Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

Review Article

Deep learning and generative methods in cheminformatics and chemical biology: navigating small molecule space intelligently

Douglas B. Kell[1,2], Soumitra Samanta[1] and Neil Swainston[1]

[1]Department of Biochemistry and Systems Biology, Institute of Systems, Molecular and Integrative Biology, Faculty of Health and Life Sciences, University of Liverpool, Crown St, Liverpool L69 7ZB, U.K.; [2]Novo Nordisk Foundation Centre for Biosustainability, Technical University of Denmark, Building 220, Kemitorvet, 2800 Kgs. Lyngby, Denmark

Correspondence: Douglas B. Kell (dbk@liv.ac.uk or doukel@biosustain.dtu.dk)

The number of 'small' molecules that may be of interest to chemical biologists — chemical space — is enormous, but the fraction that have ever been made is tiny. Most strategies are discriminative, i.e. have involved 'forward' problems (have molecule, establish properties). However, we normally wish to solve the much harder generative or inverse problem (describe desired properties, find molecule). 'Deep' (machine) learning based on large-scale neural networks underpins technologies such as computer vision, natural language processing, driverless cars, and world-leading performance in games such as Go; it can also be applied to the solution of inverse problems in chemical biology. In particular, recent developments in deep learning admit the *in silico* generation of candidate molecular structures and the prediction of their properties, thereby allowing one to navigate (bio) chemical space intelligently. These methods are revolutionary but require an understanding of both (bio)chemistry and computer science to be exploited to best advantage. We give a high-level (non-mathematical) background to the deep learning revolution, and set out the crucial issue for chemical biology and informatics as a two-way mapping from the discrete nature of individual molecules to the continuous but high-dimensional latent representation that may best reflect chemical space. A variety of architectures can do this; we focus on a particular type known as variational autoencoders. We then provide some examples of recent successes of these kinds of approach, and a look towards the future.

Introduction

Much of chemical biology is involved with the study of the interactions between small molecules and biomacromolecules, along with any physiological consequences, usually with the aim of finding molecules that are in some senses 'better'. At a high level, this admits two strategies [1] (Figure 1A). The classical version of chemical genomics was data-driven or 'function first'; a small molecule was applied to the system of interest (e.g. a rodent inoculated with *Mycobacterium tuberculosis*) and it either worked (here to kill the bacteria) or it did not. No mechanistic understanding was required (though could later be sought). A major advantage was, after all, that the drug worked. Beyond the thought of trying a variety of molecules, no specific hypothesis was required. In a more modern version, a target (or, much more occasionally a set of targets) is sought, on the basis of a hypothesis, usually about the desirability of inhibiting said target, and typically on a purified protein *in vitro*. Following a terminology from genetics, the former is referred to as 'forward' chemical genomics, the latter as 'reverse' (Figure 1A). The nominal advantage of the reverse approach is that in theory one immediately has a mechanism. However, even this is illusory, as effective drugs normally have multiple targets [2], and the ability to bind to a target *in vitro* conveys little or nothing about its mechanisms, efficacy or toxicity *in vivo* [3], nor even if it can even reach the supposed target(s) (membrane transporters are normally involved [3–7]). Thus, the search for a molecule with desirable properties

Received: 2 October 2020
Revised: 11 November 2020
Accepted: 12 November 2020

Version of Record published:
8 December 2020

Biochemical Journal (2020) 477 4559–4580
https://doi.org/10.1042/BCJ20200781

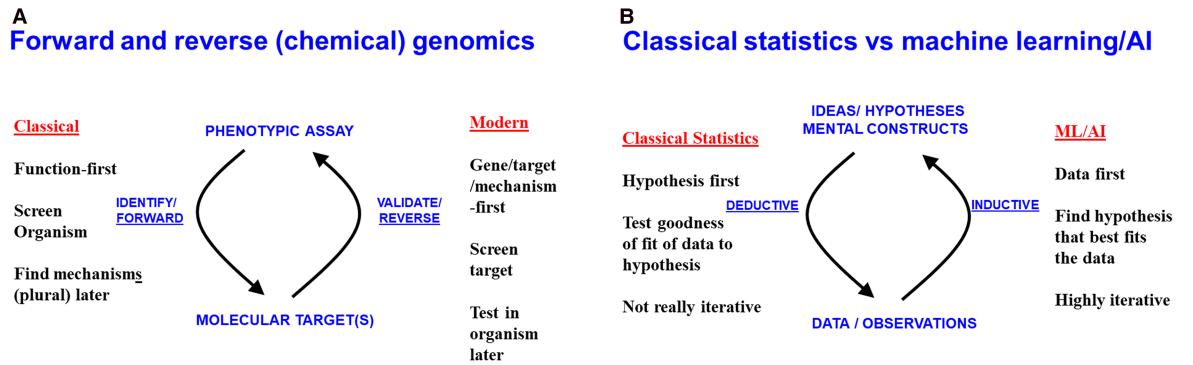

Figure 1. Two ways of relating paired attributes lead to separate strategies depending on the starting point.
(**A**) Forward and reverse chemical genomics. (**B**) Classical statistics vs machine learning. Note that Bayesian statistics is much closer in spirit to machine learning.

(however defined) typically involves a cyclic interplay of the type implied in Figure 1A. As with protein optimisation [8], it is arguably best seen as a navigation through a large search space of possible solutions [9].

Recently, there has been much excitement about the use of methods referred to as 'Machine Learning', 'Artificial Intelligence', or simply ML or AI. These too can serve to relate the world of ideas to the world of data. Perhaps surprisingly, their relationship to classical (Neyman-Pearson or frequentist) statistics [10] (Figure 1B), is similar to that between forward and reverse chemical genomics (Figure 1A).

Chemical space

The essential problem is that the number of small molecules of potential interest ('chemical space') is vast [11–13]. A widely quoted figure, based on simple calculations, albeit dependent on the number of C-atoms considered, is 10^{60} [14,15]. In contrast, the numbers of synthesised and purchasable molecules as recorded at the ZINC database [16] (http://zinc15.docking.org/) are just over 10^9 and 6.10^6, respectively (even most of the simple heterocyclic scaffolds have never been made [17,18]). Restricting the number of heavy atoms to just 17, including halogens as well as C, N, O and S, gives more than 10^{11} molecules [19,20]. This corresponds to an average molecular weight (MW) ∼250 [19], while MWs of 500 and 1000 imply 'Universes' (for even these restricted chemical spaces) of ca 10^{36} and $\sim10^{72}$, respectively [15]. An earlier list of 1387 marketed drugs [21] includes over 200 of them (some 15%) with MW exceeding 500 (Figure 2A), while a 2D mapping of ∼6 million ZINC compounds, 150 000natural products, ∼150 fluorophores, ∼1100 endogenous human metabolites (Recon2), and the same marketed drugs (based on [22]) is given in Figure 2B.

Recent advances in computational learning have changed the game of how we can understand and navigate this very large chemical spaces, from a focus on **discriminative**[1] methods, that are largely descriptive, to a suite of **generative** methods in which we develop and deploy the ability to create novel matter computationally and in principled ways. The purpose of this review is to describe these changes.

A brief history of virtual screening and the multilayer perceptron

Because of these very large numbers, that far exceed the capacity of even high-throughput screening assays, virtual screening (VS) has come to the fore. VS [28–31] refers to the use of computational techniques to explore a large compound library *in silico*, thereby to select a small subset of potentially interesting (here bio-active) molecules that may then be tested experimentally. It is useful to discriminate '**unsupervised**' from '**supervised**' learning methods, where in the former we know only the inputs, e.g. molecular structures, without knowing the outputs, e.g. activities of interest. Complementarily, semi-supervised or 'self-supervised' [32,33] methods leverage knowledge of the output values (or class labels) where they are known for a subset, while reinforcement leaning methods involve the use of machine learning methods to make decisions based on

[1]**Terms in boldface** are defined further in a glossary at the end

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

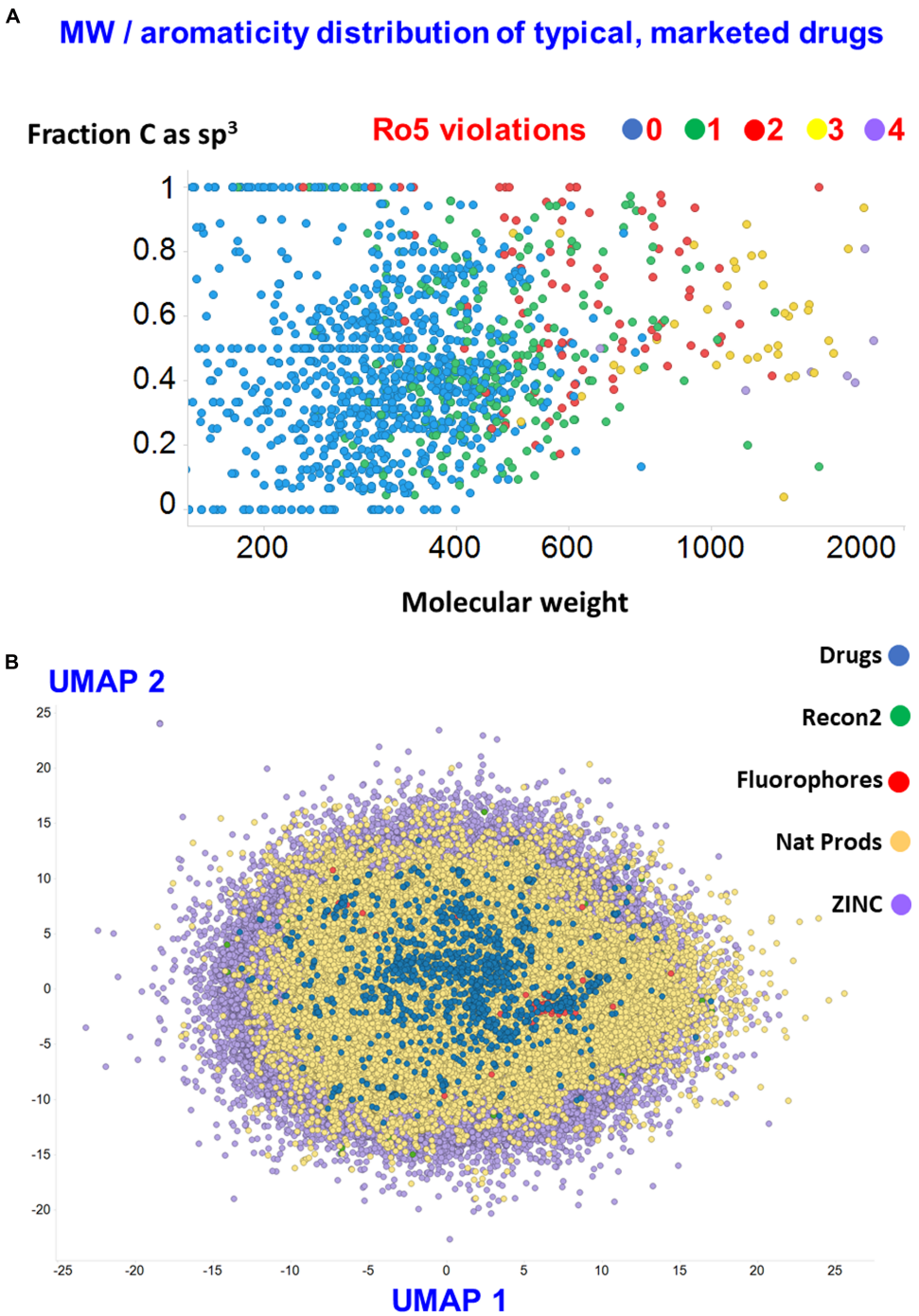

Figure 2. The areas of chemical space presently occupied by some 1387 marketed drugs.

(**A**) Fraction of sp^3 hybridisation (a measure of aromaticity) plotted vs molecular weight. The number of violations of the 'Rule of 5' [23] is also shown. (**B**) UMAP [24] representation of the chemical space of ~6 M 'druglike' molecules from ZINC. This is largely seen to contain the ~150 000 natural products, ~150 fluorophores, ~1100 endogenous human metabolites (Recon2) and 1387 marketed drugs studied previously [25]. Molecules were extracted by the present authors [26] to a latent space of 100 dimensions using methods described in [27] and their vector values in the latent space used as the input to the UMAP algorithm.

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

various kinds of reward. Although VS covers many areas, we shall focus here on Quantitative structure-activity relationships (QSAR) (e.g. [34–36]). QSAR describes a series of techniques for supervised learning, in which we use paired inputs (X) and outputs (Y) (here they are suitably encoded molecular structures and activities, respectively) to produce a model that given the former outputs the latter.

One such means of effecting this mapping is the multilayer perceptron (MLP), in the form of a fully interconnected feedforward artificial neural network [37] (Figure 3A). The MLP consists of nodes (circles) and weights (the lines joining them). Typically in chemical biology and QSAR analyses the inputs (X, just 15 are shown in Figure 3A) would be the values of molecular descriptors or the digits of a binary fingerprint encoding of molecular structure [38]. Outputs (Y) can either be classes or values to be found via a composite nonlinear/mapping function. In the example shown we have six classes. The weights are initialised to small values (typically this was done randomly from a normal distribution; nowadays it is done in a more principled way [39,40]), and a bias term introduced as shown (Figure 3A).

Training such an MLP consists of applying the inputs (usually normalised, commonly in the range 0.1 to 0.9 or 0 to 1 for each) and multiplying them by the relevant weight. A node (shown in the lower right-hand portion of Figure 3A) is a very simple processing unit: it sums the dot product of weights and inputs, then passes it through a transfer function. Classically (in the 1980s/1990s) the transfer function was differentiable, usually something like a hyperbolic tangent function (*tanh*), that scaled inputs to the range 0 to 1 and passed them to the following layer. In Figure 3A only one 'hidden' layer is shown before the output layer. In classification problems it is common to use a *Softmax* function at the output layer to ensure that the sum of the outputs is 1, thus producing a result vector that represents the probability distribution of the potential outcomes. Training the networks (finding the 'best' values for the weights) typically involves an algorithm ('backprop') that propagates a partial derivative of the normalised error between the desired and discovered outputs back through the MLP, the weights being adjusted up and down until the outputs have been learned (i.e. can be reproduced when the input is passed forward through the network). Training can be done in batches using subsets of the paired (X-input, Y-output) training data, and each full pass backwards and forwards through the training set is typically referred to as an epoch. Because such networks (of arbitrary size) can learn any arbitrary mapping (known as '**universal approximation**') they are very prone to **overtraining** (learning the training set but failing on new inputs), and a separate validation set is used to see when this occurs (so as to prevent it). Finally, a completely independent test set is used that has not been used in training at all. This avoids the (correct) criticism by frequentist statisticians that these methods are effectively testing 1000s of hypotheses in a desperate attempt to try and find one that fits [41].

