Homework 3: Sampling

Due: 11:59PM, April 29, 2024

Software

Python packages *NumPy* and *Matplotlib* are sufficient for the calculation used this assignment. The computation of this assignment could take a while to finish on your computer, but it should not take more than *a few minutes*. Even though not required, if you are interested in accelerating the computation, we encourage using Python package *JAX* or *PyTorch* with GPU acceleration (Google Colab provides free GPU access). Feel free to use Jupyter Notebook as the programming environment.

1. (20 pts) Given the image here, convert it into a continuous probability density function over a space of 1 meter by 1 meter (you can find the example code for how to do it here). Implement rejection sampling to sample 5000 points from this image-based probability distribution. Select two different proposal distributions of your choice for your implementation.

   Turn in: Two plots showing the resulting samples, one for each of the proposal distributions that you choose. Specify which proposal distribution you choose to use. The resulting samples should look similar to the figure below.

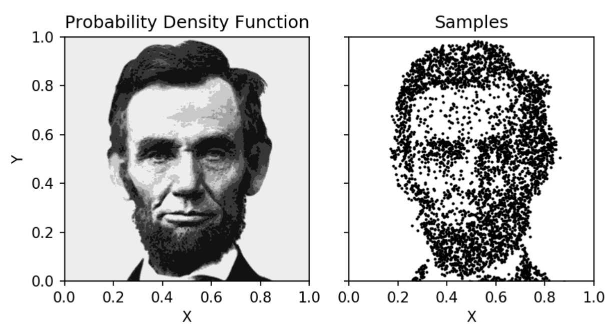

Figure 4: Example of the rejection sampling from the image.

2. (40 pts) Consider the following differential drive vehicle model. The vehicle continuously executes a constant control signal $[u_1(t), u_2(t)[= [1, -1/2]$ for a length of time $T = 2\pi sec$.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta)u_1 \\ \sin(\theta)u_1 \\ u_2 \end{bmatrix}, \ (x(0), y(0), \theta(0)) = (0, 0, \pi/2).$$

   Implement a particle filter to track the state of the vehicle. For the parameters, use $dt = 0.1$, the process noise normally distributed over x, y, θ with a variance of 0.002, and the measurement noise normally distributed over x, y, θ with a variance of 0.02.

   Turn In: A plot of the particles every 1 second for a total of 6 seconds. Include your estimate at each time step. It should look something like the figure below:

3. (40 pts) Consider the following Gaussian mixture model:

$$p(x) = w_1 \cdot \mathcal{N}(x|\mu_1, \Sigma_1) + w_2 \cdot \mathcal{N}(x|\mu_2, \Sigma_2) + w_3 \cdot \mathcal{N}(x|\mu_3, \Sigma_3), \tag{3}$$

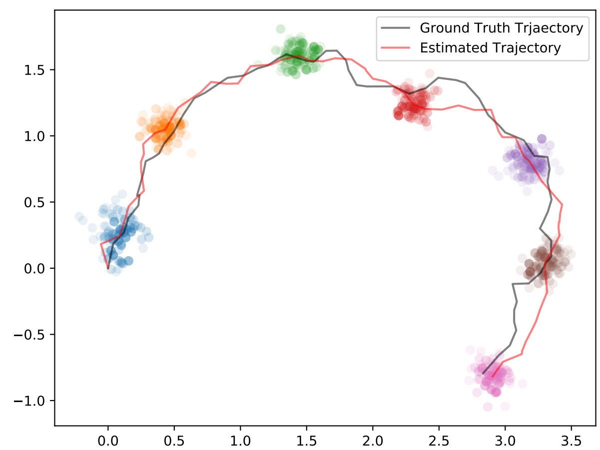

Figure 5: Each color represents a snapshot of the particles at a particular time. The red line is the state estimate of x and y every dt seconds and the black line is the ground truth.

where

$$w_1 = 0.5, w_2 = 0.2, w_3 = 0.3,$$
$$\mu_1 = [0.35, 0.38]^\top, \mu_2 = [0.68, 0.25]^\top, \mu_3 = [0.56, 0.64]^\top,$$
$$\Sigma_1 = \begin{bmatrix} 0.01 & 0.004 \\ 0.004 & 0.01 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 0.005 & -0.003 \\ -0.003 & 0.005 \end{bmatrix}, \Sigma_3 = \begin{bmatrix} 0.008 & 0.0 \\ 0.0 & 0.004 \end{bmatrix}. \tag{4}$$

Generate 100 samples from this GMM distribution. Implement the expectation-maximization algorithm and run for 5 iterations to estimate the GMM model from the samples.

Turn In: A 5-by-3 plot showing the estimated GMM model in each iteration of the expectation-maximization algorithm overlapped with the samples. Each row (5 rows in total) of the plots shows the following visualization.

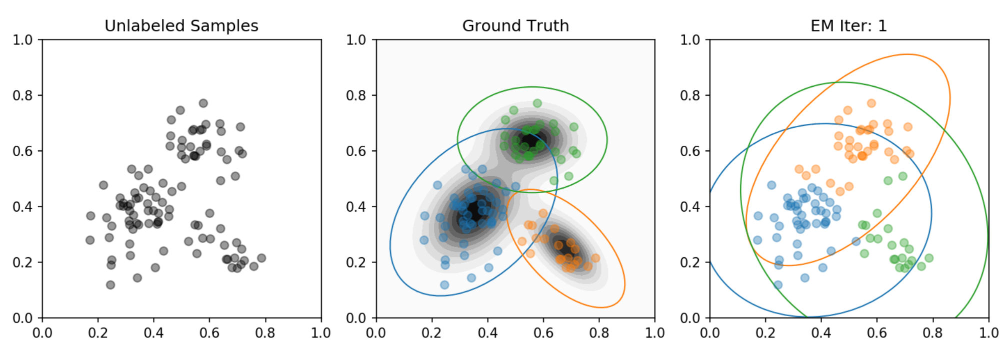

Figure 6: Example visualization of the EM iterations.