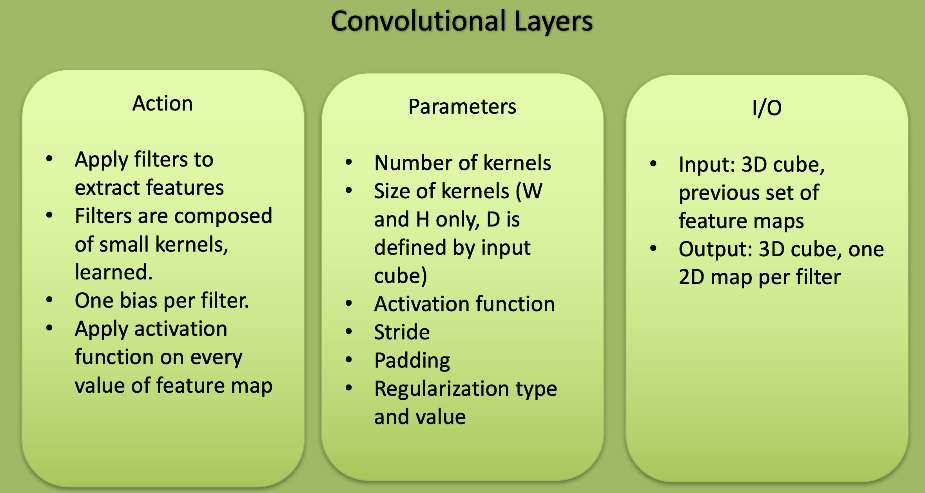
**Redes neuronales**

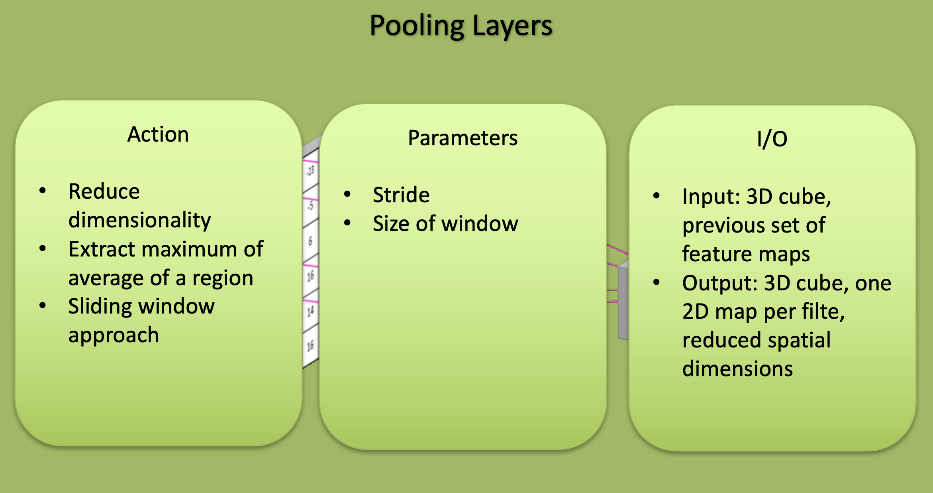
Investiguen la funcionalidad de las siguientes capas en una red neuronal:

1. Input: The input layer of a neural network is composed of artificial input neurons and brings the initial data into the system for further processing by subsequent layers of artificial neurons. The input layer is the very beginning of the workflow for the artificial neural network [1].
2. Convolutional: Convolutional layers are the layers where filters are applied to the original image, or to other feature maps in a deep CNN. This is where most of the user-specified parameters are in the network. The most important parameters are the number of kernels and the size of the kernels.



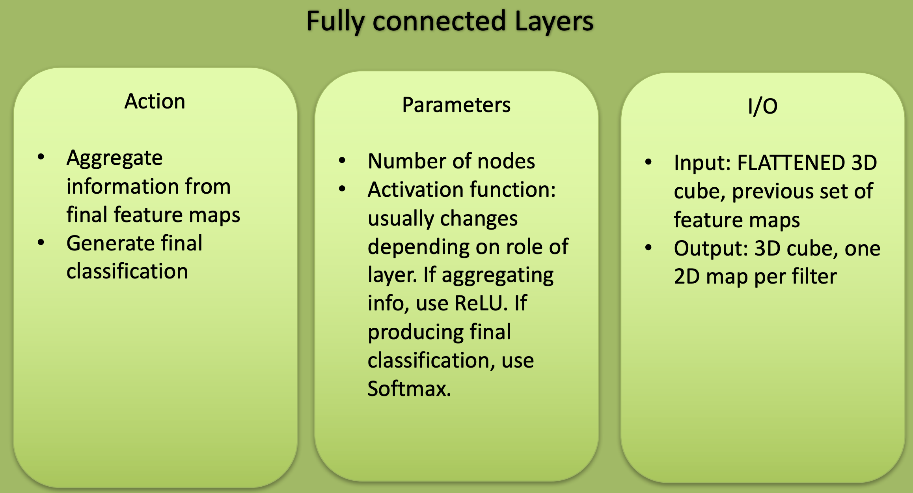
Obtained from [2]

1. Activation: An activation function is a node that you add to the output layer or between two layers of any neural network. It is also known as the transfer function. It is used to determine the output of neural network layer in between 0 to 1 or -1 to 1 etc. [3].
2. Pooling: Pooling layers are like convolutional layers, but they perform a specific function such as max pooling, which takes the maximum value in a certain filter region, or average pooling, which takes the average value in a filter region. These are typically used to reduce the dimensionality of the network.



