Table of Contents

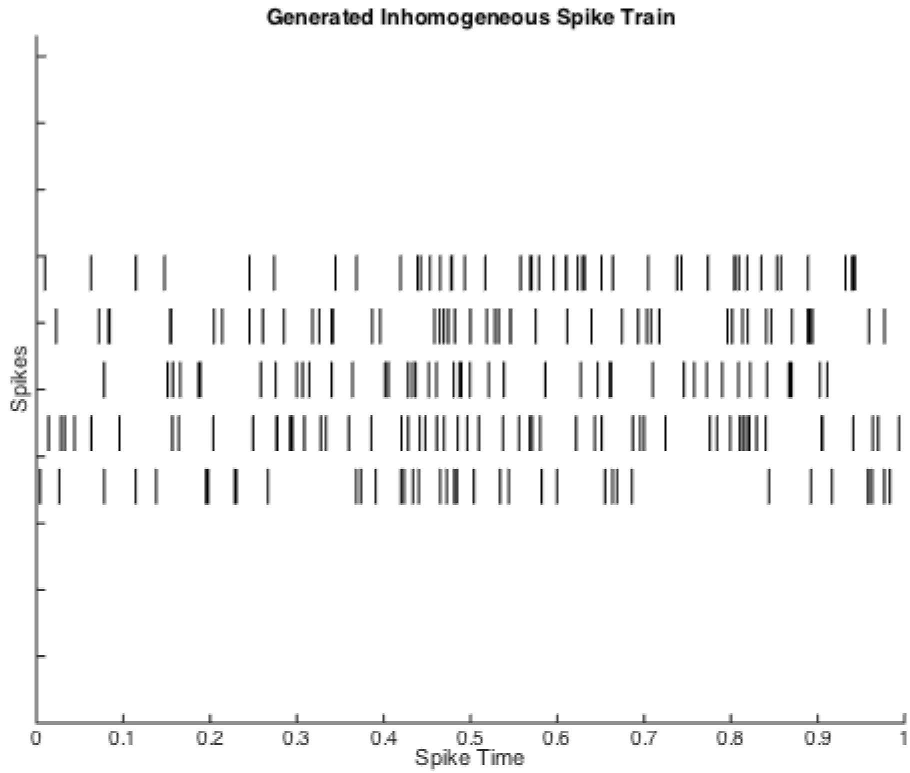

```
% EE239.2 HW 3

clc
clear
close all
```

Problem 3: Inhomogeneous Poisson Process

Part A: Spike Trains

```
r_0 = 35;
r_max = 60;
s_max = 90;

lambda_max = r_max;         % for generating homogenous poisson

n = 0;
T = 0;

time = 1;           % spike trains last 1 second
mu_max = 1./lambda_max;
numTrials = 100;

T_cell = {};        % matrix of 100 spike trains trials per lambda
T_vec = [];

for k = 1:numTrials
    while ( T < time )
        dt = exprnd(mu_max);

        T = T + dt;

        u = rand(1);                % sample from uniform distribution to
                                    % see which ISI values to throw out
        s_Tn = (T^2)*180;
        lambda_Tn = r_0 + (r_max - r_0) * cosd(s_Tn - s_max);
        % calculate instantaneous rate, to compare against maximum rate for
        % deleting values

        if (lambda_Tn/lambda_max) >= u
```

