J of Nuclear Medicine Technology, first published online August 10, 2019 as doi:10.2967/jnmt.119.232470

Intelligent Imaging: Anatomy of Machine Learning and Deep Learning

Geoff Currie

School of Dentistry & Health Sciences, Charles Sturt University, Wagga Wagga, Australia

Footline: Anatomy of ML and DL

Key words: nuclear medicine, artificial neural network, deep learning, convolutional neural network, artificial intelligence

Abstract

The emergence of artificial intelligence (AI) in nuclear medicine and radiology has been accompanied by AI commentators and experts predicted that AI would make radiologists in particular extinct. More realistic perspectives suggest significant changes will occur in medical practice. There is no escaping the disruptive technology associated with AI, neural networks and deep learning; the most significant perhaps since the early days of Roentgen, Becquerel and Curie. AI is an omen, but it need not be foreshadowing a negative event but rather heralding great opportunity. The key to sustainability lies not in resisting AI but in having a deep understanding and exploiting the capabilities of AI in nuclear medicine while mastering those capabilities unique to the human resources.

Introduction

Artificial intelligence (AI) is a general term used to describe algorithms designed for problem solving and reasoning. Applications in nuclear medicine and radiology have been widely documented. A subset of AI is associated with neural networks. In medical imaging, a neural network is an image analysis algorithm composed of layers of connected nodes (*1*). The nodes can be in the order of hundreds to millions and simulate the neuronal connections of the human brain (*2*). Nodes receive information from other nodes or patterns of nodes. Communication from one node to other nodes occurs when a threshold is exceeded and the outputs from those nodes are weighted (figure 1). The basic principle is to maximise the number of correct answers by comparing artificial neural network (ANN) estimates with a reference (grounded truth) and then adjusting the weightings on each node based on the error (*2,3*). There may be hundreds or thousands of iterations required to make the adjustments during the training phase of developing an ANN. Clearly the more data that is used to train the ANN, the greater the accuracy of the inference phase. Through each iteration and subsequent adjustment of the nodes, a mathematical solution converges on a more accurate solution in a similar manner that we might think about iterative reconstruction of single photon emission computed tomography (SPECT) and position emission tomography (PET) data.

An ANN typically has three phases; the training phase where the ANN learns, the validation phase where the learning of the ANN is evaluated against a second dataset, and the inference or application phase where the ANN is applied to actual cases. The training phase follows a diminishing return principle, eventually reaching a point where additional iterations do not improve the results or the improvement is negligible (figure 1) (*2*). The training phase can be supervised (grounded truth is human interpreted training data) or unsupervised (no grounded truth, learning based on pattern recognition) (*4*). Following the training phase, a second data set can be used to test the accuracy of inferences of the ANN to provide validation of the algorithm before being used in clinical and research applications (figure 1). The role of big data in medical imaging is to provide a reliable and large training data base for machine learning (ML), representation learning and deep learning (DL) algorithms to learn and produce accurate

outcomes (*1*). There are, however, potential clinical and research roles for ANNs in parallel to conventional statistical analysis in small data to identify key inputs (features) or combination of inputs not gleaned from multi-variate analysis.

In the validation phase, a second smaller data base of features or images are used for the ANN to evaluate and those inferences are compared to a grounded truth (figure 2). This represents the testing and validation phase and predicts the accuracy of the ANN when used clinically or in research (*5,6*). That degree of accuracy can then be expected in the application phase where the neural network makes inferences about images without a grounded truth (supplemental figure A). An ANN would have data or features entered into the input layer of the algorithm as depicted in figures 1.

DL associated with convolutional neural networks (CNN) have a higher order functionality where the neural network itself is trained to identify and extract features from images (*7*) (figure 2). The term convolution means the mathematical combination of 2 functions to generate a third function. As depicted in figure 2, the input has a number of image dimensions (X, Y and Z) and a number of images (eg. SPECT slices). The image itself has specific features identified and extracted into a convolution feature map (*7*). A kernel or rectified linear unit (ReLU) is an activation filter through which convolution data are pooled (*7*). Multiple convolution, kernel and pooling iterations may occur before the pooled features are flattened for entry into the input layer of the fully connected neural network (*7*). The depth of the CNN gives rise to the expression 'deep learning'.

Anatomy of Machine Learning

ML algorithms, including ANNs have 3 key components (*6,8*):

1. The mathematical model which is used to describe or explain the relationships within the data. Specifically, the relationships between inputs (features) and outputs (outcomes).

2. The cost function which is an evaluation of the accuracy of the mathematical model. This is a measure of how well the model predicts and outcome and the error between prediction and expected is the loss function.
3. The data itself is necessary but varies for the training phase, validation phase and then the inference phase. Big data from multi-centre trials may be used for the training phase and a smaller population of cases with known outcomes could be used for the validation phase. Typically, the same database is used and randomly split (eg. 80:20) to produce a large training set and a smaller but statistically significant validation set. A separate population of cases can then be used as the inference phase for further research (deeper validation with external validity) or for clinical decision making.

There are a variety of ML algorithms available and the preferred approach (eg. CNN versus ML) will depend on the type of data and the purpose. For simplicity, the following discussion will assume a binary output (eg. cardiac event or no cardiac event) and rich input data of extracted features in a model that resembles figure 1. One should keep in mind that this is a model meant for aiding the understanding of nomenclature and processes rather than being a fit for all ANNs; in the same way human anatomy has normal variants and differs amongst mammals despite having some commonality.

Consider a number of potential input features (eg. 4) in 1000 patients in a database. A single binary output might be a cardiac event during the follow-up period or no cardiac event in the follow-up period. The ANN architecture would include 4 scaling layer inputs, a number of hidden (perception) layers (let's assume 4) of multiple nodes in each hidden layer (perhaps 4, 8, 8, 3) (figure 3). The scaling layer is to ensure all inputs are within the prescribed range and contain input statistics (eg. mean, standard deviation, minimum, maximum etc). Each node (perceptron) in the perception layers receives numerical inputs (X_1, X_2, X_3 etc) which have weightings (W_1, W_2, W_3 etc) and combined with a bias (B) to produce a single net input value (C) where (*8,9)*:

$$combination = bias + \sum weights \cdot inputs$$

An activation function (A) defines the output (Y) of the perceptron (liner, logistic, rectified linear, hyperbolic tangent etc) (*5,7,9,10*). In the case of a linear activation function, the activation is equal to the net input value (*5,7,9,10*). The more common logistic activation function is a sigmoid function where:

$$activation = \frac{1}{1 + e^{-combination}}$$

The ANN works toward a probabilistic layer (eg. binary, continuous, competitive, softmax etc) or probabilistic output function. Between the last perception layer and the probabilistic layer, an unscaling layer is needed to convert outputs to the original units (*8-10*).

