Neuromorphic Acceleration for Permanent Dropout

Presenter: Nathan Wycoff (Virginia Tech)
Co-authors: Prasanna Balaprakash and Fangfang Xia (ANL)

International Conference on Neuromorphic Systems

24 July 2019

Table of Contents

Table of Contents

Motivation: CANDLE Combo

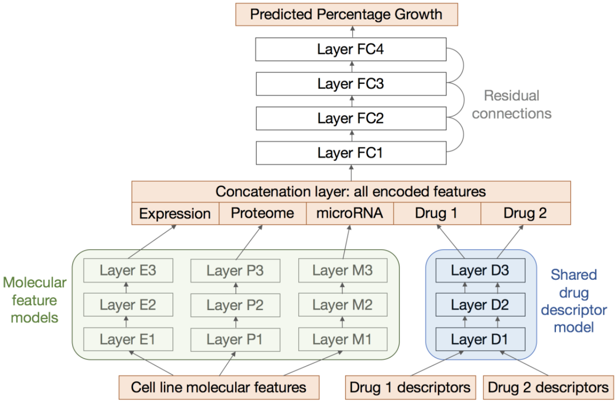


Figure: Four component neural network to predict cell line response to combination therapy.

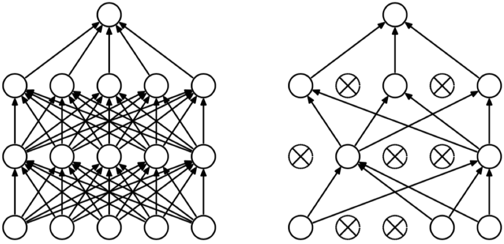
Motivation: CANDLE Combo (2)

```
$ python infer.py -s GDSC -d NCI_IOA_AOA --ns 10 --nd 5 -m saved.uq.model.h5 -w saved.uq.weights.h5 -n 100

$ cat comb_pred_GDSC_NCI_IOA_AOA.tsv
Sample  Drug1       Drug2   N           PredGrowthMean  PredGrowthStd   PredGrowthMin   PredGrowthMax
GDSC.22RV1      NSC.102816      NSC.102816      100     0.1688  0.0899  -0.0762 0.3912
GDSC.22RV1      NSC.102816      NSC.105014      100     0.3189  0.0920  0.0914  0.5550
GDSC.22RV1      NSC.102816      NSC.109724      100     0.6514  0.0894  0.4739  0.9055
GDSC.22RV1      NSC.102816      NSC.118218      100     0.5682  0.1164  0.2273  0.8891
GDSC.22RV1      NSC.102816      NSC.122758      100     0.3787  0.0833  0.1779  0.5768
GDSC.22RV1      NSC.105014      NSC.102816      100     0.1627  0.1060  -0.0531 0.5077
...
```

Figure: Summary statistics of predictive distribution.

Table of Contents

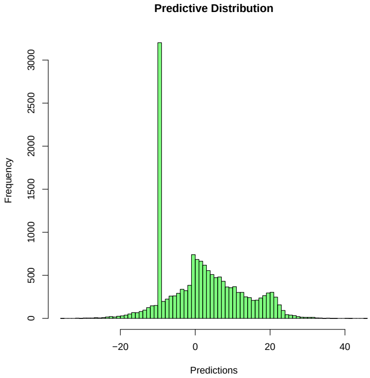
Dropout for Regularization



(a) Standard Neural Net (b) After applying dropout.

Figure: Dropout Stochastically removes network nodes. Image Credit: Srivastava, Nitish, Hinton, Geoffrey, Krizhevsky, Alex,Sutskever, Ilya, and Salakhutdinov, Ruslan. Dropout: A simple way to prevent neural networks from overfitting.J. Mach. Learn. Res., 15(1):19291958, January 2014

Dropout for UQ



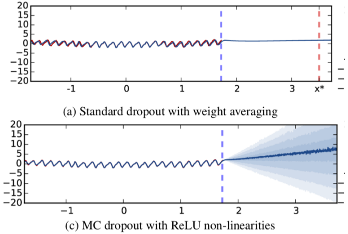


Figure: Summary statistics of predictive distribution. Right Image Credit: Gal and Ghahramani, *Dropout as a Bayesian approximation: Representing model uncertainty in deep learning*, in Proc. 33rd Int. Conf. Mach. Learn., 2016, pp. 10501059

Involves Many Model Evaluations!

