Minds and Machines (2019) 29:417–440
https://doi.org/10.1007/s11023-019-09506-6

The Rhetoric and Reality of Anthropomorphism in Artificial Intelligence

David Watson[1,2]

Received: 7 June 2019 / Accepted: 14 September 2019 / Published online: 21 September 2019

Abstract
Artificial intelligence (AI) has historically been conceptualized in anthropomorphic terms. Some algorithms deploy biomimetic designs in a deliberate attempt to effect a sort of digital isomorphism of the human brain. Others leverage more general learning strategies that happen to coincide with popular theories of cognitive science and social epistemology. In this paper, I challenge the anthropomorphic credentials of the neural network algorithm, whose similarities to human cognition I argue are vastly overstated and narrowly construed. I submit that three alternative supervised learning methods—namely lasso penalties, bagging, and boosting—offer subtler, more interesting analogies to human reasoning as both an individual and a social phenomenon. Despite the temptation to fall back on anthropomorphic tropes when discussing AI, however, I conclude that such rhetoric is at best misleading and at worst downright dangerous. The impulse to humanize algorithms is an obstacle to properly conceptualizing the ethical challenges posed by emerging technologies.

Keywords Artificial intelligence · Machine learning · Epistemology · Social epistemology · Cognitive science · Digital ethics

1 Introduction

Ever since the seminal work of Turing (1950) if not before, experts and laypeople alike have tended to frame computational achievements in explicitly epistemological terms. We speak of machines that *think*, *learn*, and *infer*. The name of the discipline itself—*artificial intelligence*—practically dares us to compare our human modes of reasoning with the behavior of algorithms. It is not always clear whether such language is meant to be literal or metaphorical.

✉ David Watson
david.watson@oii.ox.ac.uk

[1] Oxford Internet Institute, University of Oxford, 41 Saint Giles', Oxford OX1 3LW, UK

[2] Alan Turing Institute, London, UK

In this article, I attempt to move beyond the platitudes and critically examine specific examples of algorithms that employ learning strategies found in cognitive science and social epistemology. I argue that too much emphasis has been placed on the purported structural similarities between biological and artificial neural networks. More illuminating analogies can be found in other areas of computational statistics, notably three cases I shall explore in considerable depth: lasso penalties, bagging, and boosting. While each enjoys some interesting connections to modern neural networks, together they constitute an extremely general collection of techniques that can be fruitfully applied to almost any supervised learning algorithm. These methods, which are widely used in data science but mostly unfamiliar to audiences beyond this domain, demonstrate how the narrow focus on mechanism as a locus of biological verisimilitude ignores the functional aspects of human intelligence. Finally, I shall argue that while the connections between machine learning algorithms and human cognition may be intriguing and suggestive, the rhetoric of anthropomorphism can do more harm than good when it comes to conceptualizing the important ethical challenges posed by emerging technologies.

The rest of this paper is structured as follows. In Sect. 2, I briefly review some background terminology that will be essential to the proceeding analysis. In Sect. 3, I examine the neuroscientific inspiration behind the neural network algorithm and underscore three important differences between human cognition and so-called "deep learning". In Sects. 4, 5, 6, I introduce lasso penalties, bagging, and boosting, respectively. I show how each resembles or builds upon popular theories of cognitive science and social epistemology, providing informative and unexpected examples of interdisciplinary convergence. Though it is easy and tempting to speak of algorithms in anthropomorphic terms, I caution against such rhetoric in Sect. 7. I conclude in Sect. 8 with a summary.

2 Terminology

All algorithms reviewed in this article are instances of *supervised learning*. The typical supervised learning setup involves a matrix of features X (a.k.a. predictors, independent variables, etc.) and a vector of outcomes Y (a.k.a. the response, dependent variable, etc.) that together form some fixed but unknown joint distribution $P(X, Y)$. The goal is to infer a function f that predicts Y based on X. If Y is continuous, then f is a regressor; if Y is categorical, then f is a classifier. For a good textbook introduction to statistical learning, see (Hastie et al. 2009).

The performance of a supervised learner f is measured by a *loss function*, which quantifies the model's error. For instance, a classifier may record a loss of 0 for correct predictions and 1 for misclassifications. Loss can be decomposed into *bias* and *variance*, roughly akin to the concepts of accuracy and precision. A low-bias, high-variance model is one that both overshoots and undershoots the mark, occasionally by large margins but in roughly equal proportions. A high-bias, low-variance model is more consistent in its outputs—but consistently wrong. See Fig. 1 for an illustration. There is an inherent bias-variance trade-off in all supervised learning algorithms.

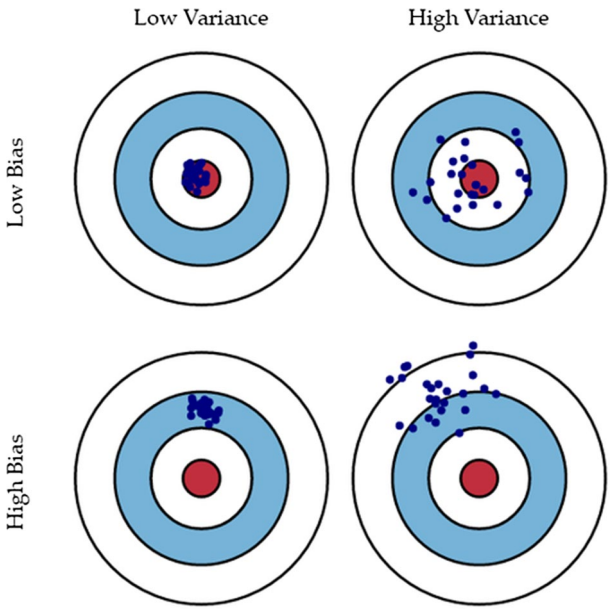

Fig. 1 Visual depiction of bias and variance, key concepts in evaluating supervised learning algorithms

Predictions from multiple models can be combined in a process known as *ensemble learning*. A learning ensemble consists of many individual *base learners* or *basis functions*, whose outputs are typically pooled either by summation or averaging. This strategy can be especially advantageous with low-bias, high-variance models such as decision or regression trees, which are often referred to as *weak learners*. Two popular forms of ensemble learning are reviewed in Sects. 5, 6.

A model f is judged by its ability to *generalize*, i.e., to successfully predict outcomes on data that were not included in its training set. If f performs well on training samples but poorly on test samples, then we say that f is *overfit*—it has learned the properties of some particular observations, but not the underlying distribution from which they were drawn, $P(X, Y)$. Overfitting may be mitigated by a number of clever strategies collectively referred to as *regularization*. Specific examples of regularization will be detailed in Sects. 4, 5, 6.

To guard against overfitting, models are typically evaluated not based on their in-sample performance, but on their out-of-sample performance. Ideally, this would be done by training on one dataset and testing on another sampled from the same population. However, data scarcity can make this strategy inefficient in practice. The typical solution is to implement a resampling procedure that divides the data in some systematic way. The most popular example of such a method is *cross-validation*. To cross-validate an algorithm, we split the data into k subsets (or *folds*) of roughly equal size. We then train k separate models, with each fold held out once for testing. The average generalization error across the k trials is reported.

Another common resampling procedure is based on *bootstrapping*. Bootstrapping was originally proposed as a nonparametric technique for estimating the variance of

statistical parameters (Efron 1979). The idea is simple. Say we observe the height of n individuals. Our goal is to compute not just the mean of this sample, but also the corresponding standard error. (Of course, we could do so analytically under minimal parametric assumptions, but the following method applies more generally.) We create a bootstrap sample by drawing n observations *with replacement* from the original data. The replacement step ensures a degree of variation across bootstraps, as some observations will appear multiple times in a single bootstrap, and others not at all. By recording the mean of each bootstrap sample and repeating the procedure some large number of times B, we get an unbiased estimate of the sampling distribution of the mean. The standard deviation of this distribution is a plug-in estimator for the standard error.

Note that a little over one third of observations will tend to be excluded from any given bootstrap sample. Specifically, each observation has an exclusion probability of $e^{-1} \approx 0.368$, which is extremely useful for model evaluation, since these unsampled cases—the so-called out-of-bag (OOB) observations—form a natural test set. This will be especially important in Sects. 5, 6.

Having reviewed this background material, we may now apply the relevant concepts with formal clarity to a number of machine learning algorithms.

3 Neural Networks

Research in neural networks began in 1958 with Frank Rosenblatt's perceptron model (Rosenblatt 1958). Rosenblatt was a US Navy psychologist, and the perceptron algorithm he developed was explicitly inspired by a mathematical idealization of the neuron, the brain's most basic information processing unit. Biological neurons are connected by synapses that enable communication through electrical or chemical signals. Building on this idea, Rosenblatt proposed a model architecture in which input features are mapped to outputs through an intermediate layer of neurons (see Fig. 2). The weights connecting these components are analogous to the synaptic strength of incoming and outgoing channels. At the output layer, values are passed through a nonlinear activator function to mimic the thresholding effect of biological neurons, which respond to stimuli by either firing or not firing.

