COMP4434 Big Data Analytics

Assignment 2

PolyU, Hong Kong

Instructor: HUANG Xiao
Logistics: You should submit your solutions through Learn@PolyU (Blackboard). The deadline is Sunday 9 March, 11:55 PM. I will no accept submission from any other channels except Blackboard. These are the best exercises that could help you be well prepared for exams. Thus, please work independently.

Problem 1 (2 points)
Please answer the following questions briefly.

(a) In the classification task, why do we prefer logistic loss over mean-squared error?

(b) How does a standard Support Vector Machine (SVM) differ from an SVM with a soft margin, and in what way do slack variables (ξ_i) facilitate the handling of non-linearly separable data?

(c) If your multi-layer perceptron model has an overfitting issue, what are the strategies that you could use to handle the issue?

(d) How does backpropagation use the gradient descent algorithm to update the weights in a neural network?

Problem 2 (5 points)
Load dataset in *problem2data.txt* by using “numpy.loadtxt()” in Python, which contains two synthetic classes separated by a non-linear function, with additive noise. The last column contains “0” or “1”, indicating the corresponding classes. You are asked to investigate the extent to which polynomial functions can be used to build a logistic regression classifier.

(a) Build a logistic regression classifier by using a polynomial function of order 1 (e.g., $\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$). Apply 5-fold-cross-validation to generate training and test sets. Use the training set to train the model. Compute its five F1 scores on test set (one F1 score for each fold). Compute its five F1 scores on training set. You will need “f1_score” from “sklearn.metrics”.

(b) Repeat part (a) for polynomials of order 2 to 10. You will need “PolynomialFeatures” from “sklearn.preprocessing”.

(c) Repeat parts (a-b) 20 times, and estimate the average F1 score for each polynomial order (average across its 5×20 runs, both for training and test sets). Generate a plot that shows the F1 scores versus the polynomial order.

(d) Discuss how the F1 scores of the model changes as a function of the polynomial order.

(e) Use Jupyter Notebook to perform implementation. You are required to submit your ".ipynb" file. You could use "LogisticRegression" from "sklearn.linear_model". Do **not** add any regularizations.

Problem 3 (2 points)
Consider a set of data points represented by the coordinates (x, y). The objective is to apply the k-means algorithm to identify two distinct clusters within the data. Use $(0.5, 1)$ as the initial centroid for the first cluster and $(3, 4)$ as the initial centroid for the second cluster. Perform a single iteration of the k-means algorithm with $k = 2$, utilizing cosine distance as the distance metric. Recall that cosine distance is defined as *1 - cosine similarity*. The data points are listed as follows.

Data#	x	y
1	1	0.5
2	0.5	2
3	3	1
4	2	1.5

(1) Determine the cluster assignments for each data point after one iteration.
(2) Compute the new centroids for each cluster in Euclidean space after completing the first iteration.