Homework 4: SVM, Clustering, and Ethics

Introduction

This homework assignment will have you work with SVMs, clustering, and engage with the ethics lecture.

Please submit the **writeup PDF to the Gradescope assignment 'HW4'**. Remember to assign pages for each question.

Please submit your **LaTeX file and code files to the Gradescope assignment 'HW4 - Supplemental'**.

You can use a **maximum of 2 late days** on this assignment. Late days will be counted based on the latest of your submissions.

Problem 1 (Fitting an SVM by hand, 10pts)

For this problem you will solve an SVM by hand, relying on principled rules and SVM properties. For making plots, however, you are allowed to use a computer or other graphical tools.

Consider a dataset with the following 7 data points each with $x \in \mathbb{R}$ and $y \in \{-1, +1\}$:

$$\{(x_i, y_i)\}_{i=1}^{7} = \{(-3, +1), (-2, +1), (-1, -1), (0, +1), (1, -1), (2, +1), (3, +1)\}$$

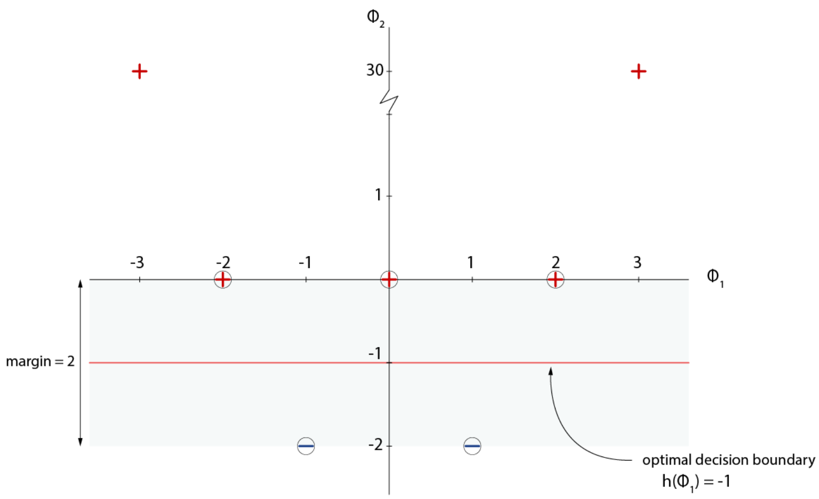
Consider mapping these points to 2 dimensions using the feature vector $\phi(x) = (x, -\frac{8}{3}x^2 + \frac{2}{3}x^4)$. The hard margin classifier training problem is:

$$\begin{aligned} &\min_{\mathbf{w}, w_0} \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & y_i(\mathbf{w}^\top \phi(x_i) + w_0) \geq 1, \ \forall i \in \{1, \dots, n\} \end{aligned}$$

Make sure to follow the logical structure of the questions below when composing your answers, and to justify each step.

1. Plot the transformed training data in $\mathbb{R}^2$ and draw the optimal decision boundary of the max margin classifier. You can determine this by inspection (i.e. by hand, without actually doing any calculations).
2. What is the value of the margin achieved by the optimal decision boundary found in Part 1?
3. Identify a unit vector that is orthogonal to the decision boundary.
4. Considering the discriminant $h(\phi(x); \mathbf{w}, w_0) = \mathbf{w}^\top \phi(x) + w_0$, give an expression for *all possible* $(\mathbf{w}, w_0)$ that define the decision boundary. Justify your answer.
5. Consider now the training problem for this dataset. Using your answers so far, what particular solution to $\mathbf{w}$ will be optimal for the optimization problem?
6. What is the corresponding optimal value of w_0 for the $\mathbf{w}$ found in Part 5 (use your result from Part 4 as guidance)? Substitute in these optimal values and write out the discriminant function $h(\phi(x); \mathbf{w}, w_0)$ in terms of the variable x .
7. What are the support vectors of the classifier? Confirm that the solution in Part 6 makes the constraints above binding for these support vectors.

