Submitted to *Journal of Vision.*

EMERGING TREND IN VISION SCIENCE

Deep learning: Using machine learning to study biological vision

Najib J. Majaj[1] and Denis G. Pelli[1,2]

[1]Center for Neural Science and [2]Department of Psychology, New York University

ABSTRACT

Today most vision-science presentations mention machine learning. Many neuroscientists use machine learning to decode neural responses. Many perception scientists try to understand recognition by living organisms. To them, machine learning offers a reference of attainable performance based on learned stimuli. This brief overview of the use of machine learning in biological vision touches on its strengths, weaknesses, milestones, controversies, and current directions.

INTRODUCTION

What does machine learning offer to biological-vision scientists? We suppose that most of our readers have heard of machine learning but are wondering how to interpret machine-learning results and whether it would be useful in their own research. We begin by naming some of its pluses and minuses.

PLUSES: WHAT IT'S GOOD FOR

Deep learning is the latest phase of machine learning, and is becoming very popular (Fig. 1). Is it just a fad? At the very least, machine learning is a powerful tool for interpreting biological data. For computer vision, the old paradigm was: feature detection, segmentation, and grouping (Marr, 1982). The new paradigm defines just the task and a feature set, and machine learning builds the classifier from a training set. Unlike the handcrafted pattern recognition (including segmentation and grouping) popular in the 70's and 80's, machine-learning algorithms are generic, with little domain-specificity. They replace hand-engineered feature detectors with filters that can be learned from the data. Advances in the mid 90's in machine learning made statistical learning theory useful for practical classification, e.g. handwriting recognition (Vapnik, 1999).

GLOSSARY

Machine learning is a computer algorithm that uses data from the environment to improve performance of a task.

Deep learning is the latest version of machine learning, distinguished by having more than three layers. It is ubiquitous in the internet.

Supervised learning refers to any algorithm that accepts a set of labeled stimuli — a training set — and returns a classifier that can label stimuli similar to those in the training set.

Unsupervised learning works without labels. It is less popular, but of great interest because labeled data are scarce while unlabeled data are plentiful. Without labels, the algorithm discovers structure and redundancy in the data.

Cost function. A function that assigns a real number representing cost to a candidate solution. Solving by optimization means minimizing cost.

Gradient descent: An algorithm that minimizes cost by incrementally changing the parameters in the direction of steepest descent of the cost function.

Convexity: A problem is convex if there are no local minima competing with the global minimum. In optimization, a convex cost function guarantees that gradient descent will always find the global minimum.

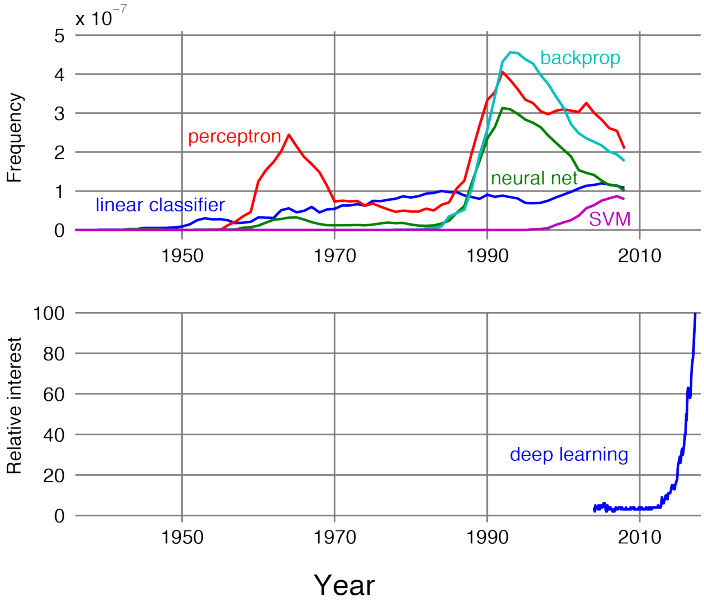

Figure 1. Top: The frequency of appearance of each of five terms — “linear classifier”, "perceptron", "support vector machine", “neural net” and “backprop” — in books indexed by Google in each year of publication. Frequency is reported as a fraction of all instances of that number of words (1,2, or 3) normalized by the number of books published that year (ngram/year/books published). The figure was created using Google’s n-gram viewer (https://books.google.com/ngrams), which contains a yearly count of n-grams found in sources printed between 1500 and 2008. **Bottom:** Numbers represent worldwide search interest relative to the highest point on the chart for the given year for the term “deep learning” (as reported by https://trends.google.com/trends/).

