Introduction

1. Give a definition of “Pattern Recognition”.

2. Which of the following problems is a suitable application for pattern recognition?

a. Classifying numbers into primes and non-primes.

b. Detecting potential fraud in credit card charges.

c. Determining where a cannon ball is likely to land given information about elevation and direction of the barrel, wind direction, etc.

d. Identifying the species a newly discovered “bug” belongs to.

3. Give brief definitions of the following terms:

a. Exemplar

b. Dataset

c. Generalization

d. Overfitting

e. Decision Theory

f. Feature Space

g. Linearly Separable

h. Dichotomizer

i. Hyper-Parameter

j. Grid search

k. Training data

l. Test data

4. What types of learning best describe the following three scenarios, in which a coin classification system is being created for a vending machine.

a. Measurements of a large number of coins are taken. The algorithm finds that these measurements fall in several “bins”. It finds decision boundaries that separate these bins and uses these to classify new coins.

b. The measurements of each coin is presented to the classifier which makes a decision. The classifier changes its decision boundaries based on whether this decision was correct of incorrect.

c. Measurements of a large number of coins are taken. The algorithm uses this data, together with known class labels, to infer decision boundaries which it then uses to classify new coins.

5. Briefly explain the following types of learning method:

a. Classification

b. Regression

c. Semi-supervised

d. Transfer

6.

Dataset 1	
Class	Features
1	5
2	2
1	4
1	7
2	1

Dataset 2	
Class	Features
1	5.5
2	2.3
1	4
1	7
2	1.8

Dataset 3	
Class	Features
1	(5,4)
2	(2,9)
1	(4,3)
1	(7,4)
2	(1,5)

Dataset 4	
Class	Features
1	(5.3,4)
2	(2.3,9.1)
1	(4,3)
1	(7,4.1)
2	(1.8,5.5)

Identify which of the above datasets are:

a. univariate-continuous

b. multivariate-discrete

c. multivariate-continuous

d. univariate-discrete

7. A classifier is designed to determine if a feature vector is or is not in a certain class. The output produced by the classifier is 1 if the sample is predicted to be in the class, and 0 otherwise. The following table shows the feature vectors for the samples in the test set, along with the class labels predicted by the classifier and the true class labels of each sample.

Features	Predicted Class	True Class
(5.3,4)	1	1
(2.3,9.1)	0	1
(4,3)	1	0
(7,4.1)	1	1
(1.8,5.5)	0	0
(6,3.1)	1	1
(4.5,3.5)	0	1

Draw a confusion matrix for this data.

8. For the results given in the previous question calculate the following performance metrics:

a. the error-rate

b. the accuracy

c. the recall

d. the precision

e. the f_1-score

9. The following table shows exemplars from two classes. Sketch the feature space, and suggest a suitable location for a decision boundary that will minimise the number of mis-classifications. If the cost of erroneously choosing class 1 is higher than the cost of erroneously choosing class two, how will this effect the location of the decision boundary?

Class	Features
1	(3,4)
1	(4,7)
1	(7,5)
2	(2,5)
2	(3,9)

7CCSMPNN: Pattern Recognition, Neural Networks and Deep Learning

KING'S College LONDON

Tutorial - 2: Discriminant functions

Instructors: David Watson, Stefanos Leonardos *Term: Spring 2025 – 20-Jan*

Exercise 1. Consider a dichotomizer defined using the linear discriminant function $g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$ where $\mathbf{w} = \begin{pmatrix}2\\1\end{pmatrix}$ and $w_0 = -5$. Plot the decision boundary and determine the class of feature vectors: $\mathbf{x}_1 = \begin{pmatrix}1\\1\end{pmatrix}, \mathbf{x}_2 = \begin{pmatrix}2\\2\end{pmatrix}, \mathbf{x}_3 = \begin{pmatrix}3\\3\end{pmatrix}$.

Exercise 2. In augmented feature space, a dichotomizer is defined using the linear discriminant function $g(\mathbf{x}) = \mathbf{a}^T\mathbf{y}$ where $\mathbf{a}^T = (-5, 2, 1)$ and $y = \begin{pmatrix}1\\\mathbf{x}\end{pmatrix}$. Determine the class of feature vectors $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ (as given in Exercise 1).

Exercise 3. Consider a 3-dimensional feature space and the quadratic discriminant function

$$g(\mathbf{x}) = x_1^2 - x_3^2 + 2x_2x_3 + 4x_1x_2 + 3x_1 - 2x_2 + 2,$$

which defines two classes, ω_1, ω_2, such that $g(\mathbf{x}) > 0$ if $x \in \omega_1$ and $g(\mathbf{x}) \leq 0$ if $x \in \omega_2$. Determine the class of the feature vectors: $\mathbf{x}_1 = (1, 1, 1)^T, \mathbf{x}_2 = (-1, 0, 3)^T, \mathbf{x_3} = (-1, 0, 0)^T$.

Exercise 4. Consider a dichotomizer defined in a 2-dimensional feature space using a quadratic function

$$g(\mathbf{x}) = \mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{x}^T\mathbf{b} + c$$

where $\mathbf{b} = \begin{pmatrix}1\\2\end{pmatrix}$, and $c = -3$. Classify the feature vectors $\mathbf{x}_1 = \begin{pmatrix}0\\-1\end{pmatrix}$, and $\mathbf{x}_2 = \begin{pmatrix}1\\1\end{pmatrix}$, when:

i) $\mathbf{A} = \begin{pmatrix}2 & 1\\1 & 4\end{pmatrix}$, ii) $\mathbf{A} = \begin{pmatrix}-2 & 5\\5 & -8\end{pmatrix}$.

Exercise 5. In augmented feature space, a dichotomizer is defined using the linear discriminant function $g(\mathbf{x}) = \mathbf{a}^T\mathbf{y}$, where $\mathbf{a}^T = (-3, 1, 2, 2, 2, 4)$, and $y^T = (1, \mathbf{x}^T)$. Classify the feature vectors $\mathbf{x}_1^T = (0, -1, 0, 0, 1)$, and $\mathbf{x}_2^T = (1, 1, 1, 1, 1)$.

