p_i = no of Football values related to Male = 7

n_i = no of Netball values related to Male = 0

$s_i = p_i + n_i = 7 + 0 = 7$

Therefore,

$I(Yes) = I(p, n) = -\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s}$

$$= -\frac{7}{7} \log_2 \frac{7}{7} - \frac{0}{7} \log_2 \frac{0}{7}$$

I(Yes) = I(p, n) = 0 .... using calculator

ii. **Female =**

p_i = no of Football values related to Female = 0

n_i = no of Netball values related to Female = 5

$s_i = p_i + n_i = 0 + 5 = 5$

Therefore,

$I(No) = I(p, n) = -\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s}$

$$= -\frac{0}{5} \log_2 \frac{0}{5} - \frac{5}{5} \log_2 \frac{5}{5}$$

= 0 .... If both values are same then the answer will be 1 for Information gain

Therefore,

Sex

Distinct values from Sex	(total related values of Football) p_i	(total related values of Netball) n_i	Information Gain of value $I(p_i, n_i)$
Male	7	0	0
Female	0	5	0

Now we will find Entropy of Sex as following,

Entropy of Sex = $\sum_{i=1}^{k} \frac{p_{i}+n_{i}}{p+n} \times I(p_i, n_i)$

Here, p + n = total count of Football and Netball from class attribute = 12

p_{i+n_i} = total count of related values from above table for distinct values in Sex attribute

$I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Sex = $\frac{p_{i+n_i}\, for\, Male}{p+n} \times I(p_i, n_i)$ of Yes + $\frac{p_{i+n_i}\, for\, Female}{p+n} \times I(p_i, n_i)$ of No

$$= \frac{7+0}{12} \times 0 + \frac{0+5}{12} \times 0$$

Entropy of Sex = 0

Gain of Sex = Total Information Gain – Entropy of Sex = 0.980 – 0 = **0.980**

4. Hair Length:

Hair Length attribute have two distinct values which are Long and Short. We will find information gain these distinct values as following

I.   Long =

p_i = no of Football values related to Long = 4

n_i = no of Netball values related to Long = 4

$S_i = p_i + n_i = 4 + 4 = 8$

Therefore,

$I(Long) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

$$= -\frac{4}{8}\log_2\frac{4}{8} - \frac{4}{8}\log_2\frac{4}{8}$$

$I(Long) = I(p, n) = 1$ .... As both values are 4 which is same so answer will be 1

II.   Short =

p_i = no of Football values related to Short = 3

n_i = no of Netball values related to Short = 1

$S_i = p_i + n_i = 3 + 1 = 4$

Therefore,

$I(Short) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

$$= -\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4}$$

$$= 0.812$$

Therefore,

Hair Length			
Distinct values from Hair length	(total related values of Football) p_i	(total related values of Netball) n_i	Information Gain of value $I(p_i, n_i)$
Long	4	4	1
Short	3	1	0.812

Scanned by CamScanner

Now we will find Entropy of Hair length as following,

Entropy of Hair Length $= \sum_{i=1}^{k} \frac{p_i+n_i}{p+n} \times I(p_i, n_i)$

Here,   p + n = total count of Football and Netball from class attribute = 12

p_{i+n_i} = total count of related values from above table for distinct values in Hair Length attribute

$I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Hair Length $= \frac{P_{i+n_i \, for \, Long}}{p+n} \times I(p_i, n_i) \, of \, Long + \frac{P_{i+n_i \, for \, Short}}{p+n} \times I(p_i, n_i) \, of \, Short$

$= \frac{4+4}{12} \times 1 + \frac{3+1}{12} \times 0.812$

Entropy of Hair Length   = 0.938

Gain of Hair Length = Total Information Gain - Entropy of Hair Length = 0.980 – 0.938 = **0.042**

Gain of all Attributes except class attribute,

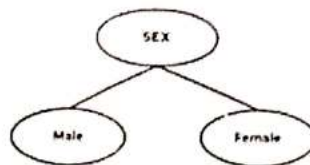Gain (Eye Colour) = 0.343

Gain (Married) = 0.042

Gain (Sex) = 0.980

Gain (Hair Length) = 0.042

Here, Sex attribute have largest value so attribute 'Sex' we be root node of Decision tree.

First we will go for Male value to get its child node,

Now we have to repeat the entire process where Sex = Male

We will take those tuples from given data which contains Male as Sex. And construct a table of those tuples.

Eye Colour	Married	Sex	Hair Length	Class
Brown	Yes	Male	Long	Football
Blue	Yes	Male	Short	Football
Brown	Yes	Male	Long	Football
Blue	No	Male	Long	Football
Brown	No	Male	Short	Football
Blue	No	Male	Long	Football
Blue	No	Male	Short	Football

Here we can see that Football is the only one value of class which is related to Male value of Sex class. So we can say that all Male plays Football.

Now we will go for Female value to get its child node,

We will take those tuples from given data which contains Female as Sex. And construct a table of those tuples.

Eye Colour	Married	Sex	Hair Length	Class
Brown	No	Female	Long	Netball
Brown	No	Female	Long	Netball
Brown	No	Female	Long	Netball
Brown	Yes	Female	Short	Netball
Brown	No	Female	Long	Netball

Here we can see that Netball is the only one value of class which is related to Female value of Sex class. So we can say that all Female plays Netball.

So Final Decision Tree will be as following
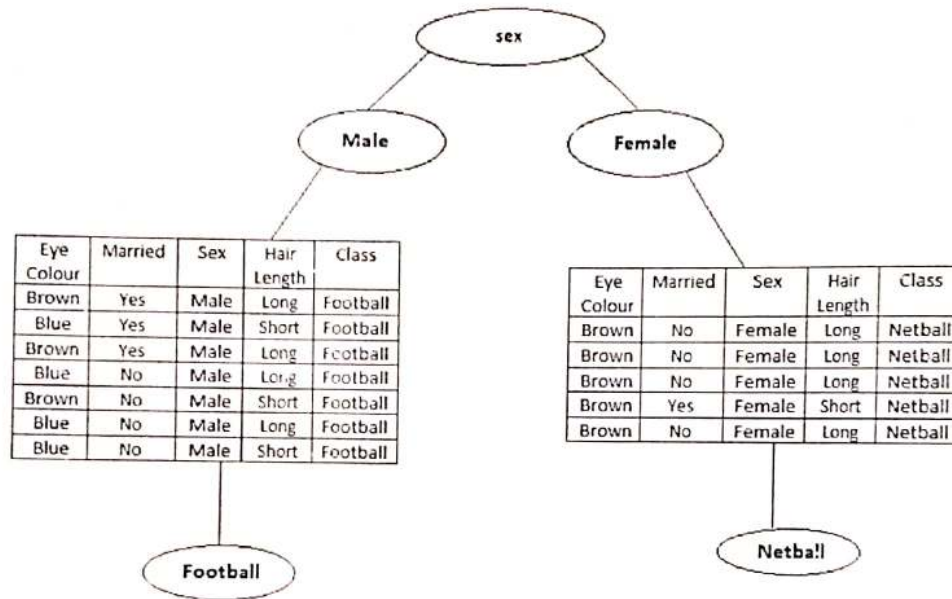
Q6. For the given data determine the entropy after classification using each attribute for classification separately and find which attribute is best as decision attribute for the root by finding information gain with respect to entropy of Temperature as reference attribute.

Sr. No	Temperature	Wind	Humidity
1	Hot	Weak	Normal
2	Hot	Strong	High
3	Mild	Weak	Normal
4	Mild	Strong	High
5	Cool	Weak	Normal
6	Mild	Strong	Normal
7	Mild	Weak	High
8	Hot	Strong	Normal
9	Mild	Strong	Normal
10	Cool	Strong	Normal

Ans:

[10M | May16]

Scanned by CamScanner

First we have to find entropy of all attributes,

1. **Temperature:**

   There are three distinct values in Temperature which are Hot, Mild and Cool.

   As there are three distinct values in reference attribute, Total information gain will be $I(p, n, r)$.

   Here, p = total count of Hot = 3

   n = total count of Mild = 5

   r = total count of cool = 2

   s = p + n + r = 3 + 5 + 2 = 10

   Therefore,

   $$I(p, n, r) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s} - \frac{r}{s}\log_2\frac{r}{s}$$

   $$= -\frac{3}{10}\log_2\frac{3}{10} - \frac{5}{10}\log_2\frac{5}{10} - \frac{2}{10}\log_2\frac{2}{10}$$

   $I(p, n, r) = 1.486$ .... using calculator

2. **Wind:**

   There are two distinct values in Wind which are Strong and Weak.

   As there are two distinct values in reference attribute, Total information gain will be $I(p, n)$.

   Here, p = total count of Strong = 6

   n = total count of Weak = 4

   s = p + n = 6 + 4 = 10

   Therefore,

   $$I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

   $$= -\frac{6}{10}\log_2\frac{6}{10} - \frac{4}{10}\log_2\frac{4}{10}$$

   $I(p, n) = 0.971$ .... as value of p and n are same, the answer will be 1.

3. **Humidity:**

   There are two distinct values in Humidity which are High and Normal.

   As there are two distinct values in reference attribute, Total information gain will be $I(p, n)$.

   Here, p = total count of High = 3

   n = total count of Normal = 7

   s = p + n = 3 + 7 = 10

   Therefore,

   $$I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

   $$= -\frac{3}{10}\log_2\frac{3}{10} - \frac{7}{10}\log_2\frac{7}{10}$$

   $I(p, n) = 0.882$ .... as value of p and n are same, the answer will be 1.

Now we will find best root node using Temperature as reference attribute.

Here, reference attribute is Temperature.

There are three distinct values in Temperature which are Hot, Mild and Cool.

As there are three distinct values in reference attribute, Total information gain will be I(p, n, r).

Here, p = total count of Hot = 3

n = total count of Mild = 5

r = total count of cool = 2

s = p + n + r = 3 + 5 + 2 = 10

Therefore,

$$I(p, n, r) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s} - \frac{r}{s}\log_2\frac{r}{s}$$

$$= -\frac{3}{10}\log_2\frac{3}{10} - \frac{5}{10}\log_2\frac{5}{10} - \frac{2}{10}\log_2\frac{2}{10}$$

I(p, n, r) = 1.486 .... using calculator

Now we will find Information Gain, Entropy and Gain of other attributes except reference attribute

1. Wind:

Wind attribute have two distinct values which are weak and strong.

We will find information gain of these distinct values as following

I. Weak =

p_i = no of Hot values related to weak = 1

n_i = no of Mild values related to weak = 2

r_i = no of Cool values related to weak = 1

s_i = p_i + n_i + r_i = 1 + 2 + 1 = 4

Therefore,

$$I(weak) = I(p, n, r) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s} - \frac{r}{s}\log_2\frac{r}{s}$$

$$= -\frac{1}{4}\log_2\frac{1}{4} - \frac{2}{4}\log_2\frac{2}{4} - \frac{1}{4}\log_2\frac{1}{4}$$

I(weak) = I(p, n, r) = 1.5 .... using calculator

II. Strong =

p_i = no of Hot values related to strong = 2

n_i = no of Mild values related to strong = 3

r_i = no of Cool values related to strong = 1

s_i = p_i + n_i + r_i = 2 + 3 + 1 = 6

Therefore,

$$I(weak) = I(p, n, r) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s} - \frac{r}{s}\log_2\frac{r}{s}$$

$$= -\frac{2}{6}\log_2\frac{2}{6} - \frac{3}{6}\log_2\frac{3}{6} - \frac{1}{6}\log_2\frac{1}{6}$$

I(weak) = I(p, n, r) = 1.460 .... using calculator

Scanned by CamScanner

Therefore,

Wind				
Distinct values from Wind	(total related values of Hot) p_i	(total related values of Mild) n_i	(total related values of Cool) r_i	Information Gain of value $I(p_i, n_i, r_i)$
Weak	1	2	1	1.5
Strong	2	3	1	1.460

Now we will find Entropy of Wind as following,

Entropy of Wind = $\sum_{i=1}^{k} \frac{p_i + n_i + r_i}{p + n + r} \times I(p_i, n_i, r_i)$

Here,   p + n + r = total count of Hot, Mild and Cold from reference attribute = 10

$p_i + n_i + r_i$ = total count of related values from above table for distinct values in Wind attribute

$I(p_i, n_i, r_i)$ = Information gain of particular distinct value of attribute

Entropy of Wind $= \frac{p_i + n_i + r_i \, for \, weak}{p+n+r} \times I(p_i, n_i, r_i) + \frac{p_i + n_i + r_i \, for \, strong}{p+n+r} \times I(p_i, n_i, r_i)$

$= \frac{1+2+1}{10} \times 1.5 + \frac{2+3+1}{10} \times 1.460$

Entropy of wind   = 1.476

Gain of wind = Entropy of Reference – Entropy of wind = 1.486 – 1.476 = **0.01**

2. **Humidity :**

Humidity attribute have two distinct values which are High and Normal.

We will find information gain of these distinct values as following

I.   **High =**

p = no of Hot values related to High = 1

n = no of Mild values related to High = 2

r = no of Cool values related to High = 0

s = p + n + r = 1 + 2 + 0 = 3

Therefore,

I(High) = I(p, n, r) = $-\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s} - \frac{r}{s} \log_2 \frac{r}{s}$

$= -\frac{1}{3} \log_2 \frac{1}{3} - \frac{2}{3} \log_2 \frac{2}{3} - \frac{0}{3} \log_2 \frac{0}{3}$

I(High) = I(p, n, r) = 0.919 .... using calculator

II.   **Normal =**

p = no of Hot values related to Normal = 2

n = no of Mild values related to Normal = 3

r = no of Cool values related to Normal = 2

s = p + n + r = 2 + 3 + 2 = 7

Therefore,

I(Normal)   = I(p, n, r) = $-\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s} - \frac{r}{s} \log_2 \frac{r}{s}$

$= -\frac{2}{7} \log_2 \frac{2}{7} - \frac{3}{7} \log_2 \frac{3}{7} - \frac{2}{7} \log_2 \frac{2}{7}$

I(weak) = I(p, n, r) = 1.557 .... using calculator

Therefore,

Humidity				
Distinct values from Humidity	(total related values of Hot) p_i	(total related values of Mild) n_i	(total related values of Cool) r_i	Information Gain of val $I(p_i, n_i, r_i)$
High	1	2	0	0.919
Normal	2	3	2	1.557

Now we will find Entropy by Humidity as following,

Entropy of Humidity = $\sum_{i=1}^{k} \frac{p_i + n_i + r_i}{p + n + r} \times I(p_i, n_i, r_i)$

Here, p + n + r = total count of Hot, Mild and Cold from reference attribute = 10

$p_{i+n_i+r_i}$ = total count of related values from above table for distinct values in Humidity attribute

$i(p_i, n_i, r_i)$ = Information gain of particular distinct value of attribute

Entropy of Humidity = $\frac{p_i+n_i+r_i \, for \, weak}{p+n+r} \times I(p_i, n_i, r_i) + \frac{p_i+n_i+r_i \, for \, strong}{p+n+r} \times I(p_i, n_i, r_i)$

Entropy of Humidity = $\frac{1+2+0}{10} \times 0.919 + \frac{2+3+2}{10} \times 1.557$

Entropy of Humidity = 1.366

Gain of wind = Entropy of Reference – Entropy of wind = 1.486 – 1.366 = **0.12**

Gain of wind = 0.01

Gain of humidity = 0.12

Here value of Gain(Humidity) is biggest so we will take Humidity attribute as root node.

Q7. **For a SunBurn dataset given below, construct a decision tree.**

Name	Hair	Height	Weight	Location	Class
Sunita	Blonde	Average	Light	No	Yes
Anita	Blonde	Tall	Average	Yes	No
Kavita	Brown	Short	Average	Yes	No
Sushma	Blonde	Short	Average	No	Yes
Xavier	Red	Average	Heavy	No	Yes
Balaji	Brown	Tall	Heavy	No	No
Ramesh	Brown	Average	Heavy	No	No
Swetha	Blonde	Short	Light	Yes	No

Ans: [10M – May17 & May18]

First we will find entropy of Class attribute as following,

There are two distinct values in Class which are Yes and No.

Here, p = total count of Yes = 5

n = total count of No = 3

s = p + n = 5 + 3 = 8

Therefore,

Scanned by CamScanner

$$I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{5}{8}\log_2\frac{5}{8} - \frac{3}{8}\log_2\frac{3}{8}$$

$$I(p, n) = 0.955 \dots\dots\dots\dots\dots \text{ using calculator}$$

Now we will find Information Gain, Entropy and Gain of other attributes except reference attribute

1. Hair:

Wind attribute have three distinct values which are Blonde, Brown and Red. We will find information gain of these distinct values as following

I. **Blonde =**

   p_i = no of Yes values related to Blonde = 2

   n_i = no of No values related to Blonde = 2

   $s_i = p_i + n_i = 2 + 2 = 4$

   Therefore,

   $$I(\text{Blonde}) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

   $$= -\frac{2}{4}\log_2\frac{2}{4} - \frac{2}{4}\log_2\frac{2}{4}$$

   $I(\text{Blonde}) = I(p, n) = 1 \dots\dots\dots\dots\dots$ as value of p and n is same, so answer will be 1

II. **Brown =**

   p_i = no of Yes values related to Brown = 0

   n_i = no of No values related to Brown = 2

   $s_i = p_i + n_i = 0 + 2 = 2$

   Therefore,

   $$I(\text{Blue}) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

   $$= -\frac{2}{2}\log_2\frac{2}{2} - \frac{0}{2}\log_2\frac{0}{2}$$

   $= 0 \dots\dots\dots\dots\dots$ If anyone value is 0 then the answer will be 0 for Information gain

III. **Red =**

   p_i = no of Yes values related to Red = 1

   n_i = no of No values related to Red = 0

   $s_i = p_i + n_i = 1 + 0 = 2$

   Therefore,

   $$I(\text{Blue}) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

   $$= -\frac{1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}$$

   $= 0 \dots\dots\dots\dots\dots$ If anyone value is 0 then the answer will be 0 for Information gain

Therefore,

Hair			
Distinct values from Eye Colour	(total related values of Football) p_i	(total related values of Netball) n_i	Information Gain of value $I(p_i, n_i)$
Blonde	2	2	1
Brown	2	0	0
Red	1	0	0

Now we will find Entropy of Hair as following,

Entropy of Hair = $\sum_{i=1}^{k} \frac{p_i + n_i}{p+n} \times I(p_i, n_i)$

Here,   p + n = total count of Yes and No from class attribute = 8

$p_i + n_i$ = total count of related values from above table for distinct values in Hair attribute

$I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Eye Colour = $\frac{P_{i+n_i} \text{ for Blonde}}{p+n} \times I(p_i, n_i) \text{ of Blonde} + \frac{P_{i+n_i} \text{ for Brown}}{p+n} \times I(p_i, n_i) \text{ of Brown} + \frac{P_{i+n_i} \text{ for Red}}{p+n}$ $I(p_i, n_i) \text{ of Red}$

$$= \frac{2+2}{8} \times 2 + \frac{2+0}{8} \times 0 + \frac{1+0}{8} \times 0$$

Entropy of Hair   = 0.5

Gain of Hair = Entropy of Class – Entropy of Hair = 0.955 – 0.5 = **0.455**

2. Height:

Height attribute have three distinct values which are Average, Tall and Short. We will find information gain of these distinct values as following

I.   Tall =

p_i = no of Yes values related to Tall = 0

n_i = no of No values related to Tall = 2

s_i = $p_i + n_i$ = 0 + 2 = 2

Therefore,

I(Average) = I(p, n) = $-\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s}$

$$= -\frac{0}{2} \log_2 \frac{0}{2} - \frac{2}{2} \log_2 \frac{2}{2}$$

I(Average) = I(p, n) = 0 .... If any value from p and n is 0 then answer will be 0

II.   Average =

p_i = no of Yes values related to Average = 2

n_i = no of No values related to Average = 1

s_i = $p_i + n_i$ = 2 + 1 = 3

Therefore,

I(Average) = I(p, n) = $-\frac{p}{s} \log_2 \frac{p}{s} - \frac{n}{s} \log_2 \frac{n}{s}$

$$= -\frac{2}{3} \log_2 \frac{2}{3} - \frac{1}{3} \log_2 \frac{1}{3}$$

$$= 0.919$$

III. **Short =**

   p_i = no of Yes values related to Short = 1

   n_i = no of No values related to Red = 2

   $s_i = p_i + n_i = 1 + 2 = 3$

   Therefore,

   $I(Short) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

   $= -\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}$

   $= 0.919$

Therefore,

Height

Distinct values from Height	(total related values of Yes) p_i	(total related values of No) n_i	Information Gain of value $I(p_i, n_i)$
Tall	0	2	0
Average	2	1	0.919
Short	1	2	0.919

Now we will find Entropy of Height as following.

