1.1 $-\ H(\text{Overweight}) = -\frac{4}{6}\log_2\frac{4}{6} - \frac{2}{6}\log_2\frac{2}{6} = 0.9183$

$-\ H(\text{Overweight} \mid \text{gender}) = \frac{4}{6}H(\frac{1}{2},\frac{1}{2}) + \frac{2}{6}H(1,0) = 0.6667$

$\Rightarrow I(\text{Overweight} : \text{gender}) = 0.2516$

$-\ H(\text{Overweight} \mid \text{Hyperlipidemia}) = \frac{3}{6}H(1,0) + \frac{3}{6}H(\frac{1}{3},\frac{2}{3}) = 0.4592$

$\Rightarrow I(\text{Overweight} : \text{Hyperlipidemia}) = 0.9183 - 0.4592 = 0.4591$

$-\ H(\text{Overweight} \mid \text{Unhealthy diet}) = \frac{2}{6}H(\frac{1}{2},\frac{1}{2}) + \frac{4}{6}H(\frac{3}{4},\frac{1}{4}) = 0.8742$

$\Rightarrow I(\text{Overweight} : \text{Unhealthy diet}) = 0.0441$

$-\ H(\text{Overweight} \mid \text{exercises}) = \frac{4}{6}H(\frac{1}{2},\frac{1}{2}) + \frac{2}{6}H(1,0) = 0.6667$

$\Rightarrow I(\text{Overweight} : \text{exercises}) = 0.2516$

We choose Hyperlipidemia as the 1st split. (Highest bits)

$\Rightarrow$ For node K:

$H(\text{overweight}) = H(\frac{1}{3},\frac{2}{3}) = 0.9183$

$-\ H(\text{Overweight} \mid \text{gender}) = \frac{1}{3}H(1,0) + \frac{2}{3}H(1,0) = 0$

$\Rightarrow I(\text{Overweight} : \text{gender}) = 0.9183$

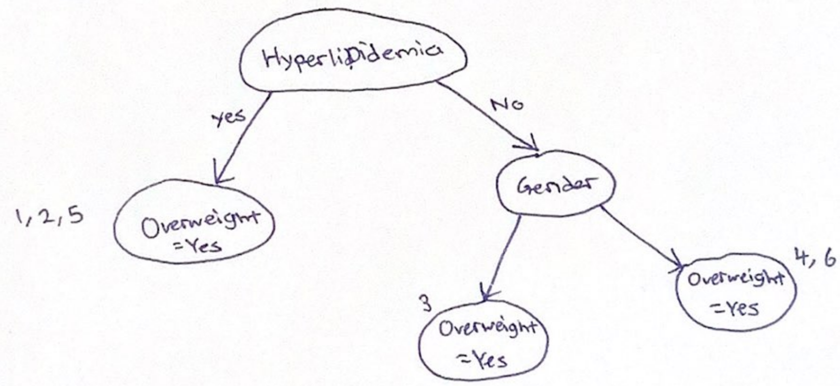
$-\ H(\text{Overweight} \mid \text{Unhealthy diet}) = \frac{2}{3}H(\frac{1}{2},\frac{1}{2}) + \frac{1}{3}H(1,0) = 0.6667$

$\Rightarrow I(\text{Overweight} : \text{Unhealthy diet}) = 0.2516$

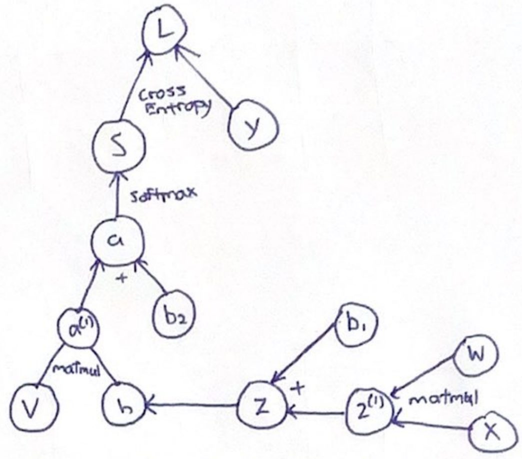
$-\ H(\text{Overweight} \mid \text{exercises}) = \frac{1}{3}H(1,0) + \frac{2}{3}H(1,0) = 0$