The key concepts of any kind of learning of this type are that (i) such an MLP can provide sensible outputs on molecules it has never seen (this is referred to as 'generalisation'), and (ii) the 'knowledge' that it has 'learned' is effectively stored in the matrix of learned weights. In contrast with say a computer memory, this memory is not held in a single location but in the whole network. This idea (which is also adopted in living organisms) is known as 'associative' or 'content-addressable' memory, and is the main reason for the 'robustness' of structures such as MLPs; destroying some of the connections has little adverse effect on them (and may even improve them — see below). Because the MLP model has learned the general mapping of inputs to outputs by storing it in the matrix of weights, any other reasonable molecule can be applied to it and a suitable output will emanate. This permits any kind of virtual screening for which one has a molecular structure that can be encoded as an input and the potential activity can then be output. Note that here we still have to provide all the molecules as inputs.

While the numbers of inputs and outputs are fixed, there are a number of so-called **hyperparameters** of an MLP or similar neural network that can improve the speed and effectiveness of learning and generalisation. For the MLP these include the number and size of the hidden layers, the transfer function, the means of weight initialisation, the learning rate, and the so-called momentum (which adds a fraction of the previous weight update to the current one). In addition, one could usefully remove individual nodes or weights that did not seem to contribute to the model [42,43].

Although such MLPs could indeed be used for virtual screening (and many other purposes), they were very slow to train (**radial basis function networks** [44,45] (as in Figure 3B) were far quicker [46,47]), and it proved impossible to train large nets with even modest (>2) numbers of hidden layers. It is widely considered that this was simply due to the fact that the gradient fed back during the backpropagation step was increasingly small as the number of weights increased (the 'vanishing gradient' problem). In other circumstances the gradient could become unstable (the 'exploding gradient' problem). Actually, the success of RBF nets gave a strong hint that

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

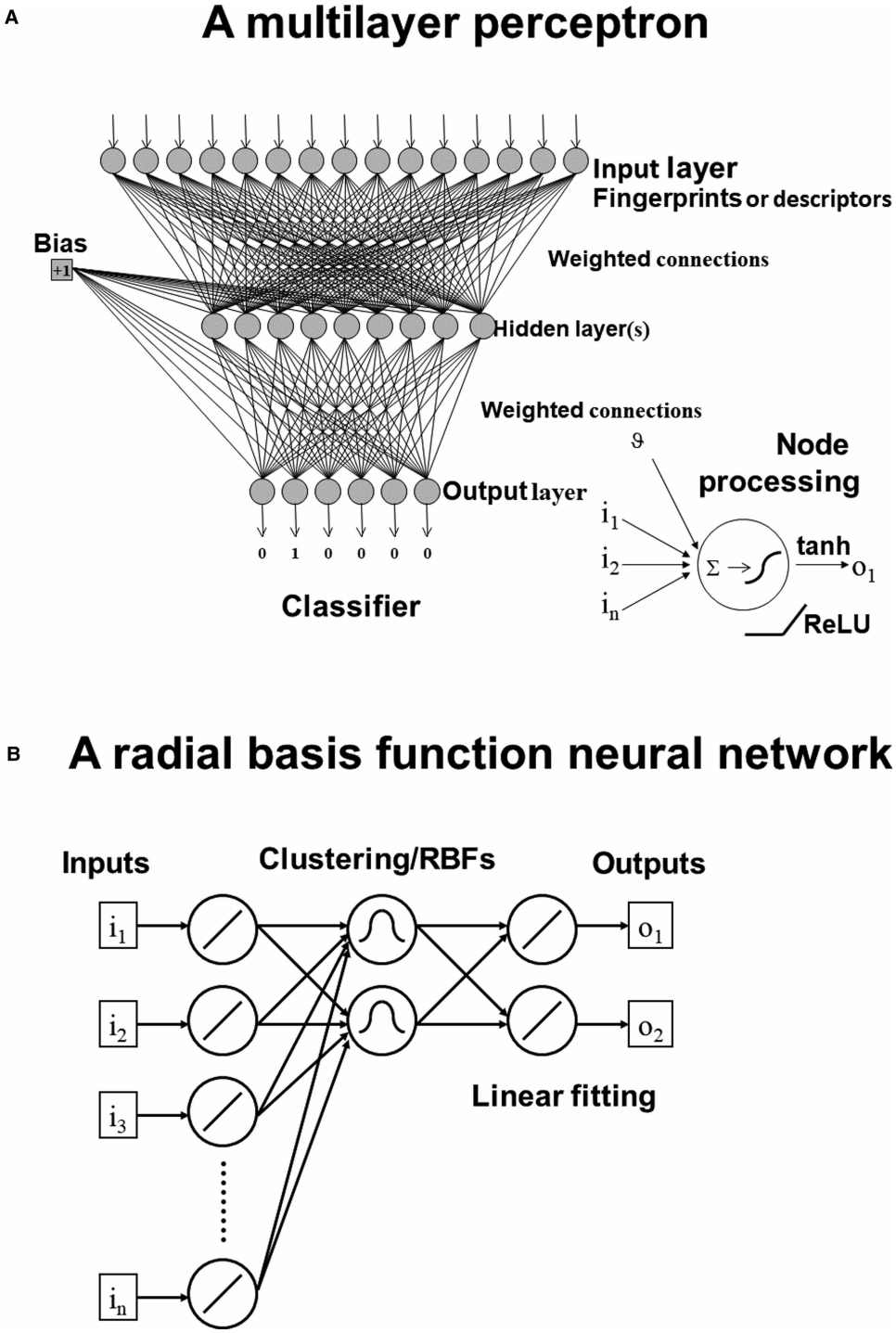

Figure 3. Early, 'shallow' neural networks.

(**A**) A fully interconnected feedforward network: the multilayer perceptron. In the lower right is shown the action of a processing unit as summing its inputs plus the bias ν then passing them through a transfer function to the next node. Usually this transfer function was a saturable, differentiable function such as tanh, but more recently 'rectified linear' nonlinearities have become popular. The example shown is a classifier, though outputs can be non-integer numbers and the ANN serve as a nonlinear regressor. (**B**) A radial basis function neural network. Here the first step is an unsupervised clustering of the input data (typically using K-means clustering) that sets the centres and widths of the radial basis functions according to the local density of points in the input space. A simple linear fitting then determines the outputs.

Biochemical Journal (2020) 477 4559–4580
https://doi.org/10.1042/BCJ20200781

an unsupervised step prior to (or as part of) the initial weight setting, especially if done layer by layer [48], could be very advantageous.

Another popular neural network architecture of the time was the self-organising feature map popularised by Teuvo Kohonen [49]. It bears some similarities to RBF networks in that it too uses unsupervised methods to cluster the input space prior to the training steps.

At all events, during the 1990s, because of the inability to train large and deep networks, the entire ANN field largely went into stasis save for the activities of a small number of pioneers.

The rise of deep learning in the 21st century

Obviously, this has all changed, since it is now entirely possible to train ANNs that are broadly similar to MLPs but that can have even hundreds of 'hidden' layers (e.g. [50]), totalling over a billion interconnections, which is why they are called 'deep networks'. These very large networks are the basis for the revolution known as 'deep learning' [51–53] that underpins widely recognised advances in image and speech recognition and the ability [54] to play board games at absolutely world-leading levels. At the time of writing (September 2020), probably the largest is GPT-3, containing as many as 170 billion weights [55]. Deep learning is now coming to the fore in drug and materials discovery [56–65]. As set out by Geoffrey Hinton, one of those neural network pioneers, the reasons for the original failure were essentially fourfold: (i) the labelled datasets were thousands of times too small; (ii) typical computers used were millions of times too slow; (iii) the weights were initialised in a naive way; (iv) the wrong types of transfer function (non-linearity) were used.

Thus, one major problem with the *tanh* transfer function (Figure 3A) is that it cannot extrapolate beyond the range on which it has been trained, whereas a rectified linear unit (ReLU) type of transfer function can [66]. This kind of transfer function (Figure 3A), that is zero for all inputs below and including zero and then linear with the sum of the inputs for positive values, is usually far more effective [67], and many others can work well too (e.g. [68,69]). The type of initialisation has also been much improved [39,40], especially using weight initialisations that are based on the size and position of the layer they are in. Finally, the advent of GPU and cloud-based computing has made very powerful computers much more widely available. With this has come the recognition that not only can deep nets 'store' more knowledge than can shallow nets, but that they require to be trained on many more input examples. Such large datasets are nowadays commonly available online; when they are not, a useful and effective trick of data augmentation is to add certain kinds of 'noise' to those that are [70,71]. This said, a variety of machine learning methods are perfectly capable of learning large datasets for 'discriminative' QSAR problems with more or less equal facility [72], and deep learning methods are likely to have only a marginal advantage [73]; a particularly detailed comparison has recently appeared [74].

Although the basic MLP architecture could be modified, it rarely was, and one of the other main features of the deep learning renaissance is the coming to prominence of a series of other architectures. We shall discuss four broad classes: convolutional (CNNs), recurrent (RNNs), long short-term memory (LSTMs), and auto-associative (AA) nets. This will lead us on the variational autoencoder that is our focus here.

Convolutional neural networks (CNNs, ConvNets)

The MLP and RBF architectures of Figure 3 used vectors as their inputs, and while it is possible to unfold a 2D matrix (such as an image) pixel by pixel, row-wise or column-wise, to form a vector, this would lose their natural spatial arrangement. (Note that colour pictures with three colour channels RGB actually form a tensor). Similarly, the number of weights (if one weight was used per pixel) would immediately become infeasibly large. Indeed a tiny 10×10 image matrix whose pixels could be just black or white ($\{0,1\}$) admits 2^{100} ($\sim 10^{30}$) variants, so some kind of feature extraction is always required [75]. Another of the pioneers of deep learning, Yann LeCun, long recognised the potential utility of ANNs in image recognition [76], and since their invention has been heavily involved in many of the developments of CNNs. Broadly (Figure 4A), a CNN uses a small matrix (e.g. 5×5) to range over the larger image, acting as a filter, and passing its output to the next layer, as in an MLP. Following this, the convolution layers are pooled. This strategy decreases both the number of weights and the likelihood of overtraining. Usually, several layers of convolution and pooling are applied, before a final, fully interconnected output layer. Otherwise, training is fairly conventional, using backprop to adjust the weights. They are widely used in image processing, e.g. in radiology [77], cytometry [78], and breast cancer screening [79]. In general, it is considered that the role of the various convolution and pooling layers is precisely to extract and combine features in abstract form (for image recognition tasks it captures the high-level

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

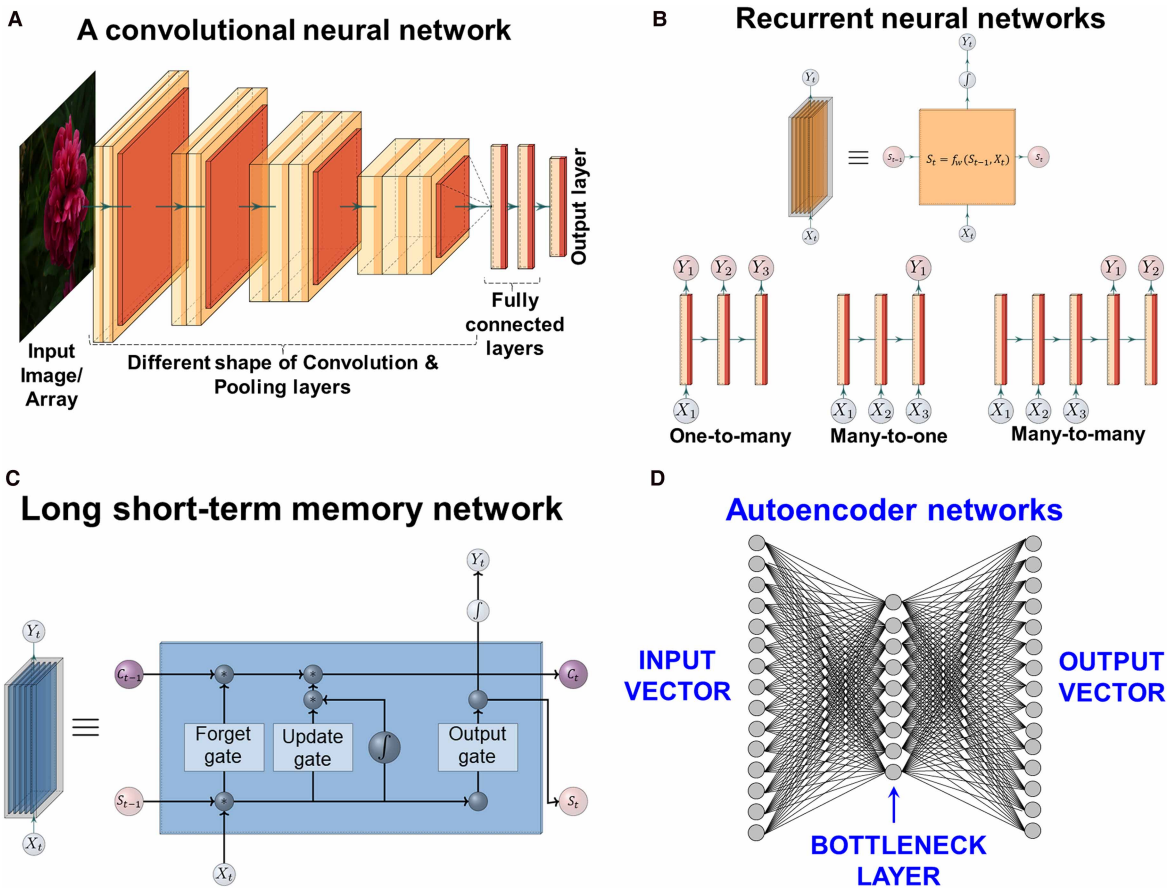

Figure 4. Networks with other architectures.

