Homework 04

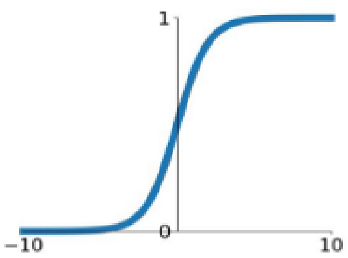
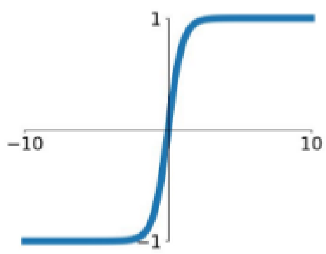
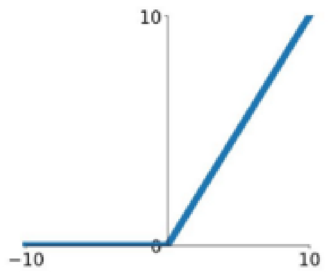
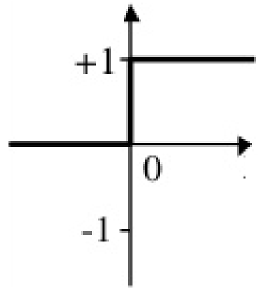
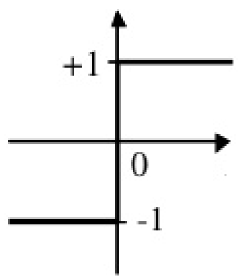
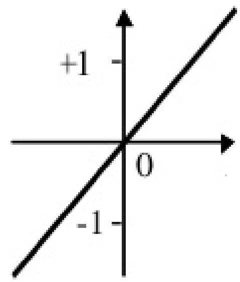
Submission Notices:

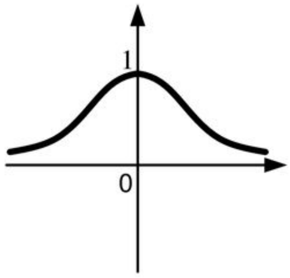
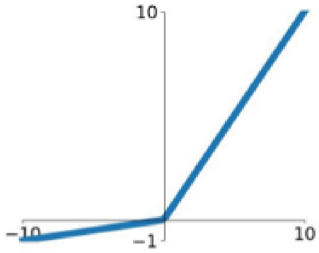
- *Conduct your homework by filling answers into the placeholders in this file (in Microsoft Word format). Questions are shown in black color, instructions/hints are shown in italics and blue color, and your content should use any color that is different from those.*
- *After completing your homework, prepare the file for submission by exporting the Word file (filled with answers) to a PDF file, whose filename follows the following format,*

  <StudentID-1>_<StudentID-2>_HW01.pdf (Student IDs are sorted in ascending order)
  E.g., ***2112001_2112002_HW02.pdf***

  and then submit the file to Moodle directly WITHOUT any kinds of compression (.zip, .rar, .tar, etc.).
- *Note that you will get zero credit for any careless mistake, including, but not limited to, the following things.*
  1. *Wrong file/filename format, e.g., not a pdf file, use "-" instead of "_" for separators, etc.*
  2. *Disorder format of problems and answers*
  3. *Conducted not in English*
  4. *Cheating, i.e., copying other students' works or letting other students copy your work.*

Problem 1. (2pts) Identify each of the following activation functions.

#	2-D representation	#	2-D representation
a		b	
c		d	
e		f	

g		h	

Please fill your answer in the table below

#	a	b	c	d
Function name	Sigmoid	Hyperbolic tangent	ReLU	Step
#	e	f	g	h
Function name	Sign	Linear	Gaussian	Leaky ReLU

Problem 2. (1pt) Present two objective metrics that can be used to evaluate the attributes for a node on the decision tree. For each metric, you need to present the formula, identify its domain (i.e., range of values), and explain for every term in the formula.

