Density Estimation

11.1 Kernel Density Estimation & Validation (6 points)

Data: The intensities of the image `testimg.jpg` are given as one byte per pixel, i.e. as gray values represented by type `unsigned char` (or `uint8`, or similar). Thus $I \in [0, 255]^{m \times n}$, where I is the image and m, n are the height and width of the image. Consider the pixel values as i.i.d. samples drawn from the same (but unknown) distribution. This assumes that there are no statistical dependencies between neighboring pixels, which is not true in this case but we will ignore this for the current problem sheet.

- Load the data into a vector and normalize it such that the values are between 0 and 1.
- Create *two new datasets* by adding Gaussian noise with zero mean and standard deviation $\sigma_N \in \{0.05, 0.1\}$.
- Create a figure showing the 3 histograms (original & 2 sets of noise corrupted data – use enough bins!). In an additional figure, show the three corresponding empirical distribution functions in one plot.

For one of the datasets

1. Take a subset of $P = 100$ observations and estimate the probability density $\hat{p}$ of intensities with a *rectangular* kernel ("gliding window") parametrized by window width h.
   - Plot the estimates $\hat{p}$ resulting for (e.g. 10) different samples of size P.
   - Calculate the negative log-likelihood per datapoint of your estimator using 5000 samples from the data not used for the density estimation (i.e. the "test-set"). Get the average of the negative log-likelihood over the 10 samples.
2. Repeat this procedure (without plotting) for a sequence of kernel widths h to get the mean log likelihood (averaged over the different samples) resulting for each value of h.

This should give you for a specific dataset, sample size, the relation between kernel-width and mean validated likelihood.

(a) Apply this procedure to all 3 datasets (original and the two noise-corruped ones) to make a plot showing the obtained likelihoods (y-axis) vs. kernel width h (x-axis) as one line for each dataset.

(b) Repeat the previous step (LL & plot) for samples of size $P = 500$.

(c) Repeat the previous steps (a & b) for the Gaussian kernel with $\sigma^2 = h$.

Interpretation: Which kernel width h yields the minimal negative log-likelihood hence the minimal generalization error, for each combination? How do you interpret this result?

11.2 Gaussian Mixture Model (4 points)

In this exercise the Expectation-Maximization (EM) algorithm is used to estimate the model parameters of a mixture of Gaussians.

1. Create a toy dataset of 2-dimensional data points $\boldsymbol{x}^{(\alpha)} = (x_1^{(\alpha)}, x_2^{(\alpha)})$, $\alpha = 1, \ldots, 100$. The points should be generated by the mixture model

$$P(\boldsymbol{x}) = P(1)\,\mathcal{N}(\boldsymbol{x};\, \boldsymbol{w}_1, \sigma_1^2) + P(2)\,\mathcal{N}(\boldsymbol{x};\, \boldsymbol{w}_2, \sigma_2^2) \tag{1}$$

   with mixture parameters $P(1) = 2/3$, $P(2) = 1/3$, as well as means $\boldsymbol{w}_1 = (2, 2)^T$, $\boldsymbol{w}_2 = (1, 1)^T$ and standard deviations $\sigma_1 = 0.7$, $\sigma_2 = 0.2$ of the two Gaussian components with densities

$$\mathcal{N}\left(\boldsymbol{x};\, \boldsymbol{w}_q, \sigma_q^2\right) = \frac{1}{(2\pi\sigma_q^2)} \exp\left\{-\frac{(\boldsymbol{x} - \boldsymbol{w}_q)^2}{2\sigma_q^2}\right\} \tag{2}$$

   for $q = 1, 2$. Save for each data point in addition to the (observed) value $\boldsymbol{x}^{(\alpha)}$ also the (number of the) component it was generated by, using for example a (hidden) binary assignment variable $m_q^{(\alpha)}$ indicating with 1 that the data point $\boldsymbol{x}^{(\alpha)}$ stems from component q (as also used for the clustering algorithms).

2. Apply the EM algorithm to obtain estimations $\{\hat{P}(q), \hat{\boldsymbol{w}}_q, \hat{\sigma}_q\}_{q=1,2}$ of the mixture model parameters. Stop iterating when the absolute difference between the new and the old value of each (estimated) parameter does not change more than $\theta = 0.001$ (tolerance parameter). Instead of the absolute value use the Euclidean norm for the difference of the parameter $\boldsymbol{w}_q$. What is the error of the estimated vs. the true parameters?

3. Compare the result of the EM algorithm to that obtained from running the (batch) K-means clustering algorithm, in terms of assignment of each data point to the components/clusters.

4. Compare the number of iterations until convergence of the EM algorithm with parameters initialized randomly vs. initialized using (batch) K-means clustering (taking as $\hat{\boldsymbol{w}}_1$ and $\hat{\boldsymbol{w}}_2$ the prototypes and for $\hat{\sigma}_1$ and $\hat{\sigma}_2$ the empirically estimated spread within each cluster).

5. Repeat the steps above for $\sigma_1 = 0.1, 0.5, 1, 1.5$ and visualize your results in a compact representation comparing the 5 different models.

Total points: 10