ARTIFICIAL NEURAL NETWORKS FOR CLASSIFICATION OF HIV INFECTION RATES

Jennifer Lynne Reese, Christo Vijay, Vishnu Vardhan Vankadara,
Mahesh Koppala and Shiva Prasad Reddy Katta
Department of Computer Science, University of Texas at Arlington
Texas, USA
Email: {jennifer.reese, christo.vijay, vishnuvardhan.vankadara, mahesh.koppala, shivaprasadredd.katta}@mavs.uta.edu

GROUP 5

***Abstract*—Epidemics are unpredictable powers of nature. However, we are able to predict outbreaks with increasing accuracy, with the use of massive advancements in technology, including neural networks. While many epidemics are deadly and can cause irreversible damages, HIV, in particular, is one of the most damaging epidemics to affect mankind. This project aims to predict the rate of change in HIV infections in the future based on historical data. We are using an Artificial Neural Network(ANN) for this project, and the Scaled Exponential Linear Unit (SELU) function serves as the activation function. Upon training the model, our model was able to correctly classify the rate of change of the HIV epidemic with about seventy percent accuracy. To be able to accomplish this, we trained the model with 65 percent of the data and tested it on the remaining 35 percent.**

I. INTRODUCTION

HIV/AIDS is a serious health issue the world is facing now. While the impact of the disease has considerably decreased over the years, the number of deaths per annum is still significant. As per the World Health Organization, in 2018 alone, 0.77 million people died due to HIV. Currently, approximately 38 million people around the globe are infected with HIV. ART, short for antiretroviral therapy, is the treatment given to HIV positive patients to prolong their life span, and also prevent them from spreading the disease to their unaffected partners. Furthermore, health institutions around the world are trying to prevent pregnant women from passing the disease to newborn babies. This project aims at predicting the HIV infection rates in the future with the help of vast amounts of past HIV patients data. We will attempt to use historical data on HIV infection rates and administration of antiretroviral (ART) treatment to HIV patients including HIV-positive pregnant women, and HIV-positive children as input to an artificial neural network (ANN) to successfully classify the rate of change in HIV infections across the globe. Ideally, this type of model would be useful to predict the rate of change in HIV infections in the future.

II. PROBLEM

In this section, we included the whole process which includes data collection, cleaning, normalization, and also building the model.

A. Data Selection and Cleaning

We selected multiple data sets from the World Health Organization ("GHO | By Category | HIV/AIDS"). During the cleaning process we removed countries for which complete information was unavailable. We also ensured the data was collected across a consistent period of time. All data was from 2018, with the exception of the percent increase in HIV cases. This increase covered the years from 2000 to 2018. This cleaning process ensured our data was consistent across the different data sets. We merged the data into a comma-separated file in the following format:

- Each row of the file represents data for a different country
- Column 1: Number of patients receiving ART in 2018
- Column 2: Percentage of HIV-positive children receiving ART in 2018
- Column 3: Percentage of HIV-positive pregnant women receiving ART in 2018
- Column 4: Percent increase in HIV cases from 2000 to 2018
- Column 5: Classifier

We created a classification system with three classification levels. A classification of 0 represented an increase in HIV cases of less than 0 percent (decline in HIV cases). A classification of 1 represented a growth rate between 0 and 100 percent (moderate increase in HIV cases). A classification of 2 represented a growth rate of greater than 100 percent (significant increase in HIV cases). Although a more sensitive classification system would be preferable (more classification levels would allow for more nuance in analysis), our model proved to be far more accurate with fewer classifications.

B. *Normalizing and Splitting Data*

We built our ANN using Java and APIs from the Deeplearning4j libraries (Eclipse Deeplearning4j Development Team). The first step was to use the CSVRecordReader and the DataSetIterator from the DataVec library to perform vectorization. We then normalized our data using the NormalizerStandardize class. This automatically gathered statistics about our data and transformed it into a uniform dataset. The ANN requires normalized data to perform optimally. Finally, we split our data set into training and testing portions. We found that using 65 percent of the data for training provided the best results with the remaining 35 percent test data.

