ECE 9305A: Introduction to Probability Theory I
Name: B. G. H. Sunanda Gamage
Email: sgamage2@uwo.ca
Student ID: 250992128

Homework Assignments

[Each homework assignment begins on a new page]

Homework 1 (2018/09/19): Simulating the frequentist definition of probability

- Generate N binary digits with $P_1 = 0.5 + 0.045 * 8 = 0.86$, $P_0 = 0.14$ for N = 10, 100 ...
- Observe the relative frequencies (F) of the two outcomes, 1 and 0 (table below).

Outcome	N = 10 Relative freq.	N = 100 Relative freq.	N = 1000 Relative freq.	N = 10000 Relative freq.	N = 100000 Relative freq.
1	0.9	0.89	0.859	0.8603	0.8599
0	0.1	0.11	0.141	0.1397	0.1401

- It was observed that as this experiment was repeated with increasing N, the two relative frequencies got closer and closer to the original probabilities $P_1 = 0.86$ and $P_0 = 0.14$.
- At around N = 100000, the changes to the relative frequencies as N was doubled (to 200000) were very small (~0.0001).
- Therefore, N = 100000 was considered as the infinity for which the following quantity approaches the limit. This quantity is the probability of the outcome according to the frequentist definition of probability.

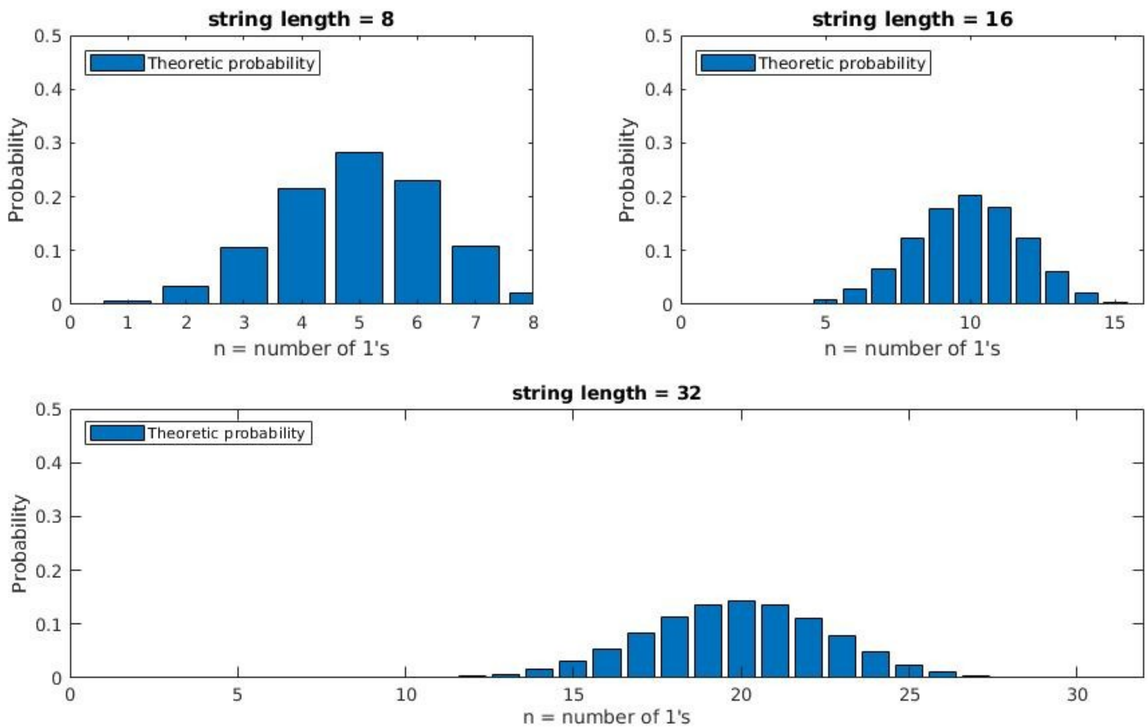
$$Probability\ of\ outcome = \lim_{N \to inf} Relative\ frequency\ of\ outcome$$

Homework 2 (2018/09/26): Probability of n number of 1's in a binary string of length N

- A binary digit string of length N is generated (consider cases N = 8, 16, 32).
  $P_0 = 0.3 + 0.01 * 8 = 0.38$, $P_1 = 0.62$.
- Part 1: the theoretic probability of having exactly n number of 1's in this string is given by

