Boston University
Department of Electrical and Computer Engineering

***ENG EC 414 Introduction to Machine Learning***

# HW 7

© 2015 – 2020 Prakash Ishwar
© 2020 Francesco Orabona

**Issued:** Fri 23 Oct 2020 **Due:** 10:00am Fri 30 Oct 2020 in Gradescope (non-code) + Blackboard

**Important:** Before you proceed, please read the documents pertaining to *Homework formatting and submission guidelines* in the Homeworks section of Blackboard. **In particular, for computer assignments you are prohibited from using any online code or built-in MATLAB functions except as indicated in the problem or skeleton code (when provided).**

**Important:** To obtain full grade, please clearly motivate all your answers.

**Note:** Problem difficulty = number of coffee cups ☕

**Problem 7.1** [23pts] *(Voronoi cells)* Let $\mathbf{x}_A = (0,0)^\top$, $\mathbf{x}_B = (2,2)^\top$, and $\mathbf{x}_C = (0,4)^\top$ be three points in $\mathbb{R}^2$.

(a) [4pts] Hand-compute and sketch the equation of the set of all points that are equidistant from $\mathbf{x}_A$ and $\mathbf{x}_B$ when distance is measured by Euclidean distance.

(b) [9pts] Hand-compute and sketch the Voronoi-tessellation of $\mathbb{R}^2$ induced by the three points $\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C$ when distance is measured by Euclidean distance.

(c) [10pts] ☕ Repeat part (a) for Manhattan (i.e., $\ell_1$) distance.

**Problem 7.2** [7pts] *(Choosing the best k in kNN classification)* Consider a training set for binary classification with positive examples (label +1) at $A = (3,3)^\top, B = (5,1)^\top, C = (5,3)^\top$ and negative examples (label −1) at $D = (1,2)^\top, E = (3,2)^\top, F = (4,1)^\top$. To select a value of $k$ for $k$-NN classification (using Euclidean distance), we perform leave-one-out cross-validation (LOOCV). For $k = 1, 3, 5$, list the validation examples that will be misclassified and the corresponding LOOCV error (0/1 loss). What is the best value of $k$ (among these choices)?

**Problem 7.3** [10pts] In this problem, we will implement $k$-NN and use it on the cats vs dogs dataset (see Lecture 12). Warning: the training/validation/test split in the code is random, so you'll get different results every time you run it.

(a) [5pts] As said in class, $k$-NN does not require training. So, implement directly the $k$-NN prediction algorithm we saw in class in a Matlab function with prototype

```
[yhat] = predict_knn(X, y, Xtest, k)
```

where `yhat` is a column vector of predictions for the $n$ testing samples, `X` is the matrix of the training samples $\in \mathbb{R}^{m \times d}$, `y` is the vector of labels $\in \mathbb{R}^m$, `Xtest` is the matrix of the testing samples $\in \mathbb{R}^{n \times d}$, and $k$ the number of neighbors to consider. There is no skeleton code. Hint: to take the majority vote of $n$ numbers in $\{-1, 1\}$ it is enough to sum them and take the sign. Also, `sort` in Matlab sorts vector of numbers.

(b) [3pts] Let's now use the function in the point above to select the best $k$ using a validation set. You decide a good range of value of $k$ to try. Plot the 0/1 loss on the validation set with respect $k$.

(c) [2pts] Test on the test set using the best $k$ found using the validation set and report the 0/1 error.

**Code-submission via Blackboard:** You must prepare 1 files: `predict_knn.m` for Problem 7.3(a). Place them in a **single** directory which should be zipped and uploaded into Blackboard. Your directory must be named as follows: `<yourBUemailID>_hwX` where `X` is the homework number. For example, if your BU email address is `charles500@bu.edu` then for homework number 7 you would submit a single directory named: `charles500_hw7.zip` which contains all the MATLAB code (and only the code).

Three corresponding skeleton code files are provided for your reference. Reach-out to the TAs via Piazza and their office/discussion hours for questions related to coding.