Entropy of Height $= \sum_{i=1}^{k} \frac{p_i + n_i}{p + n} \times I(p_i, n_i)$

Here, p + n = total count of Yes and No from class attribute = 8

   $p_i + n_i$ = total count of related values from above table for distinct values in Height attribute

   $I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Height= $\frac{P_i + n_i\ for\ Tall}{p+n} \times I(p_i, n_i)$ of Tall $+ \frac{P_i + n_i\ for\ Average}{p+n} \times I(p_i, n_i)$ of Average $+ \frac{P_i + n_i\ for\ Short}{p+n} \times I(p_i, n_i)$ of Short

   $= \frac{0+2}{8} \times 0 + \frac{2+1}{8} \times 0.919 + \frac{1+2}{8} \times 0.919$

Entropy of Height = 0.690

Gain of Height = Entropy of Class – Entropy of Height = 0.955 – 0.690 = **0.265**

3. Weight:

Weight attribute have three distinct values which are Heavy, Average and Light. We will find information gain of these distinct values as following

I. **Heavy =**

   p_i = no of Yes values related to Heavy = 1

   n_i = no of No values related to Heavy = 2

   $S_i = p_i + n_i = 1 + 2 = 3$

   Therefore,

   $I(Average) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

   $= -\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}$

   $I(Average) = I(p, n) = 0.919$

II.   **Average =**

p_i = no of Yes values related to Average = 1

n_i = no of No values related to Average = 2

$s_i = p_i + n_i = 1 + 2 = 3$

Therefore,

$$I(Average) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}$$

$$= 0.919$$

III.   **Light =**

p_i = no of Yes values related to Light = 1

n_i = no of No values related to Light = 1

$s_i = p_i + n_i = 1 + 1 = 2$

Therefore,

$$I(Light) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$$= 1 \text{ .... As value of p and n are same, so answer will be 1}$$

Therefore,

Weight			
Distinct values from Weight	(total related values of Yes) p_i	(total related values of No) n_i	Information Gain of value $I(p_i, n_i)$
Heavy	1	2	0.919
Average	1	2	0.919
Light	1	1	1

Now we will find Entropy of Weight as following,

Entropy of Weight = $\sum_{i=1}^{k}\frac{p_i+n_i}{p+n} \times I(p_i, n_i)$

Here,   p + n = total count of Yes and No from class attribute = 8

$p_i + n_i$ = total count of related values from above table for distinct values in Weight attribute

$I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy   of   Height= $\frac{p_i+n_i \, for \, Heavy}{p+n} \times I(p_i, n_i)$ of Heavy + $\frac{p_i+n_i \, for \, Average}{p+n} \times I(p_i, n_i)$ of Average + $\frac{p_i+n_i \, for \, Light}{p+n}$ ×

$I(p_i, n_i)$ of Light

$$= \frac{1+2}{8} \times 0.919 + \frac{1+2}{8} \times 0.919 + \frac{1+1}{8} \times 1$$

Entropy of Height = 0.94

Gain of Height = Entropy of Class – Entropy of Height = 0.955 – 0.94 = **0.015**

4. **Location:**

Location attribute have two distinct values which are Yes and No. We will find information gain of these distinct values as following

    I.   **Yes =**

        p_i = no of Yes values related to Yes = 0

        n_i = no of No values related to Yes = 3

        $s_i = p_i + n_i = 0 + 3 = 3$

        Therefore,

        $I(Average) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

        $= -\frac{0}{3}\log_2\frac{0}{3} - \frac{3}{3}\log_2\frac{3}{3}$

        $I(Average) = I(p, n) = 0$ .... if any one value from p and n is 0 then answer will be 0

    II.   **No =**

        p_i = no of Yes values related to No = 3

        n_i = no of No values related to No = 2

        $s_i = p_i + n_i = 3 + 2 = 5$

        Therefore,

        $I(Average) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$

        $= -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}$

        $= 0.971$

Therefore,

Location			
Distinct values from Location	(total related values of Yes) p_i	(total related values of No) n_i	Information Gain of value $I(p_i, n_i)$
Yes	0	3	0
No	3	2	0.971

Now we will find Entropy of Location as following,

Entropy of Location = $\sum_{i=1}^{k}\frac{P_i + n_i}{p+n} \times I(p_i, n_i)$

Here, p + n = total count of Yes and No from class attribute = 8

        $p_i + n_i$ = total count of related values from above table for distinct values in Location attribute

        $I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Location = $\frac{P_i + n_i \, for \, Yes}{p+n} \times I(p_i, n_i) \, of \, Yes + \frac{P_i + n_i \, for \, No}{p+n} \times I(p_i, n_i) \, of \, No$

        $= \frac{0+3}{8} \times 0 + \frac{3+2}{8} \times 0.971$

Entropy of Height = 0.607

Gain of Height = Entropy of Class – Entropy of Height = 0.955 – 0.607 = **0.348**
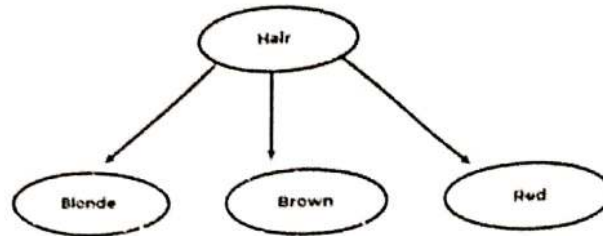
Here,

Gain (Hair) = 0.455

Gain (Height) = 0.265

Gain (Weight) = 0.015

Gain (Location) = 0.348

Here, we can see that Gain of Hair attribute is highest So we will take Hair attribute as root node.

Now we will construct a table for each distinct value of Hair attribute as following,

Bionde -

Name	Height	Weight	Location	Class
Sunita	Average	Light	No	Yes
Anita	Tall	Average	Yes	No
Sushma	Short	Average	No	Yes
Swetha	Short	Light	Yes	No

Browr: -

Name	Height	Weight	Location	Class
Kavita	Short	Average	Yes	No
Balaji	Tall	Heavy	No	No
Ramesh	Average	Heavy	No	No

Red -

Name	Height	Weight	Location	Class
Xavier	Average	Heavy	No	Yes

As we can see that for table of brown values, class value is No for all data tuples and same for table of Re values where class value is Yes for all data tuples.

So we will expand the node of Blonde value in decision tree.

We will now use following table which we have constructed above for Blonde value,

Name	Height	Weight	Location	Class
Sunita	Average	Light	No	Yes
Anita	Tall	Average	Yes	No
Sushma	Short	Average	No	Yes
Swetha	Short	Light	Yes	No

Now we will find Entropy of class attribute again as following,

There are two distinct values in Class which are Yes and No.

Here, p = total count of Yes = 2

n = total count of No = 2

s = p + n = 2 + 2 = 4

Therefore,

$$I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{2}{4}\log_2\frac{2}{4} - \frac{2}{4}\log_2\frac{2}{4}$$

$I(p, n) = 1$ .... As value of p and n are same, so answer will be 1

Now we will find Information Gain, Entropy and Gain of other attributes except reference attribute

1. Height :

Height attribute have three distinct values which are Average, Tall and Short. We will find information gain of these distinct values as following

I. Tall =

p_i = no of Yes values related to Tall = 0

n_i = no of No values related to Tall = 1

$s_i = p_i + n_i = 0 + 1 = 1$

Therefore,

$$I(Tall) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{0}{1}\log_2\frac{0}{1} - \frac{1}{1}\log_2\frac{1}{1}$$

$I(Tall) = I(p, n) = 0$ .... If any value from p and n is 0 then answer will be 0

II. Average =

p_i = no of Yes values related to Average = 1

n_i = no of No values related to Average = 0

$s_i = p_i + n_i = 1 + 0 = 1$

Therefore,

$$I(Average) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}$$

$= 0$ .... If any value from p and n is 0 then answer will be 0

III. Short =

p = no of Yes values related to Short = 1

n = no of No values related to Red = 1

$s_i = p_i + n_i = 1 + 1 = 2$

Therefore,

Scanned by CamScanner

$$I(Short) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$$= 1 \dots\dots\dots\dots \text{As value of p and n are same, answer will be 1}$$

Therefore,

Height			
Distinct values from Height	(total related values of Yes) p_i	(total related values of No) n_i	Information Gain of value $I(p_i, n_i)$
Tall	0	1	0
Average	1	0	0
Short	1	1	1

Now we will find Entropy of Height as following,

Entropy of Height = $\sum_{i=1}^{k} \frac{p_i + n_i}{p+n} \times I(p_i, n_i)$

Here, p + n = total count of Yes and No from class attribute = 4

$p_i + n_i$ = total count of related values from above table for distinct values in Height attribute

$I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Height = $\frac{p_i + n_i \text{ for Tall}}{p+n} \times I(p_i, n_i)$ of Tall + $\frac{p_i + n_i \text{ for Average}}{p+n} \times I(p_i, n_i)$ of Average + $\frac{p_i + n_i \text{ for Short}}{p+n} \times$

$I(p_i, n_i)$ of Short

$$= \frac{0+1}{4} \times 0 + \frac{1+0}{4} \times 0 + \frac{1+1}{4} \times 1$$

Entropy of Height = 0.5

Gain of Height = Entropy of Class – Entropy of Height = 1 – 0.5 = **0.5**

2. Weight:

Weight attribute have three distinct values which are Heavy, Average and Light. We will find informatio

gain of these distinct values as following

I. Heavy =

p_i = no of Yes values related to Heavy = 0

n_i = no of No values related to Heavy = 0

$s_i = p_i + n_i = 0 + 0 = 0$

Therefore,

$$I(Heavy) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= !(p, n) = 0 \dots\dots\dots\dots \text{As value of p and n are 0.}$$

II. Average =

p_i = no of Yes values related to Average = 1

n_i = no of No values related to Average = 1

$s_i = p_i + n_i = 1 + 1 = 2$

Therefore,

$$I(Average) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$= 1$ .... as value of p and n are same.

III. **Light =**

p_i = no of Yes values related to Light = 1

n_i = no of No values related to Light = 1

$s_i = p_i + n_i = 1 + 1 = 2$

Therefore,

$$I(Light) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$= 1$ .... As value of p and n are same, so answer will be 1

Therefore,

Weight			
Distinct values from Weight	(total related values of Yes) p_i	(total related values of No) n_i	Information Gain of value $I(p_i, n_i)$
Heavy	0	0	0
Average	1	1	1
Light	1	1	1

Now we will find Entropy of Weight as following,

Entropy of Weight = $\sum_{i=1}^{k} \frac{p_i + n_i}{p + n} \times I(p_i, n_i)$

Here, p + n = total count of Yes and No from class attribute = 3

$p_i + n_i$ = total count of related values from above table for distinct values in Weight attribute

$I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Weight= $\frac{P_i+n_i\ for\ Heavy}{p+n} \times I(p_i, n_i)$ of Heavy + $\frac{P_i+n_i\ for\ Average}{p+n} \times I(p_i, n_i)$ of Average + $\frac{P_i+n_i\ for\ Light}{p+n} \times$

$I(p_i, n_i)$ of Light

$$= \frac{0+0}{4} \times 0 + \frac{1+1}{4} \times 1 + \frac{1+1}{4} \times 1$$

Entropy of Weight = 1

Gain of Weight = Entropy of Class - Entropy of Weight = 1 - 1 = **0**

3. Location:

Location attribute have two distinct values which are Yes and No. We will find information gain of these distinct values as following

I. **Yes =**

p_i = no of Yes values related to Yes = 0

n_i = no of No values related to Yes = 2

$s_i = p_i + n_i = 0 + 2 = 2$

Therefore,

$$I(Yes) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{0}{2}\log_2\frac{0}{2} - \frac{2}{2}\log_2\frac{2}{2}$$

$I(Yes) = I(p, n) = 0$ .... if any one value from p and n is 0 then answer will be 0

II.   No =

p_i = no of Yes values related to No = 2

n_i = no of No values related to No = 0

$s_i = p_i + n_i = 2 + 0 = 2$

Therefore,

$$I(No) = I(p, n) = -\frac{p}{s}\log_2\frac{p}{s} - \frac{n}{s}\log_2\frac{n}{s}$$

$$= -\frac{2}{2}\log_2\frac{2}{2} - \frac{0}{2}\log_2\frac{0}{2}$$

$= 0$ .... if any one value from p and n is 0 then answer will be 0

Therefore,

Location			
Distinct values from Location	(total related values of Yes) p_i	(total related values of No) n_i	Information Gain of value $I(p_i, n_i)$
Yes	0	2	0
No	2	0	0

Now we will find Entropy of Location as following,

Entropy of Location $= \sum_{i=1}^{k} \frac{p_i + n_i}{p+n} \times I(p_i, n_i)$

Here, p + n = total count of Yes and No from class attribute = 4

$p_i + n_i$ = total count of related values from above table for distinct values in Location attribute

$I(p_i, n_i)$ = Information gain of particular distinct value of attribute

Entropy of Location $= \frac{p_i + n_i\ for\ Yes}{p+n} \times I(p_i, n_i)\ of\ Yes + \frac{p_i + n_i\ for\ No}{p+n} \times I(p_i, n_i)\ of\ No$

$$= \frac{0+2}{4} \times 0 + \frac{2+0}{4} \times 0$$

Entropy of Location = 0

Gain of Location = Entropy of Class – Entropy of Location = 1 – 0 = **1**

Here,

Gain (Height) = 0.5

Gain (Weight) = 0

Gain (Location) = 1

As Gain of location is largest value, we will take location attribute as splitting node.

Now we will construct a table for each distinct value of Location attribute as following.
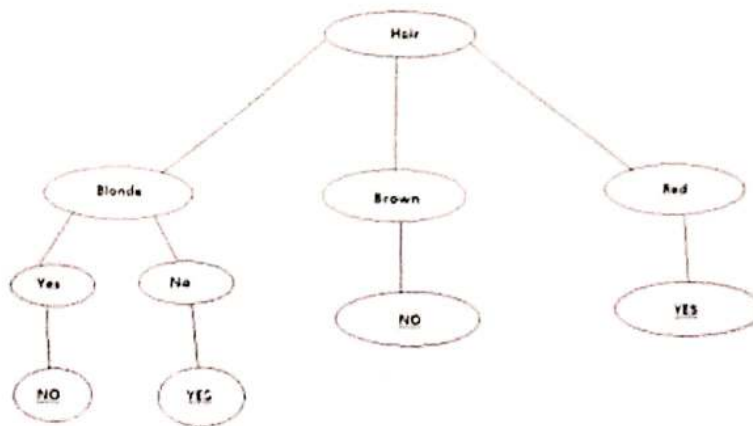
Yes -

Name	Height	Weight	Location	Class
Anita	Tall	Average	Yes	No
Swetha	Short	Light	Yes	No

No –

Name	Height	Weight	Location	Class
Sunita	Average	Light	No	Yes
Sushma	Short	Average	No	Yes

As we can see that class value for Yes (Location) is No and No(Location) is Yes. There is no need to do further classification.

The final Decision tree will be as following

CHAP - 4: SUPPORT VECTOR MACHINES

Q1. What are the key terminologies of Support Vector Machine?

Q2. What is SVM? Explain the following terms: hyperplane, separating hyperplane, margin and support vectors with suitable example.

Q3. Explain the key terminologies of Support Vector Machine

Ans: [5M | May16, Dec16 & May17]

SUPPORT VECTOR MACHINE:

1. A support vector machine is a supervised learning algorithm that sorts data into two categories.

2. A support vector machine is also known as a **support vector network (SVN)**.

3. It is trained with a series of data already classified into two categories, building the model as it is initially trained.

4. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible.

5. SVMs are used in text categorization, image classification, handwriting recognition and in the sciences.

HYPERPLANE:

1. A hyperplane is a generalization of a plane.

2. SVMs are based on the idea of finding a hyperplane that best divides a dataset into two classes/groups.

3. Figure 4.1 shows the example of hyperplane.
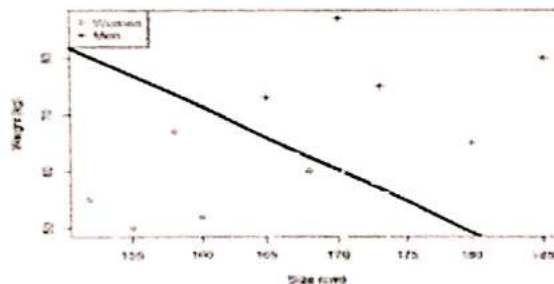
Figure 4.1: Example of hyperplane.

4. As a simple example, for a classification task with only two features as shown in figure 4.1, you can think of a hyperplane as a line that linearly separates and classifies a set of data.

5. When new testing data is added, whatever side of the hyperplane it lands will decide the class that we assign to it.

SEPARATING HYPERPLANE:

1. From figure 4.1, we can see that it is possible to separate the data.

2. We can use a line to separate the data.

3. All the data points representing men will be above the line.

4. All the data points representing women will be below the line.

5. Such a line is called a separating hyperplane.

MARGIN:

1. A margin is a separation of line to the closest class points.

2. The margin is calculated as the perpendicular distance from the line to only the closest points.

3. A good margin is one where this separation is larger for both the classes.

4. A good margin allows the points to be in their respective classes without crossing to other class.

5. The more width of margin is there, the more optimal hyperplane we get.

SUPPORT VECTORS:

1. The vectors (cases) that define the hyperplane are the support vectors.

2. Vectors are separated using hyperplane.

3. Vectors are mostly from a group which is classified using hyperplane.

4. Figure 4.2 shows the example of support vectors.
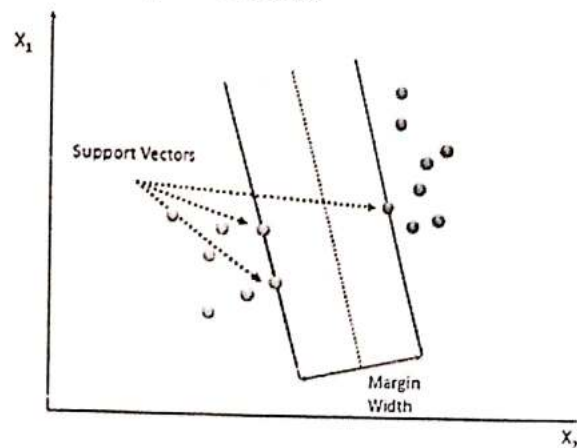
Figure 4.2. Example of support vectors.

