superseded simply by scaling up the quantity of training data, along with commensurate scaling of the model size and associated compute power used for training (Sutton, 2019). Not only can large models have superior performance on a specific task but they may be capable of solving a broader range of different problems with the same trained neural network. Large language models are a notable example as a single network not only has an extraordinary breadth of capability but is even able to outperform specialist networks designed to solve specific problems.

Section 12.3.5

We have seen that depth plays an important role in allowing neural networks to achieve high performance. One way to view the role of the hidden layers in a deep neural network is that of *representation learning* (Bengio, Courville, and Vincent, 2012) in which the network learns to transform input data into a new representation that is semantically meaningful thereby creating a much easier problem for the final layer or layers to solve. Such internal representations can be repurposed to allow for the solution of related problems through transfer learning, as we saw for skin lesion classification. It is interesting to note that neural networks used to process images may learn internal representations that are remarkably like those observed in the mammalian visual cortex. Large neural networks that can be adapted or *fine-tuned* to a range of downstream tasks are called *foundation models*, and can take advantage of large, heterogeneous data sets to create models having broad applicability (Bommasani *et al.*, 2021).

Section 10.3

In addition to scaling, there were other developments that helped in the success of deep learning. For example, in simple neural networks, the training signals become weaker as they are backpropagated through successive layers of a deep network. One technique for addressing this is the introduction of *residual connections* (He *et al.*, 2015a) that facilitate the training of networks having hundreds of layers. Another key development was the introduction of *automatic differentiation* methods in which the code that performs backpropagation to evaluate error function gradients is generated automatically from the code used to specify the forward propagation. This allows researchers to experiment rapidly with different architectures for a neural network and to combine different architectural elements in multiple ways very easily since only the relatively simple forward propagation functions need to be coded explicitly. Also, much of the research in machine learning has been conducted through open source, allowing researchers to build on the work of others, thereby further accelerating the rate of progress in the field.

Section 9.5