Solution for Q42

1. Mean vector

$$\mu = \frac{1}{4}\left(x_1 + x_2 + x_3 + x_4\right)$$

$$x_1 + x_2 + x_3 + x_4 = [2 + 0 + 3 + 1 0 + 2 + 1 + 3] = [6 6]$$

$$\mu = \frac{1}{4}[6 6] = [1.5 1.5]$$

2. Centered data matrix

Subtract mean from each sample:

$$x_1 - \mu = [0.5 \ - 1.5], \ x_2 - \mu = [- 1.5 0.5]$$

$$x_3 - \mu = [1.5 \ - 0.5], \ x_4 - \mu = [- 0.5 1.5]$$

Centered data matrix

$$X_c = [0.5 \ - 1.5 1.5 \ - 0.5 \ - 1.5 0.5 \ - 0.5 1.5]$$

3. Covariance matrix

Use sample covariance:

$$C = \frac{1}{n-1}X_c X_c^T = \frac{1}{3}X_c X_c^T$$

Compute $X_c X_c^T$:

$$X_c X_c^T = [(0.5)^2 + (- 1.5)^2 + (1.5)^2 + (- 0.5)^2 \ (0.5)(- 1.5) + (- 1.5)(0.5) + (1.5)(- 0.5)$$

$$+ (- 0.5)(1.5) \ same \ (- 1.5)^2 + (0.5)^2 + (- 0.5)^2 + (1.5)^2]$$

Diagonal:

$$0.25 + 2.25 + 2.25 + 0.25 = 5.0$$

Off-diagonal:

$$0.5(- 1.5) =- 0.75$$

$$-1.5(0.5) = -0.75$$

$$1.5(-0.5) = -0.75$$

$$-0.5(1.5) = -0.75$$

Sum = −3.0

Thus:

$$X_c X_c^T = [5 \ -3 \ -3 \ 5]$$

Now divide by 3:

$$C = \frac{1}{3}[5 \ -3 \ -3 \ 5] = [1.6667 \ -1.0000 \ -1.0000 \ 1.6667]$$

4. Eigenvalues and eigenvectors

Solve:

$$det(C - \lambda I) = 0$$

$$det[1.6667 - \lambda \ -1 \ -1 \ 1.6667 - \lambda] = (1.6667 - \lambda)^2 - 1 = 0$$

$$1.6667 - \lambda = \pm 1$$

Eigenvalues:

$$\lambda_1 = 1.6667 + 1 = 2.6667$$

$$\lambda_2 = 1.6667 - 1 = 0.6667$$

Eigenvectors

For $\lambda_1 = 2.6667$:

Eigenvector:

$$v_1 = [1 \ -1]$$

For $\lambda_2 = 0.6667$:

Eigenvector:

$$v_2 = [1\ 1\ ]$$

Eigenvalues and normalized eigenvectors

$$\lambda_1 = 2.6667, \quad v_1 = [0.7071\ \ -\ 0.7071\ ]$$

$$\lambda_2 = 0.6667, \quad v_2 = [0.7071\ 0.7071\ ]$$

5. First principal component and variance explained

The **largest eigenvalue** = 2.6667

So the **first principal component direction** is:

$$v_1 = [0.7071\ \ -\ 0.7071\ ]$$

Total variance:

$$\lambda_1 + \lambda_2 = 2.6667 + 0.6667 = 3.3334$$

Proportion:

$$\frac{2.6667}{3.3334} = 0.8000$$

Solution for Q43

We will consider a solution for a binary decision tree as well.

0) Data summary (12 examples)

Class counts (decisions):

- ➔ **cinema**: 5
- ➔ **tennis**: 3
- ➔ **stay-in**: 2
- ➔ **shopping**: 2

Total = 12

1) Root entropy $H(S)$

Probabilities:

- ➔ $p(cinema) = 5/12$
- ➔ $p(tennis) = 3/12 = 1/4$
- ➔ $p(stay - in) = 2/12 = 1/6$
- ➔ $p(shopping) = 2/12 = 1/6$

Entropy formula: $H = - \sum p_i log_2 p_i$.

