1. Introduction

Can machine learning and neural networks help us model the eye’s retina accurately? We explore 2 models: a Linear Nonlinear Poisson (LNP) model and a CNN to predict the neuronal response (spike trains) of 2 types of Retinal Ganglion cells (RGC) – ON & OFF RGCs, given visual stimuli.

2. Problem Formulation

For an RGC, given m training examples \( \{x^{(i)}, y^{(i)}\}, i \in \{1, 2, \ldots, m\} \)
- **Stimulus movie:** \( x^{(i)} \rightarrow \) frame size \( L \times B \), duration \( D = N \) frames at refresh rate \( r \)
- **Observed spike train:** \( y^{(i)} = 1\{t_{observed} \text{ a spk} \}, j \in \{1, 2, \ldots, D/\text{res} \} \)
- **Predict:** \( \hat{y} \) for a test \( x_{\text{test}} \), to minimize \( \text{err}(y_{\text{test}} - \hat{y}) \) for a reasonable error metric "err".

3. Data

- **Unique** multi-electrode recordings from large ensembles of RGCs in macaque monkey retina, visually stimulated with movies of white noise, shared generously by the Chichilnisky lab at Stanford.
- **Gaussian** binary white noise stimuli := several 30s movies of (80 x 40) frame size with 120 Hz resolution.
- For every RGC exposed to this, there is an output vector containing the cell spike times.
- A test movie stimulus is shown several times to obtain repeats of the cell responses. These are used to determine accuracy of prediction.

4. Linear Nonlinear Poisson Model

Consists of a linear block, followed by a non-linear estimate of the firing rate, subjected to a Poisson process for spike generation.1

5. Linear Block – STA

The Spike triggered average (STA) vector is given by

\[ \omega = \left( \sum_{D} s_{i} f_{i} \right) / \sum_{D} f_{i} \]

- \( s_{i} \rightarrow \) fixed fraction of the stimulus preceding time bin \( t \)
- \( f_{i} \rightarrow \) the number of spikes observed in time bin \( t \)
- Spatial structure of \( \omega \rightarrow \) neuron's receptive field that isolates region of visual space that the neuron is active to.
- Temporal structure of \( \omega \rightarrow \) impulse response of the underlying linear summation

6. Nonlinear Block

Expected rate to the stimulus, is given by

\[ R(s) = E[f|s] = \sum_{f} P(f|s) \]

- **Simplifying assumption:** \( R \) is a static non-linear function i.e. \( R(s) = N(g) \) where \( g = \omega \cdot s \) is the generator signal.
- **Estimation of \( N \):**
  - averaging \( f_{i} \) across time bins that give rise to the same generator signal \( g_{t} \)
  - By plotting \( f \) vs \( \hat{g} \) for different values, the nonlinear function \( N \) is estimated as a sigmoid.

7. Results

- **Spiking Probability**
- **Expected Correlation Coefficient** for training data 0.14% and for Test data 8.87%
- **Spiking Probability**
- **Correlation Coefficient (per layer)**
- **Spiking Probability**
- **Correlation Coefficient**
- **Spiking Probability**
- **Spiking Probability**


- **Input:** 36k (6min) training examples, \( 80 \times 40 \times 10 \) timesteps.
- **Output:** Single number that represents spiking probability, trained against Gaussian-convolved spike train (ground truth).

9. Deep Learning Results

10. Discussion and Future Work

- **The simplistic LNP model shows better test results compared to the CNN when tuned for minimum training error.**
- **CNN model needs to be tuned further to minimize test error.**
- **Possible architecture improvements:**
  1. Add 3D convolution, including an LSTM layer in the end
  2. Explore RNNs further.
  3. Use a different RGC; literature shows2 different RGCs give different \( p \) when trained on the same CNN model. It could be that we've used a neuron that has an intrinsically low \( p \).