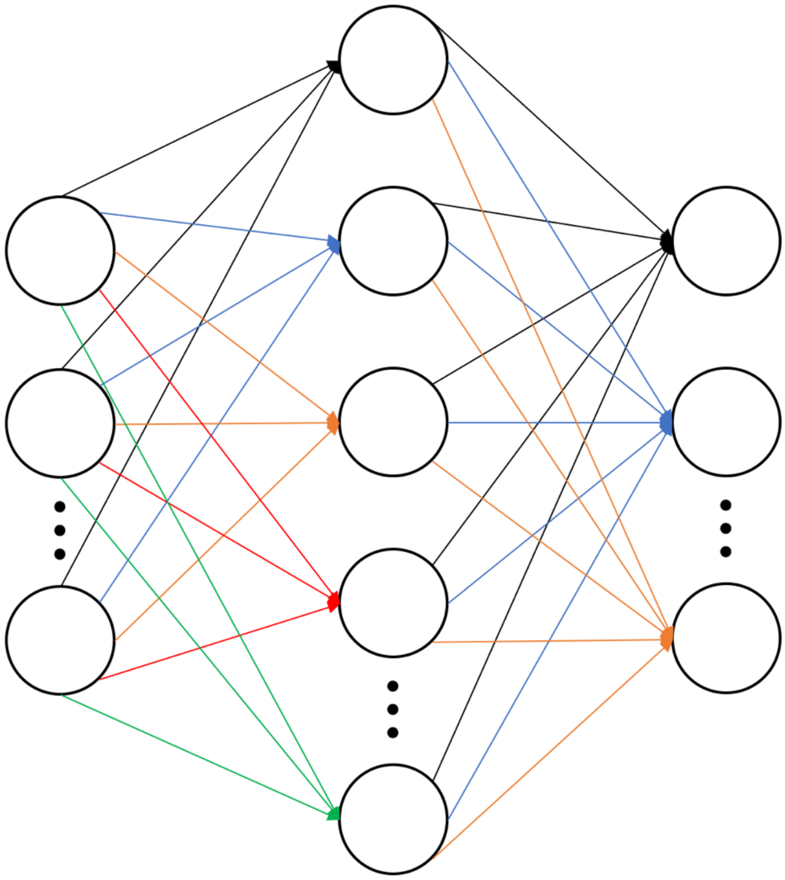
‘Practice Makes Perfect…’:

Artificial Neural Networks & Their Uses Within Physics

Vedant Varshney



Word Count: 2393

'Practice Makes Perfect…':
Artificial Neural Networks & Their Uses Within Physics

It is very difficult to predict – especially the future.
Niels Bohr

Few topic areas within the field of computer science have captured the imagination of the wider public like artificial intelligence (AI). Indeed, such is the topic's prevalence in popular culture that its mere mention within a conversation will almost undoubtedly spark quips of a dystopian future where us humans are ruled by some robotic overlord. In recent years, we have also seen an almost unparalleled adoption of such technologies within industry, with everything from Apple's voice-assistant 'Siri' to Samsung's smart fridge 'Family Hub' making use of AI [1].

As a result, the power and effectiveness of artificial intelligence is more or less unquestioned. However, an understanding of exactly how such technologies are built or their applications outside of consumer products is far less commonplace. This article aims to begin to bridge this gap by explaining the core concepts behind one of the most popular implementations of AI; artificial neural networks (or ANNs). It will also show some of its use cases within Physics research, so as to highlight how this technology may one day become the key instrument through which we understand the world around us.

Working Principles of ANNs

Artificial intelligence is a somewhat vague term which often means different things to different people, partly due to the philosophical issue as to what defines intelligence to begin with. Here we adopt the popular definition that an AI is any system capable of 'thinking' and 'learning' in a way which is analogous to humans [2]. With this, we can identify one of the most active subsets of AI to be machine learning, which refers to any algorithm which uses patterns in previously encountered data to make predictions regarding current inputs. Such models can be as 'simple' as traditional statistics, such as regression; however, the slightly more exotic 'artificial neural networks' (ANNs) have, as of late, become extremely popular [3].