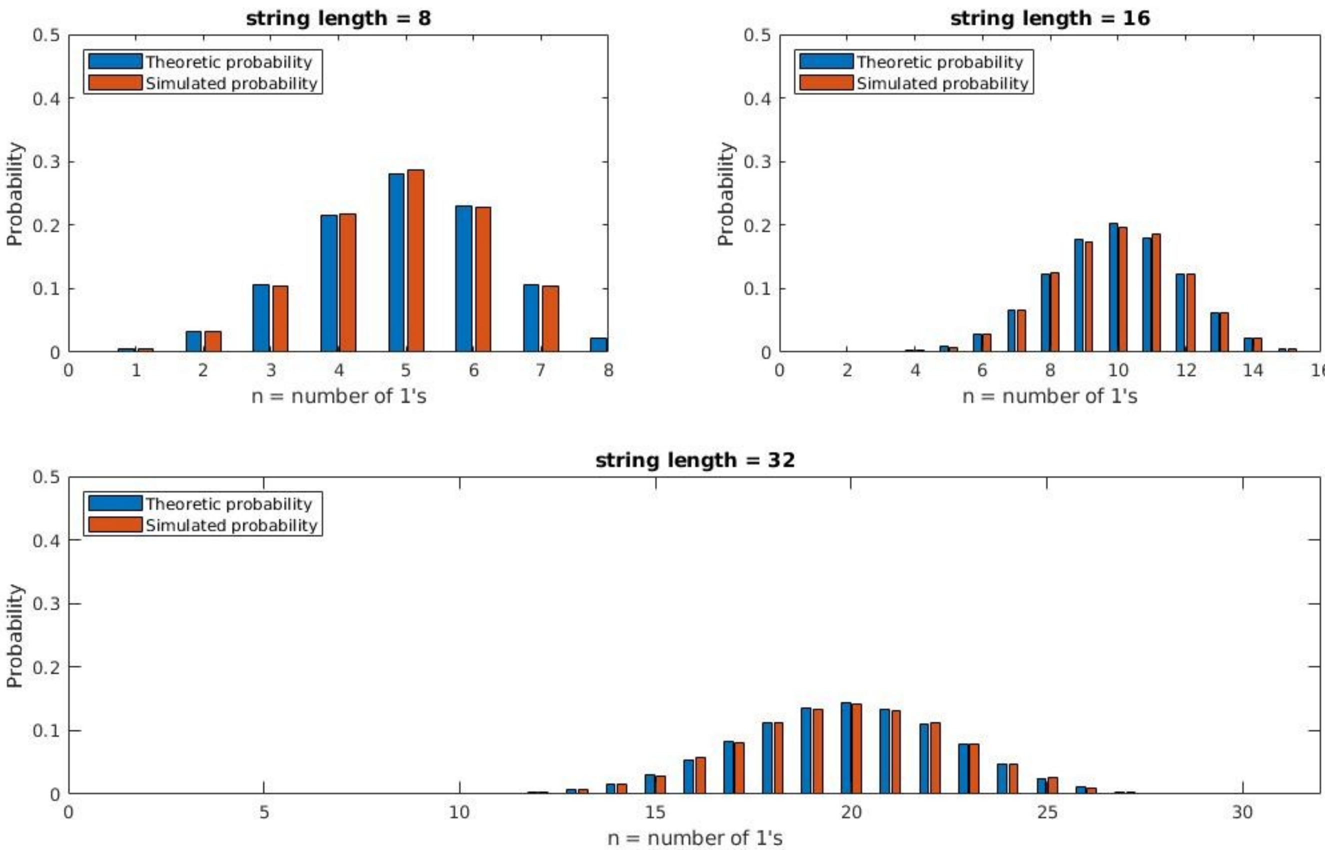
$$\frac{N!}{n!(N-n)!} . P_1^n . P_0^{N-n}$$

- This probability is calculated and plotted for below $n \in [0, N] \, for \, N = 8, 16, 32$.



Part 2: the probability of having exactly n number of 1's in the string is estimated experimentally by generating a large number of strings (10000 trials in this experiment), and counting the number of 1's in the string in each trial.

- The probabilities estimated in this way are plotted against the theoretic values in the graph below (next page).
- We can clearly see that the theoretic values and the experimental values are approximately equal. It was also seen that when the number of trials was increased to 1 million, the experimental values got even closer to the theoretic values.


string length = 8
string length = 16
string length = 32
Theoretic probability
Simulated probability
Probability
n = number of 1's
0
0.1
0.2
0.3
0.4
0.5
1
2
3
4
5
6
7
8
10
12
14
15
16
20
25
30

Homework 3 (2018/10/03): Moments of exponential and Gaussian distributions

- MATLAB Symbolic Toolbox was used in these calculations. The results were verified by cross-checking with Wolfram MathWorld reference pages.
- The probability density function of exponential and Gaussian distributions were defined. Then, multiply them by relevant polynomials, and integrate to calculate the moments.
- The following two figures show the code and the results for the two distributions. The results (moments) are highlighted in the red boxes.

```
disp(['Exponential distribution moments']);
syms x alpha sigma m;
assume(alpha>0)
assume(sigma>0)
assume(m>0) % needed by MATLAB symbolic toolbox (evaluate separately for m

pdf_exponential = piecewise(x<0, 0, x>=0, alpha * exp(-alpha * x))

% pdf_gaussian = (1/sqrt(2*pi*sigma^2)) * exp(-(x - m)^2/(2*sigma^2))

pdf = pdf_exponential;

m1_func(x) = x * pdf
m1 = int(m1_func, x, [-inf inf])

m2_func(x) = x^2 * pdf
m2 = int(m2_func, x, [-inf inf])

variance = m2 - m1^2
```

