VMLS - Study Guide Chapter 4 Name: Ben Chen

Always include this title page with your PDF. Include your name above.

- Submit your work in Gradescope as a PDF - you will identify where your “questions are.”
- Identify the question number as you submit. Since we grade "blind" if the questions are NOT identified, the work WILL NOT BE GRADED and a 0 will be recorded. Always leave enough time to identify the questions when submitting.
- One section per page (if a page or less) - We prefer to grade the main solution in a single page, extra work can be included on the following page.
- Long instructions may be removed to fit on a single page.
- **Do not start a new question in the middle of a page.**
- Solutions to book questions are provided for reference.
- You may NOT submit given solutions - this includes minor modifications - as your own.
- Solutions that do not show individual engagement with the solutions will be marked as no credit and can be considered a violation of honor code.
- If you use the given solutions you must reference or explain how you used them, in particular...

For full credit, EACH book exercise in the Study Guides must use one or more of the following methods and FOR EACH QUESTION. Identify the number the method by number to ensure full credit.

Method 1 - Provide original examples which demonstrate the ideas of the exercise in addition to your solution.

Method 2 - Include and discuss the specific topics needed from the chapter and how they relate to the question.

Method 3 - Include original Python code, of reasonable length (as screenshot r text) to show how the topic or concept was explored.

Method 4 - Expand the given solution in a significant way, with additional steps and comments. All steps are justified. This is a good method for a proof for which you are only given a basic outline.

Method 5 - Attempt the exercise without looking at the solution and then the solution is used to check work. Words are used to describe the results.

Method 6 - Provide an analysis of the strategies used to understand the exercise, describing in detail what was challenging, who helped you or what resources were used. The process of understanding is described.

Method #5

1. (20pts) Describe the K-Means algorithm in your own words to someone who has never heard of it. Explain each step.

K means algorithm is an algorithm in unsupervised machine learning that groups together data points in a dataset without having prior labels. The clustering method involves organizing the data into k-number of clusters. The data points inside the cluster are optimized to be as similar as possible while the clusters are optimized to be as different as possible. We can use K means to determine how similar different targets are to each other.

We do this by finding the centroid (the center of each cluster) and assigning data points to a cluster based on how near they are with the centroid. Typically, you just pick some arbitrary number k, and randomize initialization selecting k points from your data to serve as initial centroids. Each data point is calculating the distance in Euclidean space to each centroid and assigns the datapoint a cluster (nearest the centroid). Then after assigning all points, it recaculates the centroid of each cluster and the new centroid is the mean of all the data points assigned to the cluster. We then repeat this until centroids don't significantly change or the assignments of the data points remain relatively stable.

This algorithm is straightforward, and works well with large datasets because computation costs per iteration (assigning new data points/centroids) are linear with respect to the number of the data points and runs quite fast in practice as well.

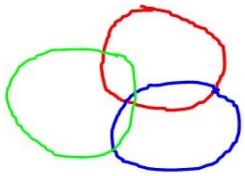
The objective is to minimize the sum of squared distances within each cluster, and requires the number of clusters to be set in advance. The number of clusters can be optimized as well by using techniques like the elbow method or silhouette analysis.

In modern industry, we can see clustering used in things like recommendation systems (like facebook ads, amazon carousel suggestion, etc.)

Method #5

2. (20 pts) Select one page or section of Chapter One of VMLS to annotate. Include a screenshot of your annotation here. (not example 4.4.1 since that is the next question)

The clusters look like -> but much more spread out for the sake of the example



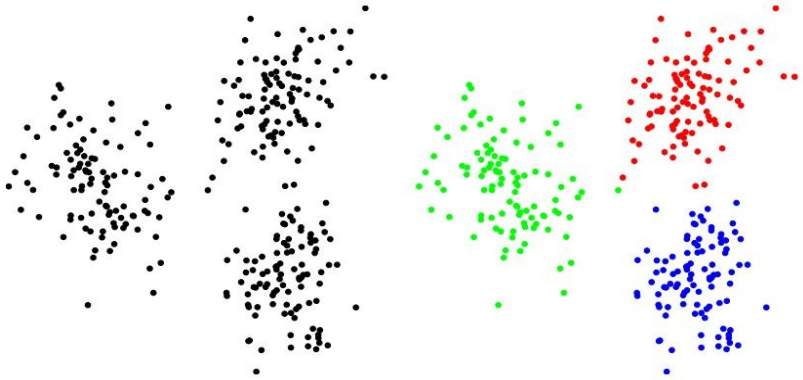


Figure 4.1 300 points in a plane. The points can be clustered in the three groups shown on the right.

