|  |  |  |  |
| --- | --- | --- | --- |
| **Input Layer** | **Hidden Layer** | **Output Layer** | **Reference** |
| Units = No. of features (88) | 2 layers x 50 units per layer | Units = No. of output classes (7) Softmax | Shaqra et al. ‘Recognizing Emotion from Speech Based on Age and Gender Using Hierarchical Models’ |
| Units = No. of features (frame-level features. i.e. shitload of features!) | 3 layers x 256 ReLU units per layer | Units = No. of output classes (5) Softmax | Han et al. ‘Speech Emotion Recognition Using Deep Neural Network and Extreme Learning Machine (Microsoft) |
| Units = No. of features (60) | 3 layers (1st layer = 256, 2nd layer = 512, 3rd layer = 64) | Units = No. of output classes (8) Softmax | Lee et al. ‘A Study on Speech Emotion Recognition Using a Deep Neural Network’ |
| Units = No. of features (2624) ReLU | 5 layers x 1024 ReLU | Units = No. of output classes (6) Softmax | Fayek et al. ‘Towards real-time Speech Emotion Recognition using deep neural networks’ |

Each neuron needs an activation function.

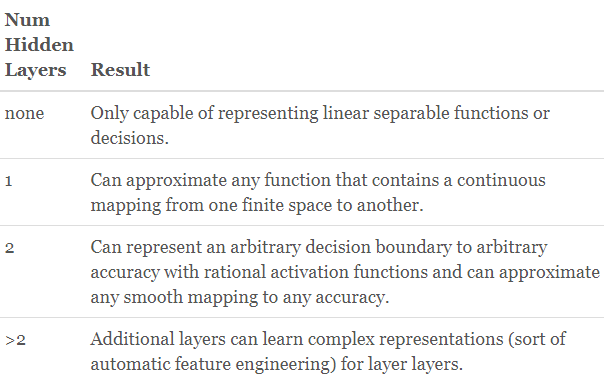
Use **ReLU** for input and hidden neurons. ReLU has advantage over sigmoid.

Use **Softmax** for output neurons to get probabilities.

For avoiding overfitting, try regularization such as **dropout**.

**Selecting the number of hidden layers in a feed-forward neural network**

A single hidden layer in a neural network is capable of universal approximation (i.e. a single hidden layer can learn any problem),  Cybenko (1989), Hornik (1991).



**Selecting the number of neurons in each hidden layer**

Using too few neurons in the hidden layers will result in something called underfitting. Underfitting occurs when there are too few neurons in the hidden layers to adequately detect the signals in a complicated data set.

Using too many neurons in the hidden layers can result in several problems. First, too many neurons in the hidden layers may result in overfitting. Overfitting occurs when the neural network has so much information processing capacity that the limited amount of information contained in the training set is not enough to train all of the neurons in the hidden layers. A second problem can occur even when the training data is sufficient. An inordinately large number of neurons in the hidden layers can increase the time it takes to train the network.

**A rule of thumb:**

I have a few rules of thumb that I use to choose hidden layers. There are many rule-of-thumb methods for determining an acceptable number of neurons to use in the hidden layers, such as the following:

* The number of hidden neurons should be between the size of the input layer and the size of the output layer.
* The number of hidden neurons should be 2/3 the size of the input layer, plus the size of the output layer.
* The number of hidden neurons should be less than twice the size of the input layer.

Source: <https://www.heatonresearch.com/2017/06/01/hidden-layers.html>

You can experiment with the number of hidden layers and the number of neurons in each layer, by fixing one parameter and varying the other and checking the accuracies, like Fayek et al. ‘Towards real-time Speech Emotion Recognition using deep neural networks’.