Exponential distribution moments

pdf_exponential =

$$\begin{cases} 0 & \text{if } x < 0 \\ \alpha\, e^{-\alpha x} & \text{if } 0 \le x \end{cases}$$

m1_func(x) =

$$\begin{cases} 0 & \text{if } x < 0 \\ \alpha\, x\, e^{-\alpha x} & \text{if } 0 \le x \end{cases}$$

m1 =

$$\boxed{\frac{1}{\alpha}}$$

m2_func(x) =

$$\begin{cases} 0 & \text{if } x < 0 \\ \alpha\, x^2\, e^{-\alpha x} & \text{if } 0 \le x \end{cases}$$

m2 =

$$\boxed{\frac{2}{\alpha^2}}$$

variance =

$$\boxed{\frac{1}{\alpha^2}}$$

```
disp(['Gaussian distribution moments']);
syms x alpha sigma m;
assume(alpha>0)
assume(sigma>0)
assume(m>0) % needed by MATLAB symbolic toolbox (evaluate separately for m

% pdf_exponential = piecewise(x<0, 0, x>=0, alpha * exp(-alpha * x))

pdf_gaussian = (1/sqrt(2*pi*sigma^2)) * exp(-(x - m)^2/(2*sigma^2))

pdf = pdf_gaussian;

m1_func(x) = x * pdf
m1 = int(m1_func, x, [-inf inf])

m2_func(x) = x^2 * pdf
m2 = int(m2_func, x, [-inf inf])

variance = m2 - m1^2
```

Gaussian distribution moments

pdf_gaussian =

$$\frac{\sqrt{2}\, e^{-\frac{(m-x)^2}{2\sigma^2}}}{2\,\sigma\,\sqrt{\pi}}$$

m1_func(x) =

$$\frac{\sqrt{2}\, x\, e^{-\frac{(m-x)^2}{2\sigma^2}}}{2\,\sigma\,\sqrt{\pi}}$$

$\boxed{\text{m1} = m}$

m2_func(x) =

$$\frac{\sqrt{2}\, x^2\, e^{-\frac{(m-x)^2}{2\sigma^2}}}{2\,\sigma\,\sqrt{\pi}}$$

$\boxed{\text{m2} = m^2 + \sigma^2}$

$\boxed{\text{variance} = \sigma^2}$

Homework 4 (2018/10/10): Comparing two ways of calculating moments

- In the class, we discussed two methods of calculating moments. In this homework, the two methods are coded for the exponential distribution using the MATLAB Symbolic Toolbox, and the results are compared.
- Method 1: By computing the integral in the definition of the n^{th} moment; m_n

$$m_n = \int_{-\infty}^{+\infty} x^n . p_\eta(x) . dx$$

- Method 2: By considering the Taylor expansion of the characteristic function, an equation for moments can be derived as follows.

$$m_n = (-j)^n . \theta^n(0)$$

- Results of the two methods for upto the 5^{th} moment of the exponential distribution are given below. As seen in the highlighted boxes, both methods give the same results, which is evidence for their correctness.

```
disp(['Calculate moments by integrating polynomial x pdf']);

for n = 1:num_moments
    disp(['Calculating moment ', num2str(n)]);

    moment_func(x) = x^n * pdf;
    moment = int(moment_func, x, [-inf inf])
end
```

```
Calculate moments by integrating polynomial x pdf
Calculating moment 1
moment =
```

$$\frac{1}{\alpha}$$

```
Calculating moment 2
moment =
```

$$\frac{2}{\alpha^2}$$

```
Calculating moment 3
moment =
```

$$\frac{6}{\alpha^3}$$

```
Calculating moment 4
moment =
```

$$\frac{24}{\alpha^4}$$

```
Calculating moment 5
moment =
```

$$\frac{120}{\alpha^5}$$

Figure: Moments calculated by method 1 (definition of moments)

```
disp(['Calculate moments by taking coefficients of the characteristic funti

syms w
% char_func = fourier(pdf, x, w)
char_func = int(pdf * exp(1i*w*x), x, [-inf, inf]); % Fourier transform, de

for n = 1:num_moments
    disp(['Calculating moment ', num2str(n)]);

    diff_func = diff(char_func, w, n);

    moment = (-1i)^n * subs(diff_func, w, 0)
end
```

```
Calculate moments by taking coefficients of the characteristic funtion
Calculating moment 1
moment =
```

$$\frac{1}{\alpha}$$

```
Calculating moment 2
moment =
```

$$\frac{2}{\alpha^2}$$

```
Calculating moment 3
moment =
```

$$\frac{6}{\alpha^3}$$

```
Calculating moment 4
moment =
```

$$\frac{24}{\alpha^4}$$

```
Calculating moment 5
moment =
```

$$\frac{120}{\alpha^5}$$

Figure: Moments calculated by method 2 (by log characteristic function)

Homework 5 (2018/10/17): Calculating entropy of the exponential distribution

- Entropy of a probability distribution is given by;

$$H=-\int_{-\infty}^{+\infty} \ln p_{\eta}(x).\, p_{\eta}(x).\, dx$$

- The entropy calculation for the exponential distribution was done using MATLAB Symbolic Toolbox and the results are given below.

```
disp(['Exponential distribution entropy']);

syms x alpha;
assume(alpha>0)

% pdf_exponential = piecewise(x<0, 0, x>=0, alpha * exp(-alpha

assume(x>=0)
pdf_exponential = alpha * exp(-alpha * x)

pdf = pdf_exponential;

entropy_func = -1 * log(pdf) * pdf

entropy = int(entropy_func, x, [0 inf])
```