Compute contributions:

- ➔ cinema: $- \frac{5}{12} log_2 \frac{5}{12} = - 0.4166666667 \times log_2(0.4166666667) = 0.528096$
- ➔ tennis: $- \frac{1}{4} log_2 \frac{1}{4} = 0.5$
- ➔ stay-in: $- \frac{1}{6} log_2 \frac{1}{6} = 0.430548$
- ➔ shopping: same as stay-in = 0.430548

Sum:

$H(S) = 0.528096 + 0.5 + 0.430548 + 0.430548 = 1.8879185026711327$

(rounded: **1.8879**)

2) Information gain for each candidate root attribute

We compute $H(S|Attribute)$ = weighted sum of entropies of attribute-value subsets, then $IG = \ H(S) - \ H(S|Attribute)$.

A — Attribute = **Weather** (values: sunny, windy, rainy)

Each value has 4 examples (4+4+4 =12).

Sunny (4 examples): decisions = {cinema:1, tennis:3}

- Probabilities: 1/4 and 3/4 → entropy
  $H(sunny) = -\frac{1}{4}log_2\frac{1}{4} - \frac{3}{4}log_2\frac{3}{4} = \ 0.8112781244591328$

Windy (4 examples): decisions = {cinema:2, shopping:1, stay-in:1}

- Probabilities: 2/4,1/4,1/4 → entropy
  $H(windy) = -\frac{1}{2}log_2\frac{1}{2} - \frac{1}{4}log_2\frac{1}{4} - \frac{1}{4}log_2\frac{1}{4} = \ 1.5$

Rainy (4 examples): decisions = {cinema:2, stay-in:1, shopping:1} → same composition → $H(rainy) = \ 1.5$

Weighted conditional entropy:

$H(S|Weather) = \frac{4}{12}(0.811278) + \frac{4}{12}(1.5) + \frac{4}{12}(1.5) = \ 1.2704260414863775$

Information gain:

$IG(Weather) = \ 1.8879185026711327 \ - \ 1.2704260414863775 \ = 0.617492461185$

B — Attribute = **Parents** (values: visit, no-visit)

Partition counts:

- visit: 6 examples
- no-visit: 6 examples

Compute entropies (calculated exactly):

$$H(S|Parents) = \ 1.5220552088742003$$

$$IG(Parents) = 1.8879185026711327 - 1.5220552088742003 = 0.3658632937969$$

C — Attribute = **Cash** (rich / poor)

$$H(S|Cash) = 1.603759374819711$$

$$IG(Cash) = 1.8879185026711327 - 1.603759374819711 = 0.2841591278514217$$

D — Attribute = **Exam** (yes / no)

$$H(S|Exam) = 1.8553885422075336$$

$$IG(Exam) = 1.8879185026711327 - 1.8553885422075336 = 0.032529960463599$$

Choose root = WEATHER (largest IG = **0.61749**)

3) Build subtrees for each WEATHER value

There are three branches: **SUNNY**, **WINDY**, **RAINY**. Each branch has 4 training examples.

Branch A — WEATHER = SUNNY (4 examples)

Examples (by id / outcome):
1 → cinema, 2 → tennis, 10 → tennis, 11 → tennis
Counts: cinema = 1, tennis = 3.
Root entropy at this node:

$$H = -\frac{1}{4}log_2\frac{1}{4} - \frac{3}{4}log_2\frac{3}{4} = 0.8112781244591328$$

We consider the remaining attributes {Parents, Cash, Exam} for splitting this 4-example subset.

Compute IG(Parents | sunny):

Partition by Parents inside sunny:

- ➔ visit: 1 example → (cinema) → entropy 0
- ➔ no-visit: 3 examples → (tennis, tennis, tennis) → entropy 0

Weighted entropy: $H(sunny|Parents) = \frac{1}{4} \cdot 0 + \frac{3}{4} \cdot 0 = 0.$

So

$IG(Parents|sunny) = 0.8112781244591328 - 0 = 0.8112781244591328$

Because this splits the sunny node into pure leaves, **Parents** is the perfect split here.

