Stoch proj 3

1. $f(x,\lambda) = \lambda e^{-\lambda x}$ <u>Exponential</u>

$$L(\lambda, \vec{x}) = \prod_{i=1}^{n} f(x_i, \lambda) \quad \text{where } \vec{x} = (x_1, \dots, x_n)$$

$$= \prod_{i=1}^{n} \lambda e^{-\lambda x_i}$$

$$= \lambda^n \exp(-\lambda \sum_{i=1}^{n} x_i)$$

$$\ln(L(\lambda, \vec{x})) = \ln(\lambda^n \exp(-\lambda \sum_{i=1}^{n} x_i))$$

$$= \ln(\lambda^n) - \lambda \sum_{i=1}^{n} x_i$$

$$\frac{d \ln(L(\lambda, \vec{x}))}{d\lambda} = \frac{n}{\lambda} - \sum_{i=1}^{n} x_i = 0$$

$$\Rightarrow \lambda = \frac{n}{\sum_{i=1}^{n} x_i}$$

Rayleigh $\quad f(x,\lambda) = \frac{x}{\lambda^2} \exp\left(-\frac{1}{2} \cdot \left(\frac{x}{\lambda}\right)^2\right)$

$$L(\lambda, \vec{x}) = \prod_{i=1}^{n} \frac{x_i}{\lambda^2} \exp\left(-\frac{1}{2} \cdot \left(\frac{x_i^2}{\lambda^2}\right)\right) \quad \text{where } \vec{x} = (x_1, \cdots, x_n)$$

$$= \frac{1}{\lambda^{2n}} \cdot \prod_{i=1}^{n} x_i \cdot \exp\left(-\frac{1}{2\lambda^2} \sum_{i=1}^{n} x_i^2\right)$$

$$\ln(L(\lambda, \vec{x})) = \ln\left(\prod_{i=1}^{n} x_i\right) - \ln(\lambda^{2n}) - \frac{1}{2\rho^2} \sum_{i=1}^{n} x_i^2$$

$$= \ln\left(\prod_{i=1}^{n} x_i\right) - 2n \ln(\lambda) - \frac{1}{2\rho^2} \sum_{i=1}^{n} x_i^2$$

$$\frac{d\ln(L(\lambda, \vec{x}))}{d\lambda} = 0 - \frac{2n}{\lambda} + \frac{1}{\lambda^3} \sum_{i=1}^{n} x_i^2 = 0$$