(**A**) Convolution neural networks (CNNs, ConvNets). (**B**) Different types of recurrent neural nets (RNNs) based on their input–output size. The upper row shows a general overview of an RNN unit with different components and the lower row shows the different types of RNNs. (**C**) Long short term nets (based on [83] and [84]), showing the architecture of the LSTM neuron and three gate units. The self-recurrent connection on the left indicates the feedback with a delay of one time step. c_{t-1} and c_t are the contents of the cell at time t−1 and t, while s_{t-1} and s_t represent the network state. For further details, see text and [83] and [84]. (**D**) Autoencoder net. This is a standard MLP without added bias. Its input and output vectors are of the same length and it contains a much smaller 'bottleneck' layer. Overall the network serves to output the closest inputs to those on which it has been trained, even when those inputs are then subjected to noise. The values of the outputs of the bottleneck layer form a latent representation of the inputs.

representation of colour, shape etc. in different layers). Some tricks and tips to train CNNs are at https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks.

Thus, AtomNet [80] used vectorised versions of 1 Å 3D grids placed over co-complexes of target proteins and small-molecules bound to them that had been sampled within the target's binding site to effect structure-based drug discovery. This said, a simpler representation of a molecule is as a graph (where atoms and bonds become nodes and edges), and there is an equivalent version of a convolutional network for these called, unsurprisingly, a graph convolutional network or GCN [81,82].

Recurrent neural nets (RNNs)

Thus far, we have looked only at simple feedforward networks, in which the output of neurons in a given layer acted as input only to neurons in the next layer. While CNNs are arguably the deep network of choice for image processing, many problems such as natural language processing use discrete words in specific sequences. Where the next character (word) is depends on the previous character (word) in a particular word (sentence); in a molecular generation task, the next SMILES character depends in part on the previous character. So

Biochemical Journal (2020) 477 4559–4580
https://doi.org/10.1042/BCJ20200781

instead of a simple input–output transfer function, we need a hidden state with which to try to capture the previous character's information. Here a different kind of architecture is preferred, involving connections between non-adjacent layers, to nodes in the same layer including themselves, and even backwards connections (hence recurrent). A very simple version is shown in Figure 4B. Clearly the number, nature and extent of the 'recurrent' connections can be varied considerably. There are different types of RNN to tackle different types of problem with respect to input and output size. These include one-to-many (one input to many outputs) for molecular property to SMILES string generation; many-to-one (multiple input to one output) for SMILES string to prediction of a particular property; and many-to-many (multiple inputs to multiple outputs) e.g. for SMILES character to different molecular properties prediction. The standard backpropagation algorithm has to be modified here, since in unfavourable cases the backpropagation of error could simply be short-circuited. In this case a variant known as 'backpropagation in time' is usually used. Even this does not always train well, due to gradient overflow and underflow. However, one specific type of RNN that has come to prominence makes use of a technique that goes back to 1997 [83] but was initially little exploited, known as long short-term memory networks.

Long short-term memory nets (LSTMs)

LSTMs [83] are a specific type of RNN, and arguably the presently most favoured solution for appropriate problems in deep learning, in which an extra kind of 'memory' is explicitly included that effectively helps the recurrent network 'remember' its own states from previous epochs during learning, which may include subsequences of textual data or previous patterns in time-course data. LSTMs contain special cells (memory cells) with a number of extra parameters that control the functioning of the 'gates' (shown in Figure 4C) in the memory cell c_j indicated. The multiplicative gate units open and close access to the flow of constant errors. These extra parameters, not all of which are used in every implementation, provide a very considerable richness of behaviour, whose mathematical intricacies we do not develop further here. They have considerable potential in drug discovery [84]. Recent reviews are at [85,86]. The amount of useful 'memory' an LSTM can effectively store is rather limited, and improving this is a particularly active area (see e.g. [87]). A similar RNN known as Gated Recurrent Units (GRUs) [88] has fewer parameters than does LSTM; it has been used in [27] for novel molecule generation.

Autoencoder (AE) nets

The last network architecture of immediate interest to us here (Figure 4D) is the autoencoder. On the face of it this is just a simple MLP with one hidden layer whose input and output vectors have the same length, and where it is intended that the weights are evolved to make the output as close to the input as possible. Crucially, the hidden layer is a 'bottleneck' layer; what this does is effectively to force the network to abstract the core signal from the input and this makes it resilient to noise in the inputs. The values of the outputs of the bottleneck layer thus provide a representation of the input of a much lower dimensionality, effectively compressing the data, and these may indeed be used for feature representation, clustering, similarity assessment, or data visualisation (somewhat as in Figure 2). This is effectively a kind of principal components analysis (PCA) when there is only one layer with a linear transfer function. Training is normally via standard backpropagation, as in MLPs.

Variational autoencoders (VAEs) and generative methods

Closely related to autoencoder nets in their appearance (but not at all in their underlying mathematics) are variational autoencoders (Figure 5A) [89–91]. Much of the previous discussion, based implicitly on QSARs, was about what are referred to as discriminative models, where one seeks to learn a predictor from observations (of paired molecules and their activities). What we would really like to have is the ability to generate the molecules themselves '*de novo*' (e.g. [59,64,65,84,92–107]), by learning what amounts to a joint distribution over all the variables (both inputs and outputs). To this end, a generative model seeks to simulate or recreate how the data are generated 'in the real world'. Generative models can be trained on existing data and used to generate novel text, images and even music. For our purposes, a generative model can learn to generate molecules that it has never 'seen' during the training phase. Of course this means in principle that we can go from the comparatively modest number of molecules that have ever been made and contemplate sampling (generating) them from the much more massive universe (described above) of molecules that might be made. A VAE (Figure 5) consists of two main halves: an encoder (sometimes called a recognition model), and the decoder (which is the generative model). Joining them is a vector (which may be, but does not have to be, of lower dimensionality) that represents the latent space between the two (Figure 5A). A VAE effectively learns stochastic mappings between an

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

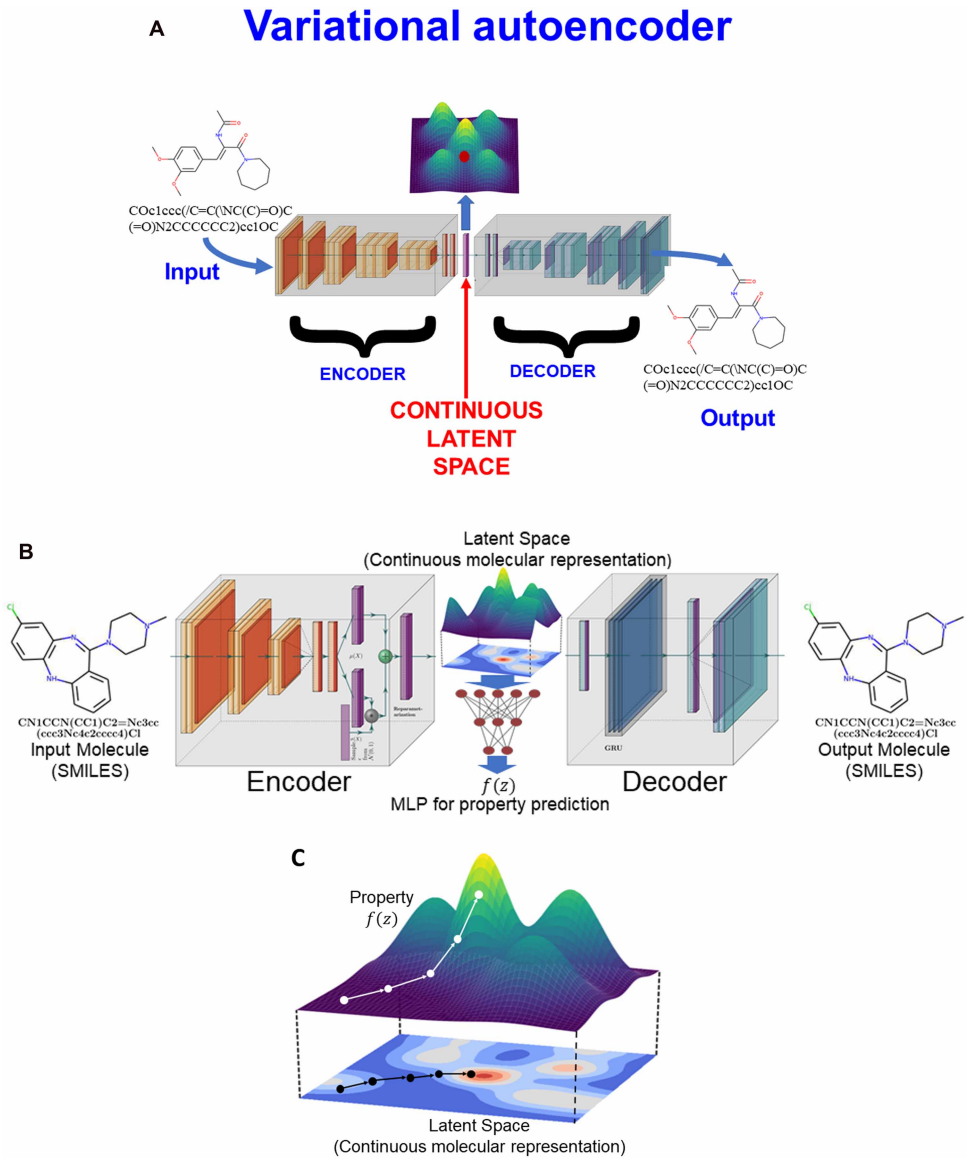

Figure 5. Variational autoencoder networks and their uses.

(**A**) Basic VAE architecture, showing the latent space. (**B**) VAE as proposed by Gómez-Bombarelli and colleagues [27]. The latent space is shown as a 2D space for ease of visualisation, but in the paper had a dimensionality of either 156 or (more commonly) 196. (**C**) Moving around the latent space, one simultaneously comes into the 'basin of attraction' of particular molecules, whose structures may be output and properties may be calculated via the MLP shown in (**A**) and described in the text (based on [27]). Using optimisation strategies such as evolutionary algorithms can guide the search for the properties and hence the 'novel' molecules.

observed (input and output) space, whose distribution is provided by the real world, and a latent space that is purposely much simpler and is continuous. Deep learning VAEs are those that are trained using general neural network principles and have multiple 'hidden' layers for both the encoding and the decoding. The particular recognition here is that we need to move between the discrete space of molecules (often encoded as SMILES strings [108], but increasingly as molecular graphs) and the continuous space of the neural networks and latent variables of the autoencoders. Fortunately this is now possible, using techniques such as molecular grammars [109,110], and direct graph generation [111–113]. We note that SMILES strings present particular difficulties because their grammar is context-sensitive: making a cut in an arbitrary place in a SMILES string (in contrast

Biochemical Journal (2020) 477 4559–4580
https://doi.org/10.1042/BCJ20200781

with doing so in a protein sequence represented in the one-letter FASTA amino acid code) does not normally lead to two fragments with valid SMILES. (One potential solution to this is to adapt the SMILES grammar to remove the most troublesome elements; DeepSMILES [114] is an example of this.).

A specific example

A particularly clear example of the utility of generative methods is provided by Gómez-Bombarelli and colleagues [27] (Figure 5B), who encoded and decoded SMILES strings and represented them using a VAE. Since they could easily calculate molecular properties from the SMILES (using RDKit, www.rdkit.org), they also trained a standard MLP to use values of the latent vector as inputs and the calculated properties as outputs. In principle, any kind of deep network might be used for the encoding, and the same or any other kind for the decoding [115]. In this case, the input (encoder) network [27] was mainly a CNN while the output used a specific type of RNN called a gated recurrent unit [116,117]. The latent space used [27] was mainly of 196 dimensions, and the VAE was trained to reproduce its inputs at the outputs (another module from RDKit was used to filter invalid SMILES strings). (Parenthetically, the inputs to be encoded could have been InChI and the outputs decoded as SMILES [118]!)

Now the magic happens. Armed with the ability to generate SMILES strings (and hence molecular structures) from the latent space, the authors could then either perturb the values of latent space vectors away from known molecules, or pick more-or-less arbitrary vectors, see what SMILES strings were generated by the trained decoder, and simultaneously estimate the molecular properties of interest (Figure 5B,C). This allowed them to generate valid molecules with desirable properties (although they did still generate many non-valid SMILES strings). Having a continuous latent space linked to molecular properties (as well as molecular structures) turns the search into an optimisation problem (Figure 5C); many methods are available for such problems [9], and they chose an evolutionary algorithm.

The power of these methods meant that they could also generate chemically valid but potentially bizarre molecules, so the objective function was varied to include high values for the quantitative evaluation of drug likeness [119] (QED) and synthetic accessibility [120] (SA) scores. Although they trained their VAE using only some 250 000 molecules from the ZINC database, they estimated that they could find (i.e. generate) 30 different molecules from any training point in the latent space, implying that their VAE had effectively 'stored' the ability to make predictions about 7.5 M molecules. Although there is no simple mapping, molecules that are encoded by vectors that are 'close' in the latent space may be expected, to some degree, to be closer in structural space as judged by conventional cheminformatic fingerprints and Tanimoto distances, and hence will tend to have similar properties [121]. Understanding the extent to which this is true this will be of considerable value in the future. Plausibly, even larger nets (or ones with different architectures) trained on much bigger datasets would have generalised to many more molecules. Indeed, Arús-Pous and colleagues [122] trained an RNN with 0.1% of the 675-million GDB-13 database and established that it could generalise to create nearly 70% of GDB-13 while sampling over 2 Bn molecules.

Finally, Figure 6 shows two of these properties of generational networks, using the same VAE as used in Figure 2. In Figure 6A, we take the molecule clozapine that has a particular position in the 100-dimensional latent space. We then perturb this vector randomly by a small amount, read the SMILES strings generated by the decoder (filtering out those that are invalid), and return the molecules so generated. Note that none of these molecules was presented to the network at any point (training, validation or test). Of course 'similarity' is in the eye of the beholder, and of the encoding [123], but the top 5 molecules so generated are also ranked and labelled with the Tanimoto similarity to the original molecule as assessed using the RDKit 'patterned' encoding, and appear sensible. Similarly, Figure 6B shows some of the drugs 'encountered' on a trip between clozapine and prazosin somewhere near the centre of the space depicted in Figure 2. Again, if one is interested in either of these one might then assess such drugs in any assays of interest, having narrowed the search space considerably.