Decision for SUNNY:

- ➔ If Parents = **visit** → **cinema**
- ➔ If Parents = **no-visit** → **tennis**

Branch B — WEATHER = WINDY (4 examples)

Examples (decisions and attributes):

- ➔ example 3: (windy, visit, rich, no) → **cinema**
- ➔ example 7: (windy, no-visit, poor, yes) → **cinema**
- ➔ example 8: (windy, no-visit, rich, yes) → **shopping**
- ➔ example 12:(windy, visit, rich, yes) → **stay-in**

Counts: cinema = 2, shopping = 1, stay-in = 1.
Root entropy at this node: $H(windy) = 1.5$

Compute IG for the three candidate attributes (Parents, Cash, Exam) **within** the windy subset:

- **Parents** split:
    - visit subset: 2 examples → decisions {cinema, stay-in} → probabilities 1/2,1/2 → entropy = 1.0
    - no-visit subset: 2 examples → decisions {cinema, shopping} → entropy = 1.0
    - Weighted entropy: $\frac{2}{4} \cdot 1 + \frac{2}{4} \cdot 1 = 1.0$
    - $IG(Parents|windy) = 1.5 - 1.0 = 0.5$
- **Cash** split:
    - poor: 1 example → (cinema) → entropy 0
    - rich: 3 examples → (cinema, shopping, stay-in) → counts 1,1,1 → distribution 1/3 each → entropy $= - 3 \cdot \frac{1}{3} log_2 \frac{1}{3} = log_2 3 \approx 1.5849625$
    - Weighted entropy: $\frac{1}{4} * 0 + \frac{3}{4} * 1.5849625 = 1.188721875540867$
    - $IG(Cash|windy) = 1.5 - 1.188721875540867 = 0.31127812445913294$
- **Exam** split:
    - no: 1 example → (cinema) → entropy 0
    - yes: 3 examples → (cinema, shopping, stay-in) → same situation as Cash-rich → entropy 1.5849625
    - Weighted entropy: same as Cash → 1.188721875540867
    - $IG(Exam|windy) = 0.31127812445913294$

Choose Parents at the windy node (largest IG = **0.5**).

Now handle the two child nodes produced by Parents split.

B1 — WINDY & Parents = visit (2 examples)

Examples:

- ex3: (windy, visit, rich, no) → cinema
- ex12:(windy, visit, rich, yes) → stay-in

Counts: {cinema:1, stay-in:1}
Entropy at this node:

$$H =- \frac{1}{2}log_2\frac{1}{2} - \frac{1}{2}log_2\frac{1}{2} = 1.0$$

Remaining attributes: {Cash, Exam}. Evaluate IG:

- **Cash**: both examples have Cash = rich → Cash partition would be a single group (no information). Weighted entropy = 1.0 → IG=0.

- **Exam**: partitions:

  - no → ex3 → cinema (pure, entropy 0)
  - yes → ex12 → stay-in (pure, entropy 0)
  Weighted entropy = 0 → IG = $1.0 - 0 = 1.0$

So split on **Exam** here:

- If Exam = **no** → **cinema**
- If Exam = **yes** → **stay-in**

B2 — WINDY & Parents = no-visit (2 examples)

Examples:

- ex7: (windy, no-visit, poor, yes) → cinema
- ex8: (windy, no-visit, rich, yes) → shopping

Counts: {cinema:1, shopping:1} → entropy = 1.0

Remaining attributes: {Cash, Exam}. Evaluate IG:

- **Exam**: both examples have Exam = yes → no split (IG=0).

- **Cash**: partitions:

  - poor → ex7 → cinema (pure)
  - rich → ex8 → shopping (pure)
    Weighted entropy = 0 → IG = $1.0 - 0 = 1.0$

So split on **Cash**:

- If Cash = **poor** → **cinema**

- If Cash = **rich** → **shopping**

Branch C — WEATHER = RAINY (4 examples)

Examples:

- ex4: (rainy, visit, poor, yes) → cinema
- ex5: (rainy, no-visit, rich, no) → stay-in
- ex6: (rainy, visit, poor, no) → cinema
- ex9: (rainy, visit, rich, no) → shopping

Counts: cinema = 2, stay-in = 1, shopping = 1
Root entropy at rainy node:

$$H(rainy) = 1.5$$

Evaluate IG for attributes {Parents, Cash, Exam} within this subset:

- **Cash** split:

  - poor: 2 examples → both cinema → entropy 0

- rich: 2 examples → decisions {stay-in, shopping} → entropy = 1.0 (because 1 each)
- Weighted entropy: $\frac{2}{4} \cdot 0 + \frac{2}{4} \cdot 1 = 0.5$
- $IG(Cash|rainy) = 1.5 - 0.5 = 1.0$ → perfect split (very strong)

❖ **Parents** split:

- visit: 3 examples → decisions {cinema:2, shopping:1} → entropy = 0.9182958340544896
- no-visit: 1 example → (stay-in) → entropy 0
- Weighted entropy: $\frac{3}{4} \cdot 0.918296 + \frac{1}{4} \cdot 0 = 0.6887218755408672$
- $IG(Parents|rainy) = 1.5 - 0.6887218755408672 = 0.811278124459133$ (also good, but Cash is better here)

❖ **Exam** split:

- yes: 1 example (cinema) → entropy 0
- no: 3 examples → {cinema, stay-in, shopping} → counts 1,1,1 → entropy = 1.5849625
- Weighted entropy = 1.188721875540867
- $IG(Exam|rainy) = 0.31127812445913294$

Choose Cash at the rainy node (IG = **1.0**).

Now the Cash children:

C1 — RAINY & Cash = poor (2 examples)

Examples: ex4 and ex6 → both **cinema**
Entropy = 0 → leaf: **cinema**

C2 — RAINY & Cash = rich (2 examples)

Examples:

- ➔ ex5: (no-visit) → stay-in
- ➔ ex9: (visit) → shopping

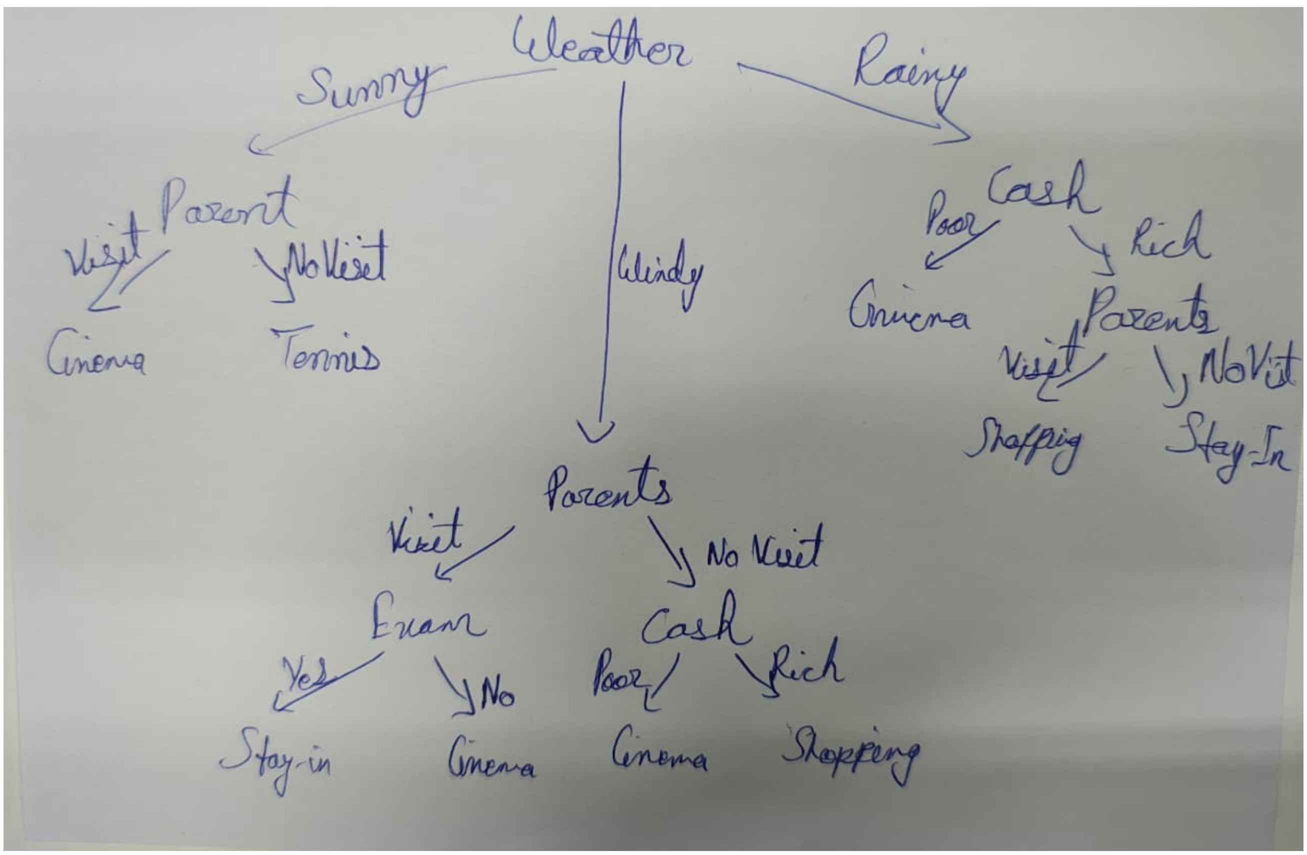
Entropy at this node = 1.0 (one of each). Remaining attributes: Parents, Exam.

