National University of Singapore
School of Computing
CS3244: Machine Learning
Tutorial 1

The Learning Paradigms, and k-NN

1. **Learning Paradigms.** Describe different instances of learning problems for the following scenarios. For each scenario, describe a *supervised*, *unsupervised* and *reinforcement* learning problem. For one of the problems, formalize the given components in the learning problem: *input*, *output*, *data*. You need not describe the *hypothesis* nor the *target function* (to think about: why?).

   (a) In NUS (or other university) student domain. Students have problems, many of them, and you are the target audience. Describe problems that you encounter on a daily, weekly or semesterly basis.

   (b) Transshipment Logistics. One of Singapore's mainstay sources of income for decades[1] has been transshipment and the logistics associated with this. Hypothesize problems that occur in this scenario.

2. **k-NN**

   (a) Suppose you are given the following data (as shown in Table 1) where x and y are the two input variables and *Origin* is the dependent variable.

x	y	*Origin*
-1	1	-
0	1	+
0	2	-
1	-1	-
1	0	+
1	2	+
2	2	-
2	3	+

Table 1: The dataset for kNN.

Figure 1 is a scatter plot which shows the above data in 2D space.

Suppose that you want to predict the class of new data point at $x = 1$ and $y = 1$ using Euclidian distance with 3-NN. Which class does this data point belong to? Does the classification change if we change the algorithm to 7-NN?

[1] Accordingly to Wikipedia, as of 2016, Singapore remains the world's busiest transshipment port.

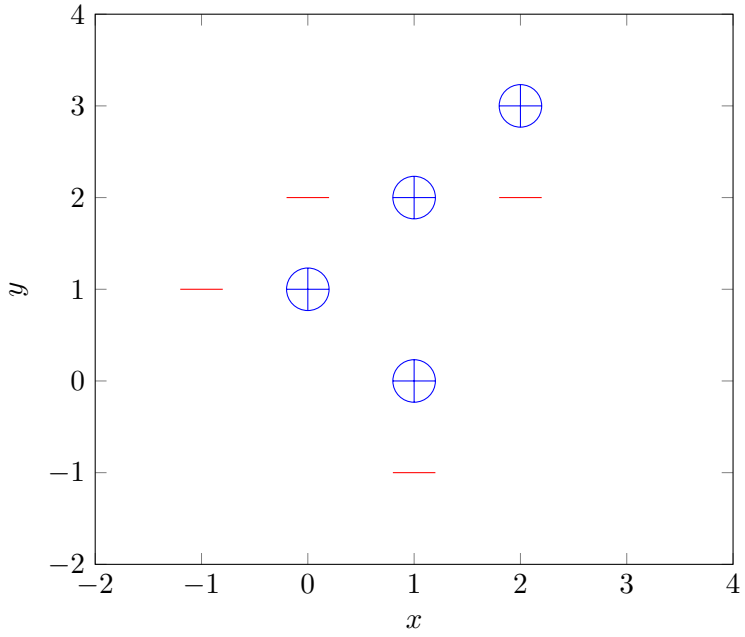

Figure 1: Data points of k-NN dataset on 2D space.

(b) Suppose you are given the following images (Figure 2). Your task is to compare the values of k in k-NN in each image where k_l, k_c and k_r are the left, center and right subfigures below, respectively.

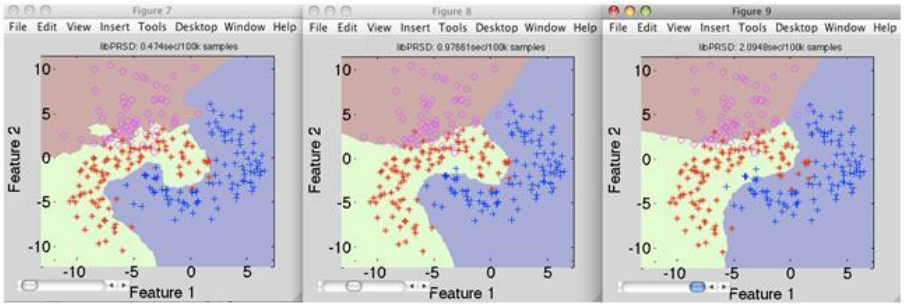

Figure 2: kNN runs for $k = 1, 2, 3$. Which is which?

Which one of k_l, k_c and k_r is the largest k and which one is the smallest?

(c) Suppose that you have trained a kNN model and now you want to perform prediction on test data. Before executing the prediction (*a.k.a.* inference) task, you want to calculate the time that kNN will take for predicting the class for test data. Let's denote the time to calculate the distance between 2 observations as t. What would the time taken by 1-NN be if there are m (some very large number) observations in the training data? What about for 2-NN or 3-NN? (We only consider the time used for calculating distances.)

3. **Analysing k-NN Inference** Alice and Bob have proposed 2 ways of doing k-NN inference. Both algorithms are explained below. There are n number of training samples and the time taken to calculate the distance between two samples is $O(d)$.

   - **Algorithm by Alice**
     (a) Initialize $S[i] = 0$ for all the training samples. Here $i \in \{1, 2, 3, ..., n\}$
     (b) For each training sample, compute $D[i]$. Here $D[i]$ denotes the distance between training sample i and the new observation.
     (c) Iterate k number of times through all the training samples to do the following procedure in every iteration. Find the smallest $D[i]$ with the condition $S[i] = 0$. After the full scan through all the samples, mark $S[min] = 1$. Here min is the location where $D[min]$ is small and $S[min] = 0$.
     (d) Return k samples with indices where $S[i] = 1$.
   - **Algorithm by Bob**
     (a) Initialize $S[i] = 0$ for all the training samples. Here $i \in \{1, 2, 3, ..., n\}$
     (b) Iterate k number of times through all the training samples to do the following procedure in every iteration. Find the distance between each sample and the new observation. This steps are only done for the locations is' with $S[i] = 0$. The minimum distance location will be marked with $S[min] = 1$.
     (c) Return k samples with indices where $S[i] = 1$.

   (a) Verify whether the algorithms of Alice and Bob are correct or not? If they are correct, give the running time for single inference in terms of n, d, k. Which is the best algorithm with respect to the running time?

   (b) Propose a way to improve the best algorithm in part (a). What is the running time of the new algorithm?

4. **k-NN** Suppose you have to do a classification problem to predict whether it will rain on an area based on the available dataset using the k-NN algorithm. The input variables are the humidity which varies between $50 - 90\%$, and the average temperature which ranges between $25 - 35$ degrees Celcius. Do you think applying k-NN **directly** will yield a good prediction result? If not, what improvement will you propose?

5. (Optional) **The Netflix Prize** The Netflix Prize was a competition held by Netflix, to improve its algorithm for recommending movies to its users. This was formalized by having a system predict a customer's numeric rating of a target movie. The winning entry was one that used collaborative filtering, which is a method that is based on the assumption that people who agreed in the past will agree in the future. It looked at the historical ratings that users had given movies, but not the features of the movie and user (e.g. genre, year, director, actor, etc.). However, it was never adopted. Can you think of several reasons why?