%%

close all

clear

clc

dt = 0.01E-3;

t\_max = 100;

time\_vector = [0:dt:t\_max];

sigma =50E-12;

I\_app = randn(size(time\_vector))\*sigma/sqrt(dt);

tsteps = 250;

%define parameters

G\_L = 8e-9; % Leak conductance (S)

leak\_potential = -70E-3;

V\_threshold = -50E-3;

reset\_potential = -80E-3;

%R\_m = 100E6;

R\_m = 1/10E-9;

C\_m = 100E-12;

tau\_m = R\_m\*C\_m;

a = 2E-9;

b = 0;

tau\_SRA = 150e-3;

delta\_th = 2e-3;

I\_sra = zeros(size(time\_vector));

I\_sra(1) = 0;

V\_m = zeros(size(time\_vector));

V\_m(1) = leak\_potential;

spikes = zeros(size(time\_vector));

for n = 2:length(time\_vector)

V\_m(n)= V\_m(n-1) + (G\_L\*(leak\_potential - V\_m(n-1) + delta\_th\*exp((V\_m(n-1) - V\_threshold)/delta\_th))/tau\_m - I\_sra(n-1)/C\_m + I\_app(n-1)/C\_m)\*dt;

I\_sra(n) = I\_sra(n-1) + (a\*(V\_m(n-1) - leak\_potential)/tau\_SRA - I\_sra(n-1)/tau\_SRA)\*dt;

if V\_m(n) > V\_threshold

V\_m(n) = reset\_potential;

I\_sra(n) = I\_sra(n) + b;

spikes(n) = 1;

end

end

spike\_times = find(spikes)\*dt;

ISI = diff(spike\_times);

f1 = figure;

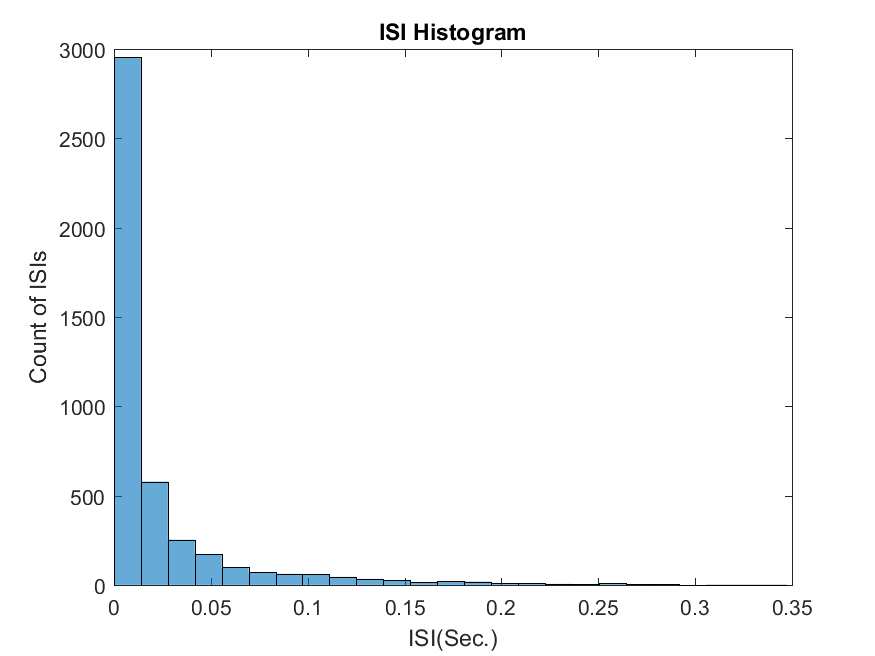
histogram(ISI, 25)

title('ISI Histogram');

xlabel('ISI(Sec.)');

ylabel('Count of ISIs');

saveas(f1, sprintf('f1\_ISIHistogram.png'));



CV=std(ISI)/mean(ISI)

CV=1.8010

dT=0.1;

N=dT/dt;

No\_spikes\_100ms\_window=zeros(size(1:t\_max/100e-3));

for k=1:length(time\_vector)/N

No\_spikes\_100ms\_window(k)=sum(spikes((k-1)\*N+1:k\*N));

end

No=No\_spikes\_100ms\_window

mean\_=(mean(No\_spikes\_100ms\_window))

mean\_ =4.4560

std\_=((std(No\_spikes\_100ms\_window))^2)

std\_ =12.5506

fano = ((std(No\_spikes\_100ms\_window))^2)/(mean(No\_spikes\_100ms\_window))

fano= 2.8166

window=time\_vector(1000:1e5);

spike=zeros(size(1000:1e5));

spike(1)=0;

l=1;

fano\_1=zeros(size(window));

for k=1000:1e5

s=spikes(k);

spike(l+1)=spike(l)+s;

std(spike(l+1));

mean(spike(l+1));

fano\_1(l) = ((std(spike(1:l+1)))^2)/(mean(spike(1:l+1)));

l=l+1;

end

f2=figure

plot(window,fano\_1(1:end))