$\Rightarrow I(\text{Overweight} : \text{exercises}) = 0.9183$

We choose either gender or exercise (same bits)



1.2 Computational Graph



1. Gradients with respect to V in Output Layer:

$$\frac{\partial L}{\partial V} = \frac{\partial L}{\partial a} \cdot \frac{\partial a}{\partial V}$$

Given: $u_2 = \frac{\partial L}{\partial a} = (p-y)$ and $a = Vh + b_2$, where $h = ReLU(z)$

$$\frac{\partial L}{\partial V} = u_2 \otimes h$$

2. Gradients with respect to b_2 in Output Layer:

$$\frac{\partial L}{\partial b_2} = \frac{\partial L}{\partial a} \cdot \frac{\partial a}{\partial b_2}$$

Given: $u_2 = \frac{\partial L}{\partial a} = (p-y)$ and $a = Vh + b_2$

$$\frac{\partial L}{\partial b_2} = u_2$$

3. Gradients with respect to W in Hidden Layer:

$$\frac{\partial L}{\partial W} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial W}$$

Given: $u_1 = \frac{\partial L}{\partial z} = (V^T u_2) \odot H(z)$ and $z = Wx + b_1$, where $h = ReLU(z)$

$$\frac{\partial L}{\partial W} = u_1 \otimes x$$

4. Gradients with respect to b_1 in Hidden Layer:

$$\frac{\partial L}{\partial b_1} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial b_1}$$

Given: $u_1 = \frac{\partial L}{\partial z} = (V^T u_2) \odot H(z)$ and $z = Wx + b_1$, where $h = ReLU(z)$

$$\frac{\partial L}{\partial b_1} = u_1$$

5. Gradients with respect to x in Input Layer:

$$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial x}$$

Given: $u_1 = \frac{\partial L}{\partial z} = (V^T u_2) \odot H(z)$ and $z = Wx + b_1$, where $h = ReLU(z)$

$$\frac{\partial L}{\partial x} = (W^T u_1)$$

1.3 If there are no hidden layers, then

$$T_k = \beta_{0k} + \beta_k^T X, \; k = 1, \ldots, K$$

thus

$$f_k(X) = g_k(X)$$

$$= \frac{\exp(\beta_{0k} + \beta_k^T X)}{\sum_{l=1}^{K} \exp(\beta_{0l} + \beta_l^T X)}$$

If we normalize these probabilities by

$$f_k(x) \leftarrow \frac{f_k(x)}{f_K(x)} \cdot \frac{1}{1 + \sum_{l=1}^{K-1} \exp(\beta_{0l} + \beta_l^T X)}, \; k = 1, \ldots, K$$

we get exactly the multinomial logistic model.

1.4

a)

Layer	Activation Volume Dimensions (memory)
INPUT	32×32×3
CONV5-10	32×32×10
POOL2	16×16×10
CONV5-10	16×16×10
POOL2	8×8×10
FC-10	10

~~Output size = (input size) + 2×padding~~

$$\text{Output size} = \left\lfloor \frac{\text{input size} + 2\times\text{padding} - (\text{kernel size} - 1) - 1}{\text{stride}} + 1 \right\rfloor$$

- Conv5(10)
  Input shape = 32×32×3 → Output shape = 32×32×10
- Maxpool$_2$
  Input shape = 32×32×10 → Output shape = 16×16×10
- Conv5(10)
  Input shape = 16×16×10 → Output shape = 16×16×10
- Maxpool$_2$
  Input shape = 16×16×10 → Output shape = 8×8×10
- FC10
  Input shape = 8×8×10 → Output shape = 10

b)

Layer	Number of parameters
INPUT	0
CONV5-10	10×(5×5×3+1)
Maxpool$_2$	0
CONV5-10	10×(5×5×10+1)
Maxpool$_2$	0
FC10	10×(8×8×10+1)

