Question # 01

Minimum support count is 2
Minimum confidence is 60%

STEP 01

K = 1

(I) Create a table containing support count of each item present in dataset C1 (Candidate set)

Item set	sup-count
I1	6
I2	7
I3	6
I4	2
I5	2

(II) Compare candidate set item's support count with minimum support (min_support = 2 if support-count of candidate set items. This gives us itemset L1.

STEP 2

Generate candidate set C2 using L1. Now find support count of these itemsets by searching in dataset.

Itemset	sup-count
I1, I2	4
I1, I3	4
I1, I4	1
I1, I5	2
I2, I3	4

itemset	sup-count
I2, I4	2
I2, I5	2
I3, I4	0
I3, I5	1
I4, I5	0

Compare candidate C2 support count win min so, we get L2

Itemset	sup-count
I1, I2	4
I1, I3	4
I1, I5	2
I2, I3	4
I2, I4	2
I2, I5	2
I3, I5	2

STEP 3

Generating C3 using L2 (join step)
by joing L2 is {I1, I2, I3} {I1, I2, I5} {I2, I3, I4} {I2, I4, I5} {I2, I3, I5}
Checking frequent

Itemset	sup count
I1, I2, I3	2
I1, I2, I5	2

Compairing C3 with min sup-count

STEP 4

Generate C4 using L3. Condition for joing K=4, should have K-2. So, for L3, first 2 elements should match. Also checking all subsets of these items are frequent or not.

We stop here because no frequent itemsets are found.

Confidence

Confidence(A→B) = Support_count (A∪B) / Support_count (A)

Itemset {I1, I2, I3} from L3

So, Rules

[I1 ^ I2] => [I3] C = sup(I1 ^ I2 ^ I3) / sup(I1 ^ I2) = 2/4 × 100 = 50%

[I1 ^ I3] => [I2] C = sup(I1 ^ I2 ^ I3) / sup(I1 ^ I3) = 2/4 × 100 = 50%

[I2 ^ I3] => [I1] C = sup(" " / sup(I2 ^ I3) = 2/4 × 100 = 50%

[I1] => [I2 ^ I3] C = " / sup(I1) = 2/6 × 100 = 33%

[I2] => [I1 ^ I3] C = " / sup(I2) = 2/7 × 100 = 28%

[I3] => [I1 ^ I2] C = " / sup(I3) = 2/6 × 100 = 33%

Question # 02

Hebbian Learning Rule Algorithm

1. Set all weights to zero

   $w_i = 0$ for $i = 1$ to n & bias to zero

2. For each input vector, S(input) : t (target) repeat step 3-5.

3. Set activations for input units with the input vector $x_i = S_i$ for $i = 1$ to n.

4. Set the corresponding output value to the output neuron i.e. $y = t$.

5. Update weight & bias by applying Hebb rule for all $i = 1$ to n:

$$w_i(new) = w_i(old) + x_i y$$
$$b(new) = b(old) + y$$

Question # 05

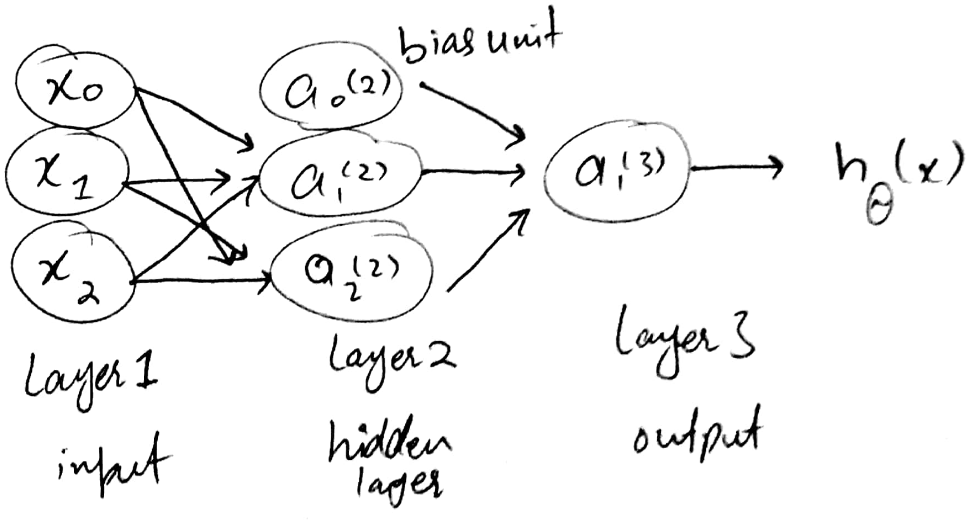
Backpropagation

It is a common method for traing a neural network. Backpropagation is a method to calculate the gradient of the loss function with respect to the weights in an artificial neural network. It is commonly used as a part of algorithms that optimize the performance of the network by adjusting the weights.

Back Propagation Training Algorithm

1. Initialize weights with random values
   Set other parameters
2. Read in the input vector & the desired output
3. Compute the actual output via the calculations, working forward through the layers.
4. Compute the error.