Q4. **Define Support Vector Machine (SVM) and further explain the maximum margin linear separators concept.**

Ans:

[10M | Dec17]

SUPPORT VECTOR MACHINE:

1. A support vector machine is a supervised learning algorithm that sorts data into two categories.

2. A support vector machine is also known as a support vector network (SVN).

3. It is trained with a series of data already classified into two categories, building the model as it is initially trained.

4. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible.

5. SVMs are used in text categorization, image classification, handwriting recognition and in the sciences.

MAXIMAL-MARGIN CLASSIFIER/SEPARATOR:

1. The Maximal-Margin Classifier is a hypothetical classifier that best explains how SVM works in practice.

2. The numeric input variables (x) in your data (the columns) form an n-dimensional space.

3. For example, if you had two input variables, this would form a two-dimensional space.

4. A hyperplane is a line that splits the input variable space.

5. In SVM, a hyperplane is selected to best separate the points in the input variable space by their class, either class 0 or class 1.

6. In two-dimensions you can visualize this as a line and let's assume that all of our input points can be completely separated by this line.

7. For example $B_0 + (B_1 * X_1) + (B_2 * X_2) = 0$

8. Where the coefficients (B_1 and B_2) that determine the slope of the line and the intercept (B_0) are found by the learning algorithm, and X_1 and X_2 are the two input variables.

9. You can make classifications using this line.

10. By plugging in input values into the line equation, you can calculate whether a new point is above or below the line.

11. Above the line, the equation returns a value greater than 0 and the point belongs to the first class.

12. Below the line, the equation returns a value less than 0 and the point belongs to the second class.

13. A value close to the line returns a value close to zero and the point may be difficult to classify.

14. If the magnitude of the value is large, the model may have more confidence in the prediction.

15. The distance between the line and the closest data points is referred to as the margin.

16. The best or optimal line that can separate the two classes is the line that as the largest margin.

17. This is called the **Maximal-Margin hyperplane**.

18. The margin is calculated as the perpendicular distance from the line to only the closest points.

19. Only these points are relevant in defining the line and in the construction of the classifier.

20. These points are called the **support vectors**.

21. They support or define the hyperplane.

22. The hyperplane is learned from training data using an optimization procedure that maximizes the margin.
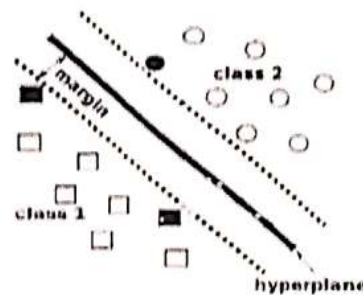

Figure 4.3: Maximum margin linear separators concept.

Q5. What is Support Vector Machine (SVM)? How to compute the margin?

Q6. What is the goal of the Support Vector Machine (SVM)? How to compute the margin?

Ans: [10M | May17 & May18]

SUPPORT VECTOR MACHINE:

Refer Q4 (SVM Part)

MARGIN:

1. A margin is a separation of line to the closest class points.

2. The margin is calculated as the perpendicular distance from the line to only the closest points.

3. A good margin is one where this separation is larger for both the classes.

4. A good margin allows the points to be in their respective classes without crossing to other class.

5. The more width of margin is there, the more optimal hyperplane we get.

Scanned by CamScanner

EXAMPLE FOR HOW TO FIND MARGIN:

Consider building an SVM over the (very little) data set shown in figure 4.4 for an example like this
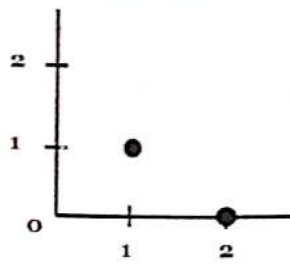

Figure 4.4

1. The maximum margin weight vector will be parallel to the shortest line connecting points of the two classes.

2. The optimal decision surface is orthogonal to that line and intersects it at the halfway point.

3. Therefore, it passes through. So, the SVM decision boundary is:

$$y = x_1 + 2x_2 - 5.5$$

4. Working algebraically, with the standard constraint that, we seek to minimize.

5. This happens when this constraint is satisfied with equality by the two support vectors.

6. Further we know that the solution is for some.

7. So we have that:

$$a + 2a + b = -1$$
$$2a + 6a + b = 1$$

8. Therefore a=2/5 and b=-11/5.

9. So the optimal hyperplane is given by

$$\vec{w} = (2/5, 4/5)$$

And b = -11/5.

10. The margin boundary is:

$$2/|\vec{w}| = 2/\sqrt{4/25 + 16/25} = 2/(2\sqrt{5}/5) = \sqrt{5}$$

11. This answer can be confirmed geometrically by examining figure 4.4.

Q7.   Write short note on - Soft margin SVM

Ans: [10M | Dec18]

SOFT MARGIN SVM:

1. Soft margin is extended version of hard margin SVM.

2. Hard margin given by Boser et al. 1992 in COLT and soft margin given by Vapnik et al. 1995.

3. Hard margin SVM can work only when data is completely linearly separable without any errors (noise or outliers).

4. In case of errors either the margin is smaller or hard margin SVM fails.

5. On the other hand soft margin SVM was proposed by Vapnik to solve this problem by introducing slack variables.

6. As for as their usage is concerned since Soft margin is extended version of hard margin SVM so we use Soft margin SVM.

7. The allowance of softness in margins (i.e. a low cost setting) allows for errors to be made while fitting the model (support vectors) to the training/discovery data set.

8. Conversely, hard margins will result in fitting of a model that allows zero errors.

9. Sometimes it can be helpful to allow for errors in the training set.

10. It may produce a more generalizable model when applied to new datasets.

11. Forcing rigid margins can result in a model that performs perfectly in the training set, but is possibly over-fit / less generalizable when applied to a new dataset.

12. Identifying the best settings for 'cost' is probably related to the specific data set you are working with

13. Currently, there aren't many good solutions for simultaneously optimizing cost, features, and kernel parameters (if using a non-linear kernel).

14. In both the soft margin and hard margin case we are maximizing the margin between support vectors, I.e. minimizing $\frac{1}{2\|w\|^2}$

15. In soft margin case, we let our model give some relaxation to few points.

16. If we consider these points our margin might reduce significantly and our decision boundary will be poorer.

17. So instead of considering them as support vectors we consider them as error points.

18. And we give certain penalty for them which is proportional to the amount by which each data point is violating the hard constraint.

19. Slack variables ε can be added to allow misclassification of difficult or noisy examples.

20. This variables represent the deviation of the examples from the margin.

21. Doing this we are relaxing the margin, we are using a soft margin.

Q8.   What is Kernel? How kernel can be used with SVM to classify non-linearly separable data? Also, list standard kernel functions.

Ans: [10M | May18]

KERNEL:

1. A kernel is a similarity function.

2. SVM algorithms use a set of mathematical functions that are defined as the kernel.

3. The function of kernel is to take data as input and transform it into the required form.

4. It is a function that you provide to a machine learning algorithm.

5. It takes two inputs and spits out how similar they are.

6. Different SVM algorithms use different types of kernel functions.

7. For example linear, nonlinear, polynomial, radial basis function (RBF), and sigmoid.

EXAMPLE ON EXPLAINING HOW KERNEL CAN BE USED FOR CLASSIFYING NON-LINEARLY SEPARABLE DATA:

1. To predict if a dog is a particular breed, we load in millions of dog information/properties like type, height, skin colour, body hair length etc.

Scanned by CamScanner

2. In ML language, these properties are referred to as 'features'.

3. A single entry of these list of features is a data instance while the collection of everything is the Training Data which forms the basis of your prediction

4. I.e. if you know the skin colour, body hair length, height and so on of a particular dog, then you can predict the breed it will probably belong to.

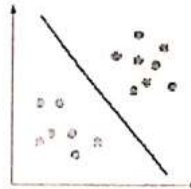5. In support vector machines, it looks somewhat like shown in figure 4.5 which separates the blue balls from red.

Figure 4.5

6. Therefore the hyperplane of a two dimensional space below is a one dimensional line dividing the red and blue dots.

7. From the example above of trying to predict the breed of a particular dog, it goes like this:

8. Data (all breeds of dog) → Features (skin colour, hair etc.) → Learning algorithm

9. If we want to solve following example in Linear manner then it is not possible to separate by straight line as we did in above steps.

Figure 4.6

10. The red and blue balls cannot be separated by a straight line as they are randomly distributed.

11. Here comes Kernel in picture.

12. In machine learning, a "kernel" is usually used to refer to the kernel trick, a method of using a linear classifier to solve a non-linear problem.

13. It entails transforming linearly inseparable data like (Figure 4.6) to linearly separable ones (Figure 4.5).

14. The kernel function is what is applied on each data instance to map the original non-linear observations into a higher-dimensional space in which they become separable.

15. Using the dog breed prediction example again, kernels offer a better alternative.

16. Instead of defining a slew of features, you define a single kernel function to compute similarity between breeds of dog.

17. You provide this kernel, together with the data and labels to the learning algorithm, and out comes a classifier.

18. **Mathematical definition:** $K(x, y) = <f(x), f(y)>$.

   Here K is the kernel function,

   x, y are n dimensional inputs.

   f is a map from n-dimension to m-dimension space.

   $< x, y >$ denotes the dot product. Usually m is much larger than n.

19. **Intuition:** Normally calculating <f(x), f(y)> requires us to calculate f(x), f(y) first, and then do the dot product.

20. These two computation steps can be quite expensive as they involve manipulations in m dimensional space, where m can be a large number.

21. But after all the trouble of going to the high dimensional space, the result of the dot product is really a scalar.

22. Therefore we come back to one-dimensional space again.

23. Now, the question we have is: do we really need to go through all the trouble to get this one number?

24. Do we really have to go to the m-dimensional space?

25. The answer is no, if you find a clever kernel.

26. Simple Example: $x = (x_1, x_2, x_3)$; $y = (y_1, y_2, y_3)$.

Then for the function $f(x) = (x_1x_1, x_1x_2, x_1x_3, x_2x_1, x_2x_2, x_2x_3, x_3x_1, x_3x_2, x_3x_3)$, the kernel is $K(x, y) = (<x, y>)^2$

27. Let's plug in some numbers to make this more intuitive:

28. Suppose $x = (1, 2, 3)$; $y = (4, 5, 6)$. Then:

$F(x) = (1, 2, 3, 2, 4, 6, 3, 6, 9)$

$F(y) = (16, 20, 24, 20, 25, 30, 24, 30, 36)$

$<F(x), F(y)> = 16 + 40 + 72 + 40 + 100 + 180 + 72 + 180 + 324 = 1024$

29. Now let us use the kernel instead:

$K(x, y) = (4 + 10 + 18)^2 = 32^2 = 1024$

30. Same result, but this calculation is so much easier.

Q9. **Quadratic Programming solution for finding maximum margin separation in Support Vector Machine.**

Ans: [10M – May16]

1. The linear programming model is a very powerful tool for the analysis of a wide variety of problems in the sciences, industry, engineering, and business.

2. However, it does have its limits.

3. Not all phenomena are linear.

4. Once nonlinearities enter the picture an LP model is at best only a first-order approximation.

5. The next level of complexity beyond linear programming is quadratic programming.

6. This model allows us to include nonlinearities of a quadratic nature into the objective function.

7. As we shall see this will be a useful tool for including the Markowitz mean-variance models of uncertainty in the selection of optimal port-folios.

8. A quadratic program (QP) is an optimization problem wherein one either minimizes or maximizes a quadratic objective function of a finite number of decision variable subject to a finite number of linear inequality and/or equality constraints.

9. A quadratic function of a finite number of variables $x = (x_1, x_2, \ldots, x_n)^T$ is any function of the form:

$$f(x) = \alpha + \sum_{j=1}^{n} c_j x_j + \frac{1}{2} \sum_{k=1}^{n} \sum_{j=1}^{n} q_{kj} x_k x_j.$$

10. Using matrix notation, this expression simplifies to

$$f(x) = a + c^T x + \frac{1}{2} x^T Q x,$$

where

$$c = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} \quad \text{and} \quad Q = \begin{pmatrix} q_{11} & q_{12} & \cdots & q_{1n} \\ q_{21} & q_{22} & \cdots & q_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ q_{n1} & q_{n2} & \cdots & q_{nn} \end{pmatrix}$$

11. The factor of one half preceding the quadratic term in the function f is included for the sake of convenience since it simplifies the expressions for the first and second derivatives of f.

12. With no loss in generality, we may as well assume that the matrix Q is symmetric since

$$x^T Q x = (x^T Q x)^T = x^T Q^T x = \frac{1}{2}(x^T Q x + x^T Q^T x) = x^T \left(\frac{Q + Q^T}{2}\right) x,$$

13. And so we are free to replace the matric Q by the symmetric matrix.

$$\frac{Q + Q^T}{2},$$

14. Henceforth, we will assume that the matrix Q is symmetric.

15. The QP standard form that we use is,

     Q minimize $c^T x + \frac{1}{2} x^T Q x$
                 subject to $Ax \le b$, $0 \le x$, where $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$.

16. Just as in the case of linear programming, every quadratic program can be transformed into one in standard form.

17. Observed that we can have simplified the expression for the objective function by dropping the constant term a since it plays no role in optimization step.

Q10. Explain how support Vector Machine can be used to find optimal hyperplane to classify linearly separable data. Give suitable example.

Ans: [10M | Dec18]

OPTIMAL HYPERPLANE:

1. Optimal hyperplane is completely defined by support vectors.

2. The optimal hyperplane is the one which maximizes the margin of the training data.
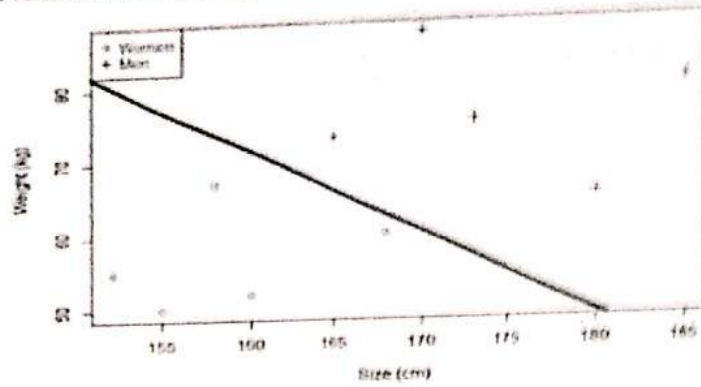
SUPPORT VECTOR MACHINE:

1. A support vector machine is a supervised learning algorithm that sorts data into two categories.

2. A support vector machine is also known as a support vector network (SVN).

3. It is trained with a series of data already classified into two categories, building the model as it is initially trained.

4. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible.

5. SVMs are used in text categorization, image classification, handwriting recognition and in the sciences.

HYPERPLANE:

1. A hyperplane is a generalization of a plane.

2. SVMs are based on the idea of finding a hyperplane that best divides a dataset into two classes/groups.

3. As a simple example, for a classification task with only two features (like the image above), you can think of a hyperplane as a line that linearly separates and classifies a set of data.

4. When new testing data is added, whatever side of the hyperplane it lands will decide the class that we assign to it.

5. **Hyperplane:** $(w \cdot x) + b = 0 \; w \in RN, b \in R$

6. **Corresponding decision function:** $f(x) = sgn((w \cdot x) + b)$

7. **Optimal hyperplane (maximal margin):**

   $max_{w,b} \; min_{i=1,\ldots,m}\{\|x - x_i\| : x \in R^N, (w \cdot x) + b = 0\}$

   with $y_i \in \{-1, +1\}$ holds: $y_i \cdot ((w \cdot x_i) + b) > 0$ for all $i = 1, \ldots, m$

   Where,   w and b not unique

   w and b can be scaled, so that $|(w \cdot x_i) + b| = 1$ for the x_i closest to the hyperplane

   Canonical form (w, b) of hyperplane, now holds:

   $y_i \cdot ((w \cdot x_i) + b) \geq 1$ for all $i = 1, \ldots, m$

   Margin of optimal hyperplane in canonical form equals $\frac{2}{\|w\|}$

Q11.   Find optimal hyperplane for the data points:

   {(1, 1), (2, 1), (1, -1), (2, -1), (4, 0), (5, 1), (5, -1), (6, 0)}

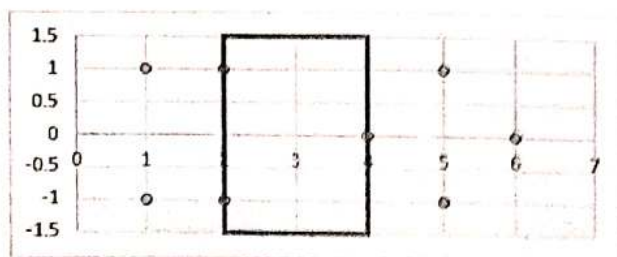**Ans:**   [10M – Dec16]

Plotting given support vector points on a graph (In exam, you can draw it roughly in paper).
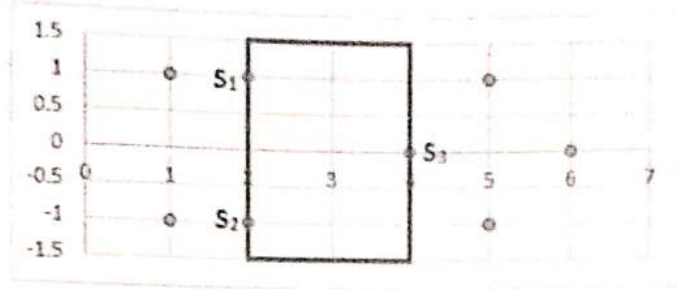

We are assuming that the hyperplane will appear in graph in following region.

We will use three support vectors which are closer to assumed region on graph.

Naming those support vector as S_1, S_2, S_3 as shown below

$$S_1 = \binom{2}{1}, \quad S_2 = \binom{2}{-1}, \quad S_3 = \binom{4}{0}$$

Based on these support vectors, we will find augmented vectors by adding 1 as bias point.

