NEURAL NETWORKS IN ECONOMICS: A SELECTIVE REVIEW

By

Xiaohong Chen, Luis Hoderlein and Jonas Lieber

June 2026

COWLES FOUNDATION DISCUSSION PAPER NO. 2539

COWLES FOUNDATION FOR RESEARCH IN ECONOMICS

YALE UNIVERSITY
Box 208281
New Haven, Connecticut 06520-8281

http://cowles.yale.edu/

Neural Networks in Economics: A Selective Review

Xiaohong Chen

Yale University

Luis Hoderlein

Yale University

Jonas Lieber

Imperial College London

Abstract

The "deep learning revolution" has led to remarkable success of neural networks in applications across a wide range of fields, such as computer vision, speech recognition, natural language processing, code generation, protein structure prediction, image and video generation, and dynamic control. This review introduces neural networks to economists. Recent advances and challenges in approximation theory, neural network architecture, computation, econometric theory and practice are presented. Finally, we survey the rapidly evolving applications of modern neural networks and Large Language Models in economic research.

1 Introduction

Economics and artificial neural networks share a long history of fruitful exchange, beginning with the pioneering work of Halbert White White (1987, 1988, 1989). Since around 2010, neural networks have undergone a dramatic transformation, often called the "deep learning revolution" (e.g., Sejnowski (2018)), that is now increasingly shaping empirical economic research. This review introduces neural networks to economists, surveys their applications in economics, and provides theoretical perspectives on what is and what is not yet understood about their properties as econometric tools. Throughout, we use "neural network," "neural net," and "NN" interchangeably.

1.1 What is a Neural Network?

It is often remarked that neural networks are like Lego structures: they have an *architecture* specifying how simple building blocks are combined. The basic building blocks are called *layers*.

Traditionally, a neural net is a map f from a vector of inputs x to a vector of outputs $\hat{y} := f(x)$. The map f is a composition of simpler functions, called layers. The first layer takes the raw input x, the second layer takes the first layer's output and possibly x, the third layer takes the second layer's output and possibly the first layer's output and x as inputs, and so on until the final layer outputs $\hat{y} = f(x)$. The total number of layers, denoted as L, is called the *depth* of the neural net. All layers in a neural net except the last (output) layer are called *hidden layers*. A neural net with one hidden layer is called a shallow NN (also called a two-layer NN due to the output layer). A neural net with more than one hidden layer is called a multi-layer NN or deep neural net (DNN). The dimension of a layer's output is the *width* of this layer. The width of a neural net is the maximum width of all of its layers. A wide NN has layers of large width, a narrow NN has layers of small width.

We thank the editor and an anonymous referee for very helpful comments that significantly improved the paper. We are grateful to Max Stinchcombe for sharing insights into his early engagement with neural networks in the late 1980s. All remaining errors are our own. Edited by Stéphane Bonhomme.

The basic layer type is the (fully-connected) feedforward layer. If the l-th layer of a NN is feedforward, it can be written as

$$h^l := f^l(W^l h^{l-1} + b^l),$$

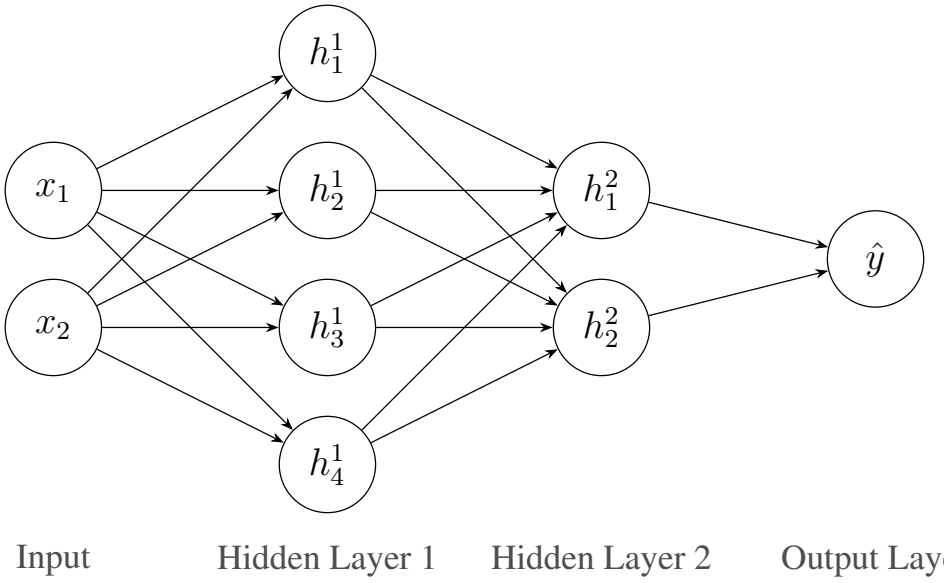
where h^{l-1} is the previous layer's output and $h^0 := x$. It is common to refer to the elements of h^l as the neurons of layer l. The dimension of h^l is the width of the l-th layer. W^l is the layer's weight matrix, and b^l is the vector of intercepts, whose elements are called *biases*.[1] f^l is the *activation function* of layer l. Traditionally, activation functions are univariate nonlinear functions that are applied element-by-element to vectors. Commonly used activation functions include the sigmoid $z \mapsto 1/(1 + \exp(-z))$, and the ReLU, defined by $z \mapsto \max\{0, z\}$. Appendix A provides a visualization and more examples of activation functions. The last layer is called the output layer and its output is denoted by y. The output layer often uses specific f's. Typical examples are the identity for regression problems and the multinomial logit, called softmax, for classification problems.

A neural net consisting only of fully-connected feedforward layers $(f^l, W^l, b^l)_{l=1,\ldots,L}$ is called a fully-connected feedforward NN. As it is the baseline specification of a neural net, it is also often referred to simply as a neural net. Other neural network architectures have qualifying names such as "convolutional neural net". In a fully-connected feedforward NN, the act of taking an input x and sequentially computing $h^1, h^2, \ldots, h^L, \hat{y}$ is called *feeding x forward*. Figure 1 illustrates this. Each circle denotes a scalar. The output layer's width is application-specific: in Figure 1 it is one, whereas a classifier over 1000 classes would have 1000 output neurons.

The expressiveness of a neural net arises from the repeated composition of nonlinear activation functions with affine transformations $W^l h^{l-1} + b^l$. The architecture of a fully-connected NN is specified by its depth, the width of each layer, and the activation function at each layer. Mathematically, even a shallow NN (a single hidden-layer NN) with a nonlinear activation function can already approximate any

[1] There is no connection to the usual notion of bias in econometrics and statistics.

Figure 1: A fully-connected neural net with 2 inputs, two hidden layers (with 4 neurons in the first and 2 neurons in the second hidden layer) and a scalar output.



continuous multivariate function on a bounded domain—the *universal approximation property*, see Hornik, Stinchcombe, and White (1989); Cybenko (1989). The practical art lies in choosing an architecture of a NN well suited to the empirical or computational task at hand. Section 3 presents additional layer types and architectures beyond the fully-connected case; also see Goodfellow, Bengio, and Courville (2016); Bishop and Bishop (2023) for textbook treatments.

1.2 Historical Notes

The current successes of neural nets arise from a long intellectual history. Building on mathematical models of the brain by McCulloch and Pitts (1943); Hebb (1949), Rosenblatt (1958) introduced the first neural network architecture, the perceptron. Ivakhnenko and Lapa (1966) trained the first feedforward neural network with multiple hidden layers using heuristic methods. Efficient gradient computation—essential for simultaneously updating all parameters via gradient descent—was enabled by backpropagation, developed independently by Linnainmaa (1970); Werbos (1974, 1982) and popularized for training neural networks in the 1980s by Rumelhart, Hinton, and Williams (1986). In the same period, John Hopfield introduced what are now called Hopfield networks, or recurrent NNs, see Hopfield (1982, 1984), modeling interactions of neurons in the brain. Geoffrey Hinton and Terry Sejnowski proposed the Boltzmann machine for

training them, see Ackley, Hinton, and Sejnowski (1985). Since Rumelhart, Hinton, and Williams (1986), backpropagation-based algorithms have been the default for training deep and wide NNs.

David Rumelhart, a UCSD faculty member from 1967 to 1987, popularized backpropagation with Geoffrey Hinton and Ronald Williams.[2] During this period, Rumelhart sparked the interest of his UCSD colleague Halbert White, who in turn drew his next-door office neighbor Max Stinchcombe, a junior game theorist at the time, into the subject. To the best of our knowledge, Hal White was the first econometrician to publish research on NNs: White (1987) establishes asymptotic results for backpropagation; White (1988) applies NNs to economic prediction in the stock market; and White (1989) is the first to apply stochastic approximation theory of Robbins and Monro (1951) to establish the consistency and asymptotic normality of the single-hidden-layer feedforward neural network nonlinear parametric least squares estimator trained via stochastic gradient descent. Together with Kurt Hornik, then a UCSD visitor, Stinchcombe and White proved one of the first universal approximation theorems for single-hidden-layer neural networks Hornik, Stinchcombe, and White (1989), followed by Hornik, Stinchcombe, and White (1990); Hornik et al. (1994) and a few more. Around the same period, Hal White also published several papers on NN approximation and consistent estimation of a smooth nonparametric conditional mean function with his friend Ron Gallant; see Gallant and White (1992).

Hal White advised many PhD students at UCSD and collaborated with several on neural networks, including Jeffrey Wooldridge (White and Wooldridge (1991)), Chien-Fu Jeff Lin (Lin (1992)), Mark Kamstra (Kamstra (1992)), Tae-Hwy Lee (Lee, White, and Granger (1993)), Mark Plutowski (Plutowski and White (1993)), Chung-ming Kuan (Kuan (1989); Kuan and White (1994)), Valentina Corradi (Corradi and White (1995)), Norm Swanson (Swanson and White (1997)), Andreas Gottschling (Gottschling (1997)), Xiaohong Chen (Chen (1993); Chen and White (1999)), and Chango Han (Han (1999)).[3] These dissertations and papers show that White's research in neural networks covered various aspects in theory

[2]Hinton was a UCSD visiting scholar in 1978–1980 and in January–June 1982; Williams was Rumelhart's PhD student. Rumelhart also advised Michael Jordan's 1985 PhD thesis at UCSD.

[3]The Mathematics Genealogy Project lists 36 PhD students of Hal White. Here we mention only those whose PhD thesis and/or published papers are on neural networks in a narrow sense. Others have published important work with Hal White on various aspects of econometrics. For more on White's influence, see Chen and Swanson (2012, 2014).

(approximation, estimation, convergence rate, asymptotic normality, and testing), algorithm/computation, and applications (to economics and finance).

To the best of our knowledge, Kuan and White (1994) is the earliest review of neural networks written for economists, covering the universal approximation property, backpropagation, the consistency and asymptotic normality of a parametric shallow neural network nonlinear least squares estimator. Other early reviews include Beltratti, Margarita, and Terna (1996), who emphasize finance and bounded rationality, and Herbrich et al. (1999), who discuss classification, time-series prediction, and preference learning.

When those reviews were written, most trainable neural networks were shallow or had only a few hidden layers. Indeed, early universal approximation results by Hornik, Stinchcombe, and White (1989); Cybenko (1989) show that a NN with a single hidden layer can approximate any multivariate continuous function arbitrarily well.[4] As documented by Hochreiter (1991), computational difficulties largely prevented the use of many hidden layers at the time. In the late 1990s, support vector machines often outperformed neural nets in practice, see Schölkopf, Burges, and Vapnik (1995), as reflected in economic reviews of that era such as Herbrich et al. (1999).

Interest in multi-layer neural networks rekindled after 2010, when advances in GPU computing by Oh and Jung (2004); Steinkraus, Buck, and Simard (2005), improvements in network architectures (see Section 3), and the availability of high-quality large-scale datasets such as ImageNet, introduced by Deng et al. (2009), enabled deep neural networks to achieve state-of-the-art performance in speech, vision, and language tasks, as demonstrated by Cireşan et al. (2010); Krizhevsky, Sutskever, and Hinton (2012). The resulting successes of overparameterized deep neural networks have been characterized as a "deep learning revolution," for which Geoffrey Hinton, Yoshua Bengio, and Yann LeCun were awarded the 2018 Turing Award. In 2024, Geoffrey Hinton and John Hopfield received the Nobel Prize in Physics "for foundational discoveries and inventions that enable machine learning with artificial neural networks." Of course, numerous other scholars have contributed to these advances; see Schmidhuber (2015); Sejnowski (2018).

[4] See section 2 for discussions on the theoretical advantages of considering deep instead of wide neural networks.

The deep learning revolution has led to the recent success of large language models. Many economists adopted these tools in the late 2010s, leading to an explosion of applications and reviews as illustrated in Figure 1 of Zheng, Xu, and Xiao (2023).

1.3 Summary of this Review

This review focuses on neural net architectures and economic applications that have emerged since the deep learning revolution around 2010. We begin with theoretical perspectives in Section 2, emphasizing what neural nets add relative to other flexible semiparametric and nonparametric tools. Section 3 then offers a more concrete guide to modern architectures and large language models. Section 4 reviews some recent economic applications. Section 5 briefly concludes.

In our view, the success of modern NNs reflects four forces working together: (i) approximation and representation results showing that rich network classes can express complex high-dimensional relationships; (ii) computational and architectural advances, including GPUs, automatic differentiation, transformers, reinforcement learning and mixtures-of-experts; (iii) large and high-quality datasets; and (iv) the human and computational resources at industrial scale needed to effectively combine (ii) and (iii). We do not think universal approximation results alone explain the current empirical successes: Universal approximation properties only establish that neural nets are expressive enough in principle. The recent practical breakthroughs come from turning that expressive power into algorithms, predictors and classifiers that can be trained using large-scale high-quality data sets, and that can sometimes generalize well.

2 Theoretical Perspectives

Two broad ideas motivate interest in neural networks. The first is *representation*: every multivariate function can be expressed as compositions and sums of simpler, often univariate, functions, though the representation is not unique. The second is *approximation*: even when an exact representation is too difficult to recover from data, neural nets can approximate rich classes of functions to arbitrary precision. Different neural net architectures may have different approximation error rates for different classes of target functions. For economics, the approximation perspective is the more useful one, because it justifies viewing neural nets as

flexible *nonlinear sieves* and focuses attention on approximation rates, complexity control, and the inferential objects that matter for empirical work.

2.1 Kolmogorov-Arnold Representation

Can every multivariate function be decomposed into univariate building blocks? The Kolmogorov–Arnold representation theorem gives a striking affirmative answer.

Theorem 2.1. *Let $d \geq 2$ and $\gamma \geq 2d+2$ be given integers. Set $a = [\gamma(\gamma-1)]^{-1}$, $\lambda_1 = 1$, $\lambda_p = \sum_{j=1}^{\infty} \gamma^{-(p-1)(d^j-1)/(d-1)}$ for $p = 2, 3, \ldots, d$ and $b_q = (2d+1)q$ for $q = 0, 1, \ldots, 2d$. Then there exists a universal, monotonically increasing function $\phi : \mathbb{R} \to \mathbb{R}$ that is Lipschitz-continuous with Lipschitz constant at most $\ln(2)/\ln(\gamma)$ such that, for every d-variable function $f : [0,1]^d \to \mathbb{R}$, there exists a one-variable function g_f satisfying*

$$f(x_1, \ldots, x_d) = \sum_{q=0}^{2d} g_f \left(\sum_{p=1}^{d} \lambda_p \phi(x_p + aq) + b_q \right),$$

where g_f can be chosen to be continuous, discontinuous bounded or unbounded respectively if f is continuous, discontinuous bounded or unbounded.

This result is Theorem 2.1 from Ismayilova and Ismailov (2024), which gives a Kolmogorov *two-hidden-layer* feedforward NN representation of any multivariate function $f : [0,1]^d \to \mathbb{R}$. In particular, univariate nonlinearities are applied to affinely transformed coordinates, then aggregated and passed through another set of univariate functions.

Although the Kolmogorov-Arnold representation theorem motivates the expressiveness of multi-layer NNs, it is very difficult to recover the exact nonlinear activation functions g_f and ϕ from data for an arbitrary unknown multivariate function f, since the second-hidden-layer activation function g_f depends on f in an unknown way. Nevertheless, this representation theorem does motivate various deep NN architectures. See, e.g., Schmidt-Hieber (2021) for deep NN with ReLU activations to approximate Hölder continuous functions, and Liu et al. (2025) and Wang et al. (2025) for multi-layer spline activation NN.

2.2 Universal Approximators and Approximation Error Rate

A classical result is that even shallow NNs can be universal approximators. One convenient version is the following theorem, which summarizes results of Cybenko (1989); Hornik, Stinchcombe, and White (1989, 1990); Leshno et al. (1993).

Theorem 2.2 (Universal Approximation, informal)**.** *Let $\Omega \subset \mathbb{R}^d$ be connected and compact and let $\sigma : \mathbb{R} \to \mathbb{R}$ be a bounded continuous activation function that is not a polynomial. Let*

$$\Sigma_m^\sigma(\Omega) := \left\{ g : \Omega \to \mathbb{R} : g(x) = \sum_{j=1}^{m} a_j \sigma(w_j' x + b_j) : a_j, b_j \in \mathbb{R},\ w_j \in \mathbb{R}^d \right\}$$

denote the class of one-hidden-layer NNs with activation function σ and width m. Then, for every continuous function $f : \Omega \to \mathbb{R}$ and every $\varepsilon > 0$, there exists a $g \in \Sigma_m^\sigma(\Omega)$ such that

$$\|f - g\|_\infty := \sup_{x \in \Omega} |f(x) - g(x)| \leq \varepsilon \,.$$

The universal approximation theorem is an *existence* statement. It shows that NNs with a single hidden layer can approximate any continuous function on a compact set arbitrarily well. This result suffices to establish consistency of a shallow NN sieve for estimation of the conditional mean in nonparametric least squares learning; see, e.g., White and Wooldridge (1991); Gallant and White (1992). However, it does not tell us how accurate any given approximation is. For numerical computation as well as for econometric estimation and inference, we need to know how fast the approximation error $\inf_{g \in \Sigma_m^\sigma(\Omega)} \|f - g\|_\infty$ vanishes as the shallow NN sieve width m grows.

2.2.1 Approximation Error Rate of Shallow NNs for Generalized Barron Class Barron (1993) established a well-known approximation error rate of $m^{-1/2}$ on how fast shallow NNs $\Sigma_m^\sigma(\Omega)$ can approximate a smooth function f belonging to the space of square-integrable functions whose Fourier transform has finite first moment, which is now called the Barron space. For a nonparametric regression function $\mathbb{E}[Y|X] = f(X)$ approximated using a shallow NN sieve with width m_n, the approximation rate $(m_n)^{-1/2}$ implies an L_2-convergence rate *slower* than $n^{-1/4}$, where n is the sample size. Barron (1993)'s approximation rate was subsequently improved to $m^{-1/2-1/(2d)}$ in Makovoz (1996) for the Heaviside activation

$\sigma(u) = 1\{u \geq 0\}$, and to $m^{-1/2-1/(d+1)}$ in Chen and White (1999) for general sigmoid activation functions. These slightly improved approximation error rates are enough to yield a *faster* than $n^{-1/4}$ convergence rate in L_2-norm for shallow NN sieve estimators of conditional means, conditional quantiles, and conditional densities; see Chen and Shen (1998); Chen and White (1999). Recently, DeVore et al. (2025) establish a sharp approximation error rate for shallow ReLU networks. The following is an informal version of their Theorem 6.1.

Theorem 2.3 (DeVore et al. (2025), informal)**.** *Let $\Omega \subset \mathbb{R}^d$ be bounded and let $\Sigma_m^{\mathrm{ReLU}}(\Omega)$ denote the class of one-hidden-layer ReLU networks on Ω with width m. If f belongs to the weighted variation space $V_w(\Omega)$ introduced by DeVore et al. (2025), then*

$$\inf_{g \in \Sigma_m^{\mathrm{ReLU}}(\Omega)} \|f - g\|_{L_2(\Omega)} \leq C \|f\|_{V_w(\Omega)} m^{-1/2-3/(2d)},$$

where C depends only on the dimension d.

As pointed out in DeVore et al. (2025), the weighted variation space $V_w(\Omega)$ contains the original Barron class as a strict subset. The importance of DeVore et al. (2025) is not only the best approximation error rate but the observation that by tailoring the weighted variation space $V_w(\Omega)$ to the geometry of the bounded domain Ω, single-hidden layer ReLU NN sieves can achieve faster approximation error rate for any target function in $V_w(\Omega)$. This approximation error rate can be used to derive a final convergence rate that is faster than $n^{-1/4}$ in estimating a nonparametric function in $V_w(\Omega)$ using ReLU shallow NNs.

2.2.2 Approximation Error Rate of Deep NNs for the Sobolev Class of Functions A sharp approximation-theoretic result for fully-connected deep ReLU sieves is given by Yang (2025), which extends earlier results by Siegel (2023); Lu et al. (2021) and others.

Theorem 2.4 (Yang (2025), informal)**.** *Let $\Omega = [0,1]^d$, let $s/d > 0$, and let $1 \leq p \leq \infty$, and $\mathcal{W}_p^s(\Omega)$ be the Sobolev space. Let $\mathcal{F}_{m,L}^{\mathrm{ReLU}}(\Omega)$ denote the class of scalar-valued fully-connected ReLU networks on Ω with width m and depth L. Then for every $f \in \mathcal{W}_p^s(\Omega)$,*

$$\inf_{g \in \mathcal{F}_{m,L}^{\mathrm{ReLU}}(\Omega)} \|f - g\|_{L_p(\Omega)} \leq C \|f\|_{\mathcal{W}_p^s(\Omega)} (mL)^{-2s/d},$$

where C depends only on (s, p, d).