Example



Question # 08

Data Dimensionality Reduction Technique

There are several techniques for dadta dimensionality Reduction. Some of techniques are;

Low Variance Filter

Data colums with little changes in the data carry little information. Thus, all data columns with a variance lower than a given threshhold can be removed. Notice that the variance ~~lower~~ depends on the column range, and therefore normalization is required before applying thic technique.

High Correlation Filter

Data columns with very similar trends are also likely to carry very similar information, and only one of them will suffice for classification. Here we calculate the Pearson product-moment correlation coefficient between numeric colums & the Pearson's Chisquare value between nominal columns. For the final classification, we only retain one we only retain one column of each pair of columns whose pairwise correlation exceeds a given threshhold. Normalization is required before applying this technique.

Question # [redacted] 04

Consider the following data set (Training Set)

Income	No. of Siblings (x_2)	High School Grade x_1	Scholarship Award (y)
1 M	3	2.3	NO
0.5 M	4	3	Yes
0.2 M	2	3.5	Yes
0.9 M	3	2.9	NO

Testing Set

x_1	x_2	x_3	Y
0.7M	2	3	?

Using Euclidean Formula

Let $K = 2$

1st row $\quad (x_1 - y_2)^2 + (x_1 - y_2)^2$

$(1 - 0.7)^2 + (2.3 - 3)^2 = 0.98$

Similarly

2nd row $\quad (0.5 - 0.7)^2 + (3 - 3)^2 = 0.66$

3rd row $\quad (0.2 - 0.7)^2 + (3.5 - 3)^2 = 0.2$

4th row $\quad (0.9 - 0.7)^2 + (2.9 - 3)^2 = 2.33$

So,

x_1	x_2	E.D	Rank	Y	Included in 2 nearest neigh
1	2.3	0.98	3	No	No
0.5	3	0.66	2	Yes	Yes
0.2	3.5	0.2	1	Yes	Yes
0.9	2.9	2.33	4	No	No

The only 2 neighbours are included they both are awarded scholarship, so our test data will also result in a scholarship award = Yes

$\boxed{\therefore Y = \text{Yes}}$ Awarded

Question # 03

Apply Hierarchical Clustering

	x	y
z_1	0.4	0.53
z_2	0.22	0.38
z_3	0.35	0.32
z_4	0.26	0.19
z_5	0.08	0.41
z_6	0.45	0.31

Using Euclidean Distance

: d = distance

$$d(z_1, z_2) = \sqrt{(0.40-0.22)^2+(0.53-0.38)^2}$$
$$= 0.23$$

Similarly

$d(z_1, z_3) = 0.22$

$d(z_1, z_4) = 0.37$

$d(z_1, z_5) = 0.34$

$d(z_1, z_6) = 0.2256$

$d(z_2, z_3) = 0.15$

$d(z_2, z_4) = 0.20$

$d(z_2, z_5) = 0.14$

$d(z_2, z_6) = 0.240$

$d(z_3, z_4) = 0.15$

$d(z_3, z_5) = 0.28$

$d(z_3, z_6) = 0.1004$

$d(z_4, z_5) = 0.29$

$d(z_4, z_6) = 0.22$

$d(z_5, z_6) = 0.3832$

Distance Matrix

	z_1	z_2	z_3	z_4	z_5	z_6
z_1	0					
z_2	0.23	0				
z_3	0.22	0.15	0			
z_4	0.37	0.20	0.15	0		
z_5	0.34	0.14	0.28	0.29	0	
z_6	0.225	0.24	0.1004	0.22	0.38	0

Smallest Value $0.1004\ (z_3, z_6)$

Update the distance matrix Max dis

$\Rightarrow \text{Max}(dis(z_3, z_1), (z_6, z_1))$

$= (0.22, 0.2256)$

$= 0.2256$

$\Rightarrow \text{Max}(dis(z_3, z_6), z_2)$

$\text{Max}((z_3, z_2), (z_6, z_2))$

$= 0.24$

$\Rightarrow \text{Max}(dis(z_3, z_6), z_5)$

$\text{Max}(dis(z_3, z_5), (z_6, z_5))$

$\text{Max}(0.28, 0.38)$

$= 0.38$

Question # 04

(a) $(148 + 14432) \bmod 10$

$= (148 + 14432) \bmod 10$

$= [(148) \bmod 10] + [14432 \bmod 10] \bmod 10$

$= (8 + 2) \bmod 10$

$= 10 \bmod 10$

$= 0$

(b) $(424 \times 32) \bmod 10$

$[(424 \bmod 10) \times (32 \bmod 10)] \bmod 10$

$= (4 \times 2) \bmod 10$

$= 8 \bmod 10$

$= 8$

Update distance Matrix for cluster

	z_1	z_2	z_3	z_4	z_5
z_1	0				
z_2	0.23	0			
z_3	0.2256	0.24	0		
z_4	0.37	0.20	0.22	0	
z_5	0.34	$\boxed{0.14}$	0.38	0.29	0

