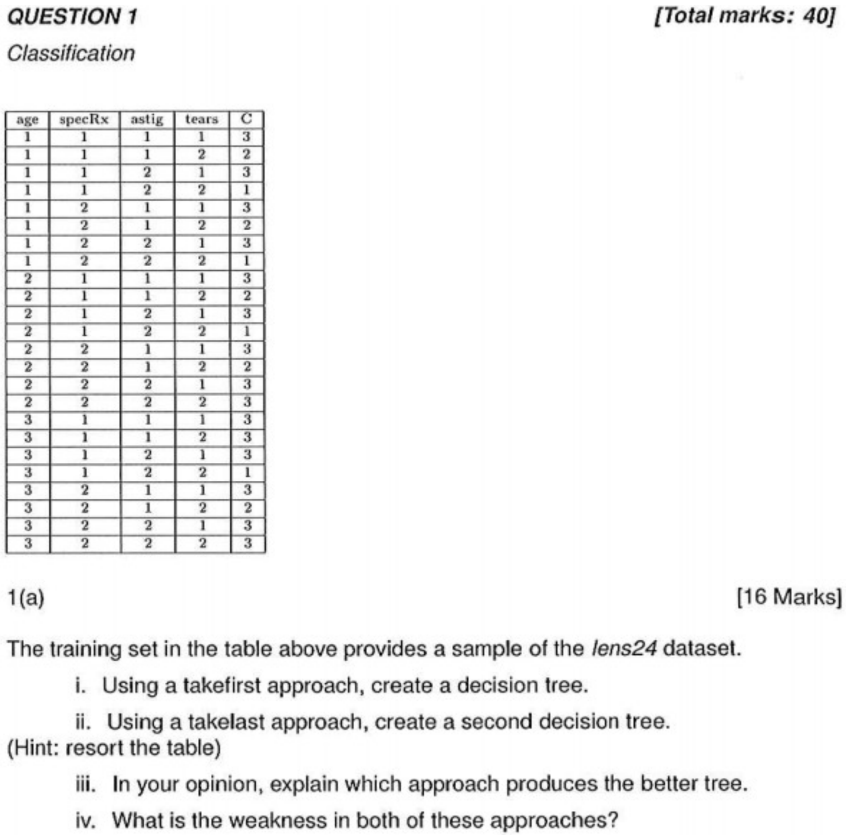
Data Mining 2015



QUESTION 1 ***[Total marks: 40]***

Classification

age	specRx	astig	tears	C
1	1	1	1	3
1	1	1	2	2
1	1	2	1	3
1	1	2	2	1
1	2	1	1	3
1	2	1	2	2
1	2	2	1	3
1	2	2	2	1
2	1	1	1	3
2	1	1	2	2
2	1	2	1	3
2	1	2	2	1
2	2	1	1	3
2	2	1	2	2
2	2	2	1	3
2	2	2	2	3
3	1	1	1	3
3	1	1	2	3
3	1	2	1	3
3	1	2	2	1
3	2	1	1	3
3	2	1	2	2
3	2	2	1	3
3	2	2	2	3

1(a) [16 Marks]

The training set in the table above provides a sample of the *lens24* dataset.

i. Using a takefirst approach, create a decision tree.

