Machine learning with convolutional neural networks for clinical cardiologists

James Philip Howard ,[1] Darrel P Francis[2]

[1]National Heart and Lung Institute, Imperial College London, London, UK
[2]Cardiology, Imperial College London, London, UK

Correspondence to
Dr James Philip Howard, National Heart and Lung Institute, Imperial College London, London W12 0NN, UK; research@cardiologists.london

Published Online First
23 July 2021

To cite: Howard JP, Francis DP. *Heart* 2022;**108**:973–981.

INTRODUCTION

Machine learning (ML) is a revolution in computer science and is set to change the face of cardiology practice. In ML, humans no longer need to convert an understanding of a problem into a stepwise algorithmic solution; instead, the computer learns to solve a task for itself.

While ML can seem intimidating, the underlying principles build on familiar and established techniques. The recent revolution that made ML so effective, however, was the recognition that numerous sequential layers of simple arithmetic, termed neural networks, become surprisingly effective at solving difficult problems. This 'deep learning' has been startlingly effective across a variety of problems, and a particular type, the convolutional neural network (CNN) has revolutionised image analysis.

CNNs are inspired by the human visual cortex and have been used successfully in cardiology to process data that are one-dimensional (1D) (ECGs, pressure waveforms), two-dimensional (2D) (X-rays, MRIs) and three-dimensional (3D) (echocardiography videos, cardiac magnetic resonance cine videos and CT volumes). We are now entering the stage where these CNNs' performances are starting to equal that of cardiologists in some domains.[1 2]

In this review, we will cover the basics of ML, before explaining the workings of neural networks, and particularly CNNs. ML will play an increasing role in medical practice and, as with any diagnostic test or piece of medical equipment, an understanding of these systems will better equip medical staff to interpret these systems' results.

ML AT ITS MOST SIMPLE

The first chapter in an ML textbook is often made up of topics that a decade ago would have been called 'statistics'. A simple example that we commonly encounter in cardiology is the formula for predicting a patient's maximum heart rate during exercise: '220–age'. This formula arises from simple linear regression:

predicted maximum heart rate=constant–(another constant×age)

Originally, pairs of ages and heart rates were used to find the best values for these two constants. These were then rounded to 220 and 1, respectively.

As cardiologists we frequently encounter examples that are one step more complex, in the form of risk scores. In these, we no longer predict a measurement, but a probability of an event (eg, whether the patient will live or die in the next 6 months in the case of the GRACE score). In figure 1, we show an illustration (using simulated data) of the risk model of age and cholesterol predicting 10-year mortality. This has several interesting features. First, it is curved, because it must be; at age 40 years the risk of death is so low that even big differences in cholesterol result in only a small increase in death. At the other extreme, a 90-year-old patient's 10 risk of death is so high that their cholesterol has little bearing on it. It is for patients in the middle zone that cholesterol has the highest influence on mortality.

These familiar practices in clinical cardiology are at the simple end of the same spectrum of ML. We can start with a dataset of inputs (eg, age) and the corresponding outputs (maximum heart rate), and we are familiar with a computer calculating the line of best fit between them, which we call linear regression. This can be extended to outputs that are probabilities (eg, death), which we term logistic regression.

Learning objectives

- ⇒ Understand the paradigm of machine learning.
- ⇒ Understand the differences between supervised and unsupervised learning.
- ⇒ Understand the design principles behind convolutional neural networks, and why they excel at medical image analysis.

SUPERVISED AND UNSUPERVISED LEARNING

For a computer to fit one of these regression models for us, we must supply it with example pairs comprising the input (eg, age) and the output we are trying to predict (eg, maximum heart rate). This process is termed 'supervised learning', because the data we are providing include the correct answers.

There is another type of task that ML can perform, called 'unsupervised learning', where there is no specific true answer, but instead we ask the computer to discover patterns in the data. For example, it could be finding phenotypic variants within a large group of patients who superficially appear to have one condition.[3]

Both supervised and unsupervised learning have undergone a revolution in recent years, for two reasons. First, algorithms have been developed to tune even models that are millions of times more complex than conventional regression models.[4] Second, computers have become fast enough to perform the tuning steps which are vastly increased not only in number but also in subtlety (because the interactions between the different elements are unimaginably more numerous).[5]

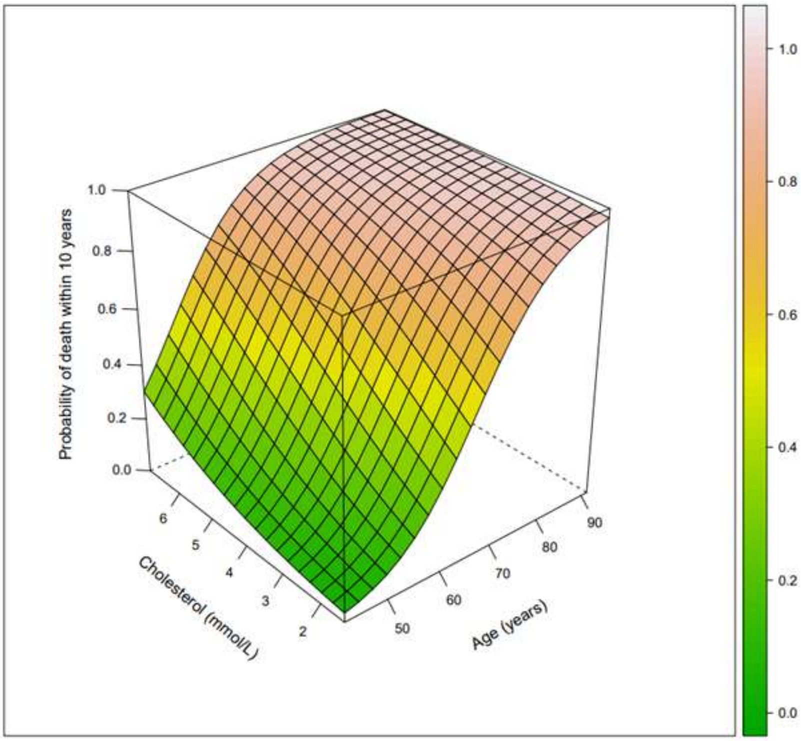

Figure 1 The relationship between age, cholesterol and mortality (probability of death) using simulated data.

