

HOMework 6: UNSUPERVISED LEARNING & ALGORITHMIC BIAS

10-301/10-601 Introduction to Machine Learning (Summer 2024)
<https://www.cs.cmu.edu/~hchai2/courses/10601/>

OUT: Tuesday, June 25th

DUE: Tuesday, July 2nd

START HERE: Instructions

- **Collaboration policy:** Collaboration on solving the homework is allowed, after you have thought about the problems on your own. It is also OK to get clarification (but not solutions) from books or online resources, again after you have thought about the problems on your own. There are two requirements: first, cite your collaborators fully and completely (e.g., “Jane explained to me what is asked in Question 2.1”). Second, write your solution *independently*: close the book and all of your notes, and send collaborators out of the room, so that the solution comes from you only. See the Academic Integrity Section on the course site for more information: <https://www.cs.cmu.edu/~hchai2/courses/10601/#Syllabus>
- **Late Submission Policy:** See the late submission policy here: <https://www.cs.cmu.edu/~hchai2/courses/10601/#Syllabus>
- **Submitting your work:**
 - **Programming:** You will submit your code for programming questions on the homework to Gradescope (<https://gradescope.com>). After uploading your code, our grading scripts will autograde your assignment by running your program on a virtual machine (VM). When you are developing, check that the version number of the programming language environment (e.g. Python 3.9.12) and versions of permitted libraries (e.g. numpy 1.23.0) match those used on Gradescope. You have a **total of 10 Gradescope programming submissions**. Use them wisely. In order to not waste code submissions, we recommend debugging your implementation on your local machine (or the linux servers) and making sure your code is running correctly first before any Gradescope coding submission.
 - **Written:** For written problems such as short answer, multiple choice, derivations, proofs, or plots, we will be using Gradescope (<https://gradescope.com/>). Please use the provided template. Submissions must be written in LaTeX. Regrade requests can be made, however this gives the TA the opportunity to regrade your entire paper, meaning if additional mistakes are found then points will be deducted. Each derivation/proof should be completed on a separate page. For short answer questions you **should not** include your work in your solution. If you include your work in your solutions, your assignment may not be graded correctly by our AI assisted grader.
- **Materials:** The data that you will need in order to complete this assignment is posted along with the writeup and template on the course website.

For multiple choice or select all that apply questions, shade in the box or circle in the template document corresponding to the correct answer(s) for each of the questions. For \LaTeX users, replace `\choice` with `\CorrectChoice` to obtain a shaded box/circle, and don't change anything else.

1 Principal Component Analysis (7 points)

For this section, refer to the PCA demo linked [here](#). In this demonstration, we have performed PCA for you on a [simple four-feature dataset](#). The questions below have also been added to the colab notebook linked for ease of access. Run the code in the notebook, then answer the questions based on the results.

- (1 point) **Select one:** Do you see any special relationships between any of the features? In particular, take a look at the `petal_length` feature. How would you describe its association with each of the **other features**? Select the correct statement with appropriate justification.
 - ☐ The features are highly correlated: we observe linearly proportional relationships where increases in `petal_length` often correspond to increases in another feature
 - ☐ The features are highly correlated: we observe that the color classes can be separated with decision boundaries along the `petal_length` axis.
 - ☐ The features are uncorrelated: we observe random noise as if the features were generated from independent distributions
 - ☐ The features are uncorrelated: we observe the “default $y = x$ ” relationship between features
- (1 point) If we wanted to find k principal components such that we preserve **at least** 95% of the variance in the data, what would be the value of k ? Hint: it is helpful here to look at the cumulative variance in the first k components, which we have calculated for you.

k

- (1 point) **Select one:** Assume we apply PCA to a matrix $\mathbf{X} \in \mathbb{R}^{n \times 2}$ and obtain two sets of PCA feature scores, $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^n$, where \mathbf{z}_1 corresponds to the first principal component and \mathbf{z}_2 corresponds to the second principal component. Which is more common in the training data:
 - ☐ a point with small feature values in \mathbf{z}_2 and large feature values in \mathbf{z}_1
 - ☐ a point with large feature values in \mathbf{z}_2 and small feature values in \mathbf{z}_1
 - ☐ a point with large feature values in \mathbf{z}_2 and large feature values in \mathbf{z}_1
 - ☐ a point with small feature values in \mathbf{z}_2 and small feature values in \mathbf{z}_1

