Exercise: Complexity of Computing Bayes' Theorem for Random Variables

(1/1 point)

This exercise is extremely important and gets at how expensive it is to compute a posterior distribution when we have many quantities we want to infer.

Consider when we have N random variables $X_1, \dots, X_N$ with joint probability distribution $p_{X_1,\dots,X_N}$, and where we have an observation Y related to $X_1, \dots, X_N$ through the known conditional probability table $p_{Y|X_1,\dots,X_N}$. Treating $X = (X_1, \dots, X_N)$ as one big random variable, we can apply Bayes' theorem to get

$$p_{X_1,X_2,\dots,X_N|Y}(x_1, x_2, \dots, x_N \mid y)$$
$$= \frac{p_{X_1,X_2,\dots,X_N}(x_1, x_2, \dots, x_N)p_{Y|X_1,X_2,\dots,X_N}(y \mid x_1, x_2, \dots, x_N)}{\sum_{x'_1}\sum_{x'_2}\cdots\sum_{x'_N} p_X(x'_1, x'_2, \dots, x'_N)p_{Y|X_1,X_2,\dots,X_N}(y \mid x'_1, x'_2, \dots, x'_N)}$$

- Suppose each X_i takes on one of k values. In the denominator, how many terms are we summing together? Express your answer in terms of k and N.

  In this part, please provide your answer as a mathematical formula (and not as Python code). Use ^ for exponentiation, e.g., x^2 denotes x^2. Explicitly include multiplication using *, e.g. x*y is xy.

k^N

Answer: k^N

k^N

Solution:

- Suppose each X_i takes on one of k values. In the denominator, how many terms are we summing together? Express your answer in terms of k and N.
- **Solution:** We are summing out over every possible configuration of x'_1, x'_2, up to x'_N. Each of these takes on k different possibilities, so the number of possible configurations is $\boxed{k^N}$.
- A computational way to think about this: The denominator is computed as a sum. Let's start from this sum being equal to 0. We next have N nested for loops. The outer-most for loop is over x'_1 and iterates over k possible values, the next for loop is over x'_2 and also iterates over k

Mini-project 1: Movie Recommendations (Week 3)
Mini-projects due Oct 12, 2016 at 21:00 UTC

values, and so forth. Finally at the inner-most for loop, we can specify the single new additional term that we're adding to the overall sum. The number of terms being added is going to be:

- k (from for loop over x'_1)
  $\times k$ (from for loop over x'_2)
  $\times k$ (from for loop over x'_3)
  $\vdots$
  $\times k$ (from for loop over x'_N)
- In particular, we have an N-fold multiplication to get k^N.
- **Important take-away message:** The number of terms being summed grows exponential in the number of variables we are inferring N. Without any sort of additional structure in the distribution, it turns out that we cannot hope to escape this exponential cost in computing the posterior distribution.
- This is a disaster! In many problems we care about, N will be very, very large! For example, if $X_1, \dots, X_N$ represents values that different pixels in an image take, then nowadays images taken for example on a mobile phone often have easily well over 10 million pixels. So N could be 10 million, and even if each X_i took on $k = 2$ values, the number of terms we would have to sum over in the denominator is already greater than the number of atoms in the known, observable universe (which is estimated to be somewhere between 10^{78} and 10^{82}).
- Structure in distributions will help us escape from this exponential cost in N.

You have used 1 of 5 submissions