1. plotting transformed data and decision boundary



2. margin = 2

3. $w = (0, 1) \perp h$

4.

i	ϕ_1	ϕ_2	y	$y_i(w^T \phi_i + w_0) \geq 1$
1	-3	30	+1	$+(-3w_1 + 30w_2) + w_0 \geq 1$
2	-2	0	+1	$+(-2w_1 + 0w_2) + w_0 \geq 1$
3	-1	-2	-1	$-(-1w_1 + -2w_2) - w_0 \geq 1$
4	0	0	+1	$+(0w_1 + 0w_2) + w_0 \geq 1$
5	1	-2	-1	$-(1w_1 + -2w_2) - w_0 \geq 1$
6	2	0	+1	$+(2w_1 + 0w_2) + w_0 \geq 1$
7	3	30	+1	$+(3w_1 + 30w_2) + w_0 \geq 1$

simplifying the expressions

$$
\begin{aligned}
-3w_1 + 30w_2 + w_0 &\geq 1 \\
-2w_1 \qquad\quad + w_0 &\geq 1 \\
w_1 + 2w_2 - w_0 &\geq 1 \\
+w_0 &\geq 1 \\
-w_1 + 2w_2 - w_0 &\geq 1 \\
2w_1 \qquad\quad + w_0 &\geq 1 \\
3w_1 + 30w_2 + w_0 &\geq 1
\end{aligned}
$$

we can use a linear solver to solve for the linear inequalities

5. optimal $\mathbf{w} \rightarrow w_1 = 0, w_2 = 1$

6. the discriminant in terms of x for the optimal $\mathbf{w}$

$$h(x) = (\frac{-8}{3}x^2 + \frac{2}{3}x^4) + 1$$

7. support vectors are highlighted with a circle in the plot of part 1 and they are:

(-2,0) , (-1,-2) , (0,0) , (1,-2), (2,0)

and plugging these vectors in the discriminant $y_i(w^T\phi_i + w_0)$ with the optimal found $\mathbf{w} \rightarrow (0,1)$ and optimal $w_0 \rightarrow 1$

$$+1\left(\begin{pmatrix}0 & 1\end{pmatrix}\begin{pmatrix}-2\\0\end{pmatrix} + 1\right) = 1$$

$$-1\left(\begin{pmatrix}0 & 1\end{pmatrix}\begin{pmatrix}-1\\-2\end{pmatrix} + 1\right) = 1$$

$$+1\left(\begin{pmatrix}0 & 1\end{pmatrix}\begin{pmatrix}0\\0\end{pmatrix} + 1\right) = 1$$

$$-1\left(\begin{pmatrix}0 & 1\end{pmatrix}\begin{pmatrix}1\\-2\end{pmatrix} + 1\right) = 1$$

$$+1\left(\begin{pmatrix}0 & 1\end{pmatrix}\begin{pmatrix}2\\0\end{pmatrix} + 1\right) = 1$$

This confirms that the solution in part 6 makes binding for all the support vectors

Problem 2 (K-Means and HAC, 20pts)

For this problem you will implement K-Means clustering and HAC from scratch. Using `numpy` is fine, but don't use a third-party machine learning implementation like `scikit-learn`. You will then apply this approach to the clustering of image data.

We've provided you with a subset of the MNIST dataset, a collection of handwritten digits used as a benchmark for image recognition (you can learn more about the data set at http://yann.lecun.com/exdb/mnist/). The MNIST task is widely used in supervised learning, and modern algorithms do very well.

Here you will apply unsupervised learning to MNIST. You have been given representations of MNIST images, each of which is a 784×1 greyscale handwritten digit from 0-9. Your job is to implement K-means clustering and HAC on MNIST, and to test whether these relatively simple algorithms can cluster similar-looking images together.

