Dr. Peter Arndt
Dr. Konrad Völkel

Heinrich-Heine-Universität Düsseldorf
Winter term 2022/23

Machine Learning
Exercise Sheet 5
(4 Exercises, 120 Points)
Due: 15.11.2022, 10:00

Exercise 1: (30 Points)

Centering matrix
The Centering matrix is defined as $H = I_n - \frac{1}{n} 1_n 1_n^\top$ where I_n is the $n \times n$ identity matrix, 1_n is the n dimensional one-vector, i.e. $1_n = [1, 1, \ldots, 1]^\top$. Show:

1. H is symmetric, i.e. $H = H^\top$.
2. H is idempotent, i.e. $H = HH$.
3. Show that $H1_n = 0_n$, i.e., 1_n is an eigenvector with eigenvalue 0.
4. What are the other eigenvalues of H?
5. For a data matrix $X \in \mathbb{R}^{d \times n}$ show that XH has mean zero, i.e. show $\frac{1}{n} XH1_n = 0_d$.

Exercise 2: (30 Points)

Benchmarking the centering matrix (programming task)
The centering matrix is a very handy tool for theoretical derivations, however it is not a computational efficient way to center a data matrix as you will see by solving this exercise.

1. Write a Python function `center_with_matrix(data)` that subtracts the row-wise mean from the input array `data` by multiplying with the centering matrix.
2. Write a Python function `centering_with_numpy(data)` that performs the same operation using basic NumPy-functions.
3. Sample random data matrices with uniformly distributed entries with 10 rows and a different number of columns. Plot the number of columns of the data matrix against the elapsed runtime for both functions. Also add a legend to your plot.

Exercise 3: (30 Points)

PCA on iris data (programming task)
For this exercise we will use the iris data set which is available online[1]. The goal is to perform a principal component analysis (PCA) and derive with a new feature subspace. You may use NumPy/SciPy functions for the computations.

1. Write a Python function that calculates the covariance matrix

$$\Sigma = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \hat{\mu})(x_i - \hat{\mu})^T$$

   of a dataset $X = [x_1, \ldots, x_n] \in \mathbb{R}^{D \times n}$, where $\hat{\mu} \in \mathbb{R}^D$ is the mean of the data matrix. Do not use the function `np.cov`.

[1] `https://archive.ics.uci.edu/ml/datasets/iris`

Machine Learning
Exercise Sheet 5
(4 Exercises, 120 Points)
Due: 15.11.2022, 10:00

---

2. Implement a function `pca(X, d, whitening=False)` that performs PCA on the input data `X` and returns the projected (and optionally whitened) data `Y`, the matrix of eigenvectors `V`, and the eigenvalues `Lambda`. For the eigenvector decomposition you can use the function `np.linalg.eigh`.

3. Project the Iris data onto a two-dimensional feature space. Create scatter plots visualizing the projected data points before and after applying whitening.

Important: This exercise assumes that the data points are stacked columnwise!

Exercise 4: (30 Points)

Inhomogeneous kernel function
For $a, b \in \mathbb{R}^2$, consider the inhomogeneous polynomial kernel function $k(a, b) = (a^\top b+1)^p$.

1. Show that $\Phi(a) = [a_1^2, a_2^2, \sqrt{2}a_1a_2, \sqrt{2}a_1, \sqrt{2}a_2, 1]^T$ is a feature map for the kernel $k(a, b) = (a^\top b + 1)^2$, i. e., show that $k(a, b) = \Phi(a)^T \Phi(b)$.

2. What is the corresponding feature map for $k(a, b) = (a^\top b + 1)^3$? What is the dimensionality of that feature space?

3. What is the feature space dimension of $k(a, b) = (a^\top b + 1)^p$? Prove your answer.