Assignment 1

Methods of PCA, MDS and Isomap

DD2434 Advanced Machine Learning
Filip Bergentoft, bergento@kth.se

Problem 1

It is assumed throughout this problem that whenever a matrix A is referenced, it is a real and symmetric matrix of size $n \times n$.

Part (i): *Prove that a real symmetric matrix has real eigenvalues*

Let λ be an eigenvalue to A with a corresponding vector v, which is by the definition of an eigenvector non-zero.

The norm of a complex vector z is given by $\|z\| = \sqrt{z^T \overline{z}}$ and is non-zero for all $z \in \mathbb{C}$ except $z = 0$.

Given that A is real we can use that

$$\overline{Av} = \overline{\lambda v} = A\overline{v} = \overline{\lambda v} \tag{1}$$

Then we can expand $v^T A \overline{v}$ in the two following ways

$$v^T A\overline{v} = v^T(A\overline{v}) = \left\{\text{Using equation (1)}\right\} = v^T \overline{\lambda v} = \overline{\lambda} v^T \overline{v} = \overline{\lambda}\|v\|^2 \tag{2}$$

$$v^T A\overline{v} = (A^T v)^T \overline{v} = \left\{A = A^T\right\} = (Av)^T \overline{v} = \lambda v^T \overline{v} = \lambda \|v\|^2 \tag{3}$$

Thus since equation (2) and (3) are equal (from the left hand side) we get that

$$\overline{\lambda}\|v\|^2 = \lambda\|v\|^2 \Rightarrow \lambda = \overline{\lambda} \tag{4}$$

since v is non-zero by the definition of an eigenvector. Thus are the eigenvalues of a real and symmetric matrix real.

Part (ii): qqq: Detta måste fixas!!

Part (iii): *Prove that a positive semi definite matrix has non negative eigenvalues*

Let B be a complex $n \times n$ positive semi-definite matrix accompanied by an eigenvalue λ and an associated eigenvector v. By the definition of a positive semi-definite matrix we know that for any vector $z \in \mathbb{C}$ it holds that $\overline{z}^T B z \geq 0$. The following thus holds

$$\overline{v}^T B v = \overline{v}^T \lambda v = \lambda \overline{v}^T v = \lambda \|v\|^2 \geq 0 \tag{5}$$

where $\|v\|^2$ is necessarily positive since eigenvectors are by definition non-zero. This implies that $\lambda \geq 0$, which was to be proven.

Part (iv): *Let $A \in \mathbb{R}^{n \times n}$ symmetric and positive semi-definite matrix. Define a matrix $D = \{D | D_{ij} = A_{ii} + A_{jj} - 2A_{ij}\}$. Show that there exists n vectors $v_1, ..., v_n,\ v_i \in \mathbb{R}^n\ \forall i$* such that $D_{ij} = \|v_i - v_j\|_2^2$.

$$D_{ij} = \|v_i - v_j\|_2^2 = v_i^T v_i + v_j^T v_j - 2v_i^T v_j \tag{6}$$

Thus, if we can show that any matrix $A \in \mathbb{R}^{n \times n}$ that fulfils the given conditions, can have each of its elements expressed as a dot product between n given vectors $v_1, ..., v_n,\ v_i \in \mathbb{R}^n\ \forall i$ we have shown what is asked.

Given that A is symmetric and positive semi-definite it can be decomposed into the following eigen-decomposition $A = Q\Lambda Q^T$ where Λ is a diagonal matrix of real eigenvalues (since A is PSD) and Q is an orthogonal matrix of eigenvectors of A. The decomposition can then be rewritten in the following manner

$$A = Q\Lambda Q^T = (\Lambda^{-\frac{1}{2}} Q^T)^T (\Lambda^{-\frac{1}{2}} Q^T) = V^T V \tag{7}$$

If we thus choose every $v_i,\ i = 1, ..., n$ to be the i:th column in the matrix $(\Lambda^{-\frac{1}{2}} Q^T)$ we get that $A_{ij} = (V^T V)_{ij} = v_i^T v_j$ which yields the final result

$$D_{ij} = A_{ii} + A_{jj} - 2A_{ij} = v_i^T v_i + v_j^T v_j - 2v_i^T v_j = \|v_i - v_j\|_2^2 \tag{8}$$

which was to be proven.

Problem 2

Given that we want to do PCA using k components on a matrix $Y \in \mathbb{R}^{p \times n},\ p \leq n$ where the columns correspond to data points, we can decompose Y in the following manner using SVD.

$$Y = U\Sigma V^T \tag{9}$$

Where U contains the left singular vectors of Y and V contains the right singular vectors of Y.