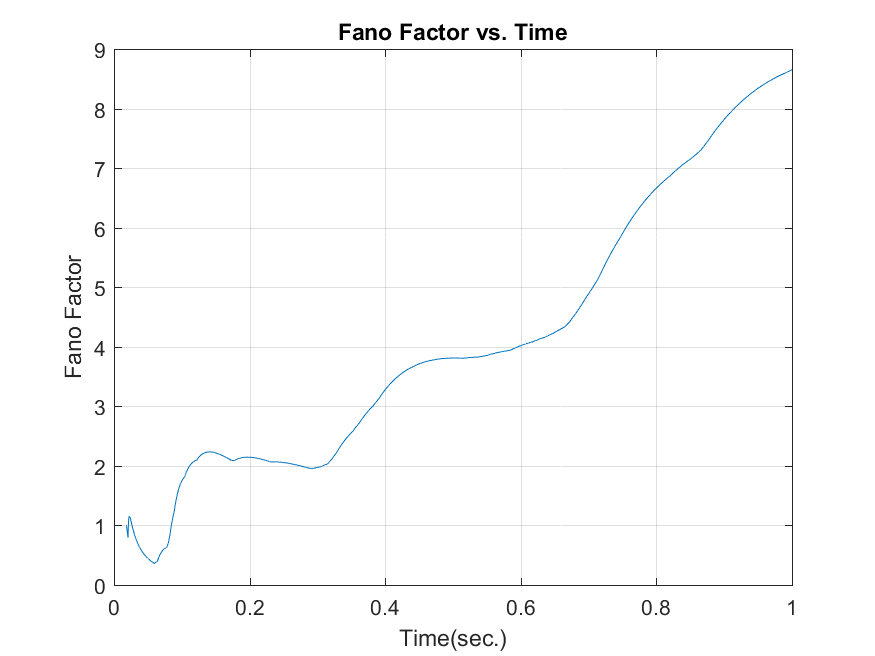
grid on

title('Fano Factor vs. Time');

xlabel('Time(sec.)');

ylabel('Fano Factor');

saveas(f2, sprintf('f2\_FanoFactor.png'));



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%1.b

b=1e-9;

I\_sra\_b = zeros(size(time\_vector));

I\_sra\_b(1) = 0;

V\_m\_b = zeros(size(time\_vector));

V\_m\_b(1) = leak\_potential;

spikes\_b = zeros(size(time\_vector));

for n = 2:length(time\_vector)

V\_m\_b(n)= V\_m\_b(n-1) + (G\_L\*(leak\_potential - V\_m\_b(n-1) + delta\_th\*exp((V\_m\_b(n-1) - V\_threshold)/delta\_th))/tau\_m - I\_sra\_b(n-1)/C\_m + I\_app(n-1)/C\_m)\*dt;

I\_sra\_b(n) = I\_sra\_b(n-1) + (a\*(V\_m\_b(n-1) - leak\_potential)/tau\_SRA - I\_sra\_b(n-1)/tau\_SRA)\*dt;

if V\_m\_b(n) > V\_threshold

V\_m\_b(n) = reset\_potential;

I\_sra\_b(n) = I\_sra\_b(n) + b;

spikes\_b(n) = 1;

end

end

spike\_times\_b = find(spikes\_b)\*dt;

ISI\_b = diff(spike\_times\_b);

f3 = figure;

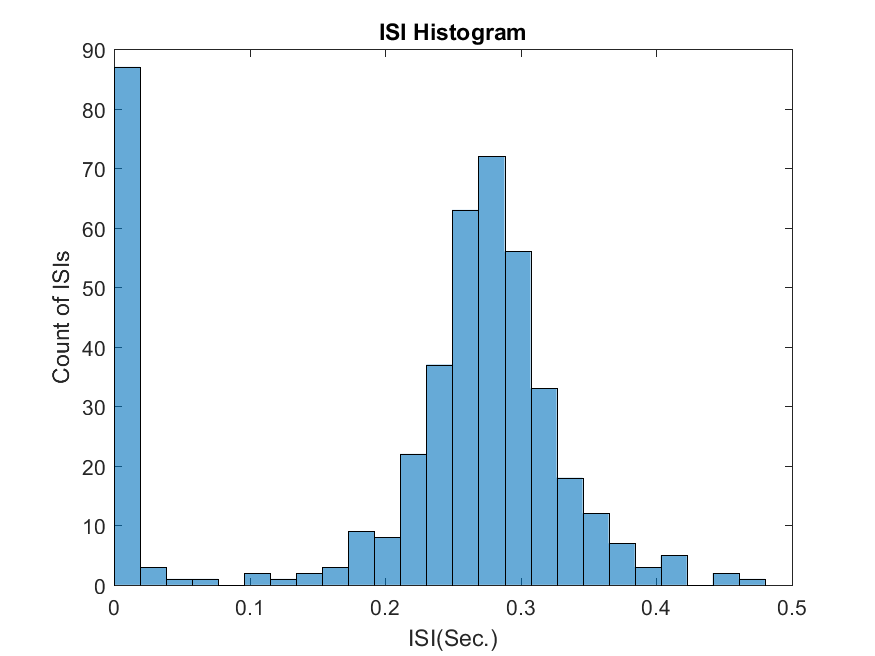
histogram(ISI\_b, 25)

title('ISI Histogram');

xlabel('ISI(Sec.)');

ylabel('Count of ISIs');

saveas(f3, sprintf('ISIHistogram\_2.png'));



CV\_2=std(ISI\_b)/mean(ISI\_b)

CV\_2 =0.5322

dT=0.1;

N=dT/dt;

No\_spikes\_100ms\_window\_b=zeros(size(1:t\_max/100e-3));

for k=1:length(time\_vector)/N

No\_spikes\_100ms\_window\_b(k)=sum(spikes\_b((k-1)\*N+1:k\*N));

end

No\_b=No\_spikes\_100ms\_window\_b

mean\_b=(mean(No\_spikes\_100ms\_window\_b))

mean\_b =0.4490

std\_b=((std(No\_spikes\_100ms\_window\_b))^2)

std\_b =0.4218

fano\_b = ((std(No\_spikes\_100ms\_window\_b))^2)/(mean(No\_spikes\_100ms\_window\_b))

fano\_b =0.9395

window=time\_vector(1000:1e5);

spike\_b=zeros(size(1000:1e5));

spike\_b(1)=0;

l=1;

fano\_1\_b=zeros(size(window));

for k=1000:1e5

s=spikes\_b(k);

spike\_b(l+1)=spike\_b(l)+s;

std(spike\_b(l+1));

mean(spike\_b(l+1));

fano\_1\_b(l) = ((std(spike\_b(1:l+1)))^2)/(mean(spike\_b(1:l+1)));

l=l+1;

end

f4=figure

plot(window,fano\_1\_b(1:end))

