Introduction to Neuromorphic Engineering

Assignment 1

Ashutosh Kumar Jatav [B16CS004]

Usage

Requirements

- NumPy
- Matplotlib

How to Use

- Python Assignment.py -h -n -m -p
- N: Length of Stimulus
- M: Length of window
- P: probability of spike

Implementation

- **Stimulus:** Length N stimulus has been generated by assuming gaussian distribution
- **Spikes:** Binomial Distribution has been assumed to generate spikes in an interval of length n with probability p
- **Filter:**
    - M length windows has been sliced from stimulus at the position of spikes
    - Average of all such windows will give us filter
- **Smooth Stimulus:** Stimulus has been convoluted with filter
- **Histogram**
    - Histogram of filtered stimulus is generated with (n/10) bins '$\boldsymbol{P}(\boldsymbol{Sf})$'.
    - For generation of $\boldsymbol{P}(\boldsymbol{Sf}|\boldsymbol{r})$ histogram values of filtered stimulus has been chosen at the position of spikes.
    - Probability of response given Stimulus then calculated as follow

$$\boldsymbol{P}(\boldsymbol{r}|\boldsymbol{Sf}) = \boldsymbol{P}(\boldsymbol{Sf}|\boldsymbol{r}) \cdot \boldsymbol{P}(\boldsymbol{r}) \,/\, \boldsymbol{P}(\boldsymbol{Sf})$$

