Problem 1

The problem is to use importance sampling to estimate:

$$\theta = \int_0^\infty x \frac{e^{-(y-x)^2/2} e^{-3x}}{Z} dx$$
$$Z = \int_0^\infty e^{-(y-x)^2/2} e^{-3x} dx; \quad y = 0.5$$

Observe:

$$\begin{aligned}
\theta &= \int_0^\infty x \frac{e^{-(y-x)^2/2} e^{-3x}}{Z} dx \\
&= \int_0^\infty x \frac{e^{-(y-x)^2/2} e^{-3x}}{\int_0^\infty e^{-(y-x)^2/2} e^{-3x} dx} dx \\
&= \int_0^\infty x \frac{e^{-(y-x)^2/2} \; 3 * e^{-3x}}{\int_0^\infty e^{-(y-x)^2/2} \; 3 * e^{-3x} dx} dx
\end{aligned}$$

Recall that the pdf, $f(x)$, of an exponentially distributed random variable, x, with parameter $\lambda = \frac{1}{\beta}$ is:

$$\beta * e^{-\beta x}$$

Thus $3 * e^{-3x}$ is the pdf of an exponentially distributed r.v. with $\lambda = \frac{1}{3}$. Our problem thus becomes:

$$\begin{aligned}
\theta &= \int_0^\infty x \frac{e^{-(y-x)^2/2} \; 3 * e^{-3x}}{\int_0^\infty e^{-(y-x)^2/2} \; 3 * e^{-3x} dx} dx \\
&= \int_0^\infty \frac{x * e^{-(y-x)^2/2} \; 3 * e^{-3x}}{\mathbb{E}[e^{-(y-x)^2/2}]} dx \\
\theta &= \frac{\mathbb{E}[x * e^{-(y-x)^2/2}]}{\mathbb{E}[e^{-(y-x)^2/2}]}
\end{aligned}$$

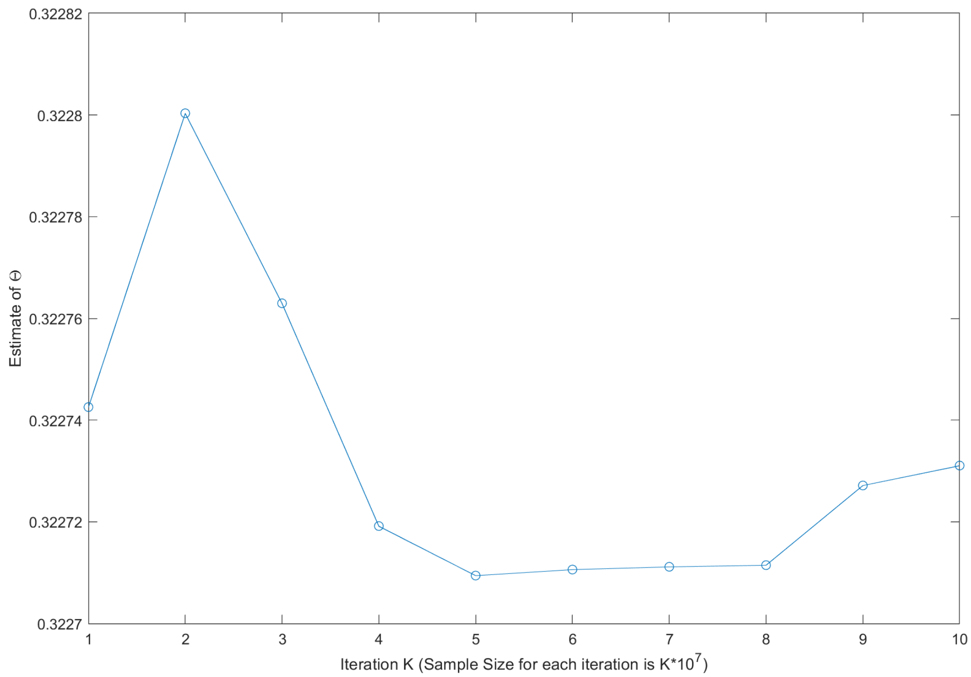
By the Weak Law of Large Numbers:

$$\lim_{n\to\infty} \Pr\left(\left|\sum_{i=1}^n x_i * e^{-(y-x_i)^2/2} - \mathbb{E}[x * e^{-(y-x)^2/2}]\right| > \varepsilon\right) = 0$$

Thus, we can employ the estimator:

$$\hat{\theta} = \frac{\sum_{i=1}^n [x_i * e^{-(0.5-x_i)^2/2}]}{\sum_{i=1}^n [e^{-(0.5-x_i)^2/2}]}$$

Where $x_i \overset{i.i.d}{\sim} \exp(\frac{1}{3})$.