The transformation matrix W is then given by the first k columns of the matrix U.

If we instead want to perform PCA using k components on Y^T we can use that

$Y^T = (U\Sigma V^T)^T = V\Sigma U^T$

The transformation matrix $\widetilde{W}$ can thus be chosen as the first k columns in V.

Thus is a single SVD computation sufficient for computing PCA on both columns and rows.

Problem 3

We want to maximise $\mathrm{Tr}(Y^T W W^T Y)$ w.r.t. W where W is any orthogonal matrix and $Y \in \mathbb{R}^{d\times n}$. We begin by expressing Y in terms of its SVD: $Y = U\Sigma V^T = \sum_{i=1}^{d} \sigma_i u_i v_i^T$. Substituting this into the trace expression yields the following:

$$\mathrm{Tr}(Y^T W W^T Y) = \mathrm{Tr}(\sum_{i=1}^{d} \sigma_i v_i u_i^T W W^T \sum_{j=1}^{d} \sigma_j u_j v_j^T) \tag{10}$$

$$= \mathrm{Tr}(\sum_{i=1}^{d}\sum_{j=1}^{d} \sigma_i \sigma_j v_i u_i^T W W^T u_j v_j^T) = \left\{ u_i^T W W^T u_j \text{ is a scalar } \right\} \tag{11}$$

$$= \sum_{i=1}^{d}\sum_{j=1}^{d} \sigma_i \sigma_j u_i^T W W^T u_j \mathrm{Tr}(v_i v_j^T) \tag{12}$$

$$= \left\{ \text{Using that } \mathrm{Tr}(v_i v_j^T) = v_j^T v_i = 1, i = j, 0 \text{ otherwise} \right\} \tag{13}$$

$$= \sum_{i=1}^{d} \sigma_i^2 u_i^T W W^T u_i = \sum_{i=1}^{d}\sum_{i=1}^{k} \sigma_i^2 u_i^T w_j w_j^T u_i \tag{14}$$

$$\tag{15}$$

Thus is

$$\max_W \mathrm{Tr}(Y^T W W^T Y) = \max_{w_j, j=1,\ldots,k} \sum_{i=1}^{d}\sum_{i=1}^{k} \sigma_i^2 u_i^T w_j w_j^T u_i \tag{16}$$

Since $k \leq d$ and U is an orthogonal matrix, any vector w_j, $j = 1, ..., k$ can be expressed in terms of the columns in U. Thus substituting $w_j = Ua_j$ into the RHS of equation (16) yields the following

$$\max_{w_j,j=1,\ldots,k}\sum_{i=1}^{d}\sum_{i=1}^{k}\sigma_i^2 u_i^T w_j w_j^T u_i = \max_{a_j,j=1,\ldots,k}\sum_{i=1}^{d}\sum_{i=1}^{k}\sigma_i^2 u_i^T U a_j a_j^T U^T u_i \tag{17}$$

$$= \max_{a_j,j=1,\ldots,k}\sum_{i=1}^{d}\sum_{i=1}^{k}\sigma_i^2 e_i^T a_j a_j^T e_i = \max_{a_j,j=1,\ldots,k}\sum_{i=1}^{d}\sum_{i=1}^{k}\sigma_i^2 a_{ij}^2 \tag{18}$$

$$= \max_{a_j,j=1,\ldots,k}\sum_{i=1}^{k} a_j^T \Lambda a_j \tag{19}$$

Problem 4

In order to show that it provides a correct estimation of the Gram matrix S we need to prove that we can derive a similarity matrix S from a matrix $D$where$(D)_{ij} = d_{ij} = (z_i - z_j)^T(z_i - z_j)$ such that

1. $S = Z^T Z$
2. Z gives 0 reconstruction error for the matrix D

Normally $s_{ij} = y_i T y_j$, we claim that it is valid to express

$$s_{ij} = -\frac{1}{2}(d_{ij}^2 - d_{1j}^2 - d_{1i}^2) \tag{20}$$

where

$$d_{ij}^2 = (y_i - y_j)^2 = y_i^2 + y_j^2 - 2y_i y_j \tag{21}$$

.

