
Project one. Problem one. Mathematics 502 Probability and Statistics

Nasser Abbasi, September 26, 2007. California State University, Fullerton

1. [10 points] The distribution function for the exponential distribution with parameter λ is given by

$$F(x) = \begin{cases} 1 - e^{-\lambda x} & x \geq 0 \\ 0 & \text{Otherwise.} \end{cases}$$

- a. Write an R program that uses F^{-1} and the uniform random number generator in R (`runif`) to generate n numbers from the exponential distribution with λ . Use `set.seed(your birthday MMDDYY)`.
- b. Generate $n = 10000$ values from the exponential random variable with parameter $\lambda = 2$, using your program (don't give me the numbers generated!). Use the `hist` command in R to plot the *density* histogram of the *relative frequencies* for the generated data. Overlay this histogram by the graph of the density of the exponential random variable with parameter $\lambda = 2$. The commands `dexp` and `lines`, or `curves` can be used. Select an appropriate number of bins for your histogram. Briefly comment on the relationship between the histogram and the density curve.

Problem 1 part (a)

The CDF given is defined as $F(x) = \begin{cases} 1 - e^{-\lambda x} & x \geq 0 \\ 0 & \text{o.w.} \end{cases}$ To find F^{-1} we need to solve for x in the equation $1 - e^{-\lambda x}$ for $x \geq 0$ Hence we write

$$\begin{aligned} y &= 1 - e^{-\lambda x} \\ e^{-\lambda x} &= 1 - y \\ -\lambda x &= \ln(1 - y) \\ x &= \frac{-1}{\lambda} \ln(1 - y) \end{aligned}$$

Therefore

$$F^{-1}(y) = \frac{-1}{\lambda} \ln(1 - y)$$

Now to generate random numbers which belongs to an exponential distribution, we will now generate random numbers from $U(0, 1)$ and for each such number generated, we will apply the above function F^{-1} on it, and the result will be a random number which belongs to the exponential distribution. For example, if $\lambda = 2$ and a uniform random number is say 0.4, then we evaluate $F^{-1}(0.4) = \frac{-1}{2} \ln(1 - 0.4) = 0.25541$

And so this is the idea to implement. We need to first seed the uniform random number generator before we start.

Algorithm

Input: λ : parameter, n : number of random numbers to generate

output: a list of n random numbers from the probability density function $\sim F(x)$ given above.

1. Seed the uniform random number generator with (010101).
2. initialize the array `d` of size n which will contain the list of random numbers generated below.

This loop below is just an algorithmic view. In actual code, a 'vector' operation `Table[]` is used for speed.

3. For i in $1..n$ LOOP

 Generate y_k which is a random generated from uniform distribution using the built in function `RandomReal[0,1]`

$d[i] = F^{-1}(y_k)$ using input λ .

END LOOP

4. Find histogram of d . Select an appropriate number of bins. Let f_a be the histogram found.
5. Now find the relative frequency f_r by dividing set f_a by the number of observations n . Hence histogram now is $f_r = \frac{f_a}{n}$
6. Now scale the histogram such that it is density. Total area is 1. Do this by finding total area under histogram, and divide each bin count by this area.
7. Plot the histogram and the exponential distribution $\lambda e^{-\lambda x}$ on the same plot.

Code Implementation

Define the function F^{-1} which was derived earlier. This is the inverse of the CDF of the exponential density function $\lambda e^{-\lambda x}$

```
In[156]:= Remove["Global`*"];
gDebug = False;
```

```
In[158]:= inverseCDFofExponentialDistribution[λ_, n_] := Module[{i},
  -1/λ Log[1 - n]]
```

Function below is called to generate N random numbers using the above F^{-1} function (User needs to seed before calling

```
In[159]:= getRandomNumbersFromExponential[λ_, nRandomVariables_] := Module[{i},
  Table[inverseCDFofExponentialDistribution[λ, RandomReal[]], {i, nRandomVariables}]]
```

Problem 1 part(b)

Generate $n = 10000$ for $\lambda = 2$ and overlay with relative frequency, use appropriate number of bins. See appendix for the function `postProcessForPartOne[]` which generate the plots. Removed below to reduce code clutter in the main report.