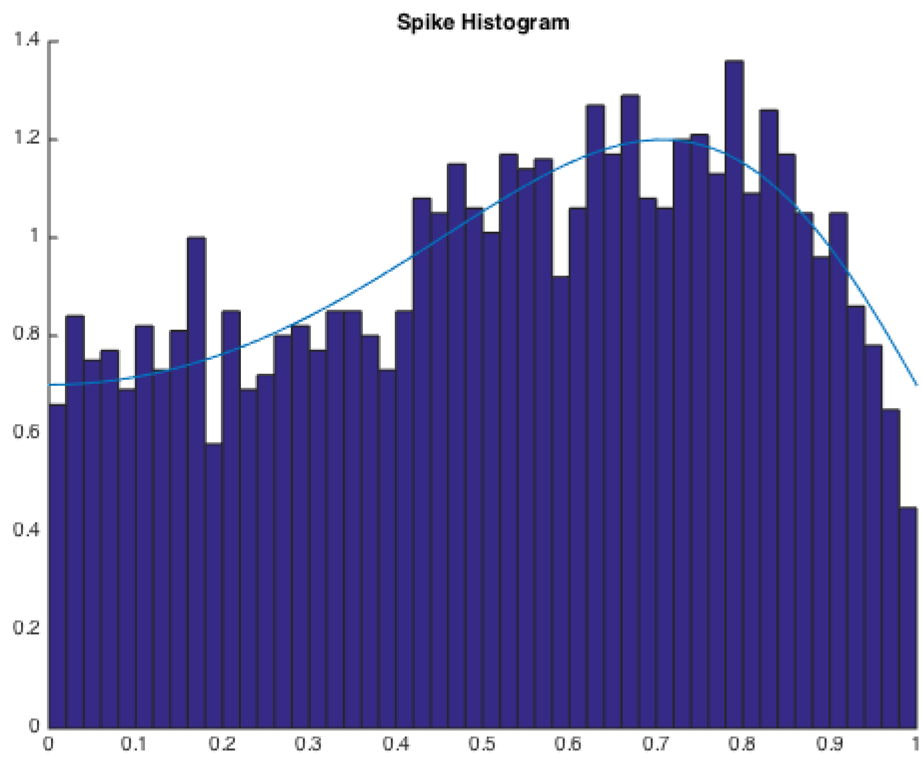
```
            n = n + 1;
            T_vec(n) = T;           % only keep values that satisfy the threshold
        end

    end

    T_vec = T_vec(:,1:end-1);

    % delete last value because it will go over the set time

    T_cell{1,k} = T_vec;
    n = 0;      % reset n, t, and T_vec for each trial
    T = 0;
    T_vec = [];

end

figure(1)
plotRaster(T_cell(1:5))
title('Generated Inhomogeneous Spike Train')
xlabel('Spike Time')
ylabel('Spikes')
```



Part B: Spike Histogram

```
bins = 0:0.020:1;
```

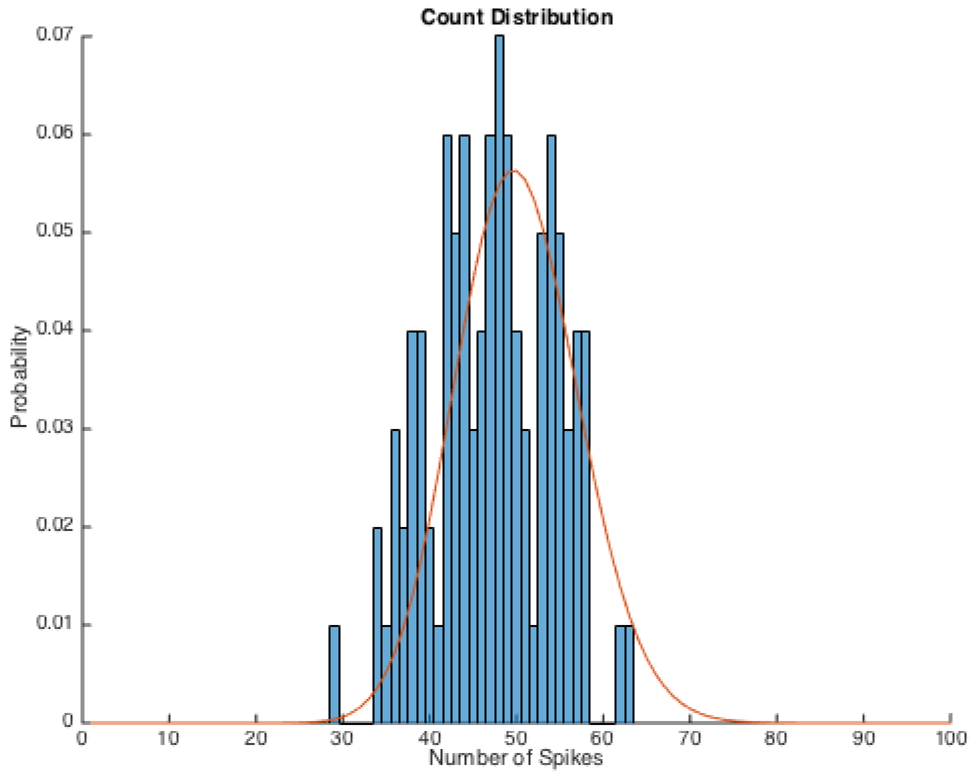
```
counts_sum = zeros(1,length(bins));

for i=1:numTrials
    counts = histc(T_cell{1,i}, bins);
    counts_sum = counts_sum+counts;
end
counts_sum = counts_sum(1:end-1);
counts_avg = counts_sum/numTrials;
figure(2)

lambda_exp = r_0 + (r_max - r_0) * cosd(180*(bins).^2 - s_max);
hold on
bar(bins(1:end-1),counts_avg,'histc')
plot(bins, 0.02*lambda_exp)
title('Spike Histogram')
hold off

% The spike histogram agrees with the expected firing rate profile, as can
% be seen from the plot. The expected firing rate is a function of time,
% and the way we generated the ISIs was also dependent on the instantaneous
% lambda value.
```



