Republic of Korea Air Force Academy

Digital Image Processing - Problem Sheet

Lecturer - Sungkwon On

April 2023

1. K-Nearest Neighbor

(a) Explain the K-Nearest Neighbor(KNN) algorithm with an aid of diagrams.
(K-Nearest Neighbor(KNN) 알고리즘에 대하여 도표와 함께 설명하시오.)

(b) How does the hyperparameter, ”K”, affects the performance of the KNN algorithm?
(Hyperparameter ”K”가 KNN 알고리즘 성능에 어떤 영향을 끼치는가?)

(c) Why is the KNN’s performance poor for image classification tasks?
(KNN의 성능이 이미지 구별에서는 왜 안좋은가?)

2. Data Sets

(a) When given a data set to train a Deep Learning model, we usually separate the data sets to mainly three folds, Training Set, Validation Set, and Testing Set. Explain why we do this.
(Data set이 딥러닝 모델 훈련을 위해 주어졌을 때, 우리는 3가지의 용도로 나눈다, Training Set, Validation Set, and Testing Set. 이렇게 나누는 이유를 설명하시오.)

(b) What is the purpose of the Validation Data Set?
(Validation Data Set의 목적은 무엇인가?)

(c) What is Cross-Validation?

3. Linear Regression Consider the figure 0.1. A Linear Regression method will be used to estimate a model that fits the blue point best. We will be using a polynomial curve;

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + ... + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

(그림 0.1를 참고하시오. 다음과 같은 Linear Regression(선형 회귀) 모델이 그림에 있는 파란색 포인트를 맞추는데 사용될 것이다.)

(a) By stating the name of the phenomenon, explain why $M = 0$ and $M = 1$ are not suitable choices for M and by intuition, state a suitable value of M for this model.
($M = 0$ and $M = 1$의 값들이 왜 부적절한지 설명하고, 해당 현상의 명칭을 제시하시오.)

(b) There are 20 data points on the figure, when we choose M as a value that is smaller but close to 20, state how the trained model would look like and state the name of this phenomenon.
(해당 그림에는 20개의 ponit들이 있다. M의 값을 20보다는 작지만 근접하게 정한다면 훈련된 모델은 어떤 형상을 보이는가? 해당 현상의 명칭을 제시하시오.)

(c) For the case of $M = 20$, state two methods to prevent overfitting.
($M = 20$일 때, overfitting(과적합)을 방지하는 두가지 방법을 제시하시오.)

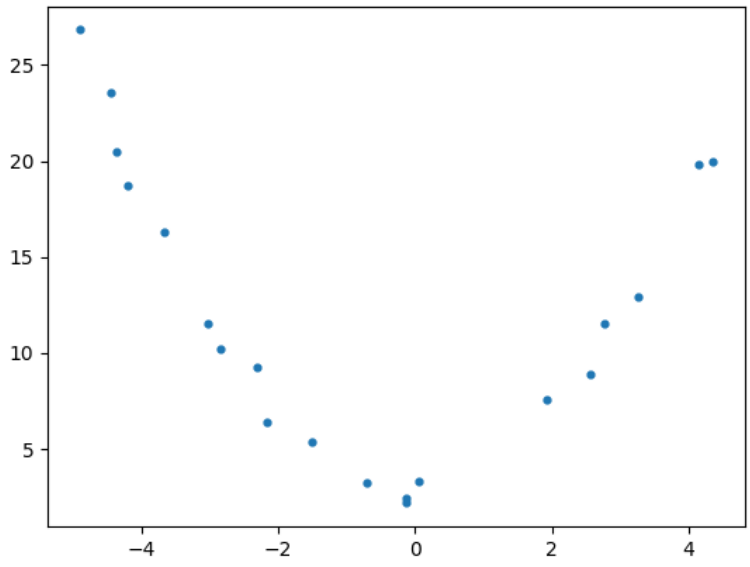

Figure 0.1

4. Perceptron

(a) Design a perceptron that represents AND, OR, and NAND logic gates.
(AND, OR, and NAND 논리 게이트를 나타내는 Perceptron을 디자인 하시오.)