This function makes a histogram which is scaled to be used to overlay density plots, or other functions.

Input: `originalData`: this is an array of numbers which represents the data to bin

`nBins`: number of bins

output: the histogram itself but scaled such that area is ONE

```
In[160]:= Needs["BarCharts`"]
mmaMakeDensityHistogram[originalData_, nBins_] :=
Module[{freq, binSize, from, to, scaleFactor, j, a, currentArea},
  to = Max[originalData];
  from = Min[originalData];
  binSize = (to - from) / nBins;
  freq = BinCounts[originalData, binSize];
  currentArea = Sum[binSize * freq[[i]], {i, nBins}];

  freq =  $\frac{\text{freq}}{\text{currentArea}}$ ;
  a = from;
  Table[{a + (j - 1) * binSize, freq[[j]], binSize}, {j, 1, nBins}]
]
```

This function to overlay the histogram and the PDF. It is used by the simulation program as well (that is why it is a little larger than needed)

```

In[162]:= postProcessForPartOne[randomNumbers_, nBins_, λ_, nRandomVariables_, fromX_, toX_] :=
Module[{frequency, relativeFrequency, p1, p2,
  x, factor, max, imSize = 300, dx, p, pCDF, pinvCDF, gz, g},

  (*find dx which is bin size, needed by Mathematica BinCount function*)
  dx = (toX - fromX) / nBins;

  (*generate frequency count using the above bin size*)
  frequency = BinCounts[randomNumbers, {fromX, toX, dx}];

  (*now normalize by total number of observation to obtain the relative frequencies*)
  relativeFrequency = N[frequency / nRandomVariables];

  (*Now divide by scale factor λ, to scale it *)
  max = Max[relativeFrequency];
  factor = λ / max;
  relativeFrequency = relativeFrequency * factor;

  gz = nmaMakeDensityHistogram[randomNumbers, nBins];
  p1 = GeneralizedBarChart[gz, BarStyle → White,
    ImageSize → imSize, PlotLabel → "λ=" <> ToString[λ] <> " variables=" <>
      ToString[nRandomVariables] <> " bins=" <> ToString[nBins]];

  p2 = Plot[PDF[ExponentialDistribution[λ], x], {x, fromX, toX},
    PlotRange → All, Frame → True, PlotStyle → {Red(*,Thick*)}, ImageSize → imSize];

  p = Show[{p1, p2}];

  pinvCDF = Plot[inverseCDFofExponentialDistribution[λ, y], {y, 0, λ},
    PlotLabel → "x=F-1(y) =  $-\frac{1}{\lambda} \log[1-y]$ ", ImageSize → 200, AxesLabel → {"y", "x"}];

  pCDF = Plot[1 - Exp[-λ x], {x, fromX, toX},
    PlotLabel → "y=F(x) = 1 - e-λx", ImageSize → 200, AxesLabel → {"x", "y"}];

  g = Grid[{pinvCDF, pCDF}];
  Grid[{p}, {g}], Alignment → {{Center}, {Center}}, Frame → All]
]

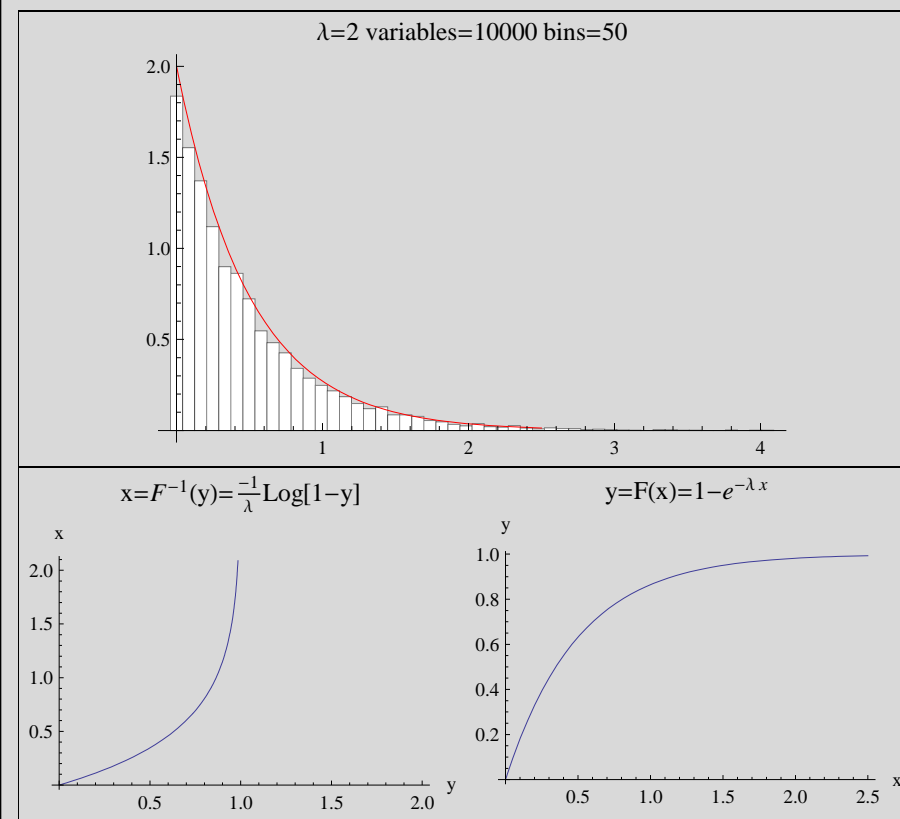
```