$\boldsymbol{P}(\boldsymbol{r})$ is the rate of spike which is taken equal to p

Results

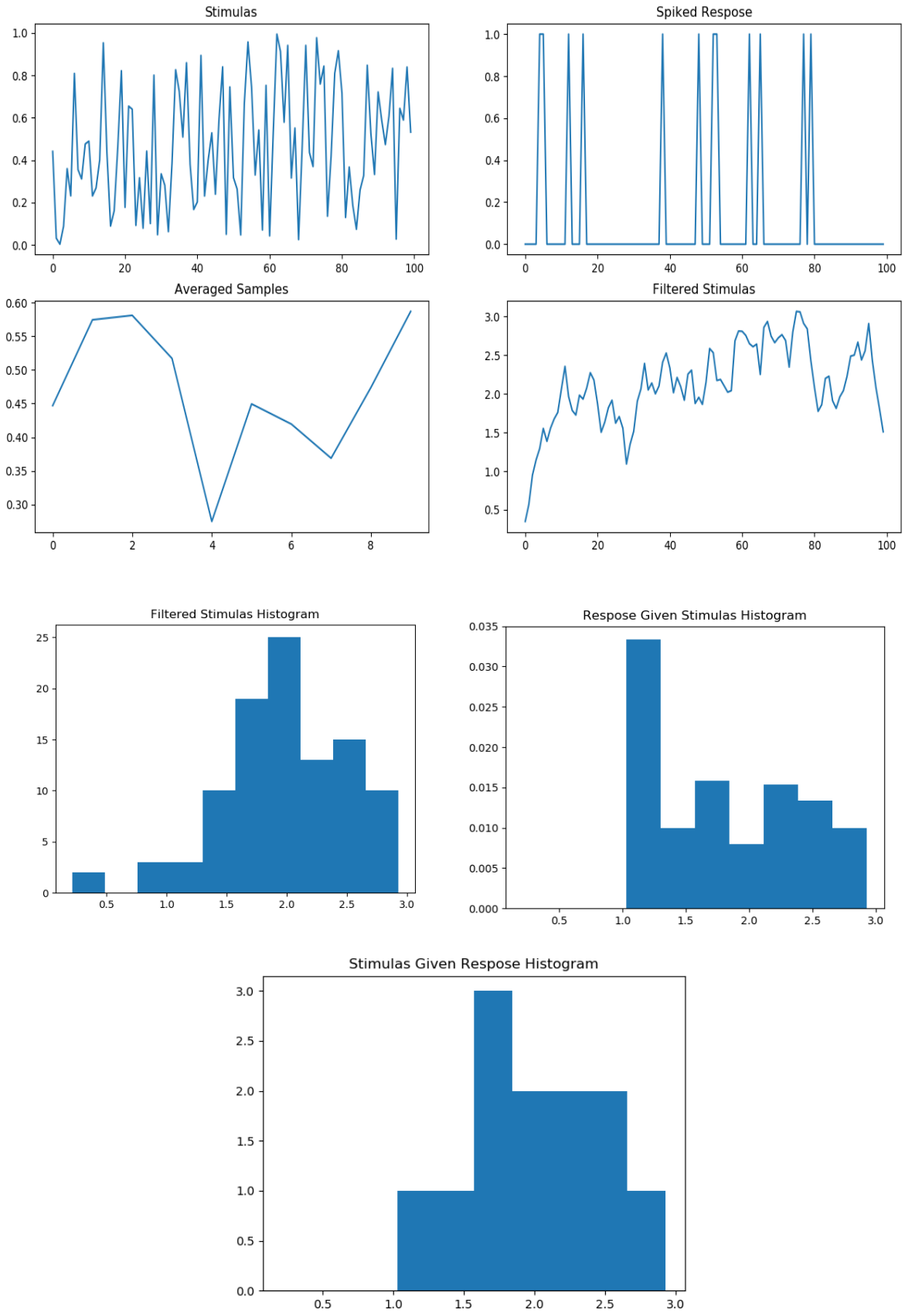

Figure 1 n = 100, m = 10, p = 0.1

For n = 100, m=20, p=0.1

Figure 2 n = 100, m = 20, p = 0.1

For n = 1000, m=10, p=0.1

Figure 3 n = 1000, m = 10, p = 0.1

For n = 1000, m=50, p=0.1

Figure 4 n = 1000, m = 50, p = 0.1

For n = 1000, m=50, p=0.5

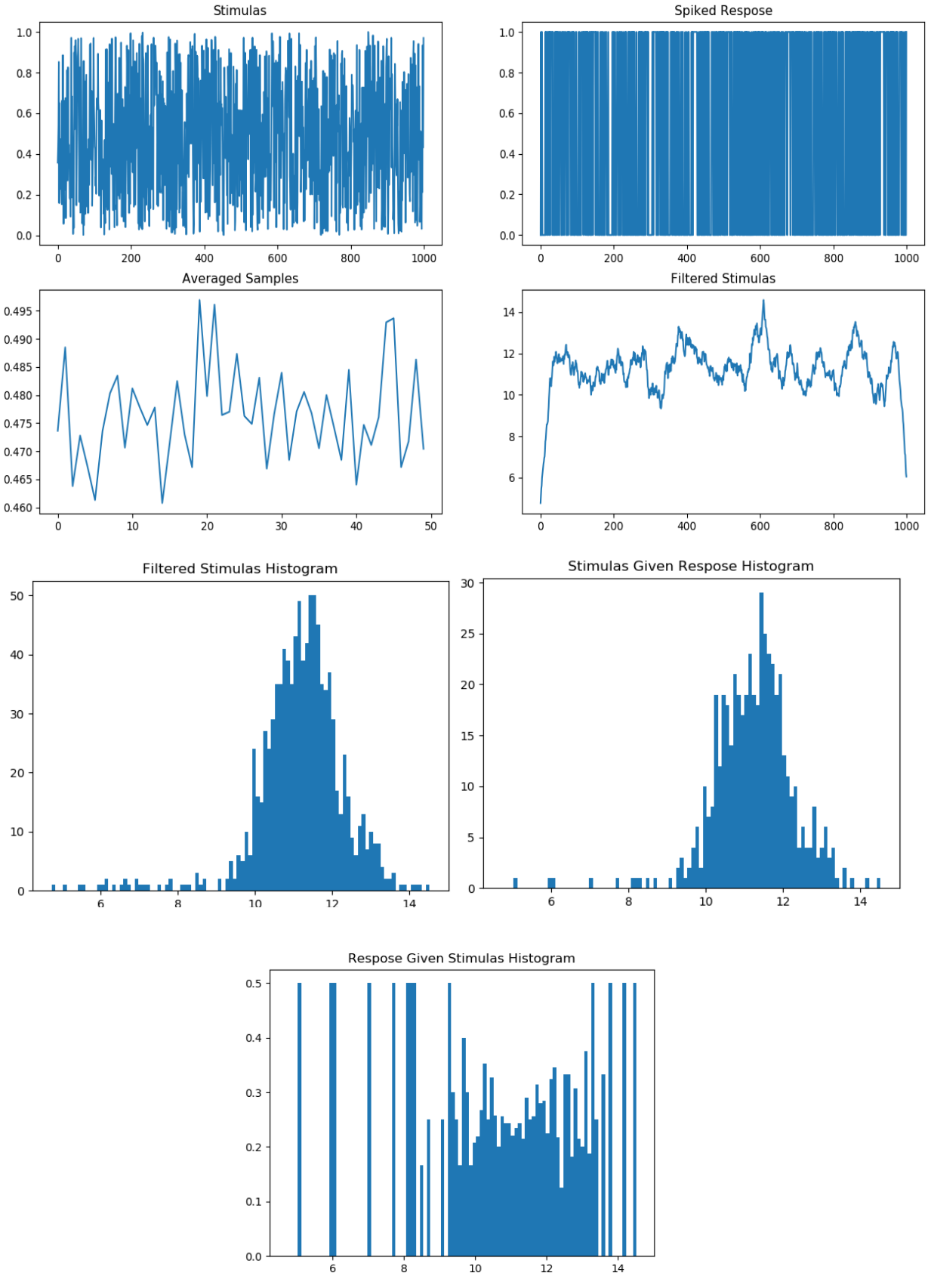

Figure 5 n = 1000, m = 50, p = 0.5

Discussion:

- PCA can be used for multiple filters instead of using average of all samples, we can select suitable component, so different components can act like different filters