4. (2 points) **Select all that apply:** To get the principal components of the features, we calculate the eigenvectors of the covariance matrix, which are orthogonal, along with their corresponding eigenvalues. Which of the following are consequences of the principal components being orthogonal to each other?
- ☐ The variance of the data is maximized.
 - ☐ The reconstruction error is minimized.
 - ☐ The dot product of any two principal components will be 1.0.
 - ☐ We can attribute certain variations in the data to unique principal components.
 - ☐ In the dimensionality-reduced space, the covariance of the first and second dimensions will always be zero.
 - ☐ It ensures that our lower-dimensional data will be linearly separable.
 - ☐ None of the above.
5. (2 points) **Select all that apply:** If we wanted to perform dimensionality reduction to have just two features and train a classifier on them, we could represent our data by EITHER (a) picking any two features from the dataset, OR (b) using the first 2 principal components we obtained from PCA. Which one should we prefer and why?
- ☐ We prefer (a), because it ensures randomness in the selection and have a better chance of representing the data well.
 - ☐ We prefer (a), because PCA introduces artificial bias and does not reflect the original features.
 - ☐ We prefer (b), because PCA preserves higher variance of the data than two raw features.
 - ☐ We prefer (b), because PCA ensures that variance is evenly distributed across the features.
 - ☐ We prefer neither, because it is impossible to train such a classifier.
 - ☐ Either is fine, because there are only two features in this dataset.
 - ☐ None of the above.

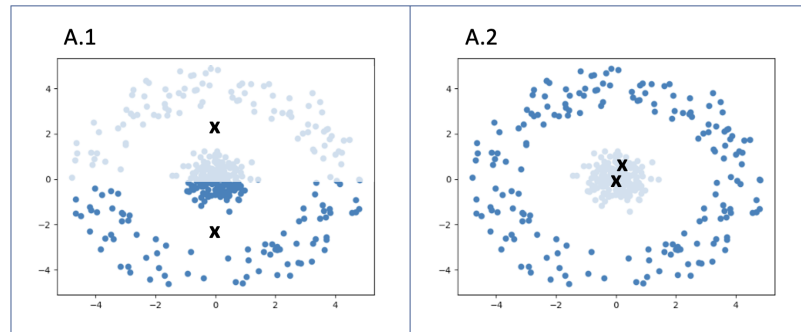
2 k -Means (12 points)

1. Consider the 2 datasets A and B. Each dataset is classified into k clusters, with centers marked X and cluster membership represented by different colors in the figure. For each dataset, exactly one clustering was generated by k -means with Euclidean distance. Select the image with clusters generated by k -means.

(a) (1 point) Dataset A

Select one:

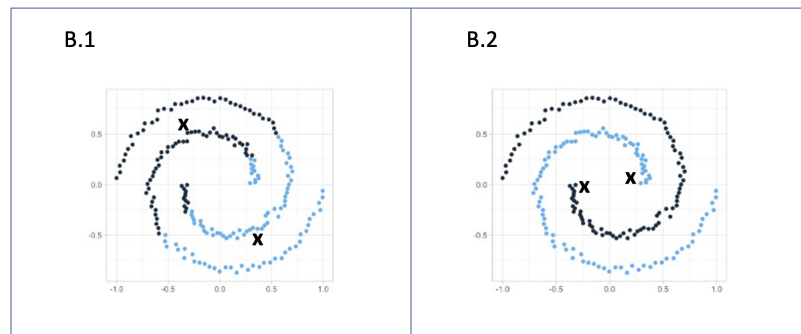
- ☐ A.1
☐ A.2



(b) (1 point) Dataset B

Select one:

- ☐ B.1
☐ B.2



2. Consider a dataset \mathcal{D} with 5 points as shown below. Perform a k -means clustering on this dataset with $k = 2$ using the Euclidean distance as the distance function. Remember that in the k -means algorithm, one iteration consists of following two steps: first, we assign each data point to its nearest cluster center; second, we recompute each center as the average of the data points assigned to it. Initially, the 2 cluster centers are chosen randomly as $\mu_0 = (5.3, 3.5)$, $\mu_1 = (5.1, 4.2)$. Parts (a) through (d) refer only to the first iteration of k -means clustering performed on \mathcal{D} .

$$\mathcal{D} = \begin{bmatrix} 5.5 & 3.1 \\ 5.1 & 4.8 \\ 6.6 & 3.0 \\ 5.5 & 4.6 \\ 6.8 & 3.8 \end{bmatrix}$$

- (a) (1 point) **Select one:** Which of the following points will be the new center for cluster 0?

- ☐ (5.7 , 4.1)
☐ (5.6 , 4.8)
☐ (6.3 , 3.3)
☐ (6.7 , 3.4)

- (b) (1 point) **Select one:** Which of the following points will be the new center for cluster 1?

- ☐ (6.1 , 3.8)
☐ (5.5 , 4.6)
☐ (5.4 , 4.7)
☐ (5.3 , 4.7)

- (c) (1 point) How many points will belong to cluster 0, using the new centers?

Answer

- (d) (1 point) How many points will belong to cluster 1, using the new centers?

Answer

3. Recall that in k -means clustering we attempt to find k cluster centers μ_1, \dots, μ_k such that the total distance between each point and the nearest cluster center is minimized. We thus solve

$$\operatorname{argmin}_{\mu_1, \dots, \mu_k} \sum_{i=1}^N \min_{j \in \{1, \dots, k\}} \|\mathbf{x}^{(i)} - \mu_j\|_2^2$$

where n is the number of data points. Instead of holding the number of clusters k fixed, your friend John tries to also minimize the objective over k , solving

$$\operatorname{argmin}_k \operatorname{argmin}_{\mu_1, \dots, \mu_k} \sum_{i=1}^N \min_{j \in \{1, \dots, k\}} \|\mathbf{x}^{(i)} - \mu_j\|_2^2$$

You found this idea to be a bad one.

- (a) (1 point) What is the minimum possible value of the objective function when minimizing over k ?

Answer

- (b) (1 point) What is a value of k for which we achieve the minimum possible value of the objective function when $N = 100$?

Answer

4. Consider the following brute-force algorithm for minimizing the k -means objective: Iterate through each possible assignment of the points to k clusters, $\mathbf{z} = [z^{(1)}, \dots, z^{(N)}]$. For each assignment $\mathbf{z} \in \{1, \dots, k\}^N$, you evaluate the following objective function:

$$J(\mathbf{z}) = \operatorname{argmin}_{\mu_1, \dots, \mu_k} \sum_{i=1}^N \|\mathbf{x}^{(i)} - \mu_{z^{(i)}}\|_2^2$$

At the end, you pick the assignment \mathbf{z} that had lowest $J(\mathbf{z})$.

- (a) (1 point) Suppose we have N points and k clusters. For how many possible assignments \mathbf{z} does the brute force algorithm have to evaluate $J(\mathbf{z})$?

Answer

- (b) (1 point) Suppose $N = 1000$, $k = 10$, and it takes us 0.01 seconds to evaluate $J(\mathbf{z})$ for a single assignment \mathbf{z} . How many seconds will the brute force algorithm take to check all assignments?

Answer

5. Initializing the centers has a big impact on the performance of the k -means clustering algorithm. Usually, we randomly initialize k cluster centers. However, there are other methods, namely, furthest point initialization and k -means++ initialization.

- (a) (1 point) **Select one:** Clustering at convergence generated by furthest point initialization is sensitive to outliers. Which of the following statements is correct about this phenomenon?
- ☐ Although outliers will not be selected in the first several iterations, they will temporarily be chosen as centers during training.
 - ☐ Outliers will slow convergence, but will never be centers at convergence time.
 - ☐ Outliers are likely to be selected as centers in the first few iterations.