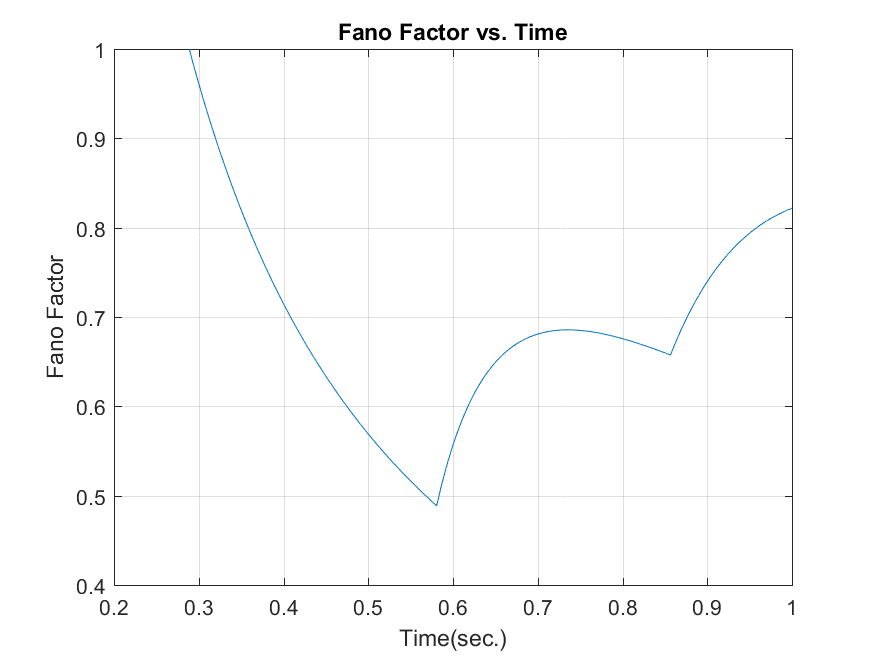
grid on

title('Fano Factor vs. Time');

xlabel('Time(sec.)');

ylabel('Fano Factor');

saveas(f4, sprintf('f4\_FanoFactor\_b.png'));



* With a higher adaptation strength, the after hyperpolarization following a spike will be deeper and longer, making it harder for the neuron to reach the threshold potential for the next spike. This will lead to an increase in the average ISI, resulting in a shift of the ISI histogram towards longer ISIs.
* The ISI distribution is widen, indicating increased variability in spike timing.
* The Fano Factor measures the variability in the number of spikes within a specific time window. Since increasing b reduces the variability in ISIs, it will also lead to a decrease in the Fano Factor. This implies the spiking becomes more regular and less Poisson-like.
* The Fano Factor gets closer to 1 Depending on the original Fano factor, it gets closer to 1 indicating purely Poisson-like spiking, where the variance equals the mean, suggesting high randomness.

%%

%%%%%%%%%%%%%%%%%%%%%%%%

%1.c

b=0;

sigma =20E-12;

I\_sra = zeros(3,length(time\_vector));

I\_sra(:,1) = 0;

V\_m = zeros(3,length(time\_vector));

V\_m(:,1) = leak\_potential;

spikes = zeros(3,length(time\_vector));

for w=1:3

I\_app(w,:) = randn(size(time\_vector))\*sigma/sqrt(dt)+0.1e-9\*(w-1);

for n = 2:length(time\_vector)

V\_m(w,n)= V\_m(w,n-1) + (G\_L\*(leak\_potential - V\_m(w,n-1) + delta\_th\*exp((V\_m(w,n-1) - V\_threshold)/delta\_th))/tau\_m - I\_sra(w,n-1)/C\_m + I\_app(w,n-1)/C\_m)\*dt;

I\_sra(w,n) = I\_sra(w,n-1) + (a\*(V\_m(w,n-1) - leak\_potential)/tau\_SRA - I\_sra(w,n-1)/tau\_SRA)\*dt;

if V\_m(w,n) > V\_threshold

V\_m(w,n) = reset\_potential;

I\_sra(w,n) = I\_sra(w,n) + b;

spikes(w,n) = 1;

end

end

end

spike\_times\_0 = find(spikes(1,:))\*dt;

spike\_times\_1 = find(spikes(2,:))\*dt;

spike\_times\_2 = find(spikes(3,:))\*dt;

ISI\_0 = diff(spike\_times\_0);

ISI\_1 = diff(spike\_times\_1);

ISI\_2 = diff(spike\_times\_2);

f5 = figure;

