Assignment 4

CMSC 678 — Introduction to Machine Learning

Due Tuesday May 16th, 11:59 PM

Item	**Summary**
Assigned	Thursday May 4th
Due	Tuesday May 16th
Topic	Clustering/Kernels, and Probabilistic Modeling
Points	100

In this assignment you will gain experience implementing clustering algorithms, in particular k-means. You will also gain experience with generative probabilistic models, and do an in-depth reading and analysis of a seminal work on PGMs.

You are to *complete* this assignment on your own: that is, the code and writeup you submit must be entirely your own. However, you may discuss the assignment at a high level with other students or on the discussion board. Note at the top of your assignment who you discussed this with or what resources you used (beyond course staff, any course materials, or public Discord discussions).

The following table gives the overall point breakdown for this assignment.

Question	1	2	3
Points	20	35	45

What To Turn In Turn in a **PDF** writeup that answers the questions; turn in all requested code necessary to replicate your results. Be sure to include specific instructions on how to build (compile) and run your code. Answers to the following questions should be long-form. Provide any necessary analyses and discussion of your results.

How To Submit Submit the assignment on the submission site:

https://www.csee.umbc.edu/courses/graduate/678/spring23/submit.

Be sure to select "`Assignment 4`."

1. (**20 points**) In class, we studied K-means; this is often called a *hard assignment* clustering algorithm. This question looks at Gaussian mixture models (GMM), a *soft assignment* clustering algorithm. You can think of it as a version of the 3-coins problem, but where the observations are real-valued vectors rather than coin flip outcomes. Your tasks in this question are to (a) derive an iterative update algorithm for GMM, and (b) argue for a link between K-means and GMM.

   Suppose we have a dataset of N data points $\{x_1, x_2, \ldots, x_N\}$, where each datum is a D-dimensional real vector, $x_i \in \mathbb{R}^D$. Let the mixture model P be given by

$$P(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x; \mu_k, \sigma^2 I_D) \qquad \text{[Eq-1]}$$

   where $x, \mu_k \in \mathbb{R}^D$, σ^2 is a scalar variance, I_D is the $D \times D$ identity matrix, and π_k represents a mixing distribution. That is, $\sum_k \pi_k = 1$ and for each k, $0 \leq \pi_k \leq 1$.

   The GMM mixture given by [Eq-1] is a soft clustering algorithm: it's a clustering algorithm because it assumes that each data point x_i is generated according to a particular Gaussian distribution k, but it's *soft* because we don't know *which* Gaussian generated each data point. We represent this uncertainty in part by π_k.

   (a) Write the log-likelihood $\mathcal{L}$ of data $\{x_1, \ldots, x_N\}$ according to [Eq-1]. You may assume that the data are i.i.d.

   (b) What is the (computation/equation) for the marginal likelihood $P(x_i)$?

   (c) Assume that σ is a fixed value. Derive the gradients for the log-likelihood with respect to μ_k and π_k. You may find $\gamma_{i,k}$, the posterior probability of cluster k for a point x_i, to be useful.

$$\gamma_{i,k} = p(x_i \text{ generated from Gaussian } k | x_i) = \frac{\pi_k \mathcal{N}(x_i; \mu_k, \sigma^2 I_D)}{P(x_i)} \qquad \text{[Eq-2]}$$

   (d) How would you use the gradients you found above to optimize [Eq-1]?

   (e) What happens as $\sigma \to 0$? Argue for a link between this mixture model and K-means. You do not have to be formal and fully rigorous, but don't simply say, "There is a link between them." Show us that you understand where this link is, and why (even if you can't formally prove it).

   Hint: to help answer this problem, you may want to read through question 2.

2. (**35 points**) Implement K-means clustering algorithm (you will probably want to implement Lloyd's algorithm, as discussed in class). Recall that given K clusters $C_1, \ldots, C_K$ that partition N different data points, K-means minimizes the objective

$$J = \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - \mu_k\|^2 = \sum_{k=1}^{K} \sum_{i=1}^{N} \lambda_{i,k} \|x_i - \mu_k\|^2 \qquad \text{[Eq-3]}$$

[Eq-3] presents two equivalent formulations of the objectives. Note that $\lambda_{i,k} \in \{0, 1\}$ in the second formulation represents whether point x_i is assigned to cluster k. In particular, with the constraint that $\sum_k \lambda_{i,k} = 1$, for each i, each point can only be assigned to one cluster.