Obtained from [2]

1. Fully connected: Fully connected layers are placed before the classification output of a CNN and are used to flatten the results before classification. This is similar to the output layer of an MLP.



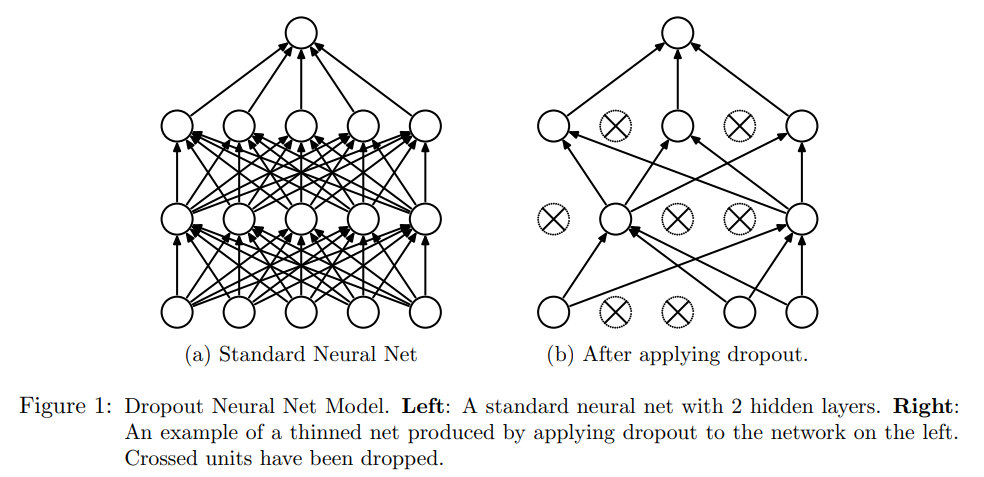
Obtained from [2]

1. Batch normalization: Batch normalization reduces the amount by what the hidden unit values shift around (covariance shift). Batch normalization allows each layer of a network to learn by itself a little bit more independently of other layers.

* We can use higher learning rates because batch normalization makes sure that there’s no activation that’s gone really high or really low. And by that, things that previously couldn’t get to train, it will start to train.
* It reduces overfitting because it has a slight regularization effects. Similar to dropout, it adds some noise to each hidden layer’s activations. Therefore, if we use batch normalization, we will use less dropout, which is a good thing because we are not going to lose a lot of information. However, we should not depend only on batch normalization for regularization; we should better use it together with dropout.

Obtained from [4]

1. Dropout: Dropout is a technique that addresses both these issues. It prevents overfitting and provides a way of approximately combining exponentially many different neural network architectures efficiently. The term “dropout” refers to dropping out units (hidden and visible) in a neural network. By dropping a unit out, we mean temporarily removing it from the network, along with all its incoming and outgoing connections [5].



Simply put, dropout refers to ignoring units (i.e. neurons) during the training phase of certain set of neurons which is chosen at random. By “ignoring”, I mean these units are not considered during a particular forward or backward pass [6].

1. Softmax: The softmax function is a function that turns a vector of K real values into a vector of K real values that sum to 1. The input values can be positive, negative, zero, or greater than one, but the softmax transforms them into values between 0 and 1, so that they can be interpreted as probabilities. If one of the inputs is small or negative, the softmax turns it into a small probability, and if an input is large, then it turns it into a large probability, but it will always remain between 0 and 1.

The softmax function is sometimes called the softargmax function, or multi-class logistic regression. This is because the softmax is a generalization of logistic regression that can be used for multi-class classification, and its formula is very similar to the sigmoid function which is used for logistic regression. The softmax function can be used in a classifier only when the classes are mutually exclusive.

Many multi-layer neural networks end in a penultimate layer which outputs real-valued scores that are not conveniently scaled and which may be difficult to work with. Here the softmax is very useful because it converts the scores to a normalized probability distribution, which can be displayed to a user or used as input to other systems. For this reason it is usual to append a softmax function as the final layer of the neural network.

Obtained from [7]

|  |  |
| --- | --- |
| [1] | «What is an Input Layer? - Definition from Techopedia,» [En línea]. Available: https://www.techopedia.com/definition/33262/input-layer-neural-networks. |
| [2] | «Simple Introduction to Convolutional Neural Networks | by Matthew Stewart, PhD Researcher | Towards Data Science,» [En línea]. Available: https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac. |
| [3] | «How to Select Activation Function for Deep Neural Network - engMRK,» [En línea]. Available: https://engmrk.com/activation-function-for-dnn/. |
| [4] | «Batch normalization in Neural Networks | by F D | Towards Data Science,» [En línea]. Available: https://towardsdatascience.com/batch-normalization-in-neural-networks-1ac91516821c. |
| [5] | N. Srivastava, G. Hinton, A. Krizhevsky y R. Salakhutdinov, «Dropout: A Simple Way to Prevent Neural Networks from Overfitting,» 2014. |
| [6] | «Dropout in (Deep) Machine learning | by Amar Budhiraja | Medium,» [En línea]. Available: https://medium.com/@amarbudhiraja/https-medium-com-amarbudhiraja-learning-less-to-learn-better-dropout-in-deep-machine-learning-74334da4bfc5. |
| [7] | «Softmax Function Definition | DeepAI,» [En línea]. Available: https://deepai.org/machine-learning-glossary-and-terms/softmax-layer. |