This upper bound is sharp for fully-connected feedforward NNs' approximation to $\mathcal{W}_p^s(\Omega)$; see Zhou (2020) for an approximation error rate for deep convolutional NNs. Using this approximation error rate, Yang (2025) obtains the almost optimal nonparametric least squares (LS) estimation error rate. Without imposing extra structure on the unknown target function f in the Sobolev space, the approximation error rate by fully-connected NNs and the nonparametric LS NN estimation still suffers the "curse of dimensionality d".

2.2.3 Sparse DNN for Sparse Compositional Smooth Class of Functions Schmidt-Hieber (2020) studies the approximation error rate of a class of sparse deep ReLU networks to a target smooth function with a special compositional structure.

Theorem 2.5 (Schmidt-Hieber (2020), informal)**.** *Suppose*

$$f_0 = g_q \circ g_{q-1} \circ \cdots \circ g_0,$$

where each component map g_i depends on at most t_i variables and has Hölder smoothness index β_i. Define the effective smoothness indices $\beta_i^* := \beta_i \prod_{\ell=i+1}^{q} (\beta_\ell \wedge 1)$. Then:

$$\inf_{g \in \mathcal{F}_{m,L,S}^{\mathrm{ReLU}}} \|g - f_0\|_\infty \leq C \max_{i=0,\ldots,q} m^{-\beta_i^*/t_i},$$

for suitable deep ReLU architectures $\mathcal{F}_{m,L,S}^{\mathrm{ReLU}}$ with hidden widths of order m, depth $L \lesssim \log m$, and sparsity $S \lesssim m \log m$.

When the target function f_0 has a low-dimensional compositional architecture, for example additive structure, generalized additive structure as in Horowitz and Mammen (2007), or single-index type structure, the relevant dimensions are the intrinsic t_i rather than the ambient dimension d, which yields a faster approximation rate and faster estimation rates; see, e.g., Schmidt-Hieber (2020); Kohler and Langer (2021). In Section 2.3, we review how economic shape constraints can be imposed on neural networks.

2.3 Shape Restrictions

Shape restrictions such as monotonicity, concavity, convexity, and separability play an important role in identification and estimation of economic models, see, e.g., Matzkin (1994); Chetverikov, Santos, and Shaikh (2018). Imposing such shape restrictions can be viewed as yielding *economics-informed neural networks*, by analogy with physics-informed neural networks; see, e.g., Raissi, Perdikaris, and Karniadakis (2019). These restrictions can be encoded directly in the NN architectures, or alternatively be imposed through constrained estimation or a penalty on the criterion function.

First, we review the early literature which considered neural networks with a single hidden layer. Joerding and Meador (1991) enforce concavity and monotonicity of a neural network with a single hidden layer via constraints on the weights. Joerding and Li (1994) propose enforcing three types of constraints on neural networks with a single hidden layer in the context of a simulated annealing optimization routine, a general optimization routine which is guaranteed to converge eventually but typically takes prohibitively long in practice. First, they propose to enforce box constraints for the output of the neural network provided that the activation function of the last layer is bounded itself, e.g. the sigmoid. Second, they enforce linear inequality constraints on the parameters to avoid the weight space symmetry discussed above. Third, they enforce linear constraints to avoid a similar sign symmetry property, for example for the sigmoid $\sigma(z) = 1-\sigma(-z)$ which also leads to non-identification of the parameters.

When a differentiable neural net is used to approximate a scalar-valued function, then the Hessian of the neural net is symmetric by Schwarz's Theorem/Clairaut's Theorem. However, sometimes, a neural network is used to approximate a gradient. For example, when trying to estimate a production function $F(x_1, \ldots, x_n)$, one may observe input prices $p_1, \ldots, p_n$ and output price p as well as the inputs $x_1, \ldots, x_n$ but not output $F(x_1, \ldots x_n)$. However, profit maximization implies

$$\frac{\partial F(x_1, \ldots, x_n)}{\partial x_j} = \frac{p_j}{p}.$$

One can thus train a neural network to predict the outcome vector $(\frac{p_1}{p}, \ldots, \frac{p_n}{p})$ given the features $x_1, \ldots, x_n$. This identifies the gradient of the production function. The Jacobian of this gradient is the

Hessian of the production function and should thus be symmetric. But this symmetry is not directly imposed when estimating the gradient and could be violated in finite datasets. Cardell, Joerding, and Li (1995) propose to enforce this symmetry directly when estimating the production function and derive the parameter constraints sufficient to enforce this for the case of single hidden-layer neural nets using quadratic constraints.

Recent papers have considered shape constraints for neural networks with multiple hidden layers. Amos, Xu, and Kolter (2017) propose an architecture for which convexity can be imposed directly: if the hidden-to-hidden weight matrices are nonnegative and the activation functions are convex and nondecreasing, the network output is convex in the constrained inputs. Han et al. (2022) establish a universal approximation result of neural networks for symmetric and anti-symmetric functions. The symmetric function approximation is used in Han, Yang, and E (2026) to construct generalized moments inside a DNN global solution algorithm for a class of high-dimensional heterogeneous agent models with aggregate shocks. There are also results on imposing monotonicity in NNs; see, e.g., Kim and Lee (2024); Azinovic-Yang and Žemlička (2025) for recent references.

Chen and Ludvigson (2009) estimate a semiparametric consumption-based asset-pricing model with unknown habit formation, with the period t habit level denoted as $H_t := H(C_t, C_{t-1}, \ldots, C_{t-L})$, which is a homogeneous-of-degree-one unknown function of current and lagged quarterly aggregate non-durable consumption $(C_t, C_{t-1}, \ldots, C_{t-L})$. For identification and asset pricing theory they impose that $\frac{H_t}{C_t} := h\left(\frac{C_{t-1}}{C_t}, \ldots, \frac{C_{t-L}}{C_t}\right) \in (0, 1)$, which is approximated via a single-hidden-layer sigmoid network. Because the logistic activation takes values in $(0, 1)$, they can constrain the network parameters so that the implied habit stock always remains below current consumption—the architecture itself makes the constraint tractable. Menzel (2021) derives network structures from economic primitives: in his production and nested-discrete-choice examples, hidden states correspond to latent intermediate decisions and the activation functions are tied to the model's optimality conditions. The resulting architecture is both a universal approximator and a vehicle for encoding economically motivated sign and curvature restrictions. Finally, the Kolmogorov-Arnold representation also motivates some generalized additive models and other shape-restricted semi-nonparametric models such as Coppejans (2004); Horowitz and Mammen (2007).

Taken together, these examples indicate that neural nets are flexible approximations to any high-dimensional economic object of interest and can preserve/encode shape restrictions implied by economic theory. Furthermore, if the shape restrictions are correctly specified, the shape-preserving NNs can be trained more stably than unrestricted NNs and can generalize better out of sample.

2.4 NN Sieve Extremum Estimation and Inference

Most economic questions are ultimately about *economic objects*—demand functions, production functions, conditional choice probabilities, pricing kernels, continuation values, treatment effects—not about the weight parameters of a particular neural network. From an econometric standpoint, it is therefore more useful to treat a neural net as a flexible approximating class, that is, a *nonlinear sieve*, for an identified economic primitive.

Neural nets are especially attractive as nonlinear sieves when the target object is a conditional mean, conditional quantile, conditional density, or conditional choice probability depending on a moderately high-dimensional vector of covariates/features. In those settings, the function of interest may be nonparametrically identified even though the network parameters themselves are not point-identified.[5] Once neural nets are interpreted this way, the general sieve toolkit (see Chen (2007)) for consistency, convergence rates, semiparametric efficiency, and testing becomes directly relevant. Indeed, in his Nobel lecture entitled "Economic Choices", McFadden (2001) writes on page 360

> "*It is possible to use latent class models to obtain nonparametric estimates of any family of RUM [Random Utility Model]-consistent choice probabilities by the method of sieves. The latent class model is a single hidden-layer feedforward neural network (with MNL [Multinomial Logit] activation functions), and the asymptotic approximation theory that has been developed for neural networks can be applied to establish convergence rates and stopping rules (Hal White, 1989, 1992; Bing Cheng and D.Michael Titterington, 1994; Xiaohong Chen and White,*

[5]For example, a one-hidden-layer ReLU network can often be re-parameterized by permuting hidden units without changing the input-output map. This "weight-space symmetry" is one reason the network weights/parameters are rarely the economically meaningful objects.

1998; Chunrong Ai and Chen, 1999)."

2.4.1 Neural Network Semi/nonparametric M-Estimation and Inference Many econometric problems share a common semi/nonparametric structure: a finite-dimensional parameter of interest $\theta \in \Theta \subset \mathbb{R}^{d_\theta}$ appears alongside an unknown infinite-dimensional function $h \in \mathcal{H}$. Write $\alpha = (\theta, h) \in \mathcal{A} := \Theta \times \mathcal{H}$. Let $\alpha_0 = (\theta_0, h_0) \in \mathcal{A}$ denote the unique global minimizer of $\inf_{\alpha \in \mathcal{A}} \mathbb{E}[\ell(W_i; \alpha)]$, where ℓ is a general loss function and $\{W_i\}_{i=1}^n$ is an i.i.d. (or weakly dependent) sample. With cross-entropy loss, this covers classification problems in which conditional class probabilities are parameterized by a neural-net sieve. With negative log-likelihood (or quasi-likelihood), generalized least squares, quantile regression and Huber's M losses, it covers various semi/nonparametric problems in which the neural net enters through an unknown index, density, or choice-probability component while the rest of the model remains parametric.

We approximate $h \in \mathcal{H}$ by a NN sieve $\mathcal{H}_n^{NN} \subset \mathcal{H}$, and set $\mathcal{A}_n := \Theta \times \mathcal{H}_n^{NN}$. Let $\hat{\alpha}_n \in \mathcal{A}_n$ be an approximate sieve M-estimator solving

$$\frac{1}{n}\sum_{i=1}^{n} \ell(W_i; \hat{\alpha}_n) \leq \inf_{\alpha \in \mathcal{A}_n} \frac{1}{n}\sum_{i=1}^{n} \ell(W_i; \alpha) + O_P(\varepsilon_n^2), \quad \text{with} \quad \varepsilon_n \to 0. \tag{1}$$

The large sample theoretical results, including convergence rate and asymptotic normality of smooth functionals, for general sieve M-estimation and inference are directly applicable to the approximate NN sieve M problem (1); see, e.g., Shen and Wong (1994); Shen (1997) for i.i.d. data and Chen and Shen (1998) for weakly dependent data. We present one simple version of convergence rate below.

Let $\|\alpha - \alpha_0\|$ be a pseudo-metric on $\mathcal{A} = \Theta \times \mathcal{H}$ such that $\|\alpha - \alpha_0\|^2 \asymp \mathbb{E}[\ell(W_i; \alpha) - \ell(W_i; \alpha_0)]$ locally around $\alpha_0 \in \mathcal{A}$. For a negative log-likelihood criterion, this pseudo-metric becomes the Kullback-Leibler (KL) divergence; for a nonparametric regression loss $\mathbb{E}[(Y_i - h(X_i))^2/2]$ with $h_0(X) = \mathbb{E}[Y|X]$, this metric becomes $\|h - h_0\|_2 := \sqrt{\mathbb{E}[(h(X) - h_0(X))^2]}$ (the standard $L_2(X)$-norm). As stressed in Chen (2007), the general sieve estimation problem, even including deconvolution, semi/nonparametric mixture, nonparametric endogeneity, can remain "well-posed" under this criterion-induced pseudo-metric. Define

$$\mathcal{L}_n = \{\ell(\cdot\,; \alpha) - \ell(\cdot\,; \alpha_0) : \|\alpha - \alpha_0\| \leq \delta,\ \alpha \in \mathcal{A}_n\},$$

and for some constants $b, C > 0$,

$$\delta_n = \inf\left\{ \delta \in (0,1) : \frac{1}{\sqrt{n}\,\delta^2} \int_{b\delta^2}^{\delta} \sqrt{\log N_{[]}(w, \mathcal{L}_n, \|\cdot\|_2)}\, dw \leq C \right\},$$

where $N_{[]}(\cdot)$ denotes the radius w covering number of $\mathcal{L}_n$ with bracketing.

Theorem 2.6 (Chen (2007), informal)**.** *Let $\hat{\alpha}_n$ be the approximate sieve M-estimator defined by (1). Let $\pi_n(\alpha_0)$ denote the $\|\cdot\|$- projection of α_0 onto the sieve $\mathcal{A}_n$. Then*

$$\|\hat{\alpha}_n - \alpha_0\| = O_P(\varepsilon_n), \qquad \varepsilon_n = \max\{\delta_n, \|\alpha_0 - \pi_n(\alpha_0)\|\}.$$

Previously, Chen and White (1999) derived faster than $n^{-1/4}$ convergence rate for single-hidden layer NN sieve M-estimation of conditional mean, conditional quantile, conditional density, joint density as well as root-n asymptotic normality of θ_0 for stationary time series data. Chen, Racine, and Swanson (2001) established faster than $n^{-1/4}$ convergence rates for estimating semiparametric nonlinear autoregressive conditional mean and median models with exogenous variables (NLARX) via three classes of NN sieves: NNs with sigmoid activation function, NNs with radial basis activation, and NNs with ridgelet activation. As an empirical application, they compare the forecasting performance of linear and semiparametric NLARX models of U.S. inflation, and find that all semiparametric NLARX NN sieve estimated models outperform a benchmark linear ARX model based on various forecast performance measures, where all models include various lags of historical inflation and the GDP gap. Interestingly, their empirical findings support the current view of a nonlinear Phillips curve. Chen, Hong, and Shum (2007) developed a nonparametric likelihood ratio testing procedure for choosing between a parametric likelihood model and a moment-condition model when both models could be misspecified. A key input to their KLIC comparison is a shallow NN sieve estimator of the population entropy $\theta_0 = \mathbb{E}_0[\log h_0(Z)] := \int [\log h_0(z)] h_0(z) dz$. Applying the improved approximation error rate of Chen and White (1999) to estimate log-density via NN sieve maximum likelihood, they established $\mathbb{E}_0[\log \hat{h}_n - \log(h_0)] = o_p(n^{-1/2})$ and

$$\sqrt{n}\left\{ \frac{1}{n}\sum_{i=1}^{n} \log \hat{h}(Z_i) - \mathbb{E}_0[\log h_0(Z)] \right\} \rightsquigarrow \mathcal{N}(0, \mathrm{Var}_0[\log h_0(Z)])\,.$$

Consequently, in their KLIC-based model comparison test statistics, the shallow NN sieve entropy estima-

tor $\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} \log \hat{h}(Z_i)$ has the same root-n first-order behavior as if the true density $h_0(\cdot)$ were observed.

The latest improved approximation error rates $\|\pi_n(\alpha_0) - \alpha_0\|$ for different NN sieves (see, e.g., Section 2.2) will give improved convergence rates $\|\hat{\alpha}_n - \alpha_0\|$, and can be used in diverse applications with high-dimensional covariates, such as treatment effects, missing data, measurement errors, semi-nonparametric mixture, and models with latent heterogeneity. For example, Farrell, Liang, and Misra (2021b) use multi-layer NNs to approximate the propensity score function of high-dimensional covariates, and Chen et al. (2024) use shallow NNs to approximate the generalized propensity score function $P_d(x) = \mathbb{P}(D = d \mid X = x)$ for a multi-valued treatment $D \in \{0, 1, \ldots, J\}$ and growing dimensional covariates X; see Section 4.4.4 for details.

2.4.2 NN Sieves for Semi-Nonparametric Conditional Moment Restrictions Many structural economic models—Euler equations in consumption-based asset pricing, demand and supply systems with unknown nonlinearities, nonparametric IV, dynamic structural models with unknown value functions—are stated directly in terms of a conditional moment restriction derived from the economic optimality or equilibrium conditions:

$$\mathbb{E}[\rho(Y, \alpha_0) \mid X] = 0, \qquad \alpha_0 = (\theta_0, h_0),$$

where $\theta_0 \in \Theta \subset \mathbb{R}^{d_\theta}$ is a finite-dimensional structural parameter and $h_0 \in \mathcal{H}$ is an unknown structural function (habit formation, utility, production, or link function). This conditional moment restriction model includes the well-studied nonparametric instrumental variables model (NPIV), $\mathbb{E}[Y_1 - h_0(Y_2) \mid X] = 0$, as a special case. Newey and Powell (2003) provide the nonparametric identification of h_0 and a consistent estimator for h_0. Ai and Chen (2003); Chen and Pouzo (2012, 2015) obtain the convergence rate, asymptotic normality and inference based on a (penalized) sieve minimum distance criterion. When $h_0(\cdot)$ is a function of high-dimensional covariates, it can be approximated by a NN sieve $\mathcal{H}_n^{NN} \subset \mathcal{H}$. Following Ai and Chen (2003), one can first estimate the conditional moment function $m(X, \alpha) := \mathbb{E}[\rho(Y, \alpha) \mid X]$ by projecting $\rho(Y, \alpha)$ onto a flexible instrument space, and then estimate $\alpha_0 = (\theta_0, h_0)$ by solving the following NN sieve minimum distance:

$$\inf_{\alpha=(\theta,h)\in\Theta\times\mathcal{H}_n^{NN}} \left[\frac{1}{n}\sum_{i=1}^{n} \hat{m}(X_i, \alpha)' \, \hat{\Sigma}(X_i)^{-1} \, \hat{m}(X_i, \alpha) + \lambda_n \mathrm{Pen}(h)\right]. \tag{2}$$

An early real data application is Chen and Ludvigson (2009), who estimate a semiparametric habit-based asset-pricing model by NN sieve minimum distance, where the unknown habit function is approximated with a single-hidden-layer sigmoid NN. Their NN internal habit stochastic discount factor proxy explains a cross-section of size and book-market sorted portfolio equity returns better than five popular asset-pricing models and has the smallest pricing errors.

Chen, Liao, and Wang (2025) extend the sequential moment restriction of Ai and Chen (2012) to weakly dependent time-series conditional moment restrictions. Their general model is

$$\mathbb{E}\Big[\rho(Y_{t+1}, \alpha_0) \mid \mathcal{F}_t^{(r)}\Big] = 0, \qquad \phi(h_0) = \mathbb{E}[b(h_0(U_t))],$$

where $\alpha_0 = (\theta_0, h_0)$, $b(\cdot)$ is a known smooth function, and $\mathcal{F}_t^{(r)}$ is a finite-order information set. They approximate the unknown h_0 by general nonlinear sieves, including multi-layer NNs as leading examples. They derive semiparametrically efficient inference for smooth functionals of the structural function by constructing an orthogonalized quasi-likelihood ratio statistic. A practical advantage of their approach is that test inversion avoids direct estimation of Riesz representers and remains valid even for irregular functionals, i.e., when $\phi(h_0)$ is not root-n estimable, where n denotes the size of the time series data set. Applications include dynamic NPIV, i.e., $\mathbb{E}[Z_{t+1} - h_0(Y_{t+1}) \mid \mathcal{F}_t^{(r)}] = 0$, nonparametric quantile IV, and off-policy evaluation in reinforcement learning.

Recent work in machine learning applies neural networks to NPIV, i.e., $\mathbb{E}[Y_1 - h_0(Y_2) \mid X] = 0$, as well. Dikkala et al. (2020) propose an adversarial (minimax) approach: starting from the unconditional moment equality $\mathbb{E}[(Y_1 - h_0(Y_2))f(X)] = 0$, they minimize this criterion over a candidate h in an outer loop while maximizing over a critic f in an inner loop, with a regularization penalty on the second moment of f. In parallel, Bennett and Kallus (2023) applies NN sieves to approximate optimal instruments in the efficient minimax estimation of the parametric conditional moment restriction models $\mathbb{E}[\rho(Y, \theta_0) \mid X] = 0$, which has been studied in econometrics since Hansen (1982); Chamberlain (1987); Newey (1990).

Chen, Chen, and Tamer (2023) provide a detailed Monte Carlo comparison of linear spline sieves

and various multi-layer NN sieves in the NPIV setting $\mathbb{E}[Y_1 - h_0(Y_2) \mid X] = 0$, focusing on semi-parametrically efficient estimation of the average-derivative functional $\theta_0 = \mathbb{E}[a(Y_2)\nabla h_0(Y_2)]$ (e.g., an average treatment effect) of the NPIV function h_0. Their simulations suggest that the optimally weighted orthogonalized sieve minimum-distance estimator of Ai and Chen (2012) is typically more stable than efficient-score (efficient-influence-function) implementations when neural-network sieves are used. The plug-in estimator of θ_0 based on the Deep NPIV estimator Dikkala et al. (2020) of h_0 also performs poorly in their Monte Carlo designs, due to sensitivity to the choice of tuning parameters.

2.5 Computation & Stochastic Approximation

Econometric and statistical theory typically analyzes estimators as if a global minimizer of the empirical (or training) criterion were available. In practice, however, neural-net estimation is usually implemented by some versions of a stochastic approximation (SA) procedure of Robbins and Monro (1951). Let ϑ denote all trainable parameters associated with a population loss criterion $\mathbb{E}[\ell(W, \vartheta)]$, the following stochastic gradient descent (SGD) algorithm is a special case of a SA recursion:

$$\vartheta_k = \vartheta_{k-1} - \gamma_k \nabla_\vartheta \ell(W_k, \vartheta_{k-1}),$$

where $\{\gamma_k\}$ is a decreasing step-size sequence. White (1989) was the first to study the almost sure convergence and asymptotic normality of this algorithm in a parametric single-hidden-layer NN regression model. Since then, the original SA estimator has been modified and improved in many ways, including using Polyak-Ruppert averages $\bar{\vartheta}_N = \frac{1}{N}\sum_{k=1}^{N} \vartheta_k$ to stabilize the trajectory and recover optimal convergence rates.