Neural networks have evolved considerably since Rosenblatt first published his perceptron model. Modern variants of the algorithm tend to include many more layers—thence the name *deep* neural networks (DNNs)—an approach inspired at least in part by anatomical research. In their influential study of the cat visual cortex, Hubel and Wiesel (1962) differentiated between so-called "simple" cells, which detect edges and curves, and "complex" cells, which combine simple cells to identify larger shapes with greater spatial invariance. The authors hypothesized that a hierarchy of neural layers could enable increasingly complex cognition, allowing the brain to operate at higher levels of abstraction. DNNs implement this theory at scale. Employing complex convolutional architectures (Krizhevsky et al. 2012) and clever activation functions (Glorot et al. 2011), DNNs have led the latest wave of excitement about and funding for AI research. Descendants of the perceptron algorithm now power translation services for Google (Wu et al.

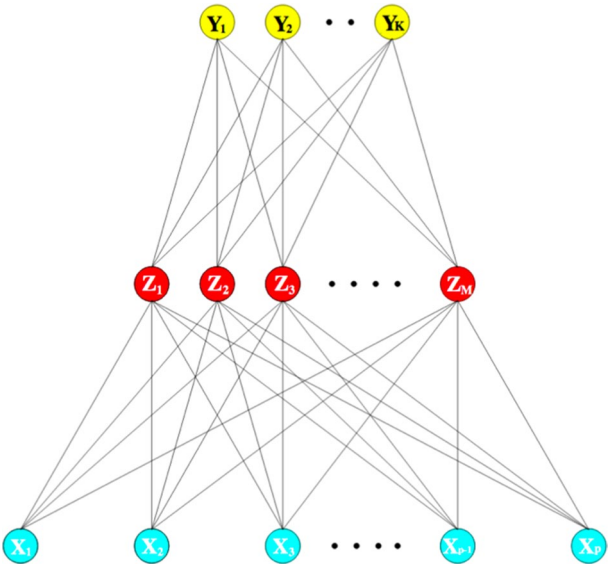

Fig. 2 Schematic depiction of a single-layered neural network. Input features X are combined at each neuron Z, which in turn combine to produce predictions Y (From Hastie et al. (2009), p. 393)

2016), facial recognition software for Facebook (Taigman et al. 2014), and virtual assistants like Apple's Siri (Siri Team 2017).

The biomimetic approach to AI has always inspired the popular imagination. Writing about Rosenblatt's perceptron, the *New York Times* declared in 1958 that "The Navy has revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence" (*New York Times* 1958, p. 25). The exuberance has only been somewhat tempered by the intervening decades. The same newspaper recently published a piece on DeepMind's AlphaZero, a DNN that is the reigning world champion of chess, shogi, and Go (Silver et al. 2018). In the essay, Steven Strogatz describes the algorithm in almost breathless language:

> Most unnerving was that AlphaZero seemed to express insight. It played like no computer ever has, intuitively and beautifully, with a romantic, attacking style. It played gambits and took risks…. AlphaZero had the finesse of a virtuoso and the power of a machine. It was humankind's first glimpse of an awesome new kind of intelligence. (Strogatz 2018)

Excitement about DNNs is hardly limited to the popular press. (Strogatz, it should be noted, is a professor of mathematics.) Some leading researchers in deep learning have suggested that the anthropomorphic connection in fact runs both ways, proposing that "neural networks from AI can be used as plausible simulacra of biological brains, potentially providing detailed explanations of the computations occurring therein" (Hassabis et al. 2017, p. 254).

Indeed, this is more or less the central tenet of *connectionism*, a decades-old movement in cognitive science and philosophy that has seen a renaissance with the recent success of deep learning (Buckner and Garson 2019). DNNs have been used to model information processing at various stages of the visual cortex of human and nonhuman primates (Cichy et al. 2016; Kriegeskorte 2015; Yamins and DiCarlo 2016), achieving state of the art predictive performance while simultaneously suggesting novel subcortical functions. Stinson (2016) reviews a number of epistemological advantages of connectionism, which she maintains is unique among computational models in its ability to reveal generic mechanisms in the brain. At least one philosopher has argued that DNNs instantiate a mode of "transformational abstraction" that resolves longstanding debates between rationalists and empiricists (Buckner 2018).

There is no denying that the achievements of AlphaZero and other top performing DNNs are impressive. But a large and growing strain of literature in computational statistics has recently emphasized the limitations of these algorithms, which deviate from human modes of learning in several fundamental and alarming ways. A complete list of the differences between DNNs and biological neural networks would be too long to enumerate here. See (Marcus 2018) for a good overview. Instead I will highlight three especially noteworthy dissimilarities that underscore the shortcomings of this paradigm, which I argue has been vastly overhyped since DNNs first attained state of the art performance in speech recognition (Dahl et al. 2012; Mohamed et al. 2012; Raina et al. 2009) and image classification tasks (Krizhevsky et al. 2012; LeCun et al. 2015; Lecun et al. 1998). These results notwithstanding, there is good reason to believe that, compared to human brains, DNNs are *brittle*, *inefficient*, and *myopic* in specific senses to be explained below.

DNNs tend to break down in the face of minor attacks. In a landmark paper, Goodfellow et al. (2014) introduced *generative adversarial networks* (GANs), a new class of DNNs designed to fool other DNNs through slight perturbations of the input features. For instance, by adding just a small amount of noise to the pixels of a photograph, Goodfellow et al. (2015) were able to trick the high-performing ImageNet classifier into mislabeling a panda as a gibbon, even though differences between the two images are imperceptible to the human eye (see Fig. 3). Others have fooled

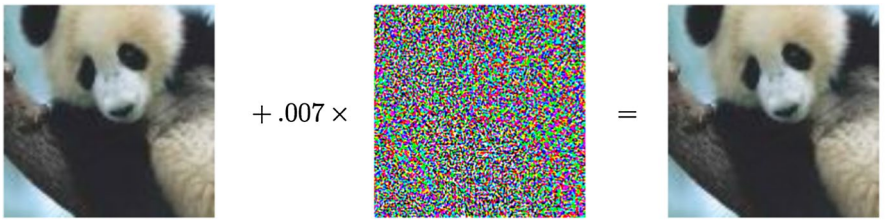

Fig. 3 Example of an adversarial perturbation from Goodfellow et al. (2015), p. 3

DNNs into misclassifying zebras as horses (Zhu et al. 2017), bananas as toasters (Brown et al. 2017), and many other absurd combinations. While GANs were originally viewed as something of a curiosity in the deep learning community, they have since been widely recognized as a profound challenge that may undermine the applicability of DNNs to safety–critical areas such as clinical medicine (Finlayson et al. 2019) or autonomous vehicles (Eykholt et al. 2018). Needless to say, humans are much more resilient to minor perturbations of our sensory stimuli. This disconnect between biological and artificial neural networks suggests that the latter lack some crucial component essential to navigating the real world.

Recent work on GANs has complicated this conclusion somewhat. Elsayed et al. (2018) have shown that adversarial attacks negatively influence the predictive performance of time-limited humans. In a much larger study, Zhou and Firestone (2019) found that participants were able to decipher a wide array of adversarial examples. The reason may have to do with the generalizability of certain visual perturbations. Ilyas et al. (2019) demonstrate that attacks designed to fool one DNN often succeed in fooling others trained independently. The authors infer from this that adversarial examples are features, not bugs—i.e., that they encode true information about "non-robust" properties of the input data that may be incomprehensible to humans. Their work has sparked intense interest among machine learning researchers—see (Engstrom et al. 2019) for a discussion—but it is not immediately clear what lessons are to be drawn for the connectionist. For even if GANs do reveal some otherwise imperceptible reality about the underlying geometry of visual data, the fact remains that those representations are largely inaccessible to humans. Zhou and Firestone attempt to show the opposite, but they specifically rule out attacks of the sort considered above, in which an image is misclassified through thousands of minor perturbations. Some ability to distinguish between what Ilyas et al. call "robust" and "non-robust" features—a distinction they acknowledge is inescapably anthropocentric—still appears essential.