Exponential distribution entropy

pdf_exponential = $\alpha\, e^{-\alpha\, x}$

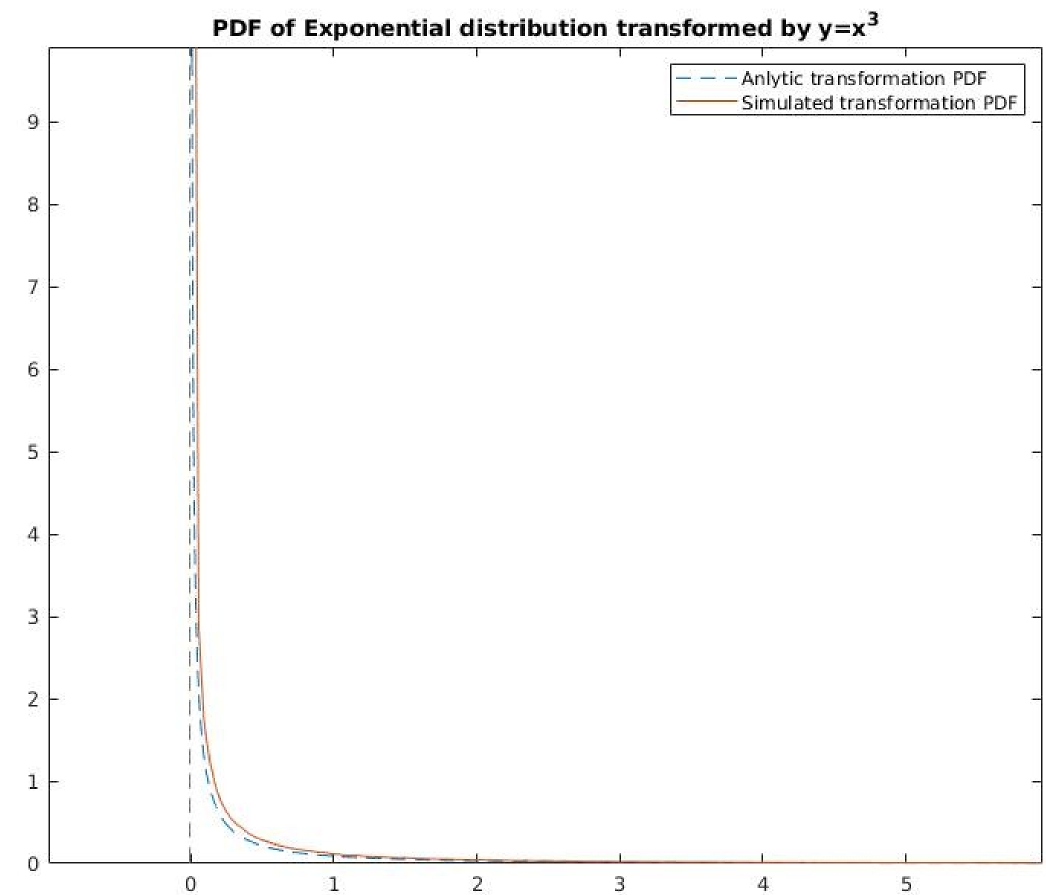
entropy_func = $-\alpha\, e^{-\alpha\, x} \log(\alpha\, e^{-\alpha\, x})$

entropy = $\boxed{1-\log(\alpha)}$

Figure: Entropy of the exponential distribution

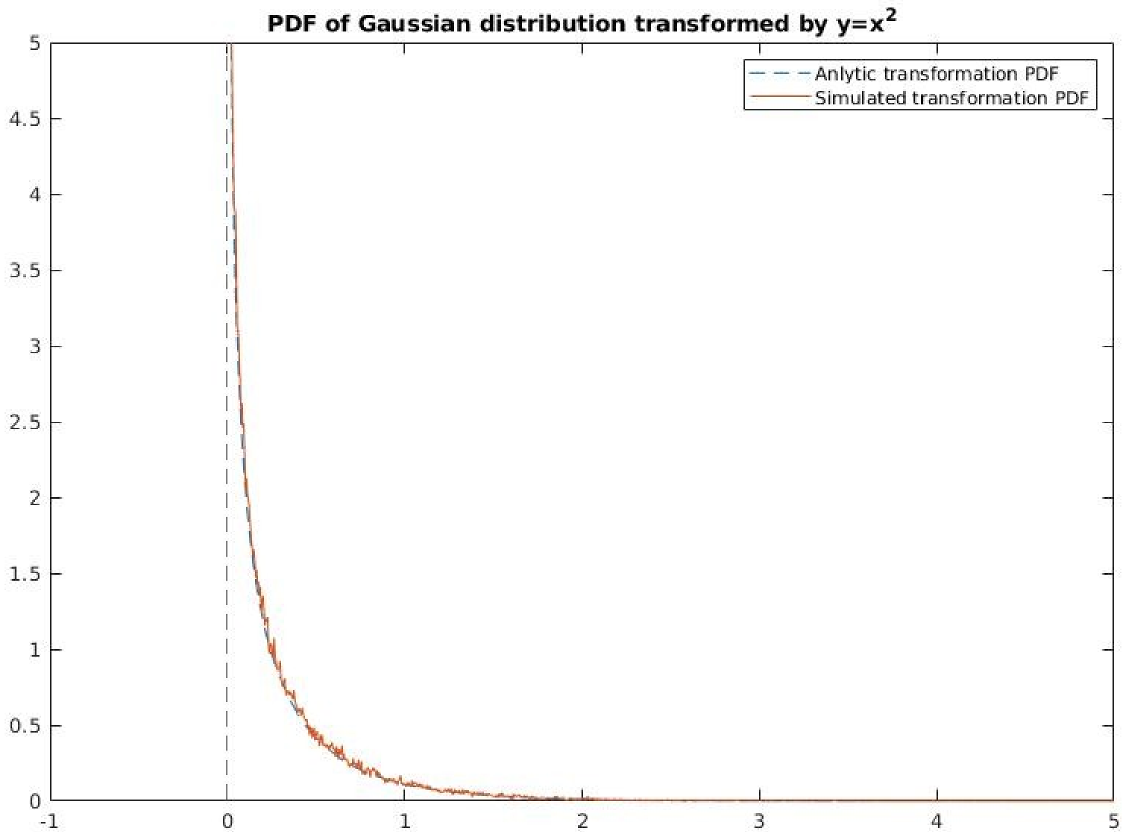
Homework 6 (2018/10/24): Cubic transformation of exponential distribution: theoretical vs. simulation results

