COURSEWORK 2 – 10%

This assignment has two parts and graded over **10 pts**. Some general remarks:

- The assignment is due on 30 Nov. 2022. 1PM GMT, to be submitted via Blackboard (see the instructions on the course website).
- You should submit a PDF report. You can do this via two ways (your choice):
    1. Prepare a PDF. Please keep the report limited to 15 pages. Fewer is appreciated. You can put relevant code for a question next to your derivations (like an IPython notebook). However **please format the code** – if you are using Word, make sure that the code is boxed and coloured. If you are using LaTeX, then use `pythonhighlight` package (or something similar) in LaTeX for readability (please do!). We do not have much handwritten derivations in this assignment, but you can include a handwritten page if you wish to, however, please organise your report cleanly in this case and submit a single PDF merged together.
    2. Prepare an IPython notebook and export it as a PDF (preferred). Again the length is limited to 15 pages. If you can't export your HTML notebook as PDF, use "Print" feature in browser (Chrome: File -> Print) and choose "Save as PDF".
- You can reuse the code from the course material, but try to personalise your code in order to avoid having problems with plagiarism checks. You can use Python's functions for sampling Normals: `np.random.normal`.

Q1: IMPORTANCE SAMPLING FOR MARGINAL LIKELIHOODS (5 PTS)

Let us consider the standard Gaussian model

$$\begin{aligned} p(x) &= \mathcal{N}(x; 0, 1) \\ p(y|x) &= \mathcal{N}(y; x, 1), \end{aligned}$$

and, again, we would like to estimate the marginal likelihood $p(y) = \int p(y|x)p(x)\mathrm{d}x$ using i.i.d Monte Carlo (MC) and importance sampling (IS). This is a quantity that computes the model's likelihood under a particular observation.

1. (1 pt) Write the analytical expression of $p(y)$ (this is provided in lecture notes – you don't have to prove it). Now assume that we observe $y = 9$. Compute $p(y = 9)$ using your exact $p(y)$ expression. (Hint: It is a small number)

2. (1 pt) What happened? We observed an unlikely observation that could come from the model. It is of practical interest to estimate such probabilities (unlikely events under the model). Next, implement the standard MC estimator (i.i.d) to estimate the integral $p(y = 9) = \int p(y = 9|x)p(x)\mathrm{d}x$ by drawing i.i.d samples from $p(x)$. Describe your test function and estimator clearly. Provide results for $N = 10, 100, 1000, 10000, 100000$ by computing the relative absolute error (RAE) (note that you know the true value from Part 1, therefore use that to compute your RAE) for these N and plot it w.r.t. N.

3. (2 pts) Next choose a proposal

$$q(x) = \mathcal{N}(x; 6, 1)$$

Note that this proposal is not optimal but it will do our job. Describe the IS estimator and compute the IS estimator to estimate $p(y = 9)$. Compute the RAE for $N = 10, 100, 1000, 10000, 100000$ and plot it w.r.t. N.

4. (1 pt) Compare the RAE for MC and IS to each other (provide a plot w.r.t. N by plotting them together). What is the difference? Comment about the difference about the accuracy of these two estimators.

   Use `plt.loglog` to plot throughout this exercise to have clear figures.

Q2: Metropolis-Hastings for 1D source localisation (5 pts)

We will consider a similar example as we described in the course. Assume that we have three sensors across 1 dimensional space:

$$s_0 = -1 \qquad s_1 = 2 \qquad s_2 = 5$$

We define our likelihood as

$$p(y_i|x, s_i) = \mathcal{N}(y_i; \|x - s_i\|, \sigma_y^2).$$

for $i = 0, 1, 2$. This is a model where the hidden object at $x \in \mathbb{R}$ is observed through three sensors located as above with some noise. We also set a prior:

$$p(x) = \mathcal{N}(x; \mu_x, \sigma_x^2).$$

We aim at sampling from $p(x|y_{1:3}, s_{1:3})$ using Metropolis-Hastings. We choose a symmetric random walk proposal:

$$q(x'|x) = \mathcal{N}(x'; x, \sigma_q^2).$$

1. (2 pts) Describe the model and derive your acceptance ratio for the unnormalised posterior using your prior, likelihood, and the symmetric proposal (show how they cancel). Describe the MH algorithm.

2. (2 pts) Set $\sigma_y = 1$ (note that this is standard deviation), $\mu_x = 0$ and $\sigma_x = 10$. Set $x_{\text{true}} = 4$ and consider three data points:

$$y_0 = 4.44 \qquad y_1 = 2.51 \qquad y_2 = 0.73$$

   You will use these as your data. Implement the MH algorithm using the ratio you derived earlier. Initialise your sampler at $x_0 = 10$.

   - Plot the histogram of your samples against the true value $x_{\text{true}} = 4$. Set a suitable `burnin` period (before which you discard the samples).
   - Choose $\sigma_q = 0.1, 0.01$ and comment about the difference (how to set `burnin` iterations compared to σ_q?)

3. (1 pt) Do the previous step but now set $\sigma_y = 0.1$ (you can just use $\sigma_q = 0.1$) with $y_0 = 5.01, y_1 = 1.97, y_2 = 1.02$. What changed? Why?

Note that your posterior samples (histograms) may not exactly center at the true value, this is normal, but it will be close. Set N large enough (at least $N = 100000$ but more will look better).

Use the following code snippet to plot your posteriors (more code will be provided for some examples shortly).

```
plt.clf()
# plot the true value vertically
plt.axvline(x_true, color='k', label='true value', linewidth=2)
plt.hist(x_s[burnin:N], bins=50, density=True, label='posterior',
                                    alpha=0.5, color=[0.8, 0, 0])
plt.legend()
plt.show()
```

Here `x_s` will be your samples. You can edit this code to add/improve things as you want (xlabels, ylabels etc.). See that `burnin` period is just used when you consider your samples at the end. Therefore, you will set this value before your for loop of sampling (e.g. set 1000 that would be okay – but see the question about proposal variance vs. burnin relationship). Then you will only use it when plotting your histograms to discard initial `burnin` amount of samples. There is no other place in the algorithm where the information of `burnin` will be required.