Identifying the # of clusters by visually observing is a total vibe check and not at all scientific!

and check visually if the data are clustered, and if so, how many clusters there are. In almost all applications n is larger than 2 (and typically, much larger than 2), in which case this simple visual method cannot be used. The second way in which it is not typical is that the points are very well clustered. In most applications, the data are not as cleanly clustered as in this simple example; there are several or even many points that lie in between clusters. Finally, in this example, it is clear that the best choice of k is $k = 3$. In real examples, it can be less clear what the best value of k is. But even when the clustering is not as clean as in this example, and the best value of k is not clear, clustering can be very useful in practice.

"vibe check"

Examples. Before we delve more deeply into the details of clustering and clustering algorithms, we list some common applications where clustering is used.

topic, genre, and author are the features of the algorithm to identify who is close to centroid

- *Topic discovery.* Suppose x_i are word histograms associated with N documents. A clustering algorithm partitions the documents into k groups, which typically can be interpreted as groups of documents with the same or similar topics, genre, or author. Since the clustering algorithm runs automatically and without any understanding of what the words in the dictionary mean, this is sometimes called *automatic topic discovery.*

USES

BUT it's not explicitly hard-coded into the algorithm, as in, it can do all of this mathematically without knowing context!

- *Patient clustering.* If x_i are feature vectors associated with N patients admitted to a hospital, a clustering algorithm clusters the patients into k groups of similar patients (at least in terms of their feature vectors).

- *Customer market segmentation.* Suppose the vector x_i gives the quantities (or dollar values) of n items purchased by customer i over some period of time. A clustering algorithm will group the customers into k market segments, which are groups of customers with similar purchasing patterns.

If there is excessive k-number clusters, that's "over-segmentation"

Method #5

3. (20 pts) Explain example 4.4.1 in detail. Why is this an interesting example? What is K in this example?

4.4.1 Image clustering

The MNIST (Mixed National Institute of Standards) database of handwritten digits is a data set containing $N = 60000$ grayscale images of size 28×28, which we represent as n-vectors with $n = 28 \times 28 = 784$. Figure 4.6 shows a few examples from the data set. (The data set is available from Yann LeCun at `yann.lecun.com/exdb/mnist`.)

We use the k-means algorithm to partition these images into $k = 20$ clusters, starting with a random assignment of the vectors to groups, and repeating the experiment 20 times. Figure 4.7 shows the clustering objective versus iteration number for three of the 20 initial assignments, including the two that gave the lowest and the highest final values of the objective.

Figure 4.8 shows the representatives with the lowest final value of the clustering objective. Figure 4.9 shows the set with the highest value. We can see that most of the representatives are recognizable digits, with some reasonable confusion, for example between '4' and '9' or '3' and '8'. This is impressive when you consider that the k-means algorithm knows nothing about digits, handwriting, or even that the 784-vectors represent 28×28 images; it uses only the distances between 784-vectors. One interpretation is that the k-means algorithm has 'discovered' the digits in the data set.

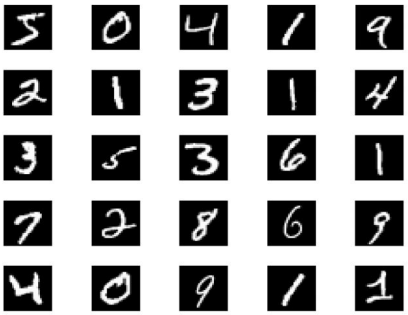


Figure 4.6 25 images of handwritten digits from the MNIST data set. Each image has size 28×28, and can be represented by a 784-vector.

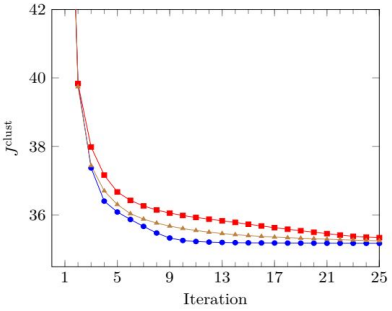


Figure 4.7 Clustering objective J^{clust} after each iteration of the k-means algorithm, for three initial partitions, on digits of the MNIST set.

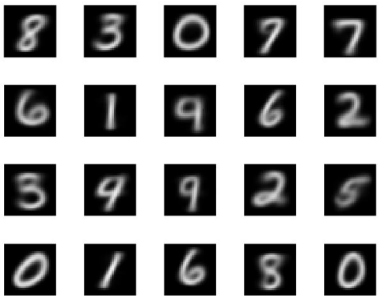


Figure 4.8 Group representatives found by the k-means algorithm applied to the MNIST set.