C. *Building Our Model*

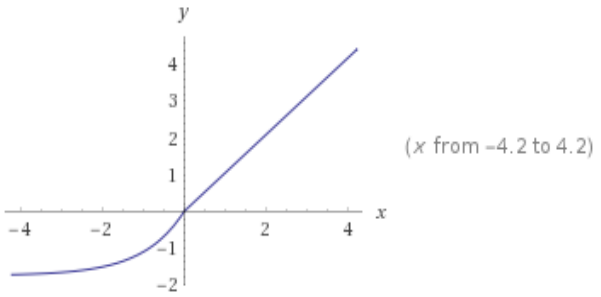
We used a fluent-style model to create a multi-layer perceptron network. Because our data set was relatively small, we configured our model to perform 1,000 iterations on the data before we evaluated the results. If we wanted to extend this project and were able to acquire significantly more data, we could run fewer iterations because we would have more data with which to train the ANN. The limitations of our data set required more iterations.

ANNs require an activation function and we chose the Scaled Exponential Linear Unit (SELU) function. This function performed significantly better than other functions, including a simple tanh function and the Rectified Linear Unit (RELU) function. This is because the SELU function maintains the mean and variance from each previous layer in the network, a concept known as internal normalization (Böhm).

The SELU function is as follows (Böhm):

$$f(\alpha, x) = \lambda \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$$

The graph of the SELU function generally follows this format (Böhm):



We set our learning rate to 0.1. This value allowed us to prevent overfitting of the training data, which can occur when the learning rate is set too high. We also used the regularization method to prevent the network from overweighting the training data. Our network consisted of three layers. The first layer, the input layer, contained one node for every column in our training data. This layer had three output nodes, which corresponded to three input nodes in our middle layer. Our output layer had three output nodes, corresponding to the three classifications for our data.

Finally, we initialized our model and trained the data. We used the dl4j Evaluation class to display information about the success of our classification model. The results varied slightly with each execution of the model, but the model usually classified the data with approximately 70 percent accuracy. Below is sample output from one execution:

```
Examples labeled as 0 classified by
model as 0: 2 times
Examples labeled as 0 classified by
model as 1: 19 times
Examples labeled as 1 classified by
model as 0: 3 times
Examples labeled as 1 classified by
model as 1: 28 times
Examples labeled as 1 classified by
model as 2: 5 times
Examples labeled as 2 classified by
model as 1: 1 times
Examples labeled as 2 classified by
model as 2: 38 times

==========================Scores==
 # of classes:    3
 Accuracy:        0.7083
 Precision:       0.6224
 Recall:          0.6158
 F1 Score:        0.5824
Precision, recall & F1: macro-
averaged (equally weighted avg. of
3 classes)
================================
```

III. CONCLUSION

Our model is a simple experiment with using ANNs to classify data. While correctly classifying data with 70 percent accuracy is satisfactory for some applications, it would not be particularly useful for large-scale epidemic prediction or to inform the decision-making process for handling an existing HIV outbreak. To improve this model we need significantly more data. An interesting extension would be to find older data (prior to 2000) and build a model using this data. This model could then attempt to classify the data we used in this experiment. We would know the actual results and could better evaluate the efficacy of our model in this way.

Additionally, we could add more layers to the model if we had more data. This could provide more nuance and accuracy to the classifications, which would allow us to classify the rate of change into more than three categories. The applications of these predictions would vary wildly depending on whether a country had an increase of HIV cases of 10 percent or 100 percent. In our current model, these two increases fall into the same category, which is not ideal.

REFERENCES

[1] Baeldung. A guide to Deeplearning4j, 15 Aug, 2019, https://www.baeldung.com/deeplearning4j

[2] Bohm, Timo. An Introduction to SELUs and why you should start using them as your activation functions, 28 Aug, 2018, https://towardsdatascience.com/gentle-introduction-to-selus-b19943068cd9

[3] GHO | By category | HIV/AIDS. World Health Organization. http://apps.who.int/gho/data/node.main.617?lang=en.

[4] Eclipse Deeplearning4j Development Team. Deeplearning4j: Open-source distributed deep learning for the JVM, Apache Software Foundation License 2.0. http://deeplearning4j.org

[5] Artificial Neural Network role in modeling of HIV Epidemic. International Journal of Computer Applications, Volume 44-No 16 April 2012.