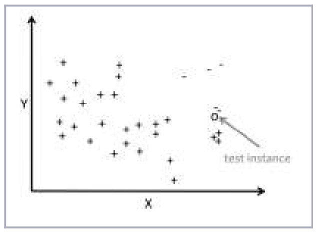
Name Date

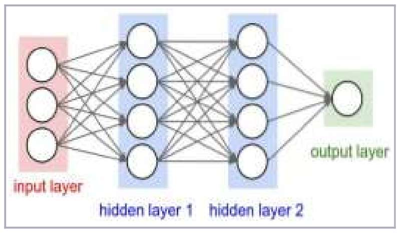
KNN and Neural Network

1. setting K to a large value seems a good idea .we get more votes! for the above data set we set k value to 11 .Choose the most appropriate comment on this decision *points: 0.5*



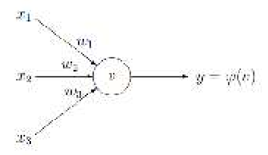
- ◯ A good idea since ,the data set is not too big ,considering more neighbour would help
- ◯ Not a good idea since class of a data point is determined more by its most closest neighbour
- ◯ not a good decision,since there are 5 negative instances in the training set.Therefore K >10 would have a majority of positive
- ◯ None of the above

2. consider the neural network,during the training of the network ,how many trainable parameters are we dealing with. *points: 1*



- ◯ 41
- ◯ 32
- ◯ 28
- ◯ None of the above

3. consider the perceptron ,let w1,w2,w3 be 2,-4,1 respectively and activation function ϕ(x) is a defined as 0 if x<0, 2x if x>0 and x%2==0 else x what is the output for input as (5 and 6) represented as 3 bit binary number , x1,x2,x3 representing each bit assume bias to be 0. y1,y2 is output when input 5 and 6 repectively *points: 1*



- ○ y1=3 and y2=-2
- ○ y1=3 and y2=0
- ○ y1=-3 and y2=0
- ○ None of the above

4. You want to map every possible image of size 64 × 64 to a binary category (cat or non-cat). Each image has 3 channels and each pixel in each channel can take an integer value between (and including) 0 and 255. How many bits do you need to represent this mapping? *points: 1*

- ○ 255^(64x64)
- ○ 255^(64)
- ○ 255^(64x64x3)
- ○ none of the above

5. The mapping from previous question clearly can not be stored in memory. Instead, you will build a classifier to do this mapping. You use a single hidden layer of size 100 for this task. Each weight in the W[1] and W[2] matrices can be represented in memory using a float of size 64 bits. How many bits do you need to store your two layer neural network (you may ignore the biases b [1] and b [2])? *points: 1*

- ○ 64 × ((256 × 3 × 100) + (64 × 64 × 1))
- ○ 64 × ((64 × 64 × 3 × 100) + (100 × 1))
- ○ 64 × ((2563 × 64 × 64 × 100) + (100 × 64))
- ○ 64 × (256 × 3 × 64 × 64 × 100)

6. Suppose you have a 3-dimensional input x = (x1, x2, x3) = (2, 2, 1) fully connected to 1 neuron with activation function gi . The forward propagation can be written: as in fig 1 After training this network, the values of the weights and bias are: w = (w1, w2, w3) = (0.5, −0.2, 0) and b = 0.1. You try 4 different activation functions (g1, g2, g3, g4) which respectively output the values (a1, a2, a3, a4) = (0.67, 0.70, 1.0, 0.70). What is a valid guess for the activation functions g1, g2, g3, g4? Note: indicator function is defined as in fig 2 *points: 1*

$$z = \left(\sum_{k=1}^{3} w_k x_k\right) + b$$
$$a_i = \mathbf{g}_i(z)$$

$$I(x)_{x\geq 0} = \begin{cases} 0 & if \quad x < 0 \\ 1 & if \quad x \geq 0 \end{cases}$$

- ○ sigmoid, tanh, indicator function, linear
- ○ linear, indicator function, sigmoid, ReLU
- ○ sigmoid, linear, indicator function, leaky ReLU
- ○ ReLU, linear, indicator function, sigmoid

7 . The task is to design an autonmous car. Your goal is to detect road signs (stop sign, pedestrian crossing sign, construction ahead sign) and traffic signals (red and green lights) in images. The goal is to recognize which of these objects appear in each image. You plan to use a deep neural network with ReLU units in the hidden layers. For the output layer, a ___________ activation would be a good choice for the output layer because this is a multi-task learning problem. *points: 0.5*

- ○ sigmoid
- ○ softmax
- ○ tanh
- ○ relu

8. Do you think that the following is a good way to calculate the weights in weighted KNN *points: 1*

$$F(x, y) = \sum_{i=1}^{n} w_i f_i$$

where n is the number of points used to interpolate, f_i are the prescribed function values at the points (e.g., the dataset values), and w_i are the weight functions assigned to each point. The classical form of the weight function is:

$$w_i = \frac{h_i^{-p}}{\sum_{j=1}^{n} h_j^{-p}}$$

○ true ○ false