Another important flaw with DNNs is that they are woefully data inefficient. High-performing models typically need millions of examples to learn distinctions that would strike a human as immediately obvious. Geoffrey Hinton, one of the pioneers of DNNs and a recent recipient of the ACM's prestigious Turing Award for excellence in computing, has raised the issue himself in interviews. "For a child to learn to recognize a cow," he remarked, "it's not like their mother needs to say 'cow' 10,000 times" (Waldrop 2019). Indeed, even very young humans are typically capable of one-shot learning, generalizing from just a single instance. This is simply impossible for most DNNs, a limitation that is especially frustrating in cases where abundant, high-quality data are prohibitively expensive or difficult to collect. Gathering large volumes of labelled photographs is not especially challenging, but comparable datasets for genetics or particle physics are another matter altogether.

Reinforcement learning arguably poses a clever workaround to this problem, in which synthetic data are generated as part of the training process (Sutton and Barto 2018). However, this approach is constrained by our ability to simulate realistic data for the target system. Preprocessing strategies have been developed for data augmentation in image recognition tasks (Perez and Wang 2017), but again these are not universally applicable. More importantly, neither solution addresses the underlying

issue—we want models to learn more with less data, not generate their own data so they can continue in their profligate ways. A more explicitly biomimetic approach would be to develop memory augmented systems, and several labs have made good progress in this area (Collier and Beel 2018; Graves et al. 2016; Vinyals et al. 2016). Unfortunately, these models often fail during training or are very slow to converge, which explains why they have only been implemented for relatively simple tasks to date. Promising though these strands of research may be, one-shot learning remains a significant challenge for DNNs.

A final important difference between human cognition and deep learning is that the latter has proven itself to be strangely myopic. The problem is most evident in the case of image classification. Careful analysis of the intermediate layers of convolutional DNNs reveals that whereas the lowest level neurons deal in pixels, higher level neurons operate on more meaningful features like eyes and ears, just as Hubel and Wiesel hypothesized (Olah et al. 2018). Yet even top performing models can learn to discriminate between objects while completely failing to grasp their interrelationships. For instance, rearranging Kim Kardashian's mouth and eye in Fig. 4 actually improved the DNN's prediction, indicating something deeply wrong with the underlying model, which performs well on out-of-sample data (Bourdakos 2017).

Zhou and Firestone (2019) hypothesize that the alleged myopia problem is just a byproduct of the requirement that DNNs select a label from a constrained choice set. They write:

> Whereas humans have separate concepts for appearing *like* something vs. appearing *to be* that thing—as when a cloud looks like a dog without looking like it *is* a dog…[DNNs] are not permitted to make this distinction, instead being forced to play the game of picking whichever label in their repertoire best matches an image… (p. 8)

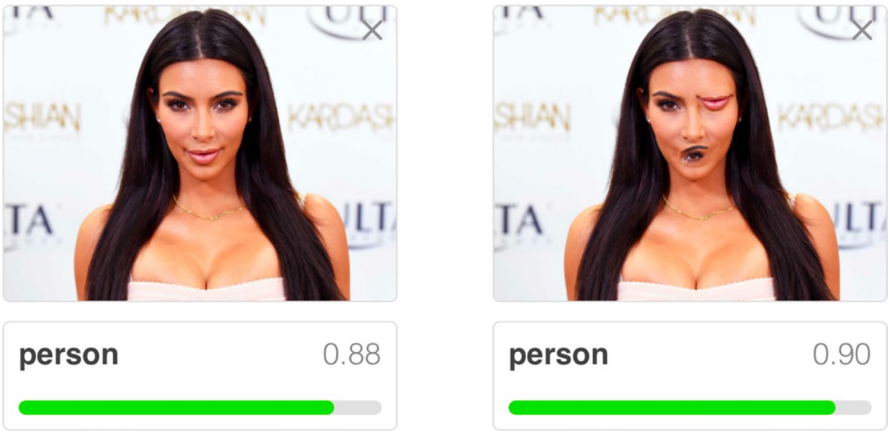

Fig. 4 Predictions from a convolutional DNN on two images of Kim Kardashian. Alarmingly, rearranging her facial features does not adversely affect the model's prediction (From Bourdakos (2017))

This explanation goes some way toward explaining the perplexing results of the perturbed Kim Kardashian image in Fig. 4. However, the true problem runs deeper than Zhou and Firestone suggest. Hinton argues that myopia is hardwired into convolutional DNNs via the max pooling function, which compresses the information between layers (Hinton et al. 2011). Max pooling discards valuable spatial information that humans use to identify and interact with objects, losing all semblance of structural hierarchies in the process. Thus any combination of eyes, nose, and mouth will suffice for a convolutional DNN—not because of external constraints on the choice set, but because of intrinsic limitations of the model architecture. Hinton et al. recently proposed a new algorithm called *capsule networks* in an effort to overcome these deficiencies (Hinton et al. 2018; Sabour et al. 2017), but the technology is still in its infancy (Fig. 5).

The problems of algorithmic brittleness, inefficiency, and myopia are not unique to DNNs—although these models are perhaps the worst offenders on all fronts—nor do they undermine the central premise of connectionism, a bold and fruitful theory that has generated much valuable research in AI, cognitive science, philosophy of mind. What these objections do establish, however, is that the ostensible affinities between biological brains and modern DNNs should be treated with skepticism. The anthropomorphic hype around deep learning is uncritical and overblown. It would be a mistake to say that these algorithms *recreate* human intelligence; instead, they introduce some new mode of inference that outperforms us in some ways and falls short in others.

Often lost in the excitement surrounding DNNs is the fact that other approaches to machine learning exist, many with considerable advantages over neural networks on a wide range of tasks. The next three sections are devoted to several such methods, with an emphasis on their epistemological underpinnings and anthropomorphic connections.

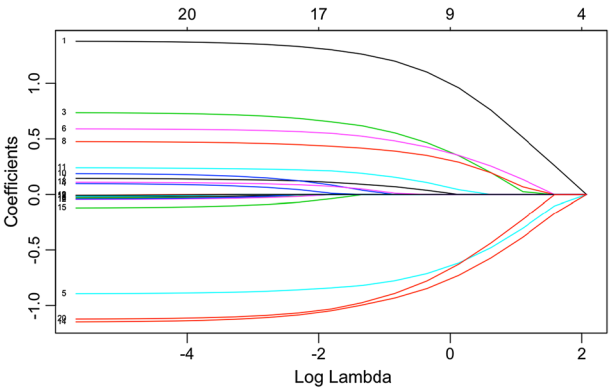

Fig. 5 Example of the "lasso path" of model coefficients in a linear regression. All weights converge toward zero as the penalty parameter increases. Adapted from the glmnet package vignette (Hastie and Qian 2014)

4 Lasso Penalties

Lasso penalties are a popular form of regularization in machine learning. They are designed for modelling *sparse* datasets, i.e., those in which at least some of our recorded variables are uninformative with respect to the response. For instance, biological knowledge tells us that only a small percentage of genes are likely to be involved in any given clinical outcome. However, high-throughput technologies allow scientists to test thousands or even millions of genetic associations in a single experiment. The lasso provides a fast, principled method for selecting top features in such settings.

Originally introduced by Robert Tibshirani (1996), lasso penalties impose a cost not just on predictive errors—that is the role of the loss function—but on the model parameters themselves, preventing them from growing too large in absolute value. For instance, a linear regression with a lasso penalty solves the following optimization problem:

$$\min_{\beta \in \mathbb{R}^p} \frac{1}{n} \|y - \mathbf{X}\beta\|_2^2 + \lambda \|\beta\|_1$$

The first summand corresponds to the mean square error, the typical loss function in regression tasks. The second summand puts a data-adaptive weight λ on the L_1-norm (i.e., the sum of absolute values) of the coefficient vector β. This term effectively shrinks all model parameters toward 0. At the optimal value of the lasso penalty λ, usually selected via cross-validation, this algorithm will tend to remove uninformative predictors altogether.

The basic intuition behind the lasso is that datasets are often intolerably noisy. We need some sensible method for eliminating variables that hinder our ability to detect and exploit signals of interest. The lasso is not the only way to achieve this goal. Several sparsity-inducing Bayesian priors have been proposed to similar effect (Carvalho et al. 2010; Ishwaran and Rao 2005). So-called "greedy" algorithms like stepwise regression and recursive feature elimination iteratively remove predictors by comparing nested models (Guyon et al. 2002). Projection techniques such as principal component analysis (Jolliffe 2002) and *t*-stochastic neighborhood embedding (van der Maaten and Hinton 2008) are designed to recover latent variables, low-dimensional data projections that preserve as much information as possible from the original high-dimensional feature space.