Substituting (21) into (20) yields the following

$$s_{ij} = -\frac{1}{2}(d_{ij}^2 - d_{1j}^2 - d_{1i}^2) \tag{22}$$

$$= -\frac{1}{2}(y_i^2 + y_j^2 - 2y_i^T y_j - y_1^2 - y_i^2 + 2y_1^T y_i - y_1^2 - y_j^2 + 2y_1^T y_j) \tag{23}$$

$$= -\frac{1}{2}(-2y_i^T y_j - 2y_1^2 + 2y_1^T y_i + 2y_1^T y_j) \tag{24}$$

$$= (y_i - y_1)^T (y_j - y_1) = z_i^T z_j \tag{25}$$

Let T be a matrix of the same size as Y, with all columns as y_1. Then the Gram matrix S can be written as

$$S = (Y - T)^T (Y - T) = Z^T Z \tag{26}$$

Thus is 1. shown. Now we want to show 2.

We know that $d_{i,j} = (z_i - z_j)^T(z_i - z_j)$ and by substituting the result from equation (25) we get the following

$$
\begin{aligned}
d_{ij}^2 &= (z_i - z_j)^T(z_i - z_j) && (27)\\
&= (y_i - y_1 - (y_j - y_1))^T(y_i - y_1 - (y_j - y_1)) && (28)\\
&= (y_i - y_j)^T(y_i - y_j) && (29)
\end{aligned}
$$

Which shows that Z gives 0 reconstruction error for the distance matrix D.

Problem 5

We are considering the classical MDS when $Y \in \mathbb{R}^{d\times n}$ is known which implies that we can construct a similarity matrix $S = Y^TY$. An MDS embedding can then be obtained by performing eigen-decomposition on S which yields the following

$$
S = Y^TY = Q\Lambda Q^T = (\Lambda^{-\frac{1}{2}}Q^T)^T(\Lambda^{-\frac{1}{2}}Q^T) \tag{30}
$$

where Λ is a diagonal matrix with eigenvalues of $S = Y^TY$ sorted in descending order.

The latent matrix X can then be chosen as

$$
X = I_{k\times n}\Lambda^{-\frac{1}{2}}Q^T,\ k < d \tag{31}
$$

However, this can also be written in terms of the SVD of $Y = U\Sigma V^T$

$$
\begin{aligned}
S = Y^TY = V\Sigma^TU^TU\Sigma V^T = V\Sigma^T\Sigma V^T &= (\Sigma V^T)^T(\Sigma V^T) && (32)\\
= (\Lambda^{-\frac{1}{2}}Q^T)^T(\Lambda^{-\frac{1}{2}}Q^T) & && (33)
\end{aligned}
$$

Since Λ and Σ has ordered diagonals by magnitude and V contains the right singular vectors we know that $V = Q$ which gives that

$$
\Lambda^{-\frac{1}{2}}Q^T = \Sigma V^T \tag{34}
$$

Thus can the latent matrix also be expressed as

$$
X = I_{k\times n}\Sigma V^T,\ k < d \tag{35}
$$

In PCA we decompose Y into its SVD, yielding

$$Y = U\Sigma V^T \tag{36}$$

The latent matrix is then chosen as

$$\begin{aligned}
X &= (UI_{n\times k})^T Y = I_{k\times n} U^T Y && (37)\\
&= \left\{ Y = U\Sigma V^T \Rightarrow U^T Y = \Sigma V^T \right\} && (38)\\
&= I_{k\times n}\Sigma V^T,\ k < d && (39)
\end{aligned}$$

Comparing equations (35) and (39) we see that the results are equivalent.

Regarding the computational efficiency I believe that there are two areas to consider: computational complexity and numerical accuracy. The complexities of computing the SVD and eigen-decomposition are similar and somewhat dependant on the properties of the input matrix, eigen-decomposition often performing a bit better. However, for the MDS we are required to perform a matrix multiplication beforehand which both introduces additional cost in time but can also introduce issues with numerical accuracy causing the resulting eigen-decomposition to have imaginary parts. For this reason the SVD should be preferred.

Problem 6

Problem 7

The goal of this problem is to visualise how similar different animals at a zoo are by projecting the given data from $\mathbb{R}^{16}$ to

$$\mathbb{R}^2$$

using three different embeddings; PCA, MDS and isomap.

Preprocessing the data

In order to enable the applications of the embeddings the data has to be preprocessed by first removing the columns 'type' and 'animal name' from the data set. These can later be used in the visualisation.

The remaining attributes are all boolean with values in $\{0, 1\}$ except the attribute 'legs' which takes values in $\{0, 2, 4, 6, 8\}$. This attribute thus takes on values up to 8 times the magnitude in comparison to the remaining attributes which corresponds to it having an importance weight of 8 and the other a weight of 1 when taking the distance between points. For this reason the magnitude of this attribute will be scaled down by 8 times, causing it to take values in $\{0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1\}$.

PCA

MDS

Isomap