Algorithm 1 PCA

Input: data set X (each column contains a data point), the dimensionality d of the projection

Output: a matrix containing principal components U_d, a matrix (or vector) containing eigen-values, and reduced version of data set Z_d

function PCA(X, d)

find the mean μ of the data set (i.e. take mean along columns),

centralize the data set $Z = X - \mu$

compute principal components U and eigen-values D % use the build-in function **eig** in Matlab

only pick the first d principal components, i.e. $U_d = U(:, 1:d)$

reduce the dimensionality of the data $Z_d = U_d^T Z$

end function

7 Practical: Dimensionality Reduction

In this assignment, we would like to:

1. use the pseudo code 1 to implement PCA,
2. apply PCA on a real world data set, called COIL20, and reduce its dimensionality to $d = 30$. It means the output of PCA is a matrix of size 30×1440 (i.e $Z_d \in \mathbb{R}^{30\times1440}$)
3. apply the build-in function tsne (in matlab) on the output of PCA to visualize Z_d.
    - Since in matlab it assumes that data points lie on rows, it is necessary to give the transpose of Z_d as input to tsne function (i.e. tsne(transpose(Z_d))).

Note that in Nestor you will find the file `COIL20.mat`. It contains 1440 images (with 32×32 pixels) of 20 objects rotated 72 times (5 degrees per image).

- if you want to take a look in i-th image, you can try

$$> imshow(reshape(X(:, i), 32, 32))$$

Report

You should hand in a structured report comprising:

- **(1 point)** An **Introduction** section that describes your assignment.
- **(3 points)** A **Methods** section in which you explain the PCA in a general manner. You need to implement the PCA yourself. Code and implementation itself will also be taken into account for the grading of this section.
- **(4 points)** A **Experimental results** section in which you provide the following details:
    - A plot showing the eigen-values profile of the data set, i.e. x-axis is the eigen-values indices $(1, 2, \cdots 1024)$ and y-axis is the eigen-value,

- A table reporting the dimensionality d if we want to keep 0.9, 0.95 and 0.98 fraction of the total variance. Use the following formula:

$$\frac{\sum_{i=1}^{d} \lambda_i}{\sum_{i=1}^{n} \lambda_i}$$

- A plot of the result of t-SNE (use labels to color the points),

- **(2 points)** A **Discussion** section that includes your observations on the results.