Part C: Count Distribution

```
rate = zeros(1,100);
```

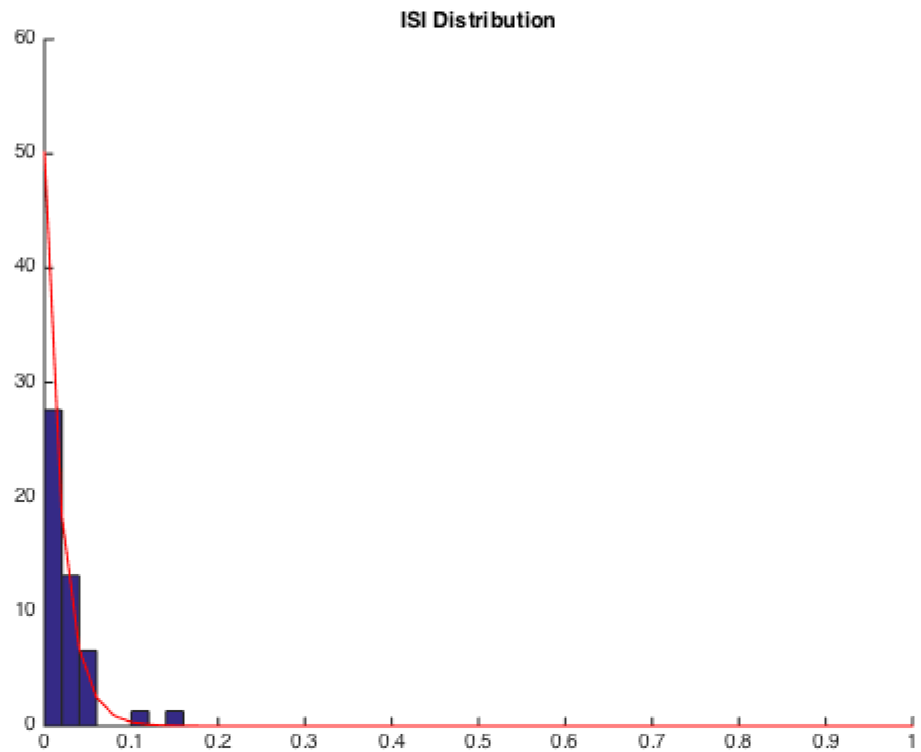
```
lambda_eff = 50.1456;        % integrated firing rate profile from t = 0 to 1

for j=1:numTrials
   rate(j) = length(T_cell{j});
end

x = 1:100;

figure(4)
hold on
histogram(rate,'Normalization','pdf')
plot(x, poisspdf(x,lambda_eff))
title('Count Distribution')
xlabel('Number of Spikes')
ylabel('Probability')
hold off

% The Poisson pdf plotted uses the effective firing rate, which was obtained
% by integrating the firing rate profile from 0 to 1, for the comparison
% against the spike counts. The spike counts are Poisson distributed and
% well fitted by this pdf.
```



Part D: ISI Distribution

```
ISI = {};
```

```
ISI_dist = {};
for k = 1:numTrials
    ISI = [ISI, diff(T_cell{k})];
end

ISI_hist = histcounts(ISI{1},bins,'Normalization','pdf');
ISI_dist = ISI_hist;

figure(6)
hold on
bar(bins(1:end-1),ISI_dist,'histc')
plot(bins, exppdf(bins,1/lambda_eff),'r')
xlim([0 1])
title('ISI Distribution')

% No, the ISIs of Inhomogeneous Poisson process are not exponentially
% distributed. We would not expect it to be, since we sampled from the
% exponential distribution but then also used a threshold to throw away
% certain values using a uniform distributed random variable.
```



Published with MATLAB® R2014b