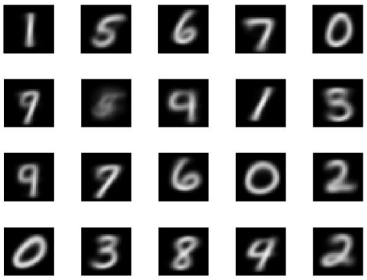


Figure 4.9 Group representatives found by the k-means algorithm applied to the MNIST set, with a different starting point than in figure 4.8.

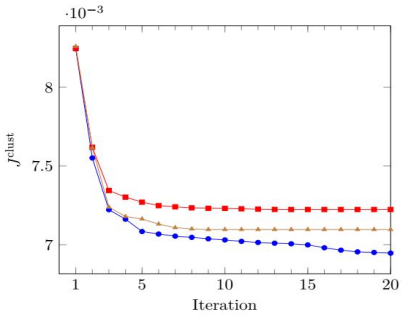


Figure 4.10 Clustering objective J^{clust} after each iteration of the k-means algorithm, for three initial partitions, on Wikipedia word count histograms.

This was Yann LeCun's famous experiment that was was published at IEEE (big fan) was discovering gradient-based learning applied to document recognition and put him on the map. The reason why this experiment was so interesting and groundbreaking at the time was because the K-means algorithm basically associated different numbers and digits purely from the distances among the images. The 784 dimensional image vectors clustered in accordance without any human labeling the data and patterns. K was the number of clusters, and it was 20 clusters.

This was all done in the late 1980-1990s when deep learning wasn't even a real thing yet. GPUs were not invented at the time. So this was done prior to anything like transformers or mixture of experts or any chain of thought that we see in modern industry.

Method #5

4. (20 pts) Explain the solution to 4.1 here in your own words. (Since you are given a solution, you will be graded on your ability to explain).

4.1 *Minimizing mean square distance to a set of vectors.* Let $x_1, \ldots, x_L$ be a collection of n-vectors. In this exercise you will fill in the missing parts of the argument to show that the vector z which minimizes the sum-square distance to the vectors,

$$J(z) = \|x_1 - z\|^2 + \cdots + \|x_L - z\|^2,$$

is the average or centroid of the vectors, $\overline{x} = (1/L)\,(x_1 + \cdots + x_L)$. (This result is used in one of the steps in the k-means algorithm. But here we have simplified the notation.)

(a) Explain why, for any z, we have

$$J(z) = \sum_{i=1}^{L} \|x_i - \overline{x} - (z - \overline{x})\|^2 = \sum_{i=1}^{L} \left(\|x_i - \overline{x}\|^2 - 2(x_i - \overline{x})^T(z - \overline{x})\right) + L\|z - \overline{x}\|^2.$$

(b) Explain why $\sum_{i=1}^{L}(x_i - \overline{x})^T(z - \overline{x}) = 0$. *Hint.* Write the left-hand side as

$$\left(\sum_{i=1}^{L}(x_i - \overline{x})\right)^T (z - \overline{x}),$$

and argue that the left-hand vector is 0.

(c) Combine the results of (a) and (b) to get $J(z) = \sum_{i=1}^{L} \|x_i - \overline{x}\|^2 + L\|z - \overline{x}\|^2$. Explain why for any $z \neq \overline{x}$, we have $J(z) > J(\overline{x})$. This shows that the choice $z = \overline{x}$ minimizes $J(z)$.