Studying the global convergence properties of various SA algorithms is particularly important for neural nets because the empirical criterion is non-convex, gradients may vanish or explode in deeper architectures, and global optimization is generally hard, as documented by Hochreiter (1991); Murty and Kabadi (1987); Bartlett and Ben-David (2002); He et al. (2016); Froese and Hertrich (2023). Since NN sieves are highly nonlinear, there is no guarantee that computational algorithms actually converge to a unique global minimizer. Unfortunately, theoretical analyses in econometrics and statistics as well as the

empirical work building on them often leave aside the fact that default DNN algorithms typically converge only to local solutions. In this subsection we present two recent contributions showing that SA algorithms for single-hidden-layer NN sieves can converge to global solutions.

Stochastic Approximation for Shallow NN Sieve M-Estimation Chen, Tamer, and Yao (2026) study online learning of a semiparametric monotone single-index model

$$Y = F_0(x_0 + X'\theta_0) + \varepsilon, \qquad \mathbb{E}[\varepsilon|x_0, X] = 0,$$

using a nonlinear Least Squares loss criterion $\mathbb{E}\{[(Y - F(x_0 + X'\theta)]^2\}$. To reinterpret their procedure as a neural-net estimator, approximate the unknown link function $F()$ by a single-hidden-layer sieve

$$F_J(z) = \beta_0 + \sum_{j=1}^{J} \beta_j \sigma(a_j z + b_j),$$

where σ is, for example, a sigmoid or $\text{ReLU}^k := [\max\{y, 0\}]^k$ (for $k \geq 1$) activation, and (a_j, b_j) are basis parameters. Then $F_J(x_0 + X'\theta)$ is a single-hidden-layer neural network in the original covariates (x_0, X), whose hidden-unit weights satisfy the index restriction $w_j = a_j(1, \theta')'$. Their online single-index sieve estimator can be viewed as an online estimator for a constrained shallow neural net. The estimation proceeds in two phases. Phase I updates the index parameter alone via a SA recursion based on a smoothed maximum-rank-correlation score. This phase avoids estimating the unknown link function F_0 and is designed to move into a neighborhood of the unknown true θ_0 from essentially arbitrary initialization. That is, Phase I generates a warm start for Phase II.[6] After this warm start, parametric and sieve components are updated jointly via mini-batch online sieve nonlinear least squares in Phase II, which automatically yields an orthogonalized score for the index parameter along the online sieve algorithm. Polyak-Ruppert averages are then formed for both θ_0 and the sieve coefficients. The theoretical analysis shows that, with suitable growth of sieve dimension and suitable step sizes, Phase II achieves rate-optimal learning for both the single-index parameter θ_0 and the sieve online estimation of the unknown link function F_0, as well as marginal effect

[6] Actually, any fast global optimizing algorithm could serve as a warm start. Indeed, Chen, Tamer, and Yao (2026) also implemented a new global optimizer of Chen et al. (2026) for general high-dimensional possibly multi-modal objective functions, and the online Phase II algorithm still performs well and its properties remain valid.

parameters. In addition, the paper provides simple online inference for marginal effects and other functionals of interest. In this sense, the paper provides a concrete bridge between classical semiparametric online learning and online training of a constrained one-hidden-layer neural network.

Adversarial Estimation of Nonparametric Conditional Moment Restrictions with Over-Parameterized Shallow Networks Zhu et al. (2024) recast nonparametric conditional moment problems $\mathbb{E}[\rho(Y; h_0(\cdot))|X] = 0$ as the following functional minimax programs:

$$\min_{h \in \Sigma_N^{\sigma_1}(\Omega_Y)} \max_{f \in \Sigma_N^{\sigma_2}(\Omega_X)} L(h, f), \qquad L(h, f) = \mathbb{E}\left[f(X)\rho(Y; h) - \frac{1}{2} f(X)^2 + \lambda \Psi(Y; h) \right],$$

where h is the structural function of interest, f is an adversarial critic, $\rho(Y; h)$ encodes the conditional moment restriction, and Ψ is a penalization/regularization term. For NPIV, one can take $\rho(Y; h) = Y_1 - h(Y_2)$. Both the primal function and the dual critic are represented by over-parameterized single-hidden layer neural nets with width N and trained by stochastic gradient descent-ascent. In the mean-field limit (the width $N \to \infty$), the training dynamics become a Wasserstein gradient flow that converges to a stationary point, and under slightly stronger convexity, the algorithm is shown to reach the solution of the conditional moment restriction equation $\mathbb{E}[\rho(Y, h_0)|X] = 0$. This provides a bridge between classical sieve minimum-distance thinking (with $\Sigma_N^{\sigma_2}(\Omega_X)$ being a linear sieve) and modern adversarial estimation for NPIV, policy evaluation, and asset pricing. It would be interesting to study the statistical properties of this over-parameterized algorithm.

2.6 Semiparametric Two-Step GMM with Black-Box NN Nuisance Estimation in the First Step

Many empirical problems in economics can be expressed as semiparametric two-step GMM problems. A finite-dimensional parameter of interest $\theta_0 \in \Theta \subset \mathbb{R}^{d_\theta}$ is identified by an unconditional moment restriction (3) after an unknown nuisance function h_0 has been identified and estimated in a first step:

$$\mathbb{E}[g(W, \theta_0, h_0)] = 0, \quad \text{with } d_g := \dim(g) \geq d_\theta. \tag{3}$$

Here h_0 may be a generated regressor, a control function, a conditional mean, a conditional choice probability, a continuation value, or the solution to a nonparametric conditional moment restriction. Examples of θ_0 include various treatment effects under unconfoundedness, control-function IV, dynamic discrete choice, semiparametric demand systems, and models with missing or mismeasured data. There is an extensive literature on semiparametric two-step GMM which allows for any nonparametric (or machine learning) estimate $\hat{h}_n$ of h_0 in the first step; see, e.g., Andrews (1994); Newey (1994); Chen, Linton, and Van Keilegom (2003).

In this review, we focus on estimation of h_0 using Black-Box neural networks, while the second-step identification of θ_0, influence function correction, and semiparametric efficiency theory remain those of classical semiparametric two-step GMM. A convenient way to describe the plug-in procedure is as follows.

Step 1 (NN sieve for h_0). Estimate unknown nuisance function $h_0 \in \mathcal{H}$ by a NN sieve estimator $\hat{h}_n \in \mathcal{H}_n^{NN}$, either by NN sieve M-estimation or NN sieve minimum distance; see Sections 2.4.1–2.4.2. NN sieves are most useful here when $h_0(\cdot)$ depends on covariates/features of moderately high dimension or when an economic model suggests a compositional, ridge, or other low-complexity representation for $h_0()$ that might be difficult to capture with tensor-product linear sieves.

Step 2 (Optimally-weighted GMM for θ_0). Given $\hat{h}_n$, estimate θ_0 by the sample GMM criterion

$$\hat{\theta}_n = \arg\min_{\theta \in \Theta} \; \bar{g}_n(\theta, \hat{h}_n)' \, \hat{V}_g^{-1} \, \bar{g}_n(\theta, \hat{h}_n), \qquad \bar{g}_n(\theta, h) = \frac{1}{n} \sum_{i=1}^{n} g(W_i, \theta, h), \tag{4}$$

with a positive-definite weighting matrix $\hat{V}_g = V_g + o_p(1)$ and

$$V_g := \left[AVar \left(\sqrt{n} \bar{g}_n(\theta_0, \hat{h}_n) \right) \right]. \tag{5}$$

Unlike the usual GMM weight matrix based only on $g(W, \theta_0, h_0)$, this matrix V_g incorporates the first-order sampling effect of estimating h_0.

Let

$$\Gamma_\theta = \mathbb{E}[\partial_\theta g(W, \theta_0, h_0)], \qquad D_{h_0}[v] = \frac{\partial}{\partial \tau} \mathbb{E}[g(W, \theta_0, h_0 + \tau v)] \Big|_{\tau = 0}$$

denote the finite-dimensional Jacobian and the pathwise derivative with respect to the nuisance function. Under some regularity conditions, including a key condition of $|\sqrt{n}D_{h_0}[\hat{h}_n - h_0]| = O_p(1)$, Chen, Linton, and Van Keilegom (2003) obtain the expansions

$$\sqrt{n}\bar{g}_n(\theta_0, \hat{h}_n) = \frac{1}{\sqrt{n}}\sum_{i=1}^{n} g(W_i, \theta_0, h_0) \; + \; \sqrt{n}D_{h_0}[\hat{h}_n - h_0] \; + \; o_p(1) \rightsquigarrow \mathcal{N}(0, V_g), \tag{6}$$

and with $V_\theta = \left(\Gamma_\theta' \, V_g^{-1} \, \Gamma_\theta\right)^{-1}$, we have:

$$\sqrt{n}\left(\hat{\theta}_n - \theta_0\right) = -\left(\Gamma_\theta' \, V_g^{-1} \, \Gamma_\theta\right)^{-1} \Gamma_\theta' \, V_g^{-1} \, \sqrt{n}\bar{g}_n(\theta_0, \hat{h}_n) \; + \; o_p(1) \rightsquigarrow \mathcal{N}(0, V_\theta). \tag{7}$$

The term $\sqrt{n}D_{h_0}[\hat{h}_n - h_0]$ in (6) is the *correction* for estimating h_0 by $\hat{h}_n$ in the first step. Unfortunately, without specifying how h_0 is identified in Step 1, there is no closed-form expression of the correction term in general. Chen, Linton, and Van Keilegom (2003) and Armstrong, Bertanha, and Hong (2014) provide bootstrap methods for confidence intervals in semiparametric two-step GMM, including cases in which $g(\cdot)$ is nonsmooth in h and/or θ.

Riesz Representer, Influence Function, Neyman Orthogonality. One influential way to operationalize the correction term is to specify the first-step criterion. Suppose h_0 is identified as the *maximizer* of a population criterion $Q(h)$, such as a maximum-likelihood-like (M) criterion with $Q(h) = \mathbb{E}[\phi(W, h)]$, or a sieve minimum-distance criterion with $Q(h) = -\frac{1}{2}\mathbb{E}[m(X,h)'\Sigma(X)^{-1}m(X,h)]$ and $m(X,h) = \mathbb{E}[\rho(Y, h(\cdot)) \mid X]$. Chen and Liao (2015), building on Ai and Chen (2003, 2007); Chen, Liao, and Sun (2014), show that once this criterion $Q(h)$ is given, the effect of first-step estimation on the second-step moments can be represented and estimated through a *Riesz representer*. Intuitively, a Riesz representer v^* is the (least favorable) direction in the first-step function space along which small errors in estimating h_0 affect the moments used to estimate θ_0 in the second step. To state this formally, equip deviations $v = h - h_0, \; \tilde{v} = \tilde{h} - h_0$ with the criterion-induced norm and inner product

$$\|v\|^2 := -\partial_\tau^2 Q(h_0 + \tau v)\big|_{\tau=0}, \quad \langle v, \tilde{v}\rangle := -\partial_{\tau,\tilde{\tau}}^2 Q(h_0 + \tau v + \tilde{\tau}\tilde{v})\big|_{\tau=0,\tilde{\tau}=0}.$$

Let $\overline{\mathcal{T}}_{h_0}$ denote the $\|\cdot\|$- closure of the first-step tangent set for h_0. For $j = 1, \ldots, d_g$, define

$$D_{h_0,j}[v] := \frac{\partial}{\partial \tau} \mathbb{E}[g_j(W, \theta_0, h_0 + \tau v)]\Big|_{\tau=0}.$$

Whenever $\sup_{v \in \overline{\mathcal{T}}_{h_0}, v \neq 0} \{|D_{h_0,j}[v]|/\|v\|\} < \infty$, there is a Riesz representer $v_j^* \in \overline{\mathcal{T}}_{h_0}$ such that $\|v_j^*\| = \sup_{v \in \overline{\mathcal{T}}_{h_0}, v \neq 0} \{|D_{h_0,j}[v]|/\|v\|\} < \infty$ and

$$D_{h_0,j}[v] = \langle v, v_j^* \rangle \qquad \text{for all } v \in \overline{\mathcal{T}}_{h_0}. \tag{8}$$

Let $\Delta(W, h_0)[v]$ denote the mean-zero first-step score contribution in direction v induced by the criterion. For example, $\Delta(W, h_0)[v] = \partial_\tau \phi(W, h_0 + \tau v)|_{\tau=0}$ for M-estimation, and $\Delta(W, h_0)[v] = -\left(\partial_\tau m(X, h_0 + \tau v)\big|_{\tau=0}\right)' \Sigma(X)^{-1} \rho(Y, h_0)$ for minimum distance. Following Ai and Chen (2003) we have:

$$\sqrt{n} D_{h_0,j}[\hat{h}_n - h_0] = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \Delta(W_i, h_0)[v_j^*] + o_p(1)\,, \quad j = 1, \ldots, d_g.$$

Let $v^* = (v_1^*, \ldots, v_{d_g}^*)'$. Then the first-step adjustment in (6) can be written as

$$\sqrt{n} \bar{g}_n(\theta_0, \hat{h}_n) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \{\underbrace{g(W_i, \theta_0, h_0) + \Delta(W_i, h_0)[v^*]}_{=:\psi(W_i; \theta_0, h_0, v^*)}\} + o_p(1), \tag{9}$$

where $\psi(W_i; \theta_0, h_0, v^*)$ automatically satisfies the Neyman orthogonality:

$$\frac{\partial}{\partial \tau} \mathbb{E}[\psi(W; \theta_0, h_0 + \tau v, v^*)]\Big|_{\tau=0} = 0 \qquad \text{for all } v \in \overline{\mathcal{T}}_{h_0}. \tag{10}$$

Sieve Riesz Representer, Sieve influence function, Sieve Neyman Orthogonality. In complicated semiparametric models, such as nonparametric additive IV regression in the first step, Riesz representer v^* rarely has an analytic expression. Chen and Liao (2015) show that the effect of first-step estimation on the second-step moments can always be approximated and estimated through a *sieve Riesz representer*, $v_{k(n)}^* = (v_{1,k(n)}^*, \ldots, v_{d_g,k(n)}^*)'$, which always has a closed-form expression via linear sieve least squares regression. Let $v(\cdot) = P_{k(n)}(\cdot)'\gamma$ be in a $k(n)$-dimensional closed linear subspace $\mathcal{V}_{k(n)}$ of $\overline{\mathcal{T}}_{h_0}$. Let $R_{k(n)}$ be the Gram matrix induced by $\langle \cdot, \cdot \rangle$ so that $\gamma' R_{k(n)} \gamma = \|P_{k(n)}(\cdot)'\gamma\|^2$. Then the sieve Riesz representer

$v^*_{j,k(n)} \in \mathcal{V}_{k(n)}$ can be computed in closed form via least squares:

$$v^*_{j,k(n)}(\cdot) = P_{k(n)}(\cdot)' R^{-1}_{k(n)} D_{h_0,j}[P_{k(n)}], \qquad j = 1, \ldots, d_g,$$

where the linear sieve dimension $k(n)$ can be chosen so that

$$\|\hat{h}_n - h_0\| \times \max_{j=1,\ldots,d_g} \|v^*_{j,k(n)} - v^*_j\| = o_p(n^{-1/2}). \tag{11}$$

The hybrid NN–linear-sieve strategy has a clean division of labor: neural-network sieves in Step 1 provide flexible approximation of h_0, while the linear sieve Riesz step delivers transparent correction terms, optimal weighting, and standard sandwich inference for θ_0; see Chen and Liao (2015) for details. We can define a *sieve influence function* for the second-step moment by

$$\psi(W; \theta_0, h_0, v^*_{k(n)}) := g(W, \theta_0, h_0) + \Delta(W, h_0)[v^*_{k(n)}]. \tag{12}$$

This is a computable finite-dimensional approximation to the unknown adjustment in (6) and (9). The correction removes the first-order effect of estimating h_0 on $\hat{\theta}_n$, so the limiting moment is Neyman-orthogonal, locally robust, or debiased in the sense of Chernozhukov et al. (2018) and Chernozhukov et al. (2022). The object is conceptually the same across the sieve semiparametric two-step GMM and DML literatures. What differs is mainly implementation: the sieve literature typically estimates h_0 by a general sieve (including NN sieve) and the v^* by a closed-form sieve Riesz representer estimate and imposes the rate condition (11), whereas the DML literature estimates nuisance functions h_0, v^* by generic machine-learning methods, uses cross-fit and the following rate condition:

$$\|\hat{h}_n - h_0\| \times \max_{j=1,\ldots,d_g} \|\hat{v}^*_n - v^*_j\| = o_p(n^{-1/2}).$$

Semiparametric Efficiency Under Locally Just-Identified First Step. Many nonparametric models in the first-step are of the form: $\mathbb{E}[\rho(Y, h_0(X)) \mid X] = 0$, which is nonparametric just-identified (Ackerberg et al. (2014); Chen and Santos (2018)). For a general first-step locally just-identified setting, Ackerberg et al. (2014) establish that the optimally weighted GMM estimator (4) in Step 2 is *semipara-*

metrically efficient for θ_0, with $V_\theta = \left(\Gamma_\theta' V_g^{-1} \Gamma_\theta\right)^{-1}$ as the efficient variance bound. Ackerberg et al. (2014) also provide an easy-to-compute sieve-based optimal weight matrix in Step 2 GMM estimation and a practical efficient variance estimator $\hat{V}_\theta$ that avoids estimating the Riesz representer directly.[7] Further, (9) becomes

$$\sqrt{n}\bar{g}_n(\theta_0, \hat{h}_n) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \underbrace{g(W_i, \theta_0, h_0) + v^*(X_i)\rho(Y_i, h_0(X_i))}_{=\psi(W_i;\theta_0,h_0,v^*)} + o_p(1), \tag{13}$$

where the Riesz representer $v^* \in L_2(X)$ has a closed-form expression. Ackerberg et al. (2014)'s results extend the key insight of Newey (1994) that, in a just-identified first step, the asymptotic variance of the second-step plug-in estimator does not depend on how h_0 is consistently estimated in the first step.

This review also helps connect recent contributions. Farrell, Liang, and Misra (2021a) and Farrell, Liang, and Misra (2021b) embed multi-layer NNs in Neyman-orthogonal scores, in the sense of Chernozhukov et al. (2018), for treatment effects and related smooth linear functionals. They allow estimation of both unknown h_0 and v^* using DNNs and perform cross fit DML for efficient estimation. Chen et al. (2024) derive semiparametrically efficient inference procedures for general treatment effect parameters when nuisance functions h_0 are approximated by neural-network sieves, without estimating the Riesz representer v^* or cross-fitting. We note that the nonparametric models in the first-step considered in these papers turn out to be nonparametrically just-identified.

Semiparametric Efficiency Under Locally Over-Identified First Step. When h_0 is identified by conditional moments $\mathbb{E}[\rho(Y, h_0(\cdot)) \mid X] = 0$, as in nonparametric IV regression or semiparametric models, the first step may be locally over-identified (Chen and Santos (2018)). In this case, the just-identified invariance result above need not apply; the weighting of the first-step conditional moments can affect the asymptotic variance for the second step GMM estimation of θ_0.

Ai and Chen (2012) establish the semiparametric efficiency bound for sequential moment restriction models that include two step models as a special case. They also provide an optimally weighted orthogonalized sieve minimum distance (SMD) efficient estimator. Ai and Chen (2012)'s construction starts from a

[7] Ackerberg, Chen, and Hahn (2012) consider an extension of the semiparametric model where the nuisance function h depends on unknown arguments, possibly different coordinate-by-coordinate.

forward-filtered residual. Define

$$\varepsilon(W;\theta,h) := g(W;\theta,h) - \Gamma(X)\rho(W;h) \quad \text{forward-filtered residual,}$$

$$\Gamma(X) = \mathbb{E}[g(W;\theta_0,h_0)\rho(W;h_0)' \mid X]\Sigma_0(X)^{-1}, \quad m(X,h) = \mathbb{E}[\rho(W;h) \mid X],$$

$\Sigma_0(X) = \mathbb{E}[\rho(W;h_0)\rho(W;h_0)' \mid X]$ and $\Sigma_1 = \mathbb{E}[\varepsilon(W;\theta_0,h_0)\varepsilon(W;\theta_0,h_0)']$.

Instead of Ai and Chen (2012)'s optimally weighted orthogonalized SMD procedure, we propose the following alternative semiparametric efficient two-step procedure:

Step 1 *Optimally weighted NN sieve minimum distance for* h_0. Let $\hat{\Sigma}_0(X)$ be a consistent estimate of $\Sigma_0(X)$.

$$\hat{h}_n \in \arg \inf_{h \in \mathcal{H}_n^{NN}} \frac{1}{n}\sum_{i=1}^{n} \hat{m}(X_i,h)'\hat{\Sigma}_0(X_i)^{-1}\hat{m}(X_i,h)$$

Step 2 *Optimally weighted GMM for* θ_0 *using forward-filtering residuals*. Let $\hat{\Gamma}(X)$ be a consistent estimate of $\Gamma(X)$.

$$\hat{\theta}_n := \arg\min_{\theta\in\Theta} \bar{\varepsilon}_n(\theta,\hat{h})'\hat{V}_\varepsilon^{-1}\bar{\varepsilon}_n(\theta,\hat{h}), \quad \bar{\varepsilon}_n(\theta,h) := n^{-1}\sum_{i=1}^{n}[g(W_i;\theta,h) - \hat{\Gamma}(X_i)\rho(W_i;h)]$$

where $\hat{V}_\varepsilon$ is a consistent estimator of

$$V_\varepsilon := AVar\left(\sqrt{n}\bar{\varepsilon}_n(\theta_0,\hat{h}_n)\right). \tag{14}$$

Theorem 2.7 (Ai and Chen (2012), Informal)**.** *Under some regularity conditions on the models* $\mathbb{E}[\rho(Y,h_0(\cdot))|X] = 0,\ \mathbb{E}[g(W,\theta_0,h_0(\cdot))] = 0$*, the Step–2 estimator satisfies*

$$\sqrt{n}(\hat{\theta}_n - \theta_0) = -\left(\Gamma_\theta' V_\varepsilon^{-1}\Gamma_\theta\right)^{-1}\Gamma_\theta' V_\varepsilon^{-1}\sqrt{n}\bar{\varepsilon}_n(\theta_0,\hat{h}_n) \ + \ o_p(1) \rightsquigarrow \mathcal{N}\big(0,\ (\Gamma_\theta' V_\varepsilon^{-1}\Gamma_\theta)^{-1}\big),$$

with $(\Gamma_\theta' V_\varepsilon^{-1}\Gamma_\theta)^{-1} = \Omega_\theta^*$*, which is the Ai and Chen (2012)'s semiparametric efficient variance bound for* θ_0 *defined by the joint model.*

Thus, a two-step GMM estimator based on the forward-filtered unconditional moment reaches the Ai–

Chen bound only when the first step is itself efficiently weighted and the second-step variance accounts for the sampling effect of estimating h_0. This is the same logic behind the OP–OSMD and efficient-influence function procedures in Chen, Chen, and Tamer (2023) for NPIV average derivatives with NN sieve for h_0. Similarly, Chen, Liao, and Wang (2025) derive optimal inference procedures when nuisance functions are approximated by NN sieves under time series weakly dependent data, without the need to estimate the Riesz representer.