Methods to improve generalisation

As mentioned, really the key concept in neural networks is their effective 'storage' of knowledge as matrices of weights, and their ability to generalise to unseen inputs that are at least reasonably related to those used in their training. Although described in various ways, the bane of neural network training hinges on their tendency to overtrain, i.e. to overfit the training data while losing the ability to predict accurately when presented with new data. Virtually all remedies are designed to minimise this and thus to improve generalisation by 'sparsifying' or regularising the trained network. Some such methods include the use of 'decoys' (especially

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

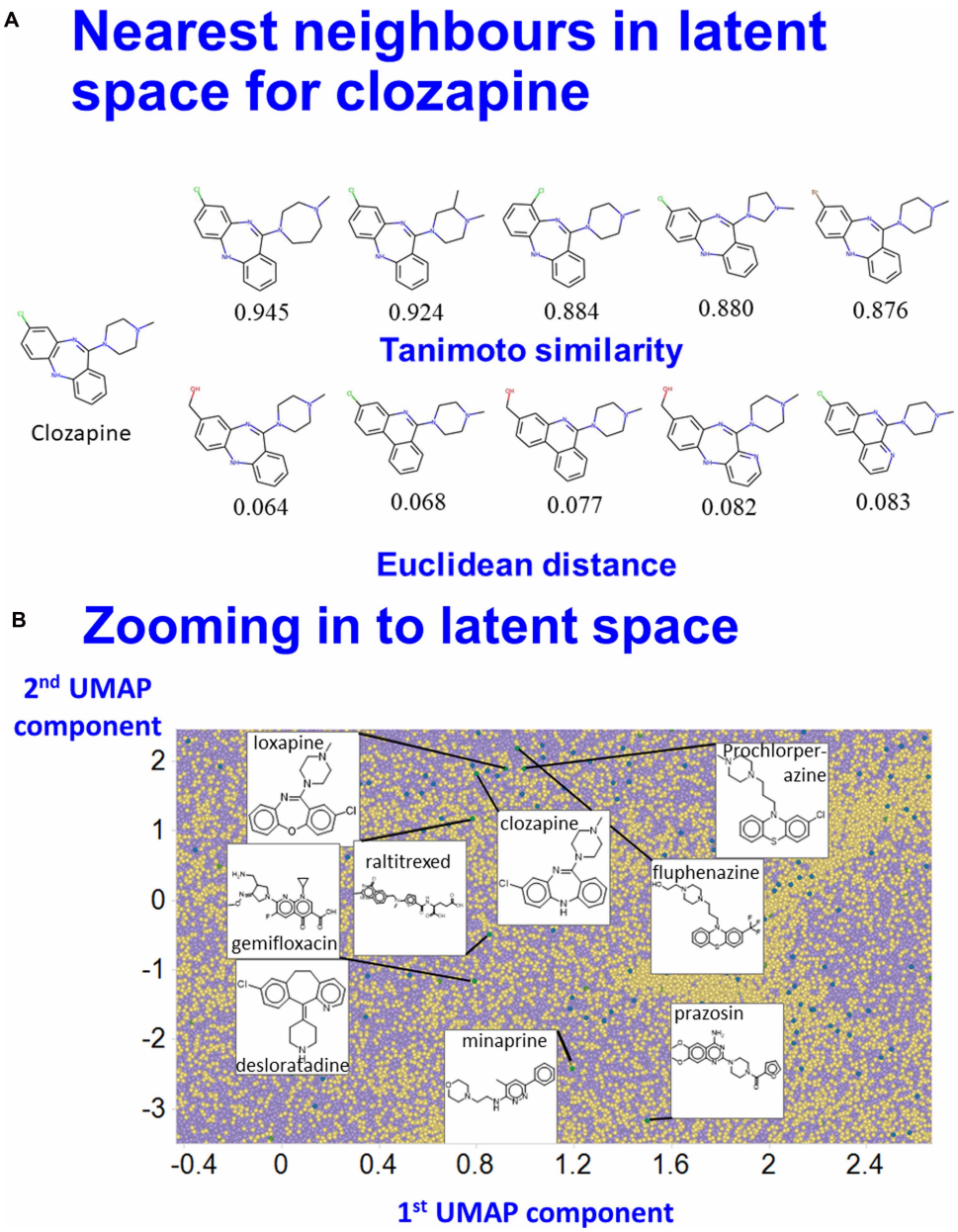

Figure 6. Local properties of latent space.

(**A**) Molecules 'near' clozapine. A VAE was trained as in Figure 2 and then the vector representing clozapine modified slightly. The closest five molecules generated are shown, as judged either by the Euclidean distance of the random modification or the Tanimoto similarity (RDKit patterned encoding). (**B**) Moving between two selected drugs (here clozapine and prazosin), it is clear that they share many obvious (and possibly non-obvious) structural similarities. UMAP components and labelling are as in Figure 2 save that selected drugs are here shown with green symbols.

generative adversarial networks [53,124,125]), the use of heavy pruning during training, especially 'dropout' [126], training in small batches [127], using ensembles of the same network [128], and the addition of noise (even 'randomised' SMILES strings [71,117]) to the inputs. It seems that in some cases the values of the hyper-parameters are critical, and they interact with each other in complex and hard-to-predict ways (and they may also be optimised using evolutionary algorithms). This makes their training computationally demanding, even before starting to vary the architecture (known as neural architecture search [129–133]). Clearly, there is no

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

limit to the possibilities with which one might combine network architecture modules. Increasing model capacity can either improve or hurt generalisation, depending on the difficulty of the generalisation problem, though the density in latent space is a key determinant of the difficult of a generalisation task [134]. There is a trend towards very large networks that (perhaps unexpectedly [135]) do not overtrain [55]. The biggest and most successful deep networks, presently GPT-3 [55], use transformer [136] architectures, including in drug discovery [137,138]. The largest flavour of GPT-3 has 96 layers with 12 299 nodes in each. At this stage we are not aware of even empirical rules relating e.g. the size of the latent space and the size of the training set of molecules, so an exploratory process, tuned to the problem of interest, is suitable.

Some further examples of deep and/or generative networks in chemical biology

The ability to 'generate' molecules *in silico* is in many ways the most exciting of the possibilities opened up by these new methods, since this allows an attack on the 'inverse' problem [139] highlighted at the beginning, i.e. to find novel molecules with desirable bioactivities even if those molecules do not themselves appear in existing databases (and of course if they do). Such molecules may not even previously have been synthesised. This is a very active area, so we give just a few examples.

Antibiotic discovery

Antimicrobial resistance is a well-known and major problem, and it would be desirable to find new anti-infectives (e.g. [140–142]). Thus, Collins and colleagues [143] trained a deep neural network on 2335 molecules that had been shown experimentally to inhibit the growth of *E. coli*. The trained network was applied *in silico* to a variety of other libraries, suggesting that a number of apparently unrelated (and untested) molecules, ones quite different in structure from known antibiotics, might prove to be of value as anti-infectives. One, named halicin (5-{(5-nitro-1,3-thiazol-2-yl)sulfanyl}-1,3,4-thiadiazol-2-amine), was shown to have a broad spectrum of antibacterial activity and appears highly promising. Although halicin is far from their structures, it is noteworthy that nitrofurantoin and metronidazole are two other existing antibiotics with a nitro group on a five-membered heterocycle, and whose mode of action means that resistance is both rare and hard to come by. Specifically, these kinds of molecule seem to have multiple transporters and are reduced to active, radical species.

One feature of *in silico* analyses is their potential for speed. Nowhere is this more obviously desirable than in the search for safe and effective molecules against coronaviruses, especially that (SARS-Cov-2) causing COVID-19. Shin et al. developed a deep learning strategy termed MT-DTI [144], which they trained on (the then) 97 million molecules in PubChem, and found that they could accurately predict binding affinities based on small molecule structures (transformed from SMILES) and the FASTA amino acid sequences of target proteins within bindingDB. Based on this, Beck and colleagues [145] could predict the potential activity against SARS-CoV-2 of a variety of known antivirals.

Drug discovery more generally

The availability of very large experimental datasets [62,146], both online and within companies, will clearly enable much better kinds of virtual screening to be performed, as properties do not simply have to be calculated but can be measured. This said, the increasing power of computers is also bringing forward the ability to calculate many more properties of interest via (time-dependent) density functional theory [147].

Gupta et al. [148] trained a variety of LSTMs in generative molecular discovery, with great success, training on 550 000 SMILES strings from ChEMBL, and generating over 25 000 novel molecules, and many others by growing fragments. Ertl et al. [149] and Segler *et al.* [150] have used similar strategies for similar purposes, and showed that such methods can perform the complete *de novo* drug design cycle to generate large sets of novel molecules for drug discovery. Yasonik [97] combined *de novo* molecular generation *in silico* (using RNNs) with a multi-objective evolutionary algorithm in an iterative method for selecting suitable molecules subjective to constraints on their physicochemical properties. Finally, here, our own laboratory has developed methods [151] based on molecular graphs and reinforcement learning for generating molecules predicted (from existing binding assays) to have a specific set of differential activities; the methods are entirely general.

Other areas

For reasons of space, we do not cover in detail a variety of other deep learning areas that may be of interest to readers. However, we give some outlines of strategies in chemical syntheses [152–154] and protein structure prediction [155,156], as well as in optimisation, where deep learning methods are also enjoying considerable success.

Chemical syntheses

Using 'intelligent' methods to decide which molecule is of interest is one thing (probably some 10^6–10^7 are easily commercially available, depending on the latency of that availability). Using intelligent methods to choose a means by which to make them oneself is entirely another [152–154,157–162]. Probably the present apotheosis of this strategy is Chematica [157,158,163] (recently commercialised as Synthia™), that started life as a manually encoded set of reaction rules and now uses (so far as one can tell from published material) a variety of means of intelligent search. The exciting prospect is for the discovery of entirely novel reactions based on those presently known; this is precisely the area in which generative methods can excel.

Protein structure prediction

Leaving aside those that need so-called chaperones, proteins are formed of strings of amino acids (primary structure) that fold up spontaneously to produce the active form of the protein (tertiary structure), which is itself assembled from the coming together of recognisable motifs that include secondary structures such as α-helices and β-sheets. Uniprot https://www.uniprot.org/ (that houses the downloadable set) presently contains some 180 M non-redundant natural protein sequences, increasing at roughly 25 M p.a. (https://bit.ly/2MZekYJ). With an average length of 337 residues, these represent ∼60Bn amino acids. In contrast, the protein databank https://www.rcsb.org/ contains only some 170k 3D structures; since sequencing speeds are far greater than are the methods of structure determination the gap is inevitably going to grow. Consequently, there has long been considerable interest in predicting structure from sequence. Correspondingly, the existence of so many sequences allows the use of unsupervised methods to help to populate the deep learning systems that can then be exploited using supervised methods for the fewer structures there are. Thus, Google DeepMind and collaborators developed Alphafold to optimise the interatomic potentials that control this assembly. They did so by combining three methods: (i) Memory-augmented simulated annealing with neural fragment generation; (ii) memory-augmented simulated annealing with neural fragment generation with distance potential, and (iii) repeated gradient descent of distance potential [156,164]. If the improvements seen during the evolution of their Go-playing reinforcement-learning-based programs [54,165,166] are anything of a guide, we may soon anticipate considerable further improvements. Similar comments might be made about the activities of specific protein sequences [167–170].

Optimisation

Most scientific problems can be cast as combinatorial search problems ('find me the best experiments to do next out of a potentially vast number') [9]. Even in a simple 'static' case where each element can take just M values in an array N of possible parameters, this scales as M^N (e.g 4 bases in a string of $30 = 4^{30} \sim 10^{18}$ possibilities [171]). This clearly applies to problems in chemistry and chemical biology that involve navigating large search spaces of molecules, and intelligent automation has been an important subset of activities here (e.g. [172–182]). 'Active learning' describes the kinds of methods that use knowledge of existing data to determine where best to explore next, and is normally used to balance exploration (looking for promising regions of the search space) with exploitation (a promising local search) [183]. Of the many strategies for this, evolutionary ('genetic') algorithms are pre-eminent, and the area is sometimes referred to as 'inverse design' [64]. Where the deep learning meets them is illustrated by the work of Nigam et al. [184], who used a genetic algorithm plus a generative model to optimise the 'penalised' logP (J_m), where $J_m = \text{logP} + \text{SA} + \text{RingPenalty}$ and in which logP is the octanol:water partition coefficient, SA is a synthetic accessibility score [120], and RingPenalty adds a penalty for rings larger than six atoms. Other areas have involved organic light-emitting diodes [147] and redox-based batteries [185]. Clearly these methods are generic, and can admit any objective function that can be calculated or measured.

Another important area of optimisation is in microbial biotechnology, whether in finding the best growth medium [186], subsets of genes to manipulate to increase productivity [187,188], or optimal sequences for generating host [189] or protein properties [190,191]. Each of these represents a combinatorial search problem [8,9].

Our last example here involves the search for an optimal (signalling) sequence for effecting protein secretion. Although secretory signals of 15–30 amino acids are more or less well known for systems such as *sec* [192] and

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

tat [193] in *E. coli*, natural evolution seems to have performed only a rather limited and stochastic search of these quite large sequence spaces. Thus, Arnold and colleagues [194] used deep learning methods to model known sequences, and could predict novel ones that were 'highly diverse in sequence, sharing as little as 58% sequence identity with the closest known native signal peptide and 73% ± 9% on average' [194]. These kinds of findings imply strongly that because Nature tends to use weak mutation and strong selection [8], necessarily becoming trapped in local optima, much is to be gained by a deeper exploration of novel sequence spaces.

Interpretability

A widespread and reasonable criticism of these 'deep learning' methods is that while they may be good at predicting interesting things, the means by which they do so is entirely opaque [195]. Unravelling this is known as 'interpretable' or 'explainable' AI. A common view is that 'disentangling' the inputs in the encoder of a VAE will lead to a representation in which individual features (or small subsets of features) of the latent space more nearly approximate features of the inputs (e.g. [196–200]). Many flavours do this by adding regulariser terms to the output objective function [201–203], such as in β-VAE [204,205], Deep VIB [206], PixelVAE [207], InfoVAE [208], PRI-VAE [203], VAE-LIME [209], Langevin-VAE [210], Discond-VAE [211], and Gaussian mixture VAE [212], while supervised methods also help with interpretability [213]. The ability to improve our understanding of which parts of a molecule are important for its activity (the 'pharmacophore') is particularly prized in drug discovery [214].

