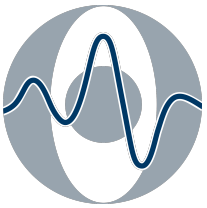
Exercises for
Pattern Analysis
Amir Davari, Dalia Rodriguez Salas, Johannes Bopp, Lina Felsner
Assignment 1, published on April 30, 2019



General Information:

Lecture:	Mon 16:15h – 17:45h and Wed 10:15h – 11:45h (H16)
Exercises:	Tue 14:00 – 15:00, 15:00 – 16:00 (02.151-113) and
	Thu 16:00 – 18:00 (0.01-142, 00.156-113, 02.151b-113)
Certificate:	Oral exam at the end of the semester
Contact:	amir.davari@fau.de & dalia.rodriguez@fau.de

Density Estimation

Exercise 1 Our First Empirical Distribution: Sampled from a Ground Truth Distribution

A central challenge in statistical machine learning is the discrepancy between a true, ideal data distribution and the actual, observed distribution. The purpose of statistical machine learning is to deduce the ideal data distribution from the observed distribution. We will write some first code here to better understand the relationship between both distributions.

A typical statement is that the observations are "noisy instances that are generated from the underlying ground truth distribution". The term *noise* is used quite sloppily here, and may refer to any modification of the data or its coordinates.

Let us get started. In this task, we create a ground truth distribution, and draw an empirical distribution from it. We can "misuse" a gray-scale image for this task, namely if we imagine that the x- and y-coordinates of a pixel are feature dimensions x_1 and x_2, and the grayscale intensity represents the relative density at its respective (x, y)-location.

To this end, use `scipy.misc.face(gray=true)` to load a gray-scale image of a raccoon face. Apply a Gaussian filter (e.g., with $\sigma = 3$) to slightly smooth the image. Use `matplotlib` to visualize the image.

Implement a function that draws new samples from this density, using the approach with the cumulative density function from the lecture. To visualize the drawn samples, it is most convenient to create a new (empty) image with the same dimensions as the raccoon, and to draw a point at every location that is sampled from the density.

Draw 10.000, 50.000, 100.000, 200.000, and 400.000 samples. What do you observe?

Parzen Window Estimator

Use the Parzen Window Estimator to "reconstruct" the image. Thus, use the sampled density from the previous task as input, and output a density from the Parzen window estimator.

More in detail, Implement a Parzen window estimator with a box kernel. Use our

empirical raccoon densities from the previous task as an input. Vary the window size of the Parzen estimator. What do you observe?

A First Taste of the Model Selection Problem

Which kernel size is best suited for our raccoon density? Implement a cross-validation to automatically determine the best kernel size from a reasonably large list of candidate kernel sizes (with at least 3 candidate sizes). The objective function shall be chosen analogously to the lecture.

If you are familiar with cross-validation, then please just go ahead and implement it. Otherwise, you may follow these steps: to perform k-fold cross-validation, split the samples $\mathcal{S}$ into a test set $\mathcal{S}_j$ of size $|\mathcal{S}_j| = N/k$ and a training set $\mathcal{T}_j = \mathcal{S} \setminus \mathcal{S}_j$. Build a density $p(\mathbf{x})_\theta^j$ from $\mathcal{T}_j$ where θ denotes the candidate kernel size. Choose the best θ with an ML estimation across all folds, by choosing $\hat{\theta} = \operatorname*{argmax}_\theta \sum -\log p(\mathbf{x})_\theta^j$ on all $\mathbf{x} \in \mathcal{S}_j$ across all folds j.

Play with it! Which observations are new or surprising to you? In other words, what do you experience in the experiments that was not obvious to you from the theoretical derivation of the model selection problem?

Demonstration of your Results: please book a date on your exercise slot in studOn, i.e., for Tuesday May 14 or Thursday May 16!