STAT 578 – Advanced Bayesian Modeling – Fall 2019

Assignment 4

According to one version of Moore's law, the number of transistors on a state-of-the-art computer microprocessor roughly doubles every two years:

$$T \approx C\, 2^{A/2}$$

where T is the number of transistors, A is the year number in which the microprocessor was introduced, and C is a positive constant.

File `mooreslawdata.csv` is a comma-separated values (CSV) file containing the name, transistor count, and year of introduction for 161 microprocessors.[1]

(a) Consider the *natural logarithm* of the number of transistors: $\log T$.

- (i) [2 pts] Demonstrate that, according to Moore's law, $\log T$ should roughly follow a simple linear regression on A. Also, what should be the value of the coefficient of A?
- (ii) [2 pts] Plot the data points as *log* transistor count versus year.

(b) Consider a normal-theory simple linear regression model of log transistor count on *centered* year of the form

$$\log T_i \mid \beta, \sigma^2, A_i \;\sim\; \text{indep. } \mathrm{N}\big(\beta_1 + \beta_2(A_i - \bar{A}),\, \sigma^2\big) \qquad i = 1, \ldots, 161$$

where $\bar{A}$ is the average of A_i over all observations. Of course, Moore's law specifies a particular value for β_2, but your initial model will not assume this. Use independent priors

$$\beta_1, \beta_2 \sim \text{iid } \mathrm{N}\big(0,\, 1000^2\big)$$
$$\sigma^2 \sim \text{Inv-gamma}(0.001,\, 0.001)$$

- (i) [2 pts] List an appropriate JAGS model.

  Now run your model. Make sure to use multiple chains with overdispersed starting points, check convergence, and monitor β_1, β_2, and σ^2 for at least 2000 iterations (per chain) after burn-in.

- (ii) [2 pts] List the `coda` summary of your results for β_1, β_2, and σ^2.
- (iii) [3 pts] Give the approximate posterior mean and 95% posterior credible interval for the slope. Does the interval contain the value you determined in part (a), in accordance with Moore's law?
- (iv) [2 pts] Give the approximate posterior mean and 95% posterior credible interval for the intercept.

[1] Data from Wikipedia contributors. (2019, July 5). Transistor count. In *Wikipedia, The Free Encyclopedia*. Retrieved 20:39, July 5, 2019, from `https://en.wikipedia.org/w/index.php?title=Transistor_count&oldid=904925496`

(c) Consider the model of the previous part. You will use it to predict the transistor count on a microprocessor introduced in 2020, and also (just for fun) to see if it extrapolates back to the invention of the transistor.

(i) [2 pts] List a modified JAGS model appropriate for answering the subparts below.

Now run your model. Make sure to use multiple chains with overdispersed starting points, check convergence, and monitor parameters for at least 2000 iterations (per chain) after burn-in.

(ii) [2 pts] List the `coda` summary you will use to help answer the subparts below.

(iii) [2 pts] Give an approximate 95% posterior *predictive* interval for the transistor count, in *billions*, on a microprocessor introduced in the year 2020. (Note: This is for the count, *NOT* the log count.)

(iv) [3 pts] Explain why the model suggests that the transistor was invented in the year

$$\bar{A} \; - \; \beta_1/\beta_2$$

and give an approximate 95% posterior interval for this quantity. (You may compare this to the actual year in which the transistor was invented.)

(d) One way to check for evidence of outliers is a posterior predictive p-value based on test quantity

$$T(y, X, \theta) \;\; = \;\; \max_i |\varepsilon_i / \sigma|$$

where ε_i is the error for observation i. The larger this quantity is, the more we should suspect the existence of an outlier.

Use your JAGS model from part (b). (Suggestion: Apply `as.matrix` to the output of `coda.samples` to obtain a matrix of simulated parameter values.)

(i) [2 pts] Show R code for computing the simulated error vectors ε (as rows of a matrix).

(ii) [2 pts] Show R code for computing simulated *replicate* error vectors ε^{rep} (as rows of a matrix), which are the error vectors for the replicate response vectors y^{rep}.

(iii) [2 pts] Show R code for computing the simulated values of $T(y, X, \theta)$ and the simulated values of $T(y^{\text{rep}}, X, \theta)$.

(iv) [2 pts] Plot the simulated values of $T(y^{\text{rep}}, X, \theta)$ versus those of $T(y, X, \theta)$, with a reference line indicating where $T(y^{\text{rep}}, X, \theta) = T(y, X, \theta)$.

(v) [2 pts] Compute the approximate posterior predictive p-value, and make an appropriate conclusion based on it. (Is there evidence for an outlier?)

(vi) [1 pt] Name the microprocessor that appears to be the most extreme outlier.

Total: 33 pts