DD2434/FDD3434 Machine Learning, Advanced Course
Assignment 1, 2021

Aristides Gionis

Deadline, see Canvas

Read before starting

Please read the assignment questions carefully before starting working on the solutions.

You are allowed to discuss the assignment questions with others, but not the solutions — you must solve the assignment individually. Your report will be automatically checked for similarities to other students' solutions as well as documents on the web in general. Please make a note of the people you have discussed the problems with. Please also cite other resources, such as textbooks or websites, that you may have used while developing your solutions.

Your report should be submitted before the deadline using Canvas. Write clearly, stating all the assumptions you have made, and explain your logical steps and derivations. Show the results of your experiments using images and graphs together with your analysis and add your code as an appendix. You should be prepared to share your code, bundled as easy-to-run scripts that can be used to generate your results, if asked.

Being able to communicate results and conclusions is a key aspect of scientific and corporate activities. It is up to you as an author to make sure that your report is well-written, precise, and self-contained. Based on the report, and only the report, we will decide if you pass the task. No detective work should be required on our side. In particular, please return neat and tidy reports!

The grading of the assignment will be as follows,

E Correctly completed (**Five out of** the questions 1.1.1 to 1.3.10) **and** 1.4.11 **and** 1.4.12.

D Correctly completed (**Seven out of** the questions 1.1.1 to 1.3.10) **and** 1.4.

Good Luck!

1.1 Principal Component Analysis

While developing the PCA method, we required that the data are "centered." This step is performed by subtracting the expectation of the data from each data point. Essentially, with this step, we translate the center of mass of the data to the origin of the Euclidean space.

Question 1.1.1: *Explain why this data-centering step is required while performing PCA. What could be an undesirable effect if we perform PCA on non-centered data?*

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 $\times$ 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 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.

Question 1.1.2: *Does the previous 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.

While developing the PCA method, we consider that each data point $\mathbf{y} \in \mathbb{R}^d$ is generated by a latent vector $\mathbf{x} \in \mathbb{R}^k$, with $k < d$, through a linear transformation

$$\mathbf{y} = \mathbf{W}\,\mathbf{x}, \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{d \times k}$ is a matrix with orthonormal columns. We then deduced that the inverse map is obtained by

$$\mathbf{x} = \mathbf{W}^+\,\mathbf{y}, \tag{2}$$

where $\mathbf{W}^+$ is the pseudo-inverse, or Moore–Penrose inverse, of $\mathbf{W}$ obtained via SVD.

Question 1.1.3: *Explain why the use of the pseudo-inverse is a good choice to obtain the inverse mapping of the linear map (1).*

In the lectures, we derived PCA using the criterion of minimizing the reconstruction error. Another commonly-used criterion to derive PCA is to ask to maximize the variance of the data when projected on the lower-dimension space.

Question 1.1.4: *Derive PCA using the criterion of variance maximization and show that one gets the same result as with the criterion of minimizing the reconstruction error.*

Show this result for projecting the data into k dimensions, not just 1 dimension.

1.2 Multidimensional Scaling (MDS) and Isomap

In the derivation of classical MDS with distance matrix, our goal is 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 $s_{ij} = -\frac{1}{2}(d_{ij}^2 - s_{ii} - s_{jj})$. In the lectures we mention the "double centering trick," and how this can be used to solve for matrix $\mathbf{S}$ given $\mathbf{D}$. The mathematical derivation for the "double centering trick" is given in the textbook of Lee and Verleysen, Section 4.2.2.

Question 1.2.5: *Explain in English what is the intuitive reason that the "double centering trick" is necessary in order to be able to solve for* $\mathbf{S}$ *given* $\mathbf{D}$*.*

Question 1.2.6: *Use the same reasoning as in the previous question (1.2.5) to argue 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.*

In particularly, argue that although the solution obtained by the "first point trick" will be different than the solution obtained by the "double centering trick", both solutions are correct.

Consider now 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 $\mathbf{S}$. Observe that PCA involves a singular-value decomposition (SVD) operation, while classical MDS involved an eigenvector decomposition (EVD) operation.

Question 1.2.7: *Show that the two methods, i.e., classical MDS when* $\mathbf{Y}$ *is known and PCA on* $\mathbf{Y}$*, are equivalent.*

Which of the two methods is more efficient? (Hint: Your answer may involve a case analysis.)

Consider the Isomap method used to reduce the dimensionality of a given dataset. Isomap requires constructing a neighborhood graph G, as discussed in the lectures and the textbook.

Question 1.2.8: *Argue that the process to obtain the neighborhood graph* G *in the Isomap method may yield a disconnected graph. Provide an example. Explain why this is problematic.*

Question 1.2.9: *Propose a heuristic to patch the problem arising in the case of a disconnected neighborhood graph. Explain the intuition of your heuristic and why it is expected to work well. How does your heuristic behave in the example you provided in the previous question?*

1.3 PCA vs. Johnson-Lindenstrauss random projections

Both PCA and Johnson-Lindenstrauss random projections are linear maps. Given data $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_n) \in \mathbb{R}^{d \times n}$, both methods find a matrix $\mathbf{A} \in \mathbb{R}^{k \times d}$ and reduce the dimension of the data from d to k by computing the projection $\mathbf{X} = \mathbf{A}\,\mathbf{Y}$.

Question 1.3.10: *Provide a qualitative comparison (short discussion) between the two methods, PCA vs. Johnson-Lindenstrauss random projections, in terms of (i) projection error; (ii) computational efficiency; and (iii) intended use cases.*

1.4 Programming task — MDS

Question 1.4.11: (Data collection)

Search the internet for an API for calculating distances between world cities. Distances could be estimated by the geodesic, flight time of an actual flight, or other heuristic, it does not matter as long as it is a reasonable approximation.

Use the API to compute the pairwise distance matrix $\mathbf{D}$ *for at least 100 different world cities of your choice. Choose the cities so that they cover the whole globe, for instance all 5 continents, or even a larger number of regions subdividing continents, e.g., Middle East Asia, Central Asia, South East Asia, etc., it is up to you to pick your criteria.*

Question 1.4.12: (Classical MDS)

Apply classical MDS to compute an (x, y) *coordinate for each city in your dataset, given the distance matrix* $\mathbf{D}$*.*

Plot the cities on a plane using the coordinates you computed. You may want to annotate the cities by their name, or abbreviation, use different colors to indicate continents, or regions, etc.

Discuss how good is the reconstructed map your created using classical MDS.

For this task, you should implement MDS by yourself, by relying only on a package for eigenvector decomposition, that is, do not try to find a library that implements MDS.

Question 1.4.13: (Metric MDS)

Repeat the task of 1.4.12, but using metric MDS this time. You may tune the parameters of the method until you are satisfied with the results.

Discuss the parameters that you tuned and their effect on the end result.

Discuss the quality of your map, and compare it with the one you obtained by classical MDS.

For this task, you are free to search for an implementation of metric MDS and use it as a black box.