So Augmented vectors will be,

$$\bar{s_1} = \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix}, \quad \bar{s_2} = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix}, \quad \bar{s_3} = \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix}$$

Now we will find three parameters α_1, α_2, α_3 based on following three linear equations,

$$(\alpha_1 \times \bar{s_1} \times \bar{s_1}) + (\alpha_2 \times \bar{s_2} \times \bar{s_1}) + (\alpha_3 \times \bar{s_3} \times \bar{s_1}) = -1$$

$$(\alpha_1 \times \bar{s_1} \times \bar{s_2}) + (\alpha_2 \times \bar{s_2} \times \bar{s_2}) + (\alpha_3 \times \bar{s_3} \times \bar{s_2}) = -1$$

$$(\alpha_1 \times \bar{s_1} \times \bar{s_3}) + (\alpha_2 \times \bar{s_2} \times \bar{s_3}) + (\alpha_3 \times \bar{s_3} \times \bar{s_3}) = 1$$

Substituting values of $\bar{s_1}$, $\bar{s_2}$, $\bar{s_3}$ in above linear equations,

$$\left\{ \alpha_1 \times \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \times \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \right\} + \left\{ \alpha_2 \times \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \times \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \right\} + \left\{ \alpha_3 \times \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \times \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \right\} = -1$$

$$\left\{ \alpha_1 \times \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \times \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \right\} + \left\{ \alpha_2 \times \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \times \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \right\} + \left\{ \alpha_3 \times \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \times \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \right\} = -1$$

$$\left\{ \alpha_1 \times \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \times \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \right\} + \left\{ \alpha_2 \times \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \times \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \right\} + \left\{ \alpha_3 \times \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \times \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \right\} = 1$$

Simplifying these equations,

$$\{\alpha_1 \times [(2 \times 2) + (1 \times 1) + (1 \times 1)]\} + \{\alpha_2 \times [(2 \times 2) + (-1 \times 1) + (1 \times 1)]\} + \{\alpha_3 \times [(4 \times 2) + (0 \times 1) + (1 \times 1)]\} = -1$$

$$\{\alpha_1 \times [(2 \times 2) + (1 \times -1) + (1 \times 1)]\} + \{\alpha_2 \times [(2 \times 2) + (-1 \times -1) + (1 \times 1)]\} + \{\alpha_3 \times [(4 \times 2) + (0 \times 1) + (1 \times 1)]\} = -1$$

$$\{\alpha_1 \times [(2 \times 4) + (1 \times 0) + (1 \times 1)]\} + \{\alpha_2 \times [(2 \times 4) + (-1 \times 0) + (1 \times 1)]\} + \{\alpha_3 \times [(4 \times 4) + (0 \times 0) + (1 \times 1)]\} = 1$$

We get,

$$6\alpha_1 + 4\alpha_2 + 9\alpha_3 = -1$$

$$4\alpha_1 + 6\alpha_2 + 9\alpha_3 = -1$$

$$9\alpha_1 + 9\alpha_2 + 17\alpha_3 = 1$$

By solving above equations we get $\alpha_1 = \alpha_2 = -3.25$ and $\alpha_3 = 3.5$

To find hyperplane, we have to discriminate positive class from negative class. So we will use fol...

equation,

$$\bar{w} = \sum_i \alpha_i \bar{s}_i$$

Putting values of a and $\bar{s}$ in above equation

$$\bar{w} = \alpha_1 \bar{s}_1 + \alpha_2 \bar{s}_2 + \alpha_3 \bar{s}_3$$

$$\bar{w} = \left\{ -3.25 \times \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \right\} + \left\{ -3.25 \times \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \right\} + \left\{ 3.5 \times \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \right\}$$

We get,

$$\bar{w} = \begin{pmatrix} 1 \\ 0 \\ -3 \end{pmatrix}$$

Now we will remove the bias point which we have added to support vectors to get augmented vect...

So we will use hyperplane equation which is y = wx + b

Here $w = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ as we have removed the bias point from it which is -3

And b = -3 which is bias point. We can write this as b + 3 = 0 also.

So we will use w and b to plot the hyperplane on graph

As $w = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$,

so it means it is an vertical line. If it were $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ then the line would be horizontal line.

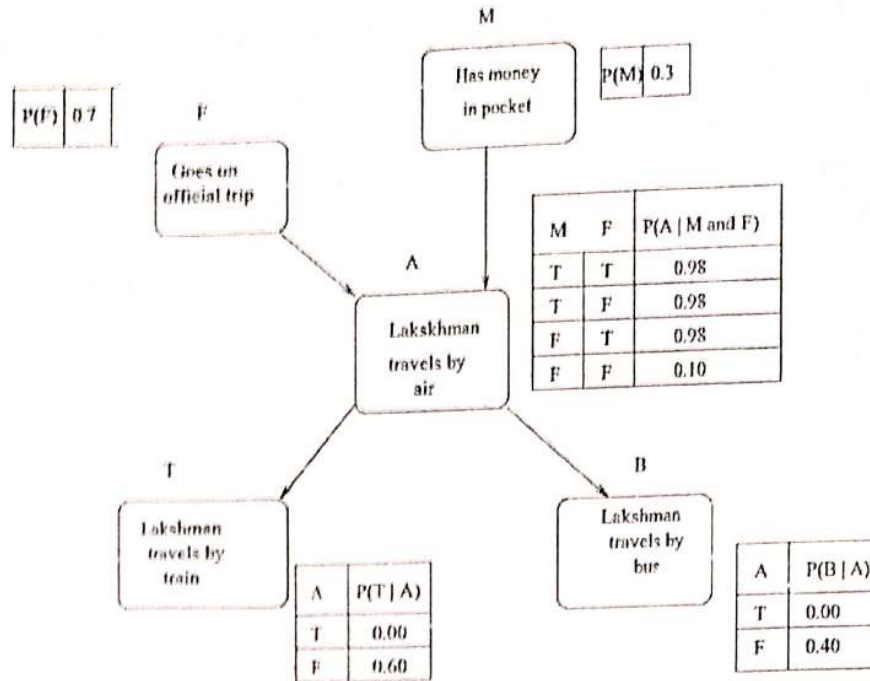And b + 3 = 0, so it means the hyperplane will go from point 3 as show below.
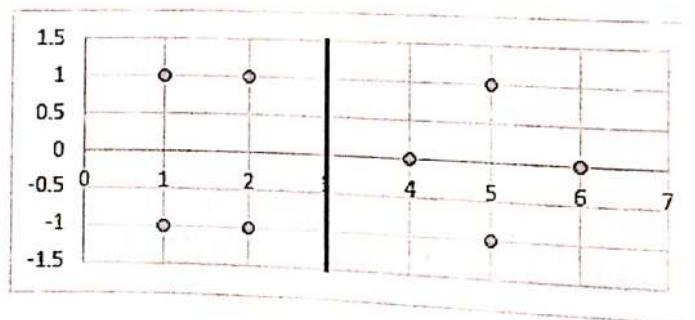
CHAP - 5: LEARNING WITH CLASSIFICATION

Q1. Explain with suitable example the advantages of Bayesian approach over classical approaches to probability.

Q2. Explain classification using Bayesian Belief Network with an example.

Q3. Explain, in brief, Bayesian Belief network.

Ans:

[10M | Dec16, Dec17 & Dec18]

BAYESIAN BELIEF NETWORK:

1. A Bayesian network is a **graphical model of a situation.**

2. It represents a set of variables and the dependencies between them by using probability.

3. The nodes in a Bayesian network represent the variables.

4. The directional arcs represent the dependencies between the variables.

5. The direction of the arrows show the direction of the dependency.

6. Each variable is associated with a conditional probability table (CPT).

7. CPT gives the probability of this variable for different values of the variables on which this node depends.

8. Using this model, it is possible to perform inference and learning.

9. BBN provides a graphical model of casual relationship on which learning can be performed.

10. We can use a trained Bayesian network for classification.

11. There are two components that defines a BBN:

    a. Directed Acyclic Graphs (DAG)

    b. A set of Conditional Probability Tables (CPT)

12. As an example, consider the following scenario.

    Pablo travels by air, if he is on an official visit. If he is on a personal visit, he travels by air if he has money. If he does not travel by plane, he travels by train but sometimes also takes a bus.

13. The variables involved are:

    a   Pablo travels by air (A)

    b.   Goes on official visit(F)

    c.   Pablo has money (M)

    d.   Pablo travels by train (T)

    e.   Pablo travels by bus (B)

14. This situation is converted into a belief network as shown in figure 5.1 below.

15. In the graph, we can see the dependencies with respect to the variables.

16. The probability values at a variable are dependent on the value of its parents.

17. In this case, the variable A is dependent on F and M.

18. The variable T is dependent on A and variable B is dependent on A.

19. The variables F and M are independent variables which do not have any parent node.

20. So their probabilities are not dependent on any other variable.

21. Node A has the biggest conditional probability table as A depends on F and M.

22. T and B depend on A.

M

Has money in pocket | P(M) 0.3

P(F) 0.7 | F

Goes on official trip

A

Lakskhman travels by air

M	F	P(A | M and F)
T	T	0.98
T	F	0.98
F	T	0.98
F	F	0.10

T

Lakshman travels by train

A	P(T | A)
T	0.00
F	0.60

B

Lakshman travels by bus

A	P(B | A)
T	0.00
F	0.40

23. First we take the independent nodes.

24. Node F has a probability of P (F) = 0.7.

25. Node M has a probability of P (M) = 0.3.

26. We next come to node A.

27. The conditional probability table for this node can be represented as

F	M	P(A | F,M and P)
T	T	0.98
T	F	0.98
F	T	0.90
F	F	0.10

28. The conditional probability table for T can be represented as

A	P	A
T		60
F		9.6

29. The conditional probability table for B is

A	P	A
T		0.0
F		9.40

30. Using the Bayesian belief network, we can get the probability of a combination of these variables.

31. For example, we can get the probability that Pablo travels by train, does not travel by air, goes on an official trip and has money.

32. In other words, we are finding P (T, ¬A, F, M).

33. The probability of each variable given its parent is found and multiplied together to give the probability.

P(T, ¬A, M, P)   = P(T | ¬A) • P(¬A | F and M) • P(F) • P(M)

= 0.6 * 0.98 * 0.7 * 0.3 = 0.123

Q4. Explain classification using Back Propagation algorithm with a suitable example

Q5. Classification using Back Propagation Algorithm

Q6. Explain how Back Propagation algorithm helps in classification

Q7. Write short note on - Back propagation algorithm

Ans: [10M | May16, Dec16, May17 & May18]

BACK PROPAGATION:

1. Back propagation is a supervised learning algorithm, for training Multi-layer Perceptron (Artificial Neural Networks).

2. Back propagation algorithm looks for minimum value of error in weight space.

3. It uses techniques like Delta rule or Gradient descent.

4. The weights that minimize the error function is then considered to be a solution to the learning problem.

BACK PROPAGATION ALGORITHM:

1. Iteratively process a set of training tuple and compare the network prediction with actual known target value.

2. For each training tuple, the weights are modified to minimize the mean squared error between this network's prediction and actual target value.

3. Modifications are made in "Backwards" direction from output layer.

4. Through each hidden layer down to first layer (hence called "back propagation").

5. The weight will eventually converge and learning process stops.

BACK PROPAGATION ALGORITHM STEPS:

1. **Initiate the weights:**
   a. The weights in networks are initialized to small random numbers
   b. Ex.- ranging from -1.0 to 1.0 or -0.5 to 0.5
   c. Each unit has a bias associated with it
   d. The biases are initialised to small random numbers

2. **Propagate the error:**
   a. The training tuple is fed to network's input layer.
   b. The input is passed through the input units, unchanged
   c. For an input j its output Oj is equal to its input value Ij
   d. Given a unit j in hidden or output layer, the net input I_j to unit j is

   $$I_j = \sum W_{ij}O_i + \theta_j$$

   e. Here,

   W_{ij} = Weight of connection from unit i in previous layer of unit j

   O_i = Output of unit i from previous layer

   j = Bias of unit

   f. Given the net input I_j to unit j, the O_j the output of unit j is computed as

   $$O_j = 1 / (1 + e^{-Ij})$$

Scanned by CamScanner

3. **Back Propagate the Error:**

    a. The error propagated backward by updating the weights and biases.

    b. It reflect the error of network's prediction.

    c. For unit j in output layer, the error $Err_j = O_j(1 - O_j)(T_j - O_j)$

    d. Where,

            O_j = actual output of unit j.

            T_j = Known target value of given tuple

            $O_j(1 - O_j)$ = derivate of logistic function

    e. The error of hidden layer unit J is

$$Err_j = O_j(1 - O_j) \sum Err_k W_{jk}$$

4. **Terminating Condition:**

    Training stops when:

    a. All W_{ij} in previous epoch are so small as to be below some specified threshold.

    b. The percentage of tuple misclassified in previous epoch is below some threshold.

    c. A pre-specified number of epochs has expired.

Q8.   Hidden Markov Model

Q9.   What are the different Hidden Markov Models

Q10.   Explain Hidden Markov Models.

Ans: [10M | May16, Dec16, May17 & Dec17]

HIDDEN MARKOV MODEL:

1. Hidden Markov Model is a statistical model based on the Markov process with unobserved (hidden) states.

2. A Markov process is referred as memory-less process which satisfies Markov property.

3. Markov property states that the conditional probability distribution of future states of a process depends only on present state, not on the sequence of events.

4. The Hidden Markov Model is one of the most popular graphical model

5. Examples of Markov process:

    a. Measurement of weather pattern.

    b. Daily stock market prices.

6. HMM generally works on set of temporal data.

7. HMM is a variant of finite state machine have following things:

    a. A set of hidden states (W).

    b. An output alphabet of visible state (V).

    c. Transition probabilities (A).

    d. Output Probability (B).

    e. Initial state probability (π)

8. The current state is not observable, instead each state produces an output with a certain probability (B).

9. Usually states (W) and outputs (V) are understood

-----

10. So an HMM is said to be a triple (A, B, π)

EXPLANATION OF HMM:

1. HMM consists of two types of states, the hidden state (W) and visible state (V).

2. Transition probability is the probability of transitioning from one state to another in single step.

3. **Emission probability:** The conditional distributions of observed variables $P(V_n \mid Z_n)$

4. HMM have to address following issues:

   a. **Evaluation problem:**
   - at for a given model θ with W(hidden states), V(visible states), A_{ij} (transition probability) and V^T (sequence of visible symbol emitted)
   - What is probability that the visible states sequence V^T will be emitted by model θ
     i.e. $P(V^T / θ) = ?$

   b. **Decoding problem:**
   - W^T, the sequence of states generated by visible symbols sequence V^T has to be calculated.
     i.e. $W^T = ?$

   c. **Training problem:**
   - For a known set of hidden states W and visible states V, the training problem is to find transition probability A_{ij} and emission probability B_{jk} from training set.
     i.e. $A_{ij} = ? \& B_{jk} = ?$

EXAMPLE OF HMM:

Coin toss

1. Heads, tails sequence with 2 coins.

2. You are in a room with a wall.

3. Person behind the wall flips coin. tells the result.

4. Coin selection and toss is hidden.

5. Cannot observe events, only output(heads, tails) from events.

6. Problem is then to build a model to explain observed sequence of heads and tails.

Q11. **Using Bayesian classification and the given data classify the tuple (Rupesh, M, 1.73 m)**

Attribute	Value	Count			Probability		
		Short	Medium	Tall	Short	Medium	Tall
Gender	M	1	2	3	1/4	2/7	3/4
	F	3	5	1	3/4	5/7	1/4
Height (range)	(0, 1.6)	2	0	0	2/4	0	0
	(1.6, 1.7)	2	0	0	2/4	0	0
	(1.7, 1.8)	0	3	0	0	3/7	0
	(1.8, 1.9)	0	3	0	0	3/7	0
	(1.9, 2)	0	1	2	0	1/7	2/4
	(2, ∞)	0	0	2	0	0	2/4

Ans:

Scanned by CamScanner

We have given Condition Probability Table (CPT).

Tuple to be classified = {Rupesh, M, 1.73m}

We will use some part of table which is red boxed as following to find probability of Short, Medium and Tall.

Attribute	Value	Count			Probability		
		Short	Medium	Tall	Short	Medium	Tall
Gender	M	1	2	3	1/4	2/7	3/4
	F	3	5	1	3/4	5/7	1/4
Height (range)	(0, 1.6)	2	0	0	2/4	0	0
	(1.6, 1.7)	2	0	0	2/4	0	0
	(1.7, 1.8)	0	3	0	0	3/7	0
	(1.8, 1.9)	0	3	0	0	3/7	0
	(1.9, 2)	0	1	2	0	1/7	2/4
	(2, ∞)	0	0	2	0	0	2/4

(There is no need to draw the above table again. The above table is just for understanding purpose)

Here, n = 15 (summing up all numbers in red box.)

Probability (Short)	= (1+3)/9	=0.45
Probability (Medium)	= (2+5)/9	=0.78
Probability (Tall)	= (3+1)/9	=0.45

Here, Short, Medium and Tall are class attributes values.

Now we will find probability of tuple with every value of class attribute which is Short, Medium and Tall.

Probability(tuple | Short)	= P[M | Short] × P[Height, Short]
Here, P[M | Short]	= probability of M with Short
and P[Height | Short]	= probability of height with Short

Probability(tuple | Short) = P[M | short] × P[(1.7 – 1.8) | Short]

(As height in given tuple is 1.73m which lies in (1.7 – 1.8) range of height.)

(Now we have to get those values from Probability part of table)

Probability(tuple | Short) $= \frac{1}{4} \times 0 = 0$

Probability(tuple | Medium) = P[M | Medium] × P[Height | Medium] $= \frac{2}{7} \times \frac{3}{7} = 0.43$

Probability(tuple | Tall) = P[M | Tall] × P[Height | Tall] $= \frac{3}{4} \times 0 = 0$

Now we have to find likelihood of tuple for all class attribute values.

Likelihood of Short = P[tuple | Short] × P[Short] = 0 × 0.45

Likelihood of Medium = P[tuple | Medium] × P[Medium] = 0.43 × 0.78 = 0.34

Likelihood of Tall = P[tuple | Tall] × P[Tall] = 0 × 0.45 = 0

Now we have to calculate estimate of likelihood values as following,

Estimate = Likelihood of Short + Likelihood of Medium + Likelihood of Tall

= 0 + 0.34 + 0 = 0.34

Now we will find actual probability for tuple using Naïve Bayes formula as following

$$p(x|y) = \frac{P(y|x) \times p(x)}{P(y)}$$

Calculating actual probability for all class attribute values,

1. **Short:**

$$P(Short \mid tuple) = \frac{P(tuple \mid Short) \times p(Short)}{P(tuple)} = \frac{0 \times 0.45}{0.34} = 0$$

2. **Medium:**

$$P(Medium \mid tuple) = \frac{P(tuple \mid Medium) \times p(Medium)}{P(tuple)} = \frac{0.43 \times 0.78}{0.34} = \frac{0.34}{0.34} = 1$$

3. **Tall:**

$$P(Tall \mid tuple) = \frac{P(tuple \mid Tall) \times p(Tall)}{P(tuple)} = \frac{0 \times 0.45}{0.34} = 0$$

By comparing actual probability of all three values, we can say that Rupesh height is of Medium.

(We will choose largest value from these actual probability values)

Important Tip:

May be some time in exam they can ask like find probability of tuple using following data –

Name	Gender	Height	Class
A	Female	1.6m	Short
B	Male	2.0m	Tall
C	Female	1.9m	Medium
D	Female	1.85m	Medium
E	Male	2.8m	Tall
F	Male	1.7m	Short
G	Male	1.8m	Medium
H	Female	1.6m	Short
I	Female	1.65m	Short

Now we have to construct a table as we have in our solved sum.

There are three distinct values in Class Attribute as Short, Medium and Tall.

There are two distinct values in Gender Attribute as Male and Female.

There are variable values in Height Attribute there we will make them as range values. As from lowest height to highest height. As (0 – 1.6m), (1.6m – 1.7m), and so on.

	Attributes	Values			Probability		
		Short	Medium	Tall	Short	Medium	Tall
Gender	Male						
	Female						

Height	(0 – 1.6m)						
	(1.6m – 1.7m)						
	(1.7m – 1.8m)						
	(1.8m – 1.9m)						
	(1.9m – 2.0m)						
	(2.0m - ∞)						

By seeing given data, We have to find how many tuples contains (Male and Short) value.

Same for (Male and Medium) values and (Male for Tall) values.

The above step will also applicable for tuples containing Female value

Now we will go through each touple and check their height value and find that in which range they come and increase the count for it in their range part.