Cross validation assesses how well a classifier generalizes. Usually the training and test stimuli are chosen to be identically-distributed independent samples.

Backprop, short for "backward propagation of errors", is widely used to apply gradient-descent learning to multi-layer networks. It uses the chain rule from calculus to iteratively compute the gradient of the cost function for each layer.

Hebbian learning and spike-timing dependent plasticity (***STDP***). According to Hebb’s rule, the efficiency of a synapse increases after correlated pre- and post-synaptic activity. In other words, neurons that fire together, wire together (Löwel & Singer, 1992).

Support Vector Machine (SVM) is a learning machine for classification. SVMs generalize well. An SVM can quickly learn to perform a nonlinear classification using what is called the “kernel trick”, mapping its input into a high-dimensional feature space (Cortes & Vapnik, 1999).

Convolutional neural networks (ConvNets) have their roots in the Neocognitron (Fukushima 1980) and are inspired by the simple and complex cells described by Hubel and Wiesel (1962). ConvNets apply backprop learning to multilayer neural networks based on convolution and pooling (LeCun et al., 1989; LeCun et al., 1990; LeCun et al., 1998).

Machine learning allows a neurophysiologist to decode neural activity without knowing the receptive fields (Seung & Sompolinsky,1993; Hung et al., 2005). Machine learning shifts the emphasis from how the cells encode to what they encode, i.e. from how they encode the stimulus to what that code tells us about the stimulus. Mapping a receptive field is the foundation of neuroscience (beginning with Weber’s 1834/1996 mapping of tactile “sensory circles”), but many young scientists are impatient with the limitations of single-cell recording: looking for minutes or hours at how one cell responds to each of perhaps a hundred different stimuli. New neuroscientists

are the first generation for whom it is patently clear that characterization of a single neuron's receptive field, which was invaluable in the retina and V1, fails to characterize how higher visual areas encode the stimulus. Statistical learning techniques reveal "how neuronal responses can best be used (combined) to inform perceptual decision-making" (Graf, Kohn, Jazayeri, & Movshon, 2010).

For psychophysics, Signal Detection Theory (SDT) proved that the optimal classifier for a signal in noise is a template matcher (Peterson, Birdsall, & Fox, 1954; Tanner & Birdsall, 1958). SDT has been a very useful reference in interpreting human psychophysical performance (e.g. Geisler, 1989; Pelli et al., 2006). However, it provides no account of learning. Machine learning shows promise of guiding today's investigations of human learning and may reveal the constraints imposed by the training set on learning. It can be hard to tell whether behavioral performance is limited by the set of stimuli, or the neural representation, or the mismatch between the neural decision process and the stimulus and task. Implications for classification performance are not readily apparent from direct inspection of families of stimuli and their neural responses. SDT specifies optimal performance for classification of known signals, but does not tell us how to generalize beyond a training set. Machine learning does.

MINUSES: COMMON COMPLAINTS

Some biologists complain that neural nets do not match what we know about neurons (Crick, 1989; Rubinov, 2015). In particular, it is not clear, given what we know about neurons and neural plasticity, whether a backpropagation network can be implemented using biologically plausible circuits (but see Mazzoni et al., 1991, and Bengio et al., 2015). With a different, perspective, engineers and computer scientists, though inspired by biological vision, focus on what works.

Some biological modelers complain that neural nets have alarmingly many parameters. Deep neural networks continue to be opaque, especially if the problem is not known to be convex. Before neural-network modeling, a model was simpler than the data it explained. Deep neural nets are typically as complex as the data, and the solutions are hard to visualize (but see Zeiler & Fergus, 2013). However, while the training sets and learned weights are long lists, the generative rules for the network (the computer programs) are short.