The code given in `T4_P2.py` loads the images into your environment into two arrays – `large_dataset` is a 5000x784 array that should be used for K-means, while `small_dataset` is a 300x784 array that will be used for HAC clustering. In your code, you should use the ℓ_2 norm (i.e. Euclidean distance) as your distance metric.

Important: Remember to include all of your plots in your PDF submission!

1. Starting at a random initialization and $K = 10$, plot the K-means objective function (the residual sum of squares) as a function of iterations and verify that it never increases.

2. Run the K-means algorithm for 5 different restarts for different values of K, setting $K = 5, 10, 20$. Plot the final K-means objective value as a function of K with error bars over the 5 random restarts. To clarify, your x-axis will be K, your y-axis will be the average objective function value after your algorithm converges, and each data point will have an error bar – to calculate these error bars you must run your K-means algorithm 5 times for each K (giving you multiple final objective values for each K) then use these values to calculate a standard deviation for each K before plotting error bars around each data point. How does the final value of the objective function and the standard deviation of the final value of the objective function change with K? (Note: Our code takes 10 minutes to run for this Part)

3. For $K = 10$ and for 5 random restarts, show the mean image (aka the centroid) for each cluster. To render an image, use the pyplot `imshow` function. There should be 50 total images. Include all of these images as part of a single plot (e.g. don't have 50 pages in your write-up with a separate image on each page).

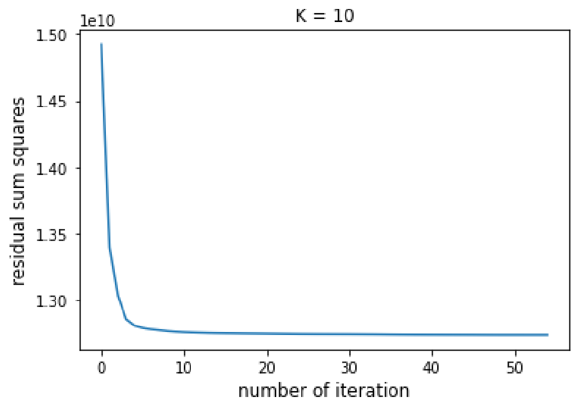
4. Repeat Part 3, but first standardize the data. That is, center the data before running K-means on it, such that each pixel has mean 0 and variance 1 (except for any pixels that have zero variance, for these you can simply divide by 1). For $K = 10$ and 5 random restarts, show the mean image (aka the centroid) for each cluster. There should be 50 total images. Again, include all of these images as part of a single plot. Compare these images to those from Part 3.

5. Implement HAC for min, max, and centroid-based linkages. Fit these models to the `small_dataset` images. For each of these 3 linkage criteria, find the mean image for each cluster when using 10 clusters, and display these images on a plot. There should be 30 total images. How do these mean images compare to those found with K-means? **Important Note:** For this part only, you may use the `scipy` package's `cdist` function to calculate the Euclidean distances between every pair of points in two arrays. DO NOT use `scipy` for anything else.

6. For each of the 3 HAC linkages (max/min/centroid), make a plot of "Distance between most recently merged clusters" (y-axis) v. "Total number of merges completed" (x-axis). Does this plot suggest that there are any natural cut points?

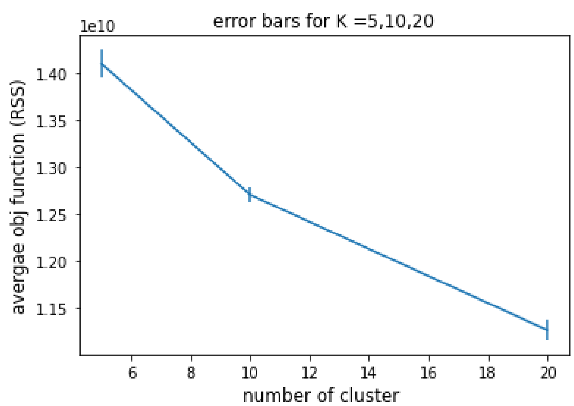
7. Re-fit a K-means with $K = 10$ model and HAC min/max/centroid models using 10 clusters on the `small_dataset` images. Use the `seaborn` module's `heatmap` function to plot a confusion matrix of clusters v. actual digits, i.e. the cell at the ith row, jth column of your confusion matrix should be the number of times that an image with the true label of j appears in cluster i. How well do the different approaches match the digits? Is this matching a reasonable evaluation metric for the clustering? Explain why or why not.