now generate the needed outout for N = 10000

In[163]:=

```
nBins = 50; λ = 2; numberOfVariables = 10 000; fromX = 0; toX = 2.5;  
postProcessForPartOne[getRandomNumbersFromExponential[λ, numberOfVariables],  
nBins, λ, numberOfVariables, fromX, toX]
```

Out[164]=



Comment and analysis

Below I show snap shots of few plots of the density overlaid with the histogram for different values of n which is the number of random variables.

We see from the plots below, that for a fixed number of bins, fixed λ , that as more random variables are generated, the histogram overlaid on top of the actual PDF becomes closer and closer to the PDF curve. The error between the histogram and the PDF curve becomes smaller the larger the number of random variables used. This indicates that this method of finding random numbers for density function will converge to the density function. We need to select an appropriate bin size to see this more clearly. The smaller the bin size the more clear this will become (but too small a bin size will make the histogram itself not too clear).

Please see appendix for additional GUI based simulation for this part of the project.

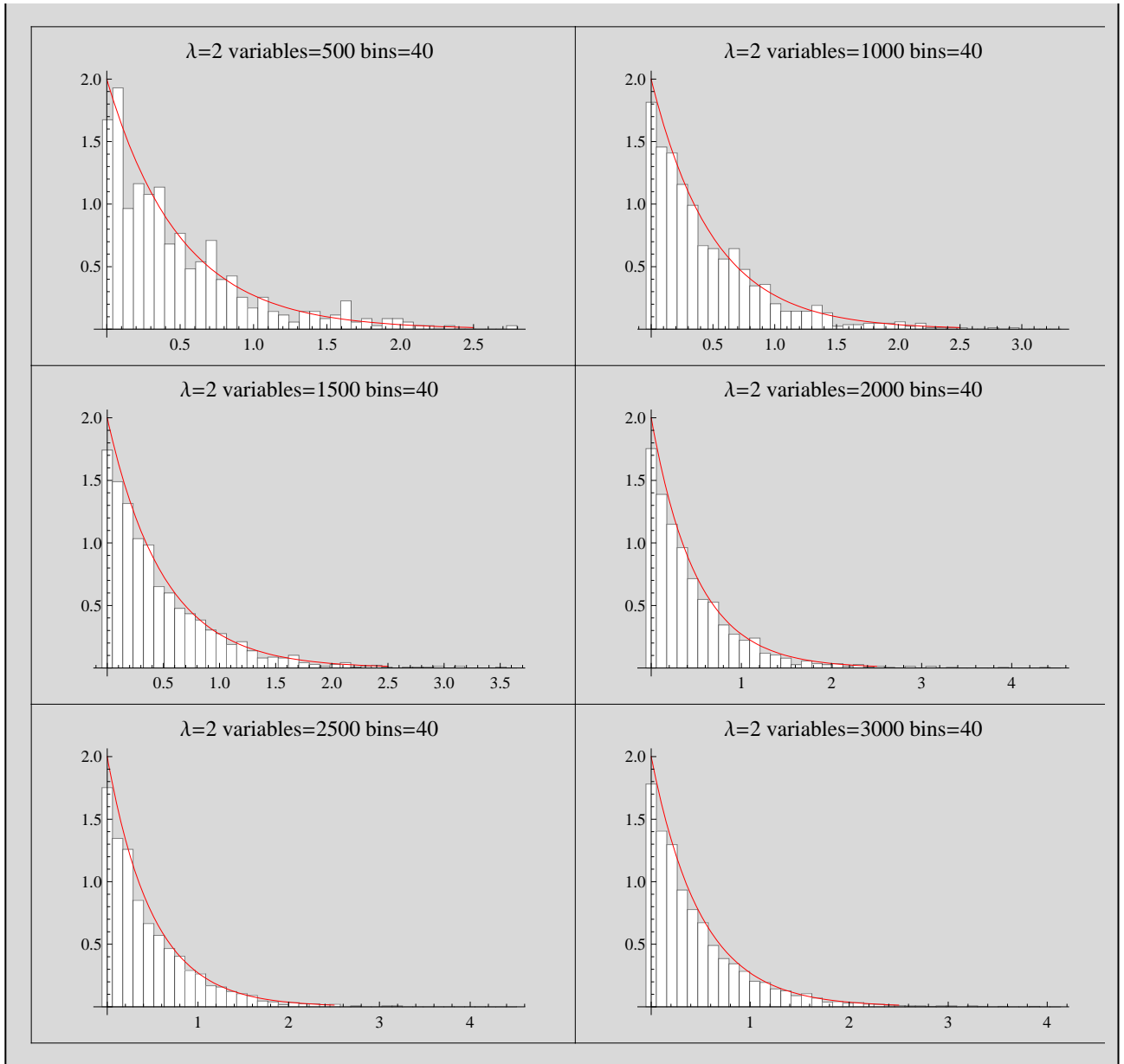
In[165]:=

```

nBins = 40;  $\lambda$  = 2; fromX = 0; toX = 2.5;