$$2n\lambda^2 = \sum_{i=1}^{n} x_i^2$$

$$\lambda = \sqrt{\frac{\sum_{i=1}^{n} x_i}{2n}}$$

Table of Contents

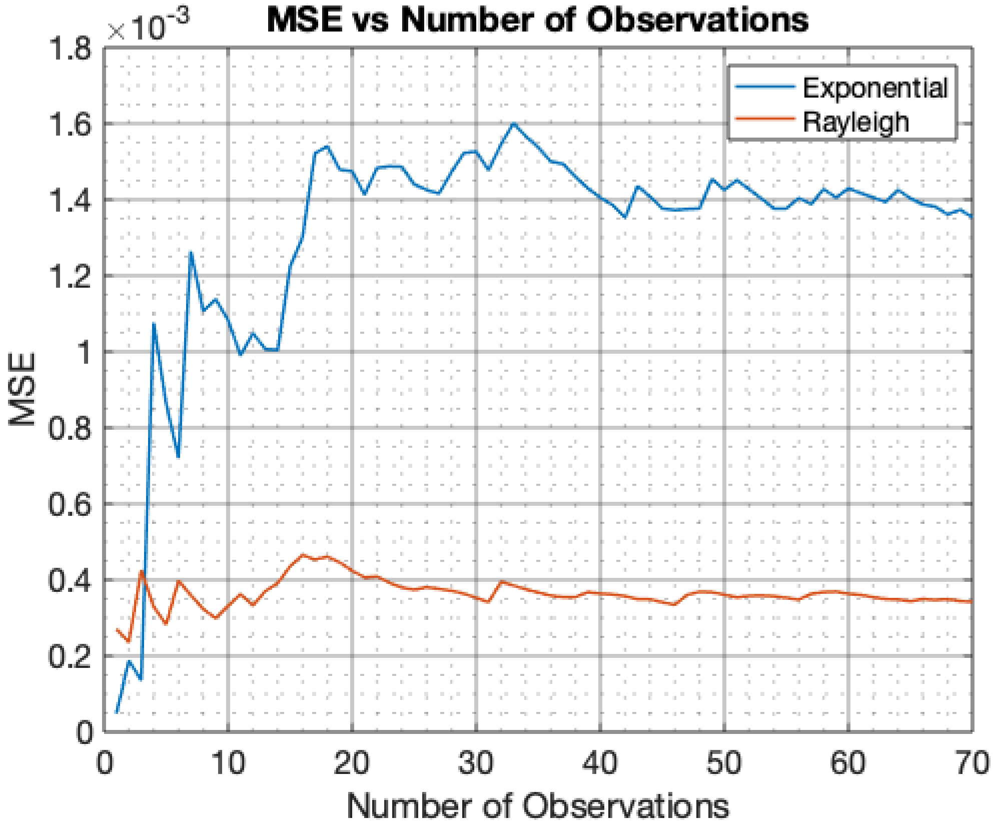
Stoch Proj 3

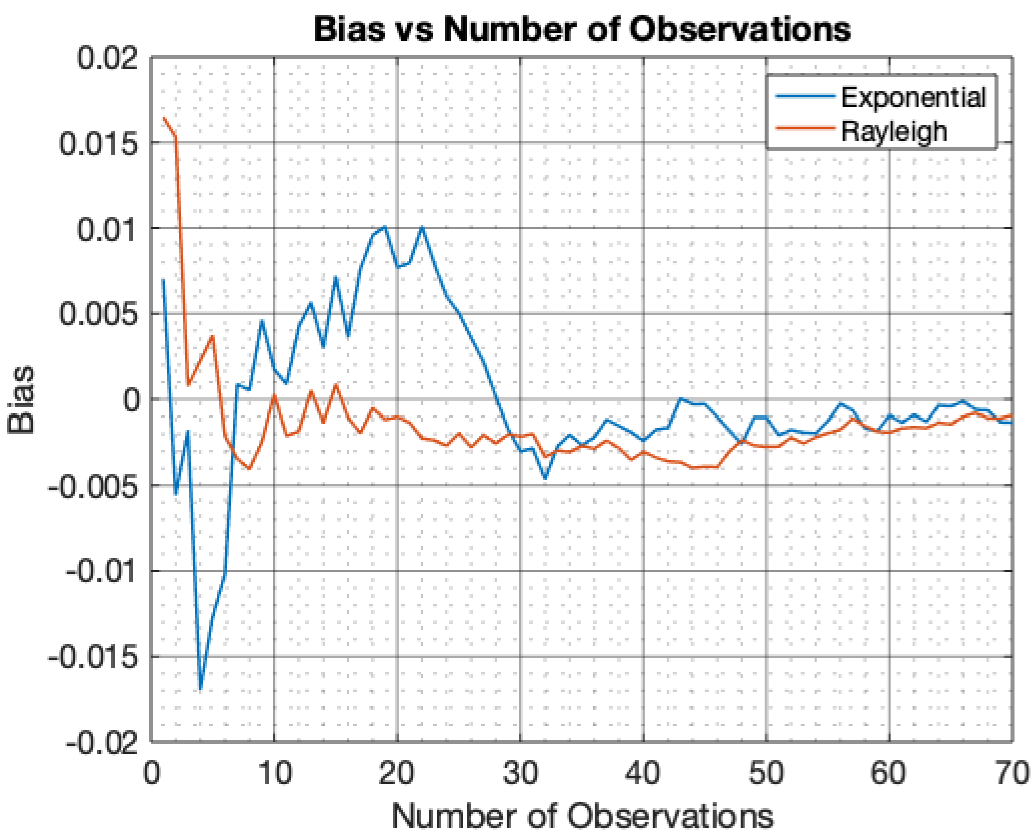
Yuval Epstain Ofek & Jason Kurian

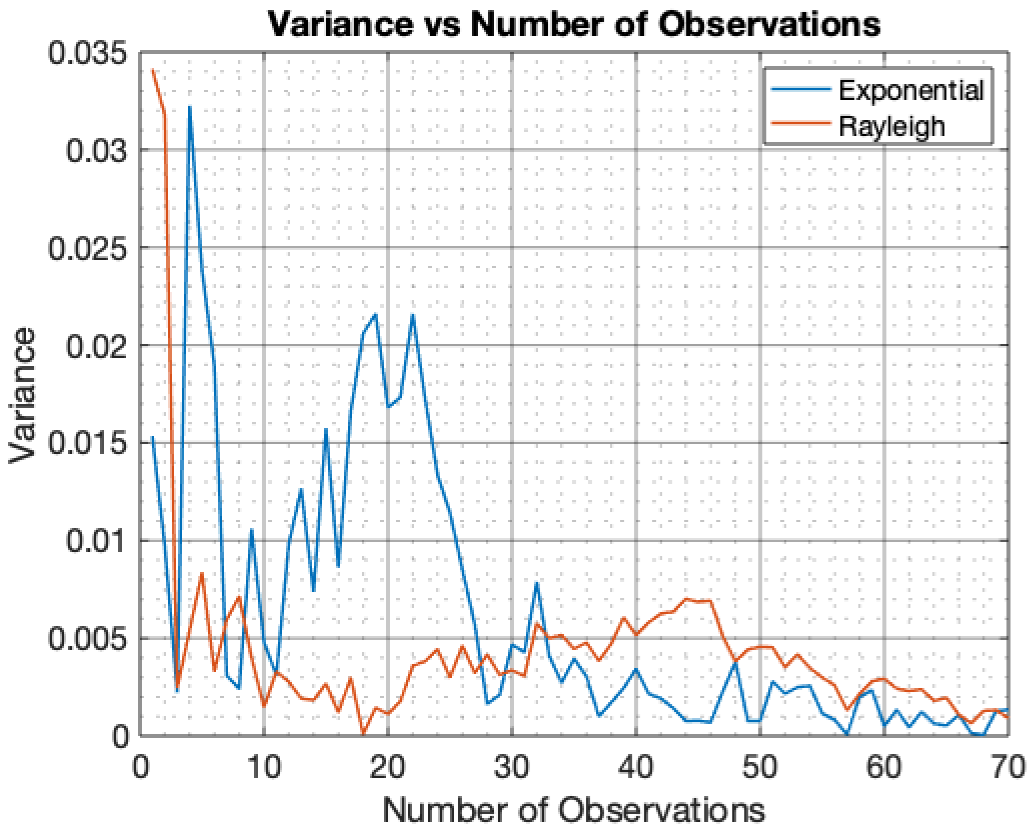
Part 2

```
clear all;close all;clc

%Implemented max likelihood estimates for lambda
estExp = @(data) max(size(data))/sum(data,'all');
estRay = @(data) sqrt(sum(data.^2,'all')/(2*max(size(data))));

N = 1000; % number of random variables
N_obs = 70;
lambda = 1;
% random draws from exponential and Rayleigh distribution
x_exp = exprnd(lambda,[N,N_obs]);
x_ray = raylrnd(lambda,[N,N_obs]);

lambdaexp_est = zeros(N_obs,1);
lambdaray_est = zeros(N_obs,1);
MSE_exp = zeros(N_obs,1);
MSE_ray = zeros(N_obs,1);
bias_exp = zeros(N_obs,1);
bias_ray = zeros(N_obs,1);
% loop to calculate lambda estimate, MSE, and bias for each number of
% observations
for ii = 1:N_obs
    lambdaexp_est(ii) = estExp(x_exp(:,ii));
    lambdaray_est(ii) = estRay(x_ray(:,ii));
    MSE_exp(ii) = mean((lambdaexp_est(1:ii)-lambda).^2);
    MSE_ray(ii) = mean((lambdaray_est(1:ii)-lambda).^2);
    bias_exp(ii) = mean(lambdaexp_est(1:ii))-lambda;
    bias_ray(ii) = mean(lambdaray_est(1:ii))-lambda;
end
% final lambda estimate is the mean of the estimates of all