Smallest Value 0.14 (z_2, z_5)

Update distance Matrix $Max(dis(z_2, z_5), z_1))$

$\Rightarrow Max(dis(z_2, z_1), (z_5, z_1))$

$(0.23, 0.34)$

$= 0.34$

$\Rightarrow Max(dis(z_2, z_5), (z_3, z_6))$

$Max(dis(z_2, z_3, z_6), (z_5, z_3, z_6))$

$M(0.24, 0.38)$

$= 0.38$

$\Rightarrow Max(dis(z_2, z_5), z_4))$

$Max((z_2, z_4), (z_5, z_4)$

$M(0.20, 0.29)$

$= 0.29$

Update distance Matrix

z_1	z_1	z_2, z_5	z_3, z_6	z_4
z_2, z_5	0.34	0		
z_4	0.37	0.29	0	
z_3, z_6	0.225	0.38	0.22	0

Smallest Value $(0.22 ((z_3, z_6), z_4)$

Again updating the Distance Matrix by finding Max for each, we get

$\Rightarrow$ Max $(dis((z_3, z_6), z_1), (z_4, z_1))$
$= 0.37$

$\Rightarrow$ Max $(dis((z_3, z_6), z_4), (z_2, z_5))$
$= 0.38$

Update Distance Matrix

	z_1	z_2, z_5	z_3, z_6, z_4
z_1	0		
z_2, z_5	0.34	0	
$z_3, z_6 z_4$	0.37	0.38	0

Smallest Value $(0.34 (z_2, z_5), z_1)$

Update Distance Matrix

$\Rightarrow \text{Max}(\text{dis}(z_2, z_5), (z_3, z_6, z_4), (z_3, z_6, z_4)$

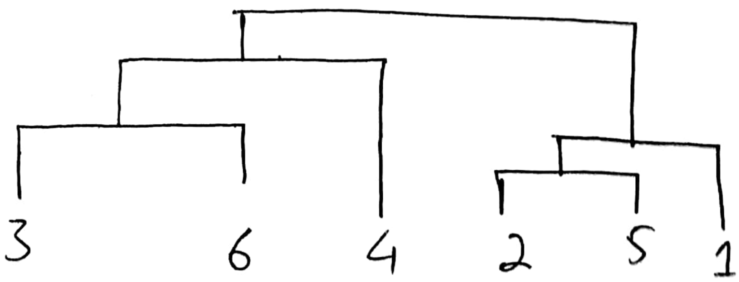
$\text{Max}(0.38, 0.37)$

$= 0.38$

Distance Matrix

	z_2, z_5, z_1	z_3, z_6, z_4
z_2, z_5, z_1	0	
z_3, z_6, z_4	0.38	0

Dendrogram



Question # 06

Applying K-Means Clustering

	Age	Salary
1	20	20,000
2	17	15,000
3	22	25,000
4	25	27,000
5	27	30,000

We choose two cluster randomly

$C_1(20, 20000)$ $C_2(17, 15,000)$

Cluster 1

Euclidean Distance

$$\text{Row } 3 = \sqrt{(22-20)^2 + (25000 - 20000)^2}$$
$$= 5000.0004$$
$$\text{Row } 4 = \sqrt{(25-20)^2 + (27000 - 20000)^2}$$
$$= 7000.00178$$
$$\text{Row } 5 = \sqrt{(27-20)^2 + (30000 - 20,000)^2}$$
$$= 10000.00245$$

Now Cluster 2

Similarly

Row 3 = 10008.5

Row 4 = 120000.00267

Row 5 = 15000.0033

So, the cluster values are

C_1	C_2
1 3 4 5	2

Now we find new cluster centroid

$$C_1 = \frac{20+22+25+27}{4}, \frac{20000+25000+27000+30000}{4}$$

$C_1 = (23.5, 25.500)$

$C_2 (17, 15000)$

⇒ Finding the value belonging the nearest cluster

Row 1 = 5500.00114

Row 3 = 500.00225

Row 4 = 1500.00075

Row 5 = 4500.001361

Now Centroid 2

Row 1 = 5000.009

Row 3 = 10005.5

Row 4 = 12000.0026

Row 5 = 15000.003

Same cluster again

C_1

1
3
4
5

C_2

2

Stopping this because of the same value.

Question 7

Apply PCA Algorithm on the given dataset:

~~Applying~~

Features →

Samples ↓

	Sepal-length	Sepal-width	Petal-L	Petal-w
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
148	5.9	3.0	5.1	1.8

Applying PCA on the two use-case:

→ Data Visualization

→ Speeding ML Algorithm

Using Sklearn's module datasets and library in Python; we get

2.110	0.702	2.3497	1.234	6.708	8.1109
1.704	2.085	8.9876	5.678	7.896	3.1108
0.702	2.045	3.873	9.861	3.4567	4.11089
1.83	2.336	5.679	10.11	3.977	9.5643
-1.808	1.221	7.891	1.1003	2.711	8.9764