232C: Assignment 1

1 Introduction

For this assignment, you will perform Bayesian inference to find a posterior, given a prior and likelihood. A and B are two independent events with discrete outcomes. You do not know how many outcomes A and B have or what those outcomes are named, but can assume each outcome occurs with some non-zero probability. The given priors rep Additionally, the given likelihood is constructed from a sample of observed data. As a result, you do not need the data to find the posterior.

2 Format and Requirements

You will provided with 'BayesianInference.py' with the function 'getPosterior()', which you need to complete. In order to receive full credit, your submission script must be named 'BayesianInference.py' and and have the method 'getPosterior()' which returns the posterior in the correct format. You do not need to generate your own priors and likelihoods, an example will be included. However, your function should be able to calculate the posterior for any prior/likelihood with discrete outcomes of the specified input format.

'getPosterior()' has:

- **Inputs:**
  - **priorOfA:** P(A), A dictionary where keys are all possible outcomes in A and values are their prior probabilities. A PMF.
  - **priorOfB:** P(B), A dictionary where keys are all possible outcomes in B and values are their prior probabilities. A PMF.
  - **likelihood:** $P(D|A, B)$. A dictionary where the keys are tuples of representing all possible joint outcomes of A and B. Values are the respective likelihoods generated from sampling the data.
- **Outputs:** a list containing the marginal posterior of A as the first element and the marginal posterior of B as the second element.
  - **marginalPosteriorOfA:** $P(A|Data)$, A dictionary where the keys are the outcomes in A and the values are the posterior probabilities of the respective outcomes. A PMF.
  - **marginalPosteriorOfB:** $P(B|Data)$, A dictionary where the keys are the outcomes in B and the values are the posterior probabilities of the respective outcomes. A PMF.