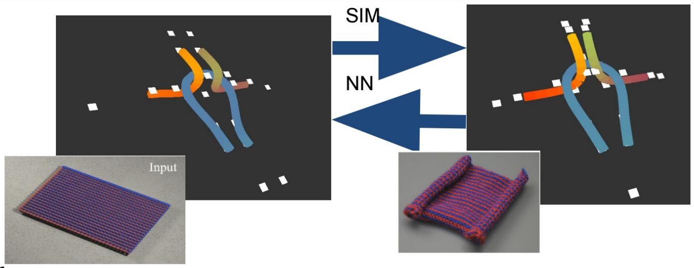
Inverting Yarn-Level Cloth Relaxation

Jonathan Leaf
jcleaf@stanford.edu
Department of Electrical Engineering, Stanford University

Predicting

To calibrate physics-based yarn-level relaxation, I built a neural network to predict physics-based simulation physical parameters from a relaxed yarn configuration. Each input is a yarn configuration, a set of ordered spline control points that defines the yarn curves. Each output is a physical parameter that we are trying to predict.

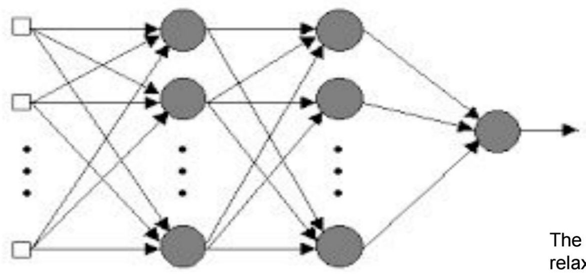


Data

The data came from a yarn-level cloth relaxation simulator based on [Kaldor 2008]. The columns of the dataset contain control point positions and physical parameters for each relaxation. The rows contain each example. Labelings are generated automatically. I scaled the data to have zero-mean and unit variance.

Models

I used a traditional neural network architecture with 2 hidden layers with RELU activation functions and a linear output layer. I used an AdamOptimizer on batches of 300 with a learning rate of 0.005.

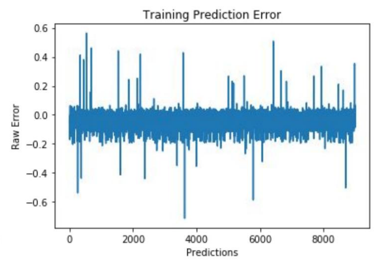


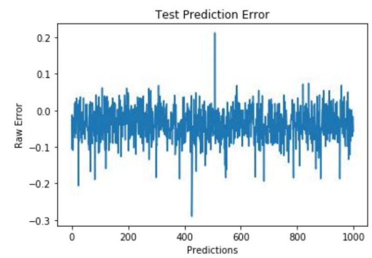
$$A^{[l]} = \sigma_{RELU}(W^{[l]}A^{[l-1]} + b^{[l]})$$

$$L = \sum (y^{(i)} - \hat{y}^{(i)})^2$$

Results

Train Set	Train Acc.	Test Set	Test Acc.
9000 examples	91.75%	1000 examples	93.6%





Discussion

The network has successfully learned the inverse relaxation of a few splines to physical parameter. The results seem to be reasonable, given an accuracy threshold of <0.1. There are many more network parameters than predicted physical parameters, so the function is feasible to learn.

Future

With more training examples and a deeper network, I could train it to learn more physical parameters with better accuracy. With measurement data I could try to predict the physical parameters of real yarn. A network would also need to be constructed for each pattern, so a library of networks would be required.