Problem Set 6

Due Friday Nov. 4, 10 am

Comments

- This covers Units 8 and 9.
- It's due at 10 am (Pacific) on November 4, both submitted as a PDF to Gradescope as well as committed to your GitHub repository.
- Please see PS1 and the grading rubric for formatting and attribution requirements.

Problems

1. Consider the following estimates of the variance of a set of numbers. The results depend on whether the magnitude of the numbers is large or small. You can assume that for a vector `w`, var(w) is calculated as $\sum_{i=1}^{n}(w_i - \bar{w})^2/(n-1)$.

```
set.seed(1)
dg <- function(x) formatC(x, 20, format = 'g')
z <- rnorm(100, 0, 1)
x <- z + 1e12
## Calculate the empirical variances
dg(var(z))
```

```
[1] "0.80676208969370799551"
```

```
dg(var(x))
```

```
[1] "0.80675835877355905 89"
```

   Explain why these two estimates agree to only a small number of decimal places and which of the two is the more accurate answer, when mathematically the variance of `z` and the variance of `x` are exactly the same (since `x` is just the addition of a constant to `z`).

2. Consider the following, in which we run into problems when trying to calculate on a computer. Suppose I want to calculate a predictive density for new data (e.g., in a model comparison in a Bayesian context):

$$f(y^*|y,x) = \int f(y^*|y,x,\theta)\pi(\theta|y,x)d\theta = E_{\theta|y,x}f(y^*|y,x,\theta).$$

Here $\pi(\theta|y,x)$ is the posterior distribution (the distribution of the parameter, θ, given the data, y, and predictors, x). All of θ, y, and x will generally be vectors.

If we have a set of samples for θ from the posterior distribution, $\theta_j \sim \pi(\theta|y,x)$, $j = 1, \ldots, m$, we can estimate that quantity for a vector of conditionally IID observations using a Monte Carlo estimate of the expectation:

$$f(y^*|y,x) \approx \frac{1}{m}\sum_{j=1}^{m}\prod_{i=1}^{n} f(y_i^*|y,x,\theta_j).$$

a. Explain why I should calculate the product in the equation above on the log scale. What is likely to happen if I just try to calculate it directly?

b. Here's a re-expression, using the log scale for the inner quantity,

$$\frac{1}{m}\sum_{j=1}^{m}\exp\sum_{i=1}^{n}\log f(y_i^*|y,x,\theta_j),$$

which can be re-expressed as

$$\frac{1}{m}\sum_{j=1}^{m}\exp(v_j)$$

where

$$v_j = \sum_{i=1}^{n}\log f(y_i^*|y,x,\theta_j).$$

What is likely to happen when I try to exponentiate v_j?

c. Consider the log predictive density,

$$\log f(y^*|y,x) \approx \log\left(\frac{1}{m}\sum_{j=1}^{m}\exp(v_j)\right).$$

Figure out how you could calculate this log predictive density without running into the issues discussed in parts (a) and (b).

Hint: recall that with the logistic regression example in class, we scaled the problematic expression to remove the numerical problem. Here you can do something similar with the $\exp(v_j)$ terms, though at the end of the day you'll only be able to calculate the log of the predictive density and not the predictive density itself. If you're having trouble thinking about it abstractly using the v_j notation, think about what you would do if you had a few actual numbers for v_j.

3. Experimenting with importance sampling.

a. Use importance sampling to estimate the mean (i.e., $\phi = E_f X$) of a truncated t distribution with 3 degrees of freedom, truncated such that $X < -4$. Have your sampling density be a normal distribution centered at -4 and then truncated so you only sample values less than -4 (this is called a half-normal distribution). You should be able to do this without discarding any samples (how?). Use $m = 10000$ samples. Create histograms of the weights $f(x)/g(x)$

to get a sense for whether $\text{Var}(\hat{\phi})$ is large. Note if there are any extreme weights that would have a very strong influence on $\hat{\phi}$. Estimate $\text{Var}(\hat{\phi})$. Hint: remember that your $f(x)$ needs to be appropriately normalized or you need to adjust the weights per the class notes.

b. Now use importance sampling to estimate the mean of the same truncated t distribution with 3 degrees of freedom, truncated such that $X < -4$, but have your sampling density be a t distribution, with 1 degree of freedom (not 3), centered at -4 and truncated so you only sample values less than -4. Again you shouldn't have to discard any samples. Respond to the same questions as above in part (a). In addition, compute a 95% uncertainty interval for your estimate, using the Monte Carlo simulation error, $\sqrt{\widehat{\text{Var}}(\hat{\phi})}$.

4. Extra credit: This problem explores the smallest positive number that R can represent and how R represents numbers just larger than the smallest positive number that can be represented. (Note: if you did this in Python you'd get the same results.)

a. By experimentation in R, find the base 10 representation of the smallest positive number that can be represented in R. Hint: it's rather smaller than 1×10^{-308}.

b. Explain how it can be that we can store a number smaller than 1×2^{-1022}, which is the value of the smallest positive number that we discussed in class. Start by looking at the bit-wise representation of 1×2^{-1022}. What happens if you then figure out the natural representation of 1×2^{-1023}? You should see that what you get is actually a well-known number that is not equal to 1×2^{-1023}. Given the actual bit-wise representation of 1×2^{-1023}, show the progression of numbers smaller than that that can be represented exactly and show the smallest number that can be represented in R written in both base 2 and base 10.

Hint: you'll be working with numbers that are not normalized (i.e., denormalized; numbers that do not have 1 as the fixed number before the decimal point in the floating point representation we discussed in Unit 8.