The lasso is unique, however, in its combination of speed and interpretability. Recursive refitting can be prohibitively slow with large datasets, and stepwise regression is inapplicable when features outnumber samples. Bayesian methods are notorious for their computational overhead. By contrast, fast optimization algorithms exist for computing complete lasso paths in generalized linear models (Friedman et al. 2010) and estimating sparse inverse covariance matrices (Friedman et al. 2007) without any dimensionality constraints. Whereas projection techniques require complex, potentially nonlinear combinations of the original inputs, a fitted lasso regression is no more difficult to understand than an ordinary linear model, with nonzero coefficients on just a subset of the original features. These practical advantages help

explain why lasso penalties have become so widespread in contemporary statistics and machine learning, as they enable the analysis of high-dimensional datasets that are challenging or impossible to model using traditional regression and classification techniques (Bühlmann and van de Geer 2011).

It is worth noting that lasso penalties can be used in conjunction with neural networks. For instance, penalizing the L_1-norm of the weight vector for a given layer will tend to encourage sparse representations that zero out uninformative nodes (Olshausen and Field 1997). Other sparsity-inducing methods are commonly deployed in DNNs to prevent against overfitting and stabilize the hidden layers of unsupervised autoencoders (Lee et al. 2008; Makhzani and Frey 2013). Indeed, the concept behind lasso penalties is extremely general, and can be used to regularize parameters in a wide array of algorithms.

The computational advantages of the lasso are not limited to machine learning problems, however. This regularization technique bears a striking resemblance to a process psychologists call *sensory gating*, i.e., the suppression of irrelevant stimuli in one's immediate phenomenal experience. Sensory gating prevents flooding of the higher cortical centers, which can make it difficult for agents to efficiently process information. Studies have shown that sensory gating is a fundamental aspect of early childhood development (Kisley et al. 2003). Gating deficiencies are associated with a wide range of psychiatric conditions, including epilepsy (Boutros et al. 2006), Alzheimer's Disease (Jessen et al. 2001), and schizophrenia (Bramon et al. 2004). Experiments conducted on animal and human subjects have revealed complex physiological underpinnings of gating behavior, which has been observed in single neurons as well as sensory, motor, and limbic subregions of the brain (Cromwell et al. 2008).

Lasso penalties have the same inhibitory effect on noisy variables that gating has on uninformative sensory inputs. Both methods thrive in complex systems where attention must be selectively apportioned. Just as a model that puts too much weight on irrelevant features will perform poorly on new datasets, so an individual who does not screen sensory data will struggle to function in new environments. Of course, there are major differences between the two. For instance, the lasso imposes a global penalty that simultaneously drives all parameters toward zero, while sensory gating is more selective in its screening mechanism. In this respect, sparsity-inducing Bayesian methods are perhaps more directly analogous to sensory gating. However, the overall effect is similar.

To the best of my knowledge, no research in lasso penalties has been explicitly motivated by connections to the cognitive process of sensory gating. Yet the success of this statistical technique can be at least partly explained by the fact that it implements a strategy that is essential to human intelligence.

5 Bagging

"Bagging" is a portmanteau of "bootstrap aggregating". The term was coined by Breiman (1996), whose seminal contributions to statistical learning include the original classification and regression tree (CART) algorithm (Breiman et al. 1984) as

well as random forests (Breiman 2001). Bagging is a prime example of ensemble learning, defined in Sect. 2. The method is completely general and can be used in conjunction with any base learner.

To bag the estimates of some model f, we simply average results across a large number of bootstrap samples. The generalization error of a bagged prediction can be easily estimated using the OOB samples randomly excluded from the individual draws. Recall from Sect. 2 that when bootstrapping, each observation has an approximately 36.8% exclusion probability. Thus, to calculate the error at a single data point, we restrict our attention to the $B^* \approx B/e$ basis functions in which it was not selected for training. By repeating this procedure across all n samples and averaging, we can efficiently compute an unbiased estimate of the ensemble's test error.

Bagging is most widely used with CART or some other tree-based algorithm as the base learner. One reason for this is that decision trees are unstable predictors—they are low-bias, high-variance models that benefit from bagging since overestimates and underestimates tend to cancel out over a sufficiently large number of bootstrap replicates. Bagging also smooths out the jagged decision boundaries and regression surfaces induced by recursive partitioning—the basis of all tree-based algorithms—which naturally produces step functions (see Fig. 6). As a practical note, bagging can take advantage of parallel processing power by distributing base learners across multiple cores, dramatically decreasing run time on modern machines.

Bagging is the key statistical innovation behind the random forest algorithm, one of the most popular techniques in all of supervised learning. Random forests have generated state of the art results in a number of quantitative disciplines, including genomics (Chen and Ishwaran 2012), econometrics (Mullainathan and Spiess 2017), and computational linguistics (Kontonatsios et al. 2014). The statistical theory underlying random forests and other bagged estimators has proven surprisingly difficult to develop, mostly due to tricky problems arising from the bootstrapping procedure. In fact, it is common for statisticians to prove theorems about a slightly modified version of the algorithm in which base learners are trained not on bootstrap samples, but rather on data subsamples—i.e., observations drawn randomly *without* replacement—which are more theoretically tractable (Mentch and Hooker 2016; Scornet et al. 2015; Wager and Athey 2018). However, bootstrapping tends to produce better results in practice, which is why the method remains popular among data scientists.

Although bagging is not typically used with neural networks, the method is often compared to a popular regularization technique for DNNs known as *dropout* (Hinton et al. 2012; Srivastava et al. 2014; Warde-Farley et al. 2013). The basic idea of dropout is simple: randomly exclude some proportion of units during each round of model training. This effectively creates an ensemble of subnetworks that share parameters, and predictions can be interpreted as outputs averaged across the ensemble. There are important differences between bagging and dropout—the former introduces randomness by bootstrapping observations, while the latter does so by sampling from the set of possible subnetworks—but the overall effect is similar. By combining the perspectives of numerous different models, the ensemble outperforms any of its constituent members.

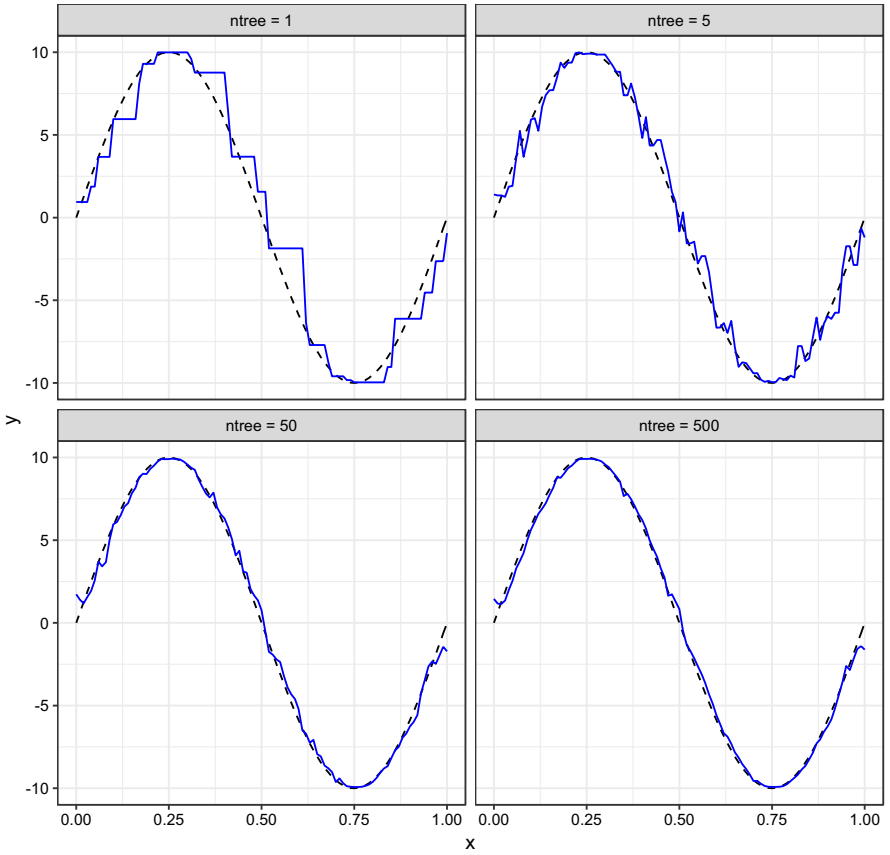

Fig. 6 Bagged estimates converging on a sine function as the number of trees in the ensemble increases

The success and broad applicability of bagging should come as no surprise to anyone familiar with the so-called "wisdom of crowds". Despite the resurgence of interest in aggregated decision-making due to web-enabled mass communication (Kittur and Kraut 2008), the basic concept at work here is in fact quite old. Condorcet's jury theorem (1785) states that any verdict reached by a set of independent and better than random jurors is more likely to be correct than the judgment of any individual juror. Moreover, the probability of a correct majority judgment approaches 1 as the jury size increases. Galton famously reported in 1907 that observers at a county fair accurately guessed the weight of an ox—not individually, but in aggregate, when their estimates were averaged (Galton 1907). Faith in humanity's collective wisdom arguably undergirds all free markets, where information from a variety of sources is efficiently combined to determine the fair price of assets (Fama 1965). Crowd sourcing has recently become popular in the natural sciences, where online enthusiasts have helped map the neural circuitry of

the mammalian retina (Kim et al. 2014) and discover new astronomical objects (Cardamone et al. 2009; Watson and Floridi 2018).

