Name: Jason Zhou
Mentor: Dr. Dongjin Song
Status Report #: 3
Time Spent on Research This Week: 3.5
Cumulative Time Spent on Research: 13.5
Miles Traveled to/from Mentor This Week: 0
Cumulative Miles Traveled to/from Mentor: 0
==========

Monday, September 20th, 2021: (0.5 hours)

Similar to last week, I continued watching the seven-hour video course on TensorFlow. At this part of the video, I learned how to set up a linear regression model for machine learning. First, one must define the input and output columns, which can be done by creating an empty array and running it through a for-loop to store all of the data. From there, the model is trained and then evaluated based on its accuracy of predictions. Overall, linear regression is an easy-to-understand concept, so it is quick to learn.

Tuesday, September 21st, 2021: (1 hour)

Moving on from linear regression, I began practicing classification, another type of machine learning algorithm. Unlike the previous machine learning algorithm which predicted numerical values, classification takes in different inputs to classify something into a category (species, color, gender, etc). The process of implementing this algorithm was straightforward. One must first define features so that the data can be predicted. It is important to separate the data so that one portion of the data can be used for training and the rest of it can be used for evaluation. After doing that, the model can be trained with the training data and then evaluated based on the testing data. Once this has been done, the model should be able to predict or classify data based on previous trends and patterns that have been noticed.

Wednesday, September 22nd, 2021: (1 hour)

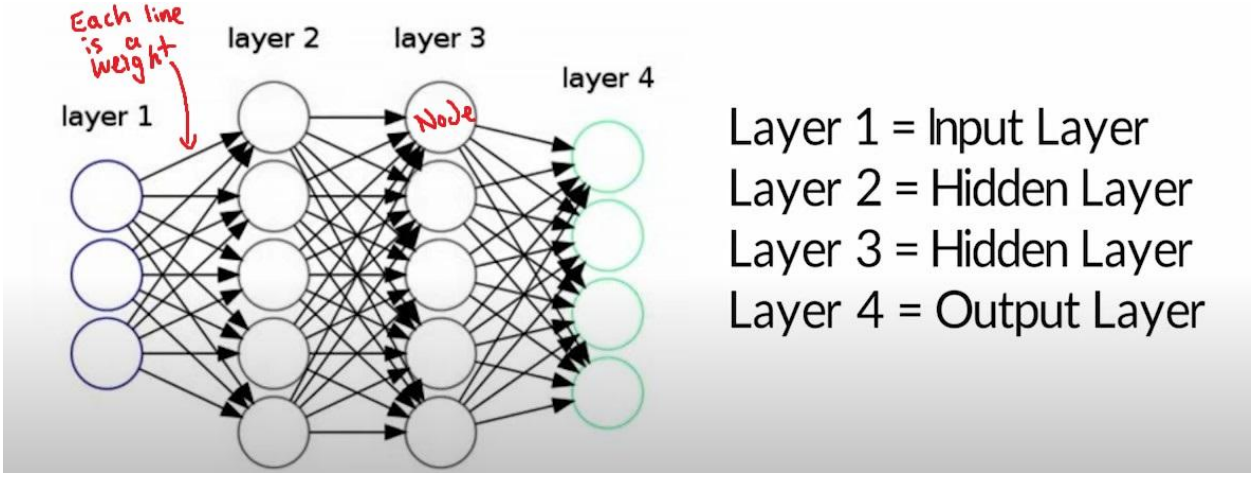
After classification, the video discusses Hidden Markov Models, which is a machine learning algorithm that operates based on possibility distribution and previous trends. Each Hidden Markov Model has states, observations, and transitions (all of which are based on chance). States is the condition or circumstances of a situation at a particular time. To illustrate, if I use the weather as an example, cloudy, sunny, windy, or snowy could all be seen as states because they are conditions of a situation (the weather). Next, transitions are the chances that the model will switch from one state to another. If it is sunny, for instance, then I could say that 40% for the weather to transition to being rainy the next day or 60% that it will continue to be sunny. Finally, observations are simply the reaction or consequence that happens as a result of a particular state. For example, if it was sunny, an observation would be that Tim is sweating due to the hot weather.

Because this type of model is apparently not very common, the video did not go too in-depth on the topic. So as of right now, I may have to do further research later on if I need to.