NEURAL NETWORKS

Neural networks are assembled in layers in a similar manner to the animal brain.[6] Each layer comprises 'neurons', which, like biological neurons, receive inputs from other neurons and combine these inputs in some way. Just as a biological neuron's output is its frequency of firing, these computational neurons output a number.

In a sense, a conventional statistical regression model can be reimagined as a neural network (figure 2), where the inputs are connected to the output via stimulatory or inhibitory synapses. The output value could be interpreted as a value like a maximum heart rate, or a probability, for example, of death.

Neural networks however can be more complicated by adding extra 'hidden layers' of neurons between the input and the outputs, allowing us to model more complex relationships than simple straight-line dependencies (eg, heart rate=220–age). While each intermediate step may be simple (such as addition or multiplication), by combining several one after the other, the network can produce surprisingly sophisticated processing.

In the neural network depicted in the right panel of figure 2, there are two hidden layers. Each neuron in the first hidden layer receives only the raw data and can therefore only compute simple linear functions of them, such as 220–age, or (haemoglobin×7)+(troponin÷42). In the second layer, however, each neuron has access to the outputs of the first layer, and therefore can compute more complex relationships by combining them. Modern neural networks can have dozens of layers.

We store the strength of each of the synapses in a network as a number, which we term a 'weight'. A positive weight can be thought of as a stimulatory synapse, and a negative weight as an inhibitory synapse. By modelling the network this way, we can perform a series of three simple arithmetic operations (one for each 'layer' of the network) to translate the five input numbers into an output number.

When we first create one of these networks, the weights are selected randomly, and so the initial outputs are meaningless. However, we then go through the process of 'training'. During training we repeatedly show the network examples of the data, along with the correct answers (supervised learning). We then compare the network's output with the correct answer, and adjust the weights in a way that would have yielded a better (more correct) output. Through this process, the network eventually 'learns' the best way of processing the input data.

At first it seems spectacularly unlikely that an automated process could make millions of small adjustments that result in a meaningful neural network evolving. However, that was the stunning insight behind 'gradient descent', the mathematical process of adjusting these weights.[4]

ADVANTAGES OF NEURAL NETWORKS OVER CLASSIC REGRESSION

In the above example, it might not be clear why an approach using a neural network is any more advantageous than fitting a regression model where we try to predict the odds of dying. In fact, if we just connected the input neurons directly to the output neurons with adjustable weights (ie, had no hidden layers), our approach would be identical to regression. However, the inclusion of the hidden layers allows the neural network to perform 'deeper' processing of the data, in two specific ways:

- The network can create complex *non-linearity* by assembling a pipeline of interconnected neurons.
- Signals from different sources can interact at *multiple stages*.

Although these two features seem simple, their combination is extraordinarily powerful, as explained below.

ORIGIN AND UTILITY OF NON-LINEARITY

After each neuron, and before the next, we use a use a simple mathematical function to transform the result. A simple function termed the 'rectified linear unit' (ReLU) is surprisingly effective,[7] though there exist many alternatives such as the 'sigmoid' function. ReLU merely changes negative numbers to 0, while letting positive numbers pass through unaffected (figure 3, left panel). The reason why we need such functions may not be immediately apparent—indeed, we could theoretically just attach the neurons together without activation functions in between them. However, it is these non-linear functions, like ReLU, that allow the neurons to make 'decisions', effectively acting like filters that

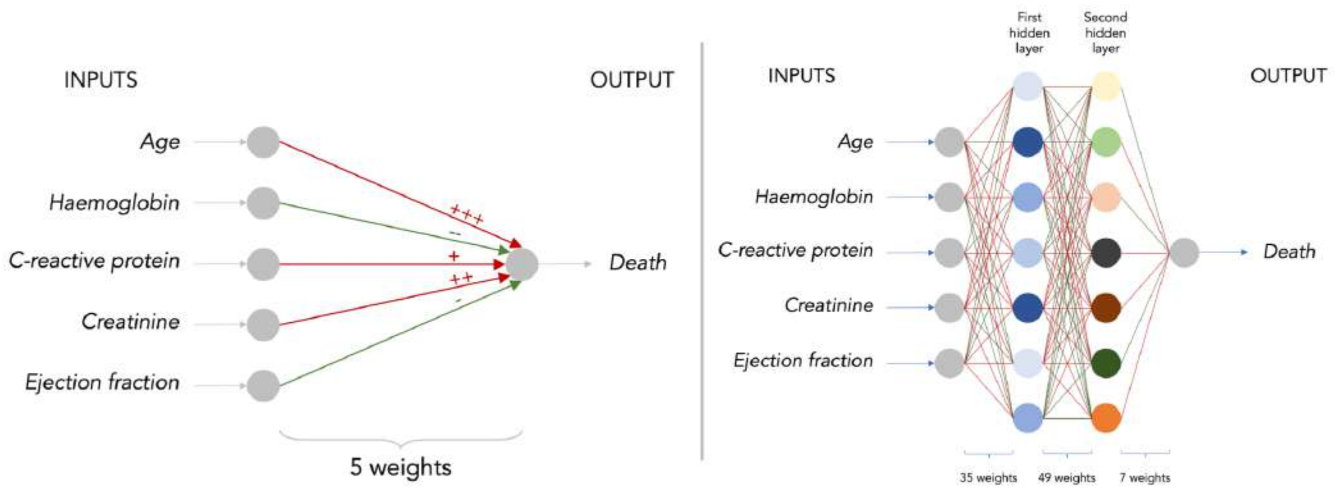

Figure 2 The left panel shows the simplest form of neural network, with five input neurons, corresponding to five input variables, which feed into one output neuron, corresponding to the chance of death. The input neurons either increase (age, C reactive protein, creatinine) or decrease (haemoglobin, ejection fraction) the activity of the output neuron, whose final value can be measured to provide a probability. The connections (weights) between the input neurons and the output neuron are adjusted to give the most accurate answer. The right panel shows a neural network with two 'hidden' layers. This provides extra processing power, because intermediate calculations can be processed further in a non-linear manner.

cause the neurons to behave differently in different settings. And so, when these neurons and their associated ReLUs are cascaded in large numbers over many layers, they can effectively emulate many complex mathematical functions.