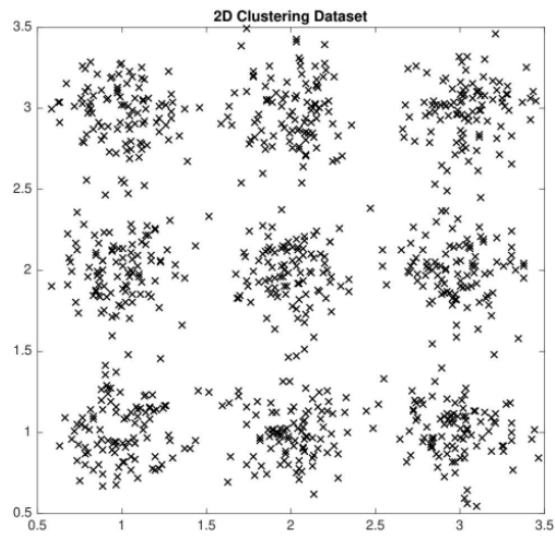


Figure 1: 2D Dataset

- (b) (1 point) **Select one:** Using the dataset in Figure 1 above, compared to random initialization, using k -means++ initialization is _____.
- ☐ more likely to choose one sample from each cluster because centers are chosen with probability proportional to squared distance from existing centers.
 - ☐ less likely to choose one sample from each cluster because the formula does not account for the number of clusters which may be found and thus won't be calibrated to correctly choose one point from each cluster.
 - ☐ equally likely to choose one sample from each cluster because as the number of points grows large, k -means++ asymptotically approaches random initialization.

3 Societal Impacts (15 points)

The fictional country, Xtopia, is in the midst of an epidemic. The Xtopian healthcare system has been under a great deal of strain in the past year due to a regional epidemic caused by an airborne virus called Xvid. The number of hospital beds is limited and as the result, healthcare professionals have to frequently make very difficult choices about which subset of Xvid patients can be hospitalized. Hospital care greatly increases the chance of recovering from the illness with no subsequent long-term health complications.

To save time and make these decisions more efficient and consistent, a team of ML practitioners have been brought in to automate the decision-making process. They have been given access to a data set consisting of the information about prior Xvid patients who sought hospital care along with the binary decision made about them by the hospital doctors ('+' indicates hospitalization and '-' indicates no hospitalization). The ML team has determined that the decision about each patient is highly correlated with his/her age as well as his/her prior utilization of medical insurance. This observation reflects the fact that Xtopian doctors are on average more likely to allocate scarce medical resources to the young and the vulnerable (i.e., those with prior medical conditions and comorbidities). Here, the insurance utilization serves as a proxy for severity of the patient's health conditions.

Figure 2 provides a snapshot of the Xvid training data and the predictive model that the ML team has come up with. Each instance corresponds to an individual patient, and each patient belongs to one of the two socially salient groups in Xtopia, indicated by blue and red.

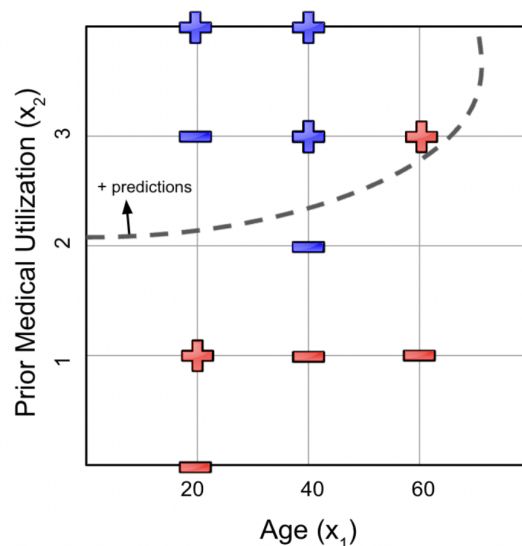


Figure 2: Xvid Training Data

Answer the following questions with respect to the above hypothetical context and data set.

1. (2 points) **Select all that apply:** Does the predictive model above satisfy the following notions of fairness across blue and red groups?