Some cognitive psychologists dismiss deep neural networks as unable to "master some of the basic things that children do, like learning the past tense of a regular verb" (Marcus et al., 1992).

Some statisticians worry that rigorous statistical tools are being displaced by machine learning, which lacks rigor (Friedman, 1998; Matloff, 2014, but see Breiman, 2001; Efron & Hastie, 2016). Assumptions are rarely stated. There are no confidence intervals on the solution. However, performance is typically cross-validated, showing generalization, and it has been proven that convex networks can compute posterior probability (Rojas, 1996).

Some of the best classifiers in computer science were inspired by biological principles (Rosenblatt, 1957; 1958; Rumelhart et al., 1986; LeCun, 1985; LeCun et al. 1989; LeCun et al. 1990; Riesenhuber & Poggio, 1999; and see LeCun, Bengio, Hinton 2015). Those classifiers are now so good that they occasionally exceed human performance and might serve as models for how biological systems classify (e.g. Ziskind, Hénaff, LeCun, & Pelli, 2014).

MATHEMATICS VS. ENGINEERING

The history of machine learning has two threads: mathematics and engineering. In the *mathematical* thread, two statisticians, Fisher and later Vapnik, developed mathematical transformations to uncover categories in data, and proved that they give unique answers. They assumed distributions and proved convergence.

In the *engineering* thread, a loose coalition of psychologists, neuroscientists, and computer scientists (e.g. Rosenblatt, Minsky, Fukushima, Hinton, Sejnowski, LeCun, Poggio) sought to reverse-engineer the brain to build a machine that learns. Their algorithms are typically applied to stimuli with unknown distributions and lack proofs of convergence.

MILESTONES IN CLASSIFICATION

1936: Linear discriminant analysis

1953: Machine learning

1958: Perceptron

1969: Death of the perceptron

1974: Backprop

1980: Neocognitron

1987: NETtalk

1989: ConvNets

1995: Support Vector Machine (SVM)

2006: Backprop, revived

2012: Deep learning

1936: Linear discriminant analysis. Fisher (1936) introduced linear discriminant analysis to classify two species of iris flower based on four measurements per flower. When the distribution of the measurements is normal and the covariance matrix between the measurements is known,

linear discriminant analysis answers the question: Supposing we use a single-valued function to classify, what linear function $y = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4$, of four measurements x_1, x_2, x_3, x_4 made on flowers, with free weights w_1, w_2, w_3, w_4, will maximize the ratio of the difference between the means to the standard deviations within species?[1] Linear classifiers are great for simple problems for which the category boundary is a hyperplane in a small number of dimensions. However, complex problems like object recognition typically require more complex category boundaries in a large number of dimensions. Furthermore, the distributions of the features are typically unknown and not necessarily normal.

Cortes & Vapnik (1995) note that the first algorithm for pattern recognition was Fisher's optimal decision function for classifying vectors from two known distributions. Fisher solved for the optimal classifier in the presence of gaussian noise and known covariance between elements of the vector. When the covariances are equal, this reduces to a linear classifier. The ideal template matcher of signal detection theory is an example of such a linear classifier (Peterson et al., 1954). This fully specified simple problem can be solved analytically. Of course, many important problems are not fully specified. In everyday perceptual tasks, we typically know only a "training" set of samples and labels.

1953: Machine learning. The first developments in machine learning were to play chess and checkers. "Could one make a machine to play chess, and to improve its play, game by game, profiting from its experience?" (Turing, 1953). Arthur Samuel defined *machine learning* as the "Field of study that gives computers the ability to learn without being explicitly programmed." (Samuel, 1959)

[1] Linear discriminant analysis is an outgrowth of regression which has a much longer history. Regression is the optimal least-squares linear combination of given functions to fit given data and was applied by Legendre (1805) and Gauss (1809) to astronomical data to determine the orbits of the comets and planets around the sun. The estimates come with confidence intervals and the fraction of variance accounted for, which rates the goodness of the explanation.

