Self Assessment

Self Assessment

Today, you'll be completing a short assessment so that we can get a sense of where you're at.

- Type your answers in a new R script file with comments indicating where the answer to each question begins.
- Write down the current time. Please email us (at `signaldatascience@gmail.com`) with your R script attached after 90 minutes have passed.
- Work individually. You can however consult R documentation, look at old assignments, use the Internet, etc., but don't copy and paste code verbatim.
- Make your code as clear, compact, and efficient as possible. Use everything that you've learned!

Part 1

Here's an interview question from *Euclid Analytics*:

> Suppose that X is uniformly distributed over $[0, 1]$. Now choose $X = x$ and let Y be uniformly distributed over $[0, x]$. Is it possible for us to calculate the "expected value of X given $Y = y$", *i.e.*, $\mathbb{E}(X|Y = y)$?

Now, we don't know the answer yet, but maybe we can get some sense of what it might look like by doing some Monte Carlo simulations. To that end:

- A *single trial* of the process described in the problem should return a pair of random values (x, y). Simulate an arbitrary number of trials of this process.
  - Plot the simulated values with `qplot()` (in the `ggplot2` library).

- Since we're interested in the *expected value* of X given some $Y = y$, we can approximate this by separating our values of Y into *bins* and taking the *mean* of X within each bin.
  - Write code to do so and use `qplot()` to view the results. Do they make sense?

Now, suppose that a magic fairy whispers into your ear:

> Here's the answer, my friend! It just so happens that $\mathbb{E}(X|Y = y) = \frac{y-1}{\ln y}$!

In light of this revelation, you want to verify your computational results from earlier. To that end:

- Generate a lot of different values of Y and calculate the corresponding values of $\mathbb{E}(X|Y = y)$ according to the equation above.
  - Graph them using `qplot()`. Does this graph match your simulated results?
- Make a *single* dataframe with both your Monte Carlo-simulated results *and* your direct calculation of the theoretical result.
  - Make a single graph with (1) a scatterplot of the Monte Carlo-simulated results and (2) a smooth line connecting the points corresponding to the theoretical values.