- **Parents** splits:
  - ➢ visit → ex9 → shopping (pure)
  - ➢ no-visit → ex5 → stay-in (pure)
    Weighted entropy = 0 → IG = 1.0
- **Exam**: both examples have Exam = no here → cannot split.

So split on **Parents**:

- ➔ If Parents = **visit** → **shopping**

- ➔ If Parents = **no-visit** → **stay-in**



Solution for Q44

Step 1: Prior probabilities

Count the outcomes:

Hirable	Count
Yes	3 (Sarah, Annie, Emily)
No	5 (Dana, Alex, Pete, John, Kathy)
Total	8

$$P(Yes) = \frac{3}{8}, \quad P(No) = \frac{5}{8}$$

Step 2: Compute conditional probabilities

We need:

- ➔ $P(GPA\ = poor|Yes)$
- ➔ $P(Effort\ = lots|Yes)$
- ➔ $P(GPA\ = poor|No)$
- ➔ $P(Effort\ = lots|No)$

A. Conditional probabilities given HIRE = YES

Examples where **Yes**:
Sarah(poor,lots), Annie(average,lots), Emily(excellent,lots)

Total = 3.

GPA

poor = 1 (Sarah)

$$P(GPA\ =\ poor|Yes) = \frac{1}{3}$$

Effort

lots = 3 (all Yes examples)

$$P(Effort\ =\ lots|Yes) = \frac{3}{3} =\ 1$$

B. Conditional probabilities given HIRE = NO

Examples where **No**:
Dana(avg,some), Alex(avg,some), Pete(exc,lots), John(exc,lots), Kathy(poor,some)

Total = 5.

GPA

poor = 1 (Kathy)

$$P(GPA\ =\ poor|No) = \frac{1}{5}$$

Effort

lots = 2 (Pete, John)

$$P(Effort\ =\ lots|No) = \frac{2}{5}$$

Step 3: Apply Naive Bayes

We compare:

$$P(Yes|poor,\ lots) \propto\ P(Yes)\, P(poor|Yes)\, P(lots|Yes)$$

$$P(No|poor,\ lots) \propto\ P(No)\, P(poor|No)\, P(lots|No)$$

Compute the scores

For YES:

$$Score(Yes) = \frac{3}{8} \cdot \frac{1}{3} \cdot\ 1$$

$$Score(Yes) = \frac{3}{8} \cdot \frac{1}{3} = \frac{1}{8} =\ 0.125$$

For NO:

$$Score(No) = \frac{5}{8} \cdot \frac{1}{5} \cdot \frac{2}{5}$$

$$= \left(\frac{5}{8} \cdot \frac{1}{5}\right) \cdot \frac{2}{5}$$

$$= \frac{1}{8} \cdot \frac{2}{5} = \frac{2}{40} = 0.05$$

Step 4: Compare scores

$$Score(Yes) = 0.125$$

$$Score(No) = 0.05$$

$$0.125 > 0.05$$

Final Decision: The student SHOULD be hired.