1958: Perceptron. Inspired by physiologically measured receptive fields, Rosenblatt (1958) showed that a very simple neural network, the perceptron, could learn to classify from training samples. Perceptrons combined several linear classifiers to implement piecewise-linear separating surfaces. The perceptron learns the weights to use in a linear combination of feature-detector outputs. The perceptron transforms the stimulus into a feature vector and then applies a linear classifier to the feature vector. The perceptron is piecewise linear and has the ability to learn from training examples without knowing the full distribution of the stimuli. Only the final layer in the perceptron learns.

1969: Death of the perceptron. However, it quickly became apparent that the perceptron and other single-layer neural networks cannot learn tasks that are not linearly separable, i.e. cannot solve problems like connectivity (Are all elements connected?) and parity (Is the number of elements odd or even?); people solve these readily (Minsky & Papert, 1969). On this basis they announced the death of artificial neural networks.

1974: Backprop. The death of the perceptron showed that learning in a one-layer network was too limited. This impasse was broken by the introduction of the backprop algorithm, which allowed learning to propagate through multiple-layer neural networks. The history of backprop is complicated (see Schmidhuber, 2015). The idea of minimisation of error through a differentiable multi-stage network was discussed as early as the 1960s (e.g. Bryson, Denham, & Dreyfus, 1963). It was applied to artificial neural networks in the 1970s (e.g. Werbos, 1974). In the 1980s, efficient backprop first gained recognition, and led to a renaissance in the field of artificial neural network research (LeCun, 1985; Rumelhart, Hinton, & Williams, 1986). During the 2000s backprop neural networks fell out of favor, due to four limitations (Vapnik, 1999): **1. *No proof of convergence.*** Backprop uses gradient descent. Gradient descent with a nonconvex error function with multiple minima is only guaranteed to find a local, not the global of the error

function. This has long been considered a major limitation, but Yann LeCun et al. (2015) claim that it hardly matters in practice in current implementations of deep learning. **2. *Slow*.** Convergence to a local minimum can be slow due to the high dimensionality of the weight space. **3. *Poorly specified*.** Backprop neural networks had a reputation of being ill-specified, an unconstrained number of units and training examples, and a step size that varied by problem. "Neural networks came to be painted as slow and fussy to train [,] beset by voodoo parameters and simply inferior to other approaches." (Cox & Dean, 2014). **4. *Not biological*.** Lastly, backprop learning may not to be physiological: While there is ample evidence for Hebbian learning (increase of a synapse's gain after correlated activity of the two cells that it connects), such changes are never propagated farther back, beyond the one synapse, to a previous layer.

1980: Neocognitron, the first convolutional neural network. Kunihiko Fukushima (1980) proposed and implemented the Neocognitron, a hierarchical, multilayered artificial neural network. It recognized stimulus patterns (numbers) despite small changes in position and shape. It didn'

1987: NETtalk, the first impressive backprop neural network. Sejnowski et al. (1987) reported the exciting success of NETtalk, a neural network that learned to convert English text to speech: "*The performance of NETtalk has some similarities with observed human performance. (i) The learning follows a power law. (ii) The more words the network learns, the better it is at generalizing and correctly pronouncing new words. (iii) The performance of the networks degrades very slowly as connections in the network are damaged: no single link or processing unit is essential. (iv) Relearning after damage is much faster than learning during the original training. (v) Distributed or spaced practice is more effective for long-term retention than massed practice.*"

1989: ConvNets. Yann LeCun and his colleagues combined convolutional neural networks with backprop to recognize handwritten characters (LeCun et al., 1989; LeCun et al., 1990). This network was commercially deployed by AT&T, and today reads millions of checks a day (LeCun, 1998). Later, adding half-wave rectification and max pooling greatly improved its accuracy in recognizing objects (Jarrett et al., 2009).

1995: Support Vector Machine (SVM). Cortes & Vapnik (1995) proposed the support vector network, a learning machine for binary classification problems. SVMs generalize well and are free of mysterious training parameters. Many versions of the SVM are convex (e.g. Lin, 2001).