Looking to the future

Although the future of these kinds of methods is very encouraging generally, it is clear that there are a number of fruitful directions to be explored, beyond the obvious one of novel and better deep learning architectures, algorithms and hyperparameters. One is the representation of small molecules [215], a key to all computational methods, in both 2D and 3D; here the SELFIES approach [216] appears very promising. Another is the use of entirely hardware architectures; these are called neuromorphic computers, and include the spiking neural network engine SpiNNaker [217]. Thus, neural networks effectively amount to (and might loosely be modelled as) electrical circuits. One electrical component that was only recently rediscovered is the memristor; they have been badly underexplored as elements in ANNs but it is easy to predict that this is likely to be an area of fruitful activity [218].

Another area that we find surprisingly underpopulated is that of neuroevolution. While backpropagation is both popular and reasonably well understood, and may even be used in biological learning [219], it is still rather slow. However, adjusting weights (and even architectures) in deep networks to reach a desirable outcome is simply another kind of optimisation problem, that is perfectly amenable to the many flavours of evolutionary computing that work very effectively in optimisation. This has long been recognised [220,221], and since evolutionary algorithms are mostly 'embarrassingly parallelisable', we may anticipate considerable advances in the use of neuroevolution for deep learning in chemical biology.

Much of the current activity uses labelled datasets, but far more unlabelled datasets are available. It has been predicted (not least since human learning is mainly unsupervised) [52] that unsupervised learning will play a much larger role in the future, and this seems reasonable. Self-supervised methods [32] seem to show particular promise.

It was always a reasonable view that the much greater ability of humans than machines to store sense data and to reason about them was largely predicated on the simple fact that human brains contain many more neurons. This does now seem to be the case, as very large nets, showing a clear appearance of natural language 'understanding' (e.g. [55,222]), become available. They do, however, require very considerable computational resources to train, even with modern GPU-based (and similar) hardware.

With regard to chemical biology, the original motivation of the 'robot scientist' project [172] was in fact to conceive and synthesise small molecules; the generative abilities of deep networks now seem capable of bringing this fully closed loop activity within reach [152]. Furthermore, understanding the relationships between molecules that are encoded by vectors in the latent spaces used by deep networks, and their closeness to each other in structural and bioactivity spaces, is likely to be of much value.

From a scientific point of view, the empirical success of deep learning comes at a major cost. That cost, as mentioned, is the cost of knowing precisely how the 'knowledge' is actually stored in a deep network, and thus exactly how inputs map to outputs . It is also assumed that such knowledge will help avoid costly and dangerous errors as we begin to trust these methods more. Thus, 'explainable AI' will continue to be an important area for the future.

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

Concluding remarks

This has been a purposely high-level overview of some of the possibilities in cheminformatics and chemical biology that have been engendered by the development of deep learning methods in general and of generative methods in particular. Our main aim has been to draw attention to these developments, and to some of the means by which readers who are only loosely acquainted with them can incorporate these methods into their own work. Its success will be judged by the rapidity with which this happens.

Glossary

- **Discriminative model.** A type of 'supervised' machine learning model used for classification or regression problems, based on the learning of relationships between pairs of inputs X and outputs Y. Such models serve to create decision boundaries in high-dimensional space, that allow the prediction of new outputs given new inputs of comparable type. They allow modelling of the probability of Y given a value of X (P (Y|X = x)). In chemical biology they are familiar as QSAR models. Most classical ML methods involve discriminative modelling.
- **Generative model**. A much more powerful (and difficult) approach to modelling that also relates inputs X to outputs Y but that allows the *de novo* prediction of X given Y. They can do so because they capture the joint probability distribution p(X, Y) over the training data, and effectively permit the prediction of candidate inputs X given a desired output Y (i.e. P(X|Y = y)). Generative models have also been used to great effect in a very particular way in deep learning to create 'fake' or 'decoy' inputs, whose avoidance can assist deep networks in learning to generalise better [53,124]. These are commonly referred to as **Generative Adversarial Networks** (GANs).
- **Generative Adversarial Networks (GANs)**. It was recognised by Ian Goodfellow [53,124] that one way to help generative networks generalise better was to try to fool them with 'fake' examples (initially random data representing images). This has been both massively successful and highly popular, with many new flavours becoming available.
- **Hyperparameters**. The training of neural networks can be affected greatly by many control variables including the size, number and connectedness of layers, means of weight initialisation, learning rate, momentum, weight decay, transfer function, and so on. These hyperparameters still tend (and need) to be adjusted empirically for specific problems. Tuning hyperparameters is itself an optimisation problem, best addressed via intelligent search.
- **Overtraining**. Equivalent to overfitting a function by adding too many terms, this is equivalent to learning a model that fits the training data perfectly but is less good at fitting the validation or test data. It is the greatest danger of ANNs of all stripes. Indeed, almost every strategy for improving training is ultimately designed to decrease the overfitting of the training set.
- **Radial basis function networks**. These networks have a broadly similar architecture to MLPs (Figure 3B) save that the initial training consists of the application of an unsupervised (clustering) method that assigns the midpoint and width of a series of (commonly Gaussian) radial basis functions based on the examples, then uses a linear fitting between the RBFs and the output layer to train the input–output mapping.
- **Supervised learning**. One of the four main types of machine learning (the others are unsupervised, semi-supervised, and reinforcement learning). This uses paired inputs X and outputs Y to learn a nonlinear mapping from one to the other. In **unsupervised learning** the class membership of the outputs Y is not given, so only clustering is possible, while in semi-supervised learning some (usually a small subset) of the class memberships (outputs) are known and may be used to guide the learning. In reinforcement learning, a software agent takes actions in an environment, which is interpreted into both a reward and a representation of the state; these are fed back to the agent and may be used iteratively in a decision-making process. Reinforcement learning underpins many of the great successes of Google DeepMind, such as in Go playing [54,166].

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

- **Universal approximation**. A very powerful theorem that shows that any feedforward network with just a single hidden layer (of unstated and hence arbitrary size) and a suitable, nonlinear transfer function can effectively approximate anything from a continuous distribution. Although perhaps a little overblown, it does provide the conceptual underpinnings for much of the power of ANNs.

Competing Interests

The authors declare that there are no competing interests associated with the manuscript.

Open Access

Open access for this article was enabled by the participation of University of Liverpool in an all-inclusive Read & Publish pilot with Portland Press and the Biochemical Society under a transformative agreement with JISC.

Acknowledgements

Present funding includes part of the UK EPSRC project SuSCoRD [EP/S004963/1], partly sponsored by AkzoNobel. DBK is also funded by the Novo Nordisk Foundation [grant NNF10CC1016517]. We apologise to authors whose contributions were not included due to lack of space.

Abbreviations

ANN, artificial neural network; CNN, convolutional neural network; GRU, Gated Recurrent Units; LSTM, long short-term memory; MLP, multilayer perceptron; QSAR, Quantitative structure-activity relationship; RBF, radial basis function; ReLU, rectified linear unit; RNN, recurrent neural network; VAE, variational autoencoder.

References

1 Kell, D.B. and Oliver, S.G. (2004) Here is the evidence, now what is the hypothesis? The complementary roles of inductive and hypothesis-driven science in the post-genomic era. *Bioessays* **26**, 99–105 https://doi.org/10.1002/bies.10385
2 Mestres, J., Gregori-Puigjané, E., Valverde, S. and Solé, R.V. (2009) The topology of drug-target interaction networks: implicit dependence on drug properties and target families. *Mol. Biosyst.* **5**, 1051–1057 https://doi.org/10.1039/b905821b
3 Kell, D.B. (2013) Finding novel pharmaceuticals in the systems biology era using multiple effective drug targets, phenotypic screening, and knowledge of transporters: where drug discovery went wrong and how to fix it. *FEBS J.* **280**, 5957–5980 https://doi.org/10.1111/febs.12268
4 Dobson, P.D. and Kell, D.B. (2008) Carrier-mediated cellular uptake of pharmaceutical drugs: an exception or the rule? *Nat. Rev. Drug Disc.* **7**, 205–220 https://doi.org/10.1038/nrd2438
5 Kell, D.B. and Oliver, S.G. (2014) How drugs get into cells: tested and testable predictions to help discriminate between transporter-mediated uptake and lipoidal bilayer diffusion. *Front. Pharmacol.* **5**, 231 https://doi.org/10.3389/fphar.2014.00231
6 César-Razquin, A., Girardi, E., Yang, M., Brehme, M., Saez-Rodriguez, J. and Superti-Furga, G. (2018) *In silico* prioritization of transporter-drug relationships from drug sensitivity screens. *Front. Pharmacol.* **9**, 1011 https://doi.org/10.3389/fphar.2018.01011
7 Girardi, E., César-Razquin, A., Lindinger, S., Papakostas, K., Lindinger, S., Konecka, J. et al. (2020) A widespread role for SLC transmembrane transporters in resistance to cytotoxic drugs. *Nat. Chem. Biol.* **16**, 469–478 https://doi.org/10.1038/s41589-020-0483-3
8 Currin, A., Swainston, N., Day, P.J. and Kell, D.B. (2015) Synthetic biology for the directed evolution of protein biocatalysts: navigating sequence space intelligently. *Chem. Soc. Rev.* **44**, 1172–1239 https://doi.org/10.1039/C4CS00351A
9 Kell, D.B. (2012) Scientific discovery as a combinatorial optimisation problem: how best to navigate the landscape of possible experiments? *Bioessays* **34**, 236–244 https://doi.org/10.1002/bies.201100144
10 Breiman, L. (2001) Statistical modeling: the two cultures. *Stat. Sci.* **16**, 199–215 https://doi.org/10.1214/ss/1009213726
11 Arús-Pous, J., Awale, M., Probst, D. and Reymond, J.L. (2019) Exploring chemical space with machine learning. *Chimia (Aarau)* **73**, 1018–1023 https://doi.org/10.2533/chimia.2019.1018
12 Probst, D. and Reymond, J.L. (2020) Visualization of very large high-dimensional data sets as minimum spanning trees. *J. Cheminform.* **12**, 12 https://doi.org/10.1186/s13321-020-0416-x
13 Alshehri, A.S., Gani, R. and You, F.Q. (2020) Deep learning and knowledge-based methods for computer-aided molecular design-toward a unified approach: state-of-the-art and future directions. *Comput. Chem. Eng.* **141**, 107005 https://doi.org/10.1016/j.compchemeng.2020.107005
14 Bohacek, R.S., McMartin, C. and Guida, W.C. (1996) The art and practice of structure-based drug design: a molecular modeling perspective. *Med. Res. Rev.* **16**, 3–50 https://doi.org/10.1002/(SICI)1098-1128(199601)16:1<3::AID-MED1>3.0.CO;2-6
15 Polishchuk, P.G., Madzhidov, T.I. and Varnek, A. (2013) Estimation of the size of drug-like chemical space based on GDB-17 data. *J. Comput. Aided Mol. Des.* **27**, 675–679 https://doi.org/10.1007/s10822-013-9672-4
16 Sterling, T. and Irwin, J.J. (2015) ZINC 15 - ligand discovery for everyone. *J. Chem. Inf. Model.* **55**, 2324–2337 https://doi.org/10.1021/acs.jcim.5b00559
17 Ertl, P., Jelfs, S., Mühlbacher, J., Schuffenhauer, A. and Selzer, P. (2006) Quest for the rings. *In silico* exploration of ring universe to identify novel bioactive heteroaromatic scaffolds. *J. Med. Chem.* **49**, 4568–4573 https://doi.org/10.1021/jm060217p