Just as lasso penalties mirror the process of sensory gating, bagging implements a computational version of this fundamental principle of social epistemology. By pooling the estimates of many better than random models (i.e., weak learners) we can create a single high-performing ensemble capable of modelling extremely complex systems with irregular decision boundaries and regression surfaces. In his book *The Wisdom of Crowds* (2004), Surowiecki identifies five criteria that he argues distinguish wise from irrational groups: (1) diversity of opinion; (2) independence; (3) decentralization; (4) aggregation; and (5) trust. Bagging meets all four criteria that are relevant in statistical applications. (It is hard to imagine how a base learner could "trust" the ensemble to be fair?) The random perturbations induced by bootstrapping ensure diversity across the submodels. Each sample is treated independently of all the rest, to the extent that base learners are often trained in parallel. The system is completely decentralized, with no global parameters governing the ensemble. Finally, aggregation is simple—voting in the case of classification and averaging in the case of regression.

6 Boosting

Boosting is another ensemble method, similar in some respects to bagging. However, whereas base learners in bagged models are fit independently of one another, boosting is a sequential procedure in which each model builds upon the last. Thus f_2 attempts to correct what f_1 got wrong, f_3 focuses on the errors of f_2, and so on. Much like bagging, boosting is a completely general approach that can work in principle with any combination of base learners. In practice, it is most often used with decision or regression trees.

Boosted predictions are made by summing across the individual basis functions. Stochastic gradient boosting, the most popular modern form of the algorithm, operates on bootstraps or subsamples of the original data to train each base learner, thereby enabling fast OOB estimation of the overall generalization error.

The first successful boosting algorithm was implemented by Freund and Shapire (1997). Their pioneering AdaBoost algorithm earned them the 2003 Gödel Prize, one of theoretical computer science's highest honors. Subsequent improvements by Friedman (2001, 2002) and more recently Chen and Guestrin (2016) have rendered boosting one of the most powerful methods in all of machine learning. The latter authors introduced XGBoost, an especially fast and scalable version that has gone on to dominate in a number of public data science competitions (Gorman 2017). Bayesian versions of boosting have also been developed (Chipman et al. 2010) with extensions to causal inference (Hahn et al. 2017; Hill 2011), survival analysis (Sparapani et al. 2016), and high-dimensional modeling (Linero 2018; Linero and Yang 2018).

The statistical properties of boosting have been difficult to establish, although solid progress has been made in the last decade, especially with regards to the original AdaBoost algorithm (Schapire and Freund 2012). Suggestive connections have been drawn between boosting and game theory (Freund and Schapire 1996), while

information geometric interpretations have opened up new lines of inquiry (Murata et al. 2004). Researchers in this area tend to follow a similar strategy to those who study bagging, relying on slight idealizations to derive convergence rates and other relevant theorems (Bühlmann and Hothorn 2007; Bühlmann and Yu 2003; Ehrlinger and Ishwaran 2012).

Boosting requires the careful calibration of several hyperparameters, which makes it somewhat less user-friendly than bagging. For instance, whereas bagged estimates do not degrade as the number of bootstraps B grows large, boosting is much more susceptible to overfitting. To offset against this, a shrinkage coefficient ν is often used to moderate the learning rate. The XGBoost algorithm includes a number of additional parameters that control everything from the overarching architecture of the model to the recursive partitioning subroutine. Bayesian methods introduce a number of extra parameters to define prior distributions, although sensible defaults have been shown to work well in a wide variety of settings. Cross-validating the optimal values for all these parameters can be time-consuming, but the extra effort is often worth it. Hastie et al. (2009) observe that boosting tends to dominate bagging in most applications.

The sequential nature of boosting bears some striking similarities to a process cognitive scientists call *predictive coding* (Rao and Ballard 1999). According to this theory, human perception is a dynamic inference problem in which the brain is constantly attempting to classify the objects of phenomenal experience and updating predictions based on new sensory information. In addition to its popularity as a model of information processing in the visual cortex (Huang and Rao 2011), predictive coding has also been extended to sensorimotor functions (Körding and Wolpert 2007) and mirror neuronal systems (Kilner et al. 2007). Some have argued that predictive coding provides a unified theory of cognition that applies to everything from perception and attention to reasoning and planning (Clark 2013). The process is often formalized along Bayesian lines, with current predictions serving as a prior distribution and new data providing a likelihood for dynamic updating (Friston 2009; Friston and Kiebel 2009). Predictive coding has also been conceptualized as a sort of backpropagation algorithm (Whittington and Bogacz 2019), in reference to the method by which neural network parameters are trained. In both routines, forward passes carry predictions and backward passes carry errors. Through iterative refinement, the system—biological or synthetic—attempts to converge on a set of maximally accurate predictions.

Bayesian and connectionist interpretations notwithstanding, I propose that boosting provides another helpful framework through which to understand predictive coding. The process begins with a single basis function fit to environmental stimuli. The resulting residual feedback becomes the target of a subsequent model, and the process repeats until convergence. Boosting has some practical advantages over Bayesian inference as a formal model for predictive coding. First, the former makes no parametric assumptions, which are often necessary to ensure the mathematical tractability of complex Bayesian updating procedures. Second, boosting with weak learners is more computationally efficient than integrating over high-dimensional distributions, an essential and time-consuming step for Bayesian inference with multiple input channels. Finally, boosting naturally strikes a data-adaptive balance

between individual basis functions, whereas Bayesian posteriors require a prior distribution to be defined upfront.

It is harder to distinguish between the statistical merits of boosting and backpropagation, since the two are basically just different implementations of the same optimization procedure, namely *gradient descent*. The gradient of a function is a vector of partial derivatives, with one entry for each parameter of interest. By taking steps proportional to the negative gradient at each point, we are guaranteed to find a local minimum of the function. Combined with the chain rule, this small bit of calculus forms the mathematical basis of backpropagation (Rumelhart et al. 1986), in which neural networks are trained by iteratively alternating between forward and backward passes to find the parameter values that jointly minimize a preselected (differentiable) loss function. In boosting, by contrast, we proceed in an additive fashion by fitting f_2 to the gradient of f_1, f_3 to the gradient of $f_1 + f_2$, and so on. In both cases, we gradually improve predictions by descending along the gradient of the loss function.

None of this is to say that the human brain literally implements a boosting procedure when engaged in predictive coding. However, I argue the prospect is at least as plausible as the Bayesian and connectionist alternatives that are currently popular in computational neuroscience. I suspect that all three models would tend to render similar results in most cases, especially as data accumulates. Moreover, there is no inconsistency between them. Neural networks can serve as basis functions for a boosted ensemble, and Bayesian variants of both algorithms are common. More interesting than the question of which model best explains predictive coding is the observation that all three are strong candidates, both individually and in combination. It is a strange and remarkable fact that these statistical methods developed on independent grounds have converged on formal procedures for modeling how the human brain processes sensory information.

7 Ethical Considerations

We have now reviewed a number of supervised learning algorithms that either deliberately or coincidentally mirror certain aspects of human cognition to varying degrees. In a sense, this is only to be expected. For better or worse, we are our own best source of inspiration when it comes to modelling intelligence. There is nothing especially remarkable or problematic about this.

However, issues arise when we begin to take these metaphors and analogies too literally. Recent years have seen AI deployed in a number of socially sensitive contexts, such as credit scoring, criminal justice, and military operations (Mittelstadt et al. 2016). These domains frequently involve high-stakes decisions with significant impact on the lives of those involved. Public and private institutions have traditionally relied upon human experts to adjudicate on matters of such extreme risk. This makes sense for at least three reasons. First, and most obviously, it is exceedingly important that we get these risky decisions right. Experts typically earn their title by demonstrating a tendency to minimize error. A second, closely related point is that we want to trust the reasoning that goes into important decisions. This amounts to an emphasis on process over product, a desire to ensure that there are no weak links

in the inferential chain connecting inputs and outputs. Finally, experts are an appropriate locus of moral responsibility. They are accountable agents deserving of praise or blame depending on the outcome of their actions.