Exercise 6. A linear discriminant function, $g(\mathbf{x})$, is used to define a dichotomizer such that $\mathbf{x}$ is assigned to class 1 if $g(\mathbf{x}) > 0$, and $\mathbf{x}$ is assigned to class 2 otherwise. Use the *batch perceptron learning algorithm* with augmented notation and sample normalisation to find parameters for the linear discriminant function, when the data set is

$\mathbf{x}^T$	$(1,5)$	$(2,5)$	$(4,1)$	$(5,1)$
class	1	1	2	2

Assume initial values $\mathbf{a}^T = (w_0, \mathbf{w}^T) = (-25, 6, 3)$, and use a learning rate of 1.

Exercise 7. Repeat the previous exercise using the *sequential perceptron learning algorithm* with augmented notation and sample normalisation.

Exercise 8. Write pseudo-code for the *sequential perceptron learning algorithm.*

Exercise 9. Consider the linearly separable data set

$\mathbf{x}^T$	$(0,2)$	$(1,2)$	$(2,1)$	$(-3,1)$	$(-2,-1)$	$(-3,-2)$
class (ω)	1	1	1	-1	-1	-1

Apply the *sequential perceptron learning algorithm* to determine the parameters of a linear discriminant function that will correctly classify the data. Assume initial values $\mathbf{a}^T = (w_0, \mathbf{w}^T) = (1, 0, 0)$, and use a learning rate of 1.

Exercise 10. Repeat the previous exercise using the sample normalisation method of implementing the *sequential perceptron learning algorithm.*

Exercise 11. A data set consists of exemplars from three classes

$$\text{Class 1: } \begin{pmatrix}1\\1\end{pmatrix}, \begin{pmatrix}2\\0\end{pmatrix}, \quad \text{Class 2: } \begin{pmatrix}0\\2\end{pmatrix}, \begin{pmatrix}-1\\1\end{pmatrix}, \quad \text{Class 3: } \begin{pmatrix}-1\\-1\end{pmatrix}.$$

Use the *sequential multi-class perceptron learning algorithm* to find the parameters for three linear discriminant functions that will correctly classify this data. Assume that initial values for all parameters are zero, and use a learning rate of 1. Note: if more than one discriminant function produce the maximum output, then choose the function that represents the largest class label.

Exercise 12. Consider the linearly separable data set

$\mathbf{x}^T$	$(0,2)$	$(1,2)$	$(2,1)$	$(-3,1)$	$(-2,-1)$	$(-3,-2)$
class (ω)	1	1	1	-1	-1	-1

Use the pseudo-inverse (command `pinv` in MATLAB) to calculate the parameters of a linear discriminant function that can be used to classify this data. Use an arbitrary margin vector $\mathbf{b}^T = (1,1,1,1,1,1)$.

Exercise 13. Repeat the previous exercise using i) $\mathbf{b}^T = (2,2,2,1,1,1)$, and ii) $\mathbf{b}^T = (1,1,1,2,2,2)$.

Exercise 14. For the same data set as in the previous exercise, apply 12 iterations of the *sequential Widrow-Hoff learning algorithm.* Assume initial values $\mathbf{a}^T = (w_0, \mathbf{w}^T) = (1,0,0)$, and use a margin vector $\mathbf{b}^T = (1,1,1,1,1,1)$, and a learning rate of 0.1.

Exercise 15. The table shows the training data set for a simple classification problem

feature vector ($\mathbf{x}^T$)	$(0.15, 0.35)$	$(0.15, 0.28)$	$(0.12, 0.2)$	$(0.1, 0.32)$	$(0.06, 0.25)$
class (ω)	1	2	2	3	3

Apply the *k-nearest-neighbour* classifier with Euclidean distance and i) $k = 1$, ii) $k = 3$, to determine the class of a new feature vector $\mathbf{x} = (0.1, 0.25)$. How would these results be affected if the first dimension of the feature space was scaled by a factor of two?

Exercise 16. Consider the data set

feature vector ($\mathbf{x}^T$)	$(0,5)$	$(5,8)$	$(10,0)$	$(5,0)$	$(10,5)$
class (ω)	1	1	1	2	2

Assuming Euclidean distance, plot the decision boundaries that result from applying

i) a *nearest neighbour classifier*, i.e., a kNN classifier with $k = 1$,
ii) a *nearest mean classifier*, i.e., one in which a new feature vector, $\mathbf{x}$, is classified by assigning it to the same category as the nearest sample mean.

Neural Networks

1. Give brief definitions of the following terms:

- neuron
- action potential
- firing rate
- synapse
- an artificial neural network

2. A neuron has a transfer function which is a linear weighted sum of its inputs and an activation function that is the Heaviside function. If the weights are $\mathbf{w} = [0.1, -0.5, 0.4]$, and the threshold zero, what is the output of this neuron when the input is: $\mathbf{x}_1 = [0.1, -0.5, 0.4]^t$ and $\mathbf{x}_2 = [0.1, 0.5, 0.4]^t$?

3. A Linear Threshold Unit has one input, x_1, that can take binary values. Apply the sequential Delta learning rule so that the output of this neuron, y, is equal to $\bar{x_1}$ (or NOT(x_1)), *i.e.*, such that:

x_1	y
0	1
1	0

Assume initial values of $\theta = 1.5$ and $w_1 = 2$, and use a learning rate of 1.

4. Repeat the above question using the batch Delta learning rule.

5. A Linear Threshold Unit has two inputs, x_1 and x_2, that can take binary values. Apply the sequential Delta learning rule so that the output of this neuron, y, is equal to x_1AND x_2, *i.e.*, such that:

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Assume initial values of $\theta = -0.5$, $w_1 = 1$ and $w_2 = 1$, and use a learning rate of 1.

6. Consider the following linearly separable data set.

$\mathbf{x}^t$	$class$
$(0, 2)$	1
$(1, 2)$	1
$(2, 1)$	1
$(-3, 1)$	0
$(-2, -1)$	0
$(-3, -2)$	0

Apply the Sequential Delta Learning Algorithm to find the parameters of a linear threshold neuron that will correctly classify this data. Assume initial values of $\theta = -1$, $w_1 = 0$ and $w_2 = 0$, and a learning rate of 1.

7. A negative feedback network has three inputs and two output neurons, that are connected with weights $\mathbf{W} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$. Determine the activation of the output neurons after 5 iterations when the input is $\mathbf{x} = (1, 1, 0)^T$, as-

suming that the output neurons are updated using parameter $\alpha = 0.25$, and the activations of the output neurons are initialised to be all zero.