- Applying the cubic transformation $y = x^3$ on the exponential distribution $p_\xi(x)$ gives the transformed distribution $p_\eta(y) = p_\xi(y^{(1/3)}) . \frac{1}{3} . y^{(-2/3)}$. The graph of the $p_\eta(y)$ can be obtained by plotting this equation.
- To check whether this equation is correct, it was simulated by generating some data samples from an exponential distribution, applying the transformation $y = x^3$ on them, and estimating the density function of the transformed data points (normalized histogram).
- The results of the graph of the equation vs. the graph of the estimated PDF is shown in the figure below. The two graphs are very similar, and therefore we can conclude that the equation is correct.



Homework 7 (2018/10/24): Quadratic transformation of Gaussian distribution: theoretical vs. simulation results

- Applying the quadratic transformation $y=x^2$ on the Gaussian distribution $p_\xi(x)$ gives the transformed distribution $p_\eta(y)=\frac{1}{\sqrt{(y)}.\sqrt{(2\pi\sigma^2)}}.\exp(\frac{-y}{2\sigma^2})$. The graph of the $p_\eta(y)$ can be obtained by plotting this equation.
- To check whether this equation is correct, it was simulated by generating some data samples from a Gaussian distribution, applying the transformation $y=x^2$ on them, and estimating the density function of the transformed data points (normalized histogram).
- The graph of the equation vs. the graph of the estimated PDF is shown in the figure below. The two graphs are very similar and overlapping (except for some noise in the simulated graph) and therefore we can conclude that the equation is correct.



Homework 8 (2018/10/24): Piecewise transformation of Gaussian distribution: theoretical vs. simulation results

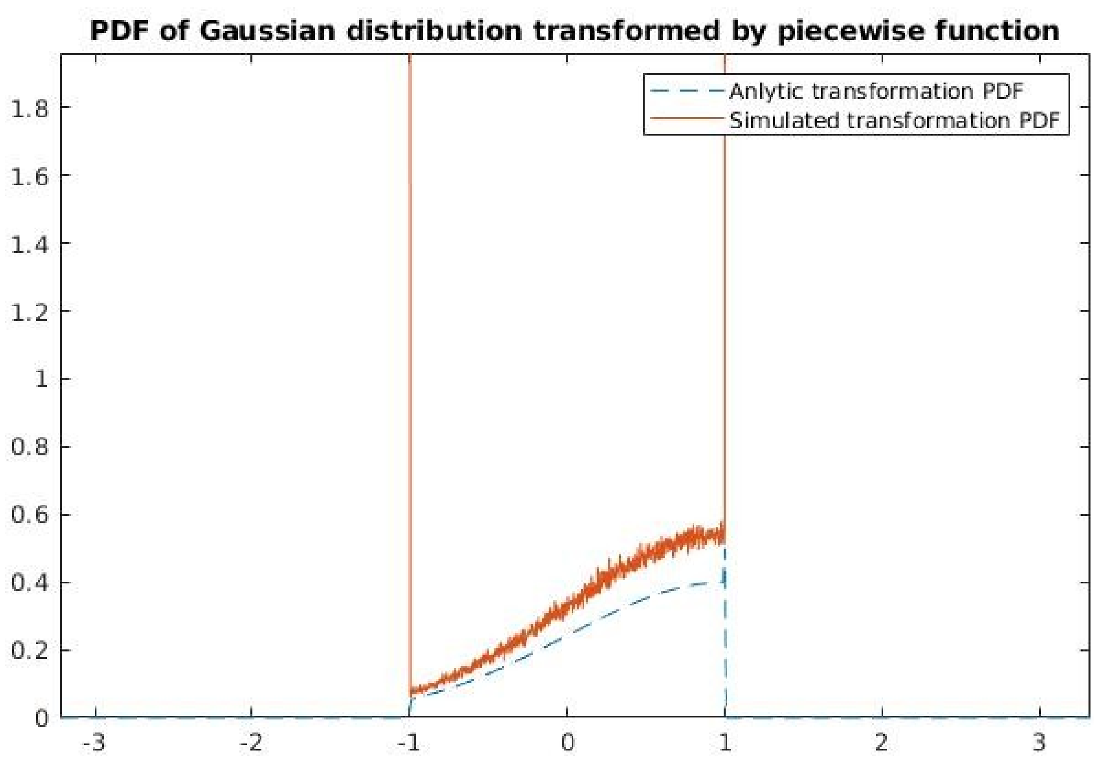
- Applying the piecewise transformation

$$y=\begin{cases} 1 & if\ x>\sigma \\ x & if\ -\sigma\le x\le\sigma \\ -1 & if\ x<-\sigma \end{cases}$$