(b) Explain why it is impossible to represent an XOR gate with a single-layer perceptron.
(왜 XOR gate를 Perceptron으로는 만들지 못하는지 설명하시오.)

(c) Use AND, OR and NAND gate once to represent XOR logic gate.
(AND, OR, NAND 논리 게이트를 각각 한번씩 사용하여 XOR 게이트를 나타내는 논리 회로를 그리시오.)

(d) Using your answers in (a) and (c), or otherwise, design a multi-layer perceptron that represents an XOR logic gate.
(위의 답들을 활용하여, 혹은 다른 방법으로, XOR 게이트를 나타내는 multi-layer perceptron을 디자인 하시오.)

(e) Why should activation functions must be non-linear functions?
(왜 activation functions(활성화 함수)들은 비선형 함수여야만 하는가?)

.

5. Forward Propagation

(a) Assume bias exists in all layers, express the size(shape) of the weight matrix between a layer with n nodes and m nodes in a neural network.
(모든 레이어에 bias(편향)이 있다고 가정하고, n개의 노드와 m개의 노드를 가진 레이어들 사이의 weight matrix(가중치 행렬)의 크기(형상)을 구하시오.)

(b) Consider figure 0.2. A human-designed feature extractor extracts a feature vector of an image and each of its elements contains information about one of 16 equally divided parts of the image, as shown in figure 0.2. Our task is to correctly classify an input image to which KPOP star the image is showing (assume the image is showing one person only). We have 100 selected celebrity labels and the output must also contain ”Not a person” and ”Not in selected 100 labels” labels. In a Neural Network model we are designing, we will be using 3 hidden layers of 200 hidden nodes each followed by 1 hidden layer of 100 nodes. We are using a softmax calculation for probability-like outputs. Roughly sketch the network model and calculate how many weights parameters are there to learn. Assume bias exists in all layers.
(그림 0.2을 참고하시오. 사람이 만든 feature extractor(특징 추출기)가 feature vector(특징 백터)를 추출한다, 해당 백터의 요소로는 이미지의 각 16등분에 대한 정보가 담겨있다. 우리의 task는 입력으로 들어온 이미지를 정확한 KPOP 스타로 분류하는 것이다(각 이미지 별로 사람은 한명만 있다고 가정한다). 우리는 선별된 100명의 연애인에 대한 label이 있으며, 출력의 옵션으로는 ”사람이 아님” 과 ”100명의 선별 label에 없음” label이 추가로 있어야 한다. 우리가 사용 할 인공신경망 모델은 200개의 노드를 가진 3개의 hidden layer(은닉층)에 이여서 100개의 노드를 가진 하나의 hidden layer로 구성되어 있다. 확률 형태의 출력을 위해서 우리는 softmax 계산을 출력 직전에 사용한다. 해당 신경망의 다이어그램을 대략적으로 그리시오. 모든 레이어에 bias가 있다고 가정하고, 모든 weight parameter의 갯수를 구하시오.)

Figure 0.2

6. Gradient Descent

(a) Compare and state differences between sum of squares and cross entropy errors.
(Sum of Squares Error랑 Cross Entropy Error를 비교하여 둘의 차이를 서술하시오.)

(b) Explain the gradient descent algorithm and state why the learning rate should not be too high nor too low.
(Gradient Descent알고리즘에 대해서 설명하고 왜 learning rate(학습률)이 너무 크거나 작으면 안 되는지 설명하시오.)

(c) State the difference between Stochastic Gradient Descent(SGD) and classic Gradient Descent(GD).
(Stochastic Gradient Descent(SGD - 확률적 경사하강법)과 일반적인 Gradient Descent(GD - 경사하강법)의 차이를 설명하시오.)