8. Repeat the previous question using a value of $\alpha = 0.5$.

9. A more stable method of calculating the activations in a negative feedback network is to use the following update rules:

$$\mathbf{e} = \mathbf{x} \oslash \left[\mathbf{W}^T \mathbf{y}\right]_{\epsilon_2}$$

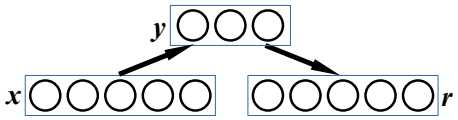
$$\mathbf{y} \leftarrow [\mathbf{y}]_{\epsilon_1} \odot \tilde{\mathbf{W}}\mathbf{e}$$

where $[v]_\epsilon = \texttt{max}(\epsilon, v)$; ϵ_1 and ϵ_2 are parameters; $\tilde{\mathbf{W}}$ is equal to $\mathbf{W}$ but with each row normalised to sum to one; and $\oslash$ and $\odot$ indicate element-wise division and multiplication respectively.

This is called Regulatory Feedback or Divisive Input Modulation.

Determine the activation of the output neurons after 5 iterations when the input is $\mathbf{x} = (1, 1, 0)^T$, and $\mathbf{W} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$, assuming that $\epsilon_1 = \epsilon_2 = 0.01$, and the activations of the output neurons are initialised to be all zero.

10. The figure below shows an autoencoder neural network.



Draw a diagram of a de-noising autoencoder and briefly explain how a de-noising autoencoder is trained.

7CCSMPNN: Pattern Recognition, Neural Networks and Deep Learning

KING'S College LONDON

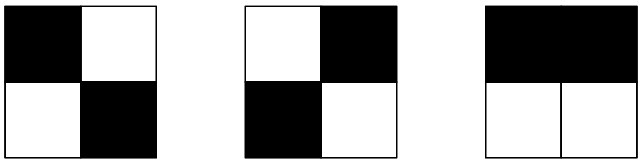
Tutorial - 4: MLPs and backpropagation

Instructors: David Watson, Stefanos Leonardos *Term: Spring 2025 – 03-Feb*

Exercise 1. Draw the diagram of a 2-input-3-output feedforward fully-connected neural network having 2 hidden layers with 4 and 5 units (nodes), respectively, and bias terms in the input and all hidden layers. How many connection weights does it have in total?

Exercise 2. Derive the equation for the number of weights in a multi-layer feedforward fully-connected neural network including biases in the input and all hidden layers in terms of: i) d = number of inputs, ii) L = number of hidden layers, iii) n_h = number of units (nodes) in the h-th hidden layer, $h = 1, 2, \ldots, L$, and iv) z = number of outputs.

Exercise 3. Figure 1 shows three patterns of a classification problem. A feedforward fully-connected neural network



(a) Pattern 1. (b) Pattern 2. (c) Pattern 3.

Figure 1: Three patterns.

is employed to classify the patterns. How many inputs and outputs should be used for the neural network? List the input patterns and target outputs in a table. Note: any pattern different from the above is classified as "otherwise".

Exercise 4. Figure 2 shows a 4-3-2 feedforward fully-connected neural network which is used to classify the patterns shown in Figure 1. The assignment of input $\mathbf{x}^T = (x_1, x_2, x_3, x_4)$ is shown in Figure 3 where 0 represents an unfilled grid and 1 represents a filled grid. The activation functions in the input, hidden and output layers are linear, hyperbolic tangent sigmoid and logarithmic sigmoid, respectively. Given the weight and bias matrices

$$\mathbf{W}_{ji} = \begin{pmatrix} -0.7057 & 1.9061 & 2.6605 & -1.1359 \\ 0.4900 & 1.9324 & -0.4269 & -5.1570 \\ 0.9438 & -5.4160 & -0.3431 & -0.2931 \end{pmatrix}, \quad \mathbf{W}_{j0} = \begin{pmatrix} 4.8432 \\ 0.3973 \\ 2.1761 \end{pmatrix},$$

$$\mathbf{W}_{kj} = \begin{pmatrix} -1.1444 & 0.3115 & -9.9812 \\ 0.0106 & 11.5477 & 2.6479 \end{pmatrix}, \quad \mathbf{W}_{k0} = \begin{pmatrix} 2.5230 \\ 2.6463 \end{pmatrix}.$$

determine the output patterns representing the three classes.

Exercise 5. Figure 4 shows a 3-layer, partially-connected feedforward neural network. The input units use linear activation functions and all hidden and output units use hyperbolic tangent sigmoid activation functions. The neural network is trained using stochastic backpropagation on the cost function $J = \frac{1}{2}\|t - z_1\|^2$, where z_1 is the network output corresponding to input pattern $\mathbf{x}$ selected from the training set and t is its corresponding target output.

i) Considering $\mathbf{x}^T = (0.1, 0.9)$ determine y_1, y_2 and z_1.
ii) Derive the backpropagation update rules for m_{10} and w_{21} and calculate the updated values for m_{10} and w_{21} for the next iteration using $\mathbf{x}^T = (0.1, 0.9)$, the weights in Figure 4, $\mathbf{t} = 0.5$ and learning rate $\eta = 0.25$.
iii) Assuming that the backpropagation algorithm only updates the weights m_{10} and w_{21}, show that the updated weights are valid.

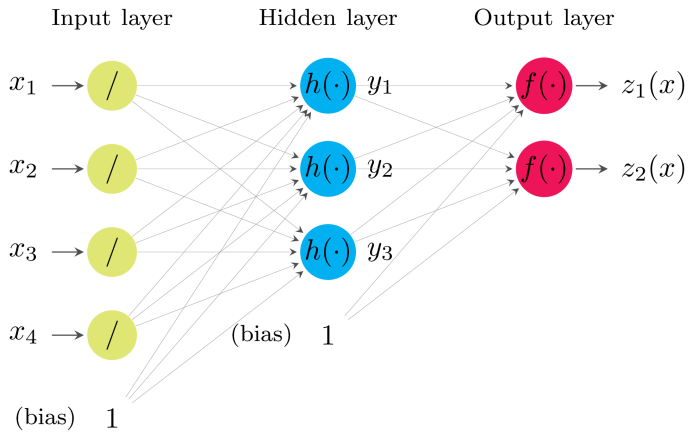


Figure 2: Diagram of the 4-3-2 feedforward ected neural network.

x_1	x_2
x_4	x_3

Figure 3: Assignment of input **x**.