1. plotting the K-means objective function (the residual sum of squares) as a function of iterations:



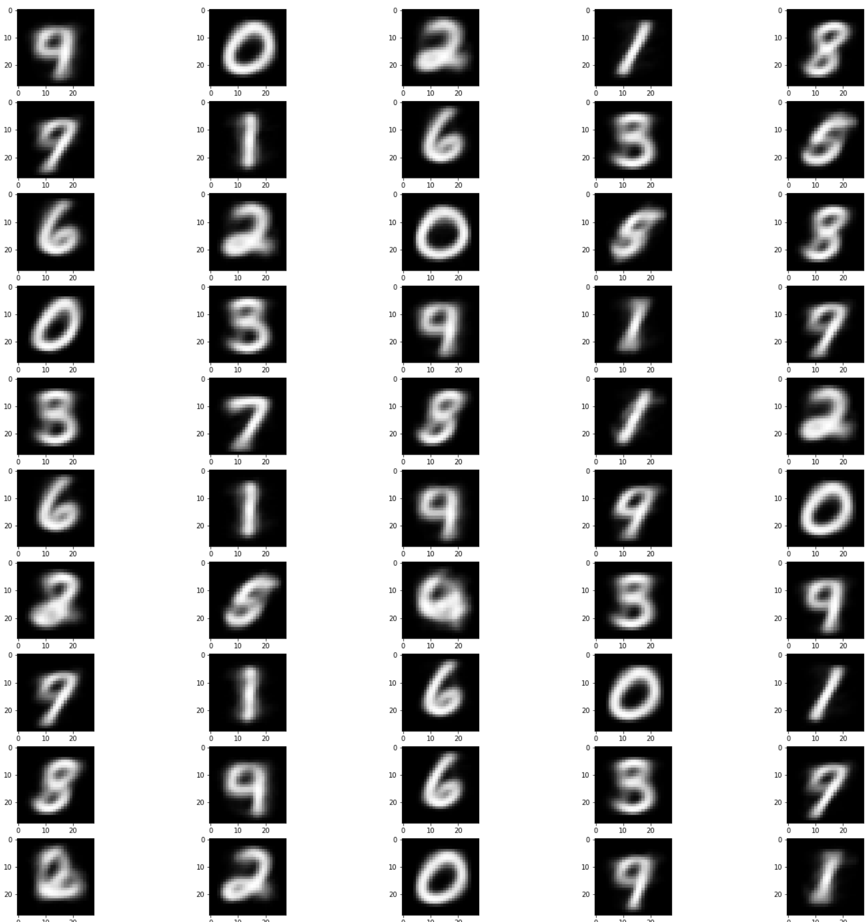
the plot shows that the objective function is never increasing with increase number of iteration, but it is decreasing until it converges at around iteration 55

2. plotting the K-means of 5 random restarts with error bar for k=5,10,20:

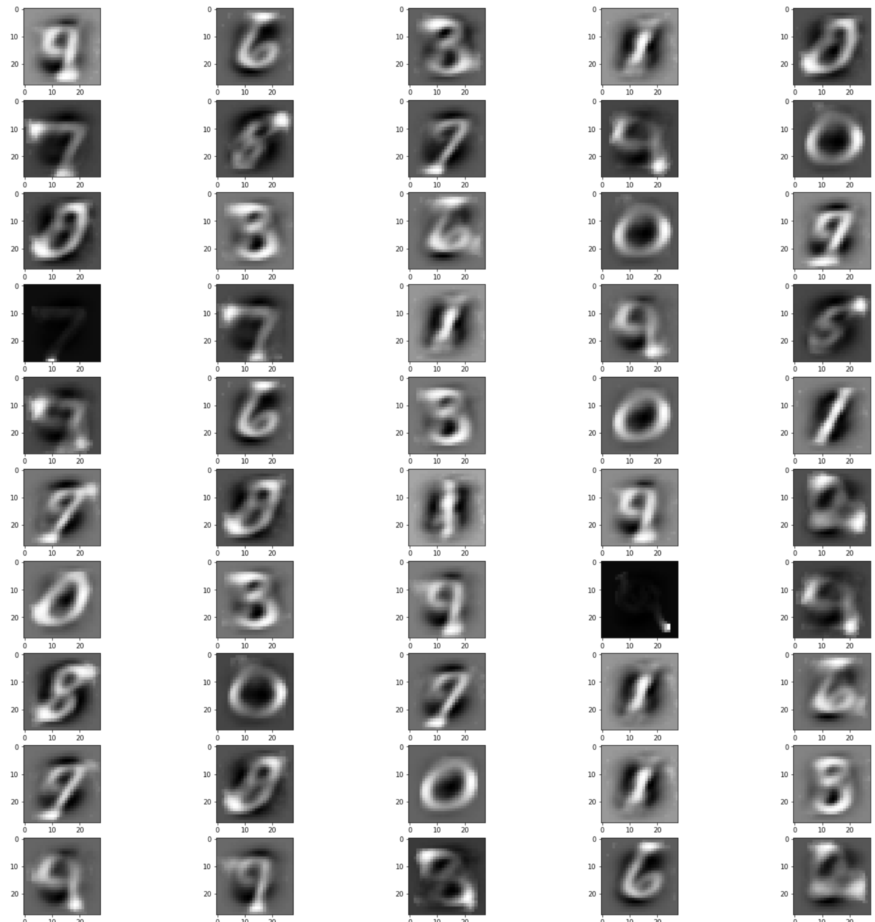


The error plot shows that the value of the objective function decreases as k increases, and increases at a higher rate between k=5 and k=10. The standard deviation for k=10 is the lowest, which indicates a better clustering of data than when k=5 and k=20

3. K= 10, 5 random restarts, no standardization



4. K= 10, 5 random restarts, with standardization



There is a slight improvement when standardizing the data and running the clustering. In restart 3 of the standardized version (third column) it clusters the 0-9 digits pretty well, except for the three and one, however not so good in the other 4 restarts.

5.

Problem 3 (Ethics Assignment, 15pts)

Imagine that you are a product manager in a technology company and the board of directors requests a detailed report of the potential social impact of the product you're building. You are currently working on an algorithm designed to allocate economic resources for various health providers. You are instructed to address various kinds of possible impacts, giving special attention to the algorithm's potential to wrongfully discriminate against vulnerable populations.

Having followed the coding process closely, you can confidently assume that there is no discriminatory intent animating the design. In fact, the product's design is based on the notion that resources should be distributed to health centers in a manner that is proportional to the needs of the populations they attend to. However, you worry that there may still be wrongful discrimination due to disparate negative impact on members of low-income populations, migrants, and racial minorities.

What further questions must you address in order to confidently determine whether the algorithm wrongfully discriminates against members of these groups? Write two questions and, for each, write a short paragraph explaining how addressing this question can help you assess the algorithm's potential for wrongful discrimination.

We expect clear, concise, and thoughtful engagement with this question, which includes providing your reasoning for your answers. In your response, depth is more important than breadth. We do *not* expect you to do any outside research, though we encourage you to connect to lecture materials where relevant.

Name

Collaborators and Resources

Whom did you work with, and did you use any resources beyond cs181-textbook and your notes?

Calibration

Approximately how long did this homework take you to complete (in hours)?