SeedRandom[010101];
p = Table[postProcessForPartOne[getRandomNumbersFromExponential[ $\lambda$ , nRandomVariables],
  nBins,  $\lambda$ , nRandomVariables, fromX, toX], {nRandomVariables, 500, 6*500, 500}];
GraphicsGrid[{ {First[p[[1, 1, 1]]], First[p[[2, 1, 1]]], {First[p[[3, 1, 1]]],
  First[p[[4, 1, 1]]], {First[p[[5, 1, 1]]], First[p[[6, 1, 1]]]}},
  Frame  $\rightarrow$  All, ImageSize  $\rightarrow$  600]

```

Out[168]=



Problem 1 simulation

Define function which accepts a list of random variables from exponential distribution, and λ and generates a plot of the histogram overlaid by the exponential density plot.

In[169]:=

```
m = Manipulate[ (SeedRandom[010101];
  postProcessForPartOne[getRandomNumbersFromExponential[ $\lambda$ , n], nBins,  $\lambda$ , n, 0, maxX]),
  {{nBins, 50, "Number of bins?"}, 1, 100, 1, ContinuousAction -> True,
  Appearance -> "Labeled"},
  {{ $\lambda$ , 2, " $\lambda$ "}, 1, 10, .01, ContinuousAction -> True, Appearance -> "Labeled"},
  {{n, 10000, "number of random variables?"},
  10, 100000, ContinuousAction -> True, Appearance -> "Labeled"},
  {{maxX, 2.5, "X Plot range?"}, 1, 100, 1, ContinuousAction -> True, Appearance -> "Labeled"},
  AutorunSequencing -> {{1, 15}, {2, 20}, {3, 15}}
]
```

Out[169]=

