Homework 5
M1522.001000 Computer Vision (2021 Fall)

Due: Thursday Dec 9, 11:59PM.

The goal of this homework is to explore SVM, K-means clustering and image classification. There are two Theory questions and eight **Python** programming questions on this assignment, and 100 points in total.

Put your **code and writeup** into a directory called "`(studentid)-(yourname)-HW5`" and pack it into a `zip` named "`(studentid)-(yourname)-HW5.zip`".

For example, `202012345-gildonghong-HW5.zip`.

Your writeup should be **typed** with **English**. Please do **not** attach your code to writeup. Upload your `zip` file to **ETL** until due date. Refer to the **ETL** page for the policies regarding collaboration, due dates, extensions, and late days.

Your homework should be formatted as following:

```
(studentid)-(yourname)-HW5
├─ writeup.pdf
└─ classifiers.ipynb
```

You should not include any other files including the dataset files automatically downloaded while running the provided preprocessing code.

Points will be deducted when your submission does not follow the above structure. Your `zip` file should be sent in before the due. Later than that, only one late day is allowed. Finally, note that we will use a code similarity checker to detect plagiarism. You are expected to work on the assignment individually. I firmly believe that every student can do his or her own work. For your sake, please do not copy and paste others code. Good Luck!

1 Theory Questions

1.1 SVM (20 points)

Suppose that training examples are points in 3-D space. The positive examples are $X_+ = \{(1,1), (-1,-1)\}$. The negative examples are $X_- = \{(1,-1), (-1,1)\}$.

(a) [4 pts] Are the positive examples linearly separable from the negative examples? (*i.e.* can you draw a line to separate the positive examples from negative examples)?

(b) [8 pts] Consider the feature transformation $\phi(x) = [1, x, y, xy]^T$, where x and y are the first and second coordinates of an example. Write down the transformed coordinates of X_+ and X_- (*i.e.* $\phi(X_+)$ and $\phi(X_-)$ for all four examples).

(c) [8 pts] Consider the prediction function $y(x) = w^T\phi(x)$. Give the coefficient w of a maximum-margin decision surface separating the positive from the negative examples.

(hint: w is $[4 \times 1]$ vector, whose elements are only 0 or 1).

1.2 K-means (20 points)

(a) [4 pts] Mahalanobis measure is one of many distance measure used for k-means. Given the below definition, describe the shape of the covariance matrix corresponding to σ_i and explain why Mahalanobis measure is also called scaled euclidean measure in this case.

$$d(x, c) = \sqrt{\sum_{i=1} \frac{(x_i - c_i)^2}{s_i^2}}$$

(b) [8 pts] Given a dataset $\mathcal{X} = \{0, 2, 4, 6, 18, 20\}$, initialize the k-means clustering algorithm with 2 cluster centers $c_1 = 3$ and $c_2 = 4$. What are the values of c_1 and c_2 after the first iteration of k-means? Also report the values after the second iteration.

(c) [8 pts] Given the same dataset as in (b), perform greedy initialization to get the initial $k = 3$ centers. Start with $c_1 = 4$. Below is the greedy initialization process.

1. Choose c_1
2. Choose the next center c_i to be $\text{argmax}_{x \in \mathcal{X}}\{D(X)\}$
3. Repeat step 2 until k centers are chosen
   where at any given time, with the current set of cluster centers $\mathcal{C}$,

   $D(x) = \min_{c \in \mathcal{C}} ||x - c||_2$

2 Programming Questions

First of all, make sure you have **Python** 3.8.3 installed. You can check your version with command `python -V`. To run the program, you will also need packages including Numpy and OpenCV. To install them with the package manager for Python, run `pip install -r requirements.txt`. Then, you should complete the code. Specifically, you have to fill in **eight blocks** including subquestions starting with `# START CODE HERE #` and ending with `# END CODE HERE #`.

It is forbidden to use any other external libraries except for the given package list in `requirements.txt`.

In this section, you will explore three techniques for image classification problem: kNN, Naive Bayes and Linear SVM. In this homework, you will complete `classifiers.ipynb` which have **eight blocks** to fill. Outline of the questions are as follows:

1. (K Nearest Neighbour)

    (a) (Distance Measure) Complete `_calculate_euclidean_distance` function to get euclidean distance between the test and training data points. [10 pts]

    (b) (Hyperparameter Tuning) Modify number of neighbours (k) and the distance function (`calculate_my_distance`) to obtain accuracy $\geq 25\%$. [5 pts]

2. (Naive Bayes)

    (a) (Counting) Update `nb_stats` variable to count necessary stats for the Naive Bayes algorithm. [10 pts]

    (b) (Likelihood) Complete `calculate_likelihood` function to calculate gaussian density of a datapoint `x` conditioned on the class index `class_idx`. [5 pts]

    (c) (Posterior) Given prior and likelihood, compute the posterior probability. [5 pts]

3. (Linear SVM) Complete the SVM inference code to get the top-1 predicted labels of the test dataset. [15 pts]

4. (Precision-Recall Curve)

    (a) (Counting) Count false positive, false negative, true positive and true negative instances. [5 pts]

    (b) (Precision and Recall) Compute precision and recall. [5 pts]

Conditions and hints for the respective questions are provided in the `classifiers.ipynb` file.

Finally, note that we will use a code similarity checker to detect plagiarism. You are expected to work on the assignment individually. We firmly believe that every students can do his or her own work. For your sake, please do not copy and paste others' code!