The architecture needs to be trained and optimised. The loss index is a tool used to measure the error associated with the algorithm executing its task (error term) and to measure the quality of the data the ANN is learning (regularisation term) (*5,7,8,10,11*). The error term can be measured in numerous ways including mean squared error, normalised squared error, weighted squared error (WSE) or Minkowski error (*9)*. The WSE method could be used to determine the loss index especially when there is an imbalance between positive and negative outputs (eg. a ratio of 1.2:1 against grounded truth). Regularisation relates to the size of changes in outputs in response to changes in inputs; small changes producing small changes being considered regular. The regularisation term is summed with the error term which will reduce weights (W) and biases (B) to produce a smoother output (*9,10*).

Optimisation is an adjustment to the weightings on individual nodes (perceptrons) in order to minimise (optimise) the loss index (*5,8-10*). This is achieved using an iterative process of successive adjustments to the weightings. Gradient is the rate of inclination or declination (slope) and represents the learning rate. Gradient descent is an optimisation method that evaluates a progressive diminishing rate of learning with each iteration (*5,9-11*). That is, the cost function is decreasing which means the loss is decreasing and the minimum point could be used to terminate

the cycle (before loss starts to increase again) (*5,7,8,10,11*). Large data sets may not be able to be processed concurrently and this requires division of the data. An epoch refers to the entire data set passed forward (forward propagation) and backward (back propagation) through the ANN once. This is often referred to as an iteration and for small datasets an iteration and an epoch are the same. In larger datasets the, the data may need to be broken into batches of smaller units. Each time a batch is forward propagated and back propagated through the ANN it is an iteration. Once all batches are passed through once, it is an epoch. For the data set of 1000 patients, the data may need to be broken into batches of 200 which means we have 5 batches requiring 5 iterations to complete 1 epoch. The optimisation algorithm, therefore, changes parameters between successive epochs (parameter increment) to minimise the loss index until a specified condition is met (eg. minimum value reached, margin loss improvement equals a set value, gradient equals pre-set value, maximum number of epochs reached, maximum time reached) (*5,10*). The optimisation algorithm itself defines how parameters are optimised (*9,10*). The Newtonian method is computationally demanding but more accurate; employing the Hessian of the loss function (second derivative matrix) (*9)*. A Quasi-Newtonian method may be a preferred option and this approach uses gradient information to estimate the inverse Hessian (mathematical function using a square matrix of second order partial derivatives) for each iteration of the algorithm ignoring second derivatives and reducing computational demand. Other approaches include gradient descent, conjugate gradient, Levenberg-Marquardt algorithm, stochastic gradient descent and adaptive linear momentum. The loss function associated with the training phase estimates the error associated with the prediction and the grounded truth for the dataset (*5,9,10*). The selection loss is an error measure of the ANNs generalisability to new data or agility. These loss functions can be used to optimise the number of hidden layers / iterations in the final architecture.

The final architecture of the ANN or model selection needs to consider selection loss, or minimise the error associated with the order and range of inputs (*5,7,9,10*). Order selection relates to depth of the ANN on their influence on the output and its accuracy by defining the number of nodes in hidden layers (*5,9,10*). It is important to balance the order selection with the complexity

of the data to avoid under or over fitting (*4*) (figure 4a). Similarly to the training error, the selection error measures the accuracy of the ANN applied to new data (generalisability) (*5,9,10*). An incremental order selection algorithm starts by measuring selection loss for a small number of nodes and incrementally adds nodes until the selection loss is optimised (meets pre-determined value). Conversely, a decremental order algorithm starts by measuring selection loss for a large number of nodes and incrementally removes nodes until the selection loss is optimised (meets pre-determined value). In this case, knowing the low complexity of data, the user has elected to begin with a more complex ANN than necessary which will see the decremental order algorithm reduce the complexity in the ANN.

Inputs selection (figure 4b) defines which specific features should be included in the ANN inputs. The inputs selection algorithm determines which input features produce the smallest selection error and, thus, provide the best generalisability for the ANN to new data (*5,9,10*). There are several algorithms that can be used. The pruning method starts with all inputs and incrementally removes inputs with the lowest correlation until the selection loss starts to rise. A growing inputs method can also be used to calculate the correlation for every input against each output in the data set. Beginning with the most highly correlated inputs, incrementally inputs are added to the network until the selection loss increases. The final architecture of the neural network reflects the optimised subset of inputs and order with the lowest selection loss (supplemental figure B).

A number of metrics can be employed to test the errors in the neural network. The final architecture can then be evaluated using a number of tests for robust validation using a second set of data (or validation partition of the original data set) (*9,10*). The loss index for the final ANN can be calculated by comparing the prediction output with the grounded truth (*7*). A number of tools are used in combination for validation including, but not limited to; sum squared error, mean squared error, root mean squared error, normalised squared error, Minkowski error, cross entropy error, hinge error and linear regression analysis. Receiver operator characteristics (ROC) analysis produces an area under the curve (AUC) that correlates with a sensitivity and specificity (*9,10*). This is further reflected in the confusion matrix (true positives, true negatives, false

negative and false positive). ANN performance may also be expressed or displayed as cumulative gain (benefit of using the ANN over a random guess), lift chart (ratio of positive events using the ANN to those without the ANN), conversion rates (percentage of predicted cases with and without the ANN), and profit chart (ANN gain over random guess). Much of the literature on ML applications in nuclear medicine and radiology are in some way the validation phase of the ANN. This may include statistical analysis of the ANN capability against human interpretation and a "gold standard". It may also include an evaluation of the predicted gain in economic or health outcome terms with and without the ML model. Post validation, the ML algorithm can be implemented by exporting and applying the mathematical model. For simple ML and ANN models, this may represent an export of the mathematical expression in simple code language like Python for incorporation in mobile device Apps on websites.