 observations
lambdahat_exp = mean(lambdaexp_est);
lambdahat_ray = mean(lambdaray_est);

var_exp = zeros(N_obs,1);
var_ray = zeros(N_obs,1);
for ii = 1:N_obs
```

```
    var_exp(ii) = abs(mean((lambdahat_exp-
mean(lambdaexp_est(1:ii)).^2)));
    var_ray(ii) = abs(mean((lambdahat_ray-
mean(lambdaray_est(1:ii)).^2)));
end

%plotting MSE, bias, and variance
figure
plot(1:N_obs,MSE_exp,1:N_obs,MSE_ray, 'LineWidth', 1.5)
title('MSE vs Number of Observations','FontSize', 16)
xlabel('Number of Observations', 'FontSize', 14)
ylabel('MSE', 'FontSize', 14)
grid on;
grid minor
ax = gca;
ax.GridAlpha = 0.5;
ax.FontSize = 16;
legend('Exponential','Rayleigh', 'FontSize', 14)

figure
plot(1:N_obs,bias_exp,1:N_obs,bias_ray, 'LineWidth', 1.5)
title('Bias vs Number of Observations','FontSize', 16)
xlabel('Number of Observations', 'FontSize', 14)
ylabel('Bias', 'FontSize', 14)
grid on;
grid minor
ax = gca;
ax.GridAlpha = 0.5;
ax.FontSize = 16;
legend('Exponential','Rayleigh', 'FontSize', 14)

figure
plot(1:N_obs,var_exp,1:N_obs,var_ray, 'LineWidth', 1.5)
title('Variance vs Number of Observations','FontSize', 16)
xlabel('Number of Observations', 'FontSize', 14)
ylabel('Variance', 'FontSize', 14)
grid on;
grid minor
ax = gca;
ax.GridAlpha = 0.5;
ax.FontSize = 16;
legend('Exponential','Rayleigh', 'FontSize', 14)
```