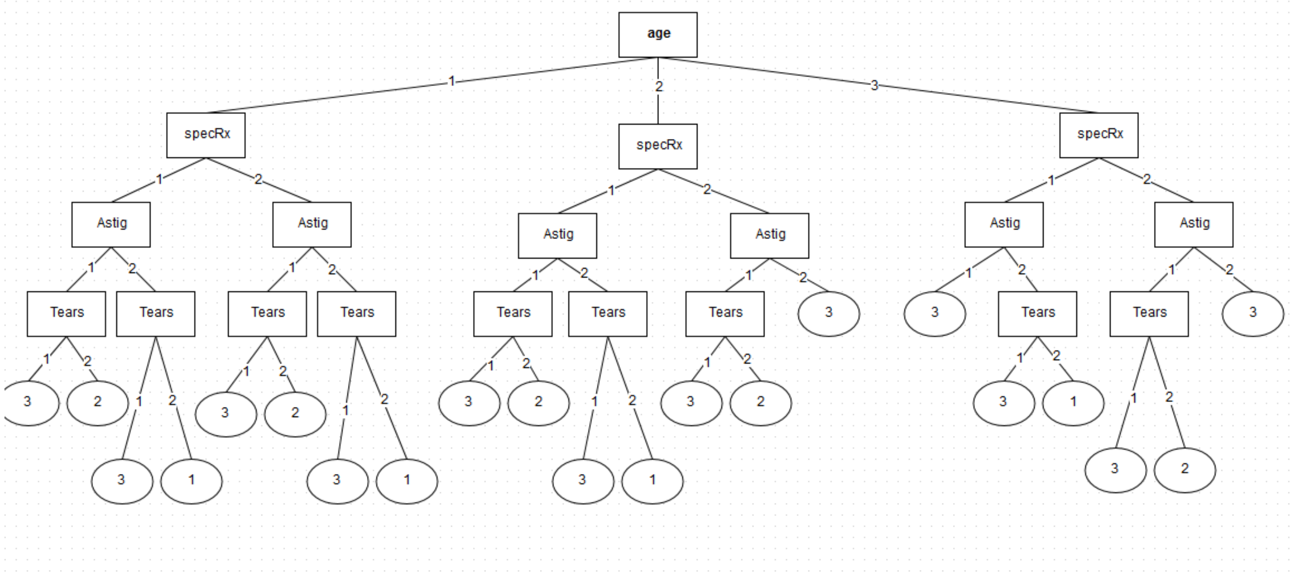
ii. Using a takelast approach, create a second decision tree.

(Hint: resort the table)

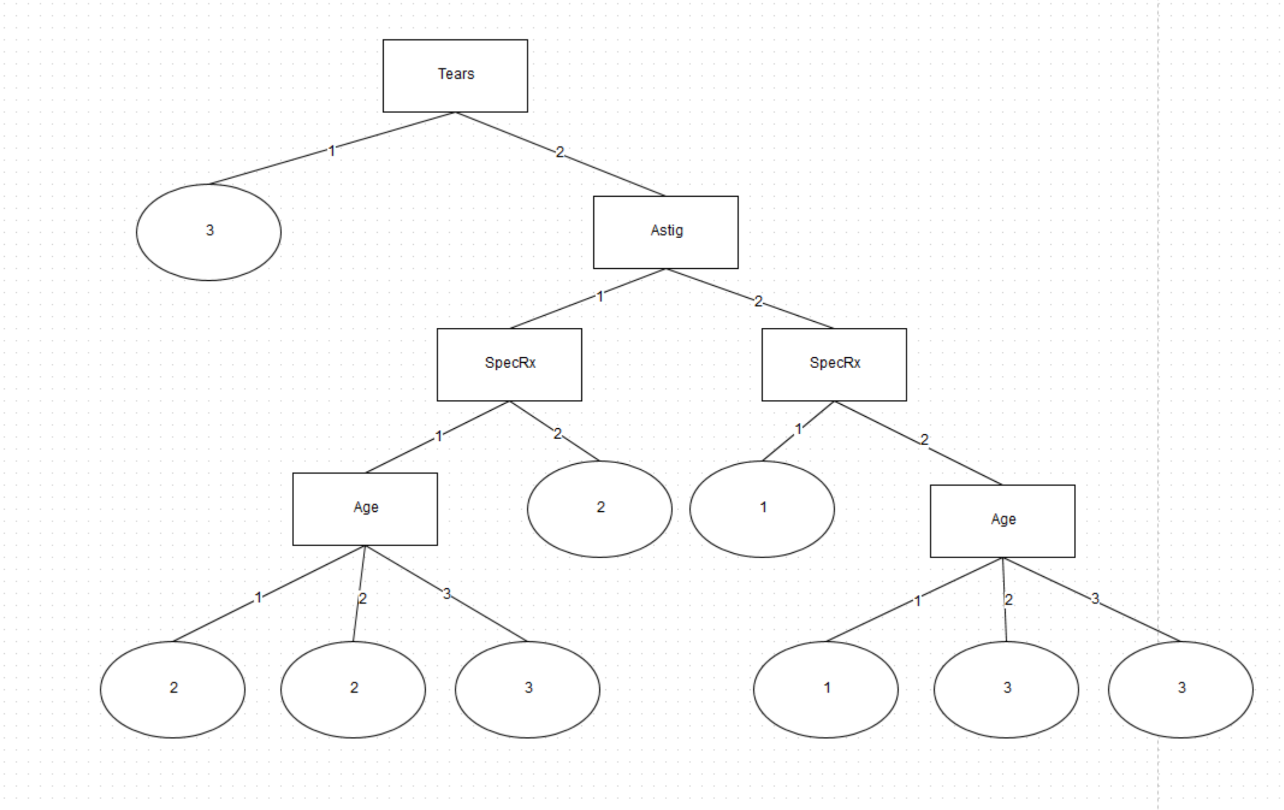
iii. In your opinion, explain which approach produces the better tree.