18 Pitt, W.R., Parry, D.M., Perry, B.G. and Groom, C.R. (2009) Heteroaromatic rings of the future. *J. Med. Chem.* **52**, 2952–2963 https://doi.org/10.1021/jm801513z
19 Ruddigkeit, L., van Deursen, R., Blum, L.C. and Reymond, J.L. (2012) Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17. *J. Chem. Inf. Model.* **52**, 2864–2875 https://doi.org/10.1021/ci300415d
20 Reymond, J.L. (2015) The chemical space project. *Acc. Chem. Res.* **48**, 722–730 https://doi.org/10.1021/ar500432k
21 O'Hagan, S., Swainston, N., Handl, J. and Kell, D.B. (2015) A 'rule of 0.5' for the metabolite-likeness of approved pharmaceutical drugs. *Metabolomics* **11**, 323–339 https://doi.org/10.1007/s11306-014-0733-z
22 Samanta, S., O'Hagan, S., Swainston, N., Roberts, T.J. and Kell, D.B. (2020) VAE-Sim: a novel molecular similarity measure based on a variational autoencoder. *Molecules* **25**, 3446 https://doi.org/10.3390/molecules25153446
23 Lipinski, C.A., Lombardo, F., Dominy, B.W. and Feeney, P.J. (1997) Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Adv. Drug Deliv. Rev.* **23**, 3–25 https://doi.org/10.1016/S0169-409X(96)00423-1
24 McInnes, L., Healy, J., Saul, N. and Großberger, L. (2018) UMAP: uniform manifold approximation and projection. *J. Open Source Softw.* https://doi.org/10.21105/joss.00861
25 O'Hagan, S. and Kell, D.B. (2019) Structural similarities between some common fluorophores used in biology and marketed drugs, endogenous metabolites, and natural products. *bioRxiv* 834325 https://doi.org/10.1101/834325
26 Samanta, S., O'Hagan, S., Swainston, N., Roberts, T.J. and Kell, D.B. (2020) VAE-Sim: a novel molecular similarity measure based on a variational autoencoder. *bioRxiv* 172908 https://doi.org/10.1101/2020.06.26.172908
27 Gómez-Bombarelli, R., Wei, J.N., Duvenaud, D., Hernández-Lobato, J.M., Sánchez-Lengeling, B., Sheberla, D. et al. (2018) Automatic chemical design using a data-Driven continuous representation of molecules. *ACS Cent. Sci.* **4**, 268–276 https://doi.org/10.1021/acscentsci.7b00572
28 Banegas-Luna, A.J., Cerón-Carrasco, J.P. and Pérez-Sánchez, H. (2018) A review of ligand-based virtual screening web tools and screening algorithms in large molecular databases in the age of big data. *Future Med. Chem.* **10**, 2641–2658 https://doi.org/10.4155/fmc-2018-0076
29 Achary, P.G.R. (2020) Applications of quantitative structure-Activity relationships (QSAR) based virtual screening in drug design: a review. *Mini Rev. Med. Chem.* **20**, 1375–1388 https://doi.org/10.2174/1389557520666200429102334
30 Gorgulla, C., Boeszoermenyi, A., Wang, Z.F., Fischer, P.D., Coote, P.W., Padmanabha Das, K.M. et al. (2020) An open-source drug discovery platform enables ultra-large virtual screens. *Nature* **580**, 663–668 https://doi.org/10.1038/s41586-020-2117-z
31 Yoshimori, A., Kawasaki, E., Kanai, C. and Tasaka, T. (2020) Strategies for design of molecular structures with a desired pharmacophore using deep reinforcement learning. *Chem. Pharm. Bull. (Tokyo)* **68**, 227–233 https://doi.org/10.1248/cpb.c19-00625
32 Chen, T., Kornblith, S., Swersky, K., Norouzi, M. and Hinton, G. (2020) Big self-Supervised models are strong semi-Supervised learners. arXiv 2006.10029 https://arxiv.org/abs/2006.10029
33 Zeng, J. and Xie, P. (2020) Contrastive self-supervised learning for graph classification. arXiv 2009.05923
34 Neves, B.J., Braga, R.C., Melo-Filho, C.C., Moreira-Filho, J.T., Muratov, E.N. and Andrade, C.H. (2018) QSAR-Based Virtual screening: advances and applications in drug discovery. *Front. Pharmacol.* **9**, 1275 https://doi.org/10.3389/fphar.2018.01275
35 Tropsha, A. and Golbraikh, A. (2007) Predictive QSAR modeling workflow, model applicability domains, and virtual screening. *Curr. Pharm. Des.* **13**, 3494–3504 https://doi.org/10.2174/138161207782794257
36 Muratov, E.N., Bajorath, J., Sheridan, R.P., Tetko, I.V., Filimonov, D., Poroikov, V. et al. (2020) QSAR without borders. *Chem. Soc. Rev.* **49**, 3525–3564 https://doi.org/10.1039/D0CS00098A
37 Zupan, J. and Gasteiger, J. (1993) *Neural Networks for Chemists*, Verlag Chemie, Weinheim
38 Gasteiger, J. (2003) *Handbook of Chemoinformatics: From Data to Knowledge*, Wiley/VCH, Weinheim
39 Glorot, X. and Bengio, Y. (2010) Understanding the difficulty of training deep feedforward neural networks. *Proc AISTATs* **9**, 249–256
40 He, K., Zhang, X., Ren, S. and Sun, J. (2015) Delving deep into rectifiers: surpassing human-Level performance on imageNet classification. arXiv 1502.01852v01851
41 Broadhurst, D. and Kell, D.B. (2006) Statistical strategies for avoiding false discoveries in metabolomics and related experiments. *Metabolomics* **2**, 171–196 https://doi.org/10.1007/s11306-006-0037-z
42 Hassibi, B., Stork, D.G. and Wolff, G.J. (1993) Optimal brain surgeon and general network pruning. *Int. Conf. Neural Netw.* **1**, 293–299 https://doi.org/10.1109/ICNN.1993.298572
43 Le Cun, Y., Denker, J.S. and Solla, S.A. (1990) Optimal brain damage. *Adv. Neural Inf. Proc. Syst.* **2**, 598–605
44 Broomhead, D.S. and Lowe, D. (1988) Multivariable function interpolation and adaptive networks. *Complex Syst.* **2**, 321–355
45 Que, Q. and Belkin, M. (2020) Back to the future: radial basis function network revisited. *IEEE Trans. Pattern Anal. Mach. Intell.* **42**, 1856–1867 https://doi.org/10.1109/TPAMI.2019.2906594
46 Goodacre, R., Timmins, É.M, Burton, R., Kaderbhai, N., Woodward, A.M., Kell, D.B. et al. (1998) Rapid identification of urinary tract infection bacteria using hyperspectral whole-organism fingerprinting and artificial neural networks. *Microbiology* **144**, 1157–1170 https://doi.org/10.1099/00221287-144-5-1157
47 Beavis, R.B., Colby, S.M., Goodacre, R., Harrington, P.D.B., Reilly, J.P., Sokolow, S. et al. (2000) Artificial intelligence and expert systems in mass spectrometry. In *Encyclopedia of Analytical Chemistry* (Meyers, R.A., ed.), pp. 11558–11597, Wiley, Chichester
48 Hinton, G.E., Osindero, S. and Teh, Y.W. (2006) A fast learning algorithm for deep belief nets. *Neural Comput.* **18**, 1527–1554 https://doi.org/10.1162/neco.2006.18.7.1527
49 Kohonen, T. (2000) *Self-organising Maps*, Springer, Berlin
50 He, K., Zhang, X., Ren, S. and Sun, J. (2015) Deep residual learning for image recognition. arXiv 1512.03385v03381
51 Schmidhuber, J. (2015) Deep learning in neural networks: an overview. *Neural Netw.* **61**, 85–117 https://doi.org/10.1016/j.neunet.2014.09.003
52 LeCun, Y., Bengio, Y. and Hinton, G. (2015) Deep learning. *Nature* **521**, 436–444 https://doi.org/10.1038/nature14539
53 Goodfellow, I., Bengio, Y. and Courville, A. (2016) *Deep Learning*, MIT Press, Boston
54 Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A. et al. (2017) Mastering the game of Go without human knowledge. *Nature* **550**, 354–359 https://doi.org/10.1038/nature24270
55 Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P. et al. (2020) Language models are Few-Shot learners. arXiv 2005.14165

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

56 Chen, H.M., Engkvist, O., Wang, Y.H., Olivecrona, M. and Blaschke, T. (2018) The rise of deep learning in drug discovery. *Drug Discov. Today* **23**, 1241–1250 https://doi.org/10.1016/j.drudis.2018.01.039
57 Gawehn, E., Hiss, J.A. and Schneider, G. (2016) Deep learning in drug discovery. *Mol. Inform.* **35**, 3–14 https://doi.org/10.1002/minf.201501008
58 Arús-Pous, J., Probst, D. and Reymond, J.L. (2018) Deep learning invades drug design and synthesis. *Chimia (Aarau)* **72**, 70–71 https://doi.org/10.2533/chimia.2018.70
59 Baskin, I.I. (2020) The power of deep learning to ligand-based novel drug discovery. *Expert Opin. Drug Discov.* **15**, 755–764 https://doi.org/10.1080/17460441.2020.1745183
60 Lavecchia, A. (2019) Deep learning in drug discovery: opportunities, challenges and future prospects. *Drug Discov. Today* **24**, 2017–2032 https://doi.org/10.1016/j.drudis.2019.07.006
61 Elton, D.C., Boukouvalas, Z., Fuge, M.D. and Chung, P.W. (2019) Deep learning for molecular design: a review of the state of the art. *Mol. Syst. Des. Eng.* **4**, 828–849 https://doi.org/10.1039/C9ME00039A
62 David, L., Arús-Pous, J., Karlsson, J., Engkvist, O., Bjerrum, E.J., Kogej, T. et al. (2019) Applications of deep-Learning in exploiting large-Scale and heterogeneous compound data in industrial pharmaceutical research. *Front. Pharmacol.* **10**, 1303 https://doi.org/10.3389/fphar.2019.01303
63 Schneider, G. (2018) Generative models for artificially-intelligent molecular design. *Mol. Inform.* **37**, 188031 https://doi.org/10.1002/minf.201880131
64 Sanchez-Lengeling, B. and Aspuru-Guzik, A. (2018) Inverse molecular design using machine learning: generative models for matter engineering. *Science* **361**, 360–365 https://doi.org/10.1126/science.aat2663
65 Schneider, P., Walters, W.P., Plowright, A.T., Sieroka, N., Listgarten, J., Goodnow, R.A. et al. (2020) Rethinking drug design in the artificial intelligence era. *Nat. Rev. Drug Discov.* **19**, 353–364 https://doi.org/10.1038/s41573-019-0050-3
66 Goodacre, R., Trew, S., Wrigley-Jones, C., Saunders, G., Neal, M.J., Porter, N. et al. (1995) Rapid and quantitative analysis of metabolites in fermentor broths using pyrolysis mass spectrometry with supervised learning: application to the screening of *penicillium chryosgenum* fermentations for the overproduction of penicillins. *Anal. Chim. Acta* **313**, 25–43 https://doi.org/10.1016/0003-2670(95)00170-5
67 Glorot, X., Borden, A. and Bengio, Y. (2011) Deep sparse rectifier neural networks. *Proc AISTATS* **15**, 315–323
68 Clevert, D.-A., Unterthiner, T. and Hochreiter, S. (2015) Fast and accurate deep network learning by exponential linear units (ELUs). arXiv 1511.07289
69 Hayou, S., Doucet, A. and Rousseau, J. (2019) On the impact of the activation function on deep neural networks training. arXiv 1902.06853
70 Cireşan, D.C., Meier, U., Gambardella, L.M. and Schmidhuber, J. (2010) Deep, big, simple neural nets for handwritten digit recognition. *Neural Comput.* **22**, 3207–3220 https://doi.org/10.1162/NECO_a_00052
71 Arús-Pous, J., Johansson, S.V., Prykhodko, O., Bjerrum, E.J., Tyrchan, C., Reymond, J.L. et al. (2019) Randomized SMILES strings improve the quality of molecular generative models. *J. Cheminform.* **11**, 71 https://doi.org/10.1186/s13321-019-0393-0
72 O'Hagan, S. and Kell, D.B. (2015) The KNIME workflow environment and its applications in genetic programming and machine learning. *Genetic Progr. Evol. Mach.* **16**, 387–391 https://doi.org/10.1007/s10710-015-9247-3
73 Ma, J.S., Sheridan, R.P., Liaw, A., Dahl, G.E. and Svetnik, V. (2015) Deep neural nets as a method for quantitative structure-Activity relationships. *J. Chem. Inf. Model.* **55**, 263–274 https://doi.org/10.1021/ci500747n
74 Lane, T.R., Foil, D.H., Minerali, E., Urbina, F., Zorn, K.M. and Ekins, S. (2020) A very large-Scale bioactivity comparison of deep learning and multiple machine learning algorithms for drug discovery. *ChemRxiv* https://doi.org/10.26434/chemrxiv.12781241.v12781241
75 Hastie, T., Tibshirani, R. and Friedman, J. (2009) *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, 2nd edition, Springer-Verlag, Berlin
76 LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W. et al. (1989) Backpropagation applied to handwritten Zip code recognition. *Neural Comput.* **1**, 541–551 https://doi.org/10.1162/neco.1989.1.4.541
77 Yamashita, R., Nishio, M., Do, R.K.G. and Togashi, K. (2018) Convolutional neural networks: an overview and application in radiology. *Insights Imaging* **9**, 611–629 https://doi.org/10.1007/s13244-018-0639-9
78 Gupta, A., Harrison, P.J., Wieslander, H., Pielawski, N., Kartasalo, K., Partel, G. et al. (2019) Deep learning in image cytometry: a review. *Cytometry A* **95**, 366–380 https://doi.org/10.1002/cyto.a.23701
79 Mayer McKinney, S., Sieniek, M., Godbole, V., Godwin, J., Antropova, N., Ashrafian, H. et al. (2020) International evaluation of an AI system for breast cancer screening. *Nature* **577**, 89–94 https://doi.org/10.1038/s41586-019-1799-6
80 Wallach, I., Dzamba, M. and Heifets, A. (2015) Atomnet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery. arXiv 1510.02855v02851
81 Dwivedi, V.P., Joshi, C.K., Laurent, T., Bengio, Y. and Bresson, X. (2020) Benchmarking graph neural networks. arXiv 2003.00982
82 Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C. and Yu, P.S. (2020) A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* https://doi.org/10.1109/TNNLS.2020.2978386
83 Hochreiter, S. and Schmidhuber, J. (1997) Long short-term memory. *Neural Comput.* **9**, 1735–1780 https://doi.org/10.1162/neco.1997.9.8.1735
84 Grisoni, F. and Schneider, G. (2019) De novo molecular design with generative long short-term memory. *Chimia* **73**, 1006–1011 https://doi.org/10.2533/chimia.2019.1006
85 Yu, Y., Si, X., Hu, C. and Zhang, J. (2019) A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* **31**, 1235–1270 https://doi.org/10.1162/neco_a_01199
86 Sherstinsky, A. (2020) Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D* **404**, 132306 https://doi.org/10.1016/j.physd.2019.132306
87 Rae, J.W., Potapenko, A., Jayakumar, S.M. and Lillicrap, T.P. (2019) Compressive transformers for long-Range sequence modelling. arXiv 1911.05507
88 Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. et al. (2014) Learning phrase representations using RNN encoder–Decoder for statistical machine translation. arXiv 1406.1078v1403
89 Kingma, D. and Welling, M. (2014) Auto-encoding variational Bayes. arXiv 1312.6114v1310
90 Kingma, D.P. and Welling, M. (2019) An introduction to variational autoencoders. *Found Trends Mach. Learn.* **12**, 4–89 https://doi.org/10.1561/2200000056
91 Rezende, D.J., Mohamed, S. and Wierstra, D. (2014) Stochastic backpropagation and approximate inference in deep generative models. arXiv 1401.4082v1403