To summarize, high risk decisions should ideally be made by (i) accurate, (ii) trustworthy, and (iii) responsible agents. Note that this is a normative claim about how expertise ought to work, not a descriptive claim about any particular class of purported experts. Of these three desiderata, AI can most plausibly be said to meet the first. Of course, the extent to which AI does in fact match or surpass human performance is an empirical question that must be handled on a case by case basis. Desideratum (iii), on the other hand, is a nonstarter. Algorithms may be causally responsible for any number of significant outcomes, but *moral* responsibility remains well beyond the ambit of even the most advanced machine learning program.

The prospects for desideratum (ii) are decidedly mixed. As we found in Sect. 3, trust cannot be guaranteed by mere accuracy alone, as high-performance models often fail in surprising ways. Proponents of algorithmic explainability are quick to point out that human experts are often unwilling or unable to articulate the reasoning behind their decisions. Human cognition is notoriously opaque (Carruthers 2011), not to mention irrational (Kahneman 2011). Yet despite some prominent arguments to the contrary (Kleinberg et al. 2019), it is not clear that automated decisions are much more accessible to external scrutiny. Putting aside the substantial issues surrounding intellectual property protections for copyrighted software (Pasquale 2015), we still face fundamental limits on our ability to trace the inductive reasoning of complex learning machines. Modern algorithms routinely contain millions of parameters describing subtle, nonlinear interactions. A frantic torrent of research in the last few years has sought to establish general-purpose tools for explainable AI [for recent surveys, see, e.g., Adadi and Berrada (2018) and Guidotti et al. (2018)], but several commentators have observed that the target of such investigations remains fundamentally underdetermined (Doshi-Velez and Kim 2017; Lipton 2016). Prominent post hoc approaches such as LIME (Ribeiro et al. 2016) and SHAP (Lundberg and Lee 2017), which find local linear approximations to a decision boundary or regression surface, rely on strong assumptions and come with no statistical guarantees. Globally transparent alternatives like SLIM (Ustun and Rudin 2017) and CORELS (Angelino et al. 2018) enjoy some desirable mathematical properties, but do not scale well with the number of features. There is reason to hope that advances in explainable AI will promote greater trust for algorithms in future. However, as long as high-performing models remain brittle, inefficient, and myopic, it seems rational to withhold judgment on just how trustworthy this technology really is, especially in high risk settings.

Perhaps we may relax these desiderata somewhat to accommodate new modes of trust and agency. For instance, Floridi and Sanders argue that ethical discourse has been "unduly constrained by its anthropocentric conception of agenthood" (2004, p. 350). They note that artificial agents (AAs) can be interactive, autonomous, and adaptable. Yet they readily concede that "it would be ridiculous to praise or blame an AA for its behaviour or charge it with a moral accusation" (p. 366), going on to clarify that AAs instantiate a form of "a responsible morality" (p. 364). In a similar vein, Taddeo (2010) argues that AAs can earn one another's *e-trust*, an emergent

second-order property arising in distributed digital systems with certain first-order relational properties. This phenomenon is not to be confused with old-fashioned human trust, a considerably messier affair that cannot be adequately modelled by neat mathematical functions or the formal apparatus of rational choice theory. These views are consonant with my remarks above that DNNs exhibit a novel kind of intelligence, similar in some respects but far from identical to the human original.

However, I am skeptical that these modified notions of agency and trust are sufficient to upgrade AI to the level required for high-stakes decision making, or indeed that many of the algorithms currently in use even meet these watered-down desiderata. I submit that our willingness to cede ever more authority to AAs derives primarily from their accuracy, and collaterally from our anthropomorphic impulse to conflate desiderata (i)–(iii). For better or worse, humans with an impressive track record of accurate judgments in some particular domain are typically regarded as trustworthy and responsible as well, at least with respect to their given area of expertise. Thus we falsely impute these latter values to the machine when its performance begins to match or exceed that of human experts. This is just another example of the well-documented cleaving power of the digital (Floridi 2017), which regularly decouples features of the world that have always been indivisible, such as location and presence, or law and territoriality. Just because we believe that accurate decisions are often made by trustworthy, responsible humans does not necessarily entail any inherent link between these traits.

In a 2011 article entitled "Anthropomorphism and AI", Proudfoot concludes that her eponymous conjuncts are inseparable. Acutely aware of the epistemological and metaphysical confusions that arise from conflating human and machine intelligence, she recommends that "anthropomorphism be *managed* rather than purged" (2011, p. 952) from AI research. The proliferation of automated decision-making systems in socially sensitive contexts adds moral urgency to her plea, and vividly demonstrates how the rhetoric of anthropomorphism has vastly outpaced the reality of contemporary AI. Algorithms are not "just like us" and the temptation to pretend they are can have profound ethical consequences when they are deployed in high-risk domains like finance (Eubanks 2018) and clinical medicine (Watson et al. 2019). By anthropomorphizing a statistical model, we implicitly grant it a degree of agency that not only overstates its true abilities, but robs us of our own autonomy.

Algorithms can only exercise their (artificial) agency as a result of a socially constructed context in which we have deliberately outsourced some task to the machine. This may be more or less reasonable in different situations. Software for filtering spam emails is probably unobjectionable; automated systems for criminal sentencing, on the other hand, raise legitimate concerns about the nature and meaning of justice in an information society. In any event, the central point—one as obvious as it is frequently overlooked—is that it is always *humans* who choose whether or not to abdicate this authority, to empower some piece of technology to intervene on our behalf. It would be a mistake to presume that this transfer of authority involves a simultaneous absolution of responsibility. It does not. The rhetoric of anthropomorphism in AI may be helpful when explaining complex models to audiences with minimal background in statistics and computer science. It is misleading and potentially dangerous, however, when used to guide (or cloud) our ethical judgment.

A more thoughtful and comprehensive approach to conceptualizing the ethical challenges posed by AI requires a proper understanding not just of how these algorithms work—their strengths and weaknesses, their capabilities and limits—but of how they fit into a larger sociotechnical framework. The anthropomorphic impulse, so pervasive in the discourse on AI, is decidedly unhelpful in this regard.

8 Conclusion

There is no denying that some of the most innovative achievements in contemporary machine learning are directly or indirectly inspired by prominent theories of neuroscience, cognitive psychology, and social epistemology. Experts and laypeople alike actively promote the notion that these technologies are humanlike in their ability to find and exploit patterns in data. Yet the tendency to focus on structural affinities between biological and artificial neural networks suggests a mechanistic interpretation of intelligence that fails to account for functional complexities. I have argued that the extent to which modern algorithms mimic human intelligence is overstated in at least one prominent instance, but also underappreciated in other less familiar cases. Borders between these methods are somewhat fluid, as they can often be used in combination with one another. In each case, anthropomorphic analogies can help to frame learning strategies and even inspire novel approaches to AI research.

However, we must be cautious in our rhetoric. The anthropomorphic tendency in AI is not ethically neutral. The temptation to grant algorithms decision-making authority in socially sensitive applications threatens to undermine our ability to hold powerful individuals and groups accountable for their technologically-mediated actions. Supervised learning provides society with some of its most powerful tools—and like all tools, they can be used either to help or to harm. The choice, as ever, is ours.

Acknowledgements Thanks to Silvia Milano, Christopher Burr, Eveliina Kuitunen, Carl Öhman, and David Kinney for their thoughtful comments on earlier drafts of this manuscript. A version of this material was originally presented to the Oxford Digital Ethics Lab, where I also received helpful feedback. Lastly, I would like to sincerely thank anonymous reviewer #1 for their exceptionally thorough reading and valuable contributions.

References

Adadi, A., & Berrada, M. (2018). Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access, 6,* 52138–52160.

Angelino, E., Larus-Stone, N., Alabi, D., Seltzer, M., & Rudin, C. (2018). Learning certifiably optimal rule lists for categorical data. *The Journal of Machine Learning Research, 18*(234), 1–78.

Bourdakos, N. (2017). *Capsule networks are shaking up AI*. Retrieved April 3, 2019 from https://hackernoon.com/capsule-networks-are-shaking-up-ai-heres-how-to-use-them-c233a0971952.

Boutros, N. N., Trautner, P., Korzyukov, O., Grunwald, T., Burroughs, S., Elger, C. E., … Rosburg, T. (2006). Mid-latency auditory-evoked responses and sensory gating in focal epilepsy: A preliminary exploration. *The Journal of Neuropsychiatry and Clinical Neurosciences*, *18*(3), 409–416.

Bramon, E., Rabe-Hesketh, S., Sham, P., Murray, R. M., & Frangou, S. (2004). Meta-analysis of the P300 and P50 waveforms in schizophrenia. *Schizophrenia Research, 70*(2), 315–329.

Breiman, L. (1996). Bagging predictors. *Machine Learning, 24*(2), 123–140.

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 1–33.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. Boca Raton: Taylor & Francis.

