DD2434 — Machine learning, advanced course

KTH Royal Institute of Technology

Assignment #1

fall 2020

Due: Dec 1, 2020

You should return your answers in Canvas as a single PDF file, which should contain a link to a git directory with all the scripts you have used to develop the programming task. You may use a personal git or the KTH GitHub services.

For the programming task, you may use the programming language of your choice. You may use existing libraries for computing eigen-decomposition, SVD, and PCA, but you are expected to implement MDS and Isomap by your own. Your code should contain a README file with simple instructions (ideally running one script) on how to reproduce the plots that are included in the report.

You should type your report in a high-quality text processor (LaTeX is highly recommended). You should submit your report as a PDF file. File formats such as docx, odt, md, html, txt, tex, png, jpg, etc., will not be accepted. Scans or photographs from handwritten text will not be accepted, even if converted to PDF.

This is an individual assignment, you should not discuss your solution with others.

There is a total of 100 points. The cut-offs for letter grades D and E are 80 and 60, respectively.

Problem 1 [10 points]

Let $\mathbf{A}$ be a real $n \times n$ symmetric matrix. We say that $\mathbf{A}$ is *positive semidefinite* if and only if $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0$, for every vector $\mathbf{x} \in \mathbb{R}^n$. Prove the following statements, which were claimed in the video lectures.

(i) Prove that a real symmetric matrix has real eigenvalues (the point being that a general matrix may have complex eigenvalues).

(ii) Prove that a real symmetric matrix has orthogonal eigenvectors. Then, prove that the eigen-decomposition of a real symmetric matrix $\mathbf{A}$ is $\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$.

(iii) Prove that a positive semidefinite matrix has non-negative eigenvalues.

(iv) Let $\mathbf{A}$ be a positive semidefinite matrix. Define matrix $\mathbf{D}$ so that $\mathbf{D}_{ij} = \mathbf{A}_{ii} + \mathbf{A}_{jj} - 2\mathbf{A}_{ij}$. Show that there exist n vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ in $\mathbb{R}^n$ so that $\mathbf{D}_{ij} = \|\mathbf{v}_i - \mathbf{v}_j\|_2^2$.

Problem 2 [5 points]

Consider a data matrix of dimension $m \times n$. In some applications the role of points and dimensions can be interchanged. For example, given a document corpus represented as a matrix of type "documents × words", we may want to analyze documents based on which words occur in them, or we may want to analyze words based on which documents they appear in. So it is meaningful to perform PCA both with respect to the rows of a matrix and with respect to its columns.

As we discussed in the video lectures, PCA relies on SVD. Moreover, since $(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T)^T = \mathbf{V}\mathbf{\Sigma}^T\mathbf{U}^T = \mathbf{V}\mathbf{\Sigma}'\mathbf{U}^T$, where $\mathbf{\Sigma}'$ differs from $\mathbf{\Sigma}$ only in terms of size, performing SVD on a matrix gives also the SVD on its transpose.

Does this argument imply that a **single** SVD operation is sufficient to perform PCA both on the rows and the columns of a data matrix?

Justify your answer.

Problem 3 [15 points]

In the derivation of PCA we asked to maximize the expression $\text{tr}(\mathbf{Y}^T\mathbf{W}\mathbf{W}^T\mathbf{Y})$, with respect to a matrix $\mathbf{W}$ having k columns, given $\mathbf{Y}$. We claimed that (i) the maximizer is given by $\mathbf{W} = \mathbf{U}_k$, that is, the k left singular vectors of $\mathbf{Y}$ associated with the k largest singular values; and (ii) the maximum value achieved is $\sum_{i=1}^{k} \sigma_i^2$.

Prove claims (i) and (ii).

Problem 4 [15 points]

In the derivation of classical MDS with distance matrix, our goal was to derive the Gram matrix (similarity matrix) $\mathbf{S} = \mathbf{Y}^T\mathbf{Y}$ from the distance matrix $\mathbf{D}$, while $\mathbf{Y}$ is unknown.

We get $s_{ij} = -\frac{1}{2}(d_{ij}^2 - s_{ii} - s_{jj})$.[1]

We claim that s_{ij} can be computed as $s_{ij} = -\frac{1}{2}(d_{ij}^2 - d_{1i}^2 - d_{1j}^2)$, where d_{1i} and d_{1j} are the distances from the first point in the dataset to points i and j, respectively.

Argue that the claim is correct, that is, it provides the correct estimation for the Gram matrix $\mathbf{S}$.

Problem 5 [10 points]

Consider the classical MDS algorithm when $\mathbf{Y}$ is known. In that case, we form $\mathbf{S} = \mathbf{Y}^T\mathbf{Y}$ and obtain the MDS embedding by the eigen-decomposition of $\mathbf{S}$. Show that this procedure is equivalent to performing PCA on $\mathbf{Y}$.

In terms of computation, which is the best way to perform the embedding?

Problem 6 [5 points]

Argue that the process to obtain the neiborhood graph G in the Isomap method may yield a disconnected graph. Propose a heuristic to patch this problem. Justify your heuristic.

[1] Note that there is a typo in the slides, the similarities s_{ii}, s_{jj} should be without a square, as $s_{ii} = \mathbf{y}_i^T\mathbf{y}_i$ and $s_{jj} = \mathbf{y}_j^T\mathbf{y}_j$. Typo corrected in the slides uploaded in canvas, but not in the video.

Problem 7 (programming task) [40 points]

We want to visualize a small set of animal species. We will use the "zoo" dataset in the UCI ML repository: `https://archive.ics.uci.edu/ml/datasets/zoo`, which contains information about a few different attributes of each species. We want to project the data in 2D so that "similar" animals are projected near to each other. The last attribute "type" can be removed from the dataset, and used as color for each point in the visualization. Since the dataset is fairly small, you may also want to annotate each point in the projection.

Note that all attributes are Boolean, except one, so first you will need to decide how to handle it.

You are asked to experiment with the following methods:

(i) PCA.

(ii) MDS, where you can try to infer the importance of different attributes, compute pair-wise distances between the species taking into account the attribute importance, and apply MDS on the resulting distance matrix.

(iii) Isomap, where you should experiment with the number of nearest-neighbor parameter used to form the neighborhood graph.

Describe what you have implemented and justify the choices you have made. Plot your results. Write down your observations and conclusions. Which method is preferable?