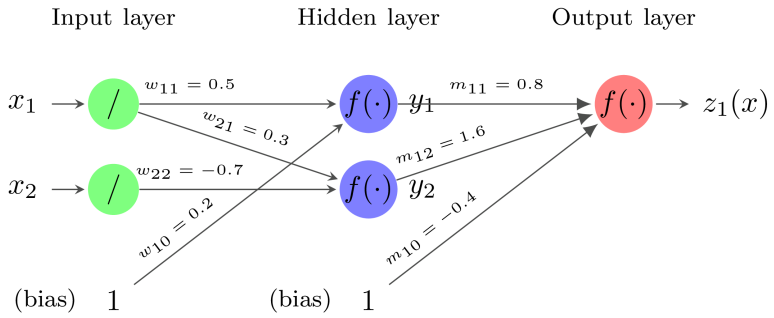


Figure 4: Diagram of a 3-layer partially-connected feedforward neural network.

Exercise 6. Consider a XOR problem given in Table 1. Design a radial basis function (RBF) neural network with bias to solve the problem. Choose input pattern 1 and pattern 4 as the centres and use Gaussian function with $\sigma_j = \frac{\rho_{\max}}{\sqrt{2n_h}}$ in the hidden units and linear function in the output units.

p	x_1	x_2	t
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	0

Table 1: XOR problem.

p	x_1	x_2
1	0.5	-0.1
2	-0.2	1.2
3	0.8	0.3
4	1.8	0.6

Table 2: Input patterns for the XOR classifier.

i) Draw a diagram showing the structure of an RBF neural network solving the XOR problem.
ii) Compute the outputs at the hidden units for all input patterns and list them in a table. Show that the patterns at the hidden units are linearly separable.
iii) Determine the output weights using the least square method. Compute the outputs at the output unit(s) and list them in a table. Show that the XOR problem is solved.
iv) Determine the class for the input patterns in Table 2.

Exercise 7. Consider a classifier which is able to separate two classes, such that x belongs to class 1 if $g(x) \geq x/2$ and x belongs to class 2 if $g(x) < x/2$, where $g(x)$ is an unknown function for which we wo only know the input-output mappings shown in Table 3

.

p	1	2	3	4	5	6	7	8	9	10	11	12
x	0.05	0.20	0.25	0.30	0.40	0.43	0.48	0.60	0.70	0.80	0.90	0.95
z	0.0863	0.2662	0.2362	0.1687	0.1260	0.1756	0.3290	0.6694	0.4573	0.3320	0.4063	0.3535

Table 3: Input-output mappings for function $g(x)$.

i) Sketch the function $g(x)$ and determine the possible class of the following points $x = 0.1, 0.35, 0.55, 0.75, 0.9$.
ii) If an RBF neural network is employed to implement the classifier, determine its number of inputs and outputs.
iii) An RBF neural network with three hidden units is employed to implement the classifier. All hidden units use Gaussian function with $\sigma = 0.1$ and all output units use linear function. Given that $c_1 = 0.2, c_2 = 0.6$, and $c_3 = 0.9$, use the least squares method to determine the output weights of the RBF neural network and provide its equation.
iv) Given clusters $S_1 = \{x_1, x_2, x_3\}, S_2 = \{x_4, x_5\}, S_3 = \{x_6, x_7, x_8, x_9\}$ and $S_4 = \{x_{10}, x_{11}, x_{12}\}$, find the centres for an RBF neural network implementing the classifier. All hidden units use Gaussian function with $\sigma = 2\rho_{\text{avg}}$ and all output use linear function. Based on the found centres, use the least squares method to find the output weights of the RBF neural network. Draw the diagram of the designed network and show its weights.

Exercise 8. A nonlinear classifier with multiple inputs and a single output is implemented by feedforward neural network. All inputs are in the range of -1 and 1. Propose a method to replace the feedforward neural network with an RBF neural network.

7CCSMPNN: Pattern Recognition, Neural Networks and Deep Learning

KING'S College LONDON

Tutorial - 5: Deep discriminative neural networks

Instructors: David Watson, Stefanos Leonardos

Term: Spring 2025 – 10-Feb

Exercise 1. Give brief definitions of the following terms

i) feature map
ii) sub-sampling
iii) adversarial example
iv) data augmentation
v) transfer learning
vi) dropout
vii) regularization

Exercise 2. Define what is meant by, and explain the causes of, the "vanishing gradient problem". Briefly describe four methods that can be used to reduce its effects.

Exercise 3. Mathematically define the following activation functions

i) ReLU
ii) LReLU
iii) PReLU

Exercise 4. The following array shows the output produced by a mask in a convolutional layer of a convolutional neural network (CNN)

$$\text{net}_j = \begin{bmatrix} 1 & 0.5 & 0.2 \\ -1 & -0.5 & -0.2 \\ 0.1 & -0.1 & 0 \end{bmatrix}.$$

Calculate the values produced by the application of the following activation functions

i) ReLU,
ii) LReLU when $a = 0.1$,
iii) tanh,
iv) heaviside function with threshold 0.1 for each neuron (define $H(0) = 0.5$).

Exercise 5. The following arrays show the output produced by a convolutional layer to all 4 samples in a batch

$$X_1 = \begin{bmatrix} 1 & 0.5 & 0.2 \\ -1 & -0.5 & -0.2 \\ 0.1 & -0.1 & 0 \end{bmatrix}, \quad X_2 = \begin{bmatrix} 1 & -1 & 0.1 \\ 0.5 & -0.5 & -0.1 \\ 0.2 & -0.2 & 0 \end{bmatrix}, \quad X_3 = \begin{bmatrix} 0.5 & -0.5 & -0.1 \\ 0 & -0.4 & 0 \\ 0.5 & 0.5 & 0.2 \end{bmatrix}, \quad X_4 = \begin{bmatrix} 0.2 & 1 & -0.2 \\ -1 & -0.6 & -0.1 \\ 0.1 & 0 & 0.1 \end{bmatrix}.$$

Calculate the corresponding outputs produced after the application of batch normalisation, assuming the following parameter values $\beta = 0, \gamma = 1$, and $\epsilon = 0.1$ which are the same for all neurons.