Table of Contents

Linear LIF Dynamics under Constant Input

$$\frac{1}{\tau_{ref} + \tau_{RC} \log(1 + \frac{\nu}{\rho(\lambda - \nu)})} \tag{1}$$

$$\rho(x) = \max[0, x] \tag{2}$$

$$\rho(x) \approx \sigma(x) := \gamma \log(1 + e^{\frac{x}{\gamma}}) \tag{3}$$

The SoftLIF Activation Function

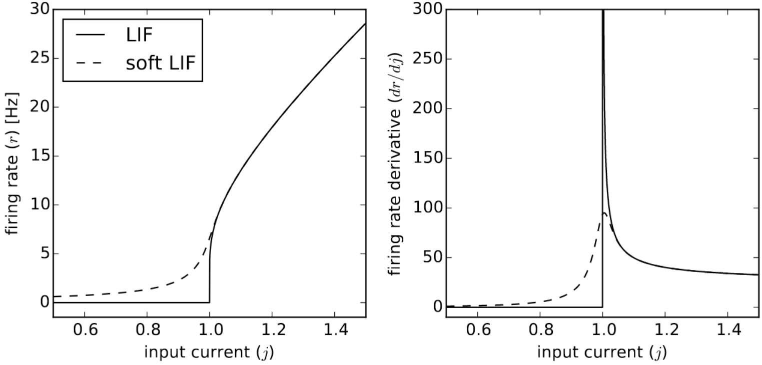
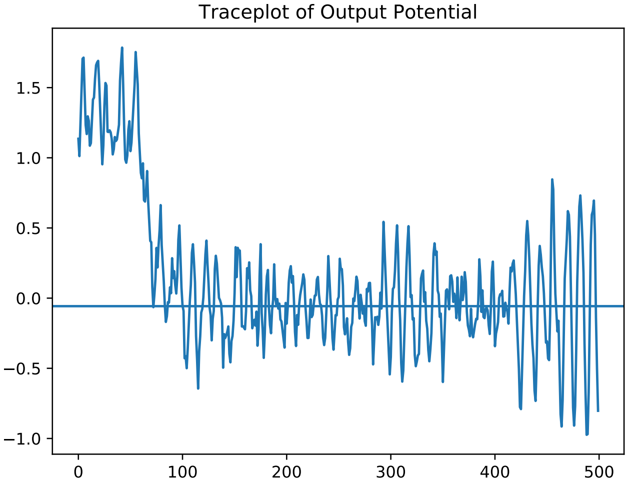


Figure: SoftLIF Function and Derivative Image Credit: Eric Hunsberger and Chris Eliasmith. 2015. Spiking Deep Networks with LIFNeurons.CoRRabs/1510.08829 (2015).

Just train a DNN with SoftLIF then transfer parameters!

Mean of SNN agrees with DNN



Nengo Extras

```python
# compile and fit Keras model
optimizer = keras.optimizers.Nadam()
kmodel.compile(loss='categorical_crossentropy',
               optimizer=optimizer,
               metrics=['accuracy'])
kmodel.fit(X_train, Y_train, batch_size=batch_size, epochs=epochs,
           verbose=1, validation_data=(X_test, Y_test))
```

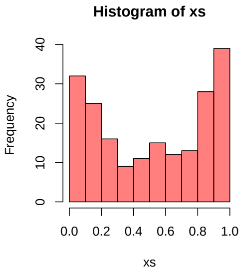
```python
# --- Run model in Nengo
with nengo.Network() as model:
    u = nengo.Node(nengo.processes.PresentInput(X_test, presentation_time))
    knet = SequentialNetwork(kmodel, synapse=nengo.synapses.Alpha(0.005))
    nengo.Connection(u, knet.input, synapse=None)

    input_p = nengo.Probe(u)
    output_p = nengo.Probe(knet.output)
```

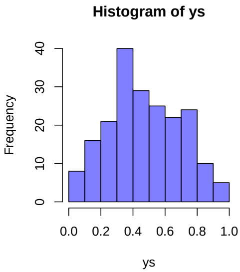
Aside: NengoDL

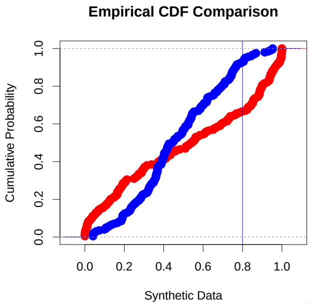
```
with nengo.Network() as net:
    <construct the model>

with nengo_dl.Simulator(net, ...) as sim:
    sim.train(<data>, <optimizer>, n_epochs=10, objective=<objective>)
```