Thursday, September 23rd, 2021: (1 hour)

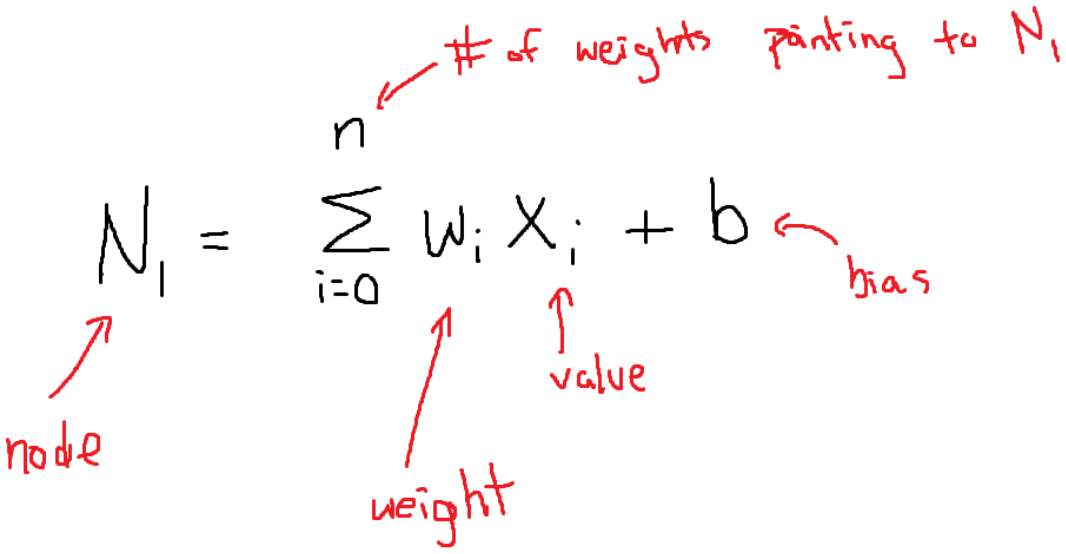
After learning about 3 different machine learning algorithms (linear regression, classification, and Hidden Markov Models), I finally had the chance to learn about neural networks, which is probably the most well-known model in machine learning. As I did not have a lot of time to watch the entire section about neural networks in the video, I was unable to gain an in-depth understanding. However, the basic information I learned was still fundamental for understanding neural networks.

Essentially, there are three main parts to a neural network: the input, output, and hidden layers. The input layer is where all the inputs are initially placed. The output layer is where the outputs will appear after processing the data, and the hidden layers are any layers in between the input and output layers. Each layer is made up of nodes (circles) that each perform a task that transforms the input into the output in some way (does not have to be transformed in one layer but can happen over the course of many layers). Connecting each layer to one another are lines called weights (Each node has multiple lines that each connect to another node in the next layer). These weights have a numerical value and determine how important one feature is versus another feature. For example, if you are trying to identify whether the fruit was a nectarine or a peach, one might say that the taste is more important, or has more weight than the color when making a decision (because peaches and nectarines have similar colors but different flavors, making it easy to differentiate by taste). In any neural network, the weights always start off as random numbers (a higher number means a higher weight). As the neural network analyzes more data, it will fine-tune these values to give more accurate outputs. Additionally, each layer has a value called a bias. In simple terms, this value would be similar to the y-intercept of a graph. It is a number that helps the data reach the desired output.



(This is a picture of a neural network. I put this in a previous status report; however, I felt that it would still be useful to better illustrate what I am talking about)

The weights of the function are used to calculate a value that is contained in each node (this value will be used to transform an input into an output, which was mentioned above). Specifically, the equation for this node value is:



(The equation for finding the value of a node based on weights)

This equation takes the summation of all the node values in the previous layer and multiplies it with all of the weighted values pointing to N1 (the node value being calculated for). Afterward, the bias is added and the node value is outputted.

Friday, September 24th, 2021:

On this day, I had my weekly meeting with my mentor. The topic was about the high school research program that I had entered at UConn and other data packages(like TensorFlow) that I could look into once I had the time.

Although I will most likely be doing something related to sound for my research project, my mentor believed that it was still a good idea to explore other possible projects. As such, Professor Song informed me that there would be a zoom demo on autonomous vehicles that I could attend, which will be happening on the 27th.

He then recommended that I look into Librosa and Melspectrogram, which are data packages that are related to the processing and manipulation of sound data.

This concludes my research for the week, and I hope to learn more in the near future!

References

freeCodeCamp.org. (2020, March 3). *TensorFlow 2.0 complete course - python neural networks for beginners tutorial* [Video]. *Youtube*. https://www.youtube.com/watch?v=tPYj3fFJGjk&t=6284s