7. Backward Propagation

(a) Consider the neural network in figure 0.3. ReLU is used for the first two layer intervals and Logistic Sigmoid is used at the end as activation functions. Bias exists in all layers. Sum of Square Errors is used as a loss function. Compute $\frac{\partial L}{\partial w^3_{1y}}$, $\frac{\partial L}{\partial w^2_{21}}$ and $\frac{\partial L}{\partial w^1_{12}}$. Where w^l_{ij} indicates the weight parameter in lth layer interval, between i and j nodes.
(그림 0.3의 신경망을 고려하시오. 첫 두 레이어 인터벌에서는 ReLU가, 마지막에는 Logistic Sigmoid의 activation function이 사용된다. 모든 레이어에 bias가 존재한다. Sum of Square Errors가 loss function으로 사용된다. $\frac{\partial L}{\partial w^3_{1y}}$, $\frac{\partial L}{\partial w^2_{21}}$ and $\frac{\partial L}{\partial w^1_{12}}$를 계산하시오.) .

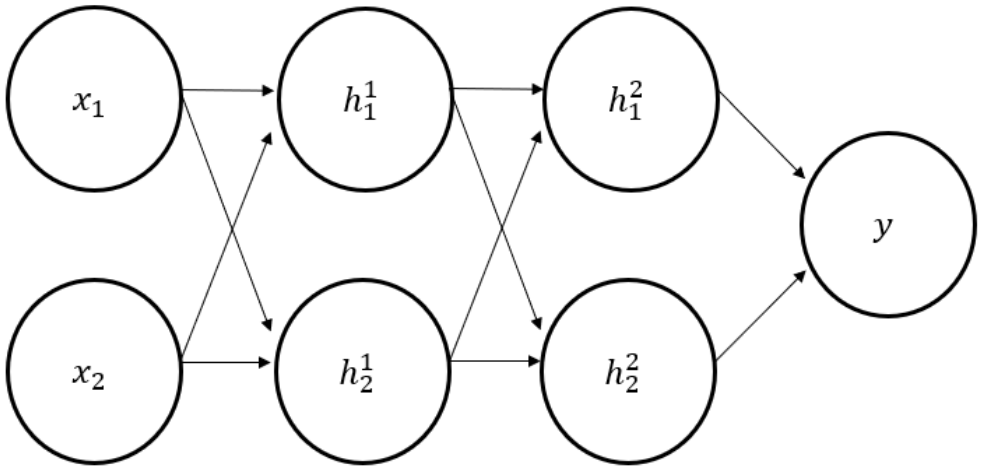

Figure 0.3

(b) Consider the neural network in figure 0.4. ReLU is used for all activation functions. Bias exists in all layers. Sum of Square Errors is used as a loss function. Consider a mini-batch of a data set; (1, 3), (2, 5), (5, 11), (7, 15), all expressed in (input, label) form. Assume all weights are initialized to 0.5 and the learning rate, η is 0.1. Compute the gradient matrix, $\frac{\partial L}{\partial \mathbf{w}}$ and perform the SGD algorithm for two iterations using the same mini-batch training data.
(그림 0.4의 신경망을 고려하시오. 모든 activation function은 ReLU로 통인된다. 모든 레이어에 bias가 존재한다. Sum of Square Errors가 loss function으로 사용된다. Data set의 한 mini-batch를 고려하라; (1, 3), (2, 5), (5, 11), (7, 15), 모두 (input, label)의 형태이다. 모든 weight의 초깃값은 0.5이고 learning rate는 0.1이다. 가중치 행렬, $\frac{\partial L}{\partial \mathbf{w}}$를 계산하시오. 그리고 SGD 알고리즘을 동일단 mini-batch로 두번 반복하시오.)

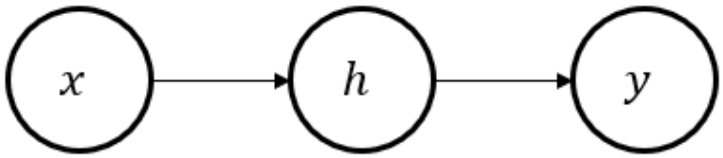

Figure 0.4