Table of Contents

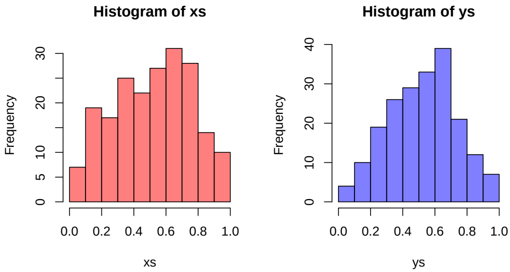
The Kolmogorov-Smirnov Statistic

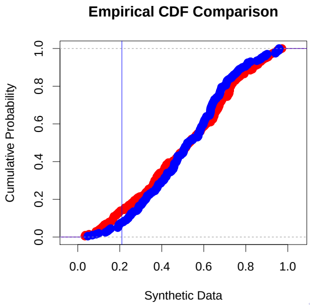






The Kolmogorov-Smirnov Statistic (2)





The Kolmogorov-Smirnov Statistic (3)

- P value: probability of getting a KS statistic as big as the one we did by chance.
- P values are uniform if distributions are the same:

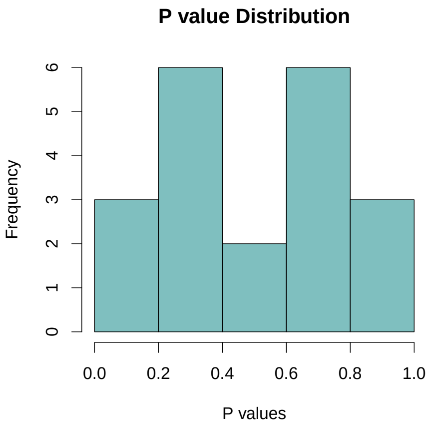
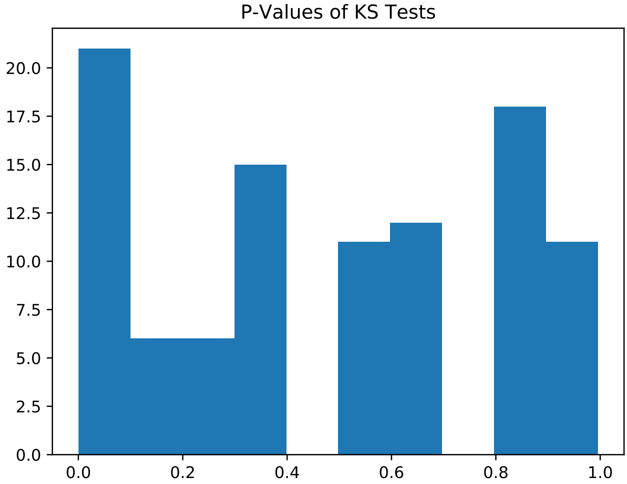


Table of Contents

Experimental Setup

1. Fit Combo with SoftLIF activation
2. Convert to SNN
3. On the first 100 observations of the Combo data, run the SNN as well as DNN with permadrop 100 times.

P-value Distribution Looks Roughly Uniform



Most extreme KS statistic looks Close

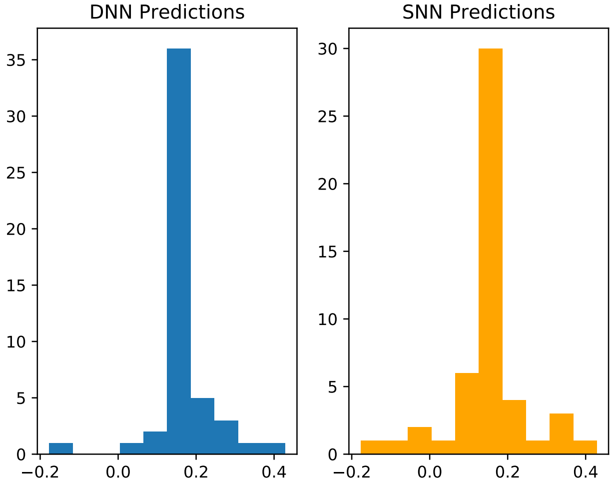
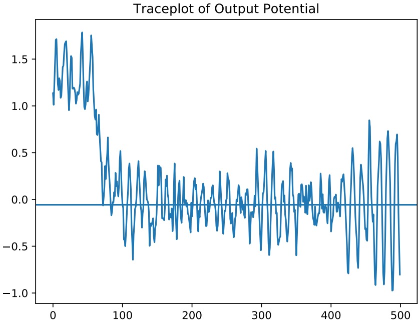


Table of Contents

Future Work

1. Run on Neuromorphic Hardware
2. Release Software
3. Get whole distribution from single run?



Contact: Nathan Wycoff (nathw95@vt.edu)

Work in progress: comments welcome.

Acknowledgements: N. Wycoff gratefully acknowledges funding from DOE LAB 17-1697 via asubaward from Argonne National Laboratory for SciDAC/DOE Office of Science ASCR and High Energy Physics. This material is based upon work supported by the U.S. Department of Energy(DOE), Office of Science, Office of Advanced Scientific Computing Research, under Contract DE-AC02-06CH11357.