iv. What is the weakness in both of these approaches?

a)
I)



ii)

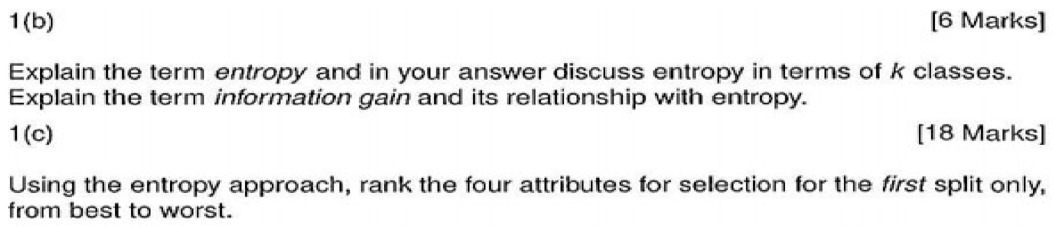


iii)

Neither, The reason being with out seeing the actual values of data and testing both on unseen instances I cannot decide if one approach is better then the other. One my first impressions I would say that the TakeLast approach produced a better tree as it resulted in less splits. However you can contradict this argument by looking at the first node tears that says if tears is 1 just classify it under classification value 3, if 2 split. Now lets speculate on what the values 1 and 2 might mean in this database. The Database in the question is Lens24 so we can say it may have something to do with glasses as tears is a attribute lets say it is for contact lenses more specifically. Now lets deduce what tears might mean, It could mean “does the users eye tear up on putting in contacts”? value 1 being yes, value 2 being no . Now with this information the TakeLast three makes no sense as the result of just giving someone a certain lens because they tear up does not work as it does not take into consideration their actual sight(Which the other attributes could be describing). So now we can say the TakeFirst approach is better, but it still might fail we cannot be certain. So the only way to decide is to test both on unseen instances and compare the results.

iv)

The weakness in both of these approaches is the method of choosing attributes with out knowing any information on how the selected attribute would contribute to the splitting process. This is the reason we use Information gain, gain ratio or the gini index to help with the selection of attributes.



1(b) [6 Marks]

Explain the term *entropy* and in your answer discuss entropy in terms of *k* classes. Explain the term *information gain* and its relationship with entropy.

1(c) [18 Marks]

Using the entropy approach, rank the four attributes for selection for the *first* split only, from best to worst.

Entropy is not covered this year instead its information gain, gain ratio or the gini index!

Information gain is defined as the difference between the original information requirement and the new requirement. It tells us how much would be gained by branching on a attribute.
Biased toward tests with many outcomes, prefers to select attributes having a large number of values. What if you have a attribute that is a unique identifier, result in a large number of partitions

Steps:
First calculate the expected information needed to classify a tuple in D:

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

$$p_i = |C_{i,D}| / |D|$$

p_i is the probability of that a tuple D belongs to class C_j

for each attribute A:
{

Calculate the expected information required to classify a tuple from D based on the partitioning by A

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

Calculate the information gain for the attribute A:

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

}
Then choose the attribute with the highest information gain to split on.