Problem 2

The problem is to use tilted sampling to estimate the quantity $\theta = \Pr(X > a)$ for $X \sim \mathcal{N}(0, 1)$.
Recall that the tilted density is given by:

$$f_t(x) = \frac{\exp(tx)f(x)}{M(t)}$$
$$M(t) = \int_{-\infty}^{\infty} \exp(tx)f(x)dx$$

Focusing on $M(t)$, we note that this is simply the moment generating function of a standard normal distribution and is thus:

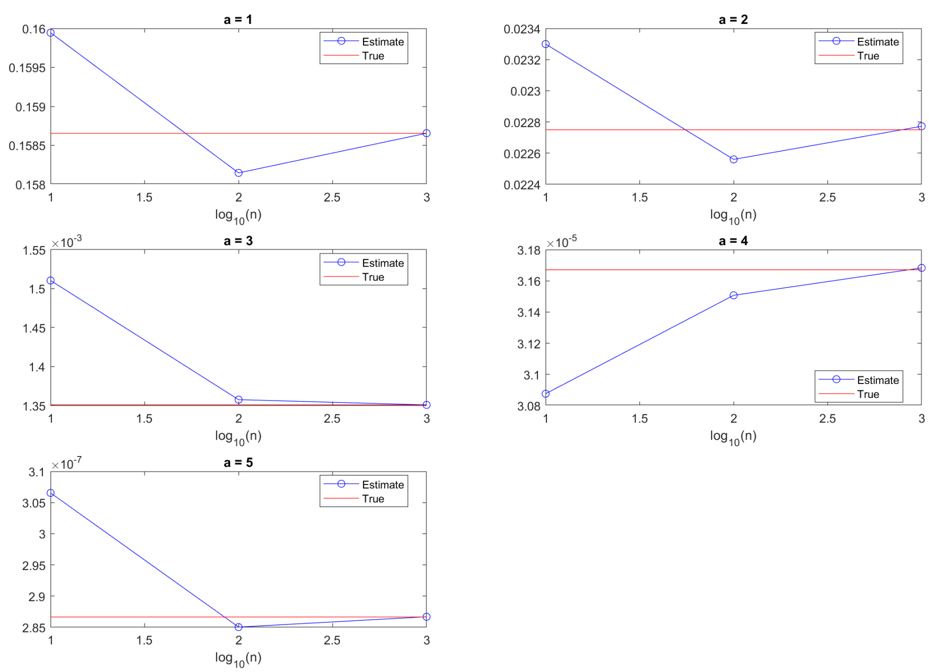
$$M(t) = \exp\left(\frac{t^2}{2}\right)$$

Thus, the tilted density becomes:

$$\begin{aligned} f_t(x) &= \frac{\exp(tx)f(x)}{M(t)} \\ &= \frac{\exp(tx)\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{(x)^2}{2}\right)}{\exp\left(\frac{t^2}{2}\right)} \\ f_t(x) &= \frac{\exp\left(\frac{-(x-t)^2}{2}\right)}{\sqrt{2\pi}} \end{aligned}$$

The tilted density is thus normal with mean t and variance 1. Choosing t such that a is the mean of this tilted density implies that $t = a$. Now,

$$\begin{aligned} f_t(x) &= \frac{\exp(tx)f(x)}{M(t)} \\ \Leftrightarrow \frac{f(x)}{f_t(x)} &= \exp(-ax)M(a) \\ &= \exp\left(\frac{a^2}{2} - ax\right) \\ \overset{WLLN}{\Longrightarrow} \hat{\theta} &= \frac{1}{n}\sum_{i=1}^{n} I_{x_i \geq a} \exp\left(\frac{a^2}{2} - ax_i\right) \end{aligned}$$



Problem 3

The problem is to use tilted sampling to estimate the quantity $\theta = \Pr(X > a)$ for $X \sim exp(\frac{1}{\lambda})$.
Recall that the tilted density is given by:

$$f_t(x) = \frac{\exp(tx)f(x)}{M(t)}$$
$$M(t) = \int_{-\infty}^{\infty} \exp(tx)f(x)dx$$

Focusing on $M(t)$:

$$\begin{aligned} M(t) &= \int_0^{\infty} \exp(tx)\lambda \exp(-\lambda x)dx \\ &= \lambda \int_0^{\infty} \exp((t-\lambda)x)dx \\ &= \lambda \frac{\exp((t-\lambda)x)}{t-\lambda}\Big|_0^{\infty} \\ &= \frac{\lambda}{\lambda - t} \end{aligned}$$

Before proceeding further, it is important to note that for all of this to work, it must be the case that $t < \lambda$.