Exercise 6. The following arrays show the feature maps that provide the input to a convolutional layer of a CNN

$$X_1 = \begin{bmatrix} 0.2 & 1 & 0 \\ -1 & 0 & -0.1 \\ 0.1 & 0 & 0.1 \end{bmatrix}, \quad X_2 = \begin{bmatrix} 1 & 0.5 & 0.2 \\ -1 & -0.5 & -0.2 \\ 0.1 & -0.1 & 0 \end{bmatrix}.$$

If a mask, H, has two channels defined as

$$H_1 = \begin{bmatrix} 1 & -0.1 \\ 1 & -0.1 \end{bmatrix}, \quad H_2 = \begin{bmatrix} 0.5 & 0.5 \\ -0.5 & -0.5 \end{bmatrix},$$

calculate the output produced by mask H when using

i) padding = 0, stride = 1,
ii) padding = 1, stride = 1,
iii) padding = 1, stride = 2,
iv) padding = 0, stride = 1, dilation = 2.

Exercise 7. The following arrays show the feature maps that provide the input to a convolutional layer of a CNN

$$X_1 = \begin{bmatrix} 0.2 & 1 & 0 \\ -1 & 0 & -0.1 \\ 0.1 & 0 & 0.1 \end{bmatrix}, \quad X_2 = \begin{bmatrix} 1 & 0.5 & 0.2 \\ -1 & -0.5 & -0.2 \\ 0.1 & -0.1 & 0 \end{bmatrix}, \quad X_2 = \begin{bmatrix} 0.5 & -0.5 & -0.1 \\ 0 & -0.4 & 0 \\ 0.5 & 0.5 & 0.2 \end{bmatrix}.$$

Calculate the output produced by 1×1 convolution when the 3 channels of the 1×1 mask are: $[1, -1, 0.5]$.

Exercise 8. The following array shows the input to a pooling layer of a CNN: $X_1 = \begin{bmatrix} 0.2 & 1 & 0 & 0.4 \\ -1 & 0 & -0.1 & -0.1 \\ 0.1 & 0 & -1 & -0.5 \\ 0.4 & -0.7 & -0.5 & 1 \end{bmatrix}$.

Calculate the output produced by the pooling layer when using

i) average pooling with a pooling region of 2×2 and stride $= 2$,
ii) max pooling with a pooling region of 2×2 and stride $= 2$,
iii) max pooling with a pooling region of 3×3 and stride $= 1$.

Exercise 9. The input to a convolutional layer of a CNN consists of 6 feature maps each of which has a height of 11 and width of 15, i.e., input is $11 \times 15 \times 6$. What size is the output produced by a single mask with 6 channels,a width of 3 and a height of 3, i.e., $3 \times 3 \times 6$ when using a stride of 2 and padding of 0.

Exercise 10. A CNN processes an image of size $200 \times 200 \times 3$ using the following sequence of layers

1) convolution with 40 masks of size $5 \times 5 \times 3$ with stride $= 1$, and padding $= 0$,
2) pooling with 2×2 pooling regions and stride $= 2$,
3) convolution with 80 masks of size 4×4 withs stride $= 2$, and padding $= 1$,
4) 1×1 convolution with 20 masks.

What is the size of the output once it has been flattened?

Exercise 11. The following images show exemplars from two datasets.

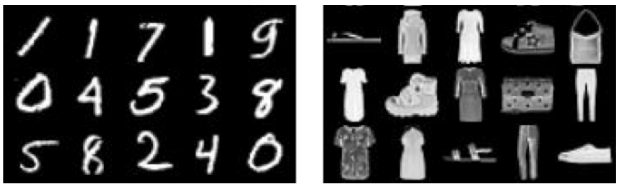


Figure 1: MNIST (left) and FashionMNIST (right).

Each dataset is to be expanded using data augmentation. Which of the following transformations are appropriate:

i) rescaling
ii) horizontal flip
iii) rotation
iv) cropping.

7CCSMPNN: Pattern Recognition, Neural Networks and Deep Learning

KING'S College LONDON

Tutorial - 6: Generative adversarial networks (GANs)

Instructors: David Watson, Stefanos Leonardos *Term: Spring 2025 – 24-Feb*

Exercise 1. Vanishing gradient is an issue when training Generative Adversarial Networks (GANs). In the literature, when training the generator, it is recommended to consider the alternative cost function: $\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}\left[-\log\left(D(G(\mathbf{z}))\right)\right]$.

i) What is the advantage of using this alternative cost function over the original one, i.e., $\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}\left[\log\left(1-D(G(\mathbf{z}))\right)\right]$?
ii) Write the pseudo-code for training a GAN with this alternative cost function.

Exercise 2. The training of GANs can be formulated as an optimisation problem as shown below

$$\min_G \max_D V(D,G) = \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[\log\left(D(\mathbf{x})\right)] + \mathbb{E}_{\mathbf{z}\sim p_z(\mathbf{z})}[\log\left(1-D(G(\mathbf{z}))\right)].$$

i) Given a generator, G, with output distribution $p_{\text{gen}}(\mathbf{x})$, find the value of the optimal discriminator $D^*(\mathbf{x})$, for each $\mathbf{x}\in\mathbf{X}$. Assume that $p_{\text{data}}(\mathbf{x}), p_{\text{gen}}(\mathbf{x}) > 0$ for all $\mathbf{x}\in\mathbf{X}$.
ii) What is the output distribution, $g^*_{\text{gen}}(\mathbf{x})$, of an optimal generator G^*. Find $D^*(\mathbf{x})$, for $\mathbf{x}\in\mathbf{X}$, against G^*.
iii) Determine the value of the loss function, $V(D^*,G^*)$, when both generator and discriminator are optimal.

