MTH225 Spring2017 Final Problem 6

Data from a bacteria culture gives the approximate cell count (in thousands) by day. It is assumed that the growth is exponential, that is, the population $P(t)$ after t days is given by:

$$P(t) = P(0) \cdot \exp(a + bt)$$

Taking logarithms, we get

$$\ln(P(t)) = \ln(P(0)) + a + bt$$

Combining $\ln(P(0))$ and a into α, and labeling $\ln(P(t))$ as y gives us a familiar simple regression equation

$$y_i = \alpha + bt_i$$

We can superimpose some Gaussian noise to get a regression model

$$y_i = \alpha + bt_i + e_i \quad \text{with} \quad e_i \sim N(0, \sigma_e)$$

If we want to recover the original counts, we would exponentiate both sides:

$$P(t_i) = e^{\alpha + bt_i + e_i}$$

The data in `MTH225_Spring2017_Final_Problem6.csv` contains daily measurements of $P(t)$.

The objective of this exercise is to fit a simple regression to estimate a and b and use them to compute a confidence interval for the count on day 40.

The variable names are:

- `date` number of days since the start of the culture
- `count` approximate cell count on (in thousands) on that day

You can use the Stan model file `simple_regression.stan` as a starting point.

- 2 points: Write R code to read the data and convert it to an R data frame, and perform the log transform on y.
- 1 point: Write the data block of a STAN model file that extracts the data from the R workspace.

- 1 point: Write the parameters block of a STAN model file that declares the paramter(s) of your model.
- 2 points: Write the model block of a STAN model file that specifies the priors and likelihood for your model.
- 1 point: Write R code to apply the `extract` function to the data structure output from the `stan` function.
- 1 point: Use the `extract()` function of the RSTAN package to obtain the values for the parameters from the posterior draw.
- 1 point: Use the posterior draw to construct to construct a simulated $\ln(\hat{y})$ value at day 40 as:

  ```
  log_y_hat = rnorm(pd$alpha+40*pd$beta,pd$sigma)
  ```

  for each (a, b) pair in the posterior draw, then convert to counts ($\hat{y}$) with

  ```
  exp(log_y_hat)
  ```

- 1 point: Use the `quantile` function on the simulated $\hat{y}$ values to compute a 95% confidence interval for $\hat{y}$.

(10 points possible)