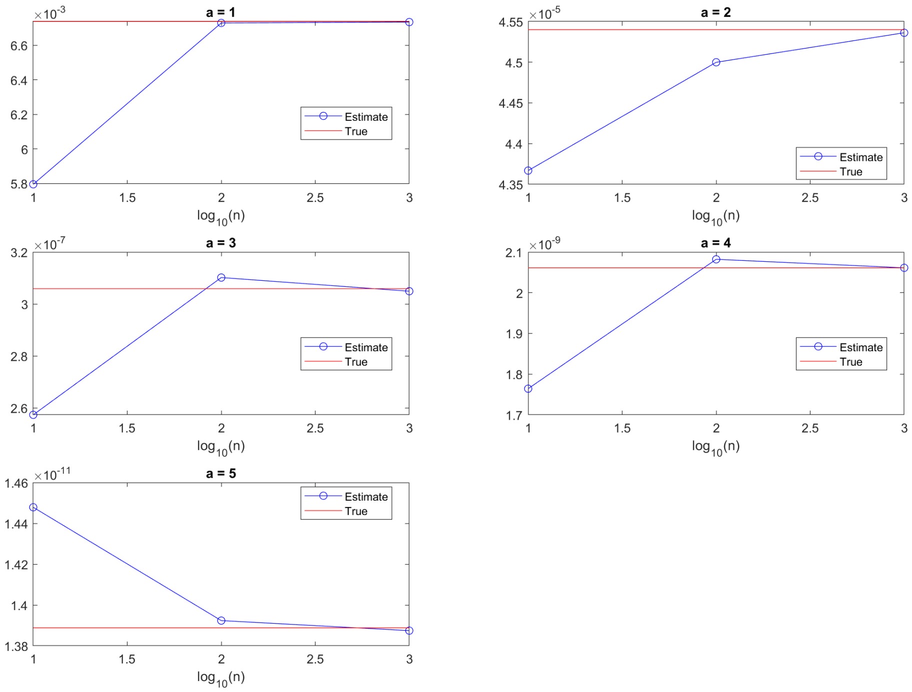
$$\begin{aligned} f_t(x) &= \frac{\exp(tx)f(x)}{M(t)} \\ &= \frac{\exp(tx)\lambda\exp(-\lambda x))}{\frac{\lambda}{\lambda-t}} \\ &= (\lambda - t)\exp(x(t-\lambda)) \\ &= (\lambda - t)\exp(-(\lambda - t)x) \end{aligned}$$

The tilted density is thus exponentially distributed with corresponding parameter $\beta = \frac{1}{\lambda - t}$ (i.e. $exp(\frac{1}{\beta})$). Recall that the mean of an exponential distribution with parameter $\frac{1}{\beta}$ is β. Choosing the optimal t to estimate θ for a given a requires a to be the mean of the tilted density. Thus:

$$\frac{1}{\lambda - t} = a \Leftrightarrow t = \lambda - \frac{1}{a}$$

So our distribution is $exp\ (a)$.

$$
\begin{aligned}
f_t(x) &= \frac{\exp(tx)f(x)}{M(t)} \\
\Leftrightarrow \frac{f(x)}{f_t(x)} &= \exp(-t(a)x)M(t(a)) \\
&= \exp\left(-\left(\lambda - \frac{1}{a}\right)x\right) * \frac{\lambda}{\lambda - \lambda + \frac{1}{a}} \\
&= \exp\left(-\left(\lambda - \frac{1}{a}\right)x\right) * \lambda a \\
\overset{WLLN}{\Longrightarrow} \hat{\theta} &= \frac{1}{n}\sum_{i=1}^{n} I_{x_i \geq a} \exp\left(-\left(\lambda - \frac{1}{a}\right)x_i\right) * \lambda a
\end{aligned}
$$