For example, by combining just two neurons in a layer between an input and an output, with a ReLU immediately following each neuron, we can mimic complex functions such as converting a number to its absolute value (removing any negative signs; figure 3, right panel, yellow graph) or approximate something close to a sigmoid function (figure 3, right panel, blue graph).

MORE LAYERS ALLOW INPUTS TO BE COMBINED AT DIFFERENT LEVELS

In conventional algebra, a function z can depend on x and y in various ways. There is a family of relationships within which x and y do not interact, namely, $z(x, y)=f(x)+g(y)$. In other words, x and y are processed separately and the results are added at the end. An example of this is the Glasgow Coma Scale, where the responses of the eyes, motor system and voice are assessed separately and summed.

There is another family of relationships in which x and y are combined by an initial linear function, and then acted on by a non-linear function. For

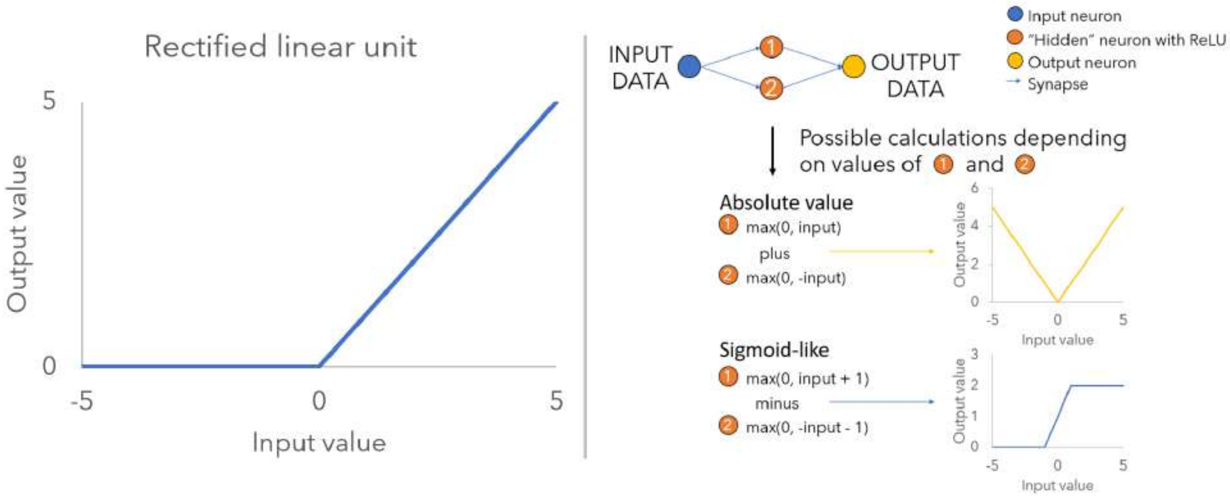

Figure 3 . The left panel shows the rectified linear unit (ReLU) activation function. Values below 0 are set to 0; positive values are unchanged. The right panel shows a schematic of a simple neural network with two hidden units between an input and output neuron. The hidden units each comprise a neuron and a ReLU. These take a maximum of 0 and a certain value, depicted as 'max(0, <value>)'. With just two neurons, we can create a small network which calculates the absolute value of an input value (yellow graph), or something which approximates a sigmoid function (blue graph).