2006: Backprop, revived. Hinton & Salakhutdinov (2006) sped up backprop learning by unsupervised pre-training. This helped to revive interest in backprop. In the same year, a supervised backprop-trained convolutional neural network set a new record on the famous MNIST handwritten-digit recognition benchmark (Ranzato et al., 2006).

2012: Deep learning. Geoff Hinton says, "it took 17 years to get deep learning right; one year thinking and 16 years of progress in computing, praise be to Intel." (Cox & Dean, 2014; LeCun, Bengio, & Hinton, 2015). It is not clear who coined the term "deep learning".[2] In their book, *Deep Learning Methods and Applications*, Deng & Yu (2014) cite Hinton et al. (2006) and Bengio (2009) as the first to use the term. However, the big debut for deep learning was an influential paper by Krizhevsky et al. (2012) describing AlexNet, a deep convolutional neural network that classified 1.2 million high-resolution images into 1000 different classes, greatly outperforming previous state-of-the-art machine learning and classification algorithms.

[2] The idea of "deep learning" is not exclusive to machine learning and neural networks (e.g. Dechter, 1986)

CONTROVERSIES

Unproven convexity. A problem is convex if there are no local minima other than the global minimum. This guarantees that gradient-descent will converge to the global minimum. As far as we know, classifiers that give inconsistent results are not useful. Conservation of a solution across seeds and algorithms is evidence for convexity. For some combinations of stimuli, categories, and classifiers, convexity can be proved. For others, empirical tests can provide qualified assurance that the solution is a global minimum. Many widely used networks are not convex, but still give mostly consistent answers (LeCun, Bengio, & Hinton, 2015). In machine learning, kernel methods, including learning by SVMs, have the advantage of easy-to-prove convexity, at the cost of limited generalization. In the 1990s, SVMs were popular because they guaranteed fast convergence even with a large number of training samples (Cortes & Vapnik, 1995). Thus, when the problem is convex, the quality of solution is assured and one can rate implementations by their demands for size of network and training sample. Deep neural networks, on the other hand, generalize well, but are not convex.

Shallow vs. deep networks. The field's imagination has focused alternately on shallow and deep networks, beginning with the Perceptron in which only one layer learned, to backprop, which allowed multiple layers and cleared the hurdles that killed the Perceptron. Then SVM, with its single layer, sidelined the multilayer backprop, and today the multilayer deep learning seems to reign. Krizhevsky et al. (2012) attributed the success of their network to its 8-layer depth; it performed worse with fewer layers.

Supervised vs. unsupervised. Learning algorithms for a classifier can be supervised or not, i.e. need labels for training, or don't. Today most machine learning is *supervised* (LeCun, Bengio, & Hinton, 2015). The images are labeled (e.g. "car" or "face"), or the network receives feedback on each trial from a cost function that assesses how well its answer matches the

image's category. In *unsupervised* learning, the network processes images, typically to minimize error in reconstruction, with no extra information about what is in the (unlabeled) image. A cost function can also reward decorrelation and sparseness. This allows learning of image statistics and has been used to train early layers in deep neural networks. Human learning of categorization is sometimes done with explicitly named objects — "Look at the tree!" — but more commonly the feedback is implicit. Consider reaching your hand to raise a glass of water. Contact informs vision.

CURRENT DIRECTIONS

What does deep learning add to the vision-science toolbox? Deep learning is more than just a souped up regression (Marblestone et al., 2016). Like Signal Detection Theory (SDT), it allows us to see more in our behavioral and neural data. In the 1940's, Norbert Wiener and others developed algorithms to automate and optimize signal detection and classification. A lot of it was engineering. The whole picture changed with the SDT theorems, mainly the proof that the maximum-likelihood receiver is optimal for a wide range of simple tasks (Peterson et al., 1954). Later work added prior probability, for a Bayesian approach. Tanner & Birdsall (1958) noted that, when figuring out how a biological system does a task, it is very helpful to know the optimal algorithm and to rate observed performance by its *efficiency* relative to the optimum. SDT solved detection and classification mathematically, as maximum likelihood. It was the classification math of the sixties. Machine learning is the classification math of today. Both enable deeper insight into how biological systems classify. In the old days we used to compare human and ideal classification performance. Today, we can also compare human and machine learning.