An example of this application is previous work with ^{123}Iodine meta-iodobenzylguanidine (^{123}I-mIBG) radionuclide imaging in heart failure [Iqbal!!!]. Traditional analysis with multivariate approaches demonstrated regional washout associated with territories adjacent to infarcted myocardium was superior to traditional planar approaches to uptake and washout in predicting cardiac events [Iqbal]. Subsequently, the same data was evaluated using an ANN in the method described above using 84 input variables and a single binary output (cardiac event or no cardiac event in the follow-up period). Training and validation phases optimised the number of inputs at just 2; a change in LVEF (Δ > -10%) and 123I mIBG planar global washout (>30%) (*12,13*). The ANN in this case revealed predictive capability not illuminated by traditional regression methods, highlighting the value of ANN/ML in parallel to conventional statistical analysis.

Anatomy of a CNN

With the general principles of an ANN outlined above, scaffolding a deeper insight into the CNN process might be of value. As outlined in figure 2, a CNN is comprised of convolution and pooling layers, and the fully connected layers of a neural network. The CNN differs from the ANN described in figure 3 in that the features are extracted from the images and the output is some form of classification (*7*). As described below, the CNN transforms 2-dimensional image data

through forward propagation but can also be applied to 3-dimensional data sets such as SPECT and PET (*7,11*).

Convolution is the extraction of image features using a linear operation that applies a kernel (typically 3 x 3) to a subset array of image elements (pixels) or input tensor (*5,7-9,11,14*) (figure 5). This process is not dissimilar to the application of a 9-point smoothing filter to planar images in nuclear medicine. The kernel is positioned over elements in the input tensor, with the distance between each successive position representing the stride (*5,7-9,11,14*). A stride of 1 means that the kernel is positioned centred over each element of the input tensor while a stride of 2 would indicate positioned centred over every second element of the input tensor. This down sampling of feature maps with strides greater than 1 can be better achieved in the pooling function (*5,7-9,14*). The product of the individual elements of the input tensor and the kernel are summed to produce a single numerical value (and position ordinates) into the feature map (output tensor) (*5,7-9,14*). A variety of kernels can be applied in a stepwise manner producing a number of convolution layers (figure 2). Of importance in convolution is that while the X and Y dimensions of the input tensor are compressed, the Z dimension does not change. The post convolution feature map is then passed through a non-linear activation function that, as previously described, is typically the ReLU before entering the pooling layer (*5,7-9,14*).

Pooling reduces the in-plane (X,Y) dimensionality of feature maps by applying a down sampling operation (*5,7-9,11,14*). Max pooling and global average pooling are 2 common approaches. As the name suggests, max pooling creates an output equal to the maximum value within a patch of data in the feature map (*5,7-9,14*). A 2x2 filter with a stride of 2 means that each set of 4 elements is represented as a single value equal to the maximum value and all other data is discarded (figure 6). Global pooling on the other hand, represents a feature map as a single value equal to the mean of the element values; essentially down sampling a feature map to a 1x1 array (*5,7-9,14*). This is typically applied once immediately prior to the fully connected layers, however, the max pooling method is more common (*5,7-9,11,14*). Multiple sequential convolution, kernel

and pooling processes produce layers of data that are transformed into a 1-directional array of vectors (numbers) through a process called flattening (*5,7,11*).

A parameter is a variable automatically learned by the CNN while a hyper-parameter is a variable that needs to be set (*7*). These vary in the different layers of the CNN (figure 7). In the convolution layer, kernels are the parameter and kernel size, kernel number, stride and activation function are the hyper-parameters. The pooling layer has no parameters but the pooling method, filter size and stride are all hyper-parameters. The fully connected layer of nodes uses weights as the parameters while the activation function and the number of weights are the hyper-parameters.

There are a wide variety of applications of CNN and DL in nuclear medicine but the application of a CNN is effectively demonstrated in recent dementia studies. SPECT images with known outcomes were used to train a CNN to evaluate the images themselves and identify key features; specifically the cingulate island sign indicative of dementia with Lewy bodies (*15*). Perhaps a more important approach would be the use of a CNN trained to identify specific features on the images themselves of findings of an urgent nature; pulmonary embolism on a lung scan for example. Rather than the CNN providing a definitive diagnosis, a list base report could be initiated and the findings used to triage a positive outcome to the front of the reporting list. Clearly, a CNN could be readily trained to identify features to drive automated segmentation or region identification and this may have significant applications in radiation dosimetry (*16*).

There is, however, a requirement for a degree of caution with application of DL and CNNs. While a CNN has the capability to identify features or relationships between features in a large volume of data not possible for a human observer, unsupervised learning may see unusual features identified. For example, consider a CNN trained to identify pulmonary embolism on a lung scan. If that CNN was shown to be more accurate than a human observer in detecting pulmonary embolism, then it makes sense that the CNN has identified features not typically considered by the human observer. This should prompt enquiry to educate the human observer to previously unconsidered features. In theory, the entire process improves. It may, however, reveal that

instead of identifying features in the lung fields themselves, the CNN may have learned other features that strongly correlate with pulmonary embolism; ECG leads insitu, annotation indicating referral for emergency, patient age or gender. Anecdotal discussion at conferences reveals a CNN to detect pneumonia on chest xrays was revealed to be making decision based on whether the study was performed in the department or by mobile xray.

Discussion

While there has been an emergence of medical literature outlining the applications of DL and CNNs in nuclear medicine and radiology, AI, ML, ANNs and CNNs afford numerous other opportunities. There are a number of key areas AI, ML, ANN or DL that have been successful or potentially impactful in radiology (*2,3,17*) and these are equally apt for nuclear medicine:

- To inform diagnostic decision tree and optimising procedure choice by predicting positivity rate amongst similar patients.
- New image reconstruction methods that produce images from lower radiation dose studies (eg. PET and SPECT), generation of pseudo-CT for attenuation correction, or with reduced imaging time (eg. MRI).
- Quality assessment algorithms built into systems to improve image quality and decrease repeat studies.
- Image triage algorithms that identify cases likely to be positive or that may have an urgent finding, allowing prioritisation of reporting and earlier intervention.
- Computer-aided detection, automated image annotation and information extraction.
- AI methods that explain analysis / interpretation and provide preliminary reporting.
- Lesion or disease detection (enhance computer aided detection) and classification.
- Automated segmentation, identification and extraction of features from images (radiomics) and quantitation.