Now we will find values for probability section

For probability of Gender values –

We will count all values for Short, Medium and Tall as following,

There is count of 4 for Short in Gender part

	Short
Male	1
Female	3
Total	4

For probability of Male = count of Male in Short part / count of all short values = 1 / 4

For probability of Female = count of Female in Short part / count of all short values = 3 / 4

There is count of 3 for Medium in Gender part

	Medium
Male	1
Female	2
Total	3

For probability of Male = count of Male in Medium part / count of all Medium values = 1 / 3

For probability of Female = count of Female in Medium part / count of all Medium values = 2 / 3

There is count of 4 for Tall in Gender part

	Tall
Male	2
Female	0
Total	2

Scanned by CamScanner

For probability of Male = count of Male in Tall part / count of all Tall values = 2 / 2

For probability of Female = count of Female in Tall part / count of all Tall values = 0 / 4 = 0

The same process can be applied for Height part.

Now we can fill the table of Condition Probability as following

	Attributes	Values			Probability		
		Short	Medium	Tall	Short	Medium	Tall
Gender	Male	1	1	2	1/4	1/3	2/2
	Female	3	2	0	3/4	2/3	0
Height	(0 – 1.6m)	2	0	0	2/4	0	0
	(1.6m – 1.7m)	2	0	0	2/4	0	0
	(1.7m – 1.8m)	0	1	0	0	1/3	0
	(1.8m – 1.9m)	0	2	0	0	2/3	0
	(1.9m – 2.0m)	0	0	1	0	0	1/2
	(2.0m - ∞)	0	0	1	0	0	1/2

Scanned by CamScanner

CHAP - 6: DIMENSIONALITY REDUCTION

Q1. Explain in detail Principal Component Analysis for Dimension Reduction

Ans: [10M – May16, Dec16 & Dec

PRINCIPAL COMPONENT ANALYSIS (PCA) FOR DIMENSION REDUCTION:

1. The main idea of PCA is to reduce dimensionality from a given data sets.
2. Given data set consists of many variables, correlated to each other either heavily or lightly.
3. Reducing is don while retaining the variation present in data set up to a maximum extent.
4. The same is done by transforming variables to a new set of variables which are known as Princi Components (PC)
5. PC are orthogonal, ordered such that retention of variation present in original components decrea as we move down in order.
6. So, in this way, the first principal component will have maximum variation that was present orthogonal component.
7. Principal Components are Eigen vectors of covariance matrix and hence they are called as orthogor
8. The data set on which PCA is to be applied must be scaled
9. The result of PCA is sensitive to relative scaling.
10. **Properties of PC:**
    a. PC are linear combination of original variables.
    b. PC are orthogonal.
    c. Variation present in PC's decreases as we move from first PC to last PC.

IMPLEMENTATION:

I) Normalize the data:

1. Data as input to PCA process must be normalised to work PCA properly.
2. This can be done by subtracting the respective means from numbers in respective columns.
3. If we have two dimensions X and Y then for all X becomes x- and Y becomes y-
4. The result give us a dataset whose means is zero.

II) Calculate covariance matrix:

1. Since we have taken 2 dimensional dataset, the covariance matrix will be
2. Matrix (covariance) = $\begin{bmatrix} Var(X1) & Cov(X1, X2) \\ Cov(X2, X1) & Var(X2) \end{bmatrix}$

III) Finding Eigen values and Eigen Vectors:

1. In this step we have to find Eigen values and Eigen vectors for covariance matrix.
2. It is possible because it is square matrix.
3. The λ will be Eigen value of matrix A.
4. If it satisfies following condition: $det(\lambda I - A) = 0$
5. Then we can find Eigen vector for each Eigen value λ by calculating: $(\lambda I - A)v = 0$

IV) Choosing components and forming features vectors:

1. We order the Eigen values from highest to lowest.
2. So we get components in order

Scanned by CamScanner

3. If a data set have n variables then we will have n Eigen values and Eigen vectors.
4. It turns out that Eigen vector with highest Eigen values is PC of dataset.
5. Now we have to decide how much Eigen values for further processing.
6. We choose first p Eigen values and discard others.
7. We do lose out some information in this process.
8. But if Eigen values are small, we do not lose much.
9. Now we form a feature vector.
10. Since we are working on 2D data, we can choose either greater Eigen value or simply take both
11. Feature vector = (Eig1, Eig2)

V) Forming Principal Components:

1. In this step, we develop Principal Component based on data from previous steps.
2. We take transpose of feature vector and transpose of scaled dataset and multiply it to get Principal Component.

New Data = Feature Vector^T X Scaled Dataset^T

New Data = Matrix of Principal Component

Q2.   Describe the two methods for reducing dimensionality

Ans: [5M | May17]

DIMENSIONALITY REDUCTION:

1. Dimension reduction refers to process of converting a set of data having vast dimensions into data with lesser dimension ensuring that it conveys similar information concisely.
2. This techniques are typically used while solving machine learning problems to obtain better features for classification.
3. Dimensions can be reduced by:
    a. Combining features using a linear or non-linear transformation.
    b. Selecting a subset of features.

METHODS FOR REDUCING DIMENSIONALITY:

I) Feature Selection:

1. It deals with finding k and d dimensions
2. It gives most information and discard the (d – k) dimensions.
3. It try to find subset of original variables.
4. There are three strategies of feature selection:
    a. Filter.
    b. Wrapper.
    c. Embedded.

II) Feature extraction:

1. It deals with finding k dimensions from combinations of d dimensions.
2. I transforms the data in high dimensional space to data in few dimensional space.
3. The data transformation may be linear like PCA or non-linear like Laplacian Eigen maps.

III) Missing values:

1. Given a sample training sample set of features.

2. Among the available features a particular features has many missing values.

3. The feature with more missing value will contribute less to classification process.

4. This features can be eliminated.

IV) Low variance:

1. Consider that particular feature has constant values for all training sample.

2. That means variance of features for different sample is comparatively less

3. This implies that feature with constant values or low variance have less impact on classification

4. This features can be eliminated.

Q3. What is independent component analysis?

Ans: [5M | May18 & Dec18]

INDEPENDENT COMPONENT ANALYSIS:

1. Independent component analysis (ICA) is a statistical and computational technique

2. It is used for revealing hidden factors that underlie sets of random variables, measurements, or signals

3. ICA defines a generative model for the observed multivariate data,

4. Given data is typically a large database of samples.

5. In the model, the data variables are assumed to be linear mixtures of some unknown latent variables,

6. The mixing system is also unknown.

7. The latent variables are assumed non-Gaussian and mutually independent,

8. This latent variables are called the independent components of the observed data.

9. These independent components, also called sources or factors, can be found by ICA.

10. ICA is superficially related to principal component analysis and factor analysis.

11. ICA is a much more powerful technique

12. However, capable of finding the underlying factors or sources when these classic methods fail completely.

13. The data analysed by ICA could originate from many different kinds of application fields.

14. It includes digital images, document databases, economic indicators and psychometric measurements.

15. In many cases, the measurements are given as a set of parallel signals or time series

16. The term blind source separation is used to characterize this problem.

EXAMPLES:

1. Mixtures of simultaneous speech signals that have been picked up by several microphones

2. Brain waves recorded by multiple sensors

3. Interfering radio signals arriving at a mobile phone

4. Parallel time series obtained from some industrial process.

AMBIGUITIES:

1. Can't determine the variances (energies) of the IC's

2. Can't determine the order of the IC's

APPLICATION DOMAINS OF ICA:

1. Image de-noising.
2. Medical signal processing.
3. Feature extraction, face recognition.
4. Compression, redundancy reduction.
5. Scientific Data Mining.

Q4. Use Principal Component analysis (PCA) to arrive at the transformed matrix for the given matrix A.

$A^T =$

2	1	0	-1
4	3	1	0.5

Ans: **[10M | May17 & May18]**

Formula for finding Covariance values =>

Covariance $= \sum_{i=1}^{n} \frac{(x-\bar{x}) \times (y-\bar{y})}{n-1}$ ....[valid for (x, y) and (y, x)]

Here we have Orthogonal Transformation A^T

We will consider upper row as x and lower row as y.

Here n = 4. (Total count of data points. Take count from either x row or y row).

Now, we will find $\bar{x}$ and $\bar{y}$ as following,

$\bar{x}$ = (sum of all values in x / count of all values in x)

$\bar{y}$ = (sum of all values in y / count of all values in y)

$\bar{x} = \frac{2+1+0+(-1)}{4} = \frac{2}{4} = 0.5$

$\bar{y} = \frac{4+3+1+0.5}{4} = \frac{8.5}{4} = 2.125$

Now we have to find values of $(x - \bar{x}), (y - \bar{y}), [(x - \bar{x}) \times (y - \bar{y})], (x - \bar{x})^2, (y - \bar{y})^2$

x	y	$(x-\bar{x})$	$(y-\bar{y})$	$[(x-\bar{x}) \times (y-\bar{y})]$	$(x-\bar{x})^2$	$(y-\bar{y})^2$
2	4	1.5	1.875	2.8185	2.25	3.5156
1	3	0.5	0.875	0.4375	0.25	0.7656
0	1	-0.5	-1.125	0.5625	0.25	1.2656
-1	0.5	-1.5	-1.625	2.4375	2.25	2.6404

Finding following values, $\sum[(x-\bar{x}) \times (y-\bar{y})], \sum(x-\bar{x})^2, \sum(y-\bar{y})^2$

$\sum[(x-\bar{x}) \times (y-\bar{y})] = 2.815 + 0.4375 + 0.5625 + 2.4375 = 6.256$

$\sum(x-\bar{x})^2 = 2.25 + 0.25 + 0.25 + 2.25 = 5$

$\sum(y-\bar{y})^2 = 3.5156 + 0.7656 + 1.2656 + 2.6404 = 8.1872$

Now we will find covariance values of (x, x)

Covariance(x, x) $= \sum_{i=1}^{n} \frac{(x-\bar{x})^2}{n-1} = \frac{5}{3} = 1.67$

Covariance(y,   y)   = $\sum_{i=1}^{n} \frac{(y-\bar{y})^2}{n-1}$ = $\frac{8.1872}{3}$ =   2.73

Covariance(x, y) = Covariance(y, x) $= \sum_{i=1}^{n} \frac{(x-\bar{x}) \times (y-\bar{y})}{n-1} = \frac{6.256}{3} = 2.09$

Scanned by CamScanner

Therefore putting these values in a matrix form as s,

$$s = \text{Covariance} = \begin{bmatrix} x & y \\ y & x \end{bmatrix} = \begin{bmatrix} 1.67 & 2.09 \\ 2.09 & 2.73 \end{bmatrix}$$

As given data is in Two dimensional form, there will be two Principal components.

Now we have to use Characteristics Equation $|s - \lambda I| = 0$ to find these Principal components

Here, s = Covariance in Matrix form

$I = \text{Identity Matrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

Putting these values in Characteristics equation,

$$\left| \begin{bmatrix} 1.67 & 2.09 \\ 2.09 & 2.73 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right| = 0$$

Getting determinant using above step,

$$\begin{vmatrix} 1.67 - \lambda & 2.09 \\ 2.09 & 2.73 - \lambda \end{vmatrix} = 0 \quad \text{.......................... (1)}$$

$(1.67 - \lambda) \times (2.73 - \lambda) - (2.09)^2 = 0$

$\lambda^2 - 4.4\lambda + 0.191 = 0$

$\lambda_1 = 4.3562$

$\lambda_2 = 0.0439$

Taking λ_1 and putting it in equation (1) to find factors a_{11} and a_{12} for Principal components

$$\begin{bmatrix} 1.67 - 4.3562 & 2.09 \\ 2.09 & 2.73 - 4.3562 \end{bmatrix} \times \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix} = 0$$

$$\begin{bmatrix} -2.6862 & 2.09 \\ 2.09 & -1.6262 \end{bmatrix} \times \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix} = 0$$

$-2.6862 \times a_{11} + 2.09 \times a_{12} = 0 \quad \text{.......................... (2)}$

$2.09 \times a_{11} - 1.6262 \times a_{12} = 0 \quad \text{.......................... (3)}$

Let's find value for a_{12} (you can also take a_{11}) by dividing equation (2) by 2.09, we get

$a_{12} = \frac{2.6862}{2.09} \times a_{11}$

$a_{12} = 1.2853 \times a_{11}$

We can now use equation of orthogonal transformation relation which is ->

$a_{11}^2 + a_{12}^2 = 1 \quad \text{.......................... (4)}$

Substituting value of a_{12} in above equation (4), we get

$a_{11}^2 + (1.2853 \times a_{11})^2 = 1$

$a_{11}^2 + 1.5620 \times a_{11}^2 = 1$

$2.5620 \times a_{11}^2 = 1$

$a_{11}^2 = 0.40$

$a_{11} = 0.64$

Scanned by CamScanner

putting a_{11} in equation (4)

$(0.64)^2 + a_{12}^2 = 1$

$0.4096 + a_{12}^2 = 1$

$a_{12}^2 = 1 - 0.4096$

$a_{12}^2 = 0.5904$

$a_{12} = 0.7684$

Now taking λ_1 and putting it in equation (1) to find factors a_{11} and a_{12} for Principal components

$$\begin{bmatrix} 1.67 - 0.0439 & 2.09 \\ 2.09 & 2.73 - 0.0439 \end{bmatrix} \times \begin{bmatrix} a_{21} \\ a_{22} \end{bmatrix} = 0$$

$$\begin{bmatrix} 1.6261 & 2.09 \\ 2.09 & 2.6861 \end{bmatrix} \times \begin{bmatrix} a_{21} \\ a_{22} \end{bmatrix} = 0$$

$1.6261 \times a_{21} + 2.09 \times a_{22} = 0$ .... (5)

$2.09 \times a_{21} + 2.6861 \times a_{22} = 0$ .... (6)

Let's find value for a_{22} (you can also take a_{21}) by dividing equation (5) by 2.09, we get

$a_{22} = \frac{1.6261}{2.09} \times a_{21}$

$a_{22} = 0.7781 \times a_{21}$

Substituting value of a_{12} in equation (4), we get

$a_{21}^2 + (0.7781 \times a_{21})^2 = 1$

$a_{21}^2 + 0.6055 \times a_{21}^2 = 1$

$1.6055 \times a_{21}^2 = 1$

$a_{21}^2 = 0.6229$

$a_{21} = 0.7893$

Putting a_{21} in equation (4)

$(0.7893)^2 + a_{22}^2 = 1$

$0.6230 + a_{22}^2 = 1$

$a_{22}^2 = 1 - 0.6230$

$a_{22}^2 = 0.377$

$a_{22} = 0.6141$

Therefore using values of $a_{11}, a_{12}, a_{21}, a_{22}$, the Principal Components are

$Z_1 = a_{11}x_1 + a_{12}x_2$

$Z_1 = 0.64x_1 + 0.7684x_2$

$Z_2 = a_{21}x_1 + a_{22}x_2$

$Z_2 = 0.7893x_1 + 0.6141x_2$

CHAP - 7: LEARNING WITH CLUSTERING

Q1. Describe the essential steps of K-means algorithm for clustering analysis. Also, explain how K-means

Q2. Explain K-means clustering algorithm giving suitable example. Also, explain how K-means clustering differs from hierarchical clustering.

[5 - 10M | Dec16 & Dec18]

Ans:

K-MEANS CLUSTERING:

1. K-means is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem.

2. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori.

3. The main idea is to define k centres, one for each cluster.

4. These centres should be placed in a cunning way because of different location causes different result.

5. So, the better choice is to place them as much as possible far away from each other.

6. The next step is to take each point belonging to a given data set and associate it to the nearest center.

7. When no point is pending, the first step is completed and an early group age is done.

8. At this point we need to re-calculate k new centroids as barycentre of the clusters resulting from the previous step.

9. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center.

10. A loop has been generated.

11. As a result of this loop we may notice that the k centres change their location step by step until no more changes are done or in other words centres do not move any more.

12. Finally, this algorithm aims at minimizing an objective function know as squared error function given by:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} \left(\left\| x_i - v_j \right\| \right)^2$$

13. Where $\|x_i - v_j\|$ is the Euclidean distance between x_i and v_j.

    c_i is the number of data points in i^{th} cluster.

    c is the number of cluster centres.

ALGORITHM STEPS:

1. Let $X = \{x_1, x_2, x_3, \ldots, x_n\}$ be the set of data points and $V = \{v_1, v_2, \ldots, v_c\}$ be the set of centres.

2. Randomly select c cluster centres.

3. Calculate the distance between each data point and cluster centres.

4. Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centres.

5. Recalculate the new cluster center using:

$$v_i = (1/c_i) \sum_{j=1}^{c_i} x_i$$

Where, c_i represents the number of data points in i^{th} cluster.

6. Recalculate the distance between each data point and new obtained cluster centres.

7. If no data point was reassigned then stop, otherwise repeat from step 3.

ADVANTAGES:

1. Fast, robust and easier to understand.

2. Relatively efficient: $O(tknd)$, where n is objects, k is clusters, d is dimension of each object, and t is iterations.

3. Gives best result when data set are distinct or well separated from each other.

DISADVANTAGES:

1. Applicable only when mean is defined i.e. fails for categorical data.

2. Unable to handle noisy data and outliers.

3. Algorithm fails for non-linear data set.

DIFFERENCE BETWEEN K MEANS AND HIERARCHICAL CLUSTERING:

1. Hierarchical clustering can't handle big data well but K Means clustering can

2. This is because the time complexity of K Means is linear i.e. $O(n)$ while that of hierarchical clustering is quadratic i.e. $O(n^2)$.

3. In K Means clustering, since we start with random choice of clusters.

4. The results produced by running the algorithm multiple times might differ.

5. While results are reproducible in Hierarchical clustering.

6. K Means is found to work well when the shape of the clusters is hyper spherical (like circle in 2D, sphere in 3D).

7. K Means clustering requires prior knowledge of K i.e. no. of clusters you want to divide your data into.

8. You can stop at whatever number of clusters you find appropriate in hierarchical clustering by interpreting the dendrogram

----

Q3. Hierarchical clustering algorithms.

Ans: [10M | Dec17]

HIERARCHICAL CLUSTERING ALGORITHMS:

1. Hierarchical clustering, also known as hierarchical cluster analysis.

2. It is an algorithm that groups similar objects into groups called clusters.

3. The endpoint is a set of clusters.

4. Each cluster is distinct from each other cluster.

5. The objects within each cluster are broadly similar to each other.

6. This algorithm starts with all the data points assigned to a cluster of their own.

7. Two nearest clusters are merged into the same cluster.

8. In the end, this algorithm terminates when there is only a single cluster left.

9. Hierarchical clustering does not require us to prespecify the number of clusters

10. Most hierarchical algorithms are deterministic.

11. Hierarchical clustering algorithm is of two types:

   a. Divisive Hierarchical clustering algorithm.

   b. Agglomerative Hierarchical clustering algorithm.

12. Both this algorithm are exactly reverse of each other.

DIVISIVE HIERARCHICAL CLUSTERING ALGORITHM OR DIANA (DIVISIVE ANALYSIS):

1. Here we assign all of the observations to a single cluster and then partition the cluster to two least similar clusters.

2. Finally, we proceed recursively on each cluster until there is one cluster for each observation.

3. There is evidence that divisive algorithms produce more accurate hierarchies than agglomerative algorithms in some circumstances but is conceptually more complex.

4. **Advantages:**

   a. More efficient if we do not generate a complete hierarchy.

   b. Run much faster than HAC algorithms.