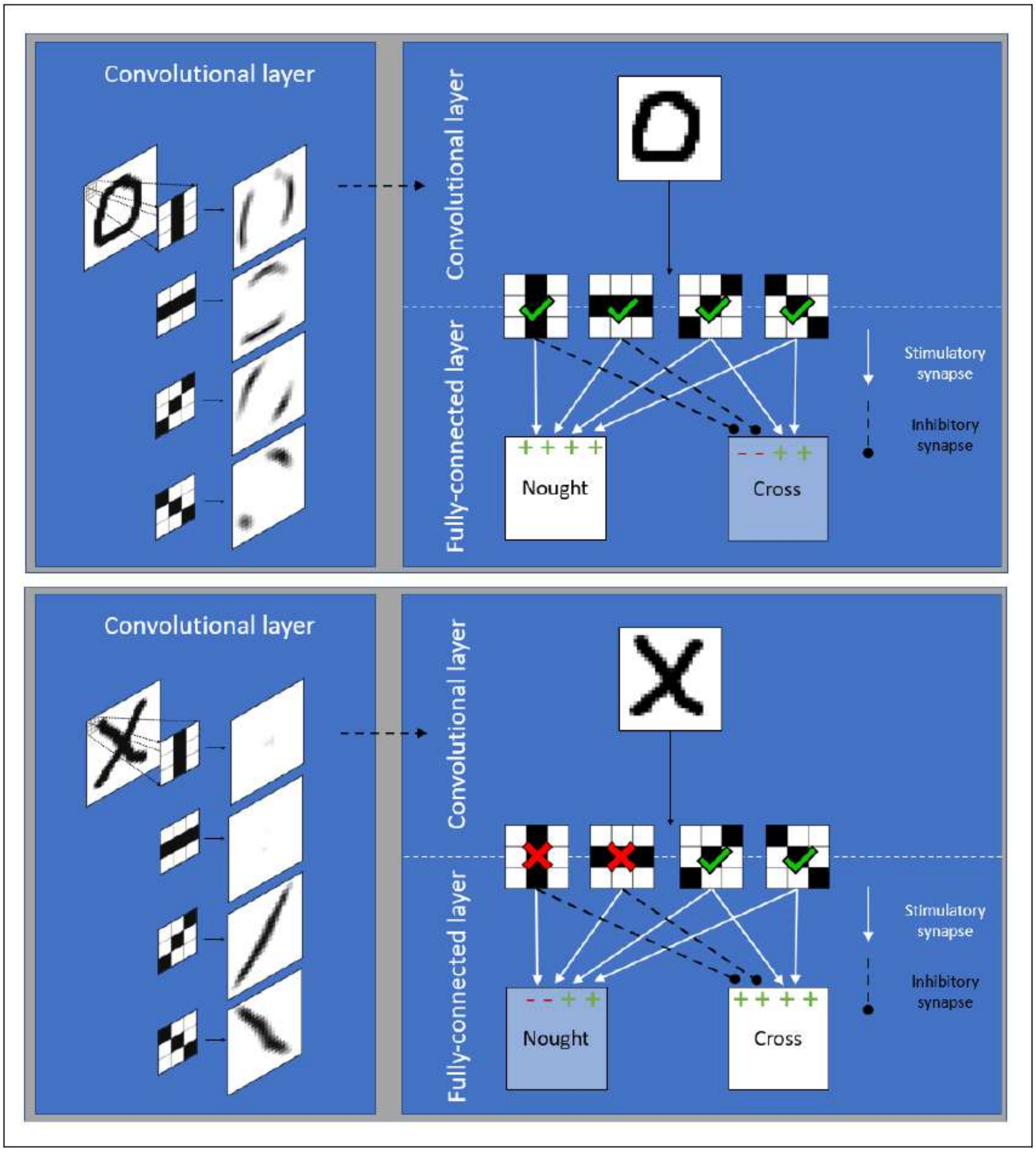

Figure 4 A schematic of a two-layer neural network designed to identify whether an image is of a nought or a cross. In the top row, we can see the image of the nought contains areas which match each of the four kernels in the convolutional layer. The cross image (bottom row), however, does not contain any features that match the vertical or horizontal kernels. The fully connected layer allows the strength of these detections to be translated into predictions; the weights connecting the vertical and horizontal kernel matches to the 'nought' prediction are stimulatory, but they are inhibitory for the 'cross' class.

example, calculation of body mass index (BMI) from imperial units:

$$BMI\ (stones,\ pounds,\ feet,\ inches) = \frac{(stones14+pounds)0.453}{((feet12+inches)0.025)^2}$$

Note that in this formula, the stone and pounds interact together linearly, and the feet and inches interact together linearly. The two results then interact non-linearly, and there are no other interactions. This process could *only* be modelled using multiple layers.

The family of functions we can model with numerous layers is vastly richer than the above, and high performing 'deep learning' networks typically have dozens of layers.

SIMPLE NEURAL NETWORK IN CARDIOLOGY, AND THEIR LIMITATIONS

While simple neural networks like those above have been used in multiple settings in cardiology,[8 9] they have not been adopted as substantially as some other network designs, especially in the field of automated medical image processing.

A major reason for their limited application is their poor 'scaling'; as the number of inputs into the network increases, the number of weights the network must store increases dramatically. One high profile paper used a neural network with 48 layers to process images of skin lesions and showed dermatologist-level performance in identifying cancers.[10] The images they fed into their

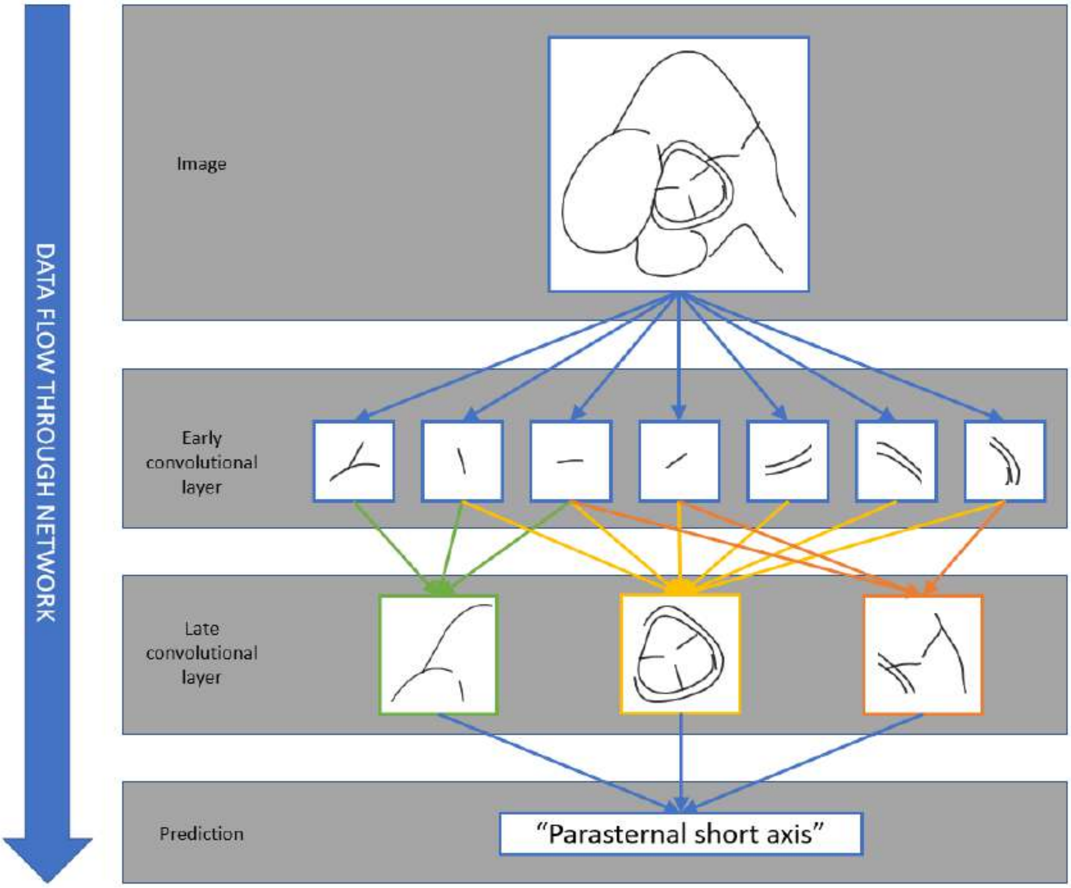

Figure 5 A schematic of a neural network used to identify echocardiographic views. Early convolutional layers in this network identify basic features such as edges, while later layers combine these features to identify anatomical structures. The presence or absence of these structures leads to a final prediction by the network.

neural network were 299 pixels tall, and 299 pixels wide, meaning 59 800 pixels total. To feed such an image into a neural network like those above, we would therefore need 59 800 input neurons. Just to connect this layer to the first hidden layer of 100 neurons would require almost 6 million weights, requiring large amounts of training data and time.