1.5 Dropout is effective in preventing complex co-adaptation of hidden units in neural networks

a) through the following intuitive reasons:

1. Forcing Robust Representations:
   - Dropout encourages the learning of robust and versatile features by preventing neurons from relying on the presence of specific other neurons.
2. Redundancy and Ensemble Learning:
   - It acts as a form of ensemble learning by training different subnetworks on each iteration, leading to the learning of redundant representations and reducing sensitivity to specific neuron configurations.
3. Reduce Overfitting:
   - Dropout prevents overfitting by discouraging the co-adaptation of neurons, which can be lead to specialization on the training data
4. Promoting independence:
   - It promotes the development of independent neurons that contribute to the model's performance in various contexts, encouraging diversity in learned features.
5. Increasing Generalisation:
   - The stochastic nature of dropout exposes the model to different subsets of neurons, aiding in generalization to unseen data and making the model adaptable to a variety of situations

b) The proportional coefficient $\frac{1}{1-p}$ in the dropout implementation is introduced to preserve the expectation of the activations during training. It compensates for the dropout of hidden units by normalizing the retained units, ensuring that the average value of the activation remains consistent. This scaling factor prevents a reduction in the overall magnitude of the activations, maintains consistency between training and inference, and supports the dropout technique's goal of preventing co-adaptation and improving generalisation

c) To derive the gradient $\frac{\partial L}{\partial h^{(l)}}$ for a dropout layer, where $h^{(l+1)} = h^{(l)} \odot \text{mask}$, we can use the chain rule. Lets express $\frac{\partial L}{\partial h^{(l)}}$ in terms of $\frac{\partial L}{\partial h^{(l+1)}}$ and the derivative of the dropout operation.

Given that $h^{(l+1)} = h^{(l)} \odot \text{mask}$, the mask is a binary representing the dropped out (zeroed) and retained (non-zero) elements. Let mask_{ij} be an element of the mask.

The dropout operation can be expressed as:

$$h_{ij}^{(l+1)} = \text{mask}_{ij} \cdot h_{ij}^{(l)}$$

Now, let's derive the gradient:

$$\frac{\partial L}{\partial h^{(l)}_{ij}} = \frac{\partial L}{\partial h^{(l+1)}_{ij}} \cdot \frac{\partial h_{ij}^{(l+1)}}{\partial h_{ij}^{(l)}}$$

Since $h_{ij}^{(l+1)} = \text{mask}_{ij} \cdot h_{ij}^{(l)}$, we have:

$$\frac{\partial h_{ij}^{(l+1)}}{\partial h_{ij}^{(l)}} = \text{mask}_{ij}$$

Therefore, gradient is:

$$\frac{\partial L}{\partial h_{ij}^{(l)}} = \frac{\partial L}{\partial h_{ij}^{(l+1)}} \cdot \text{mask}_{ij}$$

1.6

a) $f(x) = \sigma(0.5x_1 + (-0.1)x_2)$

Positive samples:

$(5,5) \rightarrow \sigma(2) = \frac{1}{1+e^{-2}} \approx 0.88$

$(3,8) \rightarrow \sigma(0.7) = \frac{1}{1+e^{-0.7}} \approx 0.67$

$(-1,8) \rightarrow \sigma(-1.3) = \frac{1}{1+e^{-(-1.3)}} \approx 0.21$

$(5,4) \rightarrow \sigma(2.1) = \frac{1}{1+e^{-2.1}} \approx 0.89$

$(-1,-2) \rightarrow \sigma(-0.3) = \frac{1}{1+e^{-(-0.3)}} \approx 0.42$

Negative samples:

$(-5,1) \rightarrow \sigma(-2.6) = \frac{1}{1+e^{-(-2.6)}} \approx 0.07$

$(2,-2) \rightarrow \sigma(1.2) = \frac{1}{1+e^{-1.2}} \approx 0.76$

$(-1,1) \rightarrow \sigma(-0.6) = \frac{1}{1+e^{-(-0.6)}} \approx 0.35$

$(-5,-10) \rightarrow \sigma(-1.5) = \frac{1}{1+e^{-(-1.5)}} \approx 0.18$

$(-10,-9) \rightarrow \sigma(-4.1) = \frac{1}{1+e^{-(-4.1)}} \approx 0.01$

b)

