Machine Learning and Physics
Winter Semester 2022/2023

Exercise Sheet #2

Due: 07.11.22 11h00

Roman Remme

General Regulations.

- Please hand in your solutions in groups of three people. A mix of attendees from Monday and Tuesday tutorials is fine.
- Your solutions to theoretical exercises can be either handwritten notes (scanned), or typeset using LaTeX.
- For the practical exercises, the data and a skeleton for your jupyter notebook are available at https://github.com/hci-unihd/mlph_sheet02. Always provide the (commented) code as well as the output, and don't forget to explain/interpret the latter. Please hand in both the notebook (`.ipynb`), as well as an exported pdf.
- Submit all your files in the Übungsgruppenverwaltung, only once for your group of three.

1 Kernel Density Estimation

(a) Implement a Quartic (biweight) kernel

$$k\left(x-\mu;w\right)=\frac{15}{16w}\left(1-\left(\frac{x-\mu}{w}\right)^2\right)^2 \qquad \text{with support in } [-w,w]$$

and plot it for $\mu = 0$ and $w = 1$ over the range $[-1, 1]$. (2 pts)

(b) Take the first $N = 50$ data points from `samples.npy`, compute and plot the kernel density estimate over the range $[-10, 20]$ for a set of different bandwidths (e.g. $w \in \{0.1, 0.5, 1, 3, 5\}$). Discuss the results and the influence of the bandwidth. Which bandwidth is optimal in your opinion? Explore what happens as you increase the number of samples N. (5 pts)

2 Bonus: Average shifted Histograms and KDE

Average shifted histograms do what their name implies: they compute a number h of histograms with random offsets and average their results.

Prove that, for $h \to \infty$, average shifted histograms converge to a kernel density estimate. What does the shape of the kernel look like for

(a) 1D histograms with uniform bin width (2 pts)

(b) 2D histograms with axis-aligned rectangular bins (1 pt)

(c) 2D histograms made from any regular tiling (covering of the plane using a single shape, without gaps or overlaps, and without rotating the shape) (1 pt)

3 Mean-Shift

(a) Gradient ascent on the KDE with the Epanechnikov kernel corresponds to the update step

$$x_j^{t+1} = x_j + \alpha_j^t \frac{2}{n} \sum_{i:||x_i - x_j^t||<1} (x_i - x_j^t)$$

For which choice of the adaptive learning rate α_j^t is this equivalent of updates to the local mean? Why is this a sensible choice of learning rate? (3 pts)

(b) Implement the updates to the local mean in python. Apply your implementation to the 1D dataset from exercise 1 and visualize how the points move over time, by plotting a line of x over t for every data point. (5 pts)

4 K-Means

(a) Derive the Updates. We aim to cluster a data set $\mathbf{X} \in \mathbb{R}^{p \times N}$ into K clusters, by choosing cluster centers $\mathbf{C} \in \mathbb{R}^{p \times K}$ and cluster memberships $\mathbf{M} \in [0,1]^{K \times N}$, with $\sum_k M_{kn} = 1$, such that

$$E(\mathbf{C}, \mathbf{M}; K) = ||\mathbf{X} - \mathbf{CM}||^2 = \sum_{n=1}^{N} \sum_{k=1}^{K} m_{kn} ||\mathbf{x}_n - \mathbf{c}_k||^2$$

is minimized. Solve this by deriving the optimal (alternating) updates for each m_{kn} and $\mathbf{c}_k$. (4 pts)

(b) Use the implementation of K-Means from `scikit-learn` and apply it to the jet-tagging dataset from the last sheet. Explore how the algorithm performs for different random starting values and different values of K. In each case plot how $E(\cdot)$ develops over time (Hint: Set `n_init=1, max_iter=1` and use the current state as initialization to get a single step). Do this both for choosing a random subset of the data as the initial cluster centers (`init="random"`) and K-Means++ initialization (`init="k-means++"`) and interpret your results. (6 pts)

5 Bonus: On KDE Bandwidth and Modes

The RBF ("Radial Basis Function") kernel is defined by

$$k(x; w) = \frac{1}{w\sqrt{2\pi}} \exp\left(-\frac{\|x\|^2}{2w^2}\right).$$

Disprove by counterexample or otherwise the following false statement:
For every set of points $x_i \in \mathbb{R}^n, i = 1, \ldots, N$, the number of modes of their KDE with the RBF kernel decreases monotonously as the bandwidth w increases. (5 pts)