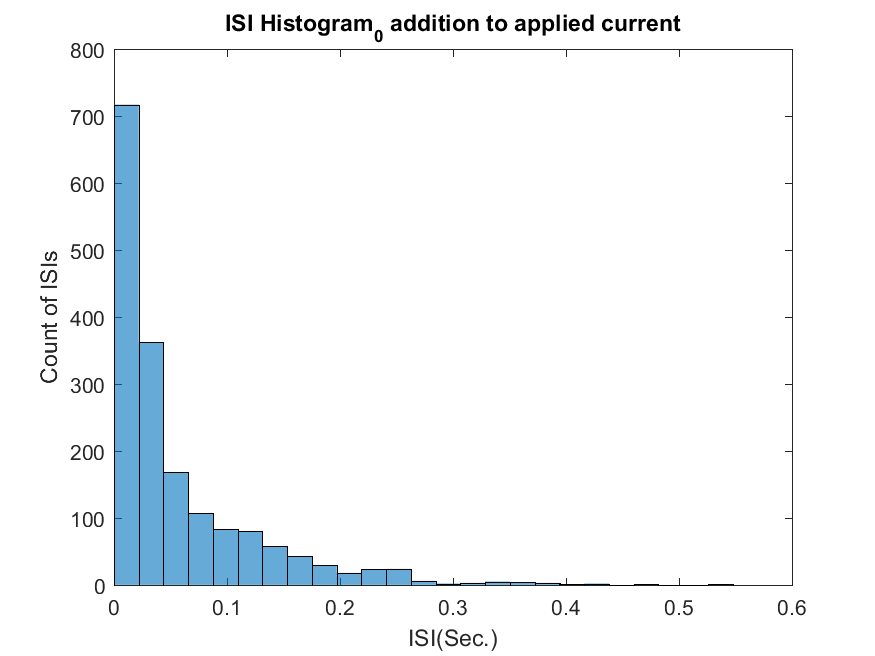
histogram(ISI\_0, 25)

title('ISI Histogram\_0 addition to applied current');

xlabel('ISI(Sec.)');

ylabel('Count of ISIs');

saveas(f5, sprintf('f5\_ISIHistogram\_1c\_0.png'));



CV=std(ISI\_0)/mean(ISI\_0)

CV= 1.1925

f6=figure;

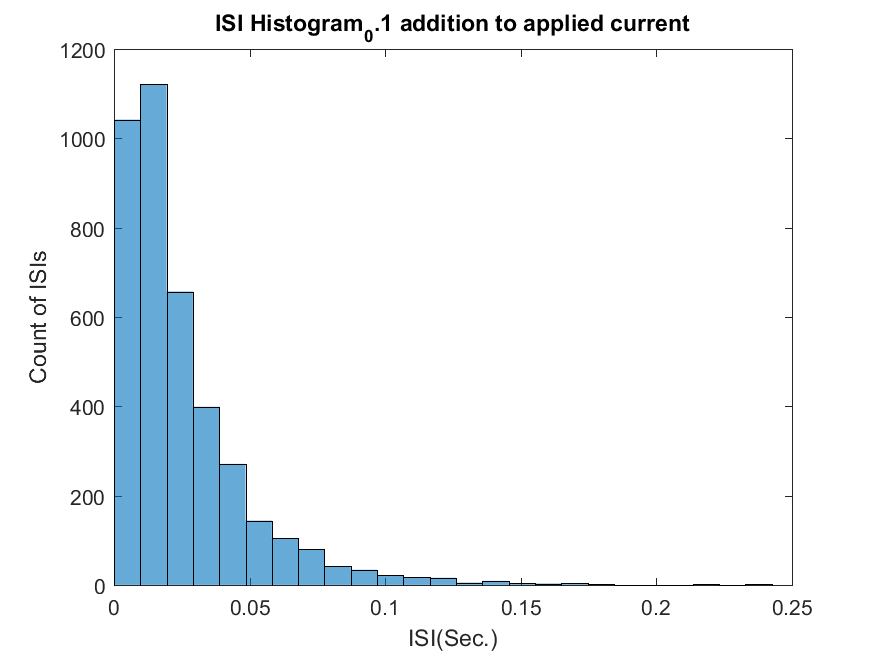
histogram(ISI\_1, 25)

title('ISI Histogram\_0.1 addition to applied current');

xlabel('ISI(Sec.)');

ylabel('Count of ISIs');

saveas(f6, sprintf('f6\_ISIHistogram\_1c\_01.png'));



CV=std(ISI\_1)/mean(ISI\_1)

CV= 0.9556

f7=figure;

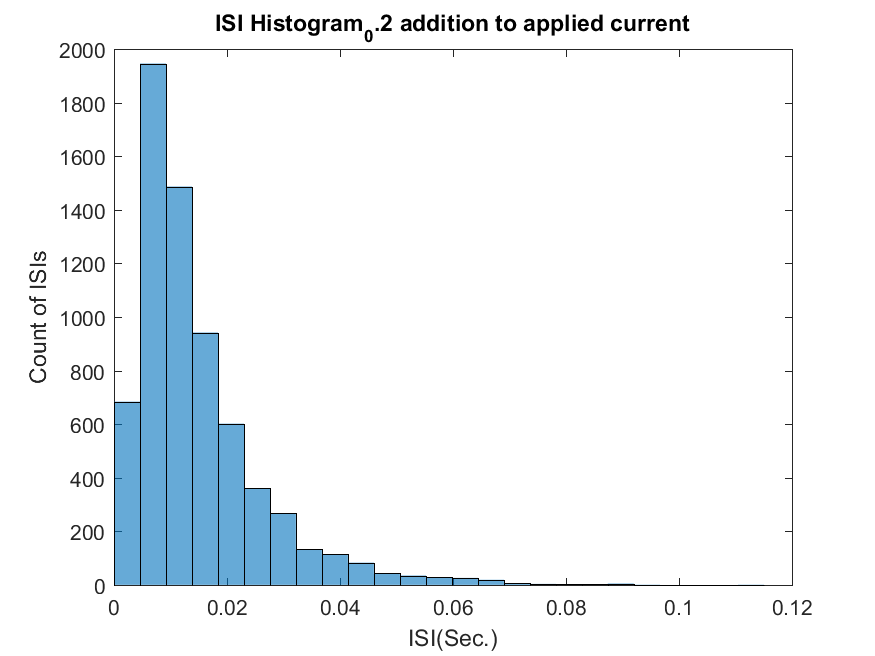
histogram(ISI\_2, 25)

title('ISI Histogram\_0.2 addition to applied current');

xlabel('ISI(Sec.)');

ylabel('Count of ISIs');

saveas(f7, sprintf('f7\_ISIHistogram\_1c\_02.png'));