Semiparametric Two-Step with Black-box AI/LLM First Step. A practical advantage of the above filtered two-step procedure is that it separates flexible first-step learning from efficient downstream inference. The first step may use a black-box AI/LLM, neural network, or other high-capacity learner to estimate a rich, possibly locally over-identified conditional-moment model and the associated nuisance/residual objects. The second step can then be interpreted as econometric downstream fine-tuning: it forward-filters the moment relevant for θ, optimally reweights the filtered residual, and computes the final low-dimensional estimate and sandwich variance. Subject to the usual identification, consistency, rate, cross-fitting or sample-splitting, and covariance-estimation conditions, this downstream step can in principle turn a predictive or generative black-box first step into an orthogonalized, semiparametrically efficient estimating equation for θ_0. Nevertheless, the locally overidentified first step models could be misspecified and hence the black-box AI/LLM estimates pseudo-true functions h^* in the first step (see Ai and Chen (2007)).

2.7 Overall Lesson

This section illustrates three complementary roles for neural nets in econometrics: as constrained nonlinear sieves inside classical loss criterion functions such as maximum-likelihood-like, minimum-distance, or GMM; or as flexible parameterization of both sides of a minimax problem; or as nuisance estimators within semiparametrically efficient plug-in or influence function (i.e., orthogonal score) frameworks. The common thread is that the econometric structure—moment restrictions, shape restrictions, semiparametric influence function, and test inversions—still does the identification and inferential work, while the neural net supplies flexible approximation to unknown functions or unknown operators in high-dimensional settings.

At present, however, inference theory using NN sieves remains uneven across objects: most NN

successes to date concern prediction or classification M-estimation problems with large data sets. For NN sieve M-estimation and out-of-sample stable generalization, Minimum Description Length (MDL) regularization has attractive theoretical properties for shallow networks, see Barron (1991); Barron, Rissanen, and Yu (1998); Chatterjee and Sudijono (2026). Monte Carlo studies in Chen, Chen, and Tamer (2023) indicate that the finite-sample performance of NN sieves for NPIV is sensitive to the choice of tuning parameters, however. Better algorithms may, in the near future, make NN sieves more useful for estimation of nonparametric conditional moment restriction models. Finally, while existing semiparametric two-step GMM theory for asymptotically linear regular functional θ is still applicable with first-step nuisance function estimated by NN sieves, there are no uniform confidence bands for NN sieve estimated conditional mean, density, or nonparametric instrumental variables regression. There is no adaptive data-driven choice of tuning parameters for any NN sieve based procedures either.

3 Modern NNs and Large Language Models

Progress in performance of neural nets over the last two decades can roughly be attributed to advances in two areas. First, increased computational power, particularly through GPUs and GPU-optimized implementations, has enabled the training of deeper networks with larger numbers of parameters. We review these developments alongside general comments on computation in Section 3.1. Second, a range of specialized architectural components and network designs, especially transformers, have been proposed. These are the focus of Section 3.2. In Section 3.3, we present three popular generative models: Autoencoders, Generative Adversarial Networks, and Diffusion Models. For LLMs specifically, we discuss reinforcement learning with human feedback and verifiable feedback in Section 3.4, and fine-tuning in Section 3.5.

3.1 High-Dimensional Optimization

Optimizing the parameters of a neural network amounts to solving a high-dimensional, typically non-convex optimization problem. The workhorse method is gradient descent. Given parameters θ_k and objective

$$f^{\text{obj}}(\theta) = \frac{1}{n}\sum_{i=1}^{n} f^{\text{loss}}(y_i \mid X = x_i, \theta),$$
$$\theta_{k+1} = \theta_k - \alpha_k \nabla f^{\text{obj}}(\theta_k) = \theta_k - \alpha_k \frac{1}{n}\sum_{i=1}^{n} \nabla f^{\text{loss}}(y_i \mid X = x_i, \theta_k), \tag{15}$$

where (α_k) is the learning-rate sequence. Large step sizes move quickly but may overshoot; small step sizes are more stable but can be slow. In practice, schedules and adaptive learning rates are common.

Equation (15) also explains why neural-net training can exploit modern hardware efficiently: the sample average decomposes over observations, so gradient evaluation is highly parallelizable. GPUs, originally designed for fast matrix operations in graphics, are well suited to this task, as shown by Oh and Jung (2004); Steinkraus, Buck, and Simard (2005), because computing the gradient reduces to a sequence of matrix-vector multiplications. Computing the full gradient can still be expensive, so, following the Stochastic Approximation idea of Robbins and Monro (1951), one often replaces it with a mini-batch update,

$$\theta_{k+1} = \theta_k - \alpha_k \frac{1}{|B|}\sum_{i \in B} \nabla f^{\text{loss}}(y_i \mid X = x_i, \theta_k), \tag{16}$$

where $B \subset \{1, \ldots, n\}$ is a random batch of training observations. Smaller batches are cheaper and sometimes regularize more strongly, while larger batches give a more accurate gradient at higher computational cost, see Wilson and Martinez (2003); Keskar et al. (2017).

A second computational challenge is differentiation. Manual differentiation and implementation are possible but tedious and error-prone; numerical finite differences can be inaccurate and scale poorly to high dimensions. Symbolic differentiation uses the rules of calculus to obtain an analytical expression for derivatives as a function of inputs. As noted by Corliss (1988), this analytical expression is often

large, although, as shown by Guenter (2007), this can be mitigated using common subexpressions. Automatic differentiation (AD), by contrast, does not construct symbolic expressions but instead evaluates derivatives alongside the original function. For a given implemented function, AD systematically applies the chain rule over the sequence of elementary operations that define the computation. AD can be applied in two principal modes: forward mode, which propagates derivatives of intermediate variables with respect to the inputs alongside evaluation, and reverse mode, which propagates sensitivities of the output with respect to intermediate variables backward through the computation. For neural networks, reverse-mode AD is called backpropagation (or backprop), and it is particularly efficient when the output is low-dimensional but the parameter vector is large. Paszke et al. (2017) developed PyTorch's `autograd`, an automatic-differentiation engine that combines AD with optimized implementations of common gradient computations, not merely a library of hand-coded derivatives. See also Baydin et al. (2018); Laue (2019); Maclaurin (2016). Several variants of gradient descent have been proposed, including Momentum by Polyak (1964), Nesterov acceleration by Nesterov (1983); Sutskever et al. (2013), Adagrad by Duchi, Hazan, and Singer (2011), RMSProp by Hinton, Srivastava, and Swersky (2012), Adam by Kingma and Ba (2014), AdamW by Loshchilov and Hutter (2017), and LAMB by You et al. (2019).

When choosing initial values, two principles matter. First, the starting values are often chosen randomly to break the weight-space symmetry inherent to neural networks: one can permute neurons in one layer and the weight and bias matrices for that layer and obtain the identical output. Second, the scale of the random initial values should keep activations and gradients in a reasonable range. Larger variance is therefore not always better: one needs sufficient randomness to break symmetry, but not so much that gradients explode or vanish. Standard schemes such as Xavier initialization, due to Glorot and Bengio (2010), and He initialization, due to He et al. (2015), choose the variance as a function of layer width precisely to stabilize gradient propagation. In the infinite-width limit, the network output at initialization behaves like a Gaussian process with mean zero and, for normalized inputs, constant variance, see Yang (2019).

Regularization Neural nets are flexible enough to approximate rich function classes, but this flexibility comes at the cost of high variance. A standard approach is penalized empirical risk minimization:

$$\hat{\theta} = \arg\min_{\theta} \left\{ \frac{1}{n} \sum_{i=1}^{n} f^{\text{loss}}(y_i \mid X = x_i, \theta) + \lambda \|w(\theta)\|_p^p \right\},$$

where $w(\theta)$ stacks the weights (and rarely the biases) of the network. Setting $p = 2$ is called Ridge regularization or weight decay, while $p = 1$ encourages sparse solutions. Other common regularizers are dropout, due to Srivastava et al. (2014), and early stopping, as in Morgan and Bourlard (1989). Early stopping is often explained operationally: monitor validation performance and stop when additional epochs no longer improve it, see chapter 7 of Goodfellow, Bengio, and Courville (2016) for a textbook description of the algorithm. Minimum Description Length regularization has recently been shown by Chatterjee and Sudijono (2026) and Abudy et al. (2025) to have some attractive theoretical properties for generalization.

3.2 Architectures

A *deep* neural network has multiple hidden layers. Depth can improve representation and feature reuse, but it also makes optimization harder. Deeper networks are more susceptible to exploding or vanishing gradients, as documented by Hochreiter (1991); Bengio, Simard, and Frasconi (1994), and in practice they may even have higher training error than shallower alternatives if the architecture is poorly chosen, a phenomenon dubbed the "degradation problem" by He et al. (2016). Thus, neural network architecture is critical. In Section 2.3, we review various economically motivated shape constraints.

Activation functions have evolved from step functions in Rosenblatt's perceptron to sigmoid and tanh before the turn of the millennium, and more recently toward piecewise-linear and smooth, non-saturating alternatives such as ReLU, SiLU, and GELU. See appendix A for references and a visualization.

Residual connections help gradients flow through deep stacks of layers. Residual networks, introduced by He et al. (2016), learn deviations from the identity map rather than full transformations from scratch. Normalization layers stabilize training. Batch normalization, proposed by Ioffe and Szegedy (2015), normalizes activations across a mini-batch, while layer normalization of Ba, Kiros, and Hinton

(2016) normalizes across features of a single observation and is therefore better suited to variable-length sequences such as text. These ideas are central to the specialized architectures discussed next.

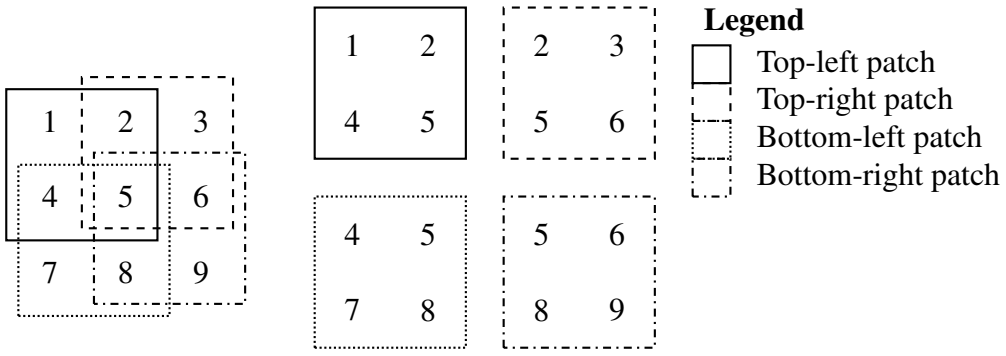
3.2.1 Convolutional Neural Networks Consider the problem of object recognition in images. A grayscale image can be represented as a matrix in $\mathbb{R}^{r\times c}$; a color image is the same object with three color channels (red, green and blue) and therefore represented as $\mathbb{R}^{r\times c\times 3}$. A fully-connected network typically "flattens" this array before processing it, for example by concatenating the rows of a grayscale image. This is problematic for at least two reasons: the dimensionality of the neural network and the structure of the data. First, flattening destroys local spatial structure: pixels that are neighbors in the image may be far apart in the resulting vector. To capture the dependence between pixels which are close in the image but far apart in the flattened vector, we need a dense layer with many parameters. For example, consider an RGB image of 1024×768 pixels (1K resolution) in three channels for a total of 2,359,296 numbers. Even if we only want 100 neurons in the first hidden layer, that would require 235,929,600 weight parameters, a very high-dimensional parameter vector. Second, if an object in an image is shifted by a few pixels, we expect the classification of the image not to change much. However, in naive fully-connected layers, moving (blocks of) pixels a little bit to the left or right has the potential to change the predicted label.

Convolutional neural nets (CNNs) preserve locality of images by looking at small patches of images of size $u \times v$. For example, patches of 3x3, 5x5 or 7x7 pixels are common. Let $X_p \in \mathbb{R}^{u\times v\times d_{\text{in}}}$ denote the image patch, and let $K^f \in \mathbb{R}^{u\times v\times d_{\text{in}}}$ denote a filter. A convolutional feature map is then

$$h_p^f = \sigma\Big(\text{sum}(K^f \cdot X_p) + b^f\Big),$$

where $K^f \cdot X_p$ is the element-wise product of image-patch and filter. The resulting matrix is then summed over, added to a bias term and fed through an activation function, in this case the sigmoid. This produces a filter-patch specific activation $h_p^f \in \mathbb{R}$. We then slide filter f through the image by considering the next patch which is s pixels to the right of the current patch where s is referred to as the "stride". Higher values of s reduce the number of patches considered and lead to fewer filter-patch specific activations. The larger the stride and the patch, the fewer activations there are. To preserve the dimensionality of the problem, it is common to "zero-pad" the image by adding zeros around it in all channels.

Figure 2: Illustration of a Filter Sliding Through an Image



CNNs depart from fully-connected neural networks in two ways. The first difference is *parameter sharing*: the same filter K^f is used at every patch p. The second important difference is the restriction of connections: Two pixels which, in the image, are further away than the size of the patch, never appear in the same filter-patch activation. CNNs are statistically and computationally more efficient than fully-connected neural nets when the relevant patterns are local and translation-invariant.

Using many filters leads to many hidden values. For example, a 256x256x3 image may be transformed into a 256x256x100 array if we apply 100 filters. Statistically and computationally, it is challenging to estimate coefficients for all of these hidden values. A common idea to reduce the dimensionality of the hidden values is to use *pooling*. A common way to do this is to take the maximum (max-pooling) or average (average-pooling) of several adjacent values of hidden units. The advantage of this approach is that it encourages local translation invariance. A convolutional layer is a block in which linear filters are applied (the convolution step), the nonlinear activation function is applied (the detector step) and pooling happens. Common architectures of CNNs involve several convolutional layers on top of each other, often ending up with several fully-connected neural layers. Notable CNN architectures include LeNet by LeCun et al. (1998), AlexNet by Krizhevsky, Sutskever, and Hinton (2012), VGGNet by Simonyan and Zisserman (2014), InceptionNet/GoogLeNet by Szegedy et al. (2015), ResNet by He et al. (2016), Densenet by Huang et al. (2017), and EfficientNet by Tan and Le (2019). As we will see below, many economic papers use these pretrained models, often available on Hugging Face, and fine-tune them to their application.

3.2.2 Graph Neural Networks A graph (or network) is a set of vertices V and edges E. For example, the vertices could be countries and the edges could be the trade-flows between countries. Graph neural networks (GNNs) are specifically designed for data from a graph. The basic idea is that each vertex has some characteristics and that conditional on the characteristics, the identity of the vertices is irrelevant. GNNs were first introduced in Scarselli et al. (2008) and have seen many adaptations. A recent survey with references to previous surveys is Khemani et al. (2024). A textbook reference is Wu et al. (2022).

A GNN typically has as many neurons in each layer as there are vertices. Hence, each neuron can be interpreted as the embedding of a particular vertex. The i-th component of layer $l+1$ depends on the i-th component of layer l and the components of $N(i)$ in layer l where $N(i)$ is the set of nodes connected to node i in the network's adjacency matrix. It is thus a feedforward network where certain links between neurons are "turned off" as a function of the adjacency matrix. GNNs are permutation invariant: the output does not depend on the ordering of vertices, which is ensured by aggregating neighbor information symmetrically, as in Graph Isomorphism Networks (GINs) of Xu et al. (2018) or Principal Neighborhood Aggregation (PNA) of Corso et al. (2020). Corso et al. (2020) propose to weigh these aggregate statistics with "degree-scalars" to alleviate the exploding gradient problem.

3.2.3 Transformers Transformers are the basis of the recent successes of LLMs. When a transformer-based chatbot is presented with a prompt, it first splits up the prompt into tokens using a tokenizer, e.g. Byte-Pair Encoding by Sennrich, Haddow, and Birch (2016). Initially, tokens are represented as one-hot encodings in the large set of tokens, the "vocabulary". For example, for GPT-3, there are $d_V = 50,257$ tokens. The vectors representing each token are then concatenated and multiplied by an embeddings matrix to bring them into a lower dimensional space, token by token. For GPT-3, this embedding space has dimensionality of $d_E = 12,288$. But natural language does not work token-by-token or word-by-word. Take, for example, the word "interest". Its meaning can change depending on its context, for example "I have much interest in cars." and "I have much interest on my car loan." Such contextual meaning is not contained in the original embedding.

The attention mechanism of Bahdanau (2014); Vaswani et al. (2017) updates tokens' embeddings to

reflect contextual meaning. To gain intuition, let's suppose again that tokens are words. Then in the two example sentences for "interest" above, we can imagine that the word "interest" in the second sentence is "attentive" to the word "loan", i.e. that its embedding must be updated to reflect that "interest" here is meant as a financial interest rather than an interest motivated by curiosity. Imagine that each token asks a question of the other tokens to update its embedding. Each token may ask different questions, e.g. the token "interest" may ask the question "Is this a sentence about finances?" Another token may ask another question. The token "have" may either be a verb of its own or used to form the present perfect and have no semantic meaning other than the tense. It may ask "Is there a past participle in this sentence?". These questions are embedded as numeric vectors, usually with an embedding dimension that is much smaller than the dimension of the original token embedding. In GPT-3 for example, the dimension of the embedding space for tokens is $d_E = 12,288$ while the dimension of the embedding for questions is just $d_Q = 128$. We call the "short" embedding of the question a "query". To learn which query is "asked" by each token, the embedding vector of each token is multiplied by a weight matrix W_Q with d_Q rows and d_E columns. If we stack the embedding vectors of all embeddings into a matrix E with d_E rows and d_T columns, where d_T is the number of tokens, then the operation of computing each token's query can be implemented with a matrix multiplication: $Q := W_Q E$, where Q are the queries.

How are these questions, represented as queries, answered? Intuitively, the query of the token "interest" may be attentive to tokens associated with financial meaning, such as "loan", "pay", "rate", "mortgage" etc. These answers are also represented as numerical vectors, of the same size as the queries. These embedded answers are called *keys*. To obtain them, we multiply the embedding matrix E with another weight matrix W_K, the weight matrix for the keys: $K = W_K E$ where W_K is of the same dimension as W_Q.

Which queries and keys correspond to each other? Whether a specific query points in the same direction as a specific key may be measured using the angle calculated via the dot product. This motivates taking the dot product of all query-key pairs. In our intuitive example, we can imagine that the query "Is this a sentence about finances?" might be answered by a key that has a large W_K weight on the token "loan". The entire operation of computing matches between query-key pairs can be represented by matrix products:

we compute matches defined as $M := Q^t K/\sqrt{d_Q}$ where the division by $\sqrt{d_Q}$ serves to scale these inner products even though it is not exactly the product of the norms of query embedding and key embedding. We can interpret $M_{i,j} = Q^t_{\cdot,i} K_{\cdot,j}/\sqrt{d_Q}$ as the degree to which query token i and key token j correspond to one another. It therefore makes sense to compare the 'match' for a specific row across columns, since large positive values mean that there is a strong match between query and key, whereas small, negative or even large negative values mean that there is not a strong match between query and key. It is useful to scale the matching matrix row-wise by applying the softmax function so that each row is non-negative and sums up to one. The resulting matrix is called the attention matrix $A \in \mathbb{R}^{d_T \times d_T}$. $A_{i,j}$ indicates how much attention each token i pays to token j in the prompt.

How do we update the attention matrix to update the embeddings? In our intuitive example, we might want to update the meaning of "interest" in the direction of "loan". To capture this intuition, transformers compute how much each token's embedding would have to be changed if it pays attention to a specific other token. This is done by a *value* matrix W_V with d_E rows and columns[8], that creates a matrix of updates, i.e. $U = W_V E$. The j-th column $U_{\cdot,j}$ indicates how much embedding vector $E_{\cdot,k}$ would be changed if it paid full attention to token j: the updated token would be $E_{\cdot,k} + U_{\cdot,j}$ for any k. In general, token k's embedding is updated with a weighted sum of updates. The weights are given by the attention matrix: $E \leftarrow E + UA^t$.

What we described in this section is the architecture of one attention head. Transformers typically work with multi-head attention where the embeddings go through various attention heads at once in parallel and the embedding updates of each attention head are then added to the original embedding matrix. For example, GPT-3 uses 96 attention heads. The intuition is that different heads learn different ways in which one token influences the meaning of another token.

[8]The value matrix W_V is often factorized to reduce its memory requirements, as $W_V = W_{V,1} W_{V,2}$ where the number of columns of $W_{V,1}$ (and therefore, the number of rows of $W_{V,2}$) is much lower than the number of rows or columns of W_V. This also has other computational advantages: it speeds up matrix multiplication by W_V into two smaller matrix multiplications.