5. **Disadvantages:**

   a. Algorithm will not identify outliers.

   b. Restricted to data which has the notion of a centre (centroid).

6. **Example:**

   a. K – Means algorithm.

   b. K – Medoids algorithm.

AGGLOMERATIVE HIERARCHICAL CLUSTERING ALGORITHM OR AGNES (AGGLOMERATIVE NESTING):

1. In agglomerative or bottom-up clustering method we assign each observation to its own cluster.

2. Then, compute the similarity (e.g., distance) between each of the clusters and join the two most similar clusters.

3. Finally, repeat steps 2 and 3 until there is only a single cluster left.

4. **Advantages:**

   a. No apriori information about the number of clusters required.

   b. Easy to implement and gives best result in some cases.

5. **Disadvantages:**

   a. No objective function is directly minimized

   b. Algorithm can never undo what was done previously.

   c. Sometimes it is difficult to identify the correct number of clusters by the dendrogram.

6. **Types of Agglomerative Hierarchical clustering:**

   a. Single-nearest distance or single linkage.

   b. Complete-farthest distance or complete linkage.

   c. Average-average distance or average linkage.

   d. Centroid distance.

   e. Ward's method - sum of squared Euclidean distance is minimized.

Q4. What is the role of radial basis function In separating nonlinear patterns.

Q5. Write short note on - Radial Basis functions

Ans:

[10M | May18 & Dec18]

RADIUS BASIS FUNCTION:

1. A radial basis function (RBF) is a real-valued function ϕ whose value depends only on the distance from the origin,

   So that $\phi(x) = \phi(||x||)$

2. Alternatively on the distance from some other point c, called a centre,

3. So any function that satisfies the property is a radial function.

4. The norm is usually Euclidean distance, although other distance functions are also possible.

5. Sums of radial basis functions are typically used to approximate given functions.

6. This approximation process can also be interpreted as a simple kind of neural network

7. RBFs are also used as a kernel in support vector classification

8. Commonly used types of radial basis functions:

   a. **Gaussian:**

   $$\phi(r) = e^{-(\varepsilon r)^2}$$

   b. **Multiquadratic**

   $$\phi(r) = \sqrt{1 + (\varepsilon r)^2}$$

   c. **Inverse quadratic**

   $$\phi(r) = \frac{1}{1 + (\varepsilon r)^2}$$

   d. **Inverse Multiquadratic**

   $$\phi(r) = \frac{1}{\sqrt{1 + (\varepsilon r)^2}}$$

9. Radial basis function network is an artificial neural network that uses radial basis functions as activation functions.

10. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters.

11. Radial basis function networks have many uses, including function approximation, time series prediction, classification, and system control.

12. Radial basis function networks is composed of input, hidden, and output layer. RBNN is strictly limited to have exactly one hidden layer. We call this hidden layer as feature vector.

13. Radial basis function networks increases dimension of feature vector.

Scanned by CamScanner
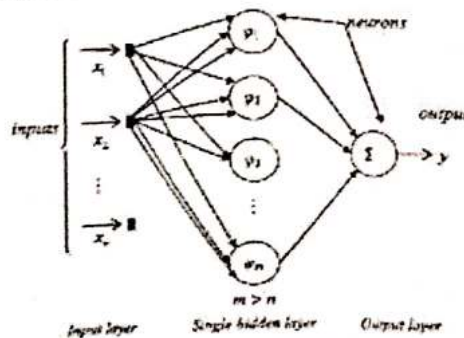
14. The hidden units provide a set of functions that constitute an arbitrary basis for the input patterns

15. Hidden units are known as radial centres and represented by the vectors c1, c2, ···, ch

16. Transformation from input space to hidden unit space is nonlinear whereas transformation from hidden unit space to output space is linear

17. dimension of each centre for a p input network is p × 1

18. The radial basis functions in the hidden layer produces a significant non-zero response only when the input falls within a small localized region of the input space.

19. Each hidden unit has its own receptive field in input space.

20. An input vector xi which lies in the receptive field for centre cj, would activate cj and by proper choice of weights the target output is obtained.

21. The output is given as:

$$y = \sum_{j=1}^{h} \phi_j w_j, \qquad \phi_j = \phi(\|x - c_j\|)$$

   w_j: weight of jth centre,

   φ: some radial function.

22. Learning in RBFN Training of RBFN requires optimal selection of the parameters vectors ci and wi, i = 1, ··· h.

23. Both layers are optimized using different techniques and in different time scales.

24. Following techniques are used to update the weights and centres of a RBFN.

    a. Pseudo-Inverse Technique (Off line)

    b. Gradient Descent Learning (On line)

    c. Hybrid Learning (On line)

Q6. What are the requirements of clustering algorithms?

Ans: [5M | May18]

REQUIREMENTS OF CLUSTERING:

1. **Scalability:** We need highly scalable clustering algorithms to deal with large databases & learning system.

2. **Ability to deal with different kinds of attributes:** Algorithms should be capable to be applied on any kind of data such as interval-based (numerical) data, categorical, and binary data.

3. **Discovery of clusters with attribute shape:** The clustering algorithm should be capable of detecting clusters of arbitrary shape. They should not be bounded to only distance measures that tend to find spherical cluster of small sizes.

4. **High dimensionality:** The clustering algorithm should not only be able to handle low-dimensional data but also the high dimensional space.

5. **Ability to deal with noisy data:** Databases contain noisy, missing or erroneous data. Some algorithms are sensitive to such data and may lead to poor quality clusters.

6. **Interpretability:** The clustering results should be interpretable, comprehensible, and usable.

Scanned by CamScanner

Q7. Apply K-means algorithm on given data for k=3. Use $C_1(2)$, $C_2(16)$ and $C_3(38)$ as initial cluster centres.

Data: 2, 4, 6, 3, 31, 12, 15, 16, 38, 35, 14, 21, 23, 25, 30

Ans:

[10M | May16 & Dec16]

Number of clusters k = 3

Initial cluster centre for $C_1 = 2$, $C_2 = 16$, $C_3 = 38$

We will check distance between data points and all cluster centres. We will use Euclidean Distance formula for finding distance.

Distance [x, a] = $\sqrt{(x - a)^2}$

OR

Distance [(x, y), (a, b)] = $\sqrt{(x - a)^2 + (y - b)^2}$

As given data is not in pair, we will use first formula of Euclidean Distance.

Finding Distance between data points and cluster centres.

We will use following notations for calculating Distance:

D_1 = Distance from cluster C_1 centre

D_2 = Distance from cluster C_2 centre

D_3 = Distance from cluster C_3 centre

2:

$D_1(2, 2) = \sqrt{(x - a)^2} = \sqrt{(2 - 2)^2} = 0$

$D_2(2, 16) = \sqrt{(x - a)^2} = \sqrt{(2 - 16)^2} = 14$

$D_3(2, 38) = \sqrt{(x - a)^2} = \sqrt{(2 - 38)^2} = 34$

Here 0 is smallest distance so Data point 2 belongs to C_1

4:

$D_1(4, 2) = \sqrt{(x - a)^2} = \sqrt{(4 - 2)^2} = 2$

$D_2(4, 16) = \sqrt{(x - a)^2} = \sqrt{(4 - 16)^2} = 12$

$D_3(4, 38) = \sqrt{(x - a)^2} = \sqrt{(4 - 38)^2} = 34$

Here 2 is smallest distance so Data point 4 belongs to C_1

6:

$D_1(6, 2) = \sqrt{(x - a)^2} = \sqrt{(6 - 2)^2} = 4$

$D_2(6, 16) = \sqrt{(x - a)^2} = \sqrt{(6 - 16)^2} = 10$

$D_3(6, 38) = \sqrt{(x - a)^2} = \sqrt{(6 - 38)^2} = 32$

Here 4 is smallest distance so Data point 6 belongs to C_1

3:

$D_1(3, 2) = \sqrt{(x - a)^2} = \sqrt{(3 - 2)^2} = 1$

$D_2(3, 16) = \sqrt{(x - a)^2} = \sqrt{(3 - 16)^2} = 13$

$D_3(3, 38) = \sqrt{(x - a)^2} = \sqrt{(3 - 38)^2} = 35$

Here 1 is smallest distance so Data point 3 belongs to C_1

Scanned by CamScanner

31:

$D_1(31, 2) = \sqrt{(x-a)^2} = \sqrt{(31-2)^2} = 29$

$D_2(31, 16) = \sqrt{(x-a)^2} = \sqrt{(31-16)^2} = 15$

$D_3(31, 38) = \sqrt{(x-a)^2} = \sqrt{(31-38)^2} = 7$

Here 0 is smallest distance so Data point 31 belongs to C_3

12:

$D_1(12, 2) = \sqrt{(x-a)^2} = \sqrt{(12-2)^2} = 10$

$D_2(12, 16) = \sqrt{(x-a)^2} = \sqrt{(12-16)^2} = 4$

$D_3(12, 38) = \sqrt{(x-a)^2} = \sqrt{(12-38)^2} = 26$

Here 0 is smallest distance so Data point 12 belongs to C_2

15:

$D_1(15, 2) = \sqrt{(x-a)^2} = \sqrt{(15-2)^2} = 13$

$D_2(15, 16) = \sqrt{(x-a)^2} = \sqrt{(15-16)^2} = 1$

$D_3(15, 38) = \sqrt{(x-a)^2} = \sqrt{(15-38)^2} = 23$

Here 1 is smallest distance so Data point 15 belongs to C_2

16:

$D_1(16, 2) = \sqrt{(x-a)^2} = \sqrt{(16-2)^2} = 14$

$D_2(16, 16) = \sqrt{(x-a)^2} = \sqrt{(16-16)^2} = 0$

$D_3(16, 38) = \sqrt{(x-a)^2} = \sqrt{(16-38)^2} = 22$

Here 0 is smallest distance so Data point 4 belongs to C_2

38:

$D_1(38, 2) = \sqrt{(x-a)^2} = \sqrt{(38-2)^2} = 36$

$D_2(38, 16) = \sqrt{(x-a)^2} = \sqrt{(38-16)^2} = 22$

$D_2(38, 38) = \sqrt{(x-a)^2} = \sqrt{(38-38)^2} = 0$

Here 0 is smallest distance so Data point 38 belongs to C_3

35:

$D_1(35, 2) = \sqrt{(x-a)^2} = \sqrt{(35-2)^2} = 33$

$D_2(35, 16) = \sqrt{(x-a)^2} = \sqrt{(35-16)^2} = 21$

$D_3(35, 38) = \sqrt{(x-a)^2} = \sqrt{(35-38)^2} = 3$

Here 3 is smallest distance so Data point 35 belongs to C_3

14:

$D_1(14, 2) = \sqrt{(x-a)^2} = \sqrt{(14-2)^2} = 12$

$D_2(14, 16) = \sqrt{(x-a)^2} = \sqrt{(14-16)^2} = 2$

$D_3(14, 38) = \sqrt{(x-a)^2} = \sqrt{(14-38)^2} = 24$

Here 2 is smallest distance so Data point 14 belongs to C_2

21:

$D_1(21, 2) = \sqrt{(x-a)^2} = \sqrt{(21-2)^2} = 19$

$D_2(21, 16) = \sqrt{(x-a)^2} = \sqrt{(21-16)^2} = 5$

$D_3(21, 38) = \sqrt{(x-a)^2} = \sqrt{(21-38)^2} = 17$

Here 5 is smallest distance so Data point 21 belongs to C_2

23:

$D_1(23, 2) = \sqrt{(x-a)^2} = \sqrt{(23-2)^2} = 21$

$D_2(23, 16) = \sqrt{(x-a)^2} = \sqrt{(23-16)^2} = 7$

$D_3(23, 38) = \sqrt{(x-a)^2} = \sqrt{(23-38)^2} = 15$

Here 7 is smallest distance so Data point 23 belongs to C_2

25:

$D_1(25, 2) = \sqrt{(x-a)^2} = \sqrt{(25-2)^2} = 23$

$D_2(25, 16) = \sqrt{(x-a)^2} = \sqrt{(25-16)^2} = 9$

$D_3(25, 38) = \sqrt{(x-a)^2} = \sqrt{(25-38)^2} = 13$

Here 9 is smallest distance so Data point 25 belongs to C_2

30:

$D_1(30, 2) = \sqrt{(x-a)^2} = \sqrt{(30-2)^2} = 28$

$D_2(30, 16) = \sqrt{(x-a)^2} = \sqrt{(30-16)^2} = 14$

$D_3(30, 38) = \sqrt{(x-a)^2} = \sqrt{(30-38)^2} = 8$

Here 8 is smallest distance so Data point 30 belongs to C_3

The clusters will be,

$C_1 = \{2, 4, 6, 3\}$,

$C_2 = \{12, 15, 16, 14, 21, 23, 25\}$,

$C_3 = \{31, 38, 35, 30\}$

Now we have to recalculate the centre of these clusters as following

$C_1 = \frac{2+4+6+3}{4} = \frac{15}{4} = 3.75$ (we can round off this value to 4 also)

$C_2 = \frac{12+15+16+14+21+23+25}{7} = \frac{126}{7} = 18$

$C_3 = \frac{31+38+35+30}{4} = \frac{134}{4} = 33.5$ (we can round of this value to 34 also)

Now we will again calculate distance from each data point to all new cluster centres,

2:

$D_1(2, 4) = \sqrt{(x-a)^2} = \sqrt{(2-4)^2} = 2$

$D_2(2, 18) = \sqrt{(x-a)^2} = \sqrt{(2-18)^2} = 16$

$D_3(2, 34) = \sqrt{(x-a)^2} = \sqrt{(2-34)^2} = 32$

Here 2 is smallest distance so Data point 2 belongs to C_1

4:

$D_1(4, 4) = \sqrt{(x-a)^2} = \sqrt{(4-4)^2} = 0$

$D_2(4, 18) = \sqrt{(x-a)^2} = \sqrt{(4-18)^2} = 14$

$D_3(4, 34) = \sqrt{(x-a)^2} = \sqrt{(4-34)^2} = 39$

Here 0 is smallest distance so Data point 4 belongs to C_1

Scanned by CamScanner

6:

$D_1(6, 4) = \sqrt{(x - a)^2} = \sqrt{(6 - 4)^2} = 2$

$D_2(6, 18) = \sqrt{(x - a)^2} = \sqrt{(6 - 18)^2} = 12$

$D_3(6, 34) = \sqrt{(x - a)^2} = \sqrt{(6 - 34)^2} = 28$

Here 2 is smallest distance so Data point 6 belongs to C_1

3:

$D_1(3, 4) = \sqrt{(x - a)^2} = \sqrt{(3 - 4)^2} = 1$

$D_2(3, 18) = \sqrt{(x - a)^2} = \sqrt{(3 - 18)^2} = 15$

$D_3(3, 34) = \sqrt{(x - a)^2} = \sqrt{(3 - 34)^2} = 31$

Here 1 is smallest distance so Data point 3 belongs to C_1

31:

$D_1(31, 4) = \sqrt{(x - a)^2} = \sqrt{(31 - 4)^2} = 27$

$D_2(31, 18) = \sqrt{(x - a)^2} = \sqrt{(31 - 18)^2} = 13$

$D_3(31, 34) = \sqrt{(x - a)^2} = \sqrt{(31 - 34)^2} = 3$

Here 3 is smallest distance so Data point 31 belongs to C_3

12:

$D_1(12, 4) = \sqrt{(x - a)^2} = \sqrt{(12 - 4)^2} = 8$

$D_2(12, 18) = \sqrt{(x - a)^2} = \sqrt{(12 - 18)^2} = 6$

$D_3(12, 34) = \sqrt{(x - a)^2} = \sqrt{(12 - 34)^2} = 22$

Here 6 is smallest distance so Data point 12 belongs to C_2

15:

$D_1(15, 4) = \sqrt{(x - a)^2} = \sqrt{(15 - 4)^2} = 11$

$D_2(15, 18) = \sqrt{(x - a)^2} = \sqrt{(15 - 18)^2} = 3$

$D_3(15, 34) = \sqrt{(x - a)^2} = \sqrt{(15 - 34)^2} = 19$

Here 3 is smallest distance so Data point 15 belongs to C_2

16:

$D_1(16, 4) = \sqrt{(x - a)^2} = \sqrt{(16 - 4)^2} = 12$

$D_2(16, 18) = \sqrt{(x - a)^2} = \sqrt{(16 - 18)^2} = 2$

$D_3(16, 34) = \sqrt{(x - a)^2} = \sqrt{(16 - 34)^2} = 18$

Here 2 is smallest distance so Data point 4 belongs to C_2

38:

$D_1(38, 4) = \sqrt{(x - a)^2} = \sqrt{(38 - 4)^2} = 34$

$D_2(38, 18) = \sqrt{(x - a)^2} = \sqrt{(38 - 18)^2} = 20$

$D_3(38, 34) = \sqrt{(x - a)^2} = \sqrt{(38 - 34)^2} = 4$

Here 4 is smallest distance so Data point 38 belongs to C_3

35:

$D_1(35, 4) = \sqrt{(x - a)^2} = \sqrt{(35 - 4)^2} = 31$

$D_2(35, 18) = \sqrt{(x - a)^2} = \sqrt{(35 - 18)^2} = 17$

Scanned by CamScanner

$D_3(35, 34) = \sqrt{(x-a)^2} = \sqrt{(35-34)^2} = 1$

Here 1 is smallest distance so Data point 35 belongs to C_3

14:

$D_1(14, 4) = \sqrt{(x-a)^2} = \sqrt{(14-4)^2} = 10$

$D_2(14, 18) = \sqrt{(x-a)^2} = \sqrt{(14-18)^2} = 4$

$D_3(14, 34) = \sqrt{(x-a)^2} = \sqrt{(14-34)^2} = 20$

Here 4 is smallest distance so Data point 14 belongs to C_2

21:

$D_1(21, 4) = \sqrt{(x-a)^2} = \sqrt{(21-4)^2} = 17$

$D_2(21, 18) = \sqrt{(x-a)^2} = \sqrt{(21-18)^2} = 3$

$D_3(21, 34) = \sqrt{(x-a)^2} = \sqrt{(21-34)^2} = 13$

Here 3 is smallest distance so Data point 21 belongs to C_2

23:

$D_1(23, 4) = \sqrt{(x-a)^2} = \sqrt{(23-4)^2} = 19$

$D_2(23, 18) = \sqrt{(x-a)^2} = \sqrt{(23-18)^2} = 5$

$D_3(23, 34) = \sqrt{(x-a)^2} = \sqrt{(23-34)^2} = 11$

Here 5 is smallest distance so Data point 23 belongs to C_2

25:

$D_1(25, 4) = \sqrt{(x-a)^2} = \sqrt{(25-4)^2} = 21$

$D_2(25, 18) = \sqrt{(x-a)^2} = \sqrt{(25-18)^2} = 7$

$D_3(25, 34) = \sqrt{(x-a)^2} = \sqrt{(25-34)^2} = 9$

Here 7 is smallest distance so Data point 25 belongs to C_2

30:

$D_1(30, 4) = \sqrt{(x-a)^2} = \sqrt{(30-4)^2} = 26$

$D_2(30, 18) = \sqrt{(x-a)^2} = \sqrt{(30-18)^2} = 12$

$D_3(30, 34) = \sqrt{(x-a)^2} = \sqrt{(30-34)^2} = 4$

Here 4 is smallest distance so Data point 30 belongs to C_3

The updated clusters will be,

$C_1 = \{2, 4, 6, 3\}$,

$C_2 = \{12, 15, 16, 14, 21, 23, 25\}$,

$C_3 = \{31, 38, 35, 30\}$

We can see that there is no difference between previous clusters and these updated clusters, so we will stop the process here.