Detection of incidental findings are important potential application of AI and ANNs not generally discussed in the literature but readily expressed in a mathematical algorithm (variation from normal). The emergence of the important role of radiation dosimetry modelling in radionuclide therapy will elevate precision nuclear medicine and theranostics; no doubt unveiling an important application of AI and ANNs.

The future of AI is very promising and looks beyond DL. Patrick Ehlen from LoopAI Labs explained in 2018 at a conference in Cologne Germany (https://www.loop.ai/ai-the-end-of-deep-learning?contentid=1302036) that the next generation of AI will go beyond DL. He used the liar paradox from star trek to highlight that AI is trained to solve problems logically. The human brain

is capable of both logical thought but also to operate in the sphere beyond logic sometimes referred to as illogical but this agility is "super logic". The simple liar paradox of AI interpreting 2 pieces of information; the first being "everything Harry says is a lie" and the second coming from Harry saying "I am lying" defies first order logic (https://www.youtube.com/watch?v=QqCiw0wD44U). Higher order logic that would prevent AI being outwitted by human "super logic" requires a framework of quantum based logic. While a tutorial on quantum computation is beyond the scope of this manuscript (*18*), the basic premise is that AI does not understand pragmatics. Humans process the contrasting context associated with pragmatism. These different foundation contexts could be seen as different basis vectors in quantum probability theory and allow AI to develop higher order reasoning and problem-solving skills. This has the potential to make dramatic steps in interpretation of complex images and pathological states associated with PET, SPECT, MRI and CT. Nuclear medicine and radiology has its strength in making clinical judgements and decisions based on data and feature extraction, not in the feature extraction and analysis itself (*19*). Thus, AI techniques like ML and DL provide an opportunity to enhance the accuracy and efficiency of the physician or radiologist without threatening redundancy. It may represent a shift in practice, with rudimentary tasks AI has a high degree of capability being lost to the radiologist or physician but this simply provides more time to focus on the higher order semantic tasks beyond, but enhanced by, the capabilities of AI. On the surface, this is a strong argument against the idea AI may make the physician or radiologist redundant. Quantum logic in AI may renew that debate.

Conclusion

AI has penetrated the daily practice of nuclear medicine over recent decades with little disruption. The emergence of ANNs and CNN applications has seen a significant shift in the landscape whose opportunity outweighs the threat. Nonetheless, understanding of the potential applications and the principles of AI, ANNs and DL will equip nuclear medicine professionals for ready assimilation; averting the ‘doomsday’ fears permeating radiology. Counter to the concerns amongst radiologists, in nuclear medicine the disruptive potential of the technology is perhaps of greatest impact on technologists and physicists rather than physicians.

References

1. Currie G. Intelligent imaging: artificial intelligence augmented nuclear medicine, *JNMT*, 2019, in press.

2. Tang A, Tam R, Cadrin-Chenevert A, et al. Canadian Association of radiologists white paper on artificial intelligence in radiology, 2018, *Can Ass Radiol J*, 69: 120-135.

3. McBee MP, Awan OA, Colucci AT, et al. Deep learning in radiology. 2018, *Acad Radiol*, 25: 1472-1480.

4. Ben-Nun T, Hoefler T. Demystifying Parallel and Distributed Deep Learning: An In-Depth Concurrency Analysis. 2018. 47pages. https://arxiv.org/abs/1802.09941 accessed 6 June 2018.

5. Lundervold AS, Lundervold A, An overview of deep learning in medical imaging focusing on MRI, 2019, *Z Med Phys*, 29: 102-127.

6. Uribe C, Mathotaarachchi S, Gaudet VC et al. Machine learning in nuclear medicine: part 1 – introduction. 2019, *J Jucl Med*, 60: 451-456.

7. Yamashita R, Nishio M, Do RKG, Togashi K. Convolutional neural networks: an overview and application in radiology. 2018, *Insights Imaging*. 9: 611-629.

8. Maier A, Syben C, Lasser T, Riess C, A gentle introduction to deep learning in medical imaging processing, 2018, *Z Med Phys*, 29: 86-101.

9. Goodfellow I, Bengio Y, Courville A, Deep learning. 2016, MIT Press http://www.deeplearningbook.org/ accessed 6 June 2019

10. Michael A. Nielsen, "Neural Networks and Deep Learning", Determination Press, 2015 http://neuralnetworksanddeeplearning.com/index.html accessed 6 June 2018

11. Shen D, Wu G, Suk H, Deep learning in medical image analysis, 2017, A*nnu Rev Biomed Eng*, 19: 221-248.

12. Iqbal B, Currie G, Kiat H, Regional cardiac sympathetic nervous system evaluation using 123I-mIBG SPECT in patients with heart failure, *Journal of Medical Imaging and Radiation Sciences*, 2018, 49: 397-405.

13. Currie GM, Iqbal B, Wheat JM, et al. Risk Stratification in Heart Failure Using 123I-MIBG. *J Nucl Med Technol,* 2011, 39:295-301.

14. Lin M, Chen Q, Yan S, Network in network. arXiv. 2013, Available online at: https://arxiv.org/pdf/1312.4400.pdf. Accessed 22 Jan 2018 accessed 6 June 2018

15. Iizuka T, Fukasawa M, Kameyama M. Deep-learning-based imaging-classification identified cingulate island sign in dementia with Lewy bodies. *Sci Rep*. 2019; 9: 8944.

16. Jackson P, Hardcastle N, Dawe N, Kron T, Hofman MS, Hicks RJ. Deep Learning Renal Segmentation for Fully Automated Radiation Dose Estimation in Unsealed Source Therapy. *Front Oncol*. 2018, 14:215.

17. Langlotz C, Allen B, Erickson B, et al. A Roadmap for Foundational Research on Artificial Intelligence in Medical Imaging: From the 2018 NIH/RSNA/ACR/The Academy Workshop**.** *Radiology*, 2019; 190613 DOI: 10.1148/radiol.2019190613

18. Ying M. Quantum computation, quantum theoru and AI, 2010, *Artificial intelligence*, 174: 162-176.

19. Liew C. The future of radiology augmented with artifical intelligence a strategy for success. 2018, *Euro J Radiol*, 102: 152-156.

