CNEURO 2022
Neural networks: HW 1

Kameron Decker Harris

July 20, 2022

0. (Setup) I assume some familiarity with python programming. If you're new to python, please pair up with a friend who is more familiar! Some nice resources:

- `https://scipy-lectures.org/intro/`
- `https://pythonnumericalmethods.berkeley.edu`

For this and our future assignment, you'll need a working installation of python3. My recommendation is to install the Anaconda distribution on your laptop or lab machine. After setting this up, install the following packages: brian2, scikit-learn, and pytorch. All of these packages have very well-written documentation with lots of example code you can learn from.

Complete the exercises you will find useful. It's up to you how you want to organize and implement your code. Sometimes jupyter notebooks are nice, but beware they can lead to strange bugs due to running cells out of order.

1. (Input-output curves for various neuron models) See the example code that produces an F-I curve (firing rate versus input) for a leaky integrate-and-fire (LIF) model `https://brian2.readthedocs.io/en/stable/examples/IF_curve_LIF.html`. This implements the model $\tau\dot{V} = V_0 - V$ with spikes fired when $V > V_{\text{thresh}}$. Now consider a *quadratic* integrate-and-fire (QIF) model, where $\tau\dot{V} = V_0 + V^2$. Here, V_0 is a steady input voltage.

1. Draw a phase portrait of $\tau\dot{V}$ versus V and explain how the input V_0 determines whether the model will spike for a threshold V_{thresh}. Compare and contrast LIF and QIF models.
2. Implement the QIF model in your code and produce the F-I curve.
3. Consider the ReLU and sigmoid nonlinearities, $\max(x, 0)$ and $1/(1 + e^{-x})$ respectively. Up to shift and scaling, what is a better match for the the F-I curves?
4. **(Harder)** Solve the QIF differential equation via separation of variables and use the solution to get an expression for the F-I curve. Plot your formula on top of the simulation data. For simplicity with the math, you can rescale time and set $\tau = 1$. (Why?)
5. **(Harder)** The Hodgkin-Huxley model is also implemented in brian as `https://brian2.readthedocs.io/en/stable/examples/IF_curve_Hodgkin_Huxley.html`. Implement a similar model known as Connor-Stevens, which incorporates an A-type potassium current, and experiment with changing the conductance of that current and how the F-I curve changes. Here is a reference pdf and xpp code for the Connor-Stevens model.

2. (Artificial neural networks) Generate a simple 2-D classification dataset with the following code

```
from sklearn.datasets import make_circles
from sklearn.model_selection import train_test_split

X, y = make_circles(n_samples=400, factor=0.3, noise=0.05, random_state=0)
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify=y, random_state=0)
```

Build and train a simple neural network with a single, fully-connected hidden layer of ReLU units and a linear readout under the logistic loss (a.k.a. cross-entropy loss) on this problem. Use 5 hidden units and plot the data along with the decision region. If the network doesn't do well, restart the optimization with new random parameters until you find a good solution.

When done, repeat with 1000 hidden units. Comment on the differences.

An example of how to build and optimize a network for a more complicated dataset is this torch example.