Architecture Aspects For natural language processing, the key advantage of the transformer architecture of Vaswani et al. (2017) over previous architectures such as recurrent neural nets (RNN) and its variants such as Long Short-Term Memory RNNs of Hochreiter and Schmidhuber (1997) and Gated recurrent units of Cho et al. (2014) is the replacement of sequential calculations with parallel computation.[9] RNNs work sequentially over the tokens of the prompt while the tokens flow through the transformer network in parallel. We will return to this in Section 3.2.4. RNNs have advantages in other applications, such as nonlinear ARMA time series; see Chen et al. (2025).

Transformers have been developed for various tasks, e.g. to classify text (e.g., is a review positive or negative), translate between languages (e.g., English to German) or to complete prompts (e.g., in a chatbot). Depending on the task, slight variations in the transformer architecture have been proposed. In each case, the attention layer is the key component of transformer architectures. Three main variations of transformer architectures have been put forward: encoder-only, decoder-only and encoder-decoder architectures.

An *encoder architecture* takes an input in natural language and outputs numbers or vectors. For example, the input could be a review of a restaurant and the output could be a number between 0 and 1 where higher numbers indicate that the restaurant is expensive, or that the service is quick, or that the food is spicy etc. Intuitively, encoders can understand text but not produce it. Examples are BERT by Devlin et al. (2019), which uses a bidirectional encoder, i.e. no masking, and variants (RoBERTa is more carefully trained, while ALBERT by Lan et al. (2019) and DistilBERT by Sanh et al. (2019) have fewer parameters and are easier to train but less expressive), as well as Electra by Clark et al. (2020), which replaces masking and predicting tokens for training with replaced token detection.

A *decoder architecture* is essentially a specific encoder with a very specific output target: given a prompt, it derives a probability distribution over the next token. By sampling from this distribution iteratively, a decoder architecture can generate texts. In other words, decoders are *language models*. Examples include the GPT family (GPT-1, GPT-2, GPT-3, ChatGPT, GPT-4, GPT-5, CodeX, InstructGPT),

[9]The other advantage is that the exploding/vanishing gradient problem is less severe.

Bard, Gemini, Claude, Llama, Jurassic (1 and 2).

An *encoder-decoder architecture* brings together an encoder to embed the input prompt into an embedding space and update it using the attention-based update described above while a decoder takes the output of the encoder and returns a sequence of tokens. Use cases that may be relevant for economic research are translation between human languages, programming languages, summarizing documents, transcribing audio, or describing images. Examples are the original transformer by Vaswani et al. (2017), BART by Lewis et al. (2019), UL2 by Tay et al. (2022), and T5 by Raffel et al. (2020) and its variations (mT5, Flan-T5).

Completion of prompts builds on the sequential prediction of individual tokens. During training the transformer will take a sentence from a library, e.g. "I need to pay my monthly rent.", and transform this sentence without a label into a sentence with a label by *masking* the last word, e.g. the transformer tries to predict the next word following the prompt "I need to pay my monthly (?)". If we allowed the transformer to update the embeddings of the tokens in this masked prompt by paying attention to the token "rent", then the transformer uses information for prediction that it will not have in practice. To prevent this *information leakage*, it is common to set the upper triangle of the match matrix M to negative infinity so that attention of earlier tokens with respect to later tokens is set to zero after applying the row-wise softmax. For example, the first token in the sentence, "I" would not update its embedding at all with respect to the other tokens. A second use of masking is to handle prompts of varying length: Shorter prompts are padded with a dummy token until all prompts are of the same length. However, we do not want the meaningful token embeddings to pay attention to dummy tokens or the number of dummy tokens. This can also be achieved by masking.[10]

The number of tokens considered by a transformer is called the *context size*. The trade-off here is clear: larger context sizes lend themselves to capturing long-range dependencies but require more parameters, i.e. more memory and computational resources and time. As an example, GPT-3 uses a context size of $d_C = 2048$. On average, a token used by GPT-3 has 4 characters while a word in English has slightly less than 5 characters on average, although, as documented by Bochkarev, Shevlyakova, and Solovyev (2015),

[10]Transformer architectures for contexts of variable length have been proposed, e.g. Dai et al. (2019).

this changes over time. Thus, the context size of GPT-3 is roughly 1638 English words. In many economics research papers, this means that the introduction will be "forgotten" by the time the methodological section is reached.

Different languages may use different tokens. For instance, many Spanish nouns end in "-ción", e.g. "educación", or "instrucción". In English, there is no "ó" and the ending "-tion" is often used for words of Latin origin, e.g. "education" or "instruction". Here, it is often practical to use one tokenization and embedding for one language, e.g. English, and one tokenization and embedding for another language, e.g. Spanish. For translation tasks, this is important. Consider the problem of translating from English to Spanish. This will be done in an encoder-decoder architecture. The encoder will embed the English tokens and use the attention mechanism described above to update each English token to the surrounding English tokens to which it pays attention. This attention mechanism is called *self-attention* because it determines the attention of English tokens with respect to other English tokens. The result of the encoder is an embedding matrix and a set of attention matrices. The decoder is like the encoder in that it first takes in a sequence of Spanish tokens and learns their embedding together with the attention matrices that describe which Spanish token is attentive to which other Spanish token. Then, the embedding of the encoder and the embedding of the decoder go through a *cross-attention* layer that is almost identical to the self-attention layer except that now keys and values come from the source language while the queries come from the target language. Intuitively, cross-attention allows one to learn which tokens from one language (e.g. "-tion" in English) correspond to which token in another language (e.g. "-ción" in Spanish). For additional details, see Wu et al. (2016). Chapter 12 in Bishop and Bishop (2023) provides a textbook treatment.

3.2.4 State Space Models, Mamba A drawback of transformers is that their computational cost typically grows quadratically with sequence length. State space models (SSMs) are a prominent alternative because they can scale linearly or near-linearly in the number of tokens. Economists will recognize the starting point, a linear recursion

$$y_i = W_x x_i + W_y y_{i-1}. \tag{17}$$

With Gaussian shocks and an observation equation, this is closely related to the linear state-space systems studied with Kalman filtering. Modern machine-learning SSMs differ because the transition and readout components are learned flexibly and are optimized for large-scale sequence modeling rather than classical filtering alone.

A major advantage of linear SSMs is that they can be parallelized using scan (prefix-sum) algorithms; see Blelloch (1990); Orvieto et al. (2023). Structured state space models such as S4 use carefully chosen parameterizations and initializations to stabilize long-range memory, see Gu et al. (2020); Gu, Goel, and Ré (2021). Mamba, proposed by Gu and Dao (2023), builds on this line of work by letting parts of the state-space update depend on the current input, so that the model can adaptively decide which information to keep and which to forget. We therefore view Mamba as one influential member of the broader SSM family. For an economic example, see Mehrabian et al. (2025), which combines bi-directional Mamba blocks and graph neural nets for stock-market prediction.

3.3 Generative Neural Networks

Generative neural nets aim to learn a distribution and then sample from it. For the univariate case, readers will be familiar with the problem: Kernel density estimation is often used in economics to estimate a distribution by estimating its density, say $\hat{f}$. Given an estimate of the density, one can compute the implied cdf $\hat{F}$ and its quantile function $\hat{F}^{-1}$. One can then draw from this distribution by sampling $U \sim U[0,1]$ and defining $X = F^{-1}(U)$. Then X will have cdf F and density f because $\mathbb{P}[X \leq t] = \mathbb{P}[F^{-1}(U) \leq t] = \mathbb{P}[U \leq F(t)] = F(t)$. But how can we extend this familiar method from the univariate to the multivariate, and in fact, the high-dimensional case? We may like to generate text, images, videos, or economic datasets with many variables.

3.3.1 Autoencoders Autoencoders, introduced by Hinton and Salakhutdinov (2006), take a high-dimensional object, for example an image, as an input. Consider an image with 1 million pixels. The image is fed through a sequence of hidden layers with fewer and fewer neurons. The thinnest layer, the bottleneck, may only have 10 neurons. The sequence of layers leading up to the bottleneck is called the encoder. Intuitively, the encoder represents the 1 million pixel numbers with 10 neurons. Then, a decoder

with an increasing number of neurons in each layer reconstructs the input, with an output layer of 1 million neurons representing the reconstructed pixel values. The autoencoder minimizes the difference between the input pixels and the output pixels. One could now try to randomly draw 10-dimensional vectors and feed them through the decoder to obtain an image. However, this unfortunately mostly generates noisy images of poor quality, so that one cannot effectively generate new images.

Variational autoencoders (VAE), developed by Kingma and Welling (2013); Rezende, Mohamed, and Wierstra (2014), change the bottleneck structure to allow for effective generation of images. Specifically, the deterministic bottleneck with 10 neurons is replaced by drawing a latent random variable. Given input w, the encoder outputs a mean vector $\mu(w)$ and a variance vector $\sigma^2(w)$, after which one draws

$$z = \mu(w) + \sigma(w) \odot \varepsilon, \qquad \varepsilon \sim N(0, I_{10}),$$

where $\odot$ denotes elementwise multiplication. Once z is generated, it is fed into the decoder. Backpropagation then considers ε as fixed. VAEs try to learn the distribution of the data by maximizing the marginal likelihood $p_\theta(x) = \int p_\theta(x \mid z)\, p(z)\, dz$ which requires integrating out a latent variable z. This integral is intractable in general so that the maximization is difficult to operationalize. Variational inference derives an auxiliary optimization problem to maximize the likelihood. Consider a tractable approximation $q_\phi(z \mid x)$ to the true posterior $p_\theta(z \mid x)$. One way to measure the degree of approximation is the Kullback-Leibler divergence

$$\begin{aligned}
\mathrm{KL}\big(q_\phi(z \mid x) \,\|\, p_\theta(z \mid x)\big) =& \mathbb{E}_{q_\phi}\big[\log q_\phi(z \mid x) - \log p_\theta(z \mid x)\big] \\
=& \mathbb{E}_{q_\phi}\big[\log q_\phi(z \mid x) - \log p_\theta(x \mid z) - \log p(z)\big] + \log p_\theta(x),
\end{aligned}$$

where the second equation uses Bayes' rule. Rearranging yields

$$\log p_\theta(x) = \underbrace{\mathbb{E}_{q_\phi}\big[\log p_\theta(x \mid z)\big] - \mathrm{KL}\big(q_\phi(z \mid x) \,\|\, p(z)\big)}_{=:\text{Evidence Lower Bound (ELBO)}} + \underbrace{\mathrm{KL}\big(q_\phi(z \mid x) \,\|\, p_\theta(z \mid x)\big)}_{\geq 0}.$$

Instead of maximizing the likelihood, variational inference maximizes the ELBO. This balances reconstruction accuracy and keeping the latent distribution close to the chosen reference distribution. See Blei,

Kucukelbir, and McAuliffe (2017) for a review of variational inference methods.

3.3.2 Generative Adversarial Networks Generative Adversarial Networks (GANs) pit two networks—a generator and a discriminator—against each other. To paraphrase Goodfellow, Bengio, and Courville (2016), the generator acts like a counterfeiter, while the discriminator tries to detect the counterfeits. If $x^1, \ldots, x^m$ are real observations and $z^1, \ldots, z^m$ are noise draws, a basic GAN solves

$$\min_{\theta_g} \max_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D(x^i, \theta_d) + \log\bigl(1 - D(G(z^i, \theta_g), \theta_d)\bigr) \right], \tag{18}$$

where $D(\cdot, \theta_d)$ is the discriminator and $G(\cdot, \theta_g)$ is the generator. At a high level, the discriminator learns where generated data differ from real data, and its gradients guide the generator in moving fake samples toward the data distribution. As noted by Arjovsky and Bottou (2017), this helps explain both why GANs can generate sharp images and why training can be unstable: If the discriminator becomes too strong relative to the generator, the generator may receive poor gradient information. Conversely, if the generator becomes too strong, it may produce samples concentrated on a small subset of modes of the data distribution that consistently fool the discriminator. This phenomenon, where diversity is lost, is known as "mode collapse". A large literature on architectural modifications to stabilize GAN training has emerged, see Jabbar, Li, and Omar (2021) for a review. Here, we highlight notable contributions in two areas: architectural adjustment and stabilization of training, and theoretical properties.

For stabilization of training, an important architectural development is the Wasserstein GAN (WGAN) of Arjovsky, Chintala, and Bottou (2017). A key issue with the original GAN objective is that the Jensen–Shannon divergence can be uninformative when the supports of the distributions are disjoint. WGAN addresses this by replacing it with the Wasserstein-1 distance between the data distribution $\mathbb{P}_d$ and the generated distribution $\mathbb{P}_g$, where $\mathbb{P}_g$ denotes the distribution of $G(z)$ with $z \sim p(z)$. By the Kantorovich–Rubinstein duality, this distance can be written as

$$W_1(\mathbb{P}_d, \mathbb{P}_g) = \sup_{f:\mathbb{R}^{\dim(x)} \to \mathbb{R} \,:\, |f(x) - f(x')| \le |x - x'|} \mathbb{E}_{x \sim \mathbb{P}_d}[f(x)] - \mathbb{E}_{x \sim \mathbb{P}_g}[f(x)].$$

The supremum is taken over all 1-Lipschitz functions f, which in practice are parameterized by a neural network (the "critic") with enforced Lipschitz constraints (a general Lipschitz bound K only rescales the objective). This leads to the WGAN objective

$$\min_{\theta_g} \max_{\theta_f \in \mathcal{F}_K} \mathbb{E}_{x \sim \mathbb{P}_d}[f(x; \theta_f)] - \mathbb{E}_{z \sim p(z)}[f(G(z, \theta_g); \theta_f)], \tag{19}$$

where $\mathcal{F}_K$ denotes the set of K-Lipschitz functions. Unlike the original GAN formulation, WGANs provide informative gradients even when the supports of $\mathbb{P}_d$ and $\mathbb{P}_g$ are disjoint, which improves training stability. Enforcing the Lipschitz constraint is not trivial. Arjovsky, Chintala, and Bottou (2017) use an ad-hoc weight-clipping approach with several disadvantages. Gulrajani et al. (2017) propose instead to penalize the gradient, which proves effective. Another challenge in training WGANs is the frequent occurrence of cycling behaviors instead of convergence. Since WGANs are a zero-sum game between generator and discriminator, Daskalakis et al. (2017) build on computational game theoretical results and show that Optimistic Mirror Descent (OMD) and Optimistic Adam avoid the cycling behavior, often reaching lower Wasserstein-distances between real and generated distribution.

From a theoretical perspective, Goodfellow et al. (2020) characterize the optimal discriminator between the pdf of the data p_d and the induced pdf of a particular generator G and p_g as $D^*_G(x) = p_d(x)/(p_d(x)+p_g(x))$. Plugging this optimal discriminator into equation (18) yields $-\log(4) + 2JS(p_d||p_g)$ where $JS(p_d||p_g)$ is the Jensen-Shannon divergence between p_d and p_g. This implies that the optimal generator satisfies $p_d(x) = p_g(x)$. Goodfellow et al. (2020) show that if the generator and discriminator networks have sufficient capacity (i.e., they are expressive enough to represent the optimal solutions), then we can analyze GAN training under the assumption that, at each step, the discriminator is trained to its optimum for the current generator. Under this assumption, the generator adjusts p_g to improve the objective

$$\mathbb{E}_{x \sim p_d}[\log(D^*_G(x))] + \mathbb{E}_{x \sim p_g}[\log(1 - D^*_G(x))].$$

Arora et al. (2017) show that when the size of generator and discriminator are not large enough, convergence fails: The generator may reach an objective arbitrarily close to optimal even though p_g is far from p_d for most of the support of p_d. They argue that due to the curse of dimensionality, it is very difficult to entertain

generators and discriminators of sufficient capacity to satisfy the conditions of Goodfellow et al. (2020). Arora, Risteski, and Zhang (2018) propose a support test to distinguish whether high objective values for the generator are due to approximate optimality of the generator or due to a restricted support of p_g. Liang (2021) studies the ability of (various versions of) GANs to approximate various target distributions (parametric and nonparametric). It is shown that the complexity of generator and discriminator should be balanced via generator-discriminator pair regularization. Liang (2021) derives novel oracle inequalities which allow one to derive the convergence rates of GANs in various contexts. Stephanovitch et al. (2024) study WGANs for one-dimensional z. The optimal WGAN for a given finite sample "walks" along the connected paths between sample points which minimize the sum of the squared Euclidean distances between the sample points as z is changed. This should remind the reader of our example of generating draws from uniform distributions at the introduction to Section 3.3. They also provide conditions under which WGANs approximate p_g as the number of observations diverges to infinity and characterize the rate of convergence.

Deb and Liang (2025) introduces a new generative modeling framework based on a discretized parabolic Monge–Ampère PDE, which arises as a continuous limit of the Sinkhorn algorithm in optimal transport. The method iteratively refines Brenier maps via a mirror gradient descent scheme, combining the sampling efficiency of one-step transport with the learnability of multi-step models. The authors provide no-regret convergence guarantees, including a novel evolution variational inequality linking geometry, transport cost, and regret, without requiring log-concavity of the target distribution. The framework bridges ideas from GANs and diffusion models which we cover next.

3.3.3 Diffusion Models The most intuitive description of a diffusion model is *learning to undo a tiny amount of noise*. These models were proposed by Sohl-Dickstein et al. (2015) and popularized by Ho, Jain, and Abbeel (2020). Starting from clean data $X_{t-\eta}$, the forward process adds a small Gaussian perturbation[11] to produce

$$X_t = \sqrt{1-\eta}\, X_{t-\eta} + \sqrt{2\eta}\, Z, \qquad Z \sim N(0, I).$$

[11] We use a particular discretization convention (with the diffusion coefficient normalized to one, in the Langevin style). Equivalent formulations lead to the same identities up to rescaling.

The key idea behind denoising diffusion models is that score estimation can be reduced to a supervised regression problem. In particular, the score function $\nabla \log p_t(x)$ can be expressed in terms of a conditional expectation. A version of Tweedie's formula gives

$$\nabla \log p_t(x) = \frac{\sqrt{1-\eta}\,\mathbb{E}[X_{t-\eta} \mid X_t = x] - x}{2\eta}.$$

Thus, if we can learn to predict the conditional expectation $\mathbb{E}[X_{t-\eta} \mid X_t]$—that is, to denoise X_t back to a slightly cleaner version—then we obtain an estimator of the score function. In this sense, score estimation is reduced to learning a conditional expectation. This perspective is closely related to empirical Bayes methods and has been explored in, e.g., Saremi and Hyvärinen (2019).

In practice, this leads to a simple supervised learning objective: given a noisy sample X_t and time t, train a model to predict either the cleaner signal $X_{t-\eta}$ or the added noise Z. These parameterizations are equivalent up to a linear transformation, and both yield the score through the identity above. In particular, predicting the noise Z determines $\mathbb{E}[X_{t-\eta} \mid X_t]$ and hence the score.

Repeating many such small denoising steps allows one to progressively transform pure noise into an approximate sample from the data distribution. In the continuous-time limit, this procedure admits both a stochastic reverse-time SDE formulation and an equivalent deterministic *probability flow ODE*, which evolves samples along the same marginal distributions. The ODE formulation can offer improved stability and has been the subject of recent theoretical analysis establishing fast convergence guarantees and connections to optimal transport; see, e.g., Chen et al. (2023); Liang, Dharmakeerthi, and Koriyama (2024). More broadly, a growing theoretical literature has analyzed when and why the diffuse-then-denoise paradigm is effective, including connections to score matching and denoising autoencoders, as in Vincent (2011); Alain and Bengio (2014), to empirical Bayes, following Saremi and Hyvärinen (2019), and to stochastic control and optimal transport formulations, as discussed in De Bortoli et al. (2021).

This perspective clarifies why diffusion models are trainable: the core task is standard supervised regression rather than direct estimation of $\nabla \log p_t$. Conditional versions work the same way, except that

the denoiser also receives the conditioning variable y (for example a text prompt or a class label). Diffusion models have been applied to super-resolution, text-to-image generation, text-to-video generation, and black-box optimization, see Ramesh et al. (2022); Rombach et al. (2022); Singer et al. (2022); Li et al. (2024); Yang et al. (2024). Fan, Gu, and Li (2025) establish convergence rates for estimation of densities admitting a factorizable low-dimensional nonparametric structure using score-based diffusion models.

3.4 Reinforcement Learning with Feedback

Reinforcement learning has played an increasingly important role in the training of LLMs, see Ouyang et al. (2022). One approach is to directly use human feedback on the quality of a prompt to fine-tune the parameters of the network, a method known as Reinforcement Learning with Human Feedback (RLHF). Another approach is to separately train a network to predict response quality, and then have the LLM respond to randomly generated prompts, with the quality of the responses evaluated by the separate network. However, this setup can create a circular problem.

Reinforcement learning with verifiable rewards addresses this issue by training an LLM on tasks where a ground truth is available, such as problems in mathematics, coding, or other STEM fields, and has been credited with improved reasoning ability, see Guo et al. (2025); Wen et al. (2025). Team et al. (2025) provide another example of using RL to improve response quality. Their setup encourages the LLM to generate "chain-of-thought" reasoning. Given a prompt x and a ground truth response y, the goal is to generate a chain of intermediate steps z_i that leads the model to the correct answer. The "truth" of the response can be evaluated in different ways; for example, if the desired output is code, it can be executed against a set of test cases. The search or planning process required to find the correct chain of thought is approximated by the LLM itself.

3.5 Supervised Fine-tuning

Estimating all parameters of a complex neural network from scratch using large datasets is difficult and expensive. Often, it is more efficient to *fine-tune* an existing neural network that has already been trained on a large dataset for the task at hand.

As a simple example, imagine that to approximate $\mathbb{E}[\text{income}|\text{education} = ed]$, a neural network $f(ed; \hat{\theta})$ has been estimated using observations from 1980–1985 by one research team. A second research team might be interested in approximating the same conditional expectation in a different period, 2002–2007. They expect that the first team has done an excellent job in estimating $\mathbb{E}[\text{income}|\text{education} = ed]$ for their period but anticipate that the conditional expectation has slightly changed over the years. Rather than using $\hat{\theta}$ without modification or training a neural network from scratch, the second team can start from $\hat{\theta}$ and use stochastic gradient descent with the 2002–2007 data and a very small learning rate to gently update the parameters. This approach is called fine-tuning and has proven extremely effective, primarily because it requires less expertise in training neural networks, less data, and less computational power, while also leveraging previous work.