92 Gómez-Bombarelli, R., Duvenaud, D., Hernández-Lobato, J., Aguilera-Iparraguirre, J., Hirzel, T.D., Adams, R.P. et al. (2016) Automatic chemical design using a data-driven continuous representation of molecules. arXiv 1610.02415v02411
93 Ståhl, N., Falkman, G., Karlsson, A., Mathiason, G. and Boström, J. (2019) Deep reinforcement learning for multiparameter optimization in de novo drug design. *J. Chem. Inf. Model.* **59**, 3166–3176 https://doi.org/10.1021/acs.jcim.9b00325
94 Brown, N., Fiscato, M., Segler, M.H.S. and Vaucher, A.C. (2019) Guacamol: benchmarking models for de novo molecular design. *J. Chem. Inf. Model.* **59**, 1096–1108 https://doi.org/10.1021/acs.jcim.8b00839
95 Button, A., Merk, D., Hiss, J.A. and Schneider, G. (2019) Automated de novo molecular design by hybrid machine intelligence and rule-driven chemical synthesis. *Nat. Mach. Intell.* **1**, 307–315 https://doi.org/10.1038/s42256-019-0067-7
96 Khemchandani, Y., O'Hagan, S., Samanta, S., Swainston, N., Roberts, T.J., Bollegala, D. et al. (2020) Deepgraphmolgen, a multiobjective, computational strategy for generating molecules with desirable properties: a graph convolution and reinforcement learning approach. *J. Cheminform.* **12**, 53 https://doi.org/10.1186/s13321-020-00454-3
97 Yasonik, J. (2020) Multiobjective de novo drug design with recurrent neural networks and nondominated sorting. *J. Cheminform.* **12**, 14 https://doi.org/10.1186/s13321-020-00419-6
98 Li, Y., Hu, J., Wang, Y., Zhou, J., Zhang, L. and Liu, Z. (2020) Deepscaffold: a comprehensive tool for scaffold-based de novo drug discovery using deep learning. *J. Chem. Inf. Model.* **60**, 77–91 https://doi.org/10.1021/acs.jcim.9b00727
99 Moret, M., Friedrich, L., Grisoni, F., Merk, D. and Schneider, G. (2020) Generative molecular design in low data regimes. *Nat. Mach. Intell.* **2**, 171–180 https://doi.org/10.1038/s42256-020-0160-y
100 Colby, S.M., Nuñez, J.R., Hodas, N.O., Corley, C.D. and Renslow, R.R. (2020) Deep learning to generate in silico chemical property libraries and candidate molecules for small molecule identification in complex samples. *Anal. Chem.* **92**, 1720–1729 https://doi.org/10.1021/acs.analchem.9b02348
101 Méndez-Lucio, O., Baillif, B., Clevert, D.A., Rouquié, D. and Wichard, J. (2020) De novo generation of hit-like molecules from gene expression signatures using artificial intelligence. *Nat. Commun.* **11**, 10 https://doi.org/10.1038/s41467-019-13807-w
102 Walters, W.P. and Murcko, M. (2020) Assessing the impact of generative AI on medicinal chemistry. *Nat. Biotechnol.* **38**, 143–145 https://doi.org/10.1038/s41587-020-0418-2
103 Schneider, G. and Fechner, U. (2005) Computer-based de novo design of drug-like molecules. *Nat. Rev. Drug Discov.* **4**, 649–663 https://doi.org/10.1038/nrd1799
104 Olivecrona, M., Blaschke, T., Engkvist, O. and Chen, H.M. (2017) Molecular de-novo design through deep reinforcement learning. *J. Cheminform.* **9**, 48 https://doi.org/10.1186/s13321-017-0235-x
105 Chen, G., Shen, Z.Q., Iyer, A., Ghumman, U.F., Tang, S., Bi, J.B. et al. (2020) Machine-Learning-Assisted De novo design of organic molecules and polymers: opportunities and challenges. *Polymers* **12**, 163 https://doi.org/10.3390/polym12010163
106 Vanhaelen, Q., Lin, Y.C. and Zhavoronkov, A. (2020) The advent of generative chemistry. *ACS Med. Chem. Lett.* **11**, 1496–1505 https://doi.org/10.1021/acsmedchemlett.0c00088
107 Lopez, R., Gayoso, A. and Yosef, N. (2020) Enhancing scientific discoveries in molecular biology with deep generative models. *Mol. Syst. Biol.* **16**, e9198 https://doi.org/10.15252/msb.20199198
108 Weininger, D. (1988) SMILES, a chemical language and information system .1. introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **28**, 31–36 https://doi.org/10.1021/ci00057a005
109 Kusner, M.J., Paige, B. and Hernández-Lobato, J.M. (2017) Grammar variational autoencoder. arXiv 1703.01925v01921
110 Kajino, H. (2018) Molecular hypergraph grammar with Its application to molecular optimization. arXiv 1809.02745v02741
111 Yang, K., Swanson, K., Jin, W., Coley, C., Eiden, P., Gao, H. et al. (2019) Analyzing learned molecular representations for property prediction. *J. Chem. Inf. Model.* **59**, 3370–3388 https://doi.org/10.1021/acs.jcim.9b00237
112 Jin, W., Barzilay, R. and Jaakkola, T. (2018) Junction tree variational autoencoder for molecular graph generation. arXiv 1802.04364v04362
113 You, J., Liu, B., Ying, R., Pande, V. and Leskovec, J. (2018) Graph convolutional policy network for goal-directed molecular graph generation. arXiv 1806.02473v02471
114 O'Boyle, N. and Dalke, A. (2018) DeepSMILES: an adaptation of SMILES for use in machine-Learning of chemical structures. *ChemRxiv* 7097960.v7097961
115 Kim, K., Kang, S., Yoo, J., Kwon, Y., Nam, Y., Lee, D. et al. (2018) Deep-learning-based inverse design model for intelligent discovery of organic molecules. *Npj Comput. Mater.* **4**, 67 https://doi.org/10.1038/s41524-018-0128-1
116 Cho, K., van Merrienboer, B., Bahdanau, D. and Bengio, Y. (2014) On the properties of neural machine translation: encoder-Decoder approaches. arXiv 1409.1259v1402
117 van Deursen, R., Ertl, P., Tetko, I.V. and Godin, G. (2020) GEN: highly efficient SMILES explorer using autodidactic generative examination networks. *J. Cheminform.* **12**, 22 https://doi.org/10.1186/s13321-020-00425-8
118 Winter, R., Montanari, F., Noé, F. and Clevert, D.A. (2019) Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chem. Sci.* **10**, 1692–1701 https://doi.org/10.1039/C8SC04175J
119 Bickerton, G.R., Paolini, G.V., Besnard, J., Muresan, S. and Hopkins, A.L. (2012) Quantifying the chemical beauty of drugs. *Nat. Chem.* **4**, 90–98 https://doi.org/10.1038/nchem.1243
120 Ertl, P. and Schuffenhauer, A. (2009) Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *J. Cheminform.* **1**, 8 https://doi.org/10.1186/1758-2946-1-8
121 Bender, A. and Glen, R.C. (2004) Molecular similarity: a key technique in molecular informatics. *Org. Biomol. Chem.* **2**, 3204–3218 https://doi.org/10.1039/b409813g
122 Arús-Pous, J., Blaschke, T., Ulander, S., Reymond, J.L., Chen, H. and Engkvist, O. (2019) Exploring the GDB-13 chemical space using deep generative models. *J. Cheminform.* **11**, 20 https://doi.org/10.1186/s13321-019-0341-z
123 O'Hagan, S. and Kell, D.B. (2017) Consensus rank orderings of molecular fingerprints illustrate the 'most genuine' similarities between marketed drugs and small endogenous human metabolites, but highlight exogenous natural products as the most important 'natural' drug transporter substrates. *ADMET DMPK* **5**, 85–125 https://doi.org/10.5599/admet.5.2.376
124 Goodfellow, I. (2017) Generative adversarial networks. arXiv 1701.00160v00161

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

125 Lin, E., Lin, C.H. and Lane, H.Y. (2020) Relevant applications of generative adversarial networks in drug design and discovery: molecular *de novo* design, dimensionality reduction, and *de novo* peptide and protein design. *Molecules* **25**, 3250 https://doi.org/10.3390/molecules25143250
126 Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R.R. (2012) Improving neural networks by preventing co-adaptation of feature detectors. arXiv 1207.0580
127 Keskar, N.S., Mudigere, D., Nocedal, J., Smelyanskiy, M. and Tang, P.T.P. (2017) On large-batch training for deep learning: generalization Gap and sharp minima. arXiv 1609.04836v04832
128 Dietterich, T.G. (2000) Ensemble methods in machine learning. *LNCS* **1857**, 1–15 https://doi.org/10.1007/3-540-45014-9_1
129 Elsken, T., Metzen, J.H. and Hutter, F. (2018) Neural architecture search: a survey. arXiv 1808.05377
130 Xie, L., Chen, X., Bi, K., Wei, L., Xu, Y., Chen, Z. et al. (2020) Weight-Sharing neural architecture search: a battle to shrink the optimization Gap. arXiv 2008.01475
131 Lindauer, M. and Hutter, F. (2019) Best practices for scientific research on neural architecture search. arXiv 1909.02453
132 Lukasik, J., Friede, D., Zela, A., Stuckenschmidt, H., Hutter, F. and Keuper, M. (2020) Smooth variational graph embeddings for efficient neural architecture search. arXiv 2010.04683
133 White, C., Neiswanger, W., Nolen, S. and Savani, Y. (2020) A study on encodings for neural architecture search. arXiv 2007.04965
134 Bozkurt, A., Esmaeili, B., Brooks, D.H., Dy, J.G. and van de Meent, J.-W. (2019) Evaluating combinatorial generalization in variational autoencoders. arXiv 1911.04594v04591
135 Belkin, M., Hsu, D., Ma, S. and Mandal, S. (2019) Reconciling modern machine-learning practice and the classical bias-variance trade-off. *Proc. Natl Acad. Sci. U.S.A.* **116**, 15849–15854 https://doi.org/10.1073/pnas.1903070116
136 Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N. et al. (2017) Attention Is All You need. arXiv 1706.03762
137 Wang, S., Guo, Y., Wang, Y., Sun, H. and Huang, J. (2019) SMILES-BERT: Large Scale Unsupervised Pre-Training for Molecular Property Prediction. ACM-BCB 429–436 https://doi.org/10.1145/3307339.3342186
138 Grechishnikova, D. (2020) Transformer neural network for protein specific de novo drug generation as machine translation problem. *bioRxiv* https://doi.org/10.1101/863415
139 Miyao, T., Kaneko, H. and Funatsu, K. (2016) Inverse QSPR/QSAR analysis for chemical structure generation (from y to x). *J. Chem. Inf. Model.* **56**, 286–299 https://doi.org/10.1021/acs.jcim.5b00628
140 Holmes, A.H., Moore, L.S.P., Sundsfjord, A., Steinbakk, M., Regmi, S., Karkey, A. et al. (2016) Understanding the mechanisms and drivers of antimicrobial resistance. *Lancet* **387**, 176–187 https://doi.org/10.1016/S0140-6736(15)00473-0
141 Moo, C.L., Yang, S.K., Yusoff, K., Ajat, M., Thomas, W., Abushelaibi, A. et al. (2019) Mechanisms of antimicrobial resistance (AMR) and alternative approaches to overcome AMR. *Curr. Drug Discov. Technol.* **17**, 430–447 https://doi.org/10.2174/1570163816666190304122219
142 Salcedo-Sora, J.E. and Kell, D.B. (2020) A quantitative survey of bacterial persistence in the presence of antibiotics: towards antipersister antimicrobial discovery. *Antibiotics* **9**, 508 https://doi.org/10.3390/antibiotics9080508
143 Stokes, J.M., Yang, K., Swanson, K., Jin, W., Cubillos-Ruiz, A., Donghia, N.M. et al. (2020) A deep learning approach to antibiotic discovery. *Cell* **180**, 688–702 e613 https://doi.org/10.1016/j.cell.2020.01.021
144 Shin, B., Park, S., Kang, K. and Ho, J.C. (2019) Self-Attention based molecule representation for predicting drug-Target interaction. *Proc. Mach. Learn. Res.* **106**, 1–18
145 Beck, B.R., Shin, B., Choi, Y., Park, S. and Kang, K. (2020) Predicting commercially available antiviral drugs that may act on the novel coronavirus (SARS-CoV-2) through a drug-target interaction deep learning model. *Comput. Struct. Biotechnol. J.* **18**, 784–790 https://doi.org/10.1016/j.csbj.2020.03.025
146 Montáns, F.J., Chinesta, F., Gómez-Bombarelli, R. and Kutz, J.N. (2019) Data-driven modeling and learning in science and engineering. *Cr. Mecanique* **347**, 845–855 https://doi.org/10.1016/j.crme.2019.11.009
147 Gómez-Bombarelli, R. and Aspuru-Guzik, A. (2019) Computational discovery of organic LED materials. *Comput. Mater. Disc.*, 423–446 https://doi.org/10.1039/9781788010122-00423
148 Gupta, A., Müller, A.T., Huisman, B.J.H., Fuchs, J.A., Schneider, P. and Schneider, G. (2018) Generative recurrent networks for *de novo* drug design. *Mol. Inform.* **37**, 1700111 https://doi.org/10.1002/minf.201700111
149 Ertl, P., Lewis, R., Martin, E. and Polyakov, V. (2017) In silico generation of novel, drug-like chemical matter using the LSTM neural network. arXiv 1712.07449
150 Segler, M.H.S., Kogej, T., Tyrchan, C. and Waller, M.P. (2018) Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Cent. Sci.* **4**, 120–131 https://doi.org/10.1021/acscentsci.7b00512
151 Khemchandani, Y., O'Hagan, S., Samanta, S., Swainston, N., Roberts, T.J., Bollegala, D. et al. (2020) Deepgraphmol, a multiobjective, computational strategy for generating molecules with desirable properties: a graph convolution and reinforcement learning approach. *bioRxiv* 2020/114165 https://doi.org/10.1101/2020.05.25.114165
152 Coley, C.W., Thomas, III, D.A., Lummiss, J.A.M., Jaworski, J.N., Breen, C.P., Schultz, V. et al. (2019) A robotic platform for flow synthesis of organic compounds informed by AI planning. *Science* **365**, 557 https://doi.org/10.1126/science.aax1566
153 Fooshee, D., Mood, A., Gutman, E., Tavakoli, M., Urban, G., Liu, F. et al. (2018) Deep learning for chemical reaction prediction. *Mol. Syst. Des. Eng.* **3**, 442–452 https://doi.org/10.1039/C7ME00107J
154 Segler, M.H.S., Preuss, M. and Waller, M.P. (2018) Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* **555**, 604–610 https://doi.org/10.1038/nature25978
155 Billings, W.M., Hedelius, B., Millecam, T., Wingate, D. and Della Corte, D. (2019) ProSPr: democratized implementation of alphafold protein distance prediction network. *bioRxiv* 830273
156 Senior, A.W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T. et al. (2020) Improved protein structure prediction using potentials from deep learning. *Nature* **577**, 706–710 https://doi.org/10.1038/s41586-019-1923-7
157 Klucznik, T., Mikulak-Klucznik, B., McCormack, M.P., Lima, H., Szymkuć, S., Bhowmick, M. et al. (2018) Efficient syntheses of diverse, medicinally relevant targets planned by computer and executed in the laboratory. *Chem* **4**, 522–532 https://doi.org/10.1016/j.chempr.2018.02.002
158 Szymkuć, S., Gajewska, E.P., Klucznik, T., Molga, K., Dittwald, P., Startek, M. et al. (2016) Computer-Assisted synthetic planning: the End of the beginning. *Angew. Chem. Int. Ed. Engl.* **55**, 5904–5937 https://doi.org/10.1002/anie.201506101