Gain ratio attempts to over come the bias with information gain on attributes having a large number of values. It does this by applying a form of normalization to info gain using a split information value:

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

SplitInfo considers the number of tuples having the outcome with respect to the total number of tuples in D.

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo(A)}$$

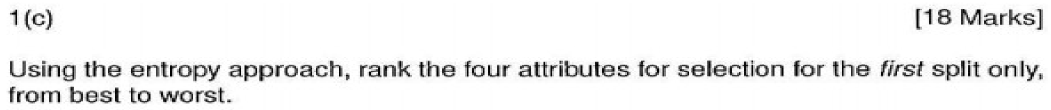
Gini index measures the impurity of a data partition or a set of training tuples. Gini index considers the binary split of each attribute. To determine the best binary split examine all subsets of A excluding the power and empty sets.

Steps for selection using Gini index:

First calculate the impurity of D:

$$Gini(D)=1-\sum_{i=1}^{m} p_i^2$$

```
For each attribute A
{
        for all possible binary splits of A, A partions D into $D_1$ and $D_2$
        {
                Calculate the gini index of that split:
                        $Gini_A(D)=\frac{|D_1|}{|D|}Gini(D_1)+\frac{|D_2|}{|D|}Gini(D_2)$
        }
        Choose the binary split with the lowest gini index.
        Calculate the change in impurity for the lowest gini index:
                        (The symbol for “change in” is delta(triangle))
                        $\Delta Gini(A)=Gini(D)-Gini_A(D)$
}
Select the attribute that maximizes the reduction in impurity and split on.
```



1(c) [18 Marks]

Using the entropy approach, rank the four attributes for selection for the *first* split only, from best to worst.

Again we entropy wasn't covered will be one of the 3 methods above instead!

Information Gain:

$$Info(D)=-\frac{4}{24}\log_2(\frac{4}{24})-\frac{5}{24}\log_2(\frac{5}{24})-\frac{15}{24}\log_2(\frac{15}{24})=1.32\,bits$$

$$Info_{age}(D)=\frac{8}{24}\times(-\frac{2}{8}\log_2(\frac{2}{8})-\frac{2}{8}\log_2(\frac{2}{8})-\frac{4}{8}\log_2(\frac{4}{8}))=0.5$$

$$+\frac{8}{24}\times(-\frac{1}{8}\log_2(\frac{1}{8})-\frac{2}{8}\log_2(\frac{2}{8})-\frac{5}{8}\log_2(\frac{5}{8}))=0.43$$

$$+\frac{8}{24}\times(-\frac{1}{8}\log_2(\frac{1}{8})-\frac{1}{8}\log_2(\frac{1}{8})-\frac{6}{8}\log_2(\frac{6}{8}))=0.35$$

$$Info_{age}(D)=1.28\,bits$$

$$Gain(age)=1.32-1.28=0.04\,bits$$

$$Info_{specRx}(D)=\frac{12}{24}\times(-\frac{3}{12}\log_2(\frac{3}{12})-\frac{2}{12}\log_2(\frac{2}{12})-\frac{7}{12}\log_2(\frac{7}{12}))=0.69$$
$$+\frac{12}{24}\times(-\frac{1}{12}\log_2(\frac{1}{12})-\frac{3}{12}\log_2(\frac{3}{12})-\frac{8}{12}\log_2(\frac{8}{12}))=0.59$$
$$Info_{age}(D)=1.28\,bits$$

$$Gain(specRx)=1.32-1.28=0.04\,bits$$

$$Info_{asig}(D)=\frac{12}{24}\times(-\frac{5}{12}\log_2(\frac{5}{12})-\frac{7}{12}\log_2(\frac{7}{12}))=0.48$$
$$+\frac{12}{24}\times(-\frac{4}{12}\log_2(\frac{4}{12})-\frac{8}{12}\log_2(\frac{8}{12}))=0.45$$
$$Info_{age}(D)=0.93\,bits$$