What computer scientists can learn from psychophysics. Computer scientists build classifiers to recognize objects. Vision scientists, including psychologists and neuroscientists,

study how people and animals classify in order to understand how the brain works. So what do computer and vision scientists have to say to each other? Machine learning accepts a set of labelled stimuli to produce a classifier. Much progress has been made in physiology and psychophysics by characterizing how well biological systems can classify stimuli. The psychophysical tools (e.g. threshold and signal detection theory) developed to characterize behavioral classification performance are immediately applicable to characterize classifiers produced by machine learning (e.g. Ziskind, Hénaff, LeCun, & Pelli, 2014).

Psychophysics. "Adversarial" examples have been presented as a major flaw in deep neural networks. These slightly doctored images of objects are misclassified by a trained network, even though the doctoring has little effect on human observers. The same doctored images are similarly misclassified by several different networks trained with the same stimuli (Szegedy, et al., 2013). Humans too have adversarial examples. Illusions are robust classification errors. The blindspot-filling-in illusion is a dramatic adversarial example in human vision. While viewing with one eye, two finger tips touching in the blindspot are perceived as one long finger. If the image is shifted a bit so that the fingertips emerge from the blindspot the viewer sees two fingers. Neural networks lacking the anatomical blindspot of human vision are hardly affected by the shift. The existence of adversarial examples is intrinsic to classifiers trained with finite data, whether biological or not. In the absence of information, neural networks interpolate and so do biological brains. Psychophysics, the scientific study of perception, has achieved its greatest advances by studying classification errors. Such errors can reveal "blindspots". Stimuli that are physically different yet indistinguishable are called *metamers*. The systematic understanding of color metamers revealed the three dimensions of human color vision (Palmer, 1777; Young, 1802; Helmholtz, 1860).

CONCLUSION

Machine learning is here to stay. Deep learning is better than the “neural” networks of the eighties. Machine learning is useful both as a model for perceptual processing, and as a decoder of neural processing, to see what information the neurons are carrying. The large size of the human cortex is a distinctive feature of our species and crucial for learning. It is anatomically homogenous yet solves diverse sensory, motor, and cognitive problems. Key biological details of cortical learning remain obscure, and may preclude backprop, but the performance of current machine learning algorithms is a useful benchmark.

ACKNOWLEDGEMENTS

Thanks to Yann LeCun for helpful conversations. Thanks to Aenne Brielmann and Laura Suciu for helpful comments on the manuscript.

REFERENCES

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1), 1–127.

Bengio, Y., Lee, D. H., Bornschein, J., Mesnard, T., & Lin, Z. (2015). Towards biologically plausible deep learning. *arXiv preprint arXiv:1502.04156*.

Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, *16*(3), 199-231.

Bryson, A. E., Denham, W. F., & Dreyfus, S. E. (1963). Optimal programming problems with inequality constraints. *AIAA journal*, 1(11), 2544-2550.

Caporale, N., & Dan, Y. (2008). Spike timing-dependent plasticity: a Hebbian learning rule. *Annu. Rev. Neurosci.*, *31*, 25-46.

Cox, D. D., & Dean, T. (2014). Neural networks and neuroscience-inspired computer vision. *Current Biology*, 24(18), R921-R929.

Crick, F. (1989). The recent excitement about neural networks. Nature, 337(6203), 129.

Dechter, R. (1986). Learning while searching in constraint-satisfaction-problems. In *Proceedings of the Fifth AAAI National Conference on Artificial Intelligence* (pp. 178-183). AAAI Press.

Deng, L., & Yu, D. (2014). Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, *7*(3–4), 197-387.

Efron, B., & Hastie, T. (2016). *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science* (Vol. 5). Cambridge University Press.