As the name suggests, ANNs are loosely based on biological neural networks in that they have a discretised, node-like structure similar to neurons in a brain; as shown in Fig. 1 [3].

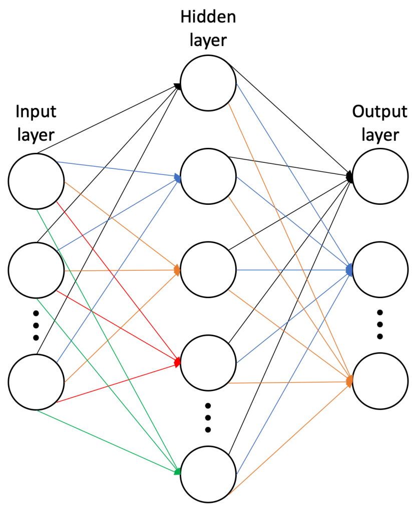


Fig. 1. Diagram showing the basic structure of an artificial neural network. The colour-coded connections between the nodes shows the parallels between ANNs and biological neural pathways. The sheer number of total connections in this very small network helps to demonstrate the plausibility of reasonably sized ANNs being able to store the necessary information to solve abstract problems.

To understand Fig. 1 in more detail, let us, for the sake of example, consider how one would go about building a simple 3-layer ANN which, when given a monochromatic image of a shape, classifies it as either a triangle or a square.

We start by quantifying the input. For this specific example, we can take each one of the bits of an image and place it in its own node within the 'input layer'.

The next step would be to multiply each of these nodal values by some random 'weight' and add a random 'bias' before passing them on to each of the nodes in the 'hidden layer'. Also, normalising the final result such that the value lies between 0 and 1 will be useful later. This normalisation is commonly achieved through the application of a sigmoid function. We can then repeat this step to move from the hidden layer to a final, 2-node output layer [3].

Let us interpret the value in the first and second node as the probability that the image is of a triangle and square respectively, which we can do given the normalisation. For the first pass through the network, these probabilities will be completely random. However, we can quantify the extent to which the model is incorrect (or the 'cost') by calculating the sum of the square differences between the obtained value and the correct value for each node [4] [3].

Each time we pass all of the available 'training' images through the network, we go back to adjust the weights and biases so as to decrease the average cost [4]. With hundreds if not thousands of training images, we may find that in time the model becomes very accurate indeed.

Mathematically, by expressing the original input as a vector $\boldsymbol{x}$ and the weights and biases for the j[th] layer of the network as a matrix W_j and vector $\boldsymbol{b}_j$ respectively, one can obtain a very concise definition of the average cost c as

$$c = \frac{1}{N}\sum_{i=1}^{N}\left|\sigma\left[W_1^{(i)}\left(W_0^{(i)}\boldsymbol{x}^{(i)} + \boldsymbol{b}_0^{(i)}\right) + \boldsymbol{b}_1^{(i)}\right] - \boldsymbol{y}_T^{(i)}\right|^2, \quad (1)$$

where the index i refers to the image being processed and σ, N and $\boldsymbol{y}_T$ refer to the normalising sigmoid function, total number of images and the correct output vector respectively.

From (1) is it clear that the average cost is a function of all of the weights and biases present in the network, henceforth denoted as a single column vector $\boldsymbol{\Theta}$. Thus, to minimise the average cost, we need only step $\boldsymbol{\Theta}$ back in the direction which most quickly decreases the cost; $-\nabla c(\boldsymbol{\Theta})$ [3]. This process of iterative gradient descent is shown in Fig. 2.

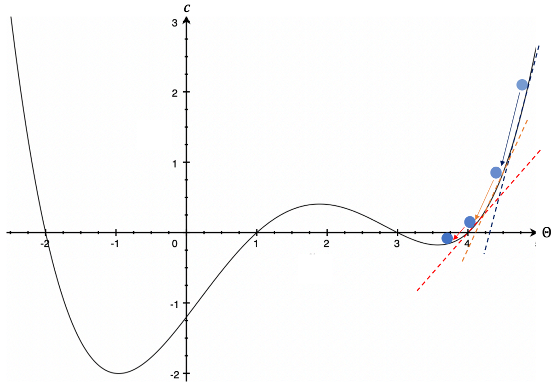


Fig. 2. Diagram showing gradient descent. Note that for simplicity Θ is taken to be a scalar here; however, the idea presented can be extended to any n-dimensions. Notice that keeping the step size proportional to the gradient $|\nabla c(\boldsymbol{\Theta})|$ prevents the descent from overshooting a minimum [5]. Notice also that minimum found may not be global, as is the case here. Thus, even with an infinite number of training examples, gradient descent does not guarantee the optimum set of weight and biases for the problem.