Furthermore, if that picture is shifted rightwards by one pixel, the inputs to every neuron in the network will change, and the network will no longer be able to recognise the image.

CONVOLUTIONAL NEURAL NETWORKS

The insight that catapulted deep learning into the forefront of image analysis was that the same visual feature might appear in any one of hundreds of positions on a large image, and should ideally be recognisable by the network, regardless of the position. To achieve this, why not have the neural network view small parts of the image in turn? That way, the network learns to recognise important features, rather than their arbitrary positions.

This is efficient, because, even if the image is tens of thousands of pixels in size, a network could be set to view many areas, each of which is much smaller, thereby needing many fewer weights.

It was this approach that the skin lesion study authors successfully used: the convolutional neural network (CNN).[6]

INSPIRATION FROM THE MAMMALIAN BRAIN

In the 1960s, Hubel and Wiesel found specific cells in the first layer of a cat's visual cortex which depolarised when the cat was viewing bright lines of a certain orientation.[11] Other neighbouring cells depolarised when lines of different orientations were present. Damage to these fundamental cortical layers leads to complete blindness. However, damage to deeper occipital lobe structures can lead to specific defects in higher level image processing, such as prosopagnosia (the inability to recognise faces).[12]

These findings demonstrate the fundamental workings of both the mammalian visual system and CNNs: the identification of an object involves earlier layers identifying the basic visual features present, and later (deeper) layers combining these features to make a final decision.

CNNS LEARN BY MATCHING TEMPLATES

The way CNNs work is simple: template matching. While classic neural networks learn how to process data from individual pixels, CNNs instead slide (or 'convolve') a series of small templates, termed

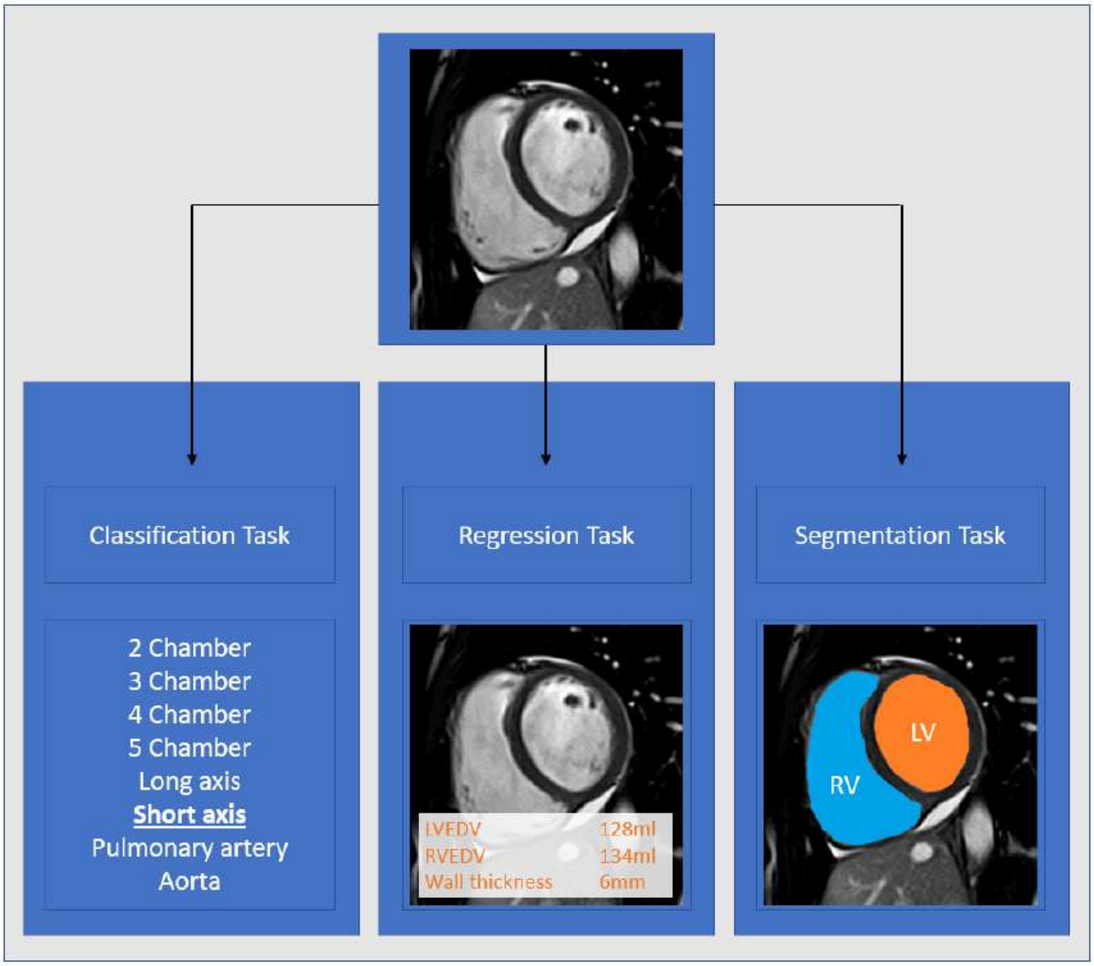

Figure 6 CNNs can be used for classification, regression and segmentation problems. Classification problems may involve deciding which class an image corresponds to (left panel). Regression tasks involve predicting one or more continuous values from an image, which could be used, for example, to predict cavity volumes. In segmentation tasks, the neural network may classify each pixel within an image as one of several classes (right panel). CNNs, convolutional neural networks; LV, left ventricle; LVEDV, left ventricular end diastolic volume; RV, right ventricle; RVEDV, right ventricular end diastolic volume.