CV=std(ISI\_2)/mean(ISI\_2)

CV= 0.7872

dT=0.1;

N=dT/dt;

No\_spikes\_100ms\_window=zeros(3,length(1:t\_max/100e-3));

fano\_c=zeros(1,3)

for w=1:3

for k=1:length(time\_vector)/N

No\_spikes\_100ms\_window(w,k)=sum(spikes(w,((k-1)\*N+1:k\*N)));

end

fano\_c(w) = ((std(No\_spikes\_100ms\_window(w,:)))^2)/(mean(No\_spikes\_100ms\_window(w,:)))

end

* The ISI distribution will be sharper with higher number of ISI counts in the left, meaning shorter ISIs will become more frequent. This is because higher current brings the membrane potential closer to the threshold, making it easier for the neuron to reach and cross the threshold, leading to more frequent spiking.
* The ISI distribution becomes narrower, indicating less variability in spike timing. This is because the increased current reduces the role of intrinsic noise and randomness in determining the timing of spikes.
* The Fano factor, which measures the variance in spike count relative to the mean, decrease because the stronger current reduces the randomness in spiking, leading to smaller fluctuations in spike count within windows.

%%

%Part B

dt = 0.1e-3; % s

rate = 20; % Hz

t\_max=100;

time\_vector\_B = 0:dt:100-dt; % initializing time

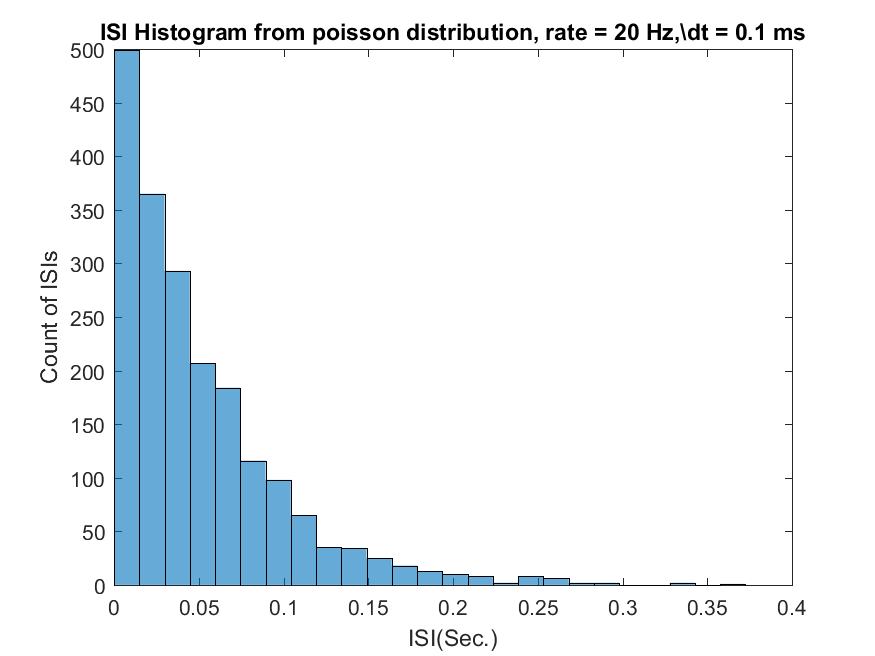
SPIKES = rand(1,length(time\_vector\_B))<rate\*dt;

ISI = diff(find(SPIKES))\*dt; % calculating ISI

f8=figure

histogram(ISI,25); title('ISI Histogram from poisson distribution, rate = 20 Hz,\dt = 0.1 ms'); xlabel('ISI(Sec.)'); ylabel('Count of ISIs');

saveas(f8, sprintf('f8\_PoissonHistogram.png'));



CV=std(ISI)/mean(ISI)

CV= 0.9740

dT=0.1;

N=dT/dt;

No\_spikes\_100ms\_window=zeros(size(1:t\_max/100e-3));

for k=1:length(time\_vector\_B)/N

No\_spikes\_100ms\_window(k)=sum(SPIKES((k-1)\*N+1:k\*N));

end

No9=No\_spikes\_100ms\_window

mean9=(mean(No\_spikes\_100ms\_window))

std9=((std(No\_spikes\_100ms\_window))^2)

fano = ((std(No\_spikes\_100ms\_window))^2)/(mean(No\_spikes\_100ms\_window))

window=time\_vector\_B(100:1e4);

spike=zeros(size(100:1e4));

spike(1)=0;

l=1;

fano\_1=zeros(size(window));

for k=100:1e4

s=SPIKES(k);

spike(l+1)=spike(l)+s;

std(spike(l+1));

mean(spike(l+1));

fano\_1(l) = ((std(spike(1:l+1)))^2)/(mean(spike(1:l+1)));

l=l+1;

end

f9=figure

plot(window,fano\_1(1:end))