$$Gain(asig)=1.32-0.93=0.39\,bits$$

$$Info_{tears}(D)=\frac{12}{24}\times(-1\log_2(1))=0$$
$$+\frac{12}{24}\times(-\frac{4}{12}\log_2(\frac{4}{12})-\frac{5}{12}\log_2(\frac{5}{12})-\frac{3}{12}\log_2(\frac{3}{12}))=0.77$$
$$Info_{tears}(D)=0.77\,bits$$

$$Gain(tears)=1.32-0.77=0.55\,bits$$

attribute ranks: tears > asig > specRx = age

Gain Ratio

$$SplitInfo_{age}(D)=\frac{8}{24}\times\log_2(\frac{8}{24})$$
$$+\frac{8}{24}\times\log_2(\frac{8}{24})$$
$$+\frac{8}{24}\times\log_2(\frac{8}{24})=1.58$$

$$SplitInfo_{specRx}(D)=\frac{12}{24}\times\log_2(\frac{12}{24})$$
$$+\frac{12}{24}\times\log_2(\frac{12}{24})=1.5$$

$$GainRatio(age)=\frac{0.04}{1.58}=0.025$$

$$GainRatio(specRx)=\frac{0.04}{1.5}=0.0026$$

$$SplitInfo_{astig}(D)=\frac{12}{24}\times\log_2(\frac{12}{24})$$
$$+\frac{12}{24}\times\log_2(\frac{12}{24})=1.5$$

$$SplitInfo_{tears}(D)=\frac{12}{24}\times\log_2(\frac{12}{24})$$
$$+\frac{12}{24}\times\log_2(\frac{12}{24})=1.5$$

$$GainRatio(astig)=\frac{0.39}{1.5}=0.26$$

$$GainRatio(tears)=\frac{0.55}{1.5}=0.36$$

Attribute ranks: tears > astig > age > specRx

Gini index

$$Gini(D)=1-(\frac{4}{24})^2-(\frac{5}{24})^2-(\frac{15}{24})^2=0.538$$

Subsets for age:
{1,2} ,{2,3} ,{1,3} ,{1} ,{2} ,{3}

$$Gini_{age}\in\{1,2\}^{(D)}$$

$$=\frac{16}{24}Gini(\{1,2\})+\frac{8}{24}Gini(\{3\})$$
$$=\frac{16}{24}(1-(\frac{3}{16})^2-(\frac{4}{16})^2-(\frac{9}{16})^2)+\frac{8}{16}(1-(\frac{1}{8})^2-(\frac{1}{8})^2-(\frac{6}{8})^2)$$
$$=\frac{16}{24}0.585+\frac{8}{16}0.406$$
$$=1.296$$
$$=Gini_{age}\in\{3\}^{(D)}$$

$$Gini_{age}\in\{2,3\}^{(D)}$$

$$=\frac{16}{24}Gini(\{2,3\})+\frac{8}{24}Gini(\{1\})$$
$$=\frac{16}{24}(1-(\frac{2}{16})^2-(\frac{3}{16})^2-(\frac{11}{16})^2)+\frac{8}{16}(1-(\frac{2}{8})^2-(\frac{2}{8})^2-(\frac{4}{8})^2)$$
$$=\frac{16}{24}0.476+\frac{8}{16}0.625$$
$$=1.442$$
$$=Gini_{age}\in\{1\}^{(D)}$$