There are some subtleties we have sidestepped here, the most important of which is how one goes about calculating $-\nabla c(\boldsymbol{\Theta})$. Nevertheless, the above process of 'learning' through minimisation of some cost accurately describes the core concept behind all 'feed-forward' neural networks.

Other Types of ANNs

Going beyond the basic structure of a feed-forward network, there exist many specialised ANNs which contain some additional complexity which enable them to better solve more complex problems.

One popular variation is known as a recurrent neural network (or RNN). Here the main change is that several nodes in the network may contain connections to either themselves or other nodes in the same layer [3]. These recurrences mean that any new input is considered in the context of previous inputs, thus the network obtains some concept of memory [3]. This ability is extremely useful for solving problems where input *order* is of importance, such as natural language processing. One can imagine the importance of distinguishing between sentences such as 'for the many not the few' and 'for the few not the many' when training a semantic classifier for example…

Another common model is the convolutional neural network (or CNN). Here there exist some nodes which instead of transforming values in the standard way with weights and biases, convolve their input with some filter matrix in a way which is completely analogous to some filter function f convolving some input function g such that the output is $f * g$ [3]. Such convolutional matrices act almost as pattern detectors in the network; thus, CNNs are useful for solving problems where input *structure* is of importance, such as image recognition [3].

The last ANN variant we consider here is the deep neural network, which refers to any network with more than one hidden layer. Such models are particularly effective at building 'hierarchal representations' [6] of their input data such that the first layer considers only the lowest-level features and the abstractness increases until the last layer considers the highest-level features [3]. This structure makes 'deep learning' an effective tool for analysing abstract or multidimensional inputs [3].

Applications in High Energy Physics

It can be shown that any feed-forward neural network with at least one hidden layer can approximate any continuous function; they are 'universal approximators' [7]. Thus, ANNs can, in principle, be used for any problem which reduces to a functional mapping from an input vector $\boldsymbol{x}$ to an output $\boldsymbol{y}$.

Practically, however, ANNs generally lend themselves to problems which satisfy two essential criteria;

i. the problem is abstract in nature; and

ii. there exists a vast amount of data through which a model can be trained.

It is exactly these criteria which make neural networks so ideal for use in high energy physics (or HEP) research. To explain why this is the case, it is useful to understand one of the core problems found in fields with a heavy emphasis on data analysis; the 'curse of dimensionality' [6].

In the context of HEP, consider an event where only one variable, say the position of a particle, is measured. An approximate probability distribution for the particle's position could then be built with some 'N samples' [6]. However, if you then choose to consider two variables simultaneously, you will invariably find that N^2 samples are now required to construct the statistical model to the same level of accuracy as before [6]. More generally, the amount of data required to construct any model to a consistent level of accuracy scales exponentially with the dimensionality of the input [6].

HEP experiments, such as those taking place at the Large Hadron Collider are infamous for the sheer number of variables they track at any given time. Thus, it would be impractical at best and impossible at worst to create any meaningful model of an HEP event without some pre-processing to reduce the dimensionality of the data collected while minimising the amount of information lost in the process [6].

This problem of low-loss dimensionality reduction satisfies the previously mentioned criteria and is indeed where neural networks found some of their earliest success within the field [6].

Historically, feed-forward and recurrent neural networks only replaced 'low-level' operations within the overall process of data reduction [8]. Here, the emphasis was very much on identifying particle tracks, noise reduction and other general pattern recognition tasks based on 'well understood' particle behaviour [6]. The subsequent 'high-level' operations, such as finding the 'angular distribution' of particle jets [8], were driven by more contentious physical models. More careful consideration of any potential model bias was needed here and thus such operations were often not completed through ANNs [8].