MSE vs Number of Observations
×10^-3
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
MSE
0
10
20
30
40
50
60
70
Number of Observations
Exponential
Rayleigh


Bias vs Number of Observations
0.02
0.015
0.01
0.005
0
-0.005
-0.01
-0.015
-0.02
Bias
0
10
20
30
40
50
60
70
Number of Observations
Exponential
Rayleigh



Part 3

```
%loading the data
data = load( 'data.mat');
data = data.data;

%We derived the ML estimates earlier, but here is a shorthand
 derivation
%just in case:

% Exponential:
% f(x,lambda) = lambda*e^-lambda*x   x nonegative
%                0 else

% L(lambda) = L(lambda;x1,x2,...,xn) = Product_j(f(xj;lambda)
%           = lambda^n*exp(-lambda[SUM_j(xj)])
%Taking log an finding max
%         l(lambda) = n*log(lambda) -lambda*SUM_j(xj)
%         dl(lambda)/dlambda = n/lambda - SUM_j(xj) = 0
%             => lambda = n/SUM_j(xj)


% Rayleigh
% f(x,lambda) = x/lambda^2 * exp(-x^2/(2*lambda^2))     x nonegative
```

```
% L(lambda) = PROD_j(xj)/(lambda^2n)*exp(-1/2*SUM((xj/lambda)^2)
% Taking log:
% l(lambda) = log(sum_j(xj))-2nlog(lambda)-1/2^n*SUM_j((xj/lambda)^2)
% dl(lambda)/dlambda = -2n/lambda + SUM_j(xj^2/lambda^3) = 0
%        => lambda = sqrt(SUM_j(xj^2)/2n)

%Computing the maximum likelyhood estimates for the data provided
lambda_exp = estExp(data);
lambda_ray = estRay(data);

%Plugging in to the likelyhood functions (log likelyhood) to determine
%which distribution we have:
l_exp = @(data,lambda,n)  (n*log(lambda) - lambda*sum(data,'all'));
l_ray = @(data,lambda,n)
 log(prod(data,'all'))-2*n*log(lambda)-0.5*sum((data./
lambda).^2,'all');

%size of input
n = max(size(data));

%checking the likelyhoods
Likely_exp = l_exp(data,lambda_exp,n)
Likely_ray = l_ray(data,lambda_ray,n)
```

Comments on Results

The data was most likely from an exponential distribution. We note that the -inf likelyhood for the rayleigh distribution is most likely due to machine errors when taking the product of all the elements of the data (machine mistakes the product as zero, yielding a log of -inf, whereas the product is not exactly zero).

```
Likely_exp =

   1.0535e+03


Likely_ray =

  -Inf
```

Published with MATLAB® R2019b