'kernels', through an image and record how well each area matches.

Figure 4 shows an example of CNN which aims to identify whether an image is a nought or a cross. It comprises only two layers, though, in practice, CNNs have many more.

In the first layer, the example network is a 'convolutional layer' which includes four small templates, termed 'kernels'. Each kernel is simply a small grid of numbers containing a range of values. In figure 4, we can see four different 3×3 kernels, with white and dark squares inside them, corresponding to high and low values, respectively. The examples show a vertical line kernel, a horizontal line kernel and two diagonal line kernels.

When we feed an image into the network in figure 4, the images enter the convolutional layer. Then, each kernel in that layer slides, or 'convolves', through the image and records how well each area matches. Through each kernel, therefore, a new image is created, termed a 'feature map', which indicates how strongly each area of the original image matches that specific kernel.

When we feed an image of a handwritten circle (nought) into the convolutional layer (figure 4, top row), there are different areas of the image which match each specific kernel; for example, the sides of the circle match the vertical kernel, whereas the top matches the horizontal kernel.

However, when we pass an image of a handwritten cross into the convolutional layer (figure 4, bottom row), we find a different pattern: we find no matches for either the horizontal or vertical kernels (the feature maps are empty), but very strong matches for the diagonal kernels.

The second layer in the example network is the 'fully connected' layer. This layer is fundamentally identical to the hidden layers in the simple neural networks discussed previously (figure 2), in that it is merely a layer where every neuron in the layer before (the convolutional layer) is connected to every neuron in the layer following (the output neurons). The fully connected layer therefore takes the results of the final convolutional layer in a network and translates this into a prediction. In our example, an image containing many matches

Table 1 Examples of the successful application of neural networks, with classification, segmentation and regression examples using different neural network architectures from across different fields within cardiology

Task	Type of problem	Type of network(s)
ECG		
Accessory pathway localisation	Classification	Classic neural network[8]
Arrhythmia detection and classification	Classification	1D CNN[2]
ECG segmentation	Segmentation	1D CNN[26]
Echocardiography		
Cardiomyopathy diagnosis from images	Classification	2D CNN[19]
View classification from videos	Classification	3D CNN[27]
Image quality assessment	Regression	2D CNN[28]
Left ventricular volume and ejection fraction calculation	Regression	CN CNN[18]
Echo chamber segmentation	Segmentation	2D CNN[19]
Cardiac MRI		
Diagnosis of hypertrophic cardiomyopathy	Classification	3D CNN[29]
Left ventricular segmentation	Segmentation	2D CNN[1]
Whole heart segmentation	Segmentation	2D CNN[30]
Cardiac CT		
Carotid artery bifurcation detection	Classification	Classic neural network[9]
Agatson score calculation	Regression	3D CNN[31]
Epicardial adipose tissue quantification	Segmentation	2D CNN[20]
Coronary calcium quantification	Segmentation	2D CNN[21]
X-ray		
Diagnoses form chest X-rays	Classification	2D CNN[14]
Pacemaker model identification	Classification	2D CNN[22]
BNP estimation from chest X-ray	Regression	2D CNN[32]
Other		
Identification of pressure damping in coronary angiography	Classification	1D CNN[15]
Identifying obstructive coronary artery disease in MPS	Classification	2D CNN[33]
Identifying heart failure from cardiac biopsy	Classification	2D CNN[34]

BNP, B-type natriuretic peptide; CNN, convolutional neural network; 1D, one-dimensional; 2D, two-dimensional; 3D, three-dimensional; MPS, myocardial perfusion scan.

from all four kernels is likely to be a nought. In contrast, an image containing only matches for the diagonal kernels is likely to be a cross. This has been 'learnt' by the network by adjusting the weights of the synapses between the vertical and horizontal kernels to be inhibitors for the cross class and stimulating for the nought class.

In practice, CNNs typically include multiple convolutional layers, with deeper layers convolving kernels through the feature maps produced by the preceding layers to identify more and more complex images. For example, a CNN trained to identify echocardiogram views will use its early layers to identify relatively simple features within an image (figure 5). Later layers combine these observed features to identify anatomical features which the neural network can use to decide which echocardiogram view is shown.

During training, the CNN will learn the optimal weights in the fully connected layer, but importantly it also learns the optimal kernels in the convolutional layers. This means that these small templates are not programmed manually, but rather naturally develop during the training process. This again is analogous to the mammalian brain; Hubel and Wiesel found that kittens which are raised in a world consisting of either solely vertical lines or solely horizontal lines end up being completely blind to objects of the opposite orientation when they are introduced into the real world,[13] presumably because they had not developed systems for recognising them.

ADVANTAGES OF CNNS OVER CLASSIC NEURAL NETWORKS

In the above example, the advantages of CNNs may not be clear. For example, the image could have merely been fed into a classic neural network, like that in figure 2. However, CNNs offer several advantages:

1. CNNs are efficient. In our above example, if we store each kernel as nine numbers (a 3×3 grid), the entire convolutional layer can be represented using 4×9=36 weights. We then require just eight further weights to connect each of these four kernels to the two output classes, making a total of 44 weights. In contrast, if we used a single-layer classic neural network as a network to process low resolution images of 28×28 pixels, we would need 28×28×2=1568 weights. Generally, networks which can perform a task with fewer weights can be trained more quickly and with less data.
2. CNNs are (relatively) 'spatially invariant'. In our CNN example, we would expect the neural

network to work even if the size of the nought or cross in the image changed; a cross confined to the left half of the image would still show strong matches for the diagonal kernels and little matches for the vertical and horizontal kernels.