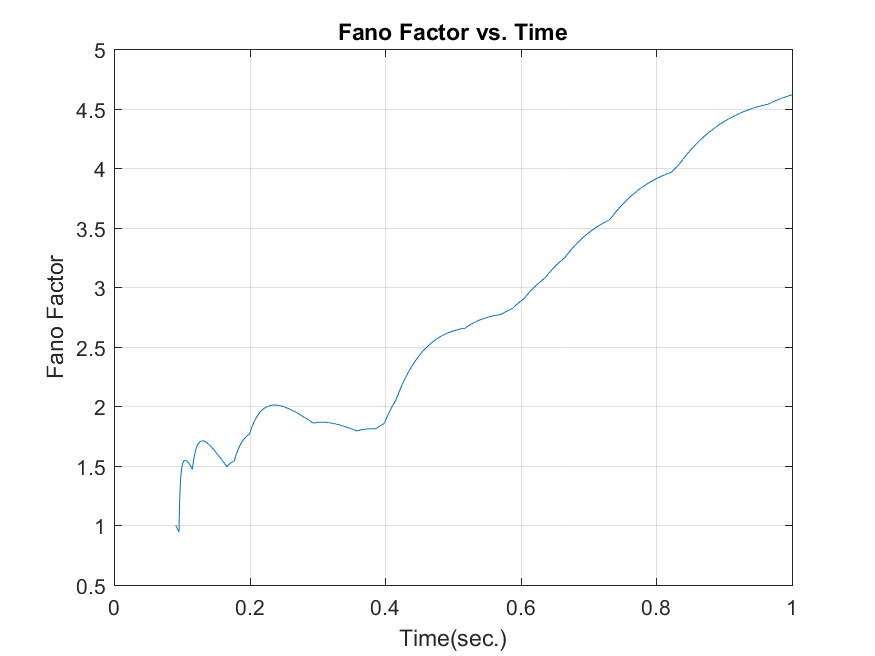
grid on

title('Fano Factor vs. Time');

xlabel('Time(sec.)');

ylabel('Fano Factor');

saveas(f9, sprintf('f9\_FanoFactor.png'));



%%2b-B

t=0:dt:10-dt;

for k=1:1000

spikemat(k,:)=rand(size(t))<rate\*dt;

end

f10=figure

plot(spikemat(1,:), 'b-', 'LineWidth', 2);

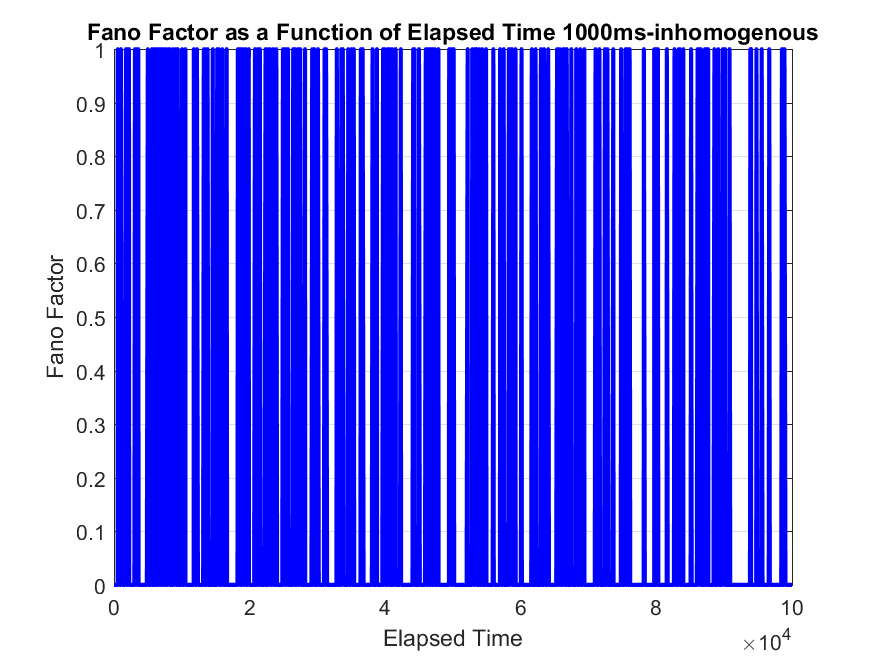
xlabel('Elapsed Time');

ylabel('Fano Factor');

title('Fano Factor as a Function of Elapsed Time 1000ms-inhomogenous');

grid on;

saveas(f10, sprintf('f10\_FanoFactor.png'));



N\_T = cumsum(spikemat,2)

variance\_across\_rows = var(N\_T,0, 1)

mean\_across\_rows = mean(N\_T, 1)

% Calculate Fano factor

fano\_factor = variance\_across\_rows ./ mean\_across\_rows

% Plot the Fano factor as a function of elapsed time

elapsed\_time = 1:size(N\_T, 2); % Assuming each row corresponds to a specific time point

f11=figure;

plot(elapsed\_time, fano\_factor, 'b-', 'LineWidth', 2);

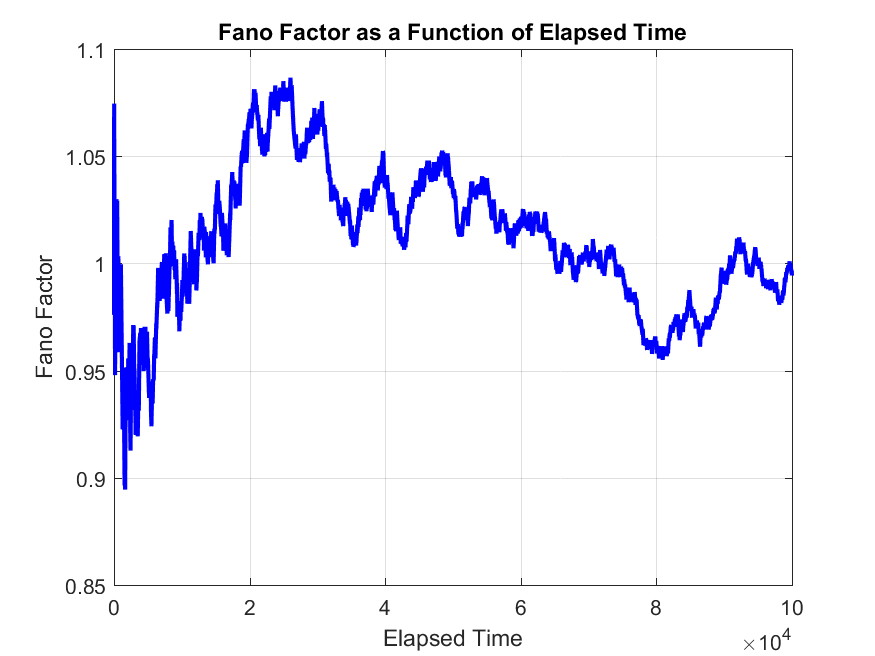
xlabel('Elapsed Time');

ylabel('Fano Factor');

title('Fano Factor as a Function of Elapsed Time');

grid on;

saveas(f11, sprintf('f11\_FanoFactor.png'));