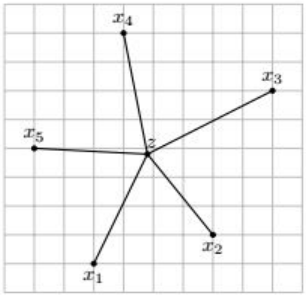
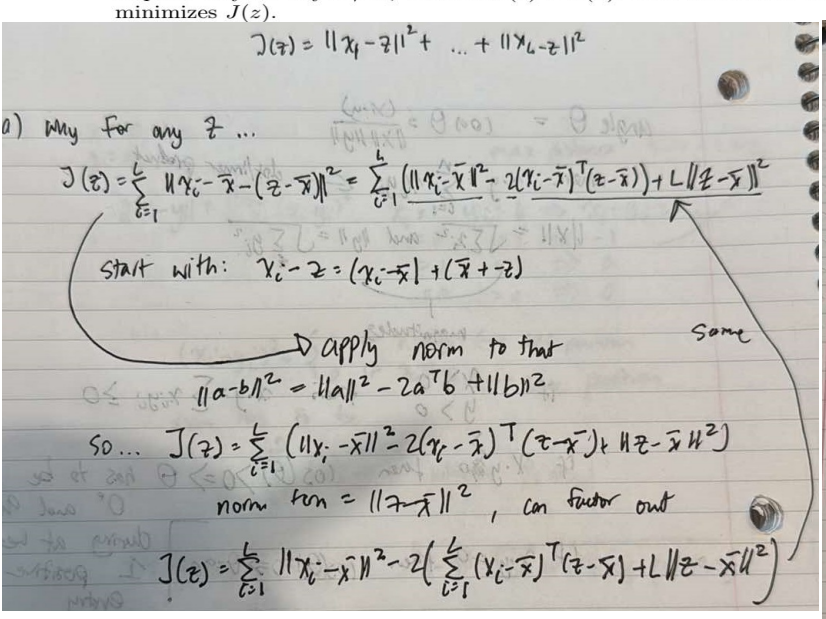
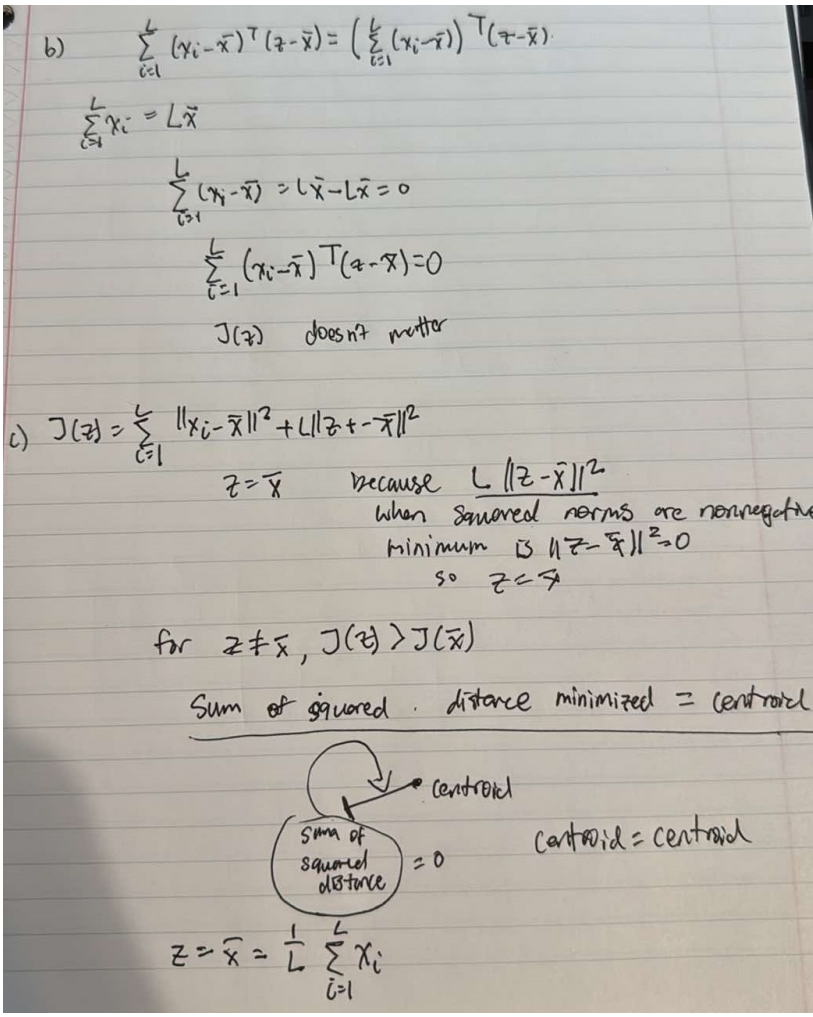


Figure 4.1 The vector z is the mean or centroid of the vectors $x_1, \ldots, x_5$. It minimizes the mean square distance to the points.





x1, x2, etc. are the n-dimensional vectors (points in the vector space)

if n=2 for example, then xi should be a 2d point like (1, 0), and if n=3, then xi should be a 3d point like (1, 0, 1) or something

