Machine Learning: Assignment Sheet #10

Due on April 26, 2022 at 10:00

Group HB

Henri Sota, Enis Mustafaj

Problem 10.1

In this task, you carry out K-means clustering with $K = 3$ by paper and pencil. To this end, you are given the following data

i	x_i	$C^{(0)}(i)$
1	$(15, 7)^\top$	2
2	$(0, -9)^\top$	1
3	$(-2, -5)^\top$	1
4	$(2, 3)^\top$	0
5	$(3, 7)^\top$	2
6	$(18, 12)^\top$	0

with its initial cluster assignment. Let the clustering algorithm "run" until it finalized its assignment. Finally, draw the just computed clusters in a two-dimensional scatter plot.

Recalling Algorithm 14, K-means clustering:
Input: input data $\{(\boldsymbol{x}_i)\}_{i=1}^N(\boldsymbol{x}_i \in \mathbb{R}^D)$, cluster count K
Output: cluster assignment C^*, centroids $\{\bar{\boldsymbol{x}}_k^*\}_{k=1}^K$

```
1. choose initial guess $C^{(0)}$
2. $s \leftarrow 0$
3. repeat:
    a) $s \leftarrow s + 1$
    b) $N_k \leftarrow |\{i|C^{(s-1)}(i) = k\}|, \quad k = 1, \dots, K$
    c) $\boldsymbol{m}_k^{(s)} \leftarrow \frac{1}{N_k}\sum_{C^{(s-1)}(i)=k} \boldsymbol{x}_i, \quad k = 1, \dots, K$    *(compute centroids)*
    d) $C^{(s)}(i) \leftarrow \arg\min_{1 \le k \le K} \left\| \boldsymbol{x}_i - \boldsymbol{m}_k^{(s)} \right\|^2, \quad i = 1, \dots, N$    *(cluster samples around closest centroid)*
   until $C^{(s)} = C^{(s-1)}$
4. $\bar{\boldsymbol{x}}_k^* \leftarrow \frac{1}{|\{i|C^{(s-1)}(i)=k\}|}\sum_{C^{(s)}(i)=k} \boldsymbol{x}_i, \quad k = 1, \dots, K$    *(compute final centroids)*
5. return $C^*, \{\bar{\boldsymbol{x}}_k^*\}_{k=1}^K$
```

With the initial cluster assignment given, we start iterating:
Iteration 1:

a) $s \leftarrow s + 1 = 1$

b) Cluster sizes with the data points based on index: $N_1 = 2(\{4, 6\}), N_2 = 2(\{2, 3\}), N_3 = 2(\{1, 5\})$

c) Calculating centroids for each cluster:

- $\boldsymbol{m}_1^{(1)} = \frac{1}{2}((2, 3)^\top + (18, 12)^\top) = (10, 7.5)^\top$
- $\boldsymbol{m}_2^{(1)} = \frac{1}{2}((0, -9)^\top + (-2, -5)^\top) = (-1, -7)^\top$
- $\boldsymbol{m}_3^{(1)} = \frac{1}{2}((15, 7)^\top + (3, 7)^\top) = (9, 7)^\top$

d) Cluster the samples based on the closest centroid:

- $C^{(1)}(1) = \arg\min_{1\le k\le K}\{5.02493781, 21.26029163, 6\} = 0$
- $C^{(1)}(2) = \arg\min_{1\le k\le K}\{19.29378138, 2.23606798, 18.35755975\} = 1$
- $C^{(1)}(3) = \arg\min_{1\le k\le K}\{17.32772345, 2.23606798, 16.2788206\} = 1$
- $C^{(1)}(4) = \arg\min_{1\le k\le K}\{9.17877988, 10.44030651, 8.06225775\} = 2$
- $C^{(1)}(5) = \arg\min_{1\le k\le K}\{7.01783442, 14.56021978, 6\} = 2$
- $C^{(1)}(6) = \arg\min_{1\le k\le K}\{9.17877988, 26.87005769, 10.29563014\} = 0$

C^1 is not the same as C^0. Continue iteration.
Iteration 2:

a) $s \leftarrow s + 1 = 2$

b) Cluster sizes with the data points based on index: $N_1 = 2(\{1,6\}), N_2 = 2(\{2,3\}), N_3 = 2(\{4,5\})$

c) Calculating centroids for each cluster:

- $\boldsymbol{m}_1^{(2)} = \frac{1}{2}((15,7)^\top + (18,12)^\top) = (16.5, 9.5)^\top$
- $\boldsymbol{m}_2^{(2)} = \frac{1}{2}((0,-9)^\top + (-2,-5)^\top) = (-1,-7)^\top$
- $\boldsymbol{m}_3^{(2)} = \frac{1}{2}((2,3)^\top + (3,7)^\top) = (2.5, 5)^\top$

d) Cluster the samples based on the closest centroid:

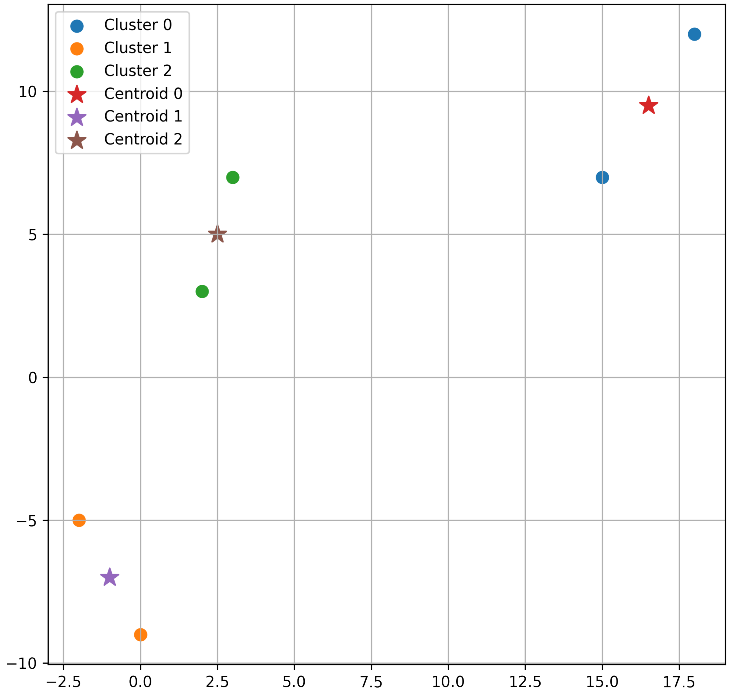
- $C^{(2)}(1) = \arg\min_{1\le k\le K}\{2.91547595, 21.26029163, 12.658988\} = 0$
- $C^{(2)}(2) = \arg\min_{1\le k\le K}\{24.78911051, 2.23606798, 14.22146265\} = 1$
- $C^{(2)}(3) = \arg\min_{1\le k\le K}\{23.50531855, 2.23606798, 10.9658561\} = 1$
- $C^{(2)}(4) = \arg\min_{1\le k\le K}\{15.89024858, 10.44030651, 2.06155281\} = 2$
- $C^{(2)}(5) = \arg\min_{1\le k\le K}\{13.72953022, 14.56021978, 2.06155281\} = 2$
- $C^{(2)}(6) = \arg\min_{1\le k\le K}\{2.91547595, 26.87005769, 17.00735135\} = 0$

C^2 is the same as C^1. Stop iteration.

Calculating the final centroids:

- $\bar{\boldsymbol{x}}_1^* = \frac{1}{2}((15,7)^\top + (18,12)^\top) = (16.5, 9.5)^\top$
- $\bar{\boldsymbol{x}}_2^* = \frac{1}{2}((0,-9)^\top + (-2,-5)^\top) = (-1,-7)^\top$
- $\bar{\boldsymbol{x}}_3^* = \frac{1}{2}((2,3)^\top + (3,7)^\top) = (2.5, 5)^\top$

The computed clusters can be seen on the scatter plot below.



Problem 10.2

You are given the data set

$$\{(3,2)^\top, (-1,0)^\top, (7,1)^\top\}$$

Manually compute the principle components of this data set by solving the associated eigenvalue problem. Recalling Theorem 9.2 for finding the solution of the optimization problem:

Let the setting of Theorem 9.1 be given. The solution $\hat{\mathcal{V}}_d$ of the optimization problem is the matrix composed of the eigenvectors for the d largest eigenvalues of the matrix

$$A = \mathcal{X}^T \mathcal{X}$$

where $\mathcal{X} \in \mathbb{R}^{N \times D}$ is the matrix created from the data as $\mathcal{X} = (\boldsymbol{x}_1 | \dots | \boldsymbol{x}_N)^\top$.
Based on Theorem 9.1, the data needs to be centered. The mean of the given data set is $(3,1)^\top$. The centered data set is then:

$$\{(0,1)^\top, (-4,-1)^\top, (4,0)^\top\}$$

Calculating A by performing the matrix multiplication of the transpose of the data set and the data set itself:

$$A = \begin{pmatrix} 0 & -4 & 4 \\ 1 & -1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -4 & -1 \\ 4 & 0 \end{pmatrix} = \begin{pmatrix} 32 & 4 \\ 4 & 2 \end{pmatrix}$$