  on the Gaussian distribution $p_{\xi}(x)$ with mean = 1, sigma = 1 gives the transformed distribution

$$p_{\eta}(y)=\begin{cases} 0.5\,\delta(y-1) & if\ x>\sigma(y=1) \\ p_{\xi}(x) & if\ -\sigma\le x\le\sigma(-1<y<1) \\ 0.0228\,\delta(y+1) & if\ x<-\sigma(y=-1) \end{cases}$$

- The graph of the $p_{\eta}(y)$ can be obtained by plotting this equation.
- To check whether this equation is correct, it was simulated by generating some data samples from a Gaussian distribution, applying the given piecewise transformation on them, and estimating the density function of the transformed data points (normalized histogram).
- The graph of the equation vs. the graph of the estimated PDF is shown in the figure below. The two graphs are similar (except for some noise in the simulated graph, and a sharper upward trend deviating from the theoretical graph). Therefore we conclude that the equation is correct.



Homework 9 (2018/10/31): Proving that the integral of the bi-variate joint Gaussian density over the full domain of both variables is 1

Homework 10 (2018/10/31): Proving that integrating the bi-variate joint Gaussian density over one variable gives the Gaussian marginal density of the other variable

- The above two integrals were evaluated using the MATLAB Symbolic Toolbox. The results are highlighted in red boxes in the figure below.

```
disp(['Joint Gaussian distribution integration']);
syms x y sigma_1 sigma_2 r;
assume(sigma_1>0)
assume(sigma_2>0)
assume(r>=-1 & r<=1)

pdf_joint_gaussian = ( 1/((2*pi*sigma_1*sigma_2)*sqrt(1-r^2)) * ...
                      exp((-1/(2*(1-r^2)))*(x^2/sigma_1^2 - 2*r*x*y/(sigma_1

pdf = pdf_joint_gaussian;

integral = int(int(pdf, x, -inf, inf, 'IgnoreAnalyticConstraints', true) ...
    , y, -inf, inf, 'IgnoreAnalyticConstraints', true)

marginal_pdf = int(pdf, y, -inf, inf, 'IgnoreAnalyticConstraints', true)
```

```
Joint Gaussian distribution integration

pdf_joint_gaussian =
```

$$\frac{e^{\frac{\frac{x^2}{\sigma_1^2}+\frac{y^2}{\sigma_2^2}-\frac{2\,r\,x\,y}{\sigma_1\,\sigma_2}}{2\,r^2-2}}}{2\,\sigma_1\,\sigma_2\,\pi\,\sqrt{1-r^2}}$$

```
integral =
```

$$\begin{cases} 0 & \text{if } r \in \{-1, 1\} \\ \boxed{1 \quad \text{if } r \in (-1, 1)} \end{cases}$$

```
marginal_pdf =
```

$$\begin{cases} 0 & \text{if } r \in \{-1, 1\} \\ \boxed{\frac{\sqrt{2}\, e^{-\frac{x^2}{2\,\sigma_1^2}}}{2\,\sigma_1\,\sqrt{\pi}} \quad \text{if } r \in (-1, 1)} \end{cases}$$

Homework 11 (2018/11/21): Simulation: observing that the mean of a number of i.i.d random variables approaches Gaussian

- Consider N independent and identically distributed (i.i.d) random variables $\xi_{1,} \xi_{2,} \ldots \xi_N$. Now consider the transformed random variable (averaging the RVs);

  $$\eta = \frac{1}{N} \sum_{i=1}^{N} \xi_i$$

- The simulated density functions of η for the cases N = 1, 2, … 5 is given below for the two cases where $\xi_i \sim Uniform[0:1]$ and $\xi_i \sim Exponential(mean=1)$.
- In both cases, as N increases, the PDF of the transformed RV (averages) approximates the Gaussian with mean = mean of ξ_1

