Lab 2

Due: Friday 09/19/2025 @ 11:59pm EST

The purpose of labs is to give you some hands on experience programming the things we've talked about in lecture. This lab will focus on learning (i.e. "fitting") probability distributions using MLE estimates. Code such as this will be used as building blocks in lots of applications, so it is important to know how to do it!

Task 0: Setup

Contained with this document is a single python file called `main.py`. Inside this file are a few classes:

- `class Distribution`. This is an abstract class (or as close as you can get in Python). This is the base class of an object heirarchy of which you will implement several child classes. This class has three abstract methods: `fit`, `prob`, and `parameters`, the purposes of which are to learn the parameters of the distribution, use the distribution to calculate the probability of data, and present the parameters of the distribution as a list respectively. The `fit` method expects an argument `X` stored inside a 2d numpy array (i.e. a matrix). Each row in this matrix is a separate example, while each column represents a feature. The `prob` method expects input of the same format, and should produce a single probability value (of that example) for each example in the input data. These probability values should be returned inside another 2d numpy array.
- `class BinomialDistribution`. This is the first class you will implement and represents random variable $X \sim Binom(n, p)$.
- `class PoissonDistribution`. This is the second class you will implement and represents random variable $X \sim Poisson(\lambda)$.
- `class GaussianDistribution` (optional, currently commented out). This is the extra credit and represents random variable $X \sim \mathcal{N}(\mu, \sigma^2)$.

Task 1: `class BinomialDistribution` (50 points)

A Binomial distribution has two parameters: `n` and `p`, where `p` is the probability of a single trial being in the "success" state, and `n` is the number of trials. A Binomial distribution models the number of successes that occur in those `n` trials. In your `fit` method, be sure to calculate `p` using the MLE estimate discussed in lecture, and be sure to use this value of `p` in your `prob` method!

Task 2: `class PoissonDistribution` (50 points)

A Binomial distribution has one parameter: λ, where λ is the rate of (fair) trials occuring. A Poisson distribution models how long it will take for the first successful trial to occur. In your `fit` method, be sure to calculate λ using the MLE estimate discussed in lecture, and be sure to use this value of λ in your `prob` method!

Task 3: Extra Credit `class GaussianDistribution` (50 points)

A Gaussian distribution has two parameters: μ, and σ^2, where μ is the center of the distribution, while σ^2 is the spread. A Gaussian distribution models how likely an example is to occur given how far away it is from the center of the distribution according to the spread. In your `fit` method, be sure to calculate μ and σ^2 using the MLE estimate discussed in lecture, and be sure to use these values of μ and σ^2 in your `prob` method!

Task 4: Submit Your Lab

To complete Lab1, please **only turn in the `main.py` file** on Gradescope. You shouldn't have to worry about zipping it up or anything, just drag and drop it in.