RANDOM NUMBER GENERATION

Quantitative Risk Management project work

Silvia Baldisserotto, Maysa Jahanbani,
Claudia Pesci, Phan Ho Tan Phat,
Andrea Venuta

Università degli Studi di Firenze - Finance and Risk Management

IMPLEMENTATIONS - LINEAR CONGRUENTIAL GENERATOR

```
1  function [ rn ] = LCG( x )
2
3    if(nargin == 0)
4      x = 1;
5    end
6
7    rn = zeros(x,1);
8
9    for i = 1:x
10     rn(i) = LCGstep();
11   end
12
13 end
```

```
14 function [ rnStep ] = LCGstep()
15
16   persistent seed;
17   M = 244944;
18   a = 1597;
19   b = 51749;
20
21   if(isempty(seed))
22     seed = 0;
23   end
24
25   seed = mod(seed * a + b, M);
26
27   rnStep = seed / M;
28
29 end
```

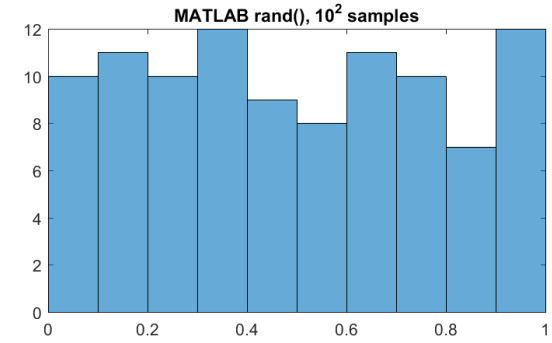
IMPLEMENTATIONS - BOX-MULLER METHOD

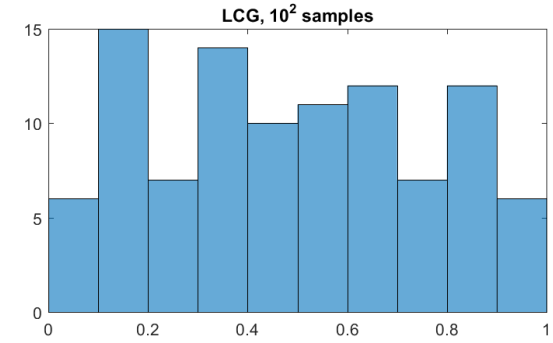
```
1  function [ Z ] = BoxMuller( x )
2
3    if(nargin == 0)
4      x = 1;
5    end
6
7    U = rand(x, 2);
8
9    theta = 2 .* pi .* U(:, 2);
10   rho   = sqrt( -2 .* log( U(:, 1) ) );
11
12   Z = [ rho .* cos(theta), rho .* sin(theta) ];
13
14 end
```

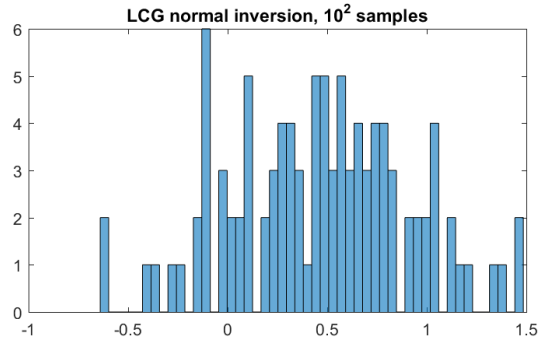
IMPLEMENTATIONS - MARSAGLIA POLAR ALGORITHM

