Problem Set 2
Econometric Theory
Due on March, 1st 2017

1. [Julia] Write a code for evaluating the following two sums

   (a) \[ y = \sum_{i=0}^{1000} 0.5^i \text{ for } i \text{ even} \]

   (b) \[ z = \sum_{i=1}^{1001} 0.5^i \text{ for } i \text{ odd} \]

2. [Julia] Code the following Algorithm in Julia

   (a) Set \( j = 1 \)

   (b) For each \( i \leq N \) draw \( x_i^{(j)} \sim F(\cdot) \) (where \( F(\cdot) \) is a distribution)

   (c) Calculate \( \bar{x}^{(j)} = \frac{\sum_{i=1}^{N} x_i}{N} \)

   (d) If \( j < M \), set \( j = j + 1 \) and go to step 2, else go to step 5

   (e) Plot the sample distribution of \( \bar{x}^{(j)} \)

Run the algorithm assuming that \( F \) is the distribution of (i) a \( N(0, 1) \); (ii) a \( \chi^2_2 \); (iii) a Cauchy; (iv) Pareto with \( \alpha = 1 \); (v) Pareto with \( \alpha = 3 \); (vi) Binomial with \( p = 0.49 \).

3. [Julia] Code the following Algorithm in Julia

   Step 1 Set \( j = 1 \).

   Step 2 For each \( i \leq N \), draw \( u_i \sim (\chi^2_2 - 2)/2 \), \( x_i = z_i^2 \) where \( z_i \sim N(5, 1) \), and set

   \[ y_i = \beta_0 + \beta_1 x_i + u_i \]

   where \( \beta_0 = 0.1 \) and \( \beta_1 = 0.4 \).

   Step 3 Calculate \( \hat{\beta}_1^{(j)} \) and \( \hat{\beta}_2^{(j)} \) and an estimate of their asymptotic variance covariance matrix, \( \hat{V}^{(j)} \).

   Step 4 If \( j < M \), set \( j = j + 1 \) and go to step 2, else go to step 5

   Step 5 Plot the sample distribution of \( \sqrt{N\hat{\beta}_1^{(j)}/\sqrt{\hat{V}_{11}}} \) and \( \sqrt{N\hat{\beta}_2^{(j)}/\sqrt{\hat{V}_{22}}} \).

Run the algorithm for \( M = 100000 \) and \( N = 20 \), \( N = 50 \) and \( N = 200 \).
Note

In Julia plotting can be done using the Plots.jl package. In JuliaBox is already installed. Running the code below should clarify how to use Plots.jl.

```julia
using Plots  ## Load the package
x = randn(1000); ## generate 1000 draws from a standard normal
Plots.histogram(x) ## Produce the histogram of x
Plots.histogram(x, bins = 30) ## use 30 bins
Plots.histogram(x, bins = 30, normalize = true) ## use 30 bins and normalized to integrate
y = 0.1 + 0.2*x + randn(1000) ## produce fake regression data
## Scatterplot
Plots.plot(x,y)
```

A Jupyter notebook that illustrates these commands can be downloaded [here.](#)