Programming Assignment 1: Sampling and Concentration Bounds

Due September 22, 2023

The purpose of this assignment is to realize probability in randomness and to concretize error bounds as provided by concentration inequalities, in particular with Hoeffding and Glivenko-Cantelli.

1 Sampling

A probability measure $\mathbb{P}_{\mathcal{X}}$ induces, for each event $A \subset \mathcal{X}$, a random variable $\mathbb{1}_{x\in A} : \mathcal{X} \to \{0,1\}$, evaluation of which we call *sampling [of] the random variable*. Let $p_A := \mathbb{E}(\mathbb{1}_{x\in A}) = \mathbb{P}_{\mathcal{X}}(A)$. While Law of Large Numbers reasoning ensures that $p_A \approx \frac{1}{m}\sum_{j=1}^{m} \mathbb{1}_{x\in A}$ with high probability, Hoeffding quantifies both precision ($\approx$) and confidence ('high probability'):

$$\mathbb{P}_{\mathcal{X}^m}\left(\left|\frac{1}{m}\sum_{j=1}^{m}\mathbb{1}_{x\in A} - p_A\right| > \varepsilon\right) \le 2e^{-2m\varepsilon^2} \tag{1}$$

Of course, realization of event $\left\{\left|\frac{1}{m}\sum_{j=1}^{m}\mathbb{1}_{x\in A} - p_A\right| > \varepsilon\right\} \subset \mathcal{X}^m$ *itself* defines a bernoulli($p_f(m)$) random variable $f_m : \mathcal{X}^m \to \{0,1\}$ defined by

$$f_m(x_1,\dots,x_m) := \mathbb{1}_{\left|\frac{1}{m}\sum \mathbb{1}_{x_j\in A} - p_A\right| > \varepsilon}. \tag{2}$$

You will sample f_m enough to verify, up to Hoeffding confidence, that $\mathbb{E}(\mathbb{1}_{x\in A}) = p_A$.

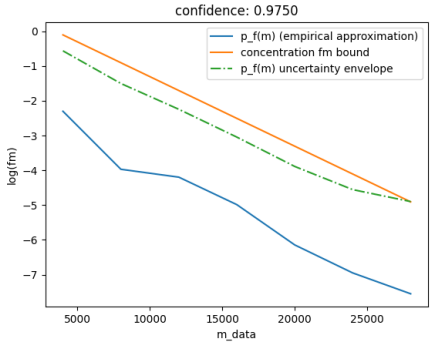
The procedure, in steps:

0. Set $\varepsilon > 0$ and $A \subset \mathcal{X}$.
1. Fix $m \in \mathbb{N}$ and sample $x_1,\dots,x_m \sim_{iid} \mathbb{P}_{\mathcal{X}}$, and evaluate the empirical mean $a_m(x_1,\dots,x_m) := \frac{1}{m}\sum_{j=1}^{m}\mathbb{1}_{x_j\in A}$ of this sample.
2. Evaluate f_m (2).
3. By Hoeffding, $p_f(m) := \mathbb{P}(f_m = 1) \le 2e^{-2m\varepsilon^2}$. Fix confidence $\delta > 0$ and run numerous experiments (executions) of f_m to verify that your empirical $\tilde{p}_f(m) \approx p_f(m)$. 'Numerous' needs to be enough to be reliable, *also* as indicated by (another!) Hoeffding, in particular, for

$$\varepsilon'(m) < 2e^{-2m\varepsilon^2} - \frac{1}{k}\sum_{j=1}^{k} f_m^j,\text{[1]} \tag{3}$$

and reasonable confidence bound $\delta \le 0.025$, i.e. using $\mathbb{P}_{\{0,1\}^k}\left(\left|\frac{1}{k}\sum_{j=1}^{k} f_m^j - p_f(m)\right| > \varepsilon'(m)\right) < \delta$

For this segment of the assignment, let $\mathbb{P}_{\mathcal{X}}$ be $\mathcal{N}(0,1)$ and $A = [0,1]$, for which $p_A \approx 0.3413447461$. Your primary output will be a plot of $\log((1))$, $\log(\tilde{p}_f(m))$, and $\log(\tilde{p}_f(m) + \varepsilon'(m))$ against m. Make sure that your uncertainty window falls under the Hoeffding bound!