List of figures

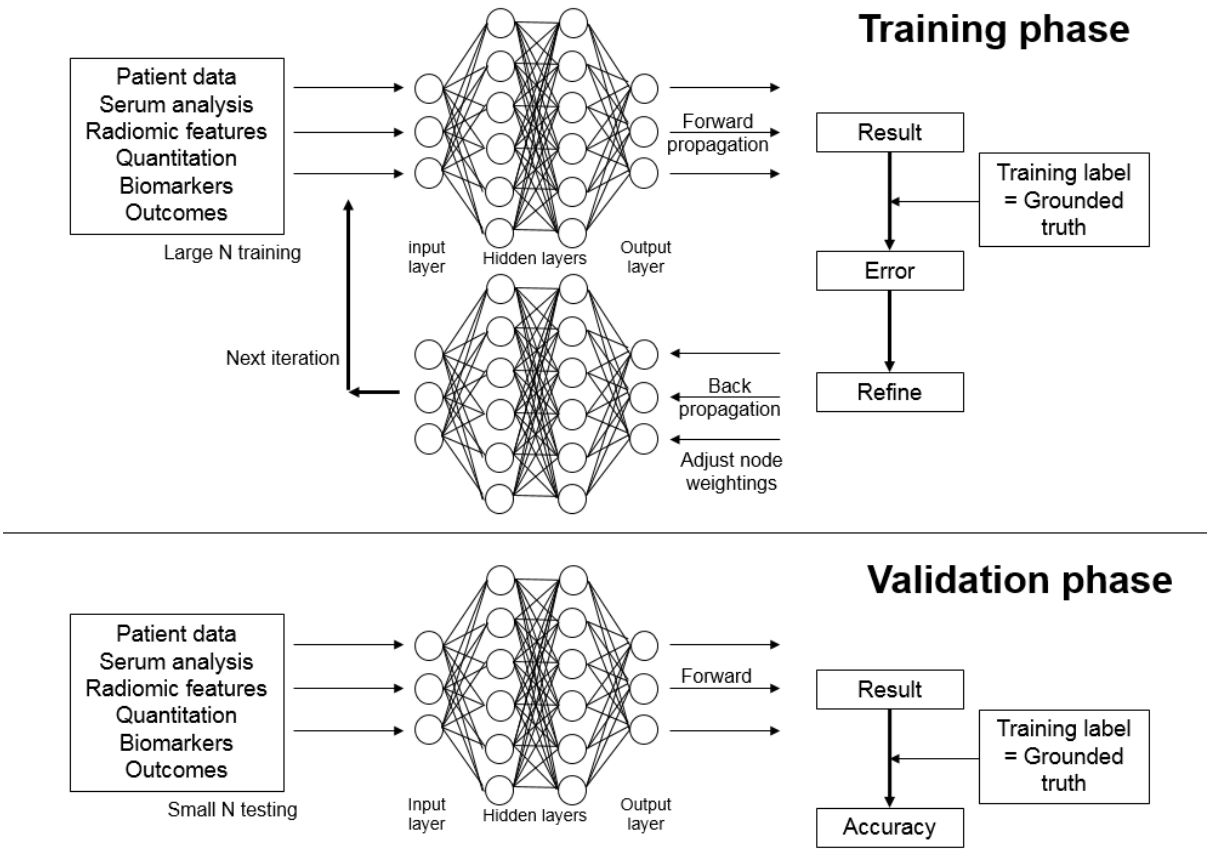


Figure 1: Training phase of a neural network using extracted features as inputs. The grounded truth defines this as a supervised artificial neural network (ANN). This is also the structure of an ANN that might be used as an analysis tool in parallel with traditional statistical analysis; importing data from a spreadsheet for example. In this example, all nodes are depicted as being connected to all others in adjacent layers and represents a "fully connected layer" which is more typical of convolutional neural networks. The validation phase evaluates the trained ANN against a new data base of known cases to determine accuracy. Adapted with permission (*1*).

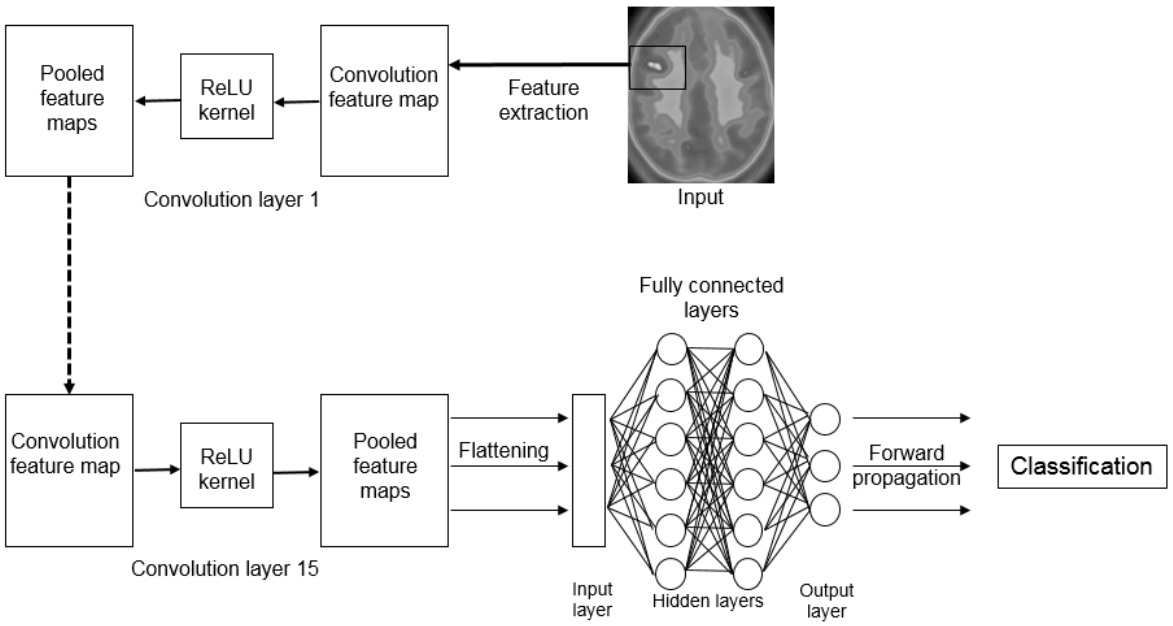


Figure 2: Basic structure of a convolutional neural network (CNN) where the network extracts the radiomic features, produces a convolution function, pools the data through a rectified linear unit (ReLU) kernel and flattens the pooled feature map for input into the fully connected hidden layers of the neural network. Reprinted with permission (*1*).