% Assuming you have spike times stored in the variable 'spike\_times' and the duration of each trial is known

window\_size = 0.2; % 200 ms window size

trial\_duration = 10; % Duration of each trial (in seconds), provide the actual value

% Calculate the number of time windows

num\_trials =(trial\_duration / window\_size);

num\_windows=window\_size/dt;

% Initialize arrays to store spike counts for each window and trial

spike\_counts = zeros(num\_trials,length(t));

% Loop over each trial

for trial = 1:num\_trials

% Loop over each time windowt

for window = 1+(trial-1)\*num\_windows:num\_windows\*trial

% Define the start and end time of the current window

% Count the number of spikes within the current window for the current trial

spike\_counts(trial,:) = spikemat(trial,window)+spike\_counts(trial,:) ;

end

end

N\_T = cumsum(spike\_counts,2)

variance\_across\_rows = var(N\_T, 0,1)

mean\_across\_rows = mean(N\_T, 1)

% Calculate Fano factor

fano\_factor = variance\_across\_rows ./ mean\_across\_rows

% Plot the Fano factor as a function of elapsed time

elapsed\_time = 1:size(spike\_counts, 2);

f12=figure

plot(fano\_factor, 'b-', 'LineWidth', 2);

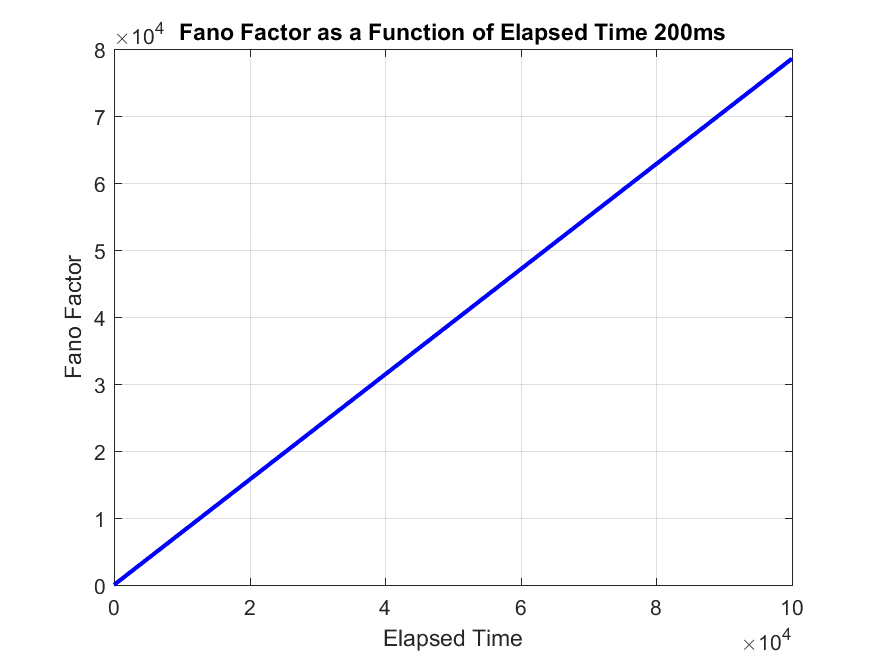
xlabel('Elapsed Time');

ylabel('Fano Factor');

title('Fano Factor as a Function of Elapsed Time 200ms');

grid on;

saveas(f12, sprintf('f12\_FanoFactor\_200ms.png'));



%%%%%

t=0:dt:10-dt;

rate = 25 + 20\*sin(2\*pi\*t);

for k=1:1000

spikemat(k,:)=rand(size(t))<rate\*dt;

end

f13=figure

plot(spikemat(1,:), 'b-', 'LineWidth', 2); %one trial

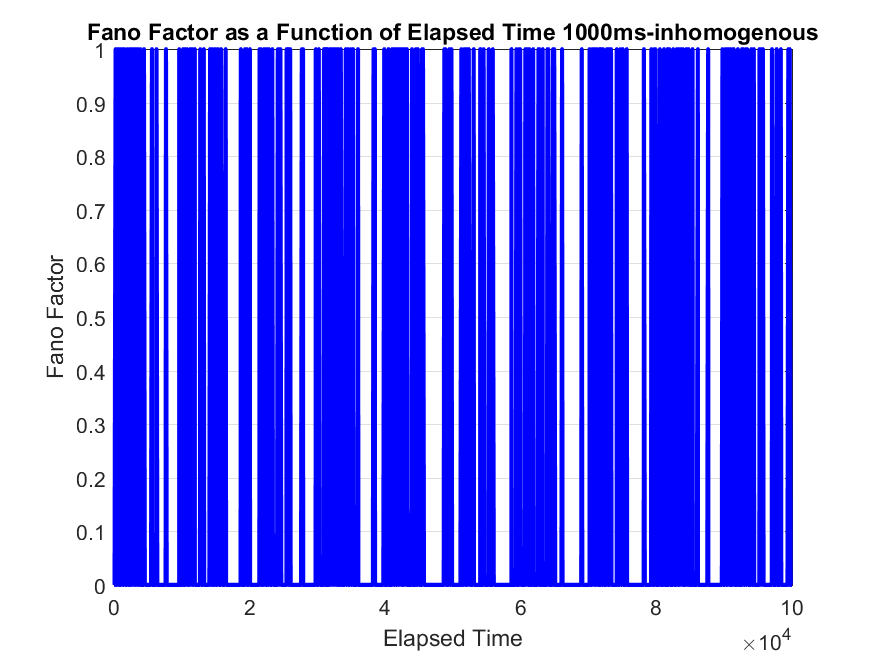
xlabel('Elapsed Time');