3. A CNN trained for one task is a good starting point for beginning training for another task. The early layers of a CNN trained to perform one task can be reused in other tasks, skipping the need to relearn the optimal kernels. This dramatically reduces training time and the amount of data required.[10 14]

CNNS BEYOND IMAGES

Because the CNNs above deal with 2D images by sliding kernels across them in two dimensions (across the width and height of the image), we term them 2D CNNs. However, 1D and 3D CNNs also exist and have shown great promise in analysis of non-image data. The 1D CNNs analyse 1D data, which are very frequent in cardiology, and include electrocardiograms[2] and coronary pressure measurements.[15] The 3D CNNs analyse 3D data, such as volumetric CT data or videos.[16]

CNNS BEYOND CLASSIFICATION

We have seen how neural networks can be used in classification tasks, such as deciding whether a patient will die, whether an image is a nought or a cross, or what view is depicted in a cardiac MRI scan (figure 6, left panel). However, CNNs can be used for other tasks, including regression and segmentation tasks.

In a regression task, the output of a neural network is not a predicted class, but instead a continuous number, for example, predicting the percentage of emphysema from CT scans.[17] One group has successfully used regression CNNs to predict end systolic volumes, diastolic volumes and ejection fractions from over 2.6 million echocardiographic images.[18]

In a segmentation task, we want the neural network to provide us with an output image where each pixel is classified by what it contains. For example, a neural network might be trained to identify which pixels in an image correspond to the left and right ventricular cavities (figure 6, right panel). The use of CNNs for medical image segmentation is exceptionally popular and powerful, because by simply measuring the segmented regions, we can easily derive computer-generated measurements, for example, of chamber dimensions.[19] This process can be applied to 3D volumetric data, allowing the quantification of epicardial adipose tissue[20] and coronary calcium on even non-gated CT scans,[21] with evidence that such systems prove useful in cardiac risk prediction.

CONVOLUTIONAL NETWORKS IN CARDIOLOGY

Table 1 shows successful examples of neural networks across five different fields within cardiology, in a variety of problems involving classification, regression and segmentation. In many of these tasks, the performance of these systems is beginning to equal that of clinicians,[1 22] and they are now beginning to transition from ‘bench to bedside’ and be used at the point of care.[23]

OVERFITTING AND APPRAISING NEURAL NETWORKS

When we train a neural network, we hope it is learning things that will allow it to work well on new examples, rather than merely ‘memorising’ each example in its training dataset. This unhelpful memorising of examples is termed ‘overfitting’. We try to minimise overfitting through several techniques. A simple one is to use a very large quantity of training data, which (1) makes it more difficult for the network to remember them all, and (2) means any real-world data are more likely to be similar to an example it has seen before. Indeed, this is the reason why neural networks are notorious for requiring lots of training data—it is not because they are large quantities to learn from, but rather they need large quantities to not overfit. It is also the reason why the training data must be of high quality—even the occasional mistake will be learnt by the network, and can adversely affect subsequent performance to a remarkable extent.[24]

Because of overfitting, it is very important that we keep data aside that we never train the neural network on, so we can see how well it generalises to these new examples. Indeed, it is best practice to hold two sets of the data back from training: one which we can use to continually appraise the network’s progress during the development stages, and a separate ‘hold out’ set which we use to report the final performance. These two datasets are called the validation and the testing datasets, although

Key messages

- ⇒ Machine learning covers a wide range of methods where a computer learns to solve a task using example data.
- ⇒ The simplest algorithms use standard statistical methods, such as regression.
- ⇒ Neural networks comprise layers of simple elements turned neurons, but have remarkable processing power.
- ⇒ Convolutional neural networks (CNNs) are inspired by the mammalian visual cortex, and excel at processing image data.
- ⇒ CNNs are now state of the art in many medical imaging tasks involving classification, regression and segmentation problems.

CME credits for Education in Heart

Education in Heart articles are accredited for CME by various providers. To answer the accompanying multiple choice questions (MCQs) and obtain your credits, click on the 'Take the Test' link on the online version of the article. The MCQs are hosted on BMJ Learning. All users must complete a one-time registration on BMJ Learning and subsequently log in on every visit using their username and password to access modules and their CME record. Accreditation is only valid for 2 years from the date of publication. Printable CME certificates are available to users that achieve the minimum pass mark.

which way around they are named is surprisingly, and frustratingly, variable.[25]

CONCLUSIONS

Cardiologists will see ML playing important role in their practice in the coming decade. ML can often be viewed as a logical extension of surprisingly simple statistical techniques. Neural networks themselves are made up of remarkably simple neuronal units, but combining them in many layers provides exceptional processing power. CNNs, in particular, are likely to form the heart of many systems involving computer vision and are becoming increasingly integral to modern cardiac imaging. As with any tool, an understanding of the workings of these systems will better allow cardiologists to appreciate their roles, strengths and limitations.

Twitter James Philip Howard @DrJHoward and Darrel P Francis @profdfrancis

Contributors JPH and DPF conceived, drafted and revised the work and have given final approval for the version to the published.

Funding JPH is supported by the Wellcome Trust (212183/Z/18/Z).

Competing interests None declared.

Patient and public involvement Patients and/or the public were not involved in the design, or conduct, or reporting, or dissemination plans of this research.

Patient consent for publication Not required.

Provenance and peer review Commissioned; externally peer reviewed.

Author note References which include a * are considered to be key references.