Solution for Q45

A) Compute network output $\hat{y}$

1. Hidden layer pre-activations

Neuron 1:

$$z_1 = 0.12(1) + (-0.07)(2) + 0.05$$

$$= 0.12 - 0.14 + 0.05 = 0.03$$

Neuron 2:

$$z_2 = 0.25(1) + 0.10(2) - 0.02$$

$$= 0.25 + 0.20 - 0.02 = 0.43$$

2. Hidden activations

$$h_1 = \sigma(0.03) = 0.5075$$

$$h_2 = \sigma(0.43) = 0.6050$$

3. Output neuron pre-activation

$$z^{out} = 0.40h_1 - 0.30h_2 + 0.10$$

$$= 0.40(0.5075) - 0.30(0.6050) + 0.10$$

$$= 0.2030 - 0.1815 + 0.10 = 0.1215$$

4. Final output

$$\hat{y} = \sigma(0.1215) = 0.5304$$

B) Compute gradients

Let:

$$\delta^{(2)} = \frac{\partial L}{\partial z^{out}} = \left(\hat{y} - y\right)\hat{y}\left(1 - \hat{y}\right)$$

$$\hat{y} = 0.5304, \; y = 0$$

$$\delta^{(2)} = 0.5304(0.5304)(1 - 0.5304) = 0.5304(0.2491) = 0.1322$$

GRADIENTS FOR OUTPUT LAYER

Weight gradients $W^{(2)}$

$$\frac{\partial L}{\partial W_i^{(2)}} = \delta^{(2)} h_i$$

For $W_1^{(2)}$:

$$0.1322(0.5075) = 0.0671$$

For $W_2^{(2)}$:

$$0.1322(0.6050) = 0.0799$$

Bias gradient

$$\frac{\partial L}{\partial b^{(2)}} = \delta^{(2)} = 0.1322$$

GRADIENTS FOR HIDDEN LAYER

Hidden layer deltas:

$$\delta_i^{(1)} = \delta^{(2)} W_i^{(2)} h_i \left(1 - h_i\right)$$

Neuron 1

$$\delta_1^{(1)} = 0.1322(0.40)(0.5075)(0.4925) = 0.0132$$

Neuron 2

$$\delta_2^{(1)} = 0.1322(-0.30)(0.6050)(0.3950) = -0.0095$$

Hidden-layer weight gradients

$$\frac{\partial L}{\partial W_{ij}^{(1)}} = \delta_i^{(1)} x_j$$

Row 1:

- ➔ $0.0132(1) = 0.0132$
- ➔ $0.0132(2) = 0.0264$

Row 2:

- ➔ $-0.0095(1) = -0.0095$
- ➔ $-0.0095(2) = -0.0190$

Bias gradients

$$\frac{\partial L}{\partial b_1^{(1)}} = 0.0132$$

$$\frac{\partial L}{\partial b_2^{(1)}} = -\ 0.0095$$

C) Parameter updates (η = 0.2)

$$\theta_{new} = \theta - \eta \frac{\partial L}{\partial \theta}$$

Output layer

Weights:

$$W_1^{(2)} = 0.40 - 0.2(0.0671) = 0.3866$$

$$W_2^{(2)} = -\ 0.30 - 0.2(0.0799) = -\ 0.3160$$

Bias:

$$b^{(2)} = 0.10 - 0.2(0.1322) = 0.0736$$

Hidden layer

Row 1:

$$W_{11}^{(1)} = 0.12 - 0.2(0.0132) = 0.1174$$

$$W_{12}^{(1)} = -\ 0.07 - 0.2(0.0264) = -\ 0.0753$$

Row 2:

$$W_{21}^{(1)} = 0.25 - 0.2(-\ 0.0095) = 0.2519$$

$$W_{22}^{(1)} = 0.10 - 0.2(-\ 0.0190) = 0.1038$$

Biases:

$$b_1^{(1)} = 0.05 - 0.2(0.0132) = 0.0474$$

$$b_2^{(1)} = -\ 0.02 - 0.2(-\ 0.0095) = -\ 0.0181$$

Final updated parameters (rounded to 4 decimals)

Hidden layer weights

$$W_{new}^{(1)} = [0.1174 \ - \ 0.0753 \ 0.2519 \ 0.1038 \ ]$$

Hidden layer biases

$$b_{new}^{(1)} = [0.0474, \quad - \ 0.0181]$$

Output layer weights

$$W_{new}^{(2)} = [0.3866, \quad - \ 0.3160]$$

Output layer bias

$$b_{new}^{(2)} = \ 0.0736$$