Exercise 3. When training a GAN, we consider a dataset consisting of real, $\mathbf{X}_{\text{real}} = \{\mathbf{x}_1, \mathbf{x}_2\}$, and generated samples, $\mathbf{X}_{\text{fake}} = \{\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2\}$, where: $\mathbf{x}_1 = \begin{pmatrix}1\\2\end{pmatrix}, \mathbf{x}_2 = \begin{pmatrix}3\\4\end{pmatrix}, \tilde{\mathbf{x}}_1 = \begin{pmatrix}5\\6\end{pmatrix}, \tilde{\mathbf{x}}_2 = \begin{pmatrix}7\\8\end{pmatrix}$. The discriminator is given by

$$D(\mathbf{x}) = \frac{1}{1+e^{-(\theta_{d_1}x_1 - \theta_{d_2}x_2 - 2)}},$$

where $\theta_{d_1} = 0.1, \theta_{d_2} = 0.2$ are the parameters of the discriminator, and $\mathbf{x} = (x_1, x_2)^T$. Assume that each sample from the (real and fake) dataset has equal probability to be selected.

i) Given the datasets $\mathbf{X}_{\text{real}}$, and $\mathbf{X}_{\text{fake}}$ compute

$$V(D,G) = \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[\log\left(D(\mathbf{x})\right)] + \mathbb{E}_{\mathbf{z}\sim p_z(\mathbf{z})}[\log\left(1-D(G(\mathbf{z}))\right)].$$

ii) Assuming that all real and fake samples are selected in the (mini)batch and that $k=1$ for GAN training, compute

$$\nabla_{\theta_d}\frac{1}{m}\sum_{i=1}^{m}\left[\log\left(D(\mathbf{x}_i)\right) + \log\left(1-D(G(\mathbf{z}_i))\right)\right],$$

and determine the updated θ_{d_1} and θ_{d_2} for the next iteration using learning rate $\eta = 0.02$.

7CCSMPNN: Pattern Recognition, Neural Networks and Deep Learning

KING'S College LONDON

Tutorial - 7: Feature extraction

Instructors: David Watson, Stefanos Leonardos *Term: Spring 2025 – 03-Mar*

Exercise 1. Give brief definitions of the following terms

i) feature engineering
ii) feature selection
iii) feature extraction
iv) dimensionality reduction
v) deep learning

Exercise 2. List five methods that can be used to perform feature extraction.

Exercise 3. Write pseudo-code for the Karhunen-Loève Transform (KLT) method for performing Principal Component Analysis (PCA).

Exercise 4. Use the KLT method to project the following 3-dimensional data set onto the first 2 principal components: $\mathbf{x_1} = (1, 2, 1)^T, \mathbf{x_2} = (2, 3, 1)^T, \mathbf{x_3} = (3, 5, 1)^T, \mathbf{x_4} = (2, 2, 1)^T$. Hint: The MATLAB command `eig` can be used to find the eigenvectors and eigenvalues.

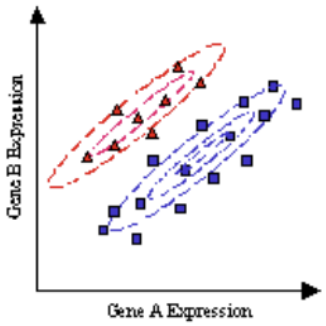
Exercise 5. What is the proportion of variance explained by the first 2 principal components in the previous exercise?

Exercise 6. Use the KLT method to project the 2-dimensional data set, $\begin{pmatrix}0\\1\end{pmatrix}, \begin{pmatrix}3\\5\end{pmatrix}, \begin{pmatrix}5\\4\end{pmatrix}, \begin{pmatrix}5\\6\end{pmatrix}, \begin{pmatrix}8\\7\end{pmatrix}, \begin{pmatrix}9\\7\end{pmatrix}$, onto the first principal component. (Hint: the MATLAB command `eig` can be used to find the eigenvectors and eigenvalues).

Exercise 7. Apply two epochs of a batch version of Oja's learning rule to the same data used in the previous exercise. Use a learning rate of 0.01, and an initial weight vector of $(-1, 0)$.

Exercise 8. The graph on the right shows a 2-dimensional data set in which exemplars come from two classes. Exemplars from the two classes are plotted using triangular and square markers, respectively. Draw the approximate direction of

i) the first principal component of this data,
ii) the axis onto which the data will be projected using Linear Discriminant Analysis (LDA).



Exercise 9. Briefly describe the optimisation method performed by Fisher's LDA to find a projection of the original data onto a subspace.

Exercise 10. For the data shown below use Fisher's LDA method to determine which of the following projection

feature vector $\mathbf{x}^T$	$(1,2)$	$(2,1)$	$(3,3)$	$(6,5)$	$(7,8)$
class	1	1	1	2	2

weights is more effective at performing LDA: i) $\mathbf{w}^T = (-1, 5)$, ii) $\mathbf{w}^T = (2, -3)$.

Exercise 11. An Extreme Learning Machine consists of a hidden layer with six neurons, and an output layer with one neuron. The weights to the hidden neurons have been assigned the following random values

$$\mathbf{V} = \begin{pmatrix} -0.62 & 0.44 & -0.91 \\ -0.81 & -0.09 & 0.02 \\ 0.74 & -0.91 & -0.60 \\ -0.82 & -0.92 & 0.71 \\ -.26 & 0.68 & 0.15 \\ 0.80 & -0.94 & -0.83 \end{pmatrix}.$$

The weights to the output neuron ar: $\mathbf{w}^T = (0, 0, 0, -1, 0, 0, 2)$. All weights are defined using augmented vector notation. Hidden neurons are *linear threshold* units, while the output neuron is linear. Calculate the response of the output neuron to each of the following input vectors: $\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}$.

Exercise 12. Given a dictionary, $\mathbf{V}^T = \begin{pmatrix} 0.4 & 0.55 & 0.5 & -0.1 & -0.5 & 0.9 & 0.5 & 0.45 \\ -0.6 & -0.45 & -0.5 & 0.9 & -0.5 & 0.1 & 0.5 & 0.55 \end{pmatrix}$, what is the best sparse code for the signal, $\mathbf{x} = \begin{pmatrix} -0.05 \\ -0.95 \end{pmatrix}$, out of the following two alternatives

i) $\mathbf{y_1}^T = (1, 0, 0, 0, 1, 0, 0, 0)$,
ii) $\mathbf{y_2}^T = (0, 0, 1, 0, 0, 0, -1, 0)$.

Assume that sparsity is measured as the count of elements that are non-zero.