Deep neural networks in particular have also proven themselves to have several useful applications in HEP. One such application is event selection, which is the process by which events of interest are isolated from auxiliary events within a single signal [6]. What is particularly interesting here is that in recent years, it has been shown that, despite the dimensionality problem, deep networks given low-level data tend to be more accurate than those given high-level data [9]. Thus, astonishingly, making use of less human intelligence can sometimes produce more favourable results…

Other Applications

Applications of neural networks are not restricted to HEP; many different subfields within physics benefit from the technology. Consider, for example, the field of fluid dynamics. A major problem within this area is that although there exists an analytical framework which accurately describes fluid behaviour, namely the Navier-Stokes equations, solving these intractable equations for systems on reasonable scales often proves to be very computationally expensive [10]. As such, scientists are always on the lookout for methods which can accurately simulate fluid behaviour with a computational complexity lower than a full numerical solution.

As was previously mentioned, multilayer neural networks are 'universal approximators' [7], thus one can immediately see the appeal in using ANNs to approximate the Navier-Stokes equations and generate accurate fluid simulations quickly. Indeed, many such networks have been built. One of note is 'FluidNet'; a convolutional neural network built in a recent Google – New York University collaboration [10] [11]. Here researchers were able to create an accurate smoke simulation with a computational runtime a remarkable two orders of magnitude lower than that for a traditional approximation method. Their results are shown in Fig. 3.

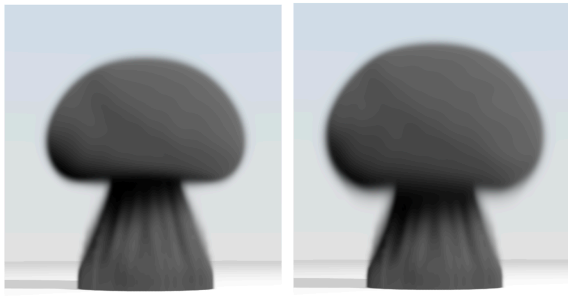


Fig. 3. A frame from a smoke plume simulation. The plume on the left and right was obtained from a traditional approximation method and 'FluidNet' respectively. The two simulations seem to give very similar results; however, the one making use of an ANN runs far more quickly. Figure reproduced from [10].

The Future of ANNs in Physics

The preceding sections mention the idea that it is often the case that an ANN is able outperform solutions based on human intelligence. It therefore follows that such networks must contain, in some abstract form, knowledge of which we are not yet aware. However, how would one go about retrieving this information from the myriad of parameters present within the model? This inability of neural networks to explain their decisions is often referred to as neural network 'opacity' [12] or, more affectionately, the 'black box' problem [13]. To date, this issue remains largely unsolved; however, let us speculate the potential benefits a future solution may bring.

Perhaps the most impactful benefit would be the removal of training data bias [14]. To illustrate this, consider a peak detection ANN trained on simulated data. Simulations are often imperfect and so it may be the case that the simulated data is biased in some way; for example, it may be that all generated peaks are negatively skewed. This may lead the neural network to 'believe' that negative skew is a general characteristic of all peaks. Thus, the model may incorrectly classify positively skewed peaks as background during a real experiment. If the network was transparent, however, and was able to explain its reasoning, scientists would be better able to identify this bias and remove it prior to model deployment. Such a change would undoubtedly reduce the systematic uncertainty present within the experiment.

Other more exotic benefits have also been speculated. One such idea is perhaps best described as the automated generation of wisdom. Here the thinking is that the network's ability to communicate its findings may enable us to more rapidly develop our existing theoretical knowledge. One may even imagine a network where, to quote Caltech physicist Jean-Roch Vlimant, "you would just throw data at this machine, and it would come back with the laws of nature" [15].

Such an idea seems fanciful right now, however, we have over recent years seen some real progress towards this endgame. Consider, for example, the recent work from the Lincoln Laboratory at MIT where researchers have developed the 'Transparency by Design Network' (or 'TbD-net') [16]. This neural network is not only capable of solving abstract 'visual reasoning' tasks but also outputting its decision-making process as human-interpretable 'attention masks', which highlight regions of the image on which the network is currently focussed [16].