- ☐ False Negative Rate (FNR) parity
- ☐ False Positive Rate (FPR) parity
- ☐ Negative Predictive Value (NPV) parity
- ☐ Positive Predictive Value (PPV) parity
- ☐ Error parity
- ☐ Statistical parity (or Selection rate parity)
- ☐ None of the above

2. (2 points) **Select all that apply:** Which of the above notions of fairness would be satisfied if the ML team could train a model with 0 true error (i.e., a model that always predicts the correct label for every patient)?

- ☐ False Negative Rate (FNR) parity
- ☐ False Positive Rate (FPR) parity
- ☐ Negative Predictive Value (NPV) parity
- ☐ Positive Predictive Value (PPV) parity
- ☐ Error parity
- ☐ Statistical parity (or Selection rate parity)
- ☐ None of the above

3. (2 points) **Select all that apply:** Which of the above notions of fairness would be satisfied in expectation by a random classifier (i.e., a model that makes a randomized prediction for every patient: with probability 0.5 the patient is hospitalized regardless of their attributes)?

- ☐ False Negative Rate (FNR) parity
- ☐ False Positive Rate (FPR) parity
- ☐ Negative Predictive Value (NPV) parity
- ☐ Positive Predictive Value (PPV) parity
- ☐ Error parity
- ☐ Statistical parity (or Selection rate parity)
- ☐ None of the above

4. (2 points) From the perspective of a patient subject to the predictions made by this model, the violation of which of the parity conditions below would be most problematic? Justify your answer.

- ☐ False Negative Rate (FNR) parity
- ☐ False Positive Rate (FPR) parity
- ☐ Negative Predictive Value (NPV) parity
- ☐ Positive Predictive Value (PPV) parity
- ☐ Error parity
- ☐ Statistical parity (or Selection rate parity)
- ☐ None of the above

Your Answer

5. (2 points) **Causes of unfairness:** Name one potential cause of disparity in false negative rates across the two groups in the above context.

Your Answer

6. **Fairness interventions:** consider the following pre-processing method to improve statistical parity:

While the selection rate is unequal across the two groups:

- (a) Pick the group with lowest selection rate.
- (b) From this group in the training data, pick the data point closest to the decision boundary predicted as negative.
- (c) Change the label of this instance to positive.
- (d) Retrain the model on the modified training data by finding the highest accuracy classifier in the hypothesis class.

Suppose our hypothesis class is the class of all linear separators defined over \mathbb{R}^2 .

- (a) (1 point) The highest accuracy linear separator is shown in the figure below (assume that points on the decision boundary are characterized as '+'):

$$(x_1)(x_2)$$

What are the coordinates of the data point whose label would be flipped first by this pre-processing method?

Your Answer

- (b) (2 points) After flipping the label of the point you identified in the previous question, plot the linear separator with the highest accuracy. For your convenience, we have provided a mechanism for you to input your answer in the LaTeX template by specifying the coordinates of two points on the decision boundary.

$$(x_1)(x_2)$$

- (c) (1 point) Using the linear decision boundary you plotted in the previous question, what are the coordinates of the data point whose label would be flipped next by this pre-processing method?

Your Answer

- (d) (1 point) **True or False:** The algorithm terminates at this point.

- ☐ True
- ☐ False

4 Empirical Questions (19 points)

The following questions should be completed after you work through the programming portion of this assignment. **For any plotting questions, you must title your graph, label your axes and provide units (if applicable), and provide a legend (if applicable) in order to receive full credit.**

Please submit computer-generated plots for all parts. We have provided you with the code to generate these plots in the handout.

1. K-means

The following questions should be completed after you work through the programming portion of this assignment. Use `mnist_train.csv` for the following questions.

- (a) (3 points) Run the K-Means algorithm with $K=2$. Include plots below of the cluster centers returned by the K-Means algorithm.

Your Answer

- (b) (3 points) Run the K-Means algorithm with $K=5$. Include plots below of the cluster centers returned by the K-Means algorithm.

Your Answer

- (c) (3 points) Run the K-Means algorithm with $K=10$. Include plots below of the cluster centers returned by the K-Means algorithm.