MATLAB Code for Problems 1, 2, 3:

```
clc
clear
%Diary
dfile ='MATLAB_Output_OM.txt';
if exist(dfile, 'file') ; delete(dfile); end
diary(dfile)
```

```
diary on

%Introduction
fprintf('----------------------------------------------------------------------\n');
fprintf('\t Oscar Martinez \t HW 10 \t STA 5106\n');
fprintf('----------------------------------------------------------------------\n');

%------Problem 1:------
fprintf('------------Problem 1------------\n');

rng(17);
n=10^7;
K=10;

for k = 1:K
    x(:,k) = exprnd(1/3,1,n); %Sampling distrbn is exp(1/3)
    z = reshape(x(:,1:k),[],1);
    num(k) = mean(z.*exp(-(0.5-z).^2/2)/3);
    den(k) = mean(exp(-(0.5-z).^2/2)/3);
    ratio(k) = num(k)/den(k);
end

figure(1)
plot(1:K,ratio,'-o')
xlabel('Iteration K (Sample Size for each iteration is K*10^7)'); %Sample Size for each iteration is K*10^7
ylabel('Estimate of \Theta');

%------Problem 2:------
fprintf('------------Problem 2------------\n');
clear

SM = [3 5 7]; %Sample Size Matrix
A = [1:5]; %Estimate MKatrix
for j = 1:3
    n = 10^eval('SM(j)');
    for a = 1:5
        x = normrnd(a,1,1,n); %Sample from Norm(a,1)
        mu(j,a) = mean((x>a).*exp(a^2/2-a*x));
        TP(a)=1-normcdf(a); %True Prob
        %Plot
```

```
        figure(2)
        subplot(3,2,a);
        plot(1:j, mu(:,a), '-bo',1:j, ones(1,j)*TP(a), 'r');
        title(['a = ', num2str(a)]);
        xlabel('log_{10}(n)');
        legend('Estimate', 'True', 'Location','best');
    end
end

%------Problem 3:------
fprintf('------------Problem 3------------\n');
clear

SM = [3 5 7]; %Sample Size Matrix
A = [1:5]; %Estimate MKatrix
lm = 5;
for j = 1:3
    n = 10^eval('SM(j)');
    for a = 1:5
        x = exprnd(a,1,n); %Sample from Exp(a)
        mu(j,a) = mean(lm*a*(x>a).*exp(-(lm-(1/a))*x));
        TP(a)=1-expcdf(a,1/lm); %True Prob
        %Plot
        figure(3)
        subplot(3,2,a);
        plot(1:j, mu(:,a), '-bo',1:j, ones(1,j)*TP(a), 'r');
        title(['a = ', num2str(a)]);
        xlabel('log_{10}(n)');
        legend('Estimate', 'True', 'Location','best');
    end
end

diary off
```