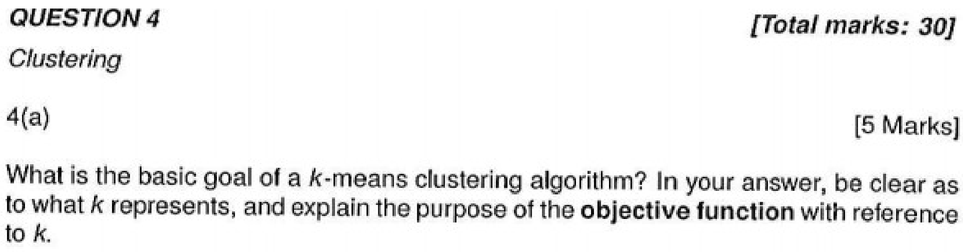
$$Gini_{age}\in\{1,3\}^{(D)}$$

$$=\frac{16}{24}Gini(\{1,3\})+\frac{8}{24}Gini(\{2\})$$
$$=\frac{16}{24}(1-(\frac{3}{16})^2-(\frac{3}{16})^2-(\frac{10}{16})^2)+\frac{8}{16}(1-(\frac{1}{8})^2-(\frac{2}{8})^2-(\frac{5}{8})^2)$$
$$=\frac{16}{24}0.539+\frac{8}{16}0.531$$
$$=1.385$$
$$=Gini_{age}\in\{2\}^{(D)}$$

Best binary split is {1,2} as it has the lowest Gini index.

$$\Delta Gini(age)=0.538-1.296=-0.758$$

Repeat the process for all attributes and then select the highest delta gini index!



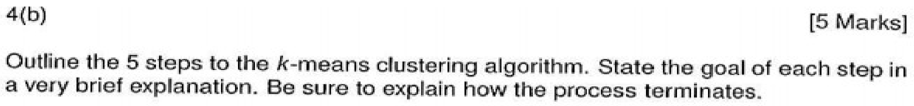
QUESTION 4 **[Total marks: 30]**

Clustering

4(a) [5 Marks]

What is the basic goal of a *k*-means clustering algorithm? In your answer, be clear as to what *k* represents, and explain the purpose of the **objective function** with reference to *k*.

a)
The goal of k-means clustering is to group points of data into clusters where each object is assigned to precisely one set of clusters. The value k represents the number of cluster we like to form from the data. An objective function is used to measure the quality of a set of clusters. We use the sum of the squares of the distances of each point from centroid of the cluster that it is assigned to. In k-means we want the value of this function to be as small as possible.



4(b) [5 Marks]

Outline the 5 steps to the *k*-means clustering algorithm. State the goal of each step in a very brief explanation. Be sure to explain how the process terminates.

b)

Step 1: Select a value for K

- This step is to determine how many clusters we want to form.

Step 2:Select K points to act as out initial centroids

- we select K points so that we can begin to assign our points to their cluster.

Step 3:Assign each of the points to the cluster of its nearest centroids

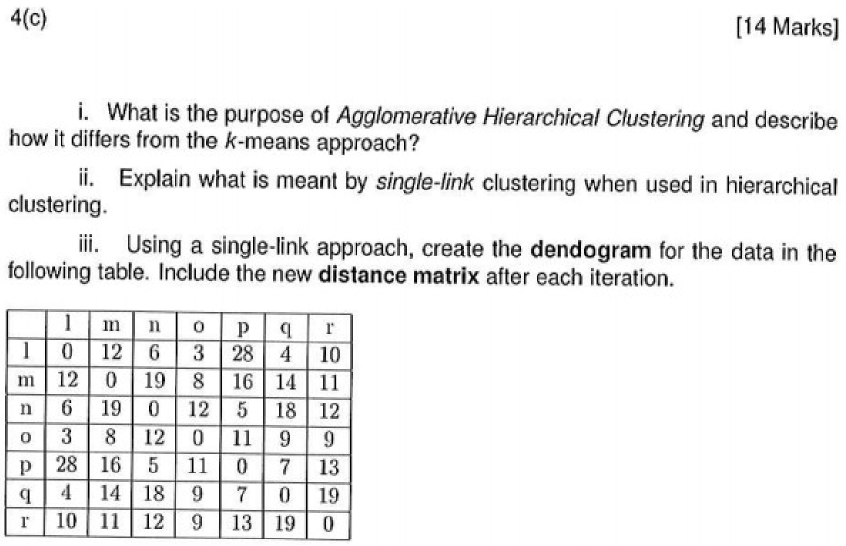
- The goal of this step is to create our k clusters

Step 4:recalculate the centroids

- as our centroids are no longer the true centroid value we need to recalculate the centroids
-

Step 5:repeat steps 3 and 4 until the centroids no longer move

The process terminates when we find the values of the centroids from step 3 have not moved once we recalculated them in step 4 i.e. Their value was the same, no points moved cluster.