After you have implemented K-means, discuss what you observe as you vary **all** of the following:

- the initial cluster points,
- the number of clusters,
- the number of data points, and
- the underlying generating distribution (including the dimensionality of the points).

As part of these discussions, plot the objective, [Eq-3].

To help you get started, you may use the 100, 2-dimensional data points available at

`https://www.csee.umbc.edu/courses/graduate/678/spring23/materials/a4-data/2d-kmeans-input.ssv`

This is the same initial data as shown in the web demo.[1] However, you should also experiment with other data. These data can be randomly generated. Equation [Eq-1] provides an easy way of doing so: set a weighting distribution $\{\pi_k\}$, and then for each data point x_i, sample a discrete index z_i (based on $\{\pi_k\}$), and then sample x_i from the z_ith distribution. (For [Eq-1], the z_ith distribution will be a multivariate Gaussian distribution. However, you could consider some to be Gaussian distributions, others to be Gamma distributions, or Poisson distributions, or pretty much any other distribution.)

3. (**45 points**) Read "the Pegasos paper" (Shalev-Shwartz et al., 2011)

```
@article{shalev2011pegasos,
  title={Pegasos: Primal Estimated sub-{G}r{A}dient
         {SO}lver for {SVM}},
  author={Shalev-Shwartz, Shai and Singer, Yoram and
          Srebro, Nathan and Cotter, Andrew},
  journal={Mathematical programming},
  volume={127},
  number={1},
  pages={3--30},
  year={2011},
  publisher={Springer}
}
```

available at `https://www.cs.huji.ac.il/~shais/papers/ShalevSiSrCo10.pdf`, You may **skim** section 3, but read all other sections carefully. (Don't let section 3 prevent you from reading the rest of the paper.) Answer the following questions about it.

[1] `https://www.csee.umbc.edu/courses/graduate/678/spring23/kmeans`

(a) The first paragraph of the introduction uses the terms "unconstrained," "empirical loss minimization," and "penalty term for the norm of the classifier." Identify the mathematical instantiations of these terms in equations 1 and 2 (i.e., what parts of equations 1 and 2 correspond to "unconstrained," "empirical loss minimization," and "penalty term?").

(b) What is the main point of the discussion of the methods on pages 3-4?

(c) Describe, in written English, the mini-batch Pegasos algorithm, including why this algorithm works on "mini-batches" and the purpose of the projection step (eqn 6).

(d) What is the importance of a kernel operator $K(x, x')$?

(e) Describe, in written English, the meanings of the formulas $w_{t+1} = \ldots$ on page 12. (If you feel it helpful, you may refer to the algorithm in Figure 3.)

(f) What is the main contribution of section 5?

(g) Earlier in the semester, we saw both hinge loss and log-loss as surrogate loss functions (to 0-1 loss).[2] Describe, in English and/or with an example (your choice), when hinge loss and log-loss will (each) be low (close to or equal to 0) and high. In your answer, consider the asymptotic behavior of both (and speculate on the implications for using them as a loss function).

(h) Describe, in English, equations 20 or 21 (the multiclass variants of hinge loss and log-loss). When will the loss be high and when will it be low?

(i) The bottom of page 14 formally defines the subgradient for a function f (remember the notation $\langle u, q \rangle$ indicates the dot product between vectors u and q). Argue why the subgradient for binary hinge loss (presented in the table on page 15) is a subgradient. You may use, without proof, the fact that hinge loss is a convex function.

(j) What is the definition of a Gaussian kernel (sometimes also called an RBF—radial basis function)?

(k) Now write a (roughly) 1/2-1 page analysis of the entire paper. In your analysis, include a summary of the method, the results, and what *you* took away from the paper. Identify and discuss items that were confusing, underspecified, or counter-intuitive. This analysis may reuse, as appropriate, portions of your previous answers.

[2] In slide deck 3, we also called log-loss logistic loss; see slide 84.