Conclusion

It is clear at this point that artificial neural networks are one of the most revolutionary methods of data analyses to emerge in recent decades. Its applications in physics are far from cute novelties as demonstrated by the fact that the technology has already impacted several subfields and indeed contributed to many remarkable scientific achievements, including the discovery of the Higgs boson in 2012 [17]. The future of the technology looks brighter still, with one particularly exciting development being the rise of transparent neural networks.

So, are we destined for a time when physicists are made redundant by our very own version of 'Deep Thought'; the supercomputer immortalised in Douglas Adam's 'The Hitchhiker's Guide to the Galaxy'? Probably not. Nevertheless, I am sure you would agree that machine learning will play a significant role in the scientific research of the years to come.

REFERENCES

[1] Samsung, "'All In' on AI, Part 3: The Family Hub Recognizes You and Provides a Personalized Diet," 13 March 2018. [Online]. Available: https://news.samsung.com/global/all-in-on-ai-part-3-the-family-hub-recognizes-you-and-provides-a-personalized-diet. [Accessed 28 December 2019].

[2] S. S. Shukla and V. Jaiswal, "Applicability of Artificial Intelligence in Different Fields of Life," *International Journal of Scientific Engineering and Research (IJSER),* vol. 1, no. 1, pp. 28-35, 2013.

[3] E. Alpaydin, Introduction to Machine Learning, Cambridge, Massachusetts: The MIT Press, 2014.

[4] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, MIT Press, 2016.

[5] A. A. Hopgood, "Hill-climbing and gradient descent algorithms," in *Intelligent Systems for Engineers and Scientists*, CRC Press, 2000, p. 169.

[6] D. Guest, K. Cranmer and D. Whiteson, "Deep Learning and Its Applications to LHC Physics," *Annual Review of Nuclear and Particle Science,* vol. 68, no. 1, pp. 161-181, 2018.

[7] K. Hornik, M. Stinchcombe and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks,* vol. 2, no. 5, pp. 359-366, 1989.

[8] B. Denby, "Neural networks in high energy physics: A ten year perspective," *Computer*

Physics Communications, vol. 119, no. 2-3, pp. 219-231, 1999.

[9] P. Baldi, “Searching for exotic particles in high-energy physics with deep learning,” *Nature Communications,* vol. 5, no. 1, 2014.

[10] J. Tompson, K. Schlachter, P. Sprechmann and K. Perlin, “Accelerating Eulerian Fluid Simulation With Convolutional Networks,” in *34th International Conference on Machine Learning*, Sydney, 2017.

[11] K. Zsolnai-Fehér, “Neural Network Learns The Physics of Fluids and Smoke | Two Minute Papers #118,” YouTube, 8 January 2017. [Online]. Available: https://www.youtube.com/watch?v=iOWamCtnwTc. [Accessed 3 January 2020].

[12] J. Burrell, “How the machine ‘thinks’: Understanding opacity in machine learning algorithms,” *Big Data & Society,* vol. 3, no. 1, 2016.

[13] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.-Y. Glorennec, H. Hjalmarsson and A. Juditsky, “Nonlinear black-box modeling in system identification: a unified overview,” *Automatica,* vol. 31, no. 12, pp. 1691-1724, 1995.

[14] A. Torralba and A. A. Efros, “Unbiased Look at Dataset Bias,” *CVPR,* vol. 1, no. 2, pp. 1521-1528, 2011.

[15] D. Castelvecchi, “Can we open the black box of AI?,” *Nature,* vol. 538, no. 7623, pp. 20-23, 2016.

[16] K. Foy, “Artificial intelligence system uses transparent, human-like reasoning to solve problems,” MIT Lincoln Library, 11 September 2018. [Online]. Available: http://news.mit.edu/2018/mit-lincoln-laboratory-ai-system-solves-problems-through-human-reasoning-0911. [Accessed 1 December 2019].

[17] The ATLAS Collaboration, “Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC,” *Physics Letters B,* vol. 716, no. 1, pp. 1-29, 2012.

[18] P. Kokkas, M. Steinacher, L. Tauscher and S. Vlachos, “Neural network real time event selection for the DIRAC experiment,” in *AIP Conference Proceedings*, Basel, 2001.