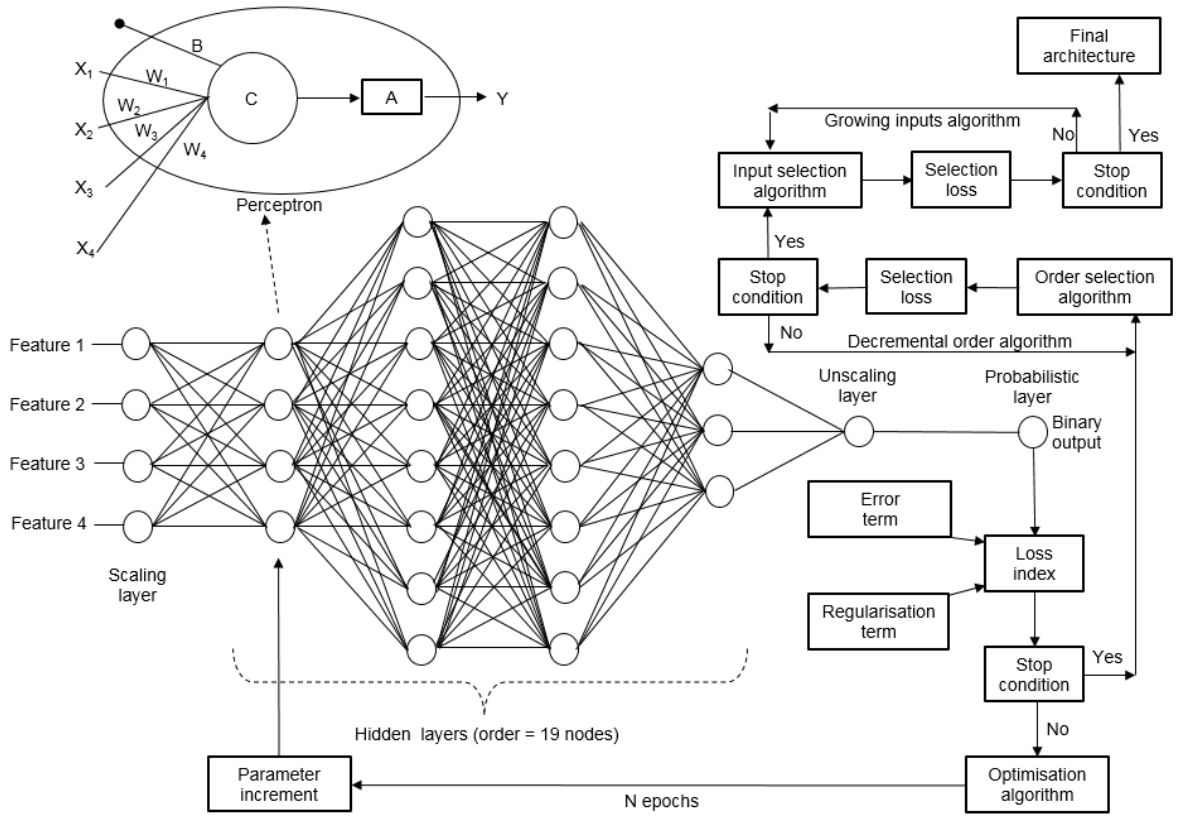


Figure 3: Overview of the anatomy of an ANN. A single node (C) can have multiple inputs (X) with different weighting factors (W) and a bias (B) but a single output (Y) via an activation function (A). The multiple lines represented exiting each node are the same output being delivered to multiple next layer nodes.

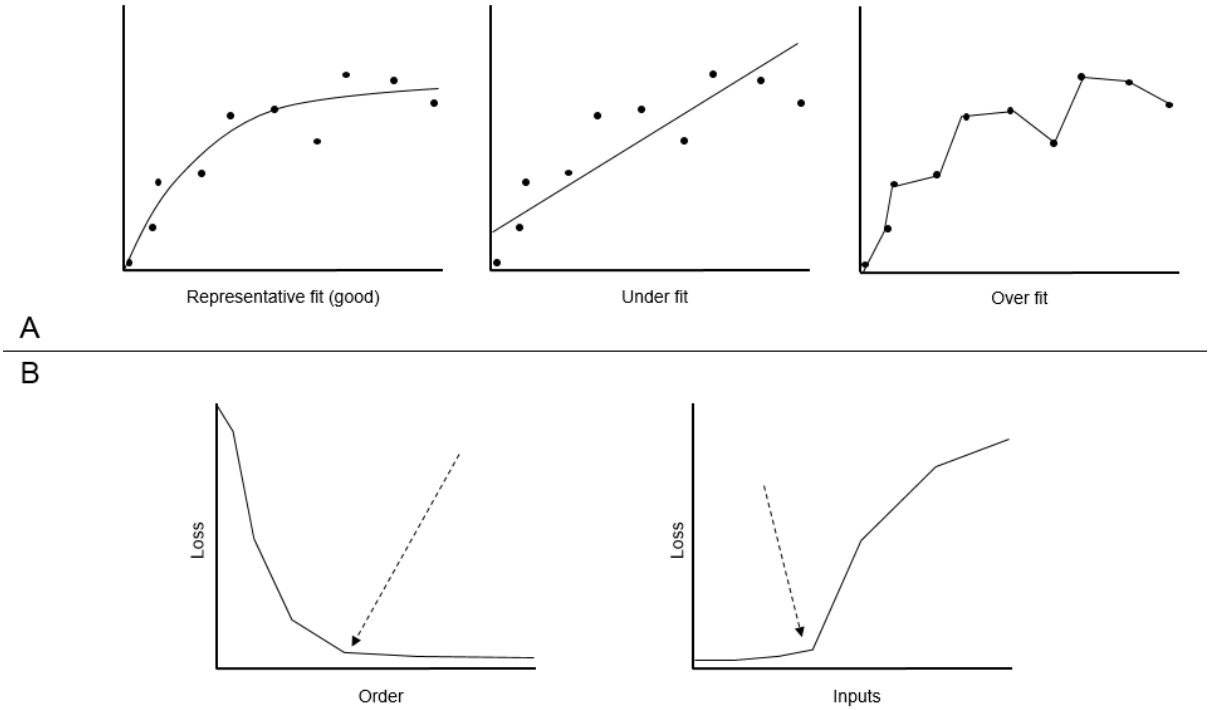


Figure 4: Schematic representation of a good fit versus under and over fitting associated with selection loss (A). Optimisation of the selection loss (B) to determine ANN complexity and node number (order) using a decremental order algorithm (left) with the arrow indicating a reasonable cut-off for total node number. Optimisation of selection loss to determine inputs (features) to be included (right) using a growing inputs algorithm with the arrow indicating a reasonable cut-off for inputs.