Neural networks trained on large datasets are sometimes called *foundation models*, a term due to Bommasani et al. (2021). Fine-tuning is simple in principle: it often requires just a few steps of gradient descent. However, foundation models are typically very large, making it infeasible for an end-user to fine-tune them on a standard computer. A literature on *parameter-efficient fine-tuning* has emerged to address this problem; see, e.g., Xin et al. (2024); Ji et al. (2025) for recent reviews. To our knowledge, the theoretical properties of fine-tuning are not well-understood and can be expected to depend on 1) what properties can be expected of the training foundation model (i.e., did it converge to a global minimum?), 2) the "difference" in distribution between the data for training the foundation model and the data used for fine-tuning. See Springer et al. (2026) for some pitfalls of fine-tuning regarding safety.

3.6 Statistical Aspects

There are many open questions around the statistical/econometrics aspects of LLMs, as reviewed in Ji et al. (2025); Su (2025). Most of these revolve around trying to explain in mathematical terms why LLMs work as well as they seem to, and how to use statistical insights to further their development. For example, Bai et al. (2024) study the in-context learning properties of transformers. Lai, Lim, and Liu (2024) point out that with ReLU-activation, attention, masked attention, and cross-attention can be expressed as cubic

splines, opening links between LLMs and classical approximation theory. Other open questions pertain to 'Scaling Laws', 'Alignment', 'Fine-tuning' and 'Uncertainty Quantification'.

'Scaling Laws' describe how model performance behaves as a function of model size and sample size. 'Alignment' concerns itself with getting LLMs to produce output that is in accordance with some notion of social good. This might also include questions about how to induce some notion of 'fairness' into LLM output while minimally sacrificing performance.

'Uncertainty Quantification' describes how uncertain we should be about the correctness of model output, and in which sense we should take this uncertainty. In the neural network community, there are two approaches to inference, called uncertainty quantification: Bayesian Neural Networks (BNNs), as developed by Welling and Teh (2011); Graves (2011); Hernández-Lobato and Adams (2015); Blundell et al. (2015); Gal and Ghahramani (2016); Ritter, Botev, and Barber (2018), and ensemble approaches proposed by Lakshminarayanan, Pritzel, and Blundell (2017); see Abdar et al. (2021) for a review and Loaiza-Ganem, Villecroze, and Wang (2025) for an interpretation of ensemble approaches as empirical Bayes. While some work has been done in these areas, many questions remain open, and the field is evolving rapidly. Some ideas for trying to answer these kinds of questions can be found in Berner et al. (2022) and Simon et al. (2026).

4 Applications in Economics

In this section, we distinguish five ways of using neural nets for economic analysis. First, they can be used to curate data by making data more accessible, for example by creating text from document scans; see Section 4.1. Second, neural nets can be used to generate interesting hypotheses or anomalies that are incompatible with an economic model; see Section 4.2. Third, they can be used to build foundation models for economics, for which we provide examples in Section 4.3. Fourth, they can be used for reduced-form analysis, e.g., the estimation of a treatment effect that is confounded in a high-dimensional, unstructured way, such as by text, images, or social networks; see Section 4.4. Finally, neural nets can be used in the computation, simulation, and estimation of complex, high-dimensional structural models; see Section 4.5.

LLM-assisted productivity Dowling and Lucey (2023); Korinek (2023, 2024a,b) discuss how LLMs (a specific type of neural nets) can increase economists' research productivity, for example in writing, coding, and deriving formulas. Cowen and Tabarrok (2023) discuss the effect of teaching and learning economics using LLMs. Nosratabadi et al. (2020) provide summary statistics on the rapid adoption of neural nets in economic research.

4.1 Data Curation

Einav and Levin (2014); Varian (2014) have discussed the advent of "big data" in economic research.

4.1.1 Document Image Analysis As noted by Gentzkow, Kelly, and Taddy (2019); Ash and Hansen (2023), text is an increasingly important source of data for economists. While today's texts are often digitized and thus readily available for scientific analysis, this is not the case for documents written in the past. One key obstacle for researchers in economic history is the digitization of documents. Given an image of a document (a scan), how can the text be recovered? One has to be able to find and correctly classify the individual characters (optical character recognition, OCR), check whether two adjacent characters are the same word or not, check whether two adjacent words belong in the same text or not: The layout of a newspaper may involve tables, columns, articles that start on one page and continue on another etc. As shown by Dell et al. (2024), digitization that does not incorporate this layout is not able to produce coherent articles in newspapers. As a result, only text analysis approaches that do not rely on the order of words can be used, limiting research opportunities. The literature on these issues is relatively large and reviewed in Dell (2024). Conceptually, OCR and layout parsing are a classical classification problem so that CNNs and visual transformers are used. EffOCR, one of the leading packages, comes with a zoo of models including CNNs and visual transformers, see Bryan et al. (2023); Shen et al. (2021).

4.1.2 Record Linkage To study long-term effects of policies or migration, it is helpful to "follow" individuals over time. Often, data is collected in snapshots. For example, the US Census has been taken every 10 years since 1790. A critical step in many studies, going back to Dunn (1946), is to link records, e.g. to say if a person appearing in the Census in 1910 is the same as a person in the Census in

1940.[12] Abramitzky et al. (2021) provides a review of the methods for record linkage and some applications in economics where approximate string matching methods are the most widely-used methods to link records.

One of the first applications of neural networks to record linkage appears to be Ebraheem et al. (2018).[13] They follow a process proposed by Fellegi and Sunter (1969) consisting of first labeling pairs of observations as match/no match, then learning rules (either by hand or ML) to predict whether a pair is a match. Next, a blocking step rules out as many pairs which cannot be a match as possible to reduce the number of possible pairs, see Quass and Starkey (2003); Christen (2011) for reviews. Finally, for the pairs which have not been excluded, the learned rules or ML algorithms are applied to determine which pairs are matches. They propose to learn embeddings for each observation's attributes and compute a measure of similarity of these attributes. A recurrent neural net with long short-term memory units is then used to predict match/no match using labeled data. The resulting algorithm, DeepER, is reported to outperform existing approaches. Mudgal et al. (2018) consider variations of DeepER, including with attention architectures. See also Sutskever (2014); Cho et al. (2014).

Using embeddings for record linkage holds the promise of using semantic information rather than just string matching. However, it has been argued that the training data is too small and that the text used for linking is often too short to benefit from updating embeddings to contextual information, e.g., in names, see Binette and Steorts (2022). In certain cases, large amounts of training data have been made available, such as the product dataset of Primpeli, Peeters, and Bizer (2019). Further progress has been made by using fine-tuning rather than training neural nets from scratch, see Section 3.5.

Peeters and Bizer (2021) propose to fine-tune the BERT model of Devlin et al. (2019) to predict ISBN numbers and match books. Li et al. (2023) propose to use pre-trained transformers for record linkage such as BERT and variants. Their approach begins by ruling out pairs of entries that are unlikely to be matches using domain-specific methods. For example, when matching employer records they discard pairs

[12] Social Security numbers started being issued in 1936.

[13] Wilson (2011) reports successes from using neural nets for record linkage. But the neural net in question has no hidden layer so that it really is a logistic regression although it is trained as a neural net Rumelhart, Hinton, and Williams (1986).

that do not have the exact same zip code. However, zip codes are not available for 60% of the employers in the data so that they also add the 20 pairs with the highest term-frequency/inverse document frequency cosine similarity[14] of name and address strings for each observation on both datasets. For a subset of possible pairs, they manually create labels on whether the pair is a match or not. Finally, they fine-tune the transformer to predict this label after adding a soft-max output layer to the BERT architecture. See also Peeters and Bizer (2023).

Arguing that the existing tools are difficult to use for social scientists without extensive coding experience, Arora and Dell (2023) develop LinkTransformer, a library to use transformers for record linkage that can integrate models from Hugging Face or OpenAI. They frame record linkage as finding the nearest neighbor of one observation's embedding in another dataset's embeddings. They use the FAISS backend to scale the searches across pairs with GPUs, see Johnson, Douze, and Jégou (2019), and provide other functionalities such as de-duplication. Boustan and Abramitzky (2026) review recent progress in record linkage and empirical work using these methods. One important development is linking of women across censuses with marriage transitions, e.g. by using marriage certificates as in Eriksson et al. (2026).

4.1.3 Measuring Low-Dimensional Features in High-Dimensional Data Neural nets have been used to extract meaningful variables from raw text, images, videos, or other data sources. The measured variables are then interesting in their own right or used in subsequent econometric analysis, often regressions. This type of application is likely the most frequent application of neural nets in economic research so far. As a result, we can only provide a few examples for each type of data.

Inference for the results of such a procedure must account for measurement error. It arises here naturally as the estimation error of the low-dimensional variable(s). While Neyman-orthogonalization or double machine learning is used in some of the papers surveyed, this requires sufficiently fast convergence

[14]Term frequency is the frequency of a specific word in a document. Document frequency counts how many documents contain the same specific word. In other words, the ratio of term frequency and document frequency measures how often a word appears in one document relative to a corpus of other documents. It was proposed in Spärck Jones (1972). Cosine similarity is simply the dot product of the normalized tf-idf vectors of two documents to measure similarity based on a notion of vector space similarity attributed to Salton, Wong, and Yang (1975), although this attribution may be wrong Dubin (2004).

of the measurement error to zero. In many cases, the required convergence rates are not established so that inference is an open problem.

Images Satellite data contain information on economic activity, see, e.g., Croft (1978); Sutton and Costanza (2002); Ghosh et al. (2009, 2010); Chen and Nordhaus (2011); Henderson, Storeygard, and Weil (2012), and Donaldson and Storeygard (2016) for an overview. Satellite data comes in the form of images, a data-type for which several neural net architectures were developed. As a result, there are numerous studies which have applied neural nets to leverage satellite imagery to study economic questions or measure relevant variables, e.g., Yeh et al. (2020); Huang, Hsiang, and Gonzalez-Navarro (2021), see Hall et al. (2023) for a review.

In an early application, Jean et al. (2016) estimate consumption and wealth at the local level in Nigeria, Tanzania, Uganda, Malawi, and Rwanda by fine-tuning AlexNet, due to Krizhevsky, Sutskever, and Hinton (2012), to predict nightlights from satellite images. In a second step, they use the selected features to predict local consumption and wealth on the subsample for which such data are available using Ridge regression. Using cross-validation as estimate of out-of-sample fit, they report their CNNs to reach an R^2 of 41-56% for consumption and 55-75% for wealth, depending on the country on which it was trained. For consumption, out of sample R^2 ranges from 25% (trained on Nigerian data and evaluated on Malawian data) to 48% (trained on Malawian data and evaluated on Tanzanian data).

Adukia et al. (2023) study the gender, skin color, race, and age of characters appearing in children's books using both image and text data. Central applications for this study are Google AutoML Vision, now called VertexAI, a proprietary software advertised for applying machine learning methods without code, and Google Vision OCR. Adukia et al. (2023) use these tools to detect faces in images and predict gender, age and race of a person as well as to detect text in scans of children's books. As noted in a companion paper Szasz et al. (2022), the proprietary face detection tool did not perform well as it was trained on photographs of humans, while faces in children's books are often illustrated. The authors therefore hand-collect a dataset of images with illustrated faces and bounding boxes to fine-tune it to detect illustrated faces. In a hold-out dataset, their best-performing model detects 76.8% of the faces, and 93.4% of detected faces are actually

faces.

To predict gender, race, and age of the detected faces, Szasz et al. (2022) collect a second fine-tuning dataset in which three different graders classified the faces into categories, e.g., male/female for gender. The three labelers agreed on a specific gender (male, female, unsure) in 75% of cases, on race (Asian, Black, Latinx, White, other, and unsure) in 60% of cases and on an age category (infant, child, teenager, adult, senior, unsure) in 52% of cases. For training, they used the modal label. Their preferred model correctly classifies faces in 53.6% for gender, 54.3% for race, and 59.3% for age. Adukia et al. (2023) use this model to document trends and stylized facts on the prevalence of characters with various demographics in influential children's books, reporting for example that the prevalence of Black and Latin American people is lower in children's books than in the US population.

Text The rise of text as a data source in economic research has been noted, reviewed, and stimulated by Gentzkow, Kelly, and Taddy (2019). The authors discuss ways of representing text (e.g. an email) as numerical vectors, use this numerical representation to predict unknown outcomes (e.g. an indicator for whether it's a spam email), and use the predicted outcomes in subsequent analysis. They also review ML methods in light of the high-dimensional nature of text data, including early NN architectures. Ash and Hansen (2023) review more recent tools for natural language processing, specifically considering four objectives: (1) quantifying document similarity, (2) measuring economic concepts contained in raw text, (3) measuring relations between concepts, and (4) relating text to quantitative metadata. They also highlight the challenges of validating the output of algorithms.

Veitch, Sridhar, and Blei (2020) study the causal effect of the presence of a theorem in a paper on success in publication and the causal effect of the gender label of the author on the popularity of a post on social media. Here, confounding variables are the text. In the case of a paper, the technicality of a paper may affect both whether there is a theorem and whether the paper is published. To estimate the ATT with a propensity score approach, they fine-tune BERT to produce three objects of interest: a text embedding (of the paper's abstract), the probability of treatment conditional on the text embedding and the conditional expectation of the outcome conditional on treatment and text's embedding. This allows them to construct

estimates of the propensity scores where one uses the embedding as control.

Chava, Du, and Paradkar (2020) study the effect of mentions of emerging technologies in earnings calls on stock market reactions. To measure references to emerging technology, they create a time-varying dictionary of words corresponding to technologies emerging at a particular point of time. To do so, they use Gartner, the MIT Technology Review and Wikipedia to manually identify words corresponding to emerging technology. Since these words are often technical and described using other terminology in earnings calls, they look for potential synonyms manually and then measure similarity between the words in their dictionary and the potential synonyms using cosine similarity of embeddings produced by SentenceBert, proposed by Reimers and Gurevych (2019). They then regress abnormal cumulative return in the three-day window around the earnings call on an indicator of whether any emerging technology was mentioned and controls which include some features of the earnings call transcript. They find that mention of emerging technologies in earnings calls elicits a positive immediate stock market reaction even though they provide evidence that investors cannot distinguish mentions of emerging technologies corresponding to the firm's innovation from mentions of emerging technologies which are just "buzzwords". Chava, Du, and Malakar (2021) use a similar approach to test whether mention of environmental issues in earnings conference calls causes higher pollution abatement and green patents and whether mention of social issues causes higher employee ratings. They find positive evidence of both.

To measure "remote work across jobs, companies, and space", Hansen et al. (2023) classify job postings on whether or not they allow for remote work using an encoder-only transformer. As baseline model, they use DistilBERT, developed by Sanh et al. (2019), a "distilled" version of BERT in the spirit of Ba and Caruana (2014) that has fewer parameters and therefore is easier to train, store, and use.

Since job postings are too long for DistilBERT, Hansen et al. (2023) split them into "sequences": title, bullet points after the title, and then each paragraph form a sequence. Each sequence is classified individually. If any sequence is classified as "allows remote work", then the entire job posting is classified as such. Hansen et al. (2023) fine-tune the parameters of DistilBERT in two ways: First, they update DistilBERT's parameters to predict masked words in job postings. Intuitively, this allows the attention

parameters that were trained on general English to adapt to job postings. Second, they obtain classification data for 10,000 "sequences" from human classification and further update the parameters to solve the supervised learning problem of classifying whether a job posting allows remote work or not. Hansen et al. (2023) run these fine-tuning steps on the Google Cloud platform for a total of 51 hours for $1,500, suggesting that a similar approach is viable even with limited research budgets.

Their fine-tuned DistilBERT model achieves an error rate of 2%, substantially outperforming dictionary methods, logit regression both with and without negation and even GPT-3, the latter exhibiting an error rate of 6% despite its much larger size. This illustrates the power of fine-tuning compared to increasing the size of a model. Fine-tuning based only on the supervised learning problem achieves an error rate of 3%, and the prediction of masked words in job postings allows them to further cut the error rate to 2%.

To measure the effect of the gender of scientists on the commercialization of their studies, Koffi and Marx (2023) would like to compare papers of similar commercial appeal but with authors of different genders. Specifically, they consider the gender of the last-mentioned author, typically the head of the laboratory. To control for the commercialization potential of an article, they use academic citations, patent citations, and estimates of the commercial relevance of a paper. They leverage a dataset of 70 million papers in the Microsoft Academic Graph, using their title and the abstract. The parameters of SBERT, proposed by Reimers and Gurevych (2019), are fine-tuned on 2.5 million research articles for each discipline.

Acikalin et al. (2022) evaluate the economic effects of a Supreme Court decision that reduced patent protection in certain cases Supreme Court of the United States (2014). To quantify the probability that one of the 3.8 million patents in the 20 years before the Supreme Court decision would be rejected on the basis of the decision, Acikalin et al. (2022) fine-tune several language models to this task. The best-performing one is Longformer, developed by Beltagy, Peters, and Cohan (2020), a pre-trained language model specifically designed for long prompts: Patents have 9,173 words on average which, after tokenization, is too long for most pre-trained models. As Longformer is trained to process up to 4,096 tokens or around 3,200 words, Acikalin et al. (2022) summarize the patents using TextRank, proposed by Mihalcea and Tarau (2004); Upasani et al. (2020). They then use patent decisions following the Supreme Court decision

to fine-tune the model to predict whether a patent was rejected based on the Supreme Court decision, using that the US Patent and Trademark Office records the legal reason for rejection of a patent.

Avivi (2024) examines the gender neutrality of patent examiners in the US where in 2016, 12% of inventors granted a patent were women USPTO (2019). To control for the unobserved quality of patents, Avivi uses a version of the BERT model of Devlin et al. (2019) trained exclusively on over 100 million patents by Google researchers Srebrovic and Yonamine (2020). Avivi considers using the values of the last, second-to-last, and third-to-last hidden layers as embeddings, distinguishing the main text and the claims of a patent separately.[15] This gives 6 embedding vectors, each of length 1024. Since each of these 6 vectors sums to one, Avivi removes one element to avoid collinearity. For any subset of these 6 vectors, she computes the adjusted R^2 of two regressions: First, a regression of initial allowance on the concatenated embedding vectors, and secondly, a regression of a dummy for the presence of any female inventor on the concatenated embedding vectors. The second-to-last layers for main text and claims achieve the highest adjusted R^2 in both regressions. Avivi then assumes that conditional on this embedding vector, the unobserved quality of the patent is independent from an indicator for initial allowance and an indicator for the presence of any female inventor. Avivi estimates the effect of gender on initial allowance controlling for patent quality by regressing initial allowance on a female indicator using the embeddings as controls. The results indicate that there is no statistically significant effect of gender, although this masks heterogeneity across time and examiners.

Audio Sometimes, economic interactions are human interactions involving communication. Communication is remarkably complex. For example, Von Thun (1981) postulates that every message has four facets: factual information, self-revelation, appeal, and relationship. In addition, human communication has various layers. Mehrabian (1971) concluded from the early experiments of Mehrabian and Wiener (1967); Mehrabian and Ferris (1967) that of the informational content of a message, words convey 7%, body language (facial expressions, gestures, posture etc.) 55%, and the tone of voice 38% of the content. These studies and the conclusions drawn from them have been criticized, see, for example, Lapakko (2007). Still, they can serve as motivation for economists to quantify the importance of body language and tone in

[15]Given a context window of 512 tokens, Avivi splits the text into paragraphs and then averages the embeddings of the paragraphs.

economic settings.

Gorodnichenko, Pham, and Talavera (2023) study the effect of the tone of voice by the Chair of the Federal Reserve in Q&A sessions of Federal Open Market Committee press conferences. For this purpose, they have to quantify the tone of voice and identify its effect separately from the economic content articulated in the words of the answer.

To measure the tone of voice, they extract various features of the audio of each answer that have been proposed in the literature, see, e.g., Pan, Shen, and Shen (2012); Gao et al. (2017); Likitha et al. (2017); Bhavan et al. (2019), using the Python package Librosa. Specifically, they extract 128 mel spectrogram frequencies measuring the loudness of frequencies at each point of time, and the distribution of chroma features using the equal-tempered scale over time[16]. To train a neural net to predict emotions based on these features, Gorodnichenko, Pham, and Talavera (2023) use two datasets from Dupuis and Pichora-Fuller (2010); Livingstone and Russo (2018) in which different actors have been asked to read the same text with various emotions (happy, sad, angry, fearful, pleasantly surprised, disgusted, and neutral). Gorodnichenko, Pham, and Talavera (2023) then train a neural net which takes the 180 vocal features as input, two hidden layers with 200 neurons each, with linear activation functions[17] and a softmax output layer to predict five emotions (happy, pleasantly surprised, neutral, sad, and angry), arguing that these are the relevant emotions of Fed Chairs. To regularize, the authors use Bernoulli dropout with $p = 0.3$. The network is trained via Adam for 2000 epochs using a batch size of 64 to minimize cross-entropy loss and correctly predicts 84% of emotions, slightly higher for angry and sad (both 87%) and somewhat lower for neutral (74%).[18]

To measure the economic content articulated in the words of the answer, Gorodnichenko, Pham, and Talavera (2023) use BERT to classify the text as either dovish, hawkish, or neutral. To fine-tune BERT for this, they manually collected a training dataset of FOMC statements in the time before their sample

[16] They also use 40 mel-frequency cepstral coefficients (MFCCs) which are specific functions of the 128 mel frequency spectrogram.

[17] Absent regularization, this leads to an overparametrized linear function.