4(c) [14 Marks]

i. What is the purpose of *Agglomerative Hierarchical Clustering* and describe how it differs from the *k*-means approach?

ii. Explain what is meant by *single-link* clustering when used in hierarchical clustering.

iii. Using a single-link approach, create the **dendogram** for the data in the following table. Include the new **distance matrix** after each iteration.

	l	m	n	o	p	q	r
l	0	12	6	3	28	4	10
m	12	0	19	8	16	14	11
n	6	19	0	12	5	18	12
o	3	8	12	0	11	9	9
p	28	16	5	11	0	7	13
q	4	14	18	9	7	0	19
r	10	11	12	9	13	19	0

I)

The purpose of AHC is to produce a single large cluster made up of merged clusters forming a hierarchy of clusters. This differs to k-means as rather then creating k clusters the goal is to create a large single cluster. each object is given its own cluster and then the pair of nearest clusters are merged until only a single cluster exists.

ii)

single-link clustering is where by the distance between two clusters is taken to be the shortest from any one cluster to any member of the other cluster.

Iii)

1st merger

	lo	m	n	p	q	r
lo	0	8	6	11	4	9
m	8	0	19	16	14	11
n	6	19	0	5	18	12
p	11	16	5	0	7	13
q	4	14	18	7	0	19
r	9	11	12	13	19	0

2nd merger

	loq	m	n	p	r
loq	0	8	6	7	9
m	8	0	19	16	11
n	6	19	0	5	12
p	7	16	5	0	13
r	9	11	12	13	0

3rd merger

	loq	m	np	r
loq	0	8	6	9
m	8	0	16	11
np	6	16	0	12
r	9	11	12	0

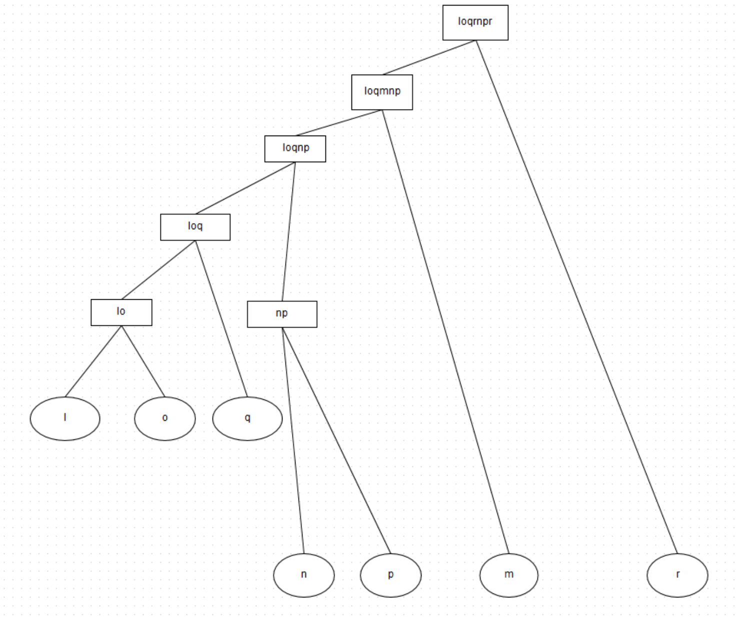
4th merger

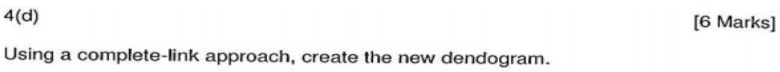
	lognp	m	r
lognp	0	8	9
m	8	0	16
r	9	16	0

5th merger

	loqrnpm	r
loqnpm	0	9
r	9	0

Dendogram**:**





4(d) [6 Marks]

Using a complete-link approach, create the new dendogram.

d)

same as above just use the longest distance