[1] There is a bit of chicken and egg problem here, because you need an approximation of $p_f(m)$ first in order to find number of runs k from which you will obtain said approximation. My recommendation is the following: supposing some slop, use Hoeffding to solve for candidate k using $\varepsilon' = e^{-2m\varepsilon^2}$ (half the right hand side of (1)) in equation $\delta = 2e^{-2k\varepsilon'}$. You will need to specify (choose) $\delta > 0$ for this step. Then run your experiment with this k to generate approximation of $p_f(m)$, and use this k with your chosen δ to obtain updated $\varepsilon'(m)$ in (3).

2 Glivenko-Cantelli

In this portion of the assignment you will illustrate, in steps, a Glivenko-Cantelli-like bound (DKW) on the error of the empirical cdf:

$$\mathbb{P}_{\mathcal{X}^m}\left(\sup_{t\in\mathbb{R}}|F(t)-F_m(t)|>\varepsilon\right)\leq 2e^{-2m\varepsilon^2}. \tag{4}$$

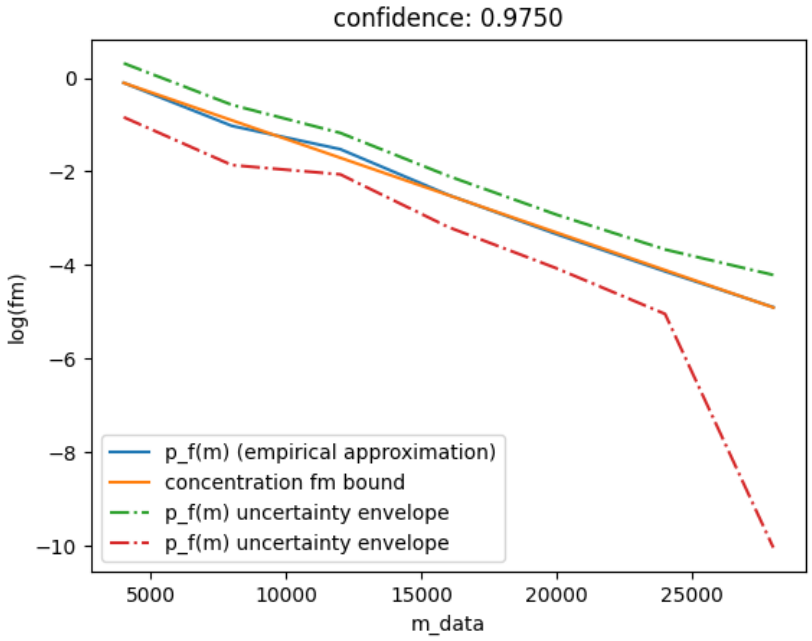
0. Set $\varepsilon > 0$.
1. Fix $m > 0$ and sample $S = (x_1,\ldots,x_m) \sim_{iid} \mathbb{P}_{\mathcal{X}}$, and define the empirical cdf $F_m(t) = \sum_{j=1}^{m} \mathbb{1}_{x_j\leq t}$.[2]
2. Evaluate f_m (5)
$$f_m(x_1,\ldots,x_m) := \mathbb{1}_{\sup_{t\in\mathbb{R}}|F_m(t)-F(t)|>\varepsilon}. \tag{5}$$
3. By (4) $p_f := \mathbb{P}(f_m = 1) \leq 2e^{-2m\varepsilon^2}$. Run numerous executions of f_m to validate this inequality. It turns out that (4) is tight, so you will not be able to trap your precision uncertainty *under* the exponential bound. However, make sure that the Hoeffding bound is at least *inside* your uncertainty window.

Notice that the term inside parentheses in both eq. (1) and (4) take the same form: the event is of all data sequences for which the absolute difference between an expectation and empirical expetation are above some threshold. The code is structured so that most of what you implement in part 1 will translate to part 2 with minimal modification. However, on this account, the code is compactified and may require careful reading through the class structure. Before beginning, consider the following hint.

To define an empirical cdf, you need to discretize your space somehow. One way is with a linear grid. It may be easier to define the *output* of the empirical cdf as a linspace instead. For this approach to make sense, you need to order your data, so that plotting the linspace against data truly shows the empirical cdf.

You will submit the same plot for this as for part 1—see below—with the proviso that your uncertainty window will not lie below the DKW bound.

As before, function calls under the main function are for your development. Remove (comment or delete) everything under and including the main function before you submit your '.py' file. Do not change the outputs returned by methods in the class (merely: update the methods so that they properly return the expected outputs). You are encouraged to use a debugger to work through the code.



[2]Note that F_m is data dependent! For a different sequence S' of length m, F_m will be different.