CS 475/675 Machine Learning: Homework 5

Due: Monday, December 5, 2022, 11:59 pm

25 Points Total Version 1.0

Make sure to read from start to finish before beginning the assignment. This assignment has undergone a major revision since the original release.

1 Introduction

This homework assignment has two parts:

1. **Analytical (12 points):** This part will ask you to consider questions related to the topics covered by recent lectures.
2. **Lab (13 points):** You will implement a machine learning algorithm and answer a series of questions in a Python notebook.

2 Collaboration Policy

The course policy is that, unless otherwise specified, all work must be your own. See the course information Google document for more details.

For this assignment, you may work in groups of 1 to 3 students. You and your group will make a single submission to Gradescope. You will be able to indicate all students in your group when you submit. The entire group will receive the same grade, so please choose your group carefully.

You can only work in teams of one, two or three students; no more. Your group can include anyone from either section (01/02/03/04) or course (475 or 675) provided that all team members are enrolled in the class and taking it for credit (not audit). We highly recommend that you do every part of the assignment together instead of splitting it up. Our intention is to include questions on the exam that require an understanding of all parts of the homework. It is to your advantage to work together on all parts.

You can work on the same Overleaf document and think through the questions together. You probably want to work with the group for the semester (only for assignments where collaboration is allowed) but it is not a requirement. Please see the course policy document for an explanation of how late hours apply in a group.

3 Analytical

Please complete the Analytical portion of the homework by using the provided Latex Template. Do not change the size of the answer boxes in the latex template.

4 Lab

For this part, you will work on the Python notebook **hw5_notebook**. You will analyze the Baltimore crime data collected by Baltimore Police Department using clustering methods. You will implement the K-Means algorithm from scratch, train the model on the given dataset, and visualize the clustering results.

4.1 Dataset

We will provide you with the dataset of crime in Baltimore: **BPD_2012_2017.cvs**. This dataset represents the location and characteristics of major crime against persons such as homicide, shooting, robbery, aggravated assault etc. within the City of Baltimore. It was recorded by the Baltimore Police Department[1] from 2012 to 2017. One can conduct various analyses on this dataset. For example, what's the major crime that happening in certain districts? Is there a pattern on both time and location for certain types of crime? Does the crime distribution change over years? Which is the most dangerous or safest district? One can also apply machine learning algorithms to make predictions in the future. For example, logistic regression can be trained to predict which type of crime is most likely to happen given the location and time.

In this assignment, we want to divide Baltimore into subareas by crime distribution. It is useful for planning the optimal locations for police stations. And we can do it by clustering.

There are 276,529 entries and 15 attributes in the dataset. We will only consider two attributes: *longitude* and *latitude* for clustering.

4.2 Implementation

You will implement the K-Means algorithm by filling in the blanks in the given code framework. Specifically, you need to implement two functions: **fit** and **predict**.

fit: Train the K-Means model on the data, i.e. find the optimal prototype or centroid for each cluster.

predict: Assign cluster label to examples.

4.2.1 Implement fit Function

You should implement the **fit** function as following.

1. Initialization: assign a random example as the prototype for each cluster.
2. Iterative training:
   a. Assign examples to clusters based on Euclidean distance.
   b. Based on the cluster assignments, update the prototypes by averaging over the examples within a cluster.

4.2.2 Check Convergence

You can check model convergence by comparing the current cluster assignments and those in previous iteration. If they stay the same, then the model converged. You should stop

[1] If you are interested in most recent crime related statistics in Baltimore, you can refer to Open Baltimore.

training. You can also set up the maximal iteration (*max_iter*) to early stop the training if you have limited computation resources.

4.2.3 Search for the Elbow

You will select the optimal number of clusters *n_clusters* by plotting the curve of model performance with respect to different choices of *n_clusters*. You need to decide on and report the elbow (best *n_clusters*).

For this homework, we will use the criterion called **inertia** to measure the performance of a K-Means model. It is defined as follows:

$$\sum_{i=1}^{N} \min_{\mu_k \in C} (||\mathbf{x}_i - \mu_k||^2), \tag{1}$$

where N is the number of samples, x_i is the ith sample, and μ_k is the prototype or centroid of the kth cluster.

For the purpose of finding the best *n_clusters*, you can train on a subset of the original dataset. (The subset *X_small* is defined in the notebook. Please don't change that line of code.) It is recommended to set *max_iter* to a number around 20 when you call the *plot_tune_n_clusters* function. You can change *max_iter* to a larger number, e.g. 100, after you have decided on the number of clusters in later experiments.

4.2.4 Visualization

You will train the model with the selected number of clusters and visualize the cluster results with the provided function.

4.3 Python Packages

For this assignment, you are only allowed to use numpy, pandas and matplotlib. You should not use any other Python packages for implementation.

4.4 Questions

You need to answer the following questions in your Python notebook.

Q1. (11 points) Show the plot of inertia with number of clusters from 1 to 15. Report which number of clusters is at the elbow.

Q2. (2 points) Train your model with the selected number of clusters in Q1 and plot the cluster assignments using the *plot_clusters* function.

5 What to Submit

For this assignment you will submit the following items to Gradescope.

1. "Homework 5: Analytical" A PDF of the Analytical homework based on the provided Latex template.

2. "Homework 5: Lab" A PDF of the Python Lab notebook with your model implementation and answers to the requested questions.

6 Questions?

Remember to submit questions about the assignment to Piazza: `https://piazza.com/class/l7542wgbgfu7a8`.