L is the total number of vectors (or data points), 5 vectors => L=5

z is an arbitrary n-dimensional vector we are trying to find to minimize the function J(z) to show that the best choice for z is the centroid of all vectors

x-bar is the mean (centroid) of the given vectors defined as 1/L(the sum of xi from i=1 iterating L times)

J(z) is the sum of squared distances from each vector xi to z, and we want to minimize this function to find the best choice for z that is as close as possible to all vectors

a) is about breaking down the formula for distance, by looking at the total squared distance from all the points xi to z and instead of thinking about the distance from xi to z directly, we rearrange the formula to break it into parts: one part for how far each point xi is from the average x-bar, and then how far the chosen point z is from the average x-bar
b) is showing that the middle term is zero because if we add up all the differences between the points and their average x-bar, we're supposed to get zero, which means the average is the center of the points, and positive and negatives differences cancel out
c) the first term is constant, no matter what z we pick, L||z+x-bar||^2 depends on how far z is from average x-bar, distance is always positive squared, so smallest possible value = 0. it's 0 when the z is the centroid

Method #5

5. (20 pts) Explain the solution to 4.2 here in your own words. (Since you are given a solution, you will be graded on your ability to explain).

4.2 *k-means with nonnegative, proportions, or Boolean vectors.* Suppose that the vectors $x_1, \ldots, x_N$ are clustered using k-means, with group representatives $z_1, \ldots, z_k$.

(a) Suppose the original vectors x_i are nonnegative, *i.e.*, their entries are nonnegative. Explain why the representatives z_j are also nonnegative.

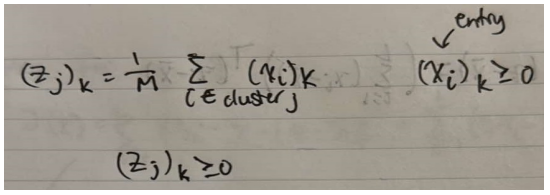
(b) Suppose the original vectors x_i represent proportions, *i.e.*, their entries are nonnegative and sum to one. (This is the case when x_i are word count histograms, for example.) Explain why the representatives z_j also represent proportions, *i.e.*, their entries are nonnegative and sum to one.

(c) Suppose the original vectors x_i are Boolean, *i.e.*, their entries are either 0 or 1. Give an interpretation of $(z_j)_i$, the ith entry of the j group representative.

Hint. Each representative is the average of some of the original vectors.

a) We assume the original vectors xi are nonnegative which means that each entry inside the vector is going to be >= 0

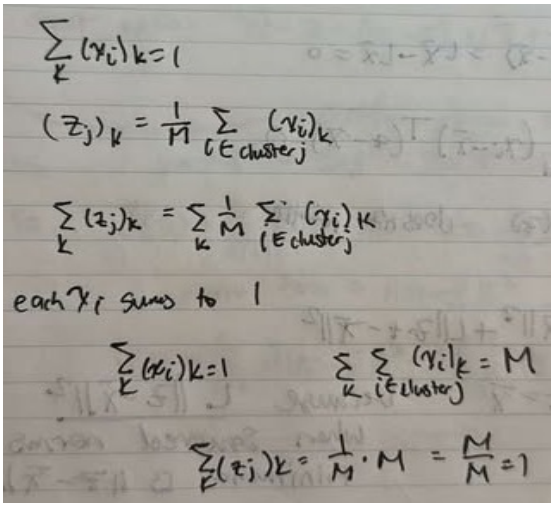
- In k means, the representative z sub-j of a cluster is the average of all the vectors assigned to the cluster where big M is the # of the vectors in the cluster
- Since each entry >= 0, their average must also be non-negative



So if all original vectors contain only nonnegative values, their average must also be non negative

b) suppose the original vectors xi represent proportions where each xi has nonnegative entries, and each xi sum to one

- This means that k means computes each cluster representative as the average of all vectors assigned to that cluster… the sum of proportions is preserved under averaging so that the k means representatives also sum to one like a bell-curve



c) k means representative z sub-j is still the average of all vectors in the cluster, since each x sub-i is either 0 or 1, the sum of this counts how many points have a 1 in the k-th position so (z sub-j)sub-k is a fraction of the vectors in the cluster that have a 1 in position k

- If (z sub-j)sub-k = 1, all vectors in the cluster had a 1 in that position
- If (z sub-j)sub-k = 0.4, then 40% of the vectors in the cluster had a 1 at that position
- (z sub-j)sub-k = 0, then 0% of vectors had 1

This tells the story that the ratio of data points in the cluster that had a 1 at position, which is probably some kind of image classification problem or voting problem that is binary