159 Badowski, T., Molga, K. and Grzybowski, B.A. (2019) Selection of cost-effective yet chemically diverse pathways from the networks of computergenerated retrosynthetic plans. *Chem. Sci.* **10**, 4640–4651 https://doi.org/10.1039/c8sc05611k
160 Badowski, T., Gajewska, E.P., Molga, K. and Grzybowski, B.A. (2020) Synergy between expert and machine-Learning approaches allows for improved retrosynthetic planning. *Angew. Chem. Int. Ed. Engl.* **59**, 725–730 https://doi.org/10.1002/anie.201912083
161 Strieth-Kalthoff, F., Sandfort, F., Segler, M.H.S. and Glorius, F. (2020) Machine learning the ropes: principles, applications and directions in synthetic chemistry. *Chem. Soc. Rev.* **49**, 6154–6168 https://doi.org/10.1039/C9CS00786E
162 Pflüger, P.M. and Glorius, F. (2020) Molecular machine learning: the future of synthetic chemistry? *Angew. Chem. Int. Ed. Engl.* **59**, 18860–18865 https://doi.org/10.1002/anie.202008366
163 Molga, K., Dittwald, P. and Grzybowski, B.A. (2019) Computational design of syntheses leading to compound libraries or isotopically labelled targets. *Chem. Sci.* **10**, 9219–9232 https://doi.org/10.1039/C9SC02678A
164 Senior, A.W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T. et al. (2019) Protein structure prediction using multiple deep neural networks in the 13th critical assessment of protein structure prediction (CASP13). *Proteins* **87**, 1141–1148 https://doi.org/10.1002/prot.25834
165 Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., van den Driessche, G. et al. (2016) Mastering the game of Go with deep neural networks and tree search. *Nature* **529**, 484–489 https://doi.org/10.1038/nature16961
166 Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A. et al. (2018) A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* **362**, 1140–1144 https://doi.org/10.1126/science.aar6404
167 Karimi, M., Wu, D., Wang, Z. and Shen, Y. (2019) Deepaffinity: interpretable deep learning of compound-protein affinity through unified recurrent and convolutional neural networks. *Bioinformatics* **35**, 3329–3338 https://doi.org/10.1093/bioinformatics/btz111
168 Dean, S.N. and Walper, S.A. (2020) Variational autoencoder for generation of antimicrobial peptides. *ACS Omega* **5**, 20746–20754 https://doi.org/10.1021/acsomega.0c00442
169 Risso, V.A., Romero-Rivera, A., Gutierrez-Rus, L.I., Ortega-Munoz, M., Santoyo-Gonzalez, F., Gavira, J.A. et al. (2020) Enhancing a *de novo* enzyme activity by computationally-focused ultra-low-throughput screening. *Chem. Sci.* **11**, 6134–6148 https://doi.org/10.1039/D0SC01935F
170 Shroff, R., Cole, A.W., Diaz, D.J., Morrow, B.R., Donnell, I., Annapareddy, A. et al. (2020) Discovery of novel gain-of-Function mutations guided by structure-Based deep learning. *ACS Synth. Biol.* **9**, 2927–2935 https://doi.org/10.1021/acssynbio.0c00345
171 Knight, C.G., Platt, M., Rowe, W., Wedge, D.C., Khan, F., Day, P. et al. (2009) Array-based evolution of DNA aptamers allows modelling of an explicit sequence-fitness landscape. *Nucleic Acids Res.* **37**, e6 https://doi.org/10.1093/nar/gkn899
172 King, R.D., Whelan, K.E., Jones, F.M., Reiser, P.G.K., Bryant, C.H., Muggleton, S.H. et al. (2004) Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature* **427**, 247–252 https://doi.org/10.1038/nature02236
173 O'Hagan, S., Dunn, W.B., Brown, M., Knowles, J.D. and Kell, D.B. (2005) Closed-loop, multiobjective optimisation of analytical instrumentation: gas-chromatography-time-of-flight mass spectrometry of the metabolomes of human serum and of yeast fermentations. *Anal. Chem.* **77**, 290–303 https://doi.org/10.1021/ac049146x
174 Henson, A.B., Gromski, P.S. and Cronin, L. (2018) Designing algorithms To Aid discovery by chemical robots. *ACS Cent. Sci.* **4**, 793–804 https://doi.org/10.1021/acscentsci.8b00176
175 Gromski, P.S., Henson, A.B., Granda, J.M. and Cronin, L. (2019) How to explore chemical space using algorithms and automation. *Nat. Rev. Chem.* **3**, 119–128 https://doi.org/10.1038/s41570-018-0066-y
176 Häse, F., Roch, L.M. and Aspuru-Guzik, A. (2019) Next-Generation experimentation with self-Driving laboratories. *Trends Chem.* **1**, 282–291 https://doi.org/10.1016/j.trechm.2019.02.007
177 Burger, B., Maffettone, P.M., Gusev, V.V., Aitchison, C.M., Bai, Y., Wang, X. et al. (2020) A mobile robotic chemist. *Nature* **583**, 237–241 https://doi.org/10.1038/s41586-020-2442-2
178 Roch, L.M., Hase, F., Kreisbeck, C., Tamayo-Mendoza, T., Yunker, L.P.E., Hein, J.E. et al. (2020) ChemOS: an orchestration software to democratize autonomous discovery. *PLoS One* **15**, e0229862 https://doi.org/10.1371/journal.pone.0229862
179 Gromski, P.S., Granda, J.M. and Cronin, L. (2020) Universal chemical synthesis and discovery with 'The chemputer'. *Trends Chem.* **2**, 4–12 https://doi.org/10.1016/j.trechm.2019.07.004
180 Coley, C.W., Eyke, N.S. and Jensen, K.F. (2020) Autonomous discovery in the chemical sciences part II: outlook. *Angew. Chem. Int. Ed. Engl.* https://doi.org/10.1002/anie.201909989
181 Coley, C.W., Eyke, N.S. and Jensen, K.F. (2020) Autonomous discovery in the chemical sciences part I: progress. *Angew. Chem. Int. Ed. Engl.* https://doi.org/10.1002/anie.201909987
182 Mehr, S.H.M., Craven, M., Leonov, A.I., Keenan, G. and Cronin, L. (2020) A universal system for digitization and automatic execution of the chemical synthesis literature. *Science* **370**, 101–108 https://doi.org/10.1126/science.abc2986
183 Jones, D.R., Schonlau, M. and Welch, W.J. (1998) Efficient global optimization of expensive black-box functions. *J. Global. Opt.* **13**, 455–492 https://doi.org/10.1023/A:1008306431147
184 Nigam, A., Friederich, P., Krenn, M. and Aspuru-Guzik, A. (2019) Augmenting genetic algorithms with deep neural networks for exploring the chemical space. arXiv 1909.11655
185 Tabor, D.P., Roch, L.M., Saikin, S.K., Kreisbeck, C., Sheberla, D., Montoya, J.H. et al. (2018) Accelerating the discovery of materials for clean energy in the era of smart automation. *Nat. Rev. Mater* **3**, 5–20 https://doi.org/10.1038/s41578-018-0005-z
186 Link, H. and Weuster-Botz, D. (2011) Medium Formulation and Development. In *Comprehensive Biotechnology* (Moo-Young, M., ed.), pp. 119–134, Elsevier, Amsterdam
187 Kell, D.B., Swainston, N., Pir, P. and Oliver, S.G. (2015) Membrane transporter engineering in industrial biotechnology and whole-cell biocatalysis. *Trends Biotechnol.* **33**, 237–246 https://doi.org/10.1016/j.tibtech.2015.02.001
188 Garst, A.D., Bassalo, M.C., Pines, G., Lynch, S.A., Halweg-Edwards, A.L., Liu, R. et al. (2017) Genome-wide mapping of mutations at single-nucleotide resolution for protein, metabolic and genome engineering. *Nat. Biotechnol.* **35**, 48–55 https://doi.org/10.1038/nbt.3718
189 Höllerer, S., Papaxanthos, L., Gumpinger, A.C., Fischer, K., Beisel, C., Borgwardt, K. et al. (2020) Large-scale DNA-based phenotypic recording and deep learning enable highly accurate sequence-function mapping. *Nat. Commun.* **11**, 3551 https://doi.org/10.1038/s41467-020-17222-4

Biochemical Journal (2020) **477** 4559–4580
https://doi.org/10.1042/BCJ20200781

190 Arnold, F.H. (2019) Innovation by evolution: bringing new chemistry to life (Nobel lecture). *Angew. Chem. Int. Ed. Engl.* **58**, 14420–14426 https://doi.org/10.1002/anie.201907729
191 Linder, J., Bogard, N., Rosenberg, A.B. and Seelig, G. (2020) A generative neural network for maximizing fitness and diversity of synthetic DNA and protein sequences. *Cell Syst.* **11**, 49–62 e16 https://doi.org/10.1016/j.cels.2020.05.007
192 Costa, T.R., Felisberto-Rodrigues, C., Meir, A., Prevost, M.S., Redzej, A., Trokter, M. et al. (2015) Secretion systems in gram-negative bacteria: structural and mechanistic insights. *Nat. Rev. Microbiol.* **13**, 343–359 https://doi.org/10.1038/nrmicro3456
193 Fröbel, J., Rose, P. and Müller, M. (2012) Twin-arginine-dependent translocation of folded proteins. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **367**, 1029–1046 https://doi.org/10.1098/rstb.2011.0202
194 Wu, Z., Yang, K.K., Liszka, M., Lee, A., Batzilla, A., Wernick, D. et al. (2020) Signal peptides generated by attention-based neural networks. *ACS Synth. Biol.* **9**, 2154–2161 https://doi.org/10.1021/acssynbio.0c00219
195 Holzinger, A., Biemann, C., Pattichis, C.S. and Kell, D.B. (2017) What do we need to build explainable AI systems for the medical domain? arXiv 1712.09923v09921
196 Bengio, Y., Courville, A. and Vincent, P. (2013) Representation learning: a review and New perspectives. *IEEE Trans. Patt. Anal. Mach. Intell.* **35**, 1798–1828 https://doi.org/10.1109/TPAMI.2013.50
197 Kumar, A., Sattigeri, P. and Balakrishnan, A. (2017) Variational inference of disentangled latent concepts from unlabeled observations. arXiv 1711.00848
198 Chen, R.T.Q., Li, X., Grosse, R. and Duvenaud, D. (2018) Isolating sources of disentanglement in variational autoencoders. arXiv 1802.04942
199 Tschannen, M., Bachem, O. and Lucic, M. (2018) Recent advances in autoencoder-Based representation learning. arXiv 1812.05069v05061
200 Mathieu, E., Rainforth, T., Siddharth, N. and Teh, Y.W. (2018) Disentangling disentanglement in variational autoencoders. arXiv 1812.02833
201 Rezende, D.J. and Viola, F. (2018) Taming VAEs. arXiv 1810.00597v00591
202 Dai, B. and Wipf, D. (2019) Diagnosing and enhancing VAE models. arXiv 1903.05789v05782
203 Li, Y., Yu, S., Principe, J.C., Li, X. and Wu, D. (2020) PRI-VAE: principle-of-Relevant-Information variational autoencoders. arXiv 2007.06503
204 Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M. et al. (2017) β-VAE: learning basic visual concepts with a constrained variational framework. *Proc ICLR*
205 Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G. et al. (2018) Understanding disentangling in β-VAE. arXiv 1804.03599
206 Alemi, A.A., Fischer, I., Dillon, J.V. and Murphy, K. (2016) Deep variational information bottleneck. arXiv 1612.00419
207 Gulrajani, I., Kumar, K., Ahmed, F., Taiga, A.A., Visin, F., Vazquez, D. et al. (2016) PixelVAE: a latent variable model for natural images. arXiv 1611.05013
208 Zhao, S., Song, J. and Ermon, S. (2017) InfoVAE: balancing learning and inference in variational autoencoders. arXiv 1706.02262v02263
209 Schockaert, C., Macher, V. and Schmitz, A. (2020) VAE-LIME: deep generative model based approach for local data-Driven model interpretability applied to the ironmaking industry. arXiv 2007.10256
210 Wang, Z. and Delingette, H. (2020) Quasi-symplectic langevin variational autoencoder. arXiv 2009.01675
211 Choi, J., Hwang, G. and Kang, M. (2020) Discond-VAE: disentangling continuous factors from the discrete. arXiv 2009.08039
212 Yang, Z., Sarkar, A. and Cooper, S. (2020) Game level clustering and generation using Gaussian mixture VAEs. arXiv
213 Nguyen, A.P and Martínez, M.R. (2020) Learning invariances for interpretability using supervised VAE. arXiv 2007.07591
214 Preuer, K., Klambauer, G., Rippmann, F., Hochreiter, S. and Unterthiner, T. (2019) Interpretable deep learning in drug discovery. arXiv 1903.02788
215 Chuang, K.V., Gunsalus, L.M. and Keiser, M.J. (2020) Learning molecular representations for medicinal chemistry. *J. Med. Chem.* **63**, 8705–8722 https://doi.org/10.1021/acs.jmedchem.0c00385
216 Krenn, M., Häse, F., Nigam, A., Friederich, P. and Aspuru-Guzik, A. (2019) Self-Referencing embedded strings (SELFIES): a 100% robust molecular string representation. arXiv 1905.13741
217 Rowley, A.G.D., Brenninkmeijer, C., Davidson, S., Fellows, D., Gait, A., Lester, D.R. et al. (2019) SpiNNTools: the execution engine for the SpiNNaker platform. *Front. Neurosci.* **13**, 231 https://doi.org/10.3389/fnins.2019.00231
218 Thomas, A. (2013) Memristor-based neural networks. *J. Phys. D* **46**, 093001 https://doi.org/10.1088/0022-3727/46/9/093001
219 Lillicrap, T.P., Santoro, A., Marris, L., Akerman, C.J. and Hinton, G. (2020) Backpropagation and the brain. *Nat. Rev. Neurosci.* **21**, 335–346 https://doi.org/10.1038/s41583-020-0277-3
220 Yao, X. (1999) Evolving artificial neural networks. *Proc. IEEE.* **87**, 1423–1447 https://doi.org/10.1109/5.784219
221 Stanley, K.O., Clune, J., Lehman, J. and Miikkulainen, R. (2019) Designing neural networks through neuroevolution. *Nat. Mach. Intell.* **1**, 24–35 https://doi.org/10.1038/s42256-018-0006-z
222 Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. (2018) BERT: pre-training of deep bidirectional transformers for language understanding. arXiv 1810.04805