Brown, T. B., Mané, D., Roy, A., Abadi, M., & Gilmer, J. (2017). *Adversarial patch*. https://arxiv.org/abs/1712.09665.

Buckner, C. (2018). Empiricism without magic: Transformational abstraction in deep convolutional neural networks. *Synthese, 195*(12), 5339–5372.

Buckner, C., & Garson, J. (2019). Connectionism. In E. N. Zalta (Ed.), *The stanford encyclopedia of philosophy (Fall 2019)*. Stanford: Metaphysics Research Lab, Stanford University.

Bühlmann, P., & Hothorn, T. (2007). Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science, 22*(4), 477–505.

Bühlmann, P., & van de Geer, S. (2011). *Statistics for high-dimensional data: Methods, theory and applications*. Berlin: Springer.

Bühlmann, P., & Yu, B. (2003). Boosting with the l_2 loss: Regression and classification. *Journal of American Statistical Association, 98*(462), 324–339.

Cardamone, C., Schawinski, K., Sarzi, M., Bamford, S. P., Bennert, N., Urry, C. M., … VandenBerg, J. (2009). Galaxy zoo green peas: Discovery of a class of compact extremely star-forming galaxies. *Monthly Notices of the Royal Astronomical Society*, *399*(3), 1191–1205.

Carruthers, P. (2011). *The opacity of mind: An integrative theory of self-knowledge*. Oxford: Oxford University Press.

Carvalho, C. M., Polson, N. G., & Scott, J. G. (2010). The horseshoe estimator for sparse signals. *Biometrika, 97*(2), 465–480.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22Nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 785–794).

Chen, X., & Ishwaran, H. (2012). Random forests for genomic data analysis. *Genomics, 99*(6), 323–329.

Chipman, H. A., George, E. I., & McCulloch, R. E. (2010). BART: Bayesian additive regression trees. *The Annal of Applied Statistics, 4*(1), 266–298.

Cichy, R. M., Khosla, A., Pantazis, D., Torralba, A., & Oliva, A. (2016). Comparison of deep neural networks to spatio-temporal cortical dynamics of human visual object recognition reveals hierarchical correspondence. *Scientific Reports, 6,* 27755.

Clark, A. (2013). Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behavioral and Brain Sciences, 36*(3), 181–204.

Collier, M., & Beel, J. (2018). Implementing neural turing machines. In V. Kůrková, Y. Manolopoulos, B. Hammer, L. Iliadis, & I. Maglogiannis (Eds.), *Artificial neural networks and machine learning—ICANN 2018*. Cham: Springer International Publishing.

Condorcet, N. (1785). *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. Paris: Imprimerie Royale.

Cromwell, H. C., Mears, R. P., Wan, L., & Boutros, N. N. (2008). Sensory gating: A translational effort from basic to clinical science. *Clinical EEG and Neuroscience, 39*(2), 69–72.

Dahl, G. E., Yu, D., Deng, L., & Acero, A. (2012). Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech and Language Processing, 20*(1), 30–42.

Doshi-Velez, F., & Kim, B. (2017). *Towards a rigorous science of interpretable machine learning*. http://arxiv.org/abs/1702.08608.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics, 7*(1), 1–26.

Ehrlinger, J., & Ishwaran, H. (2012). Characterizing L2-boosting. *The Annals of Statistics, 40*(2), 1074–1101.

Elsayed, G. F., Shankar, S., Cheung, B., Papernot, N., Kurakin, A., Goodfellow, I., & Sohl-Dickstein, J. (2018). Adversarial examples that fool both computer vision and time-limited humans. In *Proceedings of the 32nd international conference on neural information processing systems* (pp. 3914–3924).

Engstrom, L., Gilmer, J., Goh, G., Hendrycks, D., Ilyas, A., Madry, A., … Wallace, E. (2019). A discussion of "Adversarial Examples Are Not Bugs, They Are Features." *Distill*. https://doi.org/10.23915/distill.00019.

Eubanks, V. (2018). *Automating inequality: How high-tech tools profile, police, and punish the poor*. New York: St. Martin's Press.

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., … Song, D. X. (2018). Robust physical-world attacks on deep learning visual classification. In *2018 IEEE/CVF conference on computer vision and pattern recognition* (pp. 1625–1634).

Fama, E. F. (1965). The behavior of stock-market prices. *The Journal of Business, 38*(1), 34–105.

Finlayson, S. G., Bowers, J. D., Ito, J., Zittrain, J. L., Beam, A. L., & Kohane, I. S. (2019). Adversarial attacks on medical machine learning. *Science, 363*(6433), 1287–1289.

Floridi, L. (2017). Digital's cleaving power and its consequences. *Philosophy & Technology, 30*(2), 123–129.

Floridi, L., & Sanders, J. W. (2004). On the morality of artificial agents. *Minds and Machines, 14*(3), 349–379.

Freund, Y., & Schapire, R. E. (1996). Game theory, on-line prediction and boosting. In *Proceedings of the ninth annual conference on computational learning theory* (pp. 325–332).

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences, 55*(1), 119–139.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics, 29*(5), 1189–1232.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis, 38*(4), 367–378.

Friedman, J., Hastie, T., & Tibshirani, R. (2007). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics, 9*(3), 432–441.

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software, 33*(1), 1–41.

Friston, K. (2009). The free-energy principle: A rough guide to the brain? *Trends in Cognitive Sciences, 13*(7), 293–301.

Friston, K., & Kiebel, S. (2009). Predictive coding under the free-energy principle. *Philosophical Transactions of the Royal Society B Biological Sciences, 364*(1521), 1211–1221.

Galton, F. (1907). Vox Populi. *Nature, 75*(1949), 450–451.

Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep Sparse Rectifier Neural Networks. In G. Gordon, D. Dunson, & M. Dudík (Eds.), *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 315–323).

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., … Bengio, Y. (2014). Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems 27* (pp. 2672–2680).

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). *Explaining and harnessing adversarial examples*. http://arxiv.org/abs/1702.08608.

Gorman, B. (2017). A Kaggle master explains gradient boosting. *Kaggle Blog*. Retrieved from http://blog.kaggle.com/2017/01/23/a-kaggle-master-explains-gradient-boosting/.

Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., … Hassabis, D. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, *538*, 471.

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM Computing Surveys, 51*(5), 1–42.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning, 46*(1), 389–422.

Hahn, R. P., Murray, J. S., & Carvalho, C. M. (2017). *Bayesian regression tree models for causal inference: Regularization, confounding, and heterogeneous effects*. https://arxiv.org/abs/1706.09523.

Hassabis, D., Kumaran, D., Summerfield, C., & Botvinick, M. (2017). Neuroscience-inspired artificial intelligence. *Neuron, 95*(2), 245–258.

Hastie, T. & Qian, J. (2014). *Glmnet vignette*. Retrieved from: https://web.stanford.edu/~hastie/glmnet/glmnet_alpha.html.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction*. New York: Springer.