Exercise 13. Repeat the previous exercise when the two alternatives are

i) $\mathbf{y_1}^T = (1, 0, 0, 0, 1, 0, 0, 0)$,
ii) $\mathbf{y_2}^T = (0, 0, 0, -1, 0, 0, 0, 0)$.

7CCSMPNN: Pattern Recognition, Neural Networks and Deep Learning

KING'S College LONDON

Tutorial - 8: Support vector machines

Instructors: David Watson, Stefanos Leonardos *Term: Spring 2025 – 10-Mar*

Exercise 1. A Support Vector Machine (SVM) classifier is employed to classify the following points

$$\text{class 1: } \mathbf{x_1} = \begin{pmatrix}1\\1\end{pmatrix}, \mathbf{x_2} = \begin{pmatrix}1\\-1\end{pmatrix}, \quad \text{class 2: } \mathbf{x_3} = \begin{pmatrix}-1\\1\end{pmatrix}, \mathbf{x_4} = \begin{pmatrix}-1\\-1\end{pmatrix}.$$

i) Determine if the two classes are linearly separable.
ii) Design an SVM classifier to correctly classify all given points.
iii) Identify the support vectors.

Exercise 2. An SVM classifier is employed to classify the following points

$$\text{class 1: } \mathbf{x_1} = \begin{pmatrix}3\\1\end{pmatrix}, \mathbf{x_2} = \begin{pmatrix}3\\-1\end{pmatrix}, \mathbf{x_3} = \begin{pmatrix}7\\1\end{pmatrix}, \mathbf{x_4} = \begin{pmatrix}8\\0\end{pmatrix}$$
$$\text{class 2: } \mathbf{x_5} = \begin{pmatrix}1\\0\end{pmatrix}, \mathbf{x_6} = \begin{pmatrix}0\\1\end{pmatrix}, \mathbf{x_7} = \begin{pmatrix}-1\\0\end{pmatrix}, \mathbf{x_8} = \begin{pmatrix}-2\\0\end{pmatrix}.$$

i) Determine if the dataset is linearly separable.
ii) Identify the support vectors by inspection.
iii) Design an SVM classifier to correctly classify all given points. What is the margin?
iv) Classify the point $\mathbf{x}^T = (2.5, 0.5)$ using the designed SVM classifier.

Exercise 3. An SVM classifier is employed to classify the following points

$$\text{class 1: } \mathbf{x_1} = \begin{pmatrix}2\\2\end{pmatrix}, \mathbf{x_2} = \begin{pmatrix}2\\-2\end{pmatrix}, \mathbf{x_3} = \begin{pmatrix}-2\\-2\end{pmatrix}, \mathbf{x_4} = \begin{pmatrix}-2\\2\end{pmatrix}$$
$$\text{class 2: } \mathbf{x_5} = \begin{pmatrix}1\\1\end{pmatrix}, \mathbf{x_6} = \begin{pmatrix}1\\-1\end{pmatrix}, \mathbf{x_7} = \begin{pmatrix}-1\\-1\end{pmatrix}, \mathbf{x_8} = \begin{pmatrix}-1\\1\end{pmatrix}.$$

i) Determine if the data set is linearly separable.
ii) Apply the feature mapping function $\Phi(\mathbf{x}) = \begin{pmatrix}4 - \frac{x_2}{2} + |x_1 - x_2|\\4 - \frac{x_1}{2} + |x_1 - x_2|\end{pmatrix}$, if $\|\mathbf{x}\| > 2$, and $\Phi(\mathbf{x}) = \begin{pmatrix}x_1 - 2\\x_2 - 3\end{pmatrix}$, otherwise, to the data set where $\mathbf{x}^T = (x_1, x_2)$. Determine if the data set is linearly separable in the new feature space.
iii) Identify the support vectors by inspection.
iv) Design a nonlinear SVM classifier to correctly classify all given points.

Exercise 4. Compare and discuss the advantages and disadvantages of the four approaches to using SVMs for multi-class problems, namely one against one, one against all, binary decision tree and binary coded approach.

Exercise 5. Given the class assignments in Table 1 for a multi-class SVM classifier using the *binary coded approach*, list the binary codes for all classes in a table. What is the main disadvantage?

SVM Number	Class assignment $(+1 \mid -1)$
SVM 1	12345 | 6789
SVM 2	12567 | 3489
SVM 3	14578 | 2369
SVM 4	1368 | 24579

Table 1: Class assignments

Department of Engineering/Informatics, King's College London
Pattern Recognition, Neural Networks and Deep Learning (7CCSMPNN)
Tutorial 9

Q1. Consider the following dataset: $\{(\mathbf{x} = (1, 0), y = +1), (\mathbf{x} = (-1, 0), y = +1), (\mathbf{x} = (0, 1), y = -1), (\mathbf{x} = (0, -1), y = -1)\}$ where $\mathbf{x} = (x_1, x_2)$ is the input feature and y is the output class. 8 weak classifies are designed and given as follows:

$$h_1(\mathbf{x}) = \begin{cases} +1, & \text{if } x_1 > -0.5 \\ -1, & \text{otherwise} \end{cases} \quad ; \quad h_2(\mathbf{x}) = \begin{cases} -1, & \text{if } x_1 > -0.5 \\ +1, & \text{otherwise} \end{cases}$$

$$h_3(\mathbf{x}) = \begin{cases} +1, & \text{if } x_1 > 0.5 \\ -1, & \text{otherwise} \end{cases} \quad ; \quad h_4(\mathbf{x}) = \begin{cases} -1, & \text{if } x_1 > 0.5 \\ +1, & \text{otherwise} \end{cases}$$

$$h_5(\mathbf{x}) = \begin{cases} +1, & \text{if } x_2 > -0.5 \\ -1, & \text{otherwise} \end{cases} \quad ; \quad h_6(\mathbf{x}) = \begin{cases} -1, & \text{if } x_2 > -0.5 \\ +1, & \text{otherwise} \end{cases}$$

$$h_7(\mathbf{x}) = \begin{cases} +1, & \text{if } x_2 > 0.5 \\ -1, & \text{otherwise} \end{cases} \quad ; \quad h_8(\mathbf{x}) = \begin{cases} -1, & \text{if } x_2 > 0.5 \\ +1, & \text{otherwise} \end{cases}$$

a. Plot the dataset in 2-dimensional space. Is the dataset linearly separable?

b. Determine the classification error for all weak classifiers.

c. Find the minimal classifier using AdaBoost algorithm, which can reach zero training error.