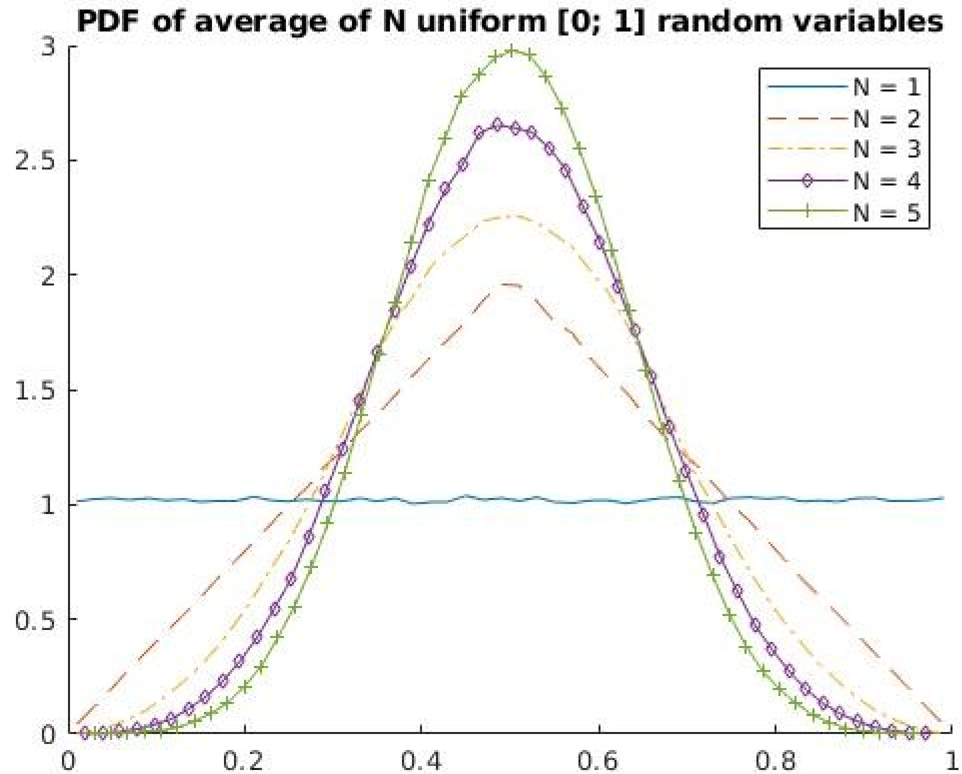


Figure: Average of N uniform RVs. This distribution quickly converge to Gaussian.

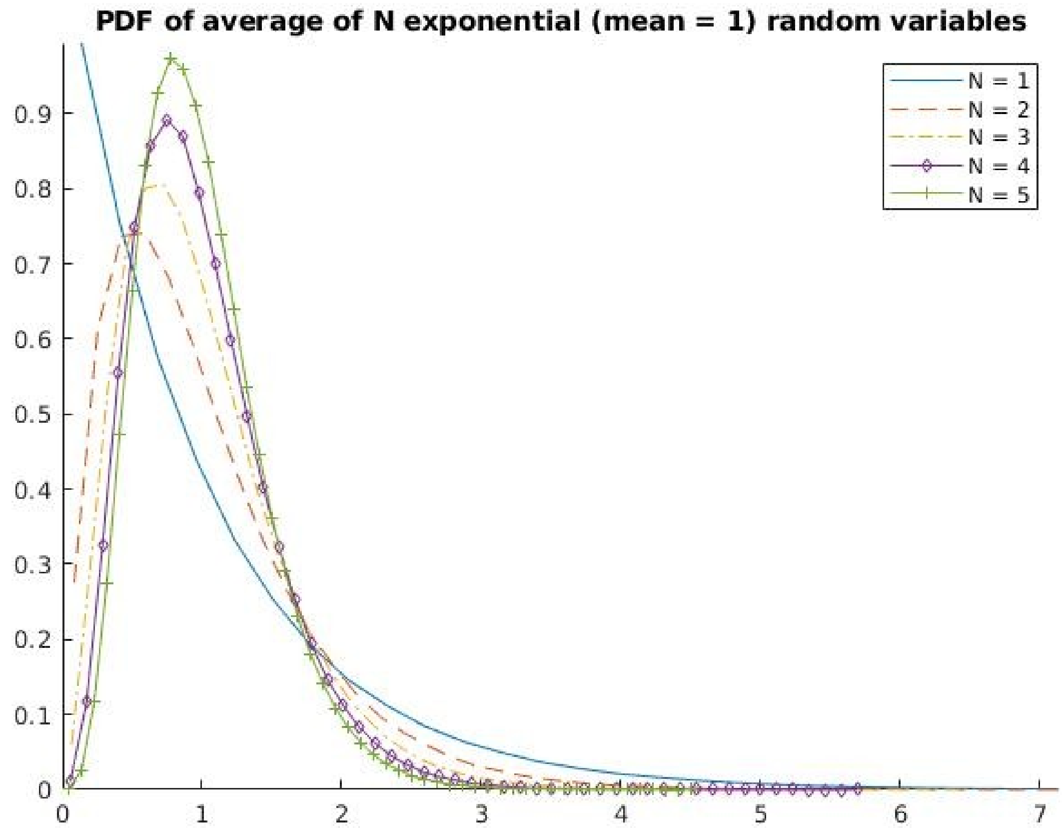


Figure: Average of N exponential RVs. This distribution is slower to converge to Gaussian (compared to the uniform distribution in the previous case.