Fisher, R. A. (1922). The goodness of fit of regression formulae, and the distribution of regression coefficients. *Journal of the Royal Statistical Society*, 85(4), 597-612. doi:10.2307/2341124.

Fisher, R. A. (1925). *Statistical methods for research workers*. Genesis Publishing Pvt Ltd.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of eugenics* 7.2, 179-188.

Fisher, R. A. (1954). *Statistical Methods for Research Workers* (Twelfth ed.). Edinburgh: Oliver and Boyd. ISBN 0-05-002170-2.

Friedman, J. H. (1998). Data mining and statistics: What's the connection? *Computing Science and Statistics*, *29*(1), 3-9

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, *36*(4), 193-202.

Galton, F. (1877). Typical laws of heredity. *Nature*, 15(389), 512-514.

Gauss, C.F. (1809). *Theoria motus corporum coelestium in sectionibus conicis solem ambientium.* Hamburg: Friedrich Perthes und I. H. Besser.

Geisler, W. S. (1989). Sequential ideal-observer analysis of visual discriminations. *Psychological review*, *96*(2), 267.

Helmholtz, H. von (1860/1925). *Handbuch der physiologischen Optik*, volume II. Leopold Voss, Leipzig, third edition. Translated as *Treatise on Physiological Optics*, volume II. The Optical Society of America, 1925. Edited by James P. C. Southall.

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7), 1527-1554.

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, *160*(1), 106-154.

Hung, C. P., Kreiman, G., Poggio, T., & DiCarlo, J. J. (2005). Fast readout of object identity from macaque inferior temporal cortex. *Science*, *310*(5749), 863-866.

Ivakhnenko, A. G. (1971). Polynomial theory of complex systems. *IEEE Transactions on Systems, Man and Cybernetics*, 4, 364-378.

Ivakhnenko, A. G. & Lapa, V. G. (1965). *Cybernetic Predicting Devices*. CCM Information Corporation.

Jarrett, K., Kavukcuoglu, K., & LeCun, Y. (2009). What is the best multi-stage architecture for object recognition? In *IEEE 12th International Conference on Computer Vision*. pp. 2146-2153.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems,* pp. 1097-1105.

LeCun, Y. (1985). Une procedure d'apprentissage pour reseau a seuil asymmetrique (A learning scheme for asymmetric threshold networks). In *Proceedings of Cognitiva* 85, Paris, France. pp. 599-604.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, *1*(4), 541-551.

LeCun, Y., Boser, B. E., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. E., & Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems* (pp. 396-404).

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.

Legendre, A. M. (1805). *Nouvelles méthodes pour la détermination des orbites des comètes (No. 1)*. F. Didot.

Lin, C. J. (2001). On the convergence of the decomposition method for support vector machines. *IEEE Transactions on Neural Networks*, *12*(6), 1288-1298.

Lowel, S., & Singer, W. (1992). Selection of intrinsic horizontal connections in the visual cortex by correlated neuronal activity. *Science*, *255*(5041), 209.

Marcus, G. F., Pinker, S., Ullman, M., Hollander, M., Rosen, T. J., Xu, F., & Clahsen, H. (1992). Overregularization in language acquisition. *Monographs of the society for research in child development*, i-178.

Marr, D. (1982). *VIsion: A computational investigation into the human representation and processing of visual information*. San Francisco, CA: Freeman and Company.

Matloff, N. (2014). Statistics: Losing Ground to CS, Losing Image Among Students. *Revolutions*. August 26, 2014. http://blog.revolutionanalytics.com/2014/08/statistics-losing-ground-to-cs-losing-image-among-students.html

Marblestone, A. H., Wayne, G., & Kording, K. P. (2016). Toward an integration of deep learning and neuroscience. *Frontiers in computational neuroscience*, *10*.

Mazzoni, P., Andersen, R. A., & Jordan, M. I. (1991). A more biologically plausible learning rule for neural networks. *Proceedings of the National Academy of Sciences*, *88*(10), 4433-4437.

Minsky, M., & Papert, S. (1988). *Perceptrons: An Introduction to Computational Geometry*. Cambridge, MA: MIT press.