[18] A similar analysis is done in Shah, Paturi, and Chava (2023).

period and had several RAs label each text independently on a numerical scale. The authors then classified each text into one of the three categories by comparing the average of the RAs' labels to cutoff points. To fine-tune BERT, they feed the 512×768 output of the last hidden layer in BERT via hyperbolic tangent activation to a bidirectional LSTM layer with 512 units, a dropout rate of 10% and a recurrent dropout rate of 10%. This layer and all following layers are followed by a 10% rate dropout layer. The second hidden layer is a global average pooling 1D layer, see Lin, Chen, and Yan (2013), to transform the 2-dimensional into one-dimensional data. The third hidden layer is a fully-connected layer with 512 neurons and ReLU activation, followed by a fourth hidden layer consisting of 128 neurons and ReLU activation after which there is no more dropout. A softmax output layer for the three sentiments (hawkish, dovish, neutral) completes the network. Both BERT and RoBERTa reach an accuracy of about 80% on average, although hawkish answers are correctly classified slightly more often than dovish and neutral answers.

Following Jordà (2005), Gorodnichenko, Pham, and Talavera (2023) regress financial indicators on voice tone, text sentiment, indicators for whether there was a news conference and controls inspired by the macroeconomics literature, see, e.g., Wu and Xia (2016); Swanson (2021). Although exogeneity of voice tone and text sentiment to financial indicators is not clear in the presence of expectations on the behavior of the Fed, the authors interpret the coefficients in this regression causally and conclude, for example, that a one standard deviation increase in positivity of the voice tone raises the S&P 500 returns by 75 basis points, an economically significant effect according to the authors.

Several other papers have taken similar approaches to different settings. For example, Handlan and Sheng (2023) use audio recordings of the NBER Summer Institute to predict the gender, age, and tone (same categories as above) of presenters, discussants and audience members asking questions or making comments. They emphasize the importance of training separate networks to predict age and gender for male and female speakers due to the different pitches of male and female speakers. Among other results, Handlan and Sheng (2023) report that female economists are less likely to be spoken to in a positive tone. Liem et al. (2018) provide a multi-disciplinary review of applications to job interviews.

Video Hu and Ma (2021) study the impact of visual, verbal and vocal communication in startup pitches to investors. To quantify these features, they use proprietary neural net architectures, including Face++, Microsoft Azure, and Google Cloud. They then aggregate these measurements in a single number per pitch, called "Pitch Factor," which they interpret as "overall level of positivity—e.g., happiness, passion, warmth, enthusiasm—in a pitch". Assuming selection-on-observables with controls for founders' education, employment background, and startup experience, they find that pitches with high "Pitch Factor" are more likely to obtain funding. However, conditional on obtaining funding, startups with higher "Pitch Factor" perform worse.

Curti and Kazinnik (2023) study the impact of (negative) facial expressions by the Fed Chair in FOMC press conference videos on minute-level financial assets. They measure facial expressions using the Microsoft Azure Cognitive Services Emotion API. To control for the content of what was said, the authors use the BERT model of Devlin et al. (2019) to quantify the sentiment of the statements. The authors manually timestamped the text from a transcript with the precise moment in the video in which it was said. They find that negative facial expressions, conditional on the sentiment of the statement, have negative effects on the financial assets they considered. Alexopoulos et al. (2024) follow a similar approach in Fed Chairs' communication during congressional testimonies using C-SPAN videos and transcripts.

4.2 Hypotheses and Anomalies

4.2.1 Hypotheses The process of generating a scientific hypothesis is much less formalized than the process of testing it.

Fudenberg and Liang (2019) is an example of a paper which uses nonparametric prediction together with economic intuition to generate a hypothesis. They consider the problem of predicting initial play in matrix games. By studying games where a nonparametric prediction algorithm, bagged decision trees, outperforms existing economic algorithms, they identify a hypothesis: A one-parameter extension of level-1 play, as studied by Stahl II and Wilson (1994); Nagel (1995), i.e. the use of a concave utility function when computing the level-1 strategy, increases the predictive performance. To understand the generalization of

this rule, Fudenberg and Liang (2019) generated matrix games randomly and collected data on the choices to predict whether in a given matrix game, the extension predicts the modal initial strategy. This prediction algorithm was then used to classify whether in additional simulated games, the extended algorithm would be likely to incorrectly predict the initial move. In this way, the authors collected 200 games in which the extended algorithm was expected to perform poorly. For these 200 games, human initial choice data confirmed that the extended algorithm performed poorly and is outperformed by a decision tree. While the authors found this tree hard to interpret, a restricted version with just two decision nodes returns predictions that could be interpreted by the authors as the Pareto dominant Nash equilibrium (PDNE). The authors then formulated a hybrid model which first separately estimates the probability that the extended model or the PDNE predicts the initial move correctly and then uses the prediction of the model with the higher success probability. This hybrid model performed better than both PDNE and the extended model. To the best of our knowledge, this paper is the first to document the formulation of the algorithmic generation of a hypothesis.

Peterson et al. (2021) consider choices under uncertainty, specifically the choice between pairs of lotteries with known distribution. Starting from the expected utility framework, they successively add non-parametric utility functions, perceived probabilities, and context dependence. Context dependence means that "expected utility" of one lottery in a choice problem may depend on the characteristics of the other. In each case, they propose a neural net architecture that, besides assuming differentiability of choice probabilities with respect to outcomes and probabilities, only incorporates the economic restrictions, e.g., correctly perceived probabilities, independence of choice and an unrestricted utility function. The best performing model is the most general one, i.e. featuring context dependence, subjective probabilities and utility functions. To produce a model that is interpretable, Peterson et al. (2021), echoing Fudenberg and Liang (2019), propose a "mixture of theories" which learns to apply one of two utility functions and one of two subjective probabilities, depending on the choice problem. This mixture model performs almost as well as the most general model despite its comparable simplicity and is interpretable: the functional forms of the two functions are similar to models of expected utility as studied by Von Neumann and Morgenstern (1944) and prospect theory as developed by Kahneman and Tversky (1979); Tversky and Kahneman (1992). Because the mixture model explicitly decides between the two utility and the two subjective probability

functions, this can be used to hypothesize about human behavior.

Ludwig and Mullainathan (2022) propose to formalize the process of generating scientific hypotheses. Specifically, following Dobbie and Yang (2021), they consider pretrial release/jail decisions which, by law, are to be made on a prediction of risk of flight and re-offending before the trial. To generate hypotheses about what matters for a judge's decisions on whom to release and whom to jail, they estimate a model of the judge by training a neural net to predict the judge's decision based on a defendant's alleged crime, demographics and mug shot. Upon realizing that the mug shot is highly predictive of the judge's decision, even after controlling for demographics and facial features considered in the psychology literature, the authors propose a way to generate an interpretable hypothesis for the judge's behavior. The idea is to generate two artificial mug shots that are very similar but differ in their probability of being released and then ask subjects to articulate the difference between these faces. The key difficulty is to generate two faces that are similar (i.e. maintain the defendant's demographics) and differ mostly by their probability of being released. A naive approach would be to change the pixels of the image in the direction of steepest ascent/descent in the probability of being released. However, this often results in "mug shots" that are not faces because the gradient is not constrained to faces. Instead, the authors propose to use GANs to generate artificial mug shots, see Section 3.3.2. A first mug shot is generated from some point in the GAN's latent space. Comparison mug shots are then generated by sampling "neighbors" of the latent variable of the first mug shot. These have to be verified, either manually or by a separate algorithm, to be sufficiently similar to the original mug shot. If the predicted probability of being released differs, the pair of mug shots, together with many others of the kind, can be handed to human subjects. The subjects are then told that one artificial mug shot has a higher probability of being released than another and asked to articulate why that may be the case.

4.2.2 Anomalies A natural counterpart to the generation of scientific hypotheses considered in Section 4.2.1 is the generation of anomalies, i.e. examples where scientific hypotheses fail. As argued by Kuhn (1962), scientific progress, at least in natural sciences, is often motivated by anomalies that the state-of-the-art theory cannot explain. Although economic models are typically designed to answer a specific question rather than offer a universal theory of the world, some examples of such anomalies exist even in economics,

see, e.g., Allais (1953); Kahneman, Knetsch, and Thaler (1991). Anomalies for an existing theory (such as the axioms of von Neumann and Morgenstern) had to be generated based on the intuition of researchers. In Fudenberg and Liang (2019) and Peterson et al. (2021), generating anomalies is part of formulating a hypothesis, see Section 4.2.1. Mullainathan and Rambachan (2024) propose an automated method of finding such examples. The idea is to cast this in an adversarial setting as in Goodfellow et al. (2020): Given an example (x, y), the discriminator seeks to match the best theory-compatible prediction with the example, i.e. a function f that is compatible with the theory such that $f(x) = y$. A generator searches for an example that maximizes the difference between the best prediction of the theory and the outcome of the example. To bring this idea to data, Mullainathan and Rambachan (2024) first use neural nets to estimate a map f^* from x to y *nonparametrically*, i.e. without any restrictions imposed by the theory. Then, they solve the following adversarial game:

$$\max_{\text{example } x} \min_{f \text{ compatible with theory}} L(f^*(x), f(x)),$$

where L is some loss function. In an application to data on choices between lotteries from Peterson et al. (2021), Mullainathan and Rambachan (2024) recreate anomalies known to the literature such as the dominated consequence effect and find new anomalies that they call the "reverse dominated consequence effect" and the "strict dominance effect". These effects could be due to noise in the estimation of f^*, so Mullainathan and Rambachan (2024) put them to the test in experiments and confirm that they are not due to noise in f^* but features of human decision-making.

4.3 Foundation Model: Job Sequences

As discussed in Section 3.5, foundation models are trained on large data sets using self-supervised learning and fine-tuned to perform downstream tasks. In economics, one setting in which such data sets are available is resumes, e.g., from LinkedIn Li et al. (2017) or Zippia Vafa et al. (2024). This motivates the estimation of foundation models to predict job sequences based on resumes, i.e. to predict an individual's next job given personal characteristics such as education, skill set, location and previous jobs, a long-standing problem in economics going back to Boskin (1974).

To the best of our knowledge, Li et al. (2017) were the first to train a foundation model in economics. Their dataset consists of more than 1M individuals on LinkedIn from more than 100 locations, 1K education institutions, with more than 10K different skills, working for more than 100K companies with more than 10K titles in more than 10M positions (company/title pairs). They use the long short-term memory (LSTM) architecture of Hochreiter and Schmidhuber (1997), see also Greff et al. (2016), a leading architecture for sequential data before the introduction of transformers by Vaswani et al. (2017), see Section 3.2.3. As we have not presented the details of LSTMs in this review, we will describe the key contributions of Li et al. (2017) in a way that invites the reader to imagine that a transformer was used. The authors use an encoder-decoder architecture where the encoder takes an individual's location, highest education institution and skill set as input to produce an embedding. To reduce the complexity of skill sets which can be of varying size, etc., Li et al. (2017) pool individual embeddings using a componentwise maximum of each skill's embedding vector. The encoder is a single layer using a tanh activation. The decoder takes as inputs the embedding as well as the jobs (company and job title), both predicted and realized. Company and job title are predicted separately using softmax output layers, following Jean et al. (2014), where the softmax is taken over a random subset of 50 options to reduce computational cost. The network is trained to maximize the likelihood of job sequences using various conditional independence assumptions to decompose the likelihood of a job sequence into the product of likelihoods of each individual job. Unfortunately, the model is not evaluated quantitatively in an economic application although a qualitative evaluation, such as sample job paths, may be considered encouraging.

In related work, Dave et al. (2018) propose three networks to recommend jobs to recruiters and workers. One of these is a job transition network trained on tokenized job transitions rather than full job histories. Zhang et al. (2019) propose to adapt the Word2Vec embedding algorithm Mikolov et al. (2013a,b), see also Ash and Hansen (2023), to compare the expertise of two workers in different companies with different titles. One element of their analysis is a job-graph that measures the similarity of jobs based on the probability of a switch from one job to another.

Vafa et al. (2024) propose a two-stage model of sequences over jobs: In the first stage, the worker decides whether to switch jobs or not. If the worker chooses to switch jobs, the worker decides in the

second stage which job to choose. A Markov assumption is imposed that the next chosen job only depends on the worker's current job and characteristics (including the time in the worker's career). The architecture is comparable to Li et al. (2017) except that transformers (see Section 3.2.3) were used instead of LSTMs, that two outcomes are modeled for each period (whether the worker changes jobs and if so, which job is switched into) and that the model allows for time-changing individual characteristics. The objective is a log-likelihood where the probability of job j is p if j is the previous job and $(1-p)\sigma$ for all other jobs, where σ is a softmax output layer. Vafa et al. (2024) evaluate the model by applying it to predict wages and so estimate the adjusted gender wage gap. Athey et al. (2024) show that the foundation model of Vafa et al. (2024) is outperformed by a fine-tuned foundation model, Llama-2, due to Touvron et al. (2023). To achieve this superior performance, the fine-tuning of Llama-2 is critical.

To fine-tune, Athey et al. (2024) propose two approaches. The first approach consists of fine-tuning Llama-2 to predict next jobs after prompting a worker's career history. The second approach involves extraction of embeddings from pre-trained and fine-tuned Llama-2 and using these embeddings as inputs to a multinomial logit classifier. One advantage of the second approach is that the new classifier only considers job titles that appear in the data rather than predicting "made-up" job titles which never appear in the data. The comparison between the model of Vafa et al. (2024) and the fine-tuned Llama-2 is not entirely fair: Llama-2 tokenizes words so that it can pick up semantic meaning in job titles, information which is not used in the model of Vafa et al. (2024). However, this necessitates a model of language reflected in the size of the models. While the model estimated in Vafa et al. (2024) has 5.6M parameters, Llama-2 has 7B, 13B or 70B parameters, depending on the version. The comparison is not entirely driven by model size: Athey et al. (2024) find that the Llama-2 (13B), fine-tuned with their second approach outperforms Llama-2 (70B), fine-tuned with their second approach. Both are outperformed by Llama-2 (70B), fine-tuned with the first approach, although the difference in performance between the best-performing fine-tuning approaches is not statistically significant when uncertainty is quantified using test-set bootstrapping.

4.4 Reduced-Form Inference

4.4.1 Experiments Since human subjects in experiments are expensive and their treatment must comply with ethical standards, the potential of AI agents (LLMs which have been given preferences, budgets, instructions on possible actions) to replace human subjects has been explored.

Aher, Arriaga, and Kalai (2023) propose a test of whether LLMs replicate the *distribution* of human behaviors in economic, psycholinguistic, and social psychological experiments. They call this test "Turing Experiment", as opposed to the Turing Test/Turing's Imitation Game proposed by Turing (1950). In the "Turing Experiment", the computer does not try to impersonate a single human but rather to predict the distribution of human behaviors in experiments. Aher, Arriaga, and Kalai (2023) emphasize that the TE should be zero-shot, i.e., that it should not be fine-tuned to existing experiments to avoid the AI agent regurgitating the data used for fine-tuning. In famous experiments such as the dictatorship game, the model may have been trained on text describing the experiment and results, limiting credibility. For newer or more niche experiments, the zero-shot assumption may be easier to justify. Aher, Arriaga, and Kalai (2023) find that in three out of four tested games, including the ultimatum game, recent AI models were able to replicate human behavior. Park et al. (2023) explore populating virtual cities with AI agents to study human behavior. Horton (2023a) considers recent AI agents as possible "homo silicus", computational models of humans which are useful for experimental economics in a variety of ways: Since they have been trained on large text corpora, AI agents incorporate some latent social norms, e.g. on fairness. Across a range of classical experiments, such as those of Samuelson and Zeckhauser (1988); Kahneman, Knetsch, and Thaler (1986); Charness and Rabin (2002), as well as a hiring scenario to study minimum wages from Horton (2023b), Horton finds that recent AI agents act similarly to humans and suggests using AI agents for pilots of real experiments. To measure the "knowledge" used by an LLM for predicting human behavior in a particular context, Gao, Han, and Liang (2026) propose the "equivalent sample size": The smallest number of observations of human behavior in the given context which allows an economic model or a nonparametric model to match the prediction accuracy of the LLM.

4.4.2 Surveys Due to the cost of surveys and the impossibility of surveying respondents in the past or in the future, several papers consider using LLMs for surveys, specifically on expectation. Bybee (2025) prompts LLMs with historical news articles from the Wall Street Journal and forecast financial and macroeconomic variables given this information. In comparison against historical forecasts, Bybee finds that the correlation between LLM and historical forecasts is high and comparable to the correlation between historical forecasts. To test for lookahead bias, Bybee (2025) repeats this procedure with news articles after the training period of the used LLM and finds similar results. Brand, Israeli, and Ngwe (2026) report mixed results for asking LLMs about consumer valuations of products in marketing, and advocate fine-tuning LLMs with existing surveys for more accurate results. Tjuatja et al. (2024) report that LLMs are more sensitive to prompts than human respondents and that LLMs survey responses are not reflective of human survey responses. Zarifhonarvar (2026) finds that LLMs can replicate some household survey patterns, but documents and warns about the sensitivity of the LLMs survey respondents and responses to the LLM itself. In particular, LLM architecture, number of parameters, and the time window of the data on which a model was trained are documented to influence the survey respondents and responses generated by the LLM. Hansen et al. (2026) find similar results for the Survey of Professional Forecasters. Wu, Xi, and Xie (2025) propose date-restrictive prompting to attenuate lookahead bias and to fix the characteristics of respondents across treatment and control arms. In contrast to Brand, Israeli, and Ngwe (2026), Wu, Xi, and Xie (2025) report alignment of heterogeneity between human and LLM surveys.

4.4.3 Using Estimated Variables We now highlight two methodological challenges when using neural networks for reduced-form inference. One is relatively specific to LLMs, the second one is more general. First, recall that Avivi (2024) uses the embeddings of Srebrovic and Yonamine (2020) which is trained on patents. Using LLMs trained on text data particularly relevant for an economic application has the advantage that embeddings may reflect the particular structure of this application, in this context, patents. One disadvantage is that outcomes may "leak" from the dataset on which the LLM is trained. An obvious example would be an LLM trained with data up to 2025 which is then used to predict stock prices in 2023, see for example Glasserman and Lin (2023); Sarkar and Vafa (2024); Lopez-Lira, Tang, and Zhu (2025). When the main goal of a paper is prediction, Ludwig, Mullainathan, and Rambachan (2025) recommend avoiding such leakage by using either LLMs with fixed, publicly available weights such as

GPT-2, DeepSeek or the BERT or Llama family, or (families of) LLMs trained on time-stamped training data, for example Sarkar (2024); He et al. (2025). A more recent development, proposed by Engelberg et al. (2025), is the possibility of "entity neutering", in which an LLM iteratively removes any information from texts about firms until another LLM can no longer infer the identity of the firm.

The purpose of using an LLM in Avivi (2024) is not prediction but creating control variables for a regression analysis. It is an example of a two-step approach in which the first step consists of using neural networks or machine learning more generally to estimate control variables, and the second step consists of running a regression with the estimated variables as control. This poses the question of how potential estimation error of the regression variable, i.e. measurement error of the regression variable, influences the regression coefficients. This is studied in various papers.

Battaglia et al. (2024) consider such plug-in estimates, specifically AI/ML-Generated labels such as age, gender, or race estimated from an image as in Adukia et al. (2023); topic models to reduce the dimensionality of unstructured data; and AI/ML-generated indices such as policy uncertainty estimated from news articles in Baker, Bloom, and Davis (2016). Battaglia et al. (2024) consider an asymptotic framework in which measurement error (i.e. the degree to which there is noise in the estimated regression variable) and sampling uncertainty (i.e. the amount of noise in the regression) have the same order of magnitude. On plug-in estimators, they find that the naive confidence interval which does not account for estimation error in the regression variable has the correct width but is not correctly centered. Because measurement error is non-classical, the bias can be attenuating or amplifying. Battaglia et al. (2024) provide analytical bias corrections which can be applied without validation data to AI/ML-generated binary labels when one has access to the classifier's expected false-positive rate as in Bursztyn et al. (2024) and for dimension reduction. As a more general solution, they propose a joint MLE for the regression variable and the regression coefficients. For this, one has to model the relation between regression coefficient and the high-dimensional object from which it is derived (i.e. the text or image) and use a Hamiltonian Monte Carlo to integrate out the unobserved regressor(s). Their theoretical analysis assumes convergence of the AI/ML estimator in the fixed-DGP case and faster than root-n convergence in the sequence-of-DGP case, a strong assumption for ML/AI classifiers which we are not aware of low-level conditions for. As one of several

examples, Battaglia et al. (2024) replicate the analysis in Gorodnichenko, Pham, and Talavera (2023), and find joint estimation of regression-coefficients and AI/ML-generated labels leads to quantitatively very different results, with the regression coefficients for sentiment differing by a factor of 3.

For the same challenge, Ludwig, Mullainathan, and Rambachan (2025) propose a simpler approach which requires a validation dataset. This may be hard for Avivi (2024) where a ground truth on patent quality is hard to observe for researchers but possible for gender, age, or race inferred from images as in Adukia et al. (2023). Ludwig, Mullainathan, and Rambachan (2025) study linear regression when either the dependent or independent variable is the output of an LLM and show that one has to essentially assume no measurement error by the LLM. They also document the sensitivity of LLM output and the resulting regression coefficients to the precise wording of the prompt, raising concerns about p-hacking. As a solution, Ludwig, Mullainathan, and Rambachan (2025) recommend constructing a (possibly small) validation sample and debiasing the LLM output to obtain consistency and asymptotic normality of OLS estimates, building on results in the literature in econometrics and machine learning, see Lee and Sepanski (1995); Chen, Hong, and Tamer (2005); Schennach (2016); Wang, McCormick, and Leek (2020); Angelopoulos et al. (2023); Egami et al. (2023) and Carlson and Dell (2025). Compared to the regression estimator which uses only the correctly measured variables (say by the researcher), the regression which also uses the potentially mismeasured outcomes and uses bias correction can have smaller variance.

