ECE8560 Spring 2019
(Takehome #2)

Canvas submission only; rev. 2-26-2019

Assigned 3/7/2019; Due 4/11/2019 11:59 PM

1 Objectives

This is a comprehensive, ambitious and interesting five part take-home assignment which follows and builds upon Takehome #1 and uses the same dataset (H and S_T). **This is to be done individually**. I recommend you start as soon as possible. It is highly unlikely you can achieve a credible result by starting a few days before the due date.

The 5 objectives of the Takehome #2 assignment are:

1. To assess your previously developed Bayesian, minimum error classifiers;

2. To consider k-NNR classification on Case 1 with k=1,3,5, and compare this with ground truth;

3. To determine a separating hyperplane between each pair of classes in Case 1 using the book technique(s);

4. To employ an SVM for the Case 1 solution. Thus will involve learning some new software; and

5. To treat the overall problem (dataset) as unsupervised, and see what we can learn.

To simplify things slightly, we'll restrict the above Objectives **in Parts 2-5 to only the first two classes (w_1 and w_2) of Takehome #1 Case 1**. Specifically:

Case 1: $P(w_i) = 1/c$. Uses data in `train_case_1` and `test_case_1`.

Key remarks are:

1. Do all 5 parts of the assignment. Use this document as a checklist for your submission.
2. Open book and notes, but **no collaboration**. (This is an individual effort.)
3. **Submit to Canvas by the deadline.**
4. **Clarity of the presentation (in addition to technical correctness) counts significantly**.

2 Part 1: Assessing Classification Results from Takehome #1

2.1 Background

Recall this is a $c = 3$ class problem. You were given the $d = 4$ training and test data (H and S_T). Refer to Takehome #1 for more details. In this part of Takehome #2 (only), consider both Cases 1 and 2, and all three classes, namely:

1. Case 1: $P(w_i) = 1/c$. Uses data in `train_case_1` and `test_case_1`.
2. Case 2: $P(w_1) = 1/2; P(w_2) = 1/3; P(w_3) = 1/6$. Uses data in `train_case_2`, `test_case_2`.

Here you will assess your previous Bayesian classifier design and effort using your `takehome1_case_i.txt` files[1] corresponding to Cases 1 and 2 (i=1,2) with with one integer, a_i, per line. The format was

$$a_i = \begin{cases} 1 & \text{if sample i is classified as class } w_1 \\ 2 & \text{if sample i is classified as class } w_2 \\ 3 & \text{if sample i is classified as class } w_3 \end{cases}$$

[1]If you did not produce these files, skip this section of the assignment.

2.2 Assessment Knowing True Class

Below is the ground truth for the data for Takehome #1 in test files `test_case_1` and `test_case_2`. In each file, the feature vectors are arranged by class in the following sequence, starting from the first sample:

```
Case 1 test set pattern: 2-3-1-3-1-2

Case 2 test set pattern: 2-3-1-1-1-2
```

2.3 Specific Tasks

1. **Show the computed** $P(Error)$ **estimates for your classification results** using the `takehome1_case_i.txt` files you produced for Cases 1 and 2.

2. Show your classifier performance for each case **using a confusion matrix**.

Confusion Matrix. A 'confusion matrix', also referred to as a contingency table or an error matrix, is a tabular representation that allows visualization of the performance of a classification algorithm. Rows correspond to ground truth (true class); columns indicate accumulated resulting classifier decisions. A diagonal confusion matrix indicates perfect classification, i.e., estimated $P(Error) = 0$.

3 Part 2: Separating Hyperplane

In this part, you are to implement a (simple) Ho-Kayshap hyperplanar classifier. Do not use canned software. To keep things reasonable, only consider Case 1 training data (H) for two of the classes (**specifically** w_1 **and** w_2). This will enable you to test the discrimination ability of the resulting hyperplane. Clearly show the exact parameters of the resulting hyperplane. Test the hyperplane using the w_1 and w_2 samples from the Case 1 S_T[2].

[2]This requires some 'filtering' of S_T since it contains w_3.

4 Part 3: $k - NNR$ Strategies

In this approach, develop, apply and assess a k-NNR classifier for S_T (recall Case 1 only) using H. Generate results (classification error and confusion matrix) as a function of k. **For this problem part, use $k = 1$, 3, and 5.** You may use a brute-force matching strategy, but I'll be more impressed if you implement something more computationally efficient. Do not use canned software.

5 Part 4: SVM

5.1 A Simple (?) SVM Application

Again, to keep things reasonable, you will apply SVM to the Case 1 training data (H) for two of the classes (**specifically w_1 and w_2**), to test SVM discrimination ability. Test using the w_1 and w_2 samples from Case 1 S_T, as in Part 3.
As I'll note in class, you may use any SVM software package, but be sure you take the time to learn how to correctly use it. There are numerous decisions you will need to make. I recommend libSVM.

5.2 Tasks

1. Clearly indicate, justify and describe the SVM software tool you have chosen for the solution.
2. Consider linear and rbf models.
3. Show the resulting support vector set.
4. Show the corresponding hyperplane parameters (**This is important** and derivable from the support vector set).
5. Determine the classification performance with this SVM.
6. Compare these results with your results from Part 2.

Note: I need to see all of these desired results clearly described in your documentation.

6 Part 5: Unsupervised Case 1 Data Analysis Using c-means

Apply (crisp) c-means to **(all of)** Case 1 H, and determine if there is any natural clustering. **Write your own c-means implementation (i.e., do not use 'canned' software).**

1. Use $c = 2$, $c = 3$, $c = 4$, and $c = 5$.
2. Determine, justify and use at least 2 distance measures and assess their influence on the solution.
3. Compare the results of 1. and 2. above. Do any clusters naturally develop?
4. Assess whether the clusters found above are related to the known (estimated) class means.

7 Format of the Report Results

This aspect is critical. If you make it difficult for me to asses your effort, I won't. The final report must be in your solution archive in a PDF-format file named `results_takehome2.pdf`. The report results **must be in the following order**:

1. p. 0: Title page (<name>, <CU username>, ECE 8560, Takehome #2)
2. p. 1: Confusion matrix and P(error) estimate from Part 1.
3. p. 2: The hyperplane from the respective subsets of H in Part 2.
4. p. 3: k-NNR classifier results on S_T (3 classes) using H, including classification error and confusion matrix as a function of k from Part 3.
5. p. 4: SVM results from Part 4, as specified in Section 5.2.
6. p. 5: Unsupervised (c-means) results from Part 5.

7. p. 6: Comparison and analysis of Parts 1-5.

8. Pages 7 and beyond: Anything else you feel is relevant.

8 Additional Notes and Constraints

8.1 Format of the Electronic Submission

The final **zipped archive** is to be named `<yourname>`**-takehome2.zip**, **where** `<yourname>` **is your (CU) assigned user name**. You must upload this to the ECE8560 Canvas page prior to the deadline for your solution to be considered.

The minimal contents of this archive are described below.

1. Include a `readme.txt` file listing the contents of the archive and a brief description of each file. Include 'the pledge' here. Here's the pledge:

    Pledge:
    On my honor I have neither given nor received aid on this exam.

2. All documentation should be in pdf in a file named `<yourname>-takehome2.pdf` **No MS Word (doc) files.** The quality and structure of the documentation is very important.

3. Include **all** (your) source code used in your simulations for Parts 1-5 each in a single subdirectory.

4. Indicate your engineering decisions and solution derivations for each part.

5. Compare the results of Parts 1 thru 5.