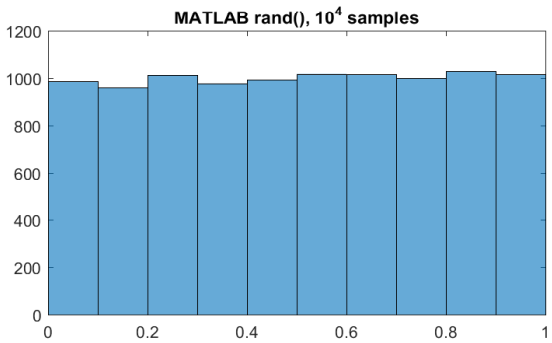
```
function [ Z ] = Marsaglia( x )

  if(nargin == 0)
    x = 1;
  end

  Z = zeros(x,2);

  for i = 1 : x
    W = 1;  V = [ 1, 1 ];
    while not (W < 1)
      V = 2 * rand(1, 2) - 1;
      W = V(1) .^ 2 + V(2) .^ 2;
    end

    Z(i, :) = V .* sqrt(-2 * log(W) / W);
  end
end
```

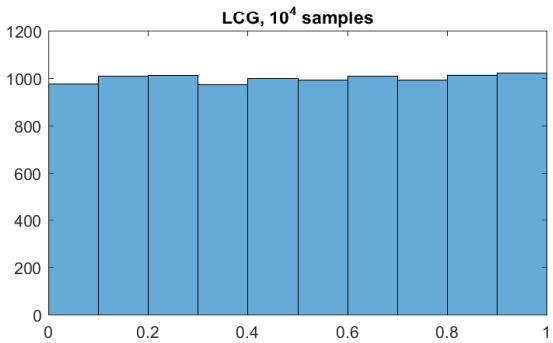
PLOTS - UNIVARIATE METHODS

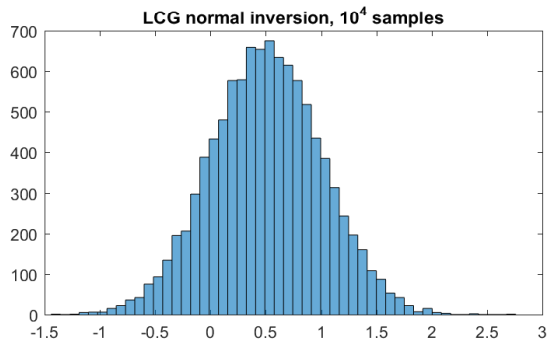


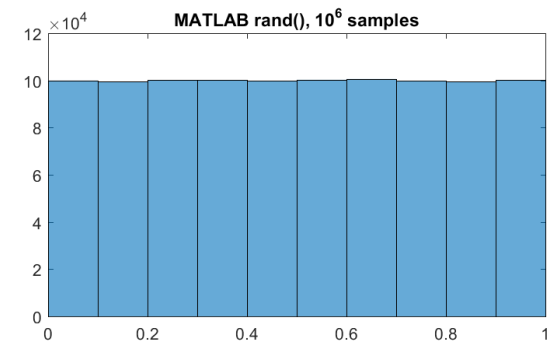


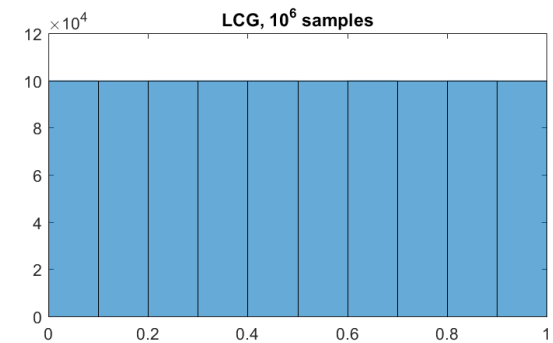


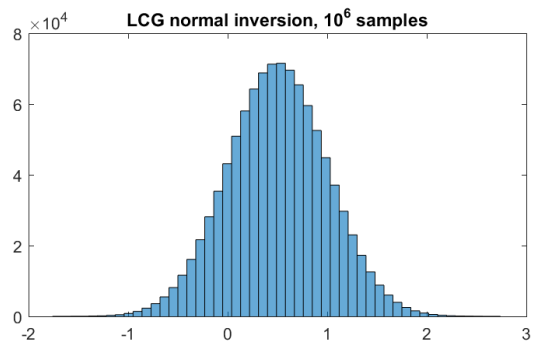












PLOTS - BIVARIATE METHODS