Hill, J. L. (2011). Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics, 20*(1), 217–240.
Hinton, G. E., Krizhevsky, A., & Wang, S. D. (2011). Transforming Auto-Encoders. In T. Honkela, W. Duch, M. Girolami, & S. Kaski (Eds.), *Artificial neural networks and machine learning—ICANN 2011* (pp. 44–51). Berlin: Springer.
Hinton, G. E., Sabour, S., & Frosst, N. (2018). Matrix capsules with EM routing. In *International conference on learning representations*. Retrieved from https://openreview.net/forum?id=HJWLfGWRb.
Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). *Improving neural networks by preventing co-adaptation of feature detectors*. https://arxiv.org/abs/1207.0580.
Huang, Y., & Rao, R. P. N. (2011). Predictive coding. *Wiley Interdisciplinary Reviews: Cognitive Science, 2*(5), 580–593.
Hubel, D., & Wiesel, T. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology, 160*(1), 106–154.
Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., & Madry, A. (2019). *Adversarial examples are not bugs, they are features*. https://arxiv.org/abs/1905.02175.
Ishwaran, H., & Rao, J. S. (2005). Spike and slab variable selection: Frequentist and Bayesian strategies. *The Annals of Statistics, 33*(2), 730–773.
Jessen, F., Kucharski, C., Fries, T., Papassotiropoulos, A., Hoenig, K., Maier, W., et al. (2001). Sensory gating deficit expressed by a disturbed suppression of the P50 event-related potential in patients with Alzheimer's disease. *American Journal of Psychiatry, 158*(8), 1319–1321.
Jolliffe, I. T. (2002). *Principal component analysis*. New York: Springer.
Kahneman, D. (2011). *Thinking, fast and slow*. New York: Penguin.
Kilner, J. M., Friston, K. J., & Frith, C. D. (2007). Predictive coding: An account of the mirror neuron system. *Cognitive Processing, 8*(3), 159–166.
Kim, J. S., Greene, M. J., Zlateski, A., Lee, K., Richardson, M., Turaga, S. C., … EyeWirers, the. (2014). Space–time wiring specificity supports direction selectivity in the retina. *Nature*, *509*, 331.
Kisley, M. A., Polk, S. D., Ross, R. G., Levisohn, P. M., & Freedman, R. (2003). Early postnatal development of sensory gating. *NeuroReport, 14*(5), 693–697.
Kittur, A., & Kraut, R. E. (2008). Harnessing the wisdom of crowds in wikipedia: Quality through coordination. In *Proceedings of the 2008 ACM conference on computer supported cooperative work* (pp. 37–46).
Kleinberg, J., Ludwig, J., Mullainathan, S., & Sunstein, C. R. (2019). Discrimination in the age of algorithms. *Journal of Legal Analysis*. https://doi.org/10.1093/jla/laz001.
Kontonatsios, G., Korkontzelos, I., Tsujii, J., & Ananiadou, S. (2014). Using a random forest classifier to compile bilingual dictionaries of technical terms from comparable corpora. In *Proceedings of the 14th conference of the european chapter of the association for computational linguistics, Vol. 2: Short papers* (pp. 111–116).
Körding, K., & Wolpert, D. (2007). Bayesian statistics and utility functions in sensorimotor control. In K. Doya, S. Ishii, A. Pouget, & R. Rao (Eds.), *Bayesian brain: Probabilistic approaches to neural coding* (pp. 299–320). Cambridge: MIT Press.
Kriegeskorte, N. (2015). Deep neural networks: A new framework for modeling biological vision and brain information processing. *Annual Review of Vision Science, 1*(1), 417–446.
Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Proceedings of the 25th international conference on neural information processing systems—Vol. 1* (pp. 1097–1105).
LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.
LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.
Lee, H., Ekanadham, C., & Ng, A. Y. (2008). Sparse deep belief net model for visual area V2. In J. C. Platt, D. Koller, Y. Singer, & S. T. Roweis (Eds.), *Advances in neural information processing systems 20* (pp. 873–880).
Linero, A. R. (2018). Bayesian regression trees for high-dimensional prediction and variable selection. *Journal of American Statistical Association, 113*(522), 626–636.
Linero, A. R., & Yang, Y. (2018). Bayesian regression tree ensembles that adapt to smoothness and sparsity. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 80*(5), 1087–1110.
Lipton, Z. C. (2016). *The mythos of model interpretability*. https://arxiv.org/abs/1606.03490.

Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. In I Guyon, U. V Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 30* (pp. 4765–4774).
Makhzani, A., & Frey, B. (2013). *k-Sparse autoencoders*. https://arxiv.org/abs/1312.5663.
Marcus, G. (2018). *Deep learning: A critical appraisal*. https://arxiv.org/abs/1312.6197.
Mentch, L., & Hooker, G. (2016). Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. *The Journal of Machine Learning Research, 17*(1), 841–881.
Mittelstadt, B. D., Allo, P., Taddeo, M., Wachter, S., & Floridi, L. (2016). The ethics of algorithms: Mapping the debate. *Big Data & Society.*. https://doi.org/10.1177/2053951716679679.
Mohamed, A., Dahl, G. E., & Hinton, G. (2012). Acoustic modeling using deep belief networks. *IEEE Transactions on Audio, Speech, and Language Processing, 20*(1), 14–22.
Mullainathan, S., & Spiess, J. (2017). Machine learning: An applied econometric approach. *Journal of Economic Perspectives, 31*(2), 87–106.
Murata, N., Takenouchi, T., Kanamori, T., & Eguchi, S. (2004). Information geometry of U-boost and bregman divergence. *Neural Computation, 16*(7), 1437–1481.
New Navy Device Learns by Doing. (1958, July 8). *New York Times*, p. 25.
Olah, C., Satyanarayan, A., Johnson, I., Carter, S., Schubert, L., Ye, K., et al. (2018). *The building blocks of interpretability. Distill.*. https://doi.org/10.23915/distill.00010.
Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research, 37*(23), 3311–3325.
Pasquale, F. (2015). *The black box society*. Cambridge: Harvard University Press.
Perez, L., & Wang, J. (2017). *The effectiveness of data augmentation in image classification using deep learning*. arXiv preprint, 1712.04621.
Proudfoot, D. (2011). Anthropomorphism and AI: Turing's much misunderstood imitation game. *Artificial Intelligence, 175*(5), 950–957.
Raina, R., Madhavan, A., & Ng, A. Y. (2009). Large-scale deep unsupervised learning using graphics processors. In *Proceedings of the 26th annual international conference on machine learning* (pp. 873–880).
Rao, R. P. N., & Ballard, D. H. (1999). Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience, 2*(1), 79–87.
Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why Should I Trust You?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135–1144).
Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review, 65,* 386–408.
Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature, 323*(6088), 533–536.
Sabour, S., Frosst, N., & Hinton, G. E. (2017). Dynamic Routing Between Capsules. In I Guyon, U. V Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 30* (pp. 3856–3866).
Schapire, R. E., & Freund, Y. (2012). *Boosting: Foundations and algorithms*. Camridge: MIT Press.
Scornet, E., Biau, G., & Vert, J. P. (2015). Consistency of random forests. *The Annals of Statistics, 43*(4), 1716–1741.
Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., … Hassabis, D. (2018). A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, *362*(6419), 1140–1144.
Sparapani, R. A., Logan, B. R., McCulloch, R. E., & Laud, P. W. (2016). Nonparametric survival analysis using Bayesian Additive Regression Trees (BART). *Statistics in Medicine, 35*(16), 2741–2753.
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research, 15,* 1929–1958.
Stinson, C. (2016). Mechanisms in psychology: Ripping nature at its seams. *Synthese, 193*(5), 1585–1614.
Strogatz, S. (2018, December 26). One giant step for a chess-playing machine. *New York Times*. Retrieved from https://www.nytimes.com/2018/12/26/science/chess-artificial-intelligence.html?rref=collection%2Ftimestopic%2FArtificialIntelligence.
Surowiecki, J. (2004). *The wisdom of crowds*. New York: Doubleday.
Sutton, R., & Barto, A. (2018). *Reinforcement learning: An introduction* (2nd ed.). Cambridge: MIT Press.

Taddeo, M. (2010). Modelling trust in artificial agents, a first step toward the analysis of e-trust. *Minds and Machines, 20*(2), 243–257.

Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. (2014). DeepFace: Closing the gap to human-level performance in face verification. *IEEE Conference on Computer Vision and Pattern Recognition, 2014,* 1701–1708.

Team, S. (2017). Hey Siri: An on-device DNN-powered voice trigger for apple's personal assistant. *Apple Machine Learning Journal*, *1*(6). https://machinelearning.apple.com/2017/10/01/hey-siri.html.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological), 58*(1), 267–288.

Turing, A. (1950). Computing machinery and intelligence. *Mind, LIX*(236), 433–460.

Ustun, B., & Rudin, C. (2017). Optimized risk scores. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1125–1134).

van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research, 9,* 2579–2605.

Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., & Wierstra, D. (2016). *Matching networks for one shot learning*. https://arxiv.org/abs/1606.04080

Wager, S., & Athey, S. (2018). Estimation and inference of heterogeneous treatment effects using random forests. *Journal of American Statistical Association, 113*(523), 1228–1242.

Waldrop, M. M. (2019). News feature: What are the limits of deep learning? *Proceedings of the National Academy of Sciences, 116*(4), 1074–1077.

Warde-Farley, D., Goodfellow, I. J., Courville, A., & Bengio, Y. (2013). *An empirical analysis of dropout in piecewise linear networks*. https://arxiv.org/abs/1312.6197.

Watson, D., & Floridi, L. (2018). Crowdsourced science: Sociotechnical epistemology in the e-research paradigm. *Synthese, 195*(2), 741–764.

Watson, D., Krutzinna, J., Bruce, I. N., Griffiths, C. E. M., McInnes, I. B., Barnes, M. R., et al. (2019). Clinical applications of machine learning algorithms: Beyond the black box. *BMJ, 364,* l886.

Whittington, J. C. R., & Bogacz, R. (2019). Theories of error back-propagation in the brain. *Trends in Cognitive Sciences, 23*(3), 235–250.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V, Norouzi, M., Macherey, W., … Dean, J. (2016). *Google's neural machine translation system: Bridging the gap between human and machine translation*.

Yamins, D. L. K., & DiCarlo, J. J. (2016). Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience, 19,* 356.

Zhou, Z., & Firestone, C. (2019). Humans can decipher adversarial images. *Nature Communications, 10*(1), 1334.

Zhu, J.-Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE international conference on computer vision (ICCV)*.