ylabel('Fano Factor');

title('Fano Factor as a Function of Elapsed Time 1000ms-inhomogenous');

grid on;

saveas(f13, sprintf('f13\_FanoFactor\_1000ms.png'));



N\_T = cumsum(spikemat,2);

variance\_across\_rows = var(N\_T,0, 1)

mean\_across\_rows = mean(N\_T, 1)

fano\_factor = variance\_across\_rows ./ mean\_across\_rows

elapsed\_time = 1:size(N\_T, 2);

f14=figure;

plot(fano\_factor, 'b-', 'LineWidth', 2);

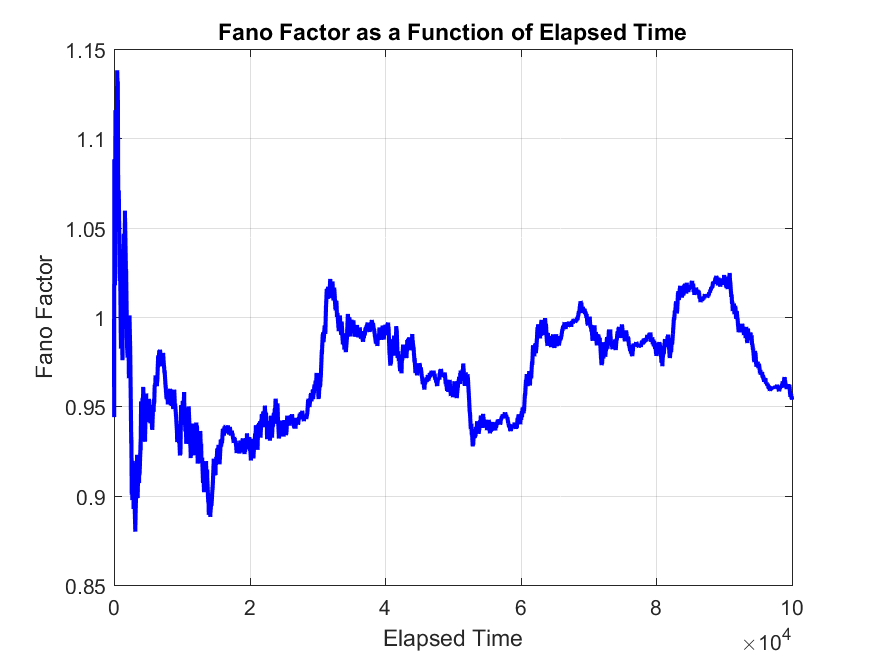
xlabel('Elapsed Time');

ylabel('Fano Factor');

title('Fano Factor as a Function of Elapsed Time');

grid on;

saveas(f14, sprintf('f14\_FanoFactor-1000.png'));



window\_size = 0.2; % 200 ms window size

trial\_duration = 10; % Duration of each trial (in seconds), provide the actual value

% Calculate the number of time windows

num\_trials =(trial\_duration / window\_size);

num\_windows=window\_size/dt;

% Initialize arrays to store spike counts for each window and trial

spike\_counts = zeros(num\_trials,length(t));

% Loop over each trial

for trial = 1:num\_trials

% Loop over each time windowt

for window = 1+(trial-1)\*num\_windows:num\_windows\*trial

% Define the start and end time of the current window

% Count the number of spikes within the current window for the current trial

spike\_counts(trial,:) = spikemat(trial,window)+spike\_counts(trial,:) ;

end

end

N\_T = cumsum(spike\_counts,2);

variance\_across\_rows = var(N\_T, 0,1);

mean\_across\_rows = mean(N\_T, 1);

% Calculate Fano factor

fano\_factor = variance\_across\_rows ./ mean\_across\_rows

Columns 99947 through 99960

0.9545 0.9542 0.9543 0.9543 0.9543 0.9548 0.9548 0.9551 0.9551 0.9551 0.9549 0.9551 0.9548 0.9550

Columns 99961 through 99974

0.9552 0.9551 0.9551 0.9552 0.9552 0.9551 0.9551 0.9553 0.9552 0.9554 0.9553 0.9553 0.9553 0.9553

Columns 99975 through 99988

0.9550 0.9550 0.9547 0.9550 0.9550 0.9550 0.9548 0.9546 0.9547 0.9546 0.9547 0.9546 0.9548 0.9545

Columns 99989 through 100000

0.9547 0.9548 0.9549 0.9549 0.9549 0.9550 0.9552 0.9545 0.9544 0.9544 0.9540 0.9536

% Plot the Fano factor as a function of elapsed time

elapsed\_time = 1:size(N\_T, 2);

f15=figure

plot(fano\_factor, 'b-', 'LineWidth', 2);

xlabel('Elapsed Time');

ylabel('Fano Factor');

title('Fano Factor as a Function of Elapsed Time 200ms');

grid on;

saveas(f15, sprintf('f15\_FanoFactor\_200ms.png'));