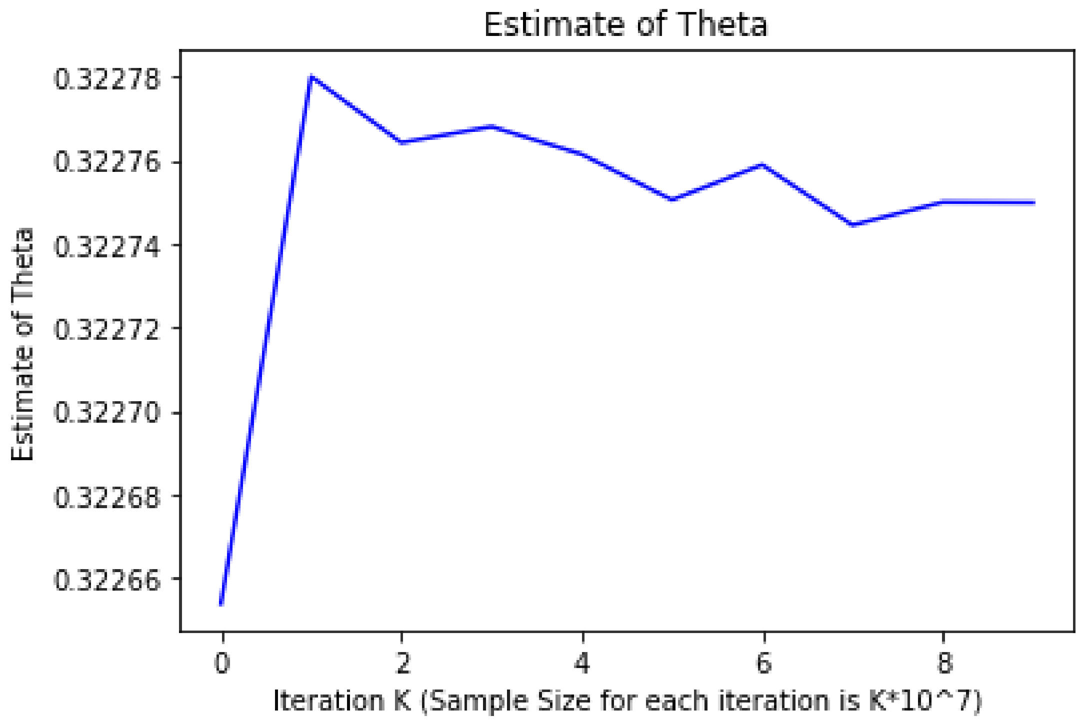
Problem 4

[1]:

```
import numpy as np
from matplotlib import pyplot
import random

#------------------------------Problem 4----------------------------------
print('Problem 4')
```

```
        random.seed(1)
        n = 10**7
        K = 10
        lm = 1/3
        x = np.zeros((n,K))
        theta = np.zeros((1,10))
        num = np.zeros((1,10))
        den = np.zeros((1,10))


        for j in range(K):
        x[:,j] = np.array( [random.expovariate(1/lm) for x in range(n)]␣
↪)
        if j == 0:
        num[0,j] =  np.mean( x[:,0]*np.exp( -0.5*(0.5-x[:,0])**2 )/3 )
        den[0,j] = np.mean( np.exp( -0.5*(0.5-x[:,0])**2 )/3 )
        theta[0,j] = num[0,j]/den[0,j]
        else:
        num[0,0:j+1] =  np.mean( x[:,0:j+1]*np.exp( -0.5*(0.5-x[:,0:
↪j+1])**2 ) )
        den[0,j] = np.mean( np.exp( -0.5*(0.5-x[:,0:j+1])**2 ) )
        theta[0,j] = num[0,j]/den[0,j]

        t = range(0,K)
        #Plot
        pyplot.plot(t, theta[0,:], 'b');
        pyplot.title('Estimate of Theta');
        pyplot.subplots_adjust(hspace=1,wspace=0.5);
        pyplot.figure(num=1, figsize=(10, 10), dpi=140, facecolor='w',␣
↪edgecolor='k');
        pyplot.xlabel('Iteration K (Sample Size for each iteration is␣
↪K*10^7)');
        pyplot.ylabel('Estimate of Theta');
        pyplot.show();
```