Please fill your answer in the table below

Metric name	Formula	Explanation
		Specific to each work
		Specific to each work

Problem 3. (2pts) You are given the following tables, which represent the outcomes of some functions. The functions take two values x and y and output the outcomes of the operations. Please identify **at least two models** for each of the functions that are perfectly represent the functions for some choice of parameters. Justify your answer. Note: there are no constraints on the architecture (e.g, the number of neurons, activation function, or the best splitting criterion), and the depth of decision tree is 0-index.

a) (1pt) $\boldsymbol{f(x, y) = x \oplus y}$

$\boldsymbol{x}$	$\boldsymbol{y}$	$\boldsymbol{x \oplus y}$
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**

☐ A neural network with no hidden layer

☐ **A neural network with a single hidden layer -> Choose this**

☐ A decision tree of depth one

☐ **A decision tree of depth two -> Choose this**

Explanation:

A perceptron (no hidden layer neural network) $g(x, y) = \alpha(\beta(x, y))$ where $\beta(x, y) = ax + by + c$ and α is a (monotonic) activation function cannot correctly classify the XOR function. Note $\beta(1, 1) = a + b + c$ and $\beta(0, 0) = c$. WLOG, suppose $a + b + c \geq c$ (we can always flip the signs of all coefficients and reflect g to make this true). Suppose for contradiction that (0, 0) and (1, 1) are both correctly classified, then $\alpha(a + b + c) = 0 = \alpha(a)$. Since activation function α must be monotonic, it follows that all points $\alpha(x)$ for $x \in [c, a + b + c]$ must also be 0. Note that one of $\beta(1, 0) = a + c$ or $\beta(0, 1) = b + c$ must lie between $[c, a + b + c]$. Thus at least one of (1, 0) or (0, 1) is misclassified, giving a contradiction.

By contrast, a neural network with a hidden layer has sufficient capacity to represent XOR. Indeed, the universal approximation theorem shows that a single hidden layer network can represent any function given enough neurons. For XOR, a small network suffices $g(x, y) = sgn\left(B \cdot sgn\left(A \cdot \begin{pmatrix} x \\ y \end{pmatrix}\right)\right)$, where $sgn(x)$ is -1 when $x < 0$, 0 when $x = 0$, $+1$ when $x > 0$ and $A = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 1 \end{pmatrix}$

A decision tree of depth one can only split on a single variable. Since XOR depends on the values of both variables, no tree of depth one can represent it. A decision tree of depth two can represent any two-variable boolean function, including XOR

b) (1pt) $\boldsymbol{f(x, y) = \neg(x \vee y)}$

$\boldsymbol{x}$	$\boldsymbol{y}$	$\boldsymbol{\neg(x \vee y)}$
0	0	**1**
0	1	**0**

1	0	**0**
1	1	**0**

- ☐ **A neural network with no hidden layer -> Choose this**
- ☐ **A neural network with a single hidden layer -> Choose this**
- ☐ A decision tree of depth one
- ☐ **A decision tree of depth two -> Choose this**

Explanation:

A single layer neural network $g(x, y) = sgn(-x - y + 1)$ classifies $f(x, y)$ correctly. A two layer neural network can also classify it (we can always make the second layer an identity). A decision tree of depth one cannot represent f since it depends on two variables. A decision tree of depth two can represent it (and any other two-variable boolean function)

Problem 4. (2pts) Consider the following training dataset, in which **Transportation** is the target attribute. Show calculations to choose an attribute for the **root node** of the ID3 decision tree