Homework 12 (2018/11/21): Calculate variance of given density function

- In the following density function (PDF), first we find the constant c.

$$p(x)=\frac{c}{x^2+1}$$

- To find c, we set the integral of p(x) over all x to 1, because it is a density function.

$$\int_{-\infty}^{\infty}\frac{c}{x^2+1}.dx=c\,\pi=1$$

$$c=\frac{1}{\pi}$$

- Now we find the variance by the definition of central moments (using MATLAB Symbolic Toolbox).
- As seen in the figure below, both the mean and variance are infinite in this PDF.

```
disp(['Variance of given function']);

syms x c;
% c = 1/pi;

pdf = c/(x^2+1)

% assume(x>=0)
m_1_func = x * pdf

m_1 = int(m_1_func, x, [-inf inf])

mu_2_func = (x - m_1) * pdf

variance = int(mu_2_func, x, [-inf inf])
```

```
Variance of given function

pdf =
```

$$\frac{c}{x^2+1}$$

```
mu_1_func =
```

$$\frac{c\,x}{x^2+1}$$

```
mu_1 = NaN

mu_2_func = NaN

variance = NaN
```

Homework 13 (2018/11/21): Max transformation of two uniform random variables

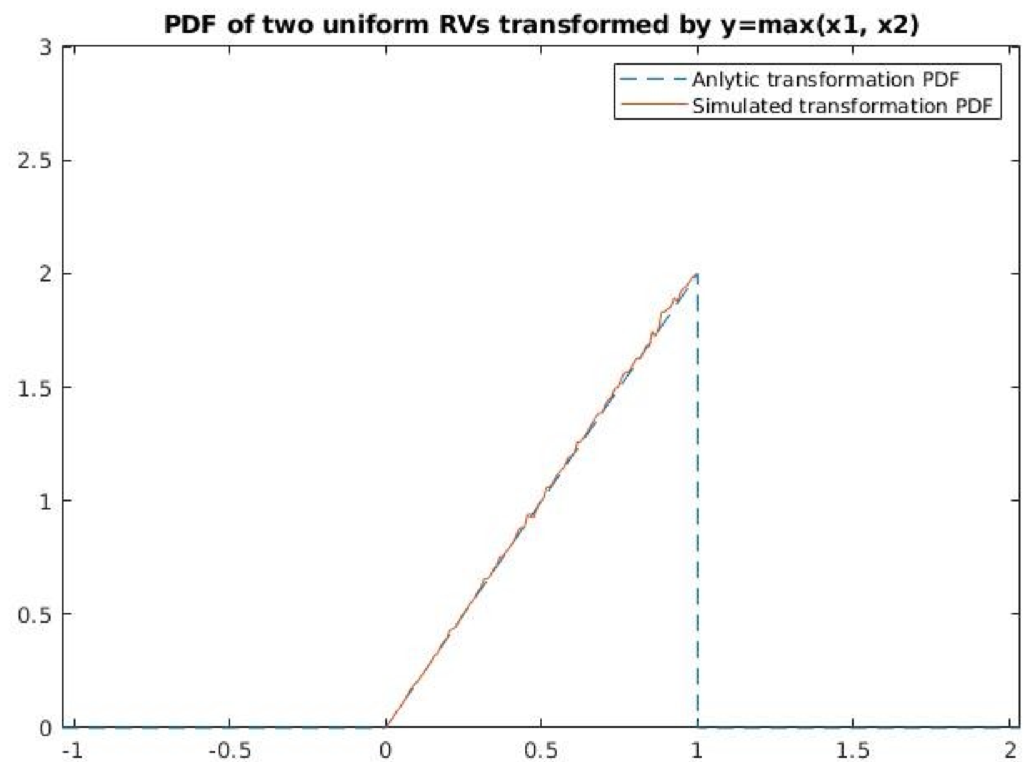
- Consider the two independent random variables $\xi_{1,} \xi_2$ uniform in [0, 1], and the max transformation applied as below.

$$\eta = max(\xi_{1,} \xi_2)$$

- The PDF of the transformed variable can be obtained analytically as;

$$p_\eta(y) = \begin{cases} 2y & if\ y \in [0,1] \\ 0 & else \end{cases}$$

- The graph of the $p_\eta(y)$ can be obtained by plotting this equation.
- To check whether this equation is correct, it was simulated by generating some data samples from two uniform distributions, taking the max of them, and estimating the density function of the transformed data points (normalized histogram).
- The two graphs (analytic and simulated) are shown below. As clearly seen in the figure, the two graphs overlap, and we can conclude that the analytic equation was correct.



Homework 14 (2018/12/05): Max transformation of two uniform random variables

- Consider the random signal $\xi(t)$ going through a linear time-invariant transformation $h(t)=\frac{d\,\xi(t)}{dt}$ and producing the random signal $\eta(t)$.
- Given the covariance function of the input signal, $C_\xi(\tau)=\sigma^2 J_0(2\pi f_0 \tau)$ where J_0 is the Bessel function of the first kind and first order, the task is to find the covariance function and the variance of the output signal, $C_\eta(\tau)$.
- For this we use the formula (derived using the Wiener–Khinchin theorem);

  $$C_\eta(\tau)=\frac{d^2 C_\xi(\tau)}{d\tau^2}$$

- The variance is the value of the covariance function at $\tau=0$.
- The evaluation of this formula was carried out by MALAB Symbolic Toolbox, and the results are shown below (see the highlighted box in red).

```
syms tau sigma f_0;
assume(sigma>0)

input_cov = sigma^2 * besselj(0, 2*pi*f_0*tau)

output_cov = - diff(input_cov, 2)

output_variance = limit(output_cov, tau, 0)
```

input_cov = $\sigma^2 J_0(2\pi f_0 \tau)$

output_cov =

$$-2\pi f_0 \sigma^2 \left(\frac{J_1(2\pi f_0 \tau)}{\tau} - 2\pi f_0 J_0(2\pi f_0 \tau)\right)$$

output_variance = $2 f_0^2 \sigma^2 \pi^2$