Finalised clusters –

$C_1 = \{2, 4, 6, 3\}$,

$C_2 = \{12, 15, 16, 14, 21, 23, 25\}$,

$C_3 = \{31, 38, 35, 30\}$

Scanned by CamScanner

Q8. Apply K-means algorithm on given, data for k=2. Use $C_1(2,4)$ & $C_2(6, 3)$ as initial cluster centre

Data : a(2,4), b(3,3), c(5,5), d(6,3), e(4,3), f(6,6)

[10M – Dec17]

Ans:

Number of clusters k = 2

Initial cluster centre for C_1 = (2, 4), C_2 = (6, 3)

We will check distance between data points and all cluster centres. We will use Euclidean Distance

formula for finding distance.

Distance [x, a] = $\sqrt{(x - a)^2}$

OR

Distance [(x, y), (a, b)] = $\sqrt{(x - a)^2 + (y - b)^2}$

As given data is in pair, we will use second formula of Euclidean Distance.

Finding Distance between data points and cluster centres.

We will use following notations for calculating Distance:

D_1 = Distance from cluster C_1 centre

D_2 = Distance from cluster C_2 centre

(2, 4):

$D_1[(2, 4), (2, 4)] = \sqrt{(x - a)^2 + (y - b)^2} = \sqrt{(2 - 2)^2 + (4 - 4)^2} = 0$

$D_2[(2, 4), (6, 3)] = \sqrt{(x - a)^2 + (y - b)^2} = \sqrt{(2 - 6)^2 + (4 - 3)^2} = 4.13$

Here 0 is smallest distance so Data point (2, 4) belongs to cluster C_1.

As Data point belongs to cluster C_1, we will recalculate the centre of cluster C_1 as following-

Using following formula for finding new centres of cluster =

Centre [(x. y), (a, b)] = $\left(\frac{x+a}{2}, \frac{y+b}{2}\right)$

Here, (x, y) = current data point

(a, b) = old centre of cluster

Updated Centre of cluster C_1 = $\left(\frac{x+a}{2}, \frac{y+b}{2}\right) = \left(\frac{2+2}{2}, \frac{4+4}{2}\right) = (2, 4)$

(3, 3):

$D_1[(3, 3), (2, 4)] = \sqrt{(x - a)^2 + (y - b)^2} = \sqrt{(3 - 2)^2 + (3 - 4)^2} = 1.42$

$D_2[(3, 3), (6, 3)] = \sqrt{(x - a)^2 + (y - b)^2} = \sqrt{(3 - 6)^2 + (3 - 3)^2} = 3$

Here 1.42 is smallest distance so Data point (3, 3) belongs to cluster C_1.

As Data point belongs to cluster C_1, we will recalculate the centre of cluster C_1 as following-

Updated Centre of cluster C_1 = $\left(\frac{x+a}{2}, \frac{y+b}{2}\right) = \left(\frac{3+2}{2}, \frac{3+4}{2}\right) = (2.5, 3.5)$

(5, 5):

$D_1[(5, 5), (2.5, 3.5)] = \sqrt{(x - a)^2 + (y - b)^2} = \sqrt{(5 - 2.5)^2 + (5 - 3.5)^2} = 2.92$

$D_2[(5, 5), (6, 3)] = \sqrt{(x - a)^2 + (y - b)^2} = \sqrt{(5 - 6)^2 + (5 - 3)^2} = 2.45$

Here 2.45 is smallest distance so Data point (5, 5) belongs to cluster C_2

Scanned by CamScanner

As Data point belongs to cluster C_2, we will recalculate the centre of cluster C_2 as following-

Updated Centre of cluster $C_2 = \left(\frac{x+a}{2}, \frac{y+b}{2}\right) = \left(\frac{5+6}{2}, \frac{5+3}{2}\right) = (5.5, 4)$

(6, 3):

$D_1[(6, 3), (2.5, 3.5)] = \sqrt{(x-a)^2 + (y-b)^2} = \sqrt{(6-2.5)^2 + (3-3.5)^2} = 3.54$

$D_2[(6, 3), (5.5, 4)] = \sqrt{(x-a)^2 + (y-b)^2} = \sqrt{(6-5.5)^2 + (3-4)^2} = 1.12$

Here 1.12 is smallest distance so Data point (6, 3) belongs to cluster C_2

As Data point belongs to cluster C_2, we will recalculate the centre of cluster C_2 as following-

Updated Centre of cluster $C_2 = \left(\frac{x+a}{2}, \frac{y+b}{2}\right) = \left(\frac{6+5.5}{2}, \frac{3+4}{2}\right) = (5.75, 3.5)$

(4, 3):

$D_1[(4, 3), (2.5, 3.5)] = \sqrt{(x-a)^2 + (y-b)^2} = \sqrt{(4-2.5)^2 + (3-3.5)^2} = 1.59$

$D_2[(4, 3), (5.75, 3.5)] = \sqrt{(x-a)^2 + (y-b)^2} = \sqrt{(4-5.75)^2 + (3-3.5)^2} = 1.83$

Here 1.59 is smallest distance so Data point (4, 3) belongs to cluster C_1

As Data point belongs to cluster C_1, we will recalculate the centre of cluster C_1 as following-

Updated Centre of cluster $C_1 = \left(\frac{x+a}{2}, \frac{y+b}{2}\right) = \left(\frac{4+2.5}{2}, \frac{3+3.5}{2}\right) = (3.25, 3.25)$

(6, 6):

$D_1[(6, 6), (3.25, 3.25)] = \sqrt{(x-a)^2 + (y-b)^2} = \sqrt{(6-3.25)^2 + (6-3.25)^2} = 3.89$

$D_2[(6, 6), (5.75, 3.5)] = \sqrt{(x-a)^2 + (y-b)^2} = \sqrt{(6-5.75)^2 + (6-3.5)^2} = 2.52$

Here 2.52 is smallest distance so Data point (6, 6) belongs to cluster C_1

The final clusters will be,

$C_1 = \{(2, 4), (3, 3), (4, 3), (6, 6)\}$,

$C_2 = \{(5, 5), (6, 3)\}$

Q9. Apply Agglomerative clustering algorithm on given data and draw dendrogram. Show three clusters with its allocated points. Use single link method.

Adjacency Matrix

	a	b	c	d	e	f
a	0	$\sqrt{2}$	$\sqrt{10}$	$\sqrt{17}$	$\sqrt{5}$	$\sqrt{20}$
b	$\sqrt{2}$	0	8	2	1	$\sqrt{18}$
c	$\sqrt{10}$	$\sqrt{8}$	0	$\sqrt{5}$	$\sqrt{5}$	2
d	$\sqrt{17}$	1	$\sqrt{5}$	0	2	3
e	$\sqrt{5}$	1	$\sqrt{5}$	2	0	$\sqrt{13}$
f	$\sqrt{20}$	$\sqrt{18}$	2	3	$\sqrt{13}$	0

[10M – May16 & Dec17]

Ans:

We have to use single link method to solve this sum. We use following formula for Single link sum,

Min [dist (a), (b)] .... We will choose smallest distance value from two distance values.

Scanned by CamScanner

We have given Adjacency matrix in which we can see that the upper bound part of diagonal identical to lower bound of diagonal so we can use any part of the matrix.

We will use Lower bound of diagonal as show below.

	a	b	c	d	e	f
a	0					
b	$\sqrt{2}$	0				
c	$\sqrt{10}$	$\sqrt{8}$	0			
d	$\sqrt{17}$	1	$\sqrt{5}$	0		
e	$\sqrt{5}$	1	$\sqrt{5}$	2	0	
f	$\sqrt{20}$	$\sqrt{18}$	2	3	$\sqrt{13}$	0

Now we have to find minimum distance value from above distance matrix. We can see that 1 is smallest distance value but it appears twice in matrix so we can choose any one value from it.

Taking distance value of (b, e) = 1

Now we will draw dendrogram for it.

Now we have to recalculate distance matrix.

We will find distance between clustered points I.e. (b, e) and other remaining points.

- **Distance between (b, e) and a:**

  Min[dist (b, e), a]

  = Min[dist (b, a), (e, a)]

  = Min[$\sqrt{2}$, $\sqrt{5}$]

  = $\sqrt{2}$ .... as we have to choose smallest value.

- **Distance between (b, e) and c:**

  Min[dist (b, e), c] = Min[dist (b, c), (e, c)] = Min[$\sqrt{8}$, $\sqrt{5}$] = $\sqrt{5}$

- **Distance between (b, e) and d:**

  Min[dist (b, e), d] = Min[dist (b, d), (e, d)] = Min[1, 2] = 1

- **Distance between (b, e) and f:**

  Min[dist (b, e), f] = Min[dist (b, f), (e, f)] = Min[$\sqrt{18}$, $\sqrt{13}$] = $\sqrt{13}$

Now we have put these values to update distance matrix:

	A	(b, e)	c	d	f
a	0				
(b, e)	$\sqrt{2}$	0			
c	$\sqrt{10}$	$\sqrt{5}$	0		
d	$\sqrt{17}$	1	$\sqrt{5}$	0	
f	$\sqrt{20}$	$\sqrt{13}$	2	3	0

Now we have to find smallest distance value from updated distance matrix again

Here distance between [(b, e), d] = 1 is smallest distance value in distance matrix.

Scanned by CamScanner

Now we have to draw dendrogram for these new clustered points.
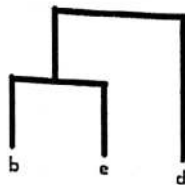
Now recalculating distance matrix again.

Finding distance between clustered points and other remaining points.

* **Distance between [(b, e), d] and a:**

  Min {dist [(b, e), d], a} = Min {dist [(b, e), a], [d, a]} = Min [√2, √17] = √2

* **Distance between [(b, e), d] and c :**

  Min {dist [(b, e), d], c} = Min {dist [(b, e), c], [d, c]} = Min [√5, √5] = √5

* **Distance between [(b, e), d] and f :**

  Min {dist [(b, e), d], f} = Min {dist [(b, e), f], [d, f]} = Min [√13, 3] = 3

Now we have to put these distance values in distance matrix to update it.

	a	(b, e, d)	c	f
A	0			
(b, e, d)	√2	0		
C	√10	√5	0	
f	√20	3	2	0

Here distance between [(b, e, d), a] = √2 is smallest distance value in updated distance matrix.

Drawing dendrogram for new clustered points
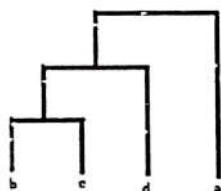
Now recalculating distance matrix.

* **Finding distance between [(b, e, d), a] and c:**

  Min {dist [(b, e, d), a], [c]} = Min {dist [(b, e, d), c], [a, c]} = Min [√5, √10] = √5

* **Finding distance between [(b, e, d), a] and f:**

  Min {dist [(b, e, d), a], [f]} = Min {dist [(b, e, d), f], [a, f]} = Min [3, √20] = 3

Putting these new distance values in distance matrix to update it.

	(b, e, d, a)	c	F
(b, e, d, a)	0		
c	√5	0	
f	3	2	0

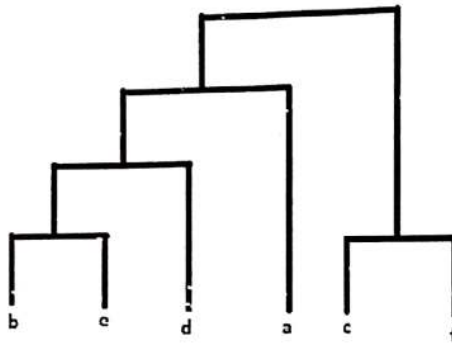Here, distance between (c, f) = 2 is smallest distance.
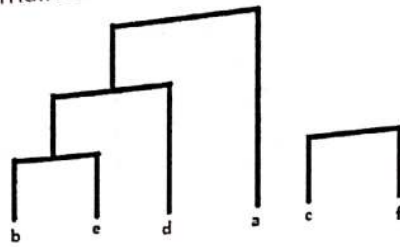
Recalculating distance matrix,

- **Finding distance between (c, f) and (b, e, d, a)**

  Min { dist (c, f), (b, e, d, a)} = Min { dist [c, (b, e, d, a)], [f, (b, e, d, a)]} = Min [$\sqrt{5}$, 3] = $\sqrt{5}$

Putting this value to update distance matrix

	(b, e, d, a)	(c, f)
(b, e, d, a)	0	
(c, f)	$\sqrt{5}$	0

Drawing dendrogram for new cluster,

Q10. For the given set of points identify clusters using complete link and average link using agglomerative clustering.

	A	B
P1	1	1
P2	1.5	1.5
P3	5	5
P4	3	4
P5	4	4
P6	3	3.5

Ans:

[10M – May18]

We have to solve this sum using complete link method and Average link method.

We use following formula for complete link sum,

Max [dist (a), (b)] .... We will choose biggest distance value from two distance values.

We use following formula for complete link sum,

Avg [dist (a), (b)] = $\frac{1}{2}$ [a + b] .... We will choose biggest distance value from two distance values.

Here, given data is not in distance / adjacency matrix form. So we will convert it to distance / adjacency matrix using Euclidean distance formula which is as following,

Distance [(x, y), (a, b)] = $\sqrt{(x-a)^2 + (y-b)^2}$

Now finding distance between P1 and P2, Distance [P1, P2] =

distance[(1, 1), (1.5, 1.5)] = $\sqrt{(x-a)^2 + (y-b)^2}$ = $\sqrt{(1-1.5)^2 + (1-1.5)^2}$ = 0.71

As per above step we will find distance between other points as well.

Distance [P1, P3] = 5.66

Distance [P1, P4] = 3.61

Distance [P1, P5] = 4.25

Distance [P1, P6] = 3.21

Distance [P2, P3] = 4.95

Distance [P2, P4] = 2.92

Distance [P2, P5] = 3.54

Distance [P2, P6] = 2.5

Distance [P3, P4] = 2.24

Distance [P3, P5] = 1.42

Distance [P3, P6] = 2.5

Distance [P4, P5] = 1

Distance [P4, P6] = 0.5

Distance [P5, P6] = 1.12

Putting these values in lower bound of diagonal of distance / adjacency matrix,

	P1	P2	P3	P4	P5	P6
P1	0					
P2	0.71	0				
P3	5.66	4.95	0			
P4	3.61	2.92	2.24	0		
P5	4.25	3.54	1.42	1	0	
P6	3.21	2.5	2.5	0.5	1.12	0

COMPLETE LINK METHOD:

Here, distance between P4 and P6 is 0.5 which is smallest distance in matrix.
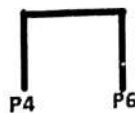
P4   P6

Recalculating distance matrix:

- Distance between (P4, P6) and P1

= Max[dist (P4, P6), P1]

= Max[dist (P4, P1), (P6, P1)]

= Max[3.61, 3.21]

= 3.61 .... We take biggest distance value as it is Complete link method

- Distance between (P4, P6) and P2

  = Max[dist (P4, P6), P2]

  = Max[dist (P4, P2), (P6, P2)]

  = Max[2.92, 2.5]

  = 2.92

- Distance between (P4, P6) and P3

  = Max[dist (P4, P6), P3]

  = Max[dist (P4, P3), (P6, P3)]

  = Max[2.24, 2.5]

  = 2.5

- Distance between (P4, P6) and P5

  = Max[dist (P4, P6), P5]

  = Max[dist (P4, P5), (P6, P5)]

  = Max[1, 1.12]

  = 1.12

Updating distance matrix:

	P1	P2	P3	(P4, P6)	P5
P1	0				
P2	0.71	0			
P3	5.66	4.95	0		
(P4, P6)	3.61	2.92	2.5	0	
P5	4.25	3.54	1.42	1.12	0

Here (P1, P2) = 0.71 is smallest distance in matrix
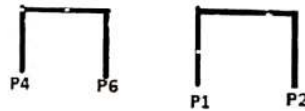
Recalculating distance matrix:

- Distance between (P1, P2) and P3

  = Max[dist (P1, P2), P3]

  = Max[dist (P1, P3), (P2, P3)]

  = Max[5.66, 4.95]

  = 5.66

- Distance between (P1, P2) and (P4, P6)

  = Max[dist (P1, P2), (P4, P6)]

  = Max[dist {P1, (P4, P6)}, {P2, (P4, P6)}]

  = Max[3.61, 2.92]

  = 3.61

Scanned by CamScanner

* Distance between (P1, P2) and P5

= Max[dist (P1, P2), P5]

= Max[dist (P1, P5), (P2, P5)]

= Max[4.25, 3.54]

= 4.25

Updating distance matrix using above values,

	(P1, P2)	P3	(P4, P6)	P5
(P1, P2)	0			
P3	5.66	0		
(P4, P6)	3.61	2.5	0	
P5	4.25	1.42	1.12	0

Here [(P4, P6), P5] = 1.12 is smallest distance in matrix,
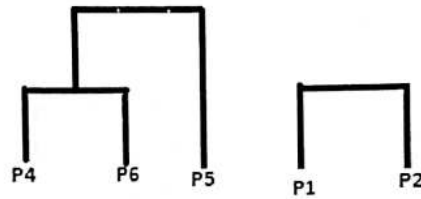
Recalculating distance matrix

* Distance between {(P4, P6), P5} and (P1, P2)

= Max[dist {(P4, P6), P5}, (P1, P2)]

= Max[dist {(P4, P6), (P1, P2)}, {P5, (P1, P2)}]

= Max[3.61, 4.25]

= 4.25

* Distance between {(P4, P6), P5} and P3

= Max[dist {(P4, P6), P5}, P3]

= Max[dist {(P4, P6), P3}, {P5, P3}]

= Max[2.5, 1.42]

= 2.5

Updating distance matrix

	(P1, P2)	P3	(P4, P6, P5)
(P1, P2)	0		
P3	5.66	0	
(P4, P6, P5)	4.25	2.5	0

Here, [(P4, P6, P5), P3] = 2.5 is smallest distance in matrix,
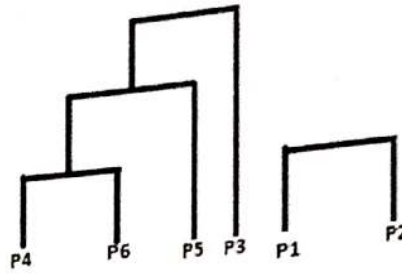
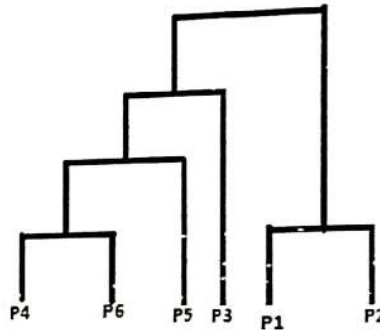## Recalculating distance matrix:

- Distance between {(P4, P6, P5), P3} and {P1, P2}

    = Max[dist {(P4, P6, P5), P3}, {P1, P2}]

    = Max[dist{(P4, P6, P5), (P1, P2)}, {P3, (P1, P2)}]

    = Max[4.25, 5.66]

    = 5.66

Updating distance matrix

	(P1, P2)	(P4, P6, P5, P3)
(P1, P2)	0	
(P4, P6, P5, P3)	5.66	0

AVERAGE LINK METHOD:

Original Distance Matrix:

	P1	P2	P3	P4	P5	P6
P1	0					
P2	0.71	0				
P3	5.66	4.95	0			
P4	3.61	2.92	2.24	0		
P5	4.25	3.54	1.42	1	0	
P6	3.21	2.5	2.5	0.5	1.12	0

Here, distance between P4 and P6 is 0.5 which is smallest distance in matrix.

