Correction of Tissue Sample Mislabeling in Genomic Data Using Neural Networks

Samuel Ivanecky
samuel.ivanecky@jacks.sdstate.edu
Advised by Dr. Xijin Ge
South Dakota State University Department of Mathematics and Statistics Spring 2019

Introduction

Genomics focuses on the study of genes and their functions with regard to the human genome. Mislabeling of tissue samples in genomic research produces invalid studies and results, leading to claims with no foundation.

- Neural networks provide machine learning algorithm based on biological processes.
- Goal of the project is to improve upon the accuracy for classification of tissue samples based on gene expression profiles.

Neural Networks

Neural networks are systems of interconnected neurons which are grouped into layers.
Basic structure of network consists of three layers:

1. **Input**: Neurons correspond to input variables
2. **Hidden**: Neuron layer(s) where processing occurs
3. **Output**: Neuron(s) correspond to predicted output value(s). Output neurons vary based on value versus classification problems.

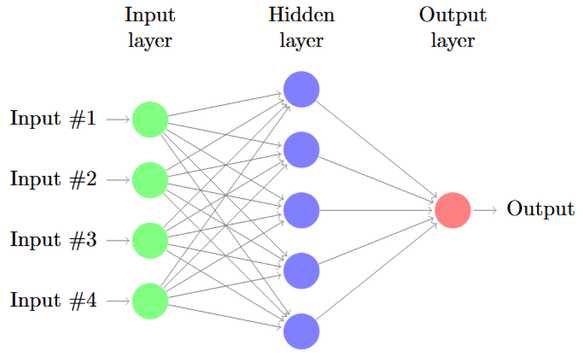

Figure 2: Simple neural network of four input variables and one predicted output

- Hidden neurons receive weighted sum or neurons in previous layer.
- Weights express importance of input variable.

Activation Functions

Activation functions are used to condense input values on a large range down to a smaller numeric range that is determined by the type of activation function selected.
The input to the *sigmoid* function is weighted sum of neurons in the previous layer.

$$z = \sum_j (w_j x_j - b) \tag{1}$$

Sigmoid Function:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{2}$$

The sigmoid function is continuous so small changes in the weights and biases results in small changes in the output of the function.

Gradient Descent Optimization

Gradient descent is primary method for optimizing cost function of neural networks.
Cost Function: This function computes the mean square error of the network. The goal is to minimize this function.

$$C(w,b) = \frac{1}{2n} \sum_x ||\mathbf{y(x)} - \mathbf{a}||^2. \tag{3}$$

The gradient is first calculated to find the minimum of C.

$$\nabla \mathbf{C} = \left(\frac{\partial C}{\partial w_1}, \frac{\partial C}{\partial w_2}, ..., \frac{\partial C}{\partial w_j}, \frac{\partial C}{\partial b_1}, \frac{\partial C}{\partial b_2}, ..., \frac{\partial C}{\partial b_k} \right)^T \tag{4}$$

The changes in weights and biases can be directly related to the gradient. The vector $\triangle \mathbf{v}$ is the changes to each weight and bias value.

$$\mathbf{v} = (w_1, w_2, ..., w_j, b_1, b_2, ..., b_k)^T \tag{5}$$

$$\triangle \mathbf{v} = (\triangle w_1, \triangle w_2, ..., \triangle w_j, \triangle b_1, \triangle b_2, ..., \triangle b_k)^T \tag{6}$$

The change in C can then be written as the rate of change of C multiplied by the changes made to the inputs (w and b) of C.

$$\triangle C = \nabla \mathbf{C} \cdot \triangle \mathbf{v} \tag{7}$$

The change in the cost function can then be rewritten as,

$$\triangle C = -\eta \nabla \mathbf{C} \cdot \triangle \mathbf{v}, \tag{8}$$

Data Analysis & Results

Initial data was accessed from the GTEx Portal.

- 12,000 tissue samples with a possible 56,000 expressed genes in each sample.
- Each sample came from one of 53 distinct tissues from around the human body.
- Subset of 3,000 tissue samples were used in the model development phase.
- Data was split into 70% training and 30% testing sets.

Model Development Process: Model development process involves running the entire training data set through the network in small batches.

1. A batch of data is run and the accuracy for the batch is calculated.
2. The weights and biases are updated using gradient descent to increase the accuracy.
3. A new batch of data is run through the network and the process repeats.
4. Once the entire training set has been used, the data is randomized and the overall process repeats.

Model Results:

- Eight different model configurations were tested.
- *Python* programming language was used to develop models.

Model	Hidden Neurons	Activation	Prediction Accuracy
Full	100	sigmoid	92.80%
Full	100	tanh	93.11%
Full	250	sigmoid	87.74%
Full	250	tanh	93.26%
Full	500	sigmoid	87.14%
Full	500	tanh	93.64%
Full	750	sigmoid	91.73%
Full	750	tanh	93.19%
Overall			91.58%

Figure 3: Results of ten different model configurations tested.

Results (cont.)

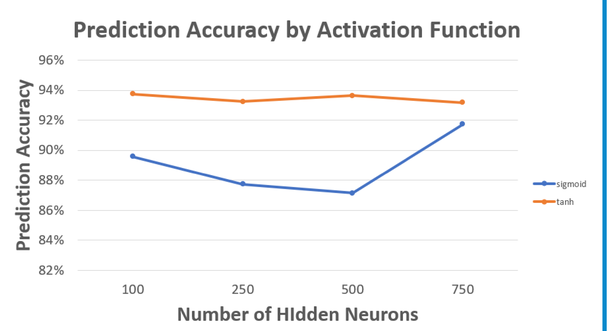

Figure 1: Model results for tanh versus sigmoid configurations using varying levels of hidden neurons.

- Increasing neurons had a greater impact on sigmoid than *tanh*.
- Average accuracy of eight configurations was 91.58%.
- *tanh* outperformed the sigmoid function by a margin of 4.36% in prediction accuracy.

Downfalls of Neural Networks

- Higher complexity of data frames greatly increases the required computational power needed to build models.
- Neural networks require a substantial amount of data to train and test models which limits application scenarios.
- Networks use a black-box approach for the computations which can make understanding the actual changes to data impossible.
- The model development phase can become greatly limited by the available computational power.

Conclusions

- Overall average error rate of 8.66% was substantial improvement from the error rate found in external studies.
- Future improvements could be made by utilizing more data with an increase in computational power.