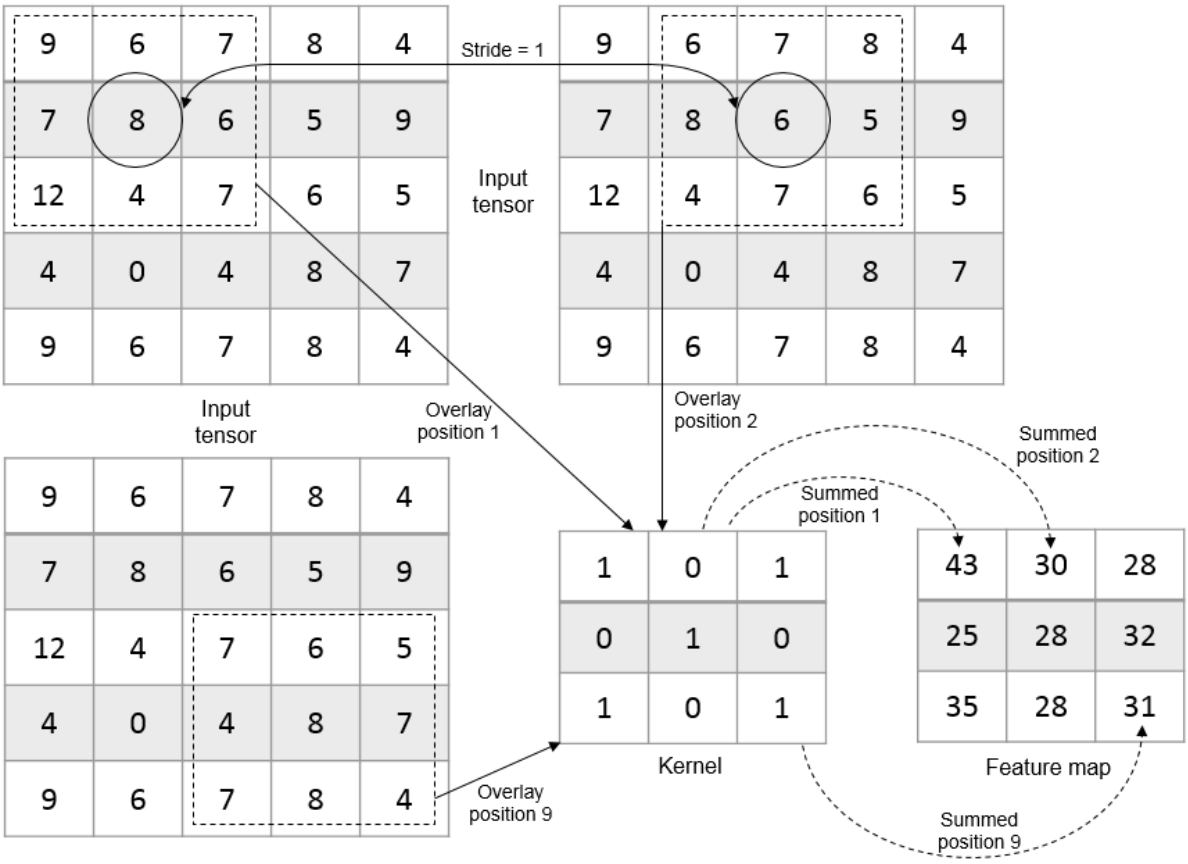


Figure 5: Convolution uses a 3x3 kernel to run sequential (in this case successive to provide a stride of 1) 3x3 array of elements. The weighted sum of the kernel for the 3x3 input tensor creates a single representative value in the feature map. Multiple feature maps are produced by different kernels.

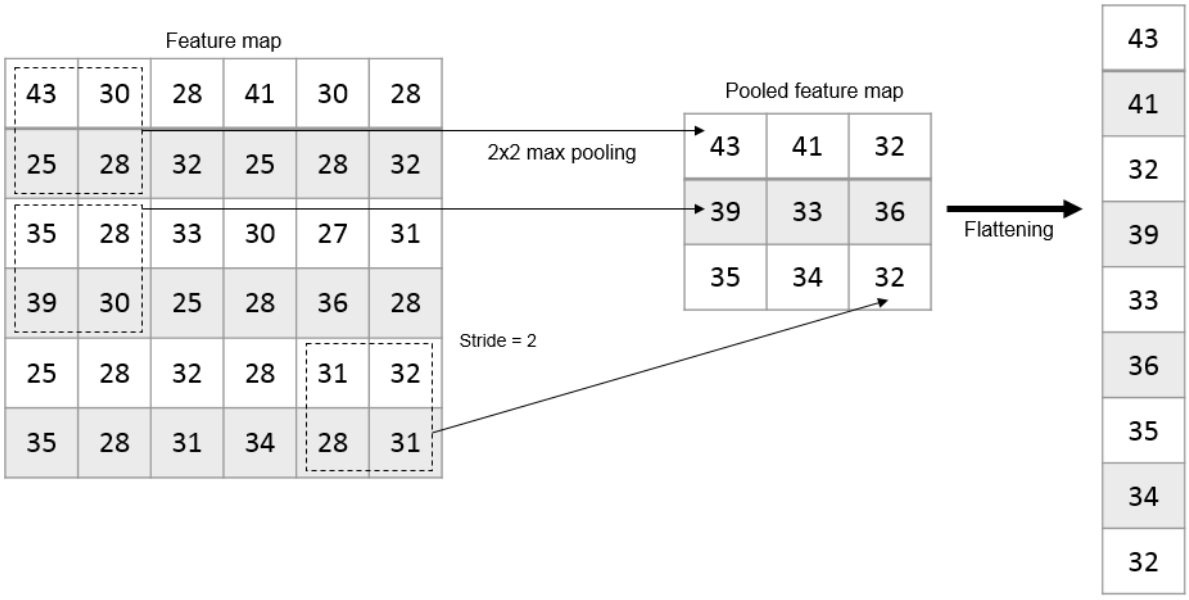


Figure 6: Pooling using the max pool method and a 2x2 array produces pooling of the maximum count amongst 4 connected elements (patch) to represent that data in the pooled feature map. Consecutive blocks of 2x2 elements means a stride of 2. The final pooled feature map immediately before input into the neural network can then be flattened from 2 dimensional data into a single dimension; this approach avoids the need for global pooling.