For some experiments, outcome variables cannot be measured and are instead estimated using an auxiliary variable.[19] One example, as in Chen and Nordhaus (2011); Henderson, Storeygard, and Weil (2012); Asher et al. (2021), would be to use night lights to predict economic activity at a geographically granular level due to the difficulty and expense of collecting this data manually. A common practice is to collect economic outcomes for some observations and train a machine learning algorithm to predict outcomes given the remotely sensed variables. One can then use the predicted outcomes of the ML procedure as outcomes in a regression to estimate treatment effects using possibly a different sample for which one can measure the treatment. As discussed by Chen, Hong, and Tamer (2005); Chen, Hong, and Tarozzi (2008);

[19]This relates to the econometric literature on surrogate variables which are sometimes used when outcomes are only measured with long temporal delays, see e.g. Prentice (1989); Athey et al. (2025).

Graham, Pinto, and Egel (2016), this procedure identifies the effect of treatment if the conditional distribution of the outcome given the auxiliary variable is stable across samples. Rambachan, Singh, and Viviano (2024) instead suppose that the conditional distribution of the auxiliary variable given the outcome is stable across samples in which case the commonly used estimator described above can be biased. Rambachan, Singh, and Viviano (2024) propose an identification argument even when the sample of observations for which treatment is measured and the sample for which outcomes are measured has no overlap. The identification argument is based on identifying the conditional distribution of the outcome, the treatment and the sample indicator given the auxiliary variable, three problems in which neural networks can be used. Importantly, their inference results do not require convergence rate assumptions of the neural network.

Carlson and Dell (2025) are interested in semiparametrically efficient estimation of a regular linear functional (θ) of some latent structure (M^*), e.g. $\theta := E[M^*]$. For all observations, U and a vector of low-dimensional characteristics X are observed. The latent structure is a deterministic function of an unstructured text U and only for a subset of the population, M is observed (the "annotated" sample). Formally, the researcher observes $M = AM^*$ and $A \in \{0, 1\}$, the annotation indicator. They assume annotation on observables, i.e., $[(U, M^*) \perp\!\!\!\perp A] \mid X$. Under this assumption, as usual $\mathbb{P}[A = 1|X, U] = \mathbb{P}[A = 1|X] =: \pi(X)$. They call π the "annotation score function" and $\mu(x, u) = \mathbb{E}[M|X, U]$ the "imputation function". For this survey, the reader can imagine that Carlson and Dell (2025) assume that "annotation score function" $\pi(A = 1|X = x)$ is perfectly known to the researcher and exhibits strong overlap.[20] The authors assume that the estimated imputation function $\hat{\mu}$ converges in MSE to μ. Under these assumptions and moment conditions, the efficient influence function for θ is $\varphi(W) := \mu(X, U) - \theta + \frac{A}{\pi(X)}[M - \mu(X, U)]$. Mathematically, this fits into the general framework of Chen, Hong, and Tarozzi (2008). Since the efficient influence function is automatically a Neyman-orthogonal moment, Carlson and Dell (2025) obtain an asymptotically efficient estimator for $\theta := E[M^*]$ that is also doubly robust. Recently, Ruan et al. (2025) propose a causal analysis leveraging language models, a principled statistical framework that integrates LLM-generated insights of RCTs to increase the precision of the RCT estimates.

[20]The assumption of a known annotation score function can be relaxed when convergence rates of estimated annotation function and imputation function are given.

4.4.4 Efficient Average Treatment Effects with One-Step NN Sieve Estimation Building directly on the general NN sieve M-estimation framework of Section 2.4.1, Chen et al. (2024) estimate average treatment effects under unconfoundedness without relying on a doubly robust Neyman-orthogonal moment of the sort used in Farrell, Liang, and Misra (2021b). Let $D \in \{0, 1, \ldots, J\}$ denote treatment, let X be a possibly high-dimensional vector of confounders, and $Y^*(d) \perp D \mid X$. Chen et al. (2024) define

$$\beta_d^* = \arg\min_{b \in \mathbb{R}} \mathbb{E}\left[\frac{\mathbf{1}\{D = d\}}{\pi_d(X)} L(Y - b)\right], \qquad \pi_d(x) = \mathbb{P}(D = d \mid X = x).$$

With different loss functions $L(v)$—squared loss v^2, check loss, or asymmetric squared loss—they obtain treatment-effect parameters for means, quantiles, and asymmetric functionals. Inference uses weighted bootstrap procedures; for example, for the squared loss $L(v) = v^2$ we have $\beta_d^* = \mathbb{E}[Y^*(d)]$ and hence ATE $= \beta_1^* - \beta_0^*$. Their main estimator approximates the generalized propensity score $\pi_d(\cdot)$ by a one-hidden-layer NN sieve and then minimizes the sample analogue of this criterion. Relative to double/debiased machine learning, the key simplification is that the NN enters as a sieve approximation to one nuisance family rather than through a doubly robust orthogonal score that requires simultaneous estimation of both propensity and outcome nuisance functions.

For the ATE specifically, Chen et al. (2024) also propose an alternative outcome-regression route that is especially appealing in practice. Let $z_d^*(x) = \mathbb{E}[Y|X = x, D = d]$. Then, under unconfoundedness, $\beta_d^* = \mathbb{E}[z_d^*(X)]$. If $\hat{z}_d$ is estimated by a one-hidden-layer NN sieve for each treatment arm, the resulting estimator is

$$\hat{\beta}_d^{OR} = \frac{1}{n} \sum_{i=1}^{n} \hat{z}_d(X_i), \qquad \widehat{\text{ATE}}^{OR} = \hat{\beta}_1^{OR} - \hat{\beta}_0^{OR}.$$

This avoids solving an estimated efficient-influence-function equation altogether. The NN is still used only as a flexible sieve for the nuisance regression function, while the target remains a low-dimensional causal functional.

The attraction of this alternative estimator is both theoretical and practical. Chen et al. (2024) show that the ANN-OR estimator for ATE has the same asymptotic variance as their ANN-IPW estimator under standard regularity conditions, yet unlike IPW and doubly robust estimators it can achieve root-n asymptotic

normality without imposing a strict lower bound on the propensity score. In designs where estimated propensity scores may be close to zero, this makes the one-step NN outcome-regression estimator more stable than orthogonal-score approaches. Their recommended inferential device is a weighted bootstrap, which is convenient because it avoids deriving complicated closed-form variance expressions. For a review article, the broader lesson is that neural nets need not enter reduced-form causal inference only through doubly robust orthogonal moments: shallow NN sieves can also support direct semiparametrically efficient estimation of the ATE.

4.4.5 Spatial Treatment Assignment Pollmann (2020) proposes to evaluate spatially assigned treatment under a spatial unconfoundedness assumption, which ensures that treatment assignment of two sufficiently distant locations is unconfounded. This motivates a doubly robust propensity score approach with controls being neighborhood characteristics, e.g. satellite images. To control for such high-dimensional characteristics, Pollmann (2020) proposes the use of convolutional neural nets (CNN); see Section 3.2.1. The network is trained twice, once to fit the outcomes and once to fit treatment indicators with the usual output layers and loss functions. With appropriate conditioning, the CNN estimated nuisance functions are used to compute the doubly-robust propensity score estimand. Alternatively, one can just use CNN to train/estimate the outcome regression once, and obtain the semiparametric efficient estimation as in Chen et al. (2024).

4.4.6 Network Interference Leung and Loupos (2024) considers network interference in potential outcomes and selection into treatment. To apply a doubly robust propensity score estimator, the challenge is to control for network position and characteristics of a unit as well as characteristics of other units. The network's adjacency matrix gives rise to a notion of distance between two units and motivates an "approximate neighborhood interference" assumption as in Leung (2022), according to which only the characteristics of s-neighbors matter, i.e. characteristics of neighbors who can be reached in s steps on the network graph, where s is some "small" number. Still, this is a high-dimensional control problem. Leung and Loupos (2024) proposes to use graph neural nets for this control problem. They consider PNA aggregations, which are vectors of aggregated statistics such as mean, median, min, max, variance etc of hidden values of connected neurons in the previous layer, see Section 3.2.2 for some details. Other approaches are possible; Veitch, Wang, and Blei (2019) proposes to use node embeddings along the lines of

Veitch, Sridhar, and Blei (2020).

The number of layers in the GNN has the economic interpretation of a bound on the maximum distance between two nodes which interfere. The network is trained twice, once to fit the outcomes and once to fit treatment indicators with the usual output layers and loss functions. With appropriate conditioning, this allows them to approximate all nuisance functions to compute the doubly-robust propensity score estimand Chernozhukov et al. (2018) to compare the effects of any two treatment exposures. Under the assumption that GNN's rates of convergence are sufficiently fast, Leung and Loupos (2024) derive inference results for the treatment effect estimates. Recently, Wang, Gu, and Otsu (2024) have studied the convergence rate of graph neural networks, which could be viewed as a nonlinear sieve M-estimation problem. Under some regularity conditions, they show that the convergence rate of graph neural networks is sufficiently fast to allow for semiparametric inference as in Farrell, Liang, and Misra (2021b). Alternatively, one can just use GNN to train/estimate the outcome regression once, and obtain the semiparametric efficient estimation as in Chen et al. (2024).

4.5 Applications in Structural Models

4.5.1 Neuro-Computation Bodéré (2023) considers a dynamic model of firm entry in the Pennsylvanian preschool education market in which firms are heterogeneous in their location and quality. To estimate the parameters influencing dynamic decisions, in particular entry cost and fixed operating cost, he uses parametric policy iteration Sweeting (2013) and the nested pseudo-likelihood algorithm of Aguirregabiria and Mira (2007). This approach requires solving the dynamic games for many parameter values, including solving a static pricing game of heterogeneous firms in every period. To make this tractable, Bodéré (2023) approximates the function that takes the state vector as inputs and returns the equilibrium price with neural nets, trained on a synthetic data set $(s, \pi^{\text{eq}}(s))$ of states s and equilibrium variable profit vectors $\pi^{\text{eq}}(s)$ for which the static pricing game was solved exactly. To approximate the value function, he uses a linear function approximation where the features are basis functions of values of the output layer and the last layer of the neural net approximating variable static profits.

Fernández-Villaverde, Hurtado, and Nuño (2023) propose a continuous-time heterogeneous agent model to study the nonlinear relationship between aggregate and financial variables. A critical object to solve this model is to obtain a good estimate of the perceived law of motion (PLM) of aggregate debt as a function of state variables. In their model, the usual log-linear specification as proposed by Krusell and Smith (1998) does not perform well so they consider more flexible functions. They report that neural nets outperform approximations with Chebyshev polynomials.

Several papers apply variational inference, introduced in Section 3.3.1, to economics and finance. Applications include team production models in Bonhomme (2021), large Bayesian VARs in Chan and Yu (2022), multinomial probit models in Loaiza-Maya and Nibbering (2023), and network formation models in Mele and Zhu (2023). Recently, Balke, Bonhomme, and Lamadon (2025) evaluate variational inference methods for estimating dynamic earnings models with latent heterogeneity. They consider various variational posterior families in a simulation calibrated to the existing estimates reported in the literature. Simple mean-field posteriors perform poorly while unrestricted Gaussian variational posteriors and their structured counterparts exploiting the dynamic structure of the model perform better. However, richer models are computationally more demanding and subject to higher variances.

4.5.2 Structural Estimation Wei and Jiang (2025) propose casting structural estimation as a supervised learning problem: For a given structural parameter value of a parametric distribution model with latent variables, one draws a dataset as large as the dataset available to the researcher. This is repeated for a large set of randomly drawn structural parameters. The supervised learning problem is now to learn the map from a dataset to a structural parameter. As a simplification, this problem could take in a set of estimated moments instead of the entire dataset. Wei and Jiang (2025) show that if the draws of the structural parameter follow a correctly specified prior distribution, then as the size of draws of the structural parameters increases, the resulting estimator based on the entire dataset converges to the conditional expectation of the structural parameter given the data moments with respect to the Bayesian posterior. This approach also allows them to estimate the posterior conditional covariance of the structural parameter if the network's output layer is extended to also include the covariance matrix.

Structural estimation with maximum likelihood often struggles due to the difficulty of writing down or evaluating the likelihood, e.g. because of intractable integrals when there are latent variables or when they involve solutions to dynamic programming problems. McFadden (1989); Pakes and Pollard (1989) propose the simulated method of moments as a tractable and intuitive solution. It requires the researcher to choose a set of moments to target. This choice involves a bias-variance trade-off: The more moments are targeted, the lower the variance of the estimator, asymptotically up to the Cramér-Rao bound in the limit. But as shown by Altonji and Segal (1996), more moments also lead to higher bias in finite samples. For certain models, it is not obvious which moments identify parameters, e.g. the discount rate in dynamic discrete choice models, see Abbring and Daljord (2020).

To form moments, Gallant and Tauchen (1996); Gallant and Long (1997) propose to use the score directly. Of course, when the likelihood cannot be expressed analytically or be evaluated, the same difficulties apply to the score. Thus, they propose to approximate the score nonparametrically with Hermite polynomial sieves. This approach is implemented in Bansal et al. (1993, 1995). Kaji, Manresa, and Pouliot (2023) propose to choose moments for the simulated method of moments based on cross-entropy Generative Adversarial Networks, see Section 3.3.2. Intuitively, the forger (generator) is trying to choose parameters of the structural model to confuse the police (discriminator) between the true data and the generated data.[21] On the one hand, Kaji, Manresa, and Pouliot (2023) show that a logistic discriminator taking in a set of moments leads to structural estimates that are asymptotically equivalent to the SMM estimator for that set of moments. On the other hand, as shown by Kaji, Manresa, and Pouliot (2021), an oracle discriminator is asymptotically equivalent to maximum likelihood estimation. Discriminators based on neural nets can be seen as middle ground between logistic and oracle discriminators, and can in theory achieve asymptotic efficiency despite intractability of the likelihood and without choosing moments by hand. Nevertheless, the practical implementation using DNN is difficult.[22]

[21] It appears that Lewis and Syrgkanis (2018) developed a similar idea independently with the variation that instead of cross-entropy, Lewis and Syrgkanis (2018) considered unconditional moment conditions implied by a set of conditional moment restrictions as the prize in a zero-sum game. This paper was later subsumed into Dikkala et al. (2020) for NPIV estimation.

[22] Cigliutti and Manresa (2022) study finite-sample properties of an adversarial method of moments with logistic discriminator in light of the finite-sample bias of GMM estimators documented by Altonji and Segal (1996).

Observed Heterogeneity Farrell, Liang, and Misra (2021a) propose to enrich structural models by letting structural parameters depend on a large set of covariates. As an example, a simple structural model could be the logit demand model with a fixed preference vector. This model has disadvantages such as the IIA property of the logit and the implied substitution patterns. Berry, Levinsohn, and Pakes (1995) showed how a model where the preference coefficients are random, which breaks the IIA property, can be estimated with aggregated data. Farrell, Liang, and Misra (2021a) consider individual-level data in which there is rich demographic information about the consumers. Consumer i has demographics d_i and preference vector $\theta_i = f(d_i)$, where f could be a neural net. The neural net takes demographics d_i, product characteristics X_i and the choice Y_i as inputs. Demographics are fed to a fully-connected multi-layer neural net whose output layer is the vector of preference coefficients $\theta_i = f(d_i)$. The estimated coefficients can then be used to study questions regarding welfare, optimal pricing etc. Endogeneity could potentially be accommodated by adding instruments as an input to the neural net and in the loss function.

Dubé and Misra (2023) apply this methodology to the pricing problem of ZipRecruiter, an online marketplace for jobs and tasks. Dubé and Misra (2023) model individual i's demand for ZipRecruiter's product as a logit with coefficients $\beta_i = f(x_i)$, where f could be a linear function estimated using LASSO or a neural net with 2 or 3 layers, all of which perform comparably. They find that charging the estimated optimal uniform price increases profits by 55% while charging the estimated optimal personalized price increases profits by 86%, so an additional 19% in profits. However, this analysis is based only on a point-estimate of demand. Inference for the profit derived from this estimated demand function is difficult. Chen, Chen, and Gao (2025) provide such inference in the context of treatment assignment based on the sign of the Conditional Average Treatment Effect, establishing a parametric convergence rate, closed form expressions for the asymptotic variance, as well as estimators for the asymptotic variance based on a geometric insight.

For personalized pricing, Zhang (2023) proposes to use Policy DNNs instead of the framework of Farrell, Liang, and Misra (2021b), i.e. to directly map individual characteristics x_i to optimal prices $p^*(x_i)$ rather than to estimate demand parameters and then use a parametric model to derive optimal prices. Zhang (2023) shows that inference results in Farrell, Liang, and Misra (2021b) also apply to this architecture. As

the European GDPR affords consumers a "right to explanation" for personalized pricing and neural nets are not transparent in this regard, Zhang (2023) considers the loss in profits if firms are required to use decision trees, which are easily interpretable, instead of neural nets. The loss of approximating the optimal prices from the neural net by a "comprehensible" policy, a tree with up to five branches that is greedily trained to fit the neural net, is only 7%.[23]

Demand Estimation Demand estimation is the basis for pricing decisions, analyses of market power, and welfare calculations, as reviewed by Berry and Haile (2021). Neural nets hold the promise of flexible nonparametric demand estimation, including in NPIV settings. One recent example of an application of NPIV to demand estimation is Chen, Chen, and Tamer (2023), discussed in Section 2.4.2. A central obstacle, however, is posed by shape restrictions: economic theory often implies monotonicity, curvature, or other restrictions that are easier to impose with splines or Bernstein polynomials, see, e.g., Chen (2007); Compiani (2022); Brand and Smith (2025); Breunig and Chen (2024). Some progress exists for specialized architectures such as input-convex neural nets studied by Amos, Xu, and Kolter (2017) and other shape restricted NNs reviewed in Section 2.3. At the moment, many empirical IO papers simply use NNs to construct product representations or embeddings that are then fed into structurally interpretable demand systems. However, as discussed in Section 2.1, flexible NN representations/embeddings are not unique so that one needs to be cautious with economic interpretations.

Measuring product characteristics can be difficult, and mismeasurement may lead to endogeneity of prices. To augment product characteristics, Magnolfi, McClure, and Sorensen (2022) propose product embeddings based on triplets data of the form "product A is closer to product B than product C". These embeddings can then be used for demand estimation in characteristics space or product space. The triplet data is collected in surveys, a relatively expensive process.

To scale the computation of embeddings, text and image data is often available. Text data is often supplied by vendors in the form of product descriptions and by consumers in the form of reviews. Vendors often also supply image data. Among the first papers to study this is Han et al. (2021) who studies the market

[23]This surprisingly small loss echoes a similar result that deep nets can often be well approximated with shallow nets, see Ba and Caruana (2014) for a discussion.

for fonts using a Hotelling-type spatial competition model. To measure the similarity of fonts, they use FaceNet, proposed by Schroff, Kalenichenko, and Philbin (2015), a CNN which generates an embedding for faces into Euclidean space such that the embeddings' Euclidean distance becomes a similarity measure for the faces. With a similar approach, Han and Lee (2025) study copyright in the market for fonts. Compiani, Morozov, and Seiler (2025) propose to use text and image data to measure similarity between products by measuring the distances of their embeddings. For text, encoder-only models such as the Sentence Transformer (ST) of Cer et al. (2018) and S-Bert by Reimers and Gurevych (2019) directly produce embeddings. For images, pre-trained classifiers such as VGG19 by Simonyan and Zisserman (2014), ResNet50 by He et al. (2016), InceptionV3 by Szegedy et al. (2016), and Xception by Chollet (2017) are used. Two ways of incorporating embeddings in demand models are proposed. In a pairwise combinatorial nested logit model, as in Small (1987); Koppelman and Wen (2000), similarity measures are used to parametrize the error correlation. In a mixed logit model, following Berry, Levinsohn, and Pakes (1995), Compiani, Morozov, and Seiler (2025) apply PCA to the embeddings of each product and then use the principal components as characteristics. Christensen and Compiani (2026) provide root-n asymptotically normally distributed bias-corrected estimates of average demand counterfactuals that allow for unstructured data generated regressors. Bajari et al. (2023) use the same embeddings to compute hedonic price indices, i.e. to predict prices based on product embeddings, to compute price indices that are more robust to product entry and exit. For Amazon data on first-party apparel sales, the hedonic price functions have a surprisingly high R^2 of 80-90%.

In related work, Ruiz, Athey, and Blei (2020) develop a model of sequential decision making during shopping with large choice sets. It is inspired by neural probabilistic models proposed by Bengio, Ducharme, and Vincent (2000). Neural probabilistic models model sequences by learning the similarity of two items (e.g. two synonymous words in a sentence or two highly substitutable goods in a shopping cart) and the probability of item sequences (i.e. that a certain word follows another specific word or that a certain item is bought after another specific item is bought). However, the model itself is a hierarchical latent variable model that does not use neural nets, so we do not go into more detail in this review.

Chiang et al. (2026) consider market counterfactuals with nonparametric supply assuming that demand has already been identified. Their flexible supply function, defined as the sum of markups and

marginal cost, allows them to identify and estimate counterfactuals which do not need to distinguish marginal cost and markups for various models of firm conduct, see also Berry and Haile (2014). They estimate the model using the Variational Method of Moments as proposed in Bennett and Kallus (2023) and illustrate that the method scales to an empirical application to airline markets. In a simulation study, they find that the method estimates counterfactuals well with datasets of typical size.

Manski (1975, 1985, 1987) studies identification and estimation of discrete choice models where the utility of an option j to consumer i is $u_{i,j} = x_j\theta + \varepsilon_{i,j}$. Here, x_j are the characteristics of option j and $\varepsilon_{i,j}$ are latent utility draws for which no distribution is assumed. Manski shows that with sufficient variation in characteristics, θ is identified up to scale and proposes an estimator for θ which involves indicator functions. In old neural network architectures, the indicator function (also called Heaviside function) has been used as an activation function Rosenblatt (1958). Due to the lack of differentiability, indicators have been replaced with various other activation functions, see appendix A. Chen, Gao, and Wen (2025) propose to use