Finding the eigenvalues:

$$\det(A - \lambda I) = \begin{vmatrix} 32-\lambda & 4 \\ 4 & 2-\lambda \end{vmatrix} = \lambda^2 - 34\lambda + 48 = (\lambda + \sqrt{241} - 17)(\lambda - \sqrt{241} + 17) = 0$$

The eigenvalues correspond to the solutions to the equation above. The eigenvalues are: $\lambda_1 = -\sqrt{241} + 17 \approx 1.4758253$, $\lambda_2 = \sqrt{241} + 17 \approx 32.5241747$

The eigenvectors corresponding to the eigenvalues are the principle components.

For $\lambda_1 = -\sqrt{241} + 17$:

$$(A - \lambda_1 I) \cdot \boldsymbol{v} = \begin{pmatrix} \sqrt{241} + 15 & 4 \\ 4 & \sqrt{241} - 15 \end{pmatrix} \cdot \boldsymbol{v} = 0$$

The solution set is represented by the eigenvector: $\begin{pmatrix} \frac{-\sqrt{241}+15}{4} \\ 1 \end{pmatrix}$

For $\lambda_2 = \sqrt{241} + 17$:

$$(A - \lambda_2 I) \cdot \boldsymbol{v} = \begin{pmatrix} -\sqrt{241} + 15 & 4 \\ 4 & -\sqrt{241} - 15 \end{pmatrix} \cdot \boldsymbol{v} = 0$$

The solution set is represented by the eigenvector: $\begin{pmatrix} \frac{\sqrt{241}+15}{4} \\ 1 \end{pmatrix}$

Problem 10.3

A well-known application of principle component analysis is lossy data compression. In this application, you are given a large data set $\{\mathbf{x}\}_{i=1}^N$ with $\mathbf{x}_i \in \mathbb{R}^D$ and reduce it to a data set $\{\bar{\mathbf{x}}\}_{i=1}^N$ with $\bar{\mathbf{x}}_i \in \mathbb{R}^d$ where $d < D$, while storing a matrix that allows to reconstruct an approximation to the $\mathbf{x}_i$ from the vectors $\bar{\mathbf{x}}_i$.

Develop and give a compression and a decompression algorithm which carry out the above described data compression and decompression tasks by using principle component analysis. You can either try to develop the idea by yourself or do a research in the internet. In the latter case, please quote the source from which you took the information.

The followings are the algorithms to compress and decompress a set of data $\{\mathbf{x}\}_{i=1}^N$:

```
Input: input data $\{\mathbf{x}\}_{i=1}^N$, nr_components
Output: compressed data $\{\bar{\mathbf{x}}_i\}_{i=1}^N$
  1) $Mean \leftarrow mean(data)$
  2) $Zero\_Centered \leftarrow data - Meanx$
  3) $Cov\_Matrix \leftarrow (Zero\_Centered^T \cdot Zero\_Centered)/(length(data) - 1)$
  4) $eig\_values, eig\_vectors \leftarrow eig(Cov\_Matrix)$
  5) $selected\_components \leftarrow eig\_vectors[:, : nr\_components]$
  6) $compressed\_data \leftarrow selected\_components^T \cdot data^T$
  7) return $compressed\_data^T$
```

```
Input: compressed data $\{\bar{\mathbf{x}}_i\}_{i=1}^N$
Output: original data $\{\mathbf{x}_i\}_{i=1}^N$
  1) $decompressed\_data \leftarrow compressed\_data \cdot selected\_components^T$
  2) return $decompressed\_data$
```

Programming Problem 10.1

In this task, you are supposed to apply your just developed data compression algorithm to the MNIST data set. To this end, consult again Example 9.4, which provides access to the hand-written digits for the value 8 in MNIST. Compress the images of these digits to a dimensionality of $d = 784, 512, 256, 128, 64, 32$, decompress them again, and plot the decompressed digits.
The implementation of the above algorithms can be found in the file `programming_exercises.ipynb`.