	N1	N2	N3	P1	N4	P2	P3	N5	P4	P5
y	0.01	0.07	0.18	0.21	0.35	0.42	0.67	0.76	0.88	0.89

↑ threshold $= 0.5$

Confusion Matrix:

	$\hat{P}$ (predicted)	$\hat{N}$ (predicted)
P (actual)	TP = 3	FN = 2
N (actual)	FP = 1	TN = 4

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} = \frac{3+4}{10} = \frac{7}{10}$$

$$\text{Precision} = \frac{TP}{TP+FP} = \frac{3}{3+1} = \frac{3}{4}$$

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{3}{3+2} = \frac{3}{5}$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = 2 \times \frac{\frac{3}{4} \times \frac{3}{5}}{\frac{3}{4} + \frac{3}{5}} = \frac{2}{3}$$

c) $TPR = \frac{TP}{TP + FN}$ $FPR = \frac{FP}{FP + TN}$

	$\hat{P}$ (predicted)	$\hat{N}$ (predicted)
P (actual)	TP = 5	FN = 0
N (actual)	FP = 5	TN = 0

$t = 0$

	$\hat{P}$ (predicted)	$\hat{N}$ (predicted)
P (actual)	TP = 5	FN = 0
N (actual)	FP = 2	TN = 3

$t = 0.2$

	$\hat{P}$ (predicted)	$\hat{N}$ (predicted)
P (actual)	TP = 4	FN = 1
N (actual)	FP = 1	TN = 4

$t = 0.4$

	$\hat{P}$ (predicted)	$\hat{N}$ (predicted)
P (actual)	TP = 3	FN = 2
N (actual)	FP = 1	TN = 4

$t = 0.6$

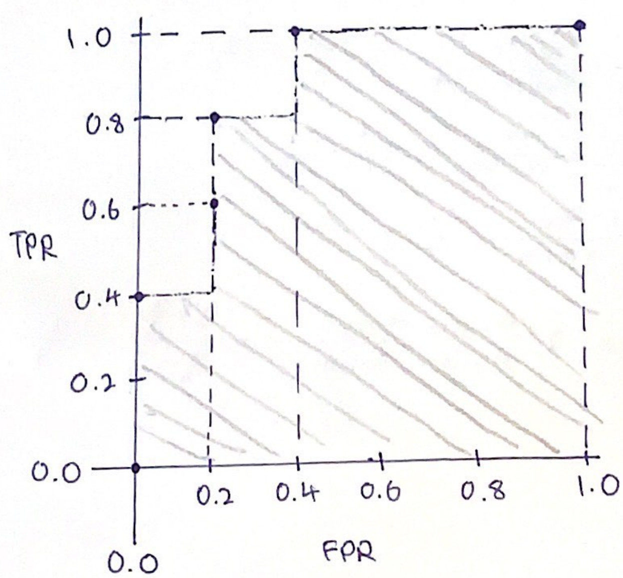
	$\hat{P}$ (predicted)	$\hat{N}$ (predicted)
P (actual)	TP = 2	FN = 3
N (actual)	FP = 0	TN = 5

$t = 0.8$

	$\hat{P}$ (predicted)	$\hat{N}$ (predicted)
P (actual)	TP = 0	FN = 5
N (actual)	FP = 0	TN = 5

$t = 1.0$

Threshold	TPR	FPR
0	1	1
0.2	1	$\frac{2}{5}$
0.4	$\frac{4}{5}$	$\frac{1}{5}$
0.6	$\frac{3}{5}$	$\frac{1}{5}$
0.8	$\frac{2}{5}$	0
1.0	0	0



d) Area under ROC curve $= 0.2 \times 0.4 + 0.2 \times 0.8 + 0.6 \times 1.0 = 0.84$