ORCID iD
James Philip Howard http://orcid.org/0000-0002-9989-6331

REFERENCES

1 Bhuva A, Bai W, Lau C, *et al*. A multicenter, Scan-Rescan, human and machine learning CMR study to test generalizability and precision in imaging biomarker analysis. *Circ Cardiovasc Imaging* 2019;12:e009214.
2 Hannun AY, Rajpurkar P, Haghpanahi M, *et al*. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nat Med* 2019;25:65–9.
3 Ahmad T, Lund LH, Rao P, *et al*. Machine learning methods improve prognostication, identify clinically distinct phenotypes, and detect heterogeneity in response to therapy in a large cohort of heart failure patients. *J Am Heart Assoc* 2018;7:e008081.
4 Ruder S. An overview of gradient descent optimization algorithms. Available: http://arxiv.org/abs/1609.04747 [Accessed 20 Aug 2020].
5 Dally B. Efficient methods and hardware for deep learning 2017.
6 Lecun Y, Bottou L, Bengio Y. LeNet. *Proc IEEE* 1998.
7 InNair V, Hinton GE. Rectified linear units improve restricted Boltzmann machines. ICML 2010 - Proceedings, 27th International Conference on Machine Learning, 2010.
8 Huang D, Yamauchi K, Inden Y, *et al*. Use of an artificial neural network to localize accessory pathways of Wolff-Parkinson-White syndrome with 12-lead electrocardiogram. *Med Inform Internet Med* 2005;30:277–86.
9 Zheng Y, Liu D, Georgescu B. *3D deep learning for efficient and robust landmark detection in volumetric data*. Cham: Springer, 2015: 565–72.
10 Esteva A, Kuprel B, Novoa RA, *et al*. Dermatologist-level classification of skin cancer with deep neural networks. *Nature* 2017;542:115–8.
11 Hubel DH, Wiesel TN. Shape and arrangement of columns in cat's striate cortex. *J Physiol* 1963;165:559–68.
*12 Schiltz C, Sorger B, Caldara R, *et al*. Impaired face discrimination in acquired prosopagnosia is associated with abnormal response to individual faces in the right middle fusiform gyrus. *Cereb Cortex* 2006;16:574–86.
*13 HUBEL DH, WIESEL TN. Receptive fields of cells in striate cortex of very young, visually INEXPERIENCED kittens. *J Neurophysiol* 1963;26:994–1002.
14 Rajpurkar P, Irvin J, Zhu K. CheXNet: Radiologist-Level pneumonia detection on chest x-rays with deep learning, 2017. Available: https://arxiv.org/pdf/1711.05225.pdf [Accessed 27 Jun 2018].
15 Howard JP, Cook CM, van de Hoef TP, *et al*. Artificial Intelligence for Aortic Pressure Waveform Analysis During Coronary Angiography: Machine Learning for Patient Safety. *JACC Cardiovasc Interv* 2019;12:2093–101.
16 Tran D, Bourdev L, Fergus R. Learning spatiotemporal features with 3D Convolutional networks. Available: https://arxiv.org/pdf/1412.0767.pdf [Accessed 24 Apr 2019].
*17 Gonzalez Serrano G, Washko GR, San José Estépar R. Deep learning for biomarker regression: application to osteoporosis and emphysema on chest CT scans. In: Angelini ED, Landman BA, eds. *Medical imaging 2018: image processing*. 52. SPIE-Intl Soc Optical Eng, 2018.
*18 Ghorbani A, Ouyang D, Abid A, *et al*. Deep learning interpretation of echocardiograms. *NPJ Digit Med* 2020;3:10.
*19 Zhang J, Gajjala S, Agrawal P, *et al*. Fully automated echocardiogram interpretation in clinical practice. *Circulation* 2018;138:1623–35.
20 Commandeur F, Goeller M, Razipour A, *et al*. Fully automated CT quantification of epicardial adipose tissue by deep learning: a multicenter study. *Radiol Artif Intell* 2019;1:e190045.
21 Zeleznik R, Foldyna B, Eslami P, *et al*. Deep convolutional neural networks to predict cardiovascular risk from computed tomography. *Nat Commun* 2021;12:715.
22 Howard JP, Fisher L, Shun-Shin MJ, *et al*. Cardiac rhythm device identification using neural networks. *JACC Clin Electrophysiol* 2019;5:576–86.
23 Circle Cardiovascular Imaging. Cardiac MRI and CT Software – Circle Cardiovascular Imaging - Deep Learning. Available: https://www.circlecvi.com/cvi42/cardiac-mri/deep-learning/ [Accessed 29 Nov 2020].
24 Hekler A, Kather JN, Krieghoff-Henning E, *et al*. Effects of label noise on deep Learning-Based skin cancer classification. *Front Med* 2020;7:177.
25 Faes L, Liu X, Wagner SK, *et al*. A clinician's guide to artificial intelligence: how to critically appraise machine learning studies. *Transl Vis Sci Technol* 2020;9:7.
26 Sereda I, Alekseev S, Koneva A. Ecg segmentation by neural networks: errors and correction. Available: http://arxiv.org/abs/1812.10386 [Accessed 10 Jul 2019].
27 Howard JP, Tan J, Shun-Shin MJ, *et al*. Improving ultrasound video classification: an evaluation of novel deep learning methods in echocardiography. *J Med Artif Intell* 2020;3:4.
28 Abdi AH, Luong C, Tsang T. *Quality assessment of echocardiographic cine using recurrent neural networks: feasibility on five standard view planes*. Cham: Springer, 2017: 302–10.
29 Biffi C, Oktay O, Tarroni G. Learning interpretable anatomical features through deep generative models: application to cardiac remodeling. Available: http://arxiv.org/abs/1807.06843 [Accessed 10 Jul 2019].
30 Howard JP, Zaman S, Ragavan A, *et al*. Automated analysis and detection of abnormalities in transaxial anatomical cardiovascular magnetic resonance images: a proof of concept study with potential to optimize image acquisition. *Int J Cardiovasc Imaging* 2021;37:1–10.
31 Cano-Espinosa C, González G, Washko GR, *et al*. Automated Agatston score computation in non-ECG gated CT scans using deep learning. *Proc SPIE Int Soc Opt Eng* 2018;10574:105742K.
32 Seah JCY, Tang JSN, Kitchen A, *et al*. Chest radiographs in congestive heart failure: visualizing neural network learning. *Radiology* 2019;290:514–22.
33 Betancur J, Commandeur F, Motlagh M, *et al*. Deep learning for prediction of obstructive disease from fast myocardial perfusion SPECT: a multicenter study. *JACC Cardiovasc Imaging* 2018;11:1654–63.
34 Nirschl JJ, Janowczyk A, Peyster EG, *et al*. A deep-learning classifier identifies patients with clinical heart failure using whole-slide images of H&E tissue. *PLoS One* 2018;13:e0192726.