BMES 710: Extra Credit 2

AUTHOR
Tony Kabilan Okeke

PUBLISHED
November 16, 2023

Generating a Poisson Spike Train

1. Select a small Δt (0.001 seconds)
2. Select a firing rate (in Hz)
3. Make a time vector (100 seconds, step size is Δt)
4. Spikes=rand(size(time_vector)) < rate * Δt ;
5. Plot your spike train
6. Plot the ISI histogram for your spike train and overlay the Poisson ISI distribution (theoretical)
    - hint: make tau step sizes and bin sizes for your histogram the same
7. Calculate the CV for your ISI

```
import matplotlib.pyplot as plt
import numpy as np

# set seed for reproducibility
np.random.seed(69)

# Step 1: Select a small Δt (0.001 seconds)
dt = 0.001

# Step 2: Select a firing rate (in Hz)
rate = 120

# Step 3: Make a time vector (100 seconds, step size is Δt)
time_vector = np.arange(0, 100, dt)

# Step 4: Generate spikes
spikes = np.random.rand(len(time_vector)) < rate * dt
spike_locs = np.where(spikes)[0]

# Step 5: Plot spike train
fig, ax = plt.subplots(figsize=(6, 2))
ax.eventplot(spike_locs, lineoffsets=0.5, color="black")
ax.set_title("Spike Train")
ax.set_xlabel("Time (s)")
ax.set_ylabel("Spikes")
ax.set_xlim([0, 100])
ax.set_ylim([0, 1])
ax.spines[['top', 'right']].set_visible(False)
```

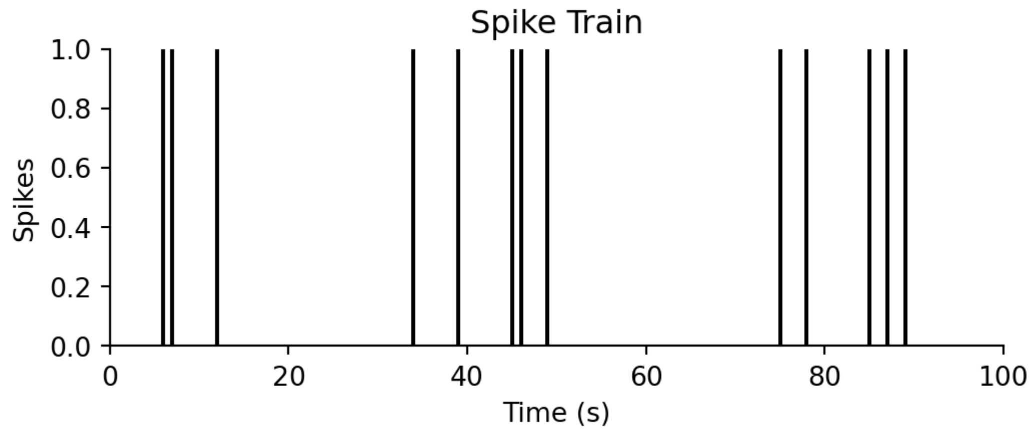

Figure 1: Spike Train generated using a Poisson Process with a firing rate of 180 Hz.

```
# Step 6: Plot ISI histogram and overlay Poisson ISI distribution
# Calculate ISI
isi = np.diff(time_vector[spike_locs])
# Histogram of ISI
hist_bins = np.arange(0, np.max(isi), dt)
hist_counts, _ = np.histogram(isi, bins=hist_bins)
# Convert to probability
hist_vals = hist_counts / sum(hist_counts)

# Poisson distribution for the given rate
tau = hist_bins[:-1]  # use the midpoints of the bins for tau
poisson_isi = rate * np.exp(-rate * tau) * dt

# Overlay the theoretical Poisson distribution on the histogram
fig, ax = plt.subplots(figsize=(6, 3))
ax.bar(
  tau, hist_vals, width=np.diff(hist_bins)[0], color='grey',
  label='Empirical ISI Distribution'
)
ax.plot(tau, poisson_isi, 'r-', label='Theoretical Poisson ISI Distribution')
ax.set_title('ISI Histogram and Poisson ISI Distribution Overlay')
ax.set_xlabel('Interspike Interval (s)')
ax.set_ylabel('Probability')
ax.legend(frameon=False)
ax.spines[['top', 'right']].set_visible(False)

# Step 7: Calculate the CV for your ISI
# CV = Standard Deviation / Mean of ISI
CV = np.std(isi) / np.mean(isi)
ax.text(0.4, 0.6, f'CV: {CV:.4f}', transform=ax.transAxes, verticalalignment='top');
```

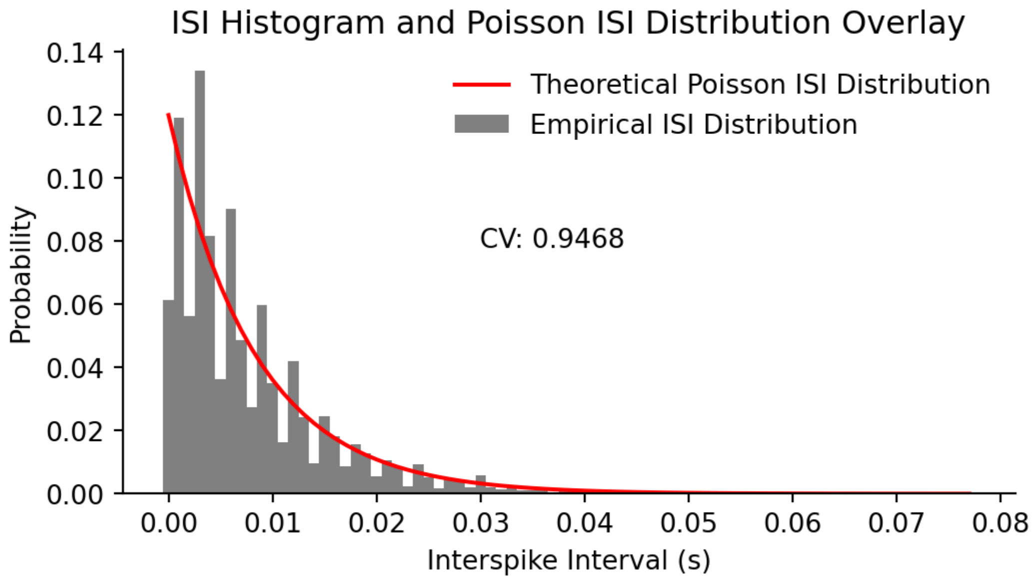

Figure 2: ISI Histogram and Poisson ISI Distribution Overlay