Problem 4

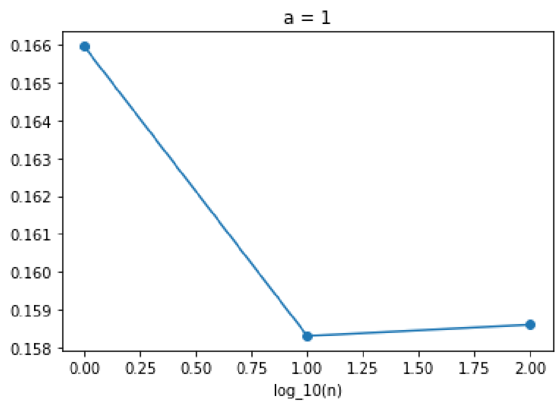


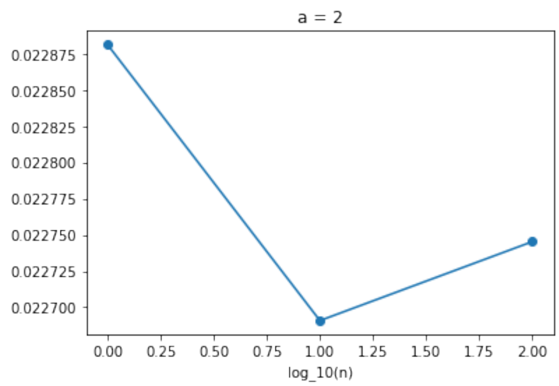
Problem 5

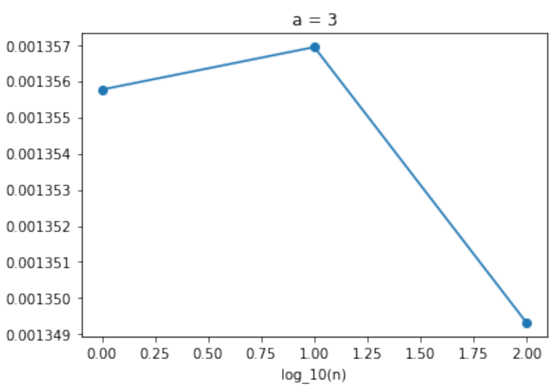
[2]:
```
import numpy as np
from matplotlib import pyplot
import random

#----------------------------Problem␣
↪5----------------------------------
print('Problem 5')
random.seed(1)
K = 5
i = 0
mu = np.zeros((3,5))

for y in range(3,8,2):
#print(y)
n = 10**y
if i == 3:
i = 0
for a in range(K):
```

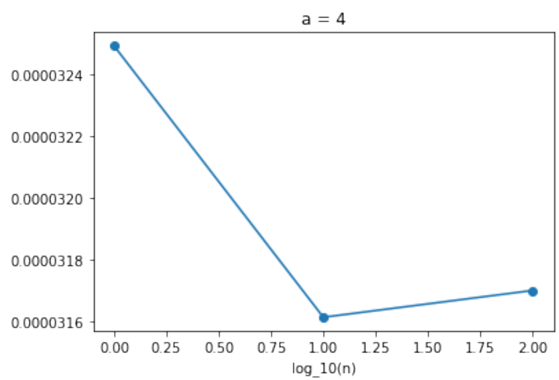
```
    #print(a)
    s = a+1
    x = np.array( [random.normalvariate(a+1,1) for x in range(n)] )
    mu[i,a] = np.mean( (x > s)*np.exp(0.5*s**2 - s*x ) )
    i = i+1

    for a in range(K):
    #Plot
    t = range(0,3)
    s = a+1
    pyplot.plot(t, mu[:,a], 'o-');
    pyplot.title('a = %i' %s);
    pyplot.xlabel('log_10(n)');
    pyplot.show();
```

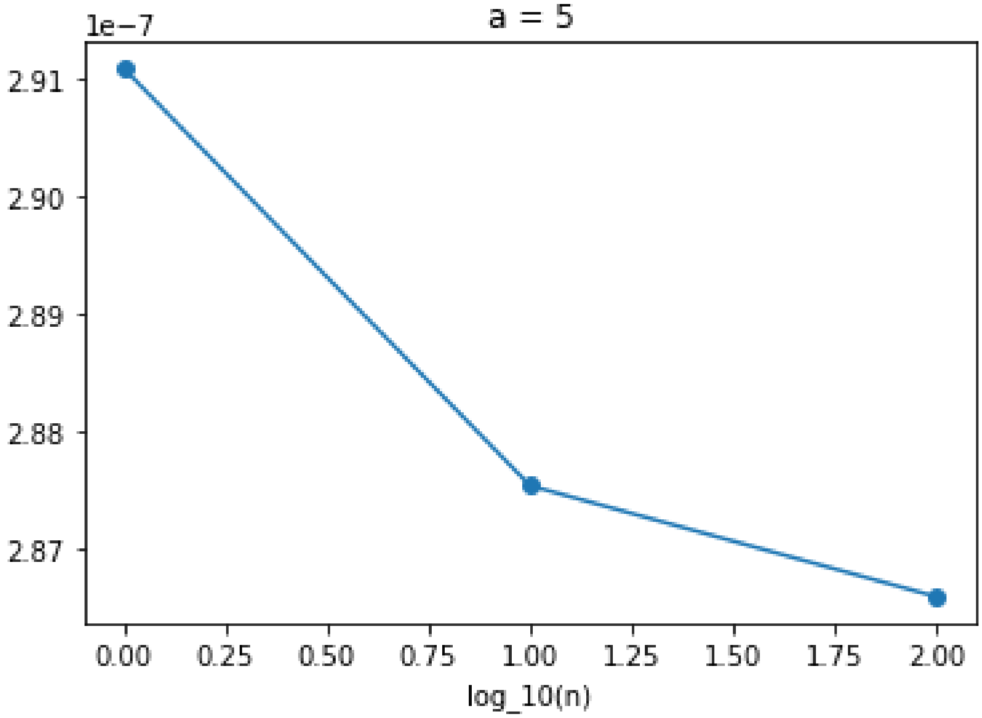
Problem 5










a = 5
1e−7
2.91
2.90
2.89
2.88
2.87
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
log_10(n)