Your Answer

- (d) (2 points) Compare the performance of the three values of K and discuss which one is optimal and why.

Your Answer

- (e) (3 points) Run the K-Means++ algorithm with $K=5$. Include plots below of the cluster centers returned by the K-Means++ algorithm.

Your Answer

- (f) (3 points) Run the K-Means++ algorithm with $K=10$. Include plots below of the cluster centers returned by the K-Means++ algorithm.

Your Answer

- (g) (2 points) Compare the performance of the K-Means ++ algorithm with the K-Means algorithm using the loss values of both with $K=10$ and explain which algorithm performs better than the other and why.

Your Answer

5 Collaboration Questions

After you have completed all other components of this assignment, report your answers to these questions regarding the collaboration policy. Details of the policy can be found [here](#).

1. Did you receive any help whatsoever from anyone in solving this assignment? If so, include full details.
2. Did you give any help whatsoever to anyone in solving this assignment? If so, include full details.
3. Did you find or come across code that implements any part of this assignment? If so, include full details.

Your Answer

Programming (41 points)

6 The Task

Our goal in this assignment is to implement a K-Means algorithm to cluster images in the MNIST dataset.

7 The Datasets

Dataset We will be using a **subset** of the Modified National Institute of Standards and Technology (MNIST) database. This data includes images of all 10 handwritten digits; our subset will include 3000 of the available images in the dataset. We will evaluate your code on this subset of 3000 samples.

File Format The dataset consists of two csv files. One of these file uses a subset of the 3000 samples given to you. The digits in this subset are restricted to only 0 and 1. Each row contains 784 columns separated by commas. These represent the pixel values of the image.

8 Model Definition

In this assignment you will implement the K-means function to perform K-means clustering on the MNIST dataset. You will also implement the K-Means variant, K-Means++ on the same dataset. The input to your dataset will be N 784-dimensional data points and your output will be the cluster assignments of each of the data points in your training input.

Your code should be able to run with K-Means or K-Means++ invariably on any dataset we pass in according to the algorithm flag passed into the command line when your code is run. Furthermore, you should be able to assign cluster centers for any positive nonzero value of K which is passed in.

For K-Means, you will be asked to initialize the cluster centers to K random points. It is critical that in order for your code to run successfully, you do not change the random seed defined at the top of the file `kmeans.py`. For K-Means++, you must randomly assign your first cluster center, and then use the definition of the K-Means++ algorithm to define the subsequent K-1 cluster centers.

8.1 Command Line Arguments

The autograder runs and evaluates the output from the files generated, using the following command:

```
$ python3 kmeans.py [args...]
```

Where above `[args...]` is a placeholder for four command-line arguments: `<train_input>` `<K>` `<algorithm>` `<train_output>`. These arguments are described in detail below:

1. `<train_input>`: path to the training input `.csv` file (see Section 7)
2. `<K>`: integer specifying the number of cluster centers which should be initialized.
3. `<algorithm>`: integer taking value 0 or 1 that specifies whether to use K-MEANS or K-MEANS++—that is, if `algorithm==0` use K-Means, and if `algorithm==1` use K-Means++ initialization.
4. `<train_output>`: cluster assignments for all data points in `train_input`.

As an example, the following command line would run your program with `K=2` on the small data provided in the handout using K-Means.

```
python3 kmeans.py mnist_small_train.csv 2 0 small_train_out.txt
```

The command line arguments are parsed for you in `kmeans.py` using the Python builtin `argparse` package.

8.2 Sample Outputs

In the handout file we have also provided you with sample outputs for the small dataset. These files correspond to the expected output for the small dataset with $K=2$ and $K=5$ as well as `algorithm=0` and `algorithm=1`. Although you will not be required to output the final cluster centers after your k-means algorithm terminates, we have included output files containing the final cluster centers for the four settings described above to help you debug your code.

8.3 Gradescope Submission

You should only submit your `kmeans.py` file to Gradescope. Please do not use any other file name for your implementation. This will cause problems for the autograder to correctly detect and run your code.

Note: For this assignment, you may make up to **10** submissions to Gradescope before the deadline, but only your last submission will be graded.