Mathematics of Learning – Worksheet 5 - – Discussion on Novermber 16/17th, 2023

- The exercise sheets will be uploaded every Monday. Solution sketches will be uploaded one week later.
- You can hand in your own solutions via StudOn and we correct them - this is not mandatory. Please hand in small groups of 2-3 students.
- For questions, please use the forum on StudOn since other students may have similar questions. If you have a more personal question about the exercises please send an email to ehsan.waiezi@fau.de or lars.weidner@fau.de respectively.

Exercise 1 [Reading assignment: Spectral Clustering].
Read chapter 14.5.3 regarding Spectral Clustering in the Hastie book. Spectral Clustering is a method which can be applied to data with some radial structure, for example. At some point in the chapter, the Laplacian of graphs will be of importance. If you do not know about graph Laplacians, inform yourself about it (it is going to be important later in the lecture). Peculiarly ambitious students can implement their version of Spectral Clustering and apply it on various data sets (extract some data from the internet or use the data sets already uploaded or which you generated on your own - be creative). Discuss the contents of the chapter with a fellow student for at least half an hour.

Exercise 2 [Reading assignment: Supervised learning].
Read chapter 2 of the Hastie Book. It gives you a good overview of supervised learning, what will be the content of the course in the next weeks. Discuss the contents of the chapter with a fellow student for at least half an hour.

Exercise 3 [Regression].
The regression problem takes as input data N data points of *explanatory* or *independent* vectors $X_i \in \mathbb{R}^p$, and some *response* or *dependent* reals $Y_i \in \mathbb{R}$. The goal of regression is now, to express the response variables as good as possible as a function of the explanatory variables, i.e., to find a function $g\colon \mathbb{R}^p \to \mathbb{R}$ such that $g(X_i) \approx Y_i$. Since

$$\tilde{g}\colon x \to \begin{cases} Y_i & if\ x = X_i \\ 0 & \text{otherwise} \end{cases}$$

would perfectly do it, but presumably has terrible out of sample behavior, we cannot allow the whole space of functions as candidates (type "overfitting" in a search engine of your choice). Rather, we'd make use of a set of pre-specified predictor functions, i.e., $f_j\colon \mathbb{R}^p \to \mathbb{R}$ which are simple or natural in some sense, e.g., $f_1(x) = x_1$, $f_2(x) = x_1 \cdot x_2$ or $f_3(x) = e^{x_1}$, and try to linearly combine a function of these to predict the response variables as good as possible. No matter which set of predictor functions

we choose, this leads to a minimization problem with linear constraints, and, depending on how we define "as good as possible fit the response variables" to different objective functions, but classically the quality of approximation is defined as minimizing the euclidean norm of the approximation error vector, leading to a convex-quadratic objective function (what is going to be the standard in the following).

Hence, confronted with $N \in \mathbb{N}$ data vectors $X_i \in \mathbb{R}^p$ and $Y_i \in \mathbb{R}$, to be fitted, the procedure is the following:

1. Choose (guess; depending on what kind of relation you expect between X and Y) functions $f_1, ..., f_m$, mapping from $\mathbb{R}^p$ to $\mathbb{R}$
2. Calculate the matrix A, $A_{i,j} := f_j(X_i)$.
3. Solve the minimization problem

$$\min_{\beta \in \mathbb{R}^m} ||A\beta - Y||_2. \tag{1}$$

The solution $\tilde{\beta}$ of the optimization problem can be interpreted as the coefficients of the predictor functions: Read it as "Y can be expressed optimally as $\beta_1 f_1(X) + \cdots + \beta_m f_m(X)$".

a) Prove: A vector $\tilde{\beta} \in \mathbb{R}^m$ solves the optimization problem 1, if and only if it solves the linear system

$$A^T A\beta = A^T Y$$

b) Show that the solution is unique if and only if A has full column rank.

Solution.
a) We are optimizing the squareroot of a positive function; hence we can just optimize the function itself. Lets calculate the derivative (once again...).

$$\frac{\partial}{\partial \beta_j} \sum_{i=1}^{N} (A_{i,\cdot}\beta - Y_i)^2 = \sum_{i=1}^{N} 2(A_{i,\cdot}\beta - Y_i) \cdot A_{i,j} = 2(A_{\cdot,j})^T \cdot (A\beta - Y)$$

hence, for this to be zero, it has to hold (necessarily and sufficiently) that $A^T(A\beta - Y) = 0$, hence, $A^T A\beta = A^T Y$.

b) If and only if A has full column rank, $A^T A$ has full rank. Further, it is equivalent, that a linear equation system is uniquely solvable if and only if the matrix has full rank.

Exercise 4 [Implementing regression].
Implement the regression algorithm described in the previous exercise.
Find out the function generating the points which are presented in the file "regression.csv". To generate the response points from the exploratory points, I used a few (but not all) of the functions $1, x, x^2, x^3, e^x, ln(|x| + 1), sqrt(|x|), sin(x), cos(x), tan(x)$ and a normal distributed perturbation to linearly combine the response values. Use half of the data points as training set and half of the data points as validation set. Hint: It could be necessary to first invest a little bit of thinking which of the prediction functions can be excluded beforehand.

Solution. See python code.

Exercise 5 [Apply techniques learned so far to (more) realistic data sets]. Download the yale faces data set at the StudOn Platform. Interpret them as grayscale vectors and apply everything what you learned so far (apply k-means-clustering and EM-clustering with known and unknown k on the data set with full and (linearly and kernel) reduced dimension). Visualize, interpret and discuss your results.