Olshausen, B. A., & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, *381*(6583), 607.

Palmer, G. (1777). *Theory of Colour and Vision*, London: Leacroft.

Pearson, K., Yule, G. U., Blanchard, N., & Lee, A. (1903). The law of ancestral heredity. *Biometrika*, 2(2), 211-236.

Pelli, D. G., Burns, C. W., Farell, B., & Moore-Page, D. C. (2006). Feature detection and letter identification. *Vision research*, *46*(28), 4646-4674.

Peterson, W. W. T. G., Birdsall, T., & Fox, W. (1954). The theory of signal detectability. *Transactions of the IRE professional group on information theory*, 4(4), 171-212.

Ranzato, M. A., Huang, F. J., Boureau, Y. L., & LeCun, Y. (2007). Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *2007 IEEE conference on computer vision and pattern recognition*, pp. 1-8.

389 Ranzato, M. A., Poultney, C., Chopra, S., & LeCun, Y. (2007). Efficient learning of sparse
390 representations with an energy-based model. In *Proceedings of NIPS*.

391 Riesenhuber, M., & Poggio, T. (1999). Hierarchical models of object recognition in
392 cortex. *Nature neuroscience*, *2*(11), 1019-1025.

393 Rojas, R. (1996). A short proof of the posterior probability property of classifier neural networks.
394 *Neural Computation*, *8*(1), 41-43.

395 Rosenblatt, F. (1958), The Perceptron: A Probabilistic Model for Information Storage and
396 Organization in the Brain, *Psychological Review*, 65, 6, pp. 386–408.

397 Rubinov, M. (2015). Neural networks in the future of neuroscience research. *Nature Reviews*
398 *Neuroscience.*

399 Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-
400 propagating errors. *Nature*, 323, 533-536. doi:10.1038/323533a0.

401 Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM*
402 *Journal of research and development*, *3*(3), 210-229.

403 Samuel, A. L. (1967). Some studies in machine learning using the game of checkers. II—recent
404 progress. *IBM Journal of research and development*, *11*(6), 601-617.

405 Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, *61*,
406 85-117.

407 Sejnowski, T. J., & Rosenberg, C. R. (1987). Parallel networks that learn to pronounce English
408 text. *Complex systems*, 1(1), 145-168.

409 Seung, H. S., & Sompolinsky, H. (1993). Simple models for reading neuronal population codes.
410 *Proceedings of the National Academy of Sciences*, *90*(22), 10749-10753.

411 Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R.
412 (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

413 Tanner Jr, W. P., & Birdsall, T. G. (1958). Definitions of *d'* and *η* as psychophysical measures.
414 *The Journal of the Acoustical society of America*, 30(10), 922-928.

415 Turing, A.M. (1953). 'Digital computers applied to games'. in 'Faster than thought', ed. B.V.
416 Bowden, London 1953. Published by Pitman Publishing.

417 Vapnik, V. (2013). *The nature of statistical learning theory*. Springer Science & Business Media.

Weber, E. H. (1834/1996). *EH Weber on the tactile senses*. Psychology Press. Translated by Helen E. Ross from E.H Weber (1834) *De Tactu.*

Werbos, P. (1974). *Beyond regression: New tools for prediction and analysis in the behavioral sciences.* PhD thesis, Harvard University.

Young, T. (1802). The Bakerian Lecture. On the theory of light and colours, *Philosophical Transactions of the Royal Society of London* 92, 12-48. doi: 10.1098/rstl.1802.0004

Yule, G. U. (1897). On the theory of correlation. *Journal of the Royal Statistical Society*, 60(4), 812-854.

Zeiler, M. D., & Fergus, R. (2013). Visualizing and understanding convolutional networks. arXiv preprint arXiv:1311.2901.

Ziskind, A.J., Hénaff, O., LeCun, Y., & Pelli, D.G. (2014) The bottleneck in human letter recognition: A computational model. Vision Sciences Society, St. Pete Beach, Florida, May 16-21, 2014, 56.583. http://f1000.com/posters/browse/summary/1095738