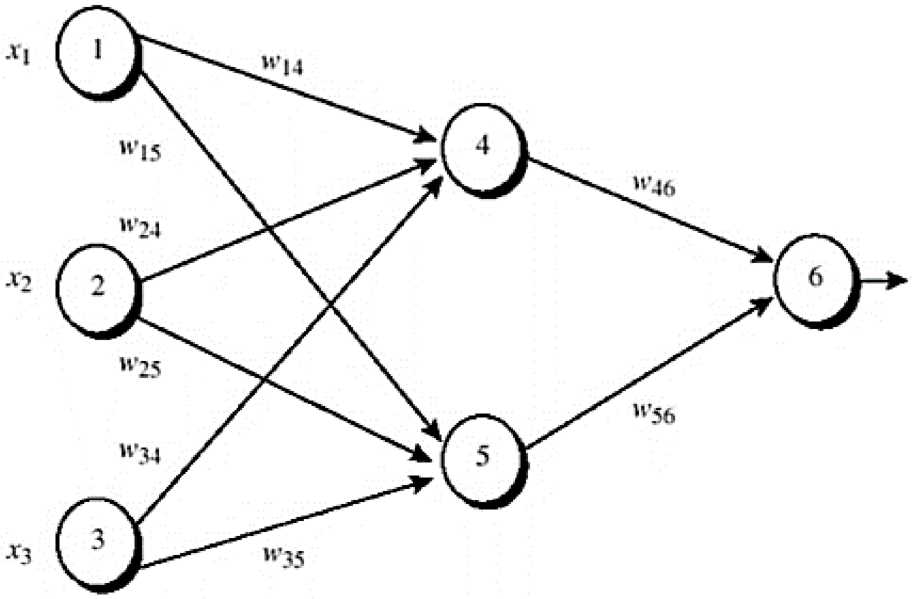
Gender	Car Ownership	Travel Cost	Income Level	Transportation
Male	0	Cheap	Low	Bus
Male	1	Cheap	Medium	Bus
Female	1	Cheap	Medium	Train
Female	0	Cheap	Low	Bus
Male	1	Cheap	Medium	Bus
Male	0	Standard	Medium	Train
Female	1	Standard	Medium	Train
Female	1	Expensive	High	Car
Male	2	Expensive	Medium	Car
Female	2	Expensive	High	Car

Please fill your answer in the table below

		Counts			Metric values		
	Attribute values	Bus	Car	Train	H	AE	IG
Whole	**10**	**4**	**3**	**3**	**1.571**		
Gender (0.5pt)	Female	1	2	2	1.522	1.446	0.125
	Male	3	1	1	1.371		
Car Ownership (0.5pt)	0	2	0	1	0.918	1.036	0.535
	1	2	1	2	1.522		
	2	0	2	0	0		
Travel Cost (0.5pt)	Cheap	4	0	1	0.722	0.361	1.21
	Expensive	0	3	0	0		

	Standard	0	0	2	0		
Income Level (0.5pt)	Low	2	0	0	0	0.875	0.696
	Medium	2	1	3	1.459		
	High	0	0	2	0		

Problem 5. (3pts) Consider the following neuron network, which includes 3 input neurons, 2 hidden neurons and 1 output neurons.



Initial input, weight and bias values are

x_1	x_2	x_3	w_{14}	w_{15}	w_{24}	w_{25}	w_{34}	w_{35}	w_{46}	w_{56}	θ_4	θ_5	θ_6
1	0	1	0.2	–0.3	0.4	0.1	–0.5	0.2	–0.3	–0.2	–0.4	0.2	0.1

The expected output value is 1. The learning rate is 0.9.
Knowing that the actual output at some neuron *j* is calculated as follows.

$$y_j(p) = \text{sigmoid}\left[\sum_{i=1}^{n} x_i(p) \times w_{ij}(p) + \theta_j\right]$$

where *n* is the number of inputs of neuron *j*, w_{ij} is the corresponding link from a neuron *i* in the previous layer to neuron *j*, and θ_j is the bias at neuron *j*.

Present all calculations required to perform the backpropagation once (i.e., one forward pass and one backward pass) on the given neural network in the following cases

a) Ignore all biases *(precision to 3 decimal places).*

(0.25pt) Ignore all biases – Forward

Neuron	4	5	6
Output	0.426	0.475	0.445

(1pt) Ignore all biases – Backward

Weight	w_{46}	w_{56}	w_{14}	w_{15}	w_{24}	w_{25}	w_{34}	w_{35}
Value	-0.247	-0.141	0.192	-0.306	0.4	0.1	-0.508	0.194

b) Consider all biases such that each bias is treated as a neuron and thus it will be also updated *(precision to 3 decimal places).*

(0.25pt) Consider all biases – Forward

Neuron	4	5	6
Output	0.332	0.525	0.474

(1.5pt) Consider all biases – Backward

Weight	w_{46}	w_{56}	w_{14}	w_{15}	w_{24}	w_{25}	w_{34}	w_{35}	θ_4	θ_5	θ_6
Value	-0.261	-0.138	0.193	-0.305	0.4	0.1	-0.507	0.195	-0.407	0.195	0.218