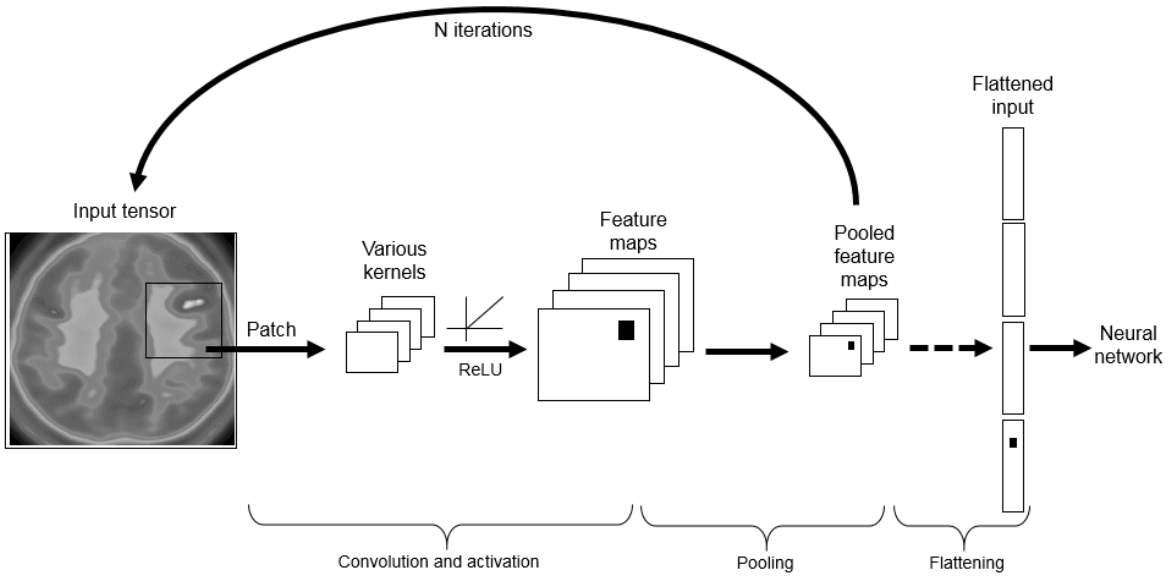
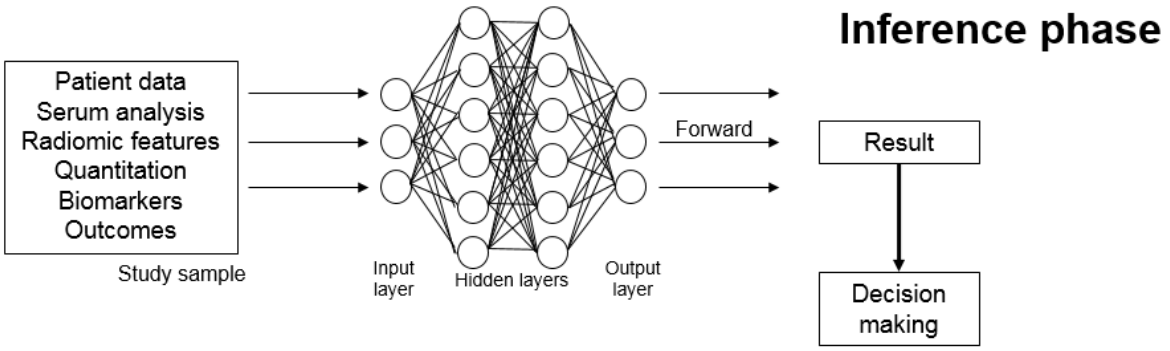
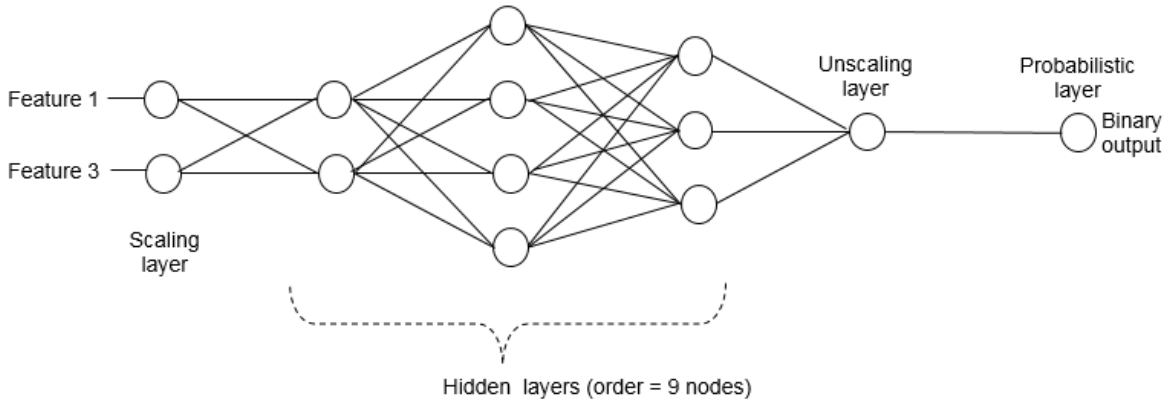


Figure 7: The CNN will have a number of convolution and pooling layers before flattening and input to the neural network. A number of kernels can be used on the same input tensor to produce layers of feature maps via the rectified linear units (ReLU) for pooling and eventually flattening.



Supplemental figure A: A trained and validated ANN can be used, with reasonably expected accuracy, to make inferences about clinical or research cases.



Supplemental figure B: Final neural network architecture convolved from figure 7 with only 2 inputs, 3 hidden layers of 2, 4 and 3 nodes respectively, an unscaling layer and a single binary output.