Scanned by CamScanner
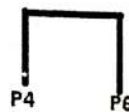
Recalculating distance matrix,

* Distance between (P4, P6) and P1

  = Avg[dist (P4, P6), P1]

  = Avg[dist (P4, P1), (P6, P1)]

  = $\frac{1}{2}$[3.61 + 3.21]

  = 3.41 .... We take average distance value as it is Average link method

* Distance between (P4, P6) and P2

  = Avg[dist (P4, P6), P2]

  = Avg[dist (P4, P2), (P6, P2)]

  = $\frac{1}{2}$ [2.92 + 2.5]

  = 2.71

* Distance between (P4, P6) and P3

  = Avg[dist (P4, P6), P3]

  = Avg[dist (P4, P3), (P6, P3)]

  = $\frac{1}{2}$ [2.24 + 2.5]

  = 2.37

* Distance between (P4, P6) and P5

  = Avg[dist (P4, P6), P5]

  = Avg[dist (P4, P5), (P6, P5)]

  = $\frac{1}{2}$ [1 + 1.12]

  = 1.06

Updating distance matrix:

	P1	P2	P3	(P4, P6)	P5
P1	0				
P2	0.71	0			
P3	5.66	4.95	0		
(P4, P6)	3.41	2.71	2.37	0	
P5	4.25	3.54	1.42	1.06	0

Here (P1, P2) = 0.71 is smallest distance in matrix

Recalculating distance matrix:

* Distance between (P1, P2) and P3

  = Avg[dist (P1, P2), P3]

  = Avg[dist (P1, P3), (P2, P3)]

  = $\frac{1}{2}$[5.66 + 4.95]

  = 5.31

Handcrafted by BackkBenchers Publications

Scanned by CamScanner

- Distance between (P1, P2) and (P4, P6)

    = Avg[dist (P1, P2), (P4, P6)]

    = Avg[dist {P1, (P4, P6)}, {P2, (P4, P6)}]

    = $\frac{1}{2}$ [3.41 + 2.71]

    = 3.06

- Distance between (P1, P2) and P5

    = Avg[dist (P1, P2), P5]

    = Avg[dist (P1, P5), (P2, P5)]

    = $\frac{1}{2}$ [4.25 + 3.54]

    = 3.90

Updating distance matrix using above values,

	(P1, P2)	P3	(P4, P6)	P5
(P1, P2)	0			
P3	5.31	0		
(P4, P6)	3.06	2.5	0	
P5	3.90	1.42	1.12	0

Here [(P4, P6), P5] = 1.12 is smallest distance in matrix,
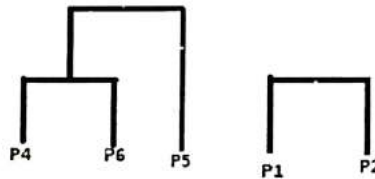
Recalculating distance matrix:

- Distance between {(P4, P6), P5} and (P1, P2)

    = Avg[dist {(P4, P6), P5}, (P1, P2)]

    = Avg[dist {(P4, P6), (P1, P2)}, {P5, (P1, P2)}]

    = $\frac{1}{2}$ [3.06 + 3.90]

    = 3.48

- Distance between {(P4, P6), P5} and P3

    = Avg[dist {(P4, P6), P5}, P3]

    = Avg[dist {(P4, P6), P3}, {P5, P3}]

    = $\frac{1}{2}$ [2.5 + 1.42]

    = 1.96

Updating distance matrix

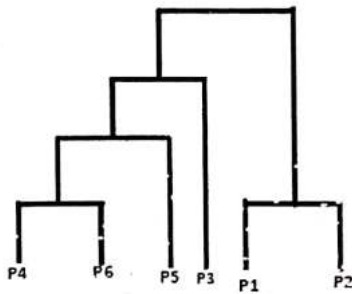|	(P1, P2)	P3	(P4, P6, P5)
(P1, P2)	0		
P3	5.66	0	
(P4, P6, P5)	3.48	1.96	0

Here, [(P4, P6, P5), P3] = 1.96 is smallest distance in matrix,
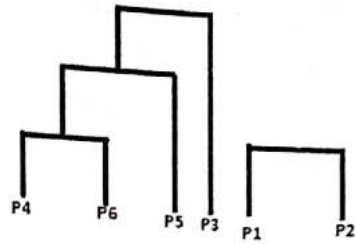
Recalculating distance matrix:

• Distance between {(P4, P6, P5), P3} and {P1, P2}

= Avg[dist {(P4, P6, P5), P3}, {P1, P2}]

= Avg[dist{(P4, P6, P5), (P1, P2)}, {P3, (P1, P2)}]

= $\frac{1}{2}$ [3.48 + 5.66]

= 4.57

Updating distance matrix

	(P1, P2)	(P4, P6, P5, P3)
(P1, P2)	0	
(P4, P6, P5, P3)	4.57	0

CHAP - 8: REINFORCEMENT LEARNING

Q1. What are the elements of reinforcement learning?

Q2. What is Reinforcement Learning? Explain with the help of an example.

Q3. Explain reinforcement learning in detail along with the various elements involved in forming the concept. Also define what is meant by partially observable state.

[5 - 10M | May16, Dec16, May17, Dec17 & May18]

Ans:

REINFORCEMENT LEARNING:

1. Reinforcement Learning is a type of machine learning algorithm.

2. It enables an agent to learn in an interactive environment by trial and error.

3. Agent uses feedback from its own actions and experiences.

4. The goal is to find a suitable action model that increase total reward of the agent.

5. Output depends on the state of the current input.

6. The next input depends on the output of the previous input.

7. Types of Reinforcement Learning:

   a. Positive RL.

   b. Negative RL.

8. The most common application of reinforcement learning are:

   a. **PC Games:** Reinforcement learning is widely being used in PC games like Assassin's Creed, Chess, etc. the enemies change their moves and approach based on your performance.

   b. **Robotics:** Most of the robots that you see in the present world are running on Reinforcement Learning.

EXAMPLE:

1. We have an agent and a reward, with many hurdles in between.

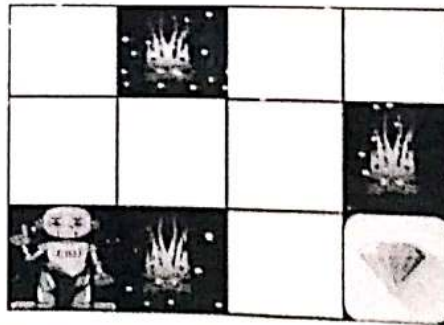2. The agent is supposed to find the best possible path to reach the reward.

Figure 8.1: Example of Reinforcement Learning.

3. The figure 8.1 s hows robot, diamond and fire.

4. The goal of the robot is to get the reward

5. Reward is the diamond and avoid the hurdles that is fire.

6. The robot learns by trying all the possible paths

7. After learning robot chooses a path which gives him the reward with the least hurdles.

8. Each right step will give the robot a reward.

9. Each wrong step will subtract the reward of the robot.

10. The total reward will be calculated when it reaches the final reward that is the diamond.

REINFORCEMENT LEARNING ELEMENTS:

There are four main sub elements of a reinforcement learning system:

1. A policy
2. A reward function
3. A value function
4. A model of the environment (optional)
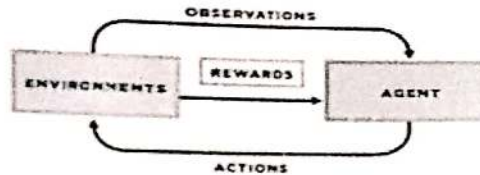
I) Policy:

1. The policy is the core of a reinforcement learning agent.
2. Policy is sufficient to determine behaviour.
3. In general, policies may be stochastic.
4. A policy defines the learning agent's way of behaving at a given time.
5. A policy is a mapping from perceived states of the environment to actions to be taken when in those states.
6. In some cases the policy may be a simple function or lookup table

II) Reward Function:

1. A reward function defines the goal in a reinforcement learning problem.
2. Reward function maps each perceived state of the environment to a single number, a reward.
3. The reward function defines what the good and bad events are for the agent.
4. The reward function must necessarily be unalterable by the agent.
5. Reward functions serve as a basis for altering the policy.

III) Value Function:

1. Whereas a reward function indicates what is good in an immediate sense,
2. A value function specifies what is good in the long run.
3. The value of a state is the total amount of reward an agent can expect to collect over the future, starting from that state.
4. Rewards are in a sense primary, whereas values, as predictions of rewards, are secondary.
5. Without rewards there could be no values, and the only purpose of estimating values is to achieve more reward.
6. It is values with which we are most concerned when making and evaluating decisions. Action choices are made based on value judgments.

IV) Model:

1. A model is an optional element of Reinforcement learning system.
2. Model is something that mimics the behaviour of the environment.
3. For example, given a state and action, the model might predict the resultant next state and next reward.
4. Models are used for planning.
5. If model know current state and action then it can predict the resultant next state and next reward.

Q4.   Model Based Learning

Ans: [10M | Dec17]

MODEL BASED LEARNING:

1.   Model-based machine learning refers to machine learning models.

2.   Those models are parameterized with a certain number of parameters which do not change as the size of training data changes.

3.   The goal is "to provide a single development framework which supports the creation of a wide range of bespoke models".

4.   For example, if you assume that a set of data {X1, ..., n, Y1,..., n} you are given is subject to a linear model Yi=sign($w^T X_i$ + b), Where w ∈ R^m and m is the dimension of each data point regardless of n.

5.   There are 3 steps to model based machine learning, namely:

    a.   **Describe the Model:** Describe the process that generated the data using factor graphs.

    b.   **Condition on Observed Data:** Condition the observed variables to their known quantities.

    c.   **Perform Inference:** Perform backward reasoning to update the prior distribution over the latent variables or parameters.

6.   This framework emerged from an important convergence of three key ideas:

    a.   The adoption of a Bayesian viewpoint,

    b.   The use of factor graphs (a type of a probabilistic graphical model), and

    c.   The application of fast, deterministic, efficient and approximate inference algorithms.

MODEL BASED LEARNING PROCESS:

1.   We start with model based learning where we completely know the environment model parameters.

2.   Environment parameters are P(rt + 1) and P(st + 1 | st, at).

3.   In such a case, we do not need any exploration.

4.   We can directly solve for the optional value function and policy using dynamic programming

5.   The optimal value function is unique and is the solution to simultaneous equations

6.   Once we have the optimal value function, the optimal policy is to choose the action that maximize value in next state as following

Π * (st) = arg$_{at}$ max(E [rt + 1 | st, at] + r ∑ P[st + 1| st, at] v * [st + 1])

BENEFITS:

1.   Provides a systematic process of creating ML solutions.

2.   Allow for incorporation of prior knowledge.

3.   Does not suffers from over fitting.

4.   Separates model from inference/training code.

5. Explain in detail Temporal Difference Learning

Ans: [10M | Dec16, May17 & Dec18]

TEMPORAL DIFFERENCE LEARNING:

- It is an approach to learning how to predict a quantity that depends on future values of a given signal.
- Temporal Difference Learning methods can be used to estimate these value functions.
- Learning happens through the iterative correction of your estimated returns towards a more accurate target return.
- TD algorithms are often used to predict a measure of the total amount of reward expected over the future.
- They can be used to predict other quantities as well.
- Different TD algorithms:
  a. TD(0) algorithm
  b. TD(1) algorithm
  c. TD(λ) algorithm
- The easiest to understand temporal-difference algorithm is the TD(0) algorithm.

TEMPORAL DIFFERENCE LEARNING METHODS:

I) On-Policy Temporal Difference methods:

1. Learns the value of the policy that is used to make decisions.
2. The value functions are updated using results from executing actions determined by some policy.
3. These policies are usually "soft" and non-deterministic.
4. It ensures there is always an element of exploration to the policy.
5. The policy is not so strict that it always chooses the action that gives the most reward.
6. On-policy algorithms cannot separate exploration from control.

II) Off-Policy Temporal Difference methods:

1. It can learn different policies for behaviour and estimation.
2. Again, the behaviour policy is usually "soft" so there is sufficient exploration going on.
3. Off-policy algorithms can update the estimated value functions using actions which have not actually been tried.
4. Off-policy algorithms can separate exploration from control.
5. Agent may end up learning tactics that it did not necessarily shows during the learning phase.

ACTION SELECTION POLICIES:

The aim of these policies is to balance the trade-off between exploitation and exploration.

I) ϵ - Greedy:

1. Most of the time the action with the highest estimated reward is chosen, called the greediest action.
2. Every once in a while, say with a small probability, an action is selected at random.
3. The action is selected uniformly, independent of the action-value estimates.
4. This method ensures that if enough trials are done, each action will be tried an infinite number of times.
5. It ensures optimal actions are discovered.

Scanned by CamScanner

ii) ε - Soft:

1. Very similar to ε - greedy.
2. The best action is selected with probability $1 - ε$.
3. Rest of the time a random action is chosen uniformly.

iii) Softmax:

1. One drawback of ε -greedy and ε -soft is that they select random actions uniformly.
2. The worst possible action is just as likely to be selected as the second best.
3. Softmax remedies this by assigning a rank or weight to each of the actions, according to their action value estimate.
4. A random action is selected with regards to the weight associated with each action.
5. It means that the worst actions are unlikely to be chosen.
6. This is a good approach to take where the worst actions are very unfavourable.

ADVANTAGES OF TD METHODS:

1. Don't need a model of the environment.
2. On-line and incremental so can be fast.
3. They don't need to wait till the end of the episode.
4. Need less memory and computation.

Q6. What is Q-learning? Explain algorithm for learning Q

Ans:

[10M | Dec18]

Q-LEARNING:

1. Q-learning is a reinforcement learning technique used in machine learning.
2. Q-learning is a values-based learning algorithm.
3. The goal of Q-learning is to learn a policy.
4. Q learning tells an agent what action to take under what circumstances.
5. Q-learning uses temporal differences to estimate the value of $Q^*(s, a)$.
6. In Q-learning, the agent maintains a table of $Q[S, A]$.

   Where, S is the set of states and

   A is the set of actions.

   $Q[s, a]$ represents its current estimate of $Q^*(s, a)$.

7. An experience (s, a, r, s') provides one data point for the value of $Q(s, a)$.
8. The data point is that the agent received the future value of $r + \gamma V(s')$,

   Where $V(s') = \max_{a'} Q(s', a')$

9. This is the actual current reward plus the discounted estimated future value.
10. This new data point is called a return.
11. The agent can use the temporal difference equation to update its estimate for $Q(s, a)$:

    $$Q[s, a] \leftarrow Q[s, a] + \alpha(r + \gamma \max_{a'} Q[s', a'] - Q[s, a])$$

12. Or, equivalently,

    $$Q[s, a] \leftarrow (1-\alpha) Q[s, a] + \alpha(r + \gamma \max_{a'} Q[s', a']).$$

---

13. It can be proven that given sufficient training under any ε - soft policy, the algorithm converges with probability 1 to a close approximation of the action-value function for an arbitrary target policy.

14. Q-Learning learns the optimal policy even when actions are selected according to a more exploratory or even random policy.

Q-TABLE:

1. Q-Table is just a simple lookup table

2. In Q-Table we calculate the maximum expected future rewards for action at each state.

3. Basically, this table will guide us to the best action at each state.

4. In the Q-Table, the columns are the actions and the rows are the states.

5. Each Q-table score will be the maximum expected future reward that the agent will get if it takes that action at that state.

6. This is an iterative process, as we need to improve the Q-Table at each iteration.

PARAMETERS USED IN THE Q-VALUE UPDATE PROCESS:

I) α - the learning rate:

1. Set between 0 and 1.

2. Setting it to 0 means that the Q-values are never updated, hence nothing is learned.

3. Setting a high value such as 0.9 means that learning can occur quickly.

II) γ - Discount factor:

1. Set between 0 and 1.

2. This models the fact that future rewards are worth less than immediate rewards.

3. Mathematically, the discount factor needs to be set less than 0 for the algorithm to converge.

III) Max$_a$ - the maximum reward:

1. This is attainable in the state following the current one.

2. i.e. the reward for taking the optimal action thereafter.

PROCEDURAL APPROACH:

1. Initialize the Q-values table, Q(s, a).

2. Observe the current state, s.

3. Choose an action, a, for that state based on one of the action selection policies (ε -soft, ε -greedy or Softmax).

4. Take the action, and observe the reward, r, as well as the new state, s'.

5. Update the Q-value for the state using the observed reward and the maximum reward possible for the next state. (The updating is done according to the formula and parameters described above.)

6. Set the state to the new state, and repeat the process until a terminal state is reached.

Q7. Explain following terms with respect to Reinforcement learning: delayed rewards, exploration, and partially observable states.(10 mark – d18)

Q8. Explain reinforcement learning in detail along with the various elements involved in forming the concept. Also define what is meant by partially observable state. (10 mark – d17)

[10M | Dec17 & Dec18]

Ans:

EXPLORATION:

1. It means gathering more information about the problem.

2. Reinforcement learning requires clever exploration mechanisms

3. Randomly selecting actions, without reference to an estimated probability distribution, shows poor performance.

4. The case of (small) finite Markov decision processes is relatively well understood.

5. However, due to the lack of algorithms that properly scale well with the number of states, simple exploration methods are the most practical.

6. One such method is ε-greedy, when the agent chooses the action that it believes has the best long-term effect with probability 1- ε.

7. If no action which satisfies this condition is found, the agent chooses an action uniformly at random.

8. Here, 0 < ε < 1 is a tuning parameter, which is sometimes changed, either according to a fixed schedule, or adaptively based on heuristics.

DELAYED REWARDS:

1. In the general case of the reinforcement learning problem, the agent's actions determine not only its immediate reward.

2. And it also determine (at least probabilistically) the next state of the environment.

3. It may take a long sequence of actions, receiving insignificant reinforcement,

4. Then finally arrive at a state with high reinforcement.

5. It has to learn from delayed reinforcement. This is called delayed rewards.

6. The agent must be able to learn which of its actions are desirable based on reward that can take place arbitrarily far in the future.

7. It can also be done with eligibility traces, which weight the previous action a lot.

8. The action before that a little less, and the action before that even less and so on. But it takes lot of computational time.

PARTIALLY OBSERVABLE STATES:

1. In certain applications, the agent does not know the state exactly.

2. It is equipped with sensors that return an observation using which the agent should estimates the state.

3. For example, we have a robot which navigates in a room

4. The robot may not know its exact location in the room, or what else is in room.

5. The robot may have a camera with which sensory observations are recorded.

6. This does not tell the robot its state exactly but gives indication as to its likely state.

7. For example, the robot may only know that there is a wall to its right.

8. The setting is like a Markov decision process, except that after taking an action at

9. The new state s_{t+1} is not known but we have an observation o_{t+1} which is stochastic function of s_t and $p(o_{t+1} \mid s_t, a_t)$.

10. This is called as partially observable state.

Scanned by CamScanner

"Education is Free.... But its Technology used & Efforts utilized which we charge"

It takes lot of efforts for searching out each & every question and transforming it into Short & Simple Language. Entire Topper's Solutions Team is working out for betterment of students, do help us.

"Say No to Photocopy...."

With Regards,

Topper's Solutions Team.