Q2. Using a Bagging algorithm, 8 classifiers shown in Q1 are obtained.

a. Determine the training error (using the training dataset in Q1) given by the Bagging classifier.

b. Instead of using 8 classifiers, which classifiers should be used to form the bagging classifier. Determine the training error using the training dataset in Q1.

Q3. Figure 1 shows two binary decision trees.

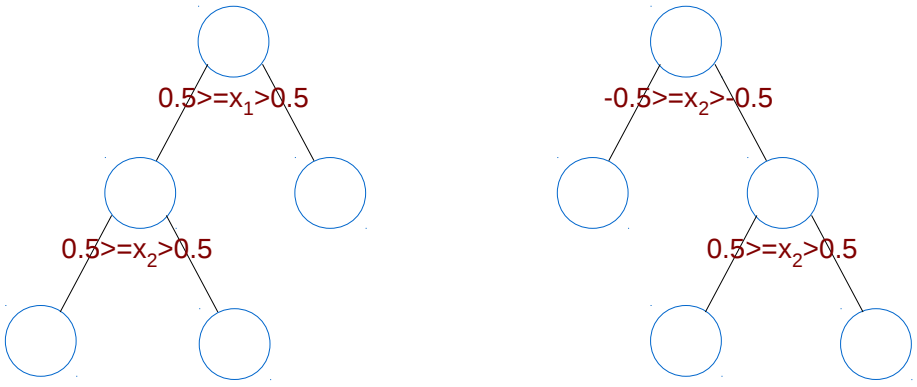


Figure 1: Two binary decision trees.

a. Using the training data given in Q1, calculate the $P(class = -1|\mathbf{x})$ for each leaf node.

b. Hence, determine the class for a new sample $\mathbf{x} = (0, 0)$ predicted by each decision tree.

c. If the two binary decision trees shown in Figure 1 were learnt using a random forest algorithm. What would be the class for a new sample $\mathbf{x} = (0, 0)$ predicted by this random forest?

Department of Engineering/Informatics, King's College London
Pattern Recognition, Neural Networks and Deep Learning (7CCSMPNN)
Tutorial 10

Q1. Apply the K-means algorithm to the following dataset, $\mathbf{S}$, to find 2 natural groupings (K = 2) using the Euclidean distance criterion. Start with initial values of cluster centres of $\mathbf{m}_1$ and $\mathbf{m}_2$ accordingly.

$\mathbf{S} = \begin{bmatrix} -1 & 1 & 0 & 4 & 3 & 5 \\ 3 & 4 & 5 & -1 & 0 & 1 \end{bmatrix}$, $\mathbf{m}_1 = \begin{bmatrix} -1 \\ 3 \end{bmatrix}$, $\mathbf{m}_2 = \begin{bmatrix} 5 \\ 1 \end{bmatrix}$

Plot the results and show a suitable linear discriminant function for the two clusters and give its weight vector (assuming $g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} = 0$)

Q2. Consider the dataset $\mathbf{X} = \begin{bmatrix} 4 & 0 & 2 & -2 \\ 2 & -2 & 4 & 0 \\ 2 & 2 & 2 & 2 \end{bmatrix}$, use principal component analysis (PCA) to:

a. project the data onto 2D plane,

b. project the data onto 1D plane.

c. The dataset consists of samples from two classes. By observation, find the cluster centres based on the transformed 2D and 1D samples. Classify the new data $\mathbf{x} = \begin{bmatrix} 3 \\ -2 \\ 5 \end{bmatrix}$ (after removing the mean) using both sets of clusters centres.

Q3. Competitive learning algorithm (without normalisation) is employed to find 3 cluster centres for the dataset $\mathbf{S}$ in Q1. The initial cluster centres are chosen as $\mathbf{x}_1/2$, $\mathbf{x}_3/2$ and $\mathbf{x}_5/2$, and $\eta = 0.1$. The algorithm is run for 5 iterations and the samples are chosen in the order of $\mathbf{x}_3$, $\mathbf{x}_1$, $\mathbf{x}_1$, $\mathbf{x}_5$, $\mathbf{x}_6$ where the left most data is chosen first.

a. Find the cluster centres for each iteration.

b. Plot the samples $\mathbf{S}$ and cluster centres in a figure. Classify the samples.

c. Classify the new data $\mathbf{x} = \begin{bmatrix} 0 \\ -2 \end{bmatrix}$.

Q4. Basic leader follower algorithm (without normalisation), with $\theta = 3$ and $\eta = 0.5$, is employed to find the cluster centres of the dataset $\mathbf{S}$ in Q1. The algorithm is run for 5 iterations and the samples are chosen in the order of $\mathbf{x}_3$, $\mathbf{x}_1$, $\mathbf{x}_1$, $\mathbf{x}_5$, $\mathbf{x}_6$ where the left most data is chosen first.

a. Find the cluster centres for each iteration.

b. Classify the samples in $\mathbf{S}$.

c. Classify the new data $\mathbf{x} = \begin{bmatrix} 0 \\ -2 \end{bmatrix}$.

Q5. Apply the Fuzzy K-means algorithm, with K = 2, to the dataset: $\mathbf{S} = \begin{bmatrix} -1 & 1 & 0 & 4 & 3 & 5 \\ 3 & 4 & 5 & -1 & 0 & 1 \end{bmatrix}$. Set parameter $b = 2$, terminate when both coordinates of both cluster centres change by less than 0.5 from one iteration to the next, and start with initial membership values equal to: $\mu = \begin{bmatrix} 1 & 0.5 & 0.5 & 0.5 & 0.5 & 0 \\ 0 & 0.5 & 0.5 & 0.5 & 0.5 & 1 \end{bmatrix}$.

Q6. Given the following dataset $\mathbf{x}_1^T = [-1, 3], \mathbf{x}_2^T = [1, 2], \mathbf{x}_3^T = [0, 1], \mathbf{x}_4^T = [4, 0], \mathbf{x}_5^T = [5, 4], \mathbf{x}_6^T = [3, 2]$ perform hierarchical agglomerative clustering until three clusters are formed ($c = 3$). Use the Euclidean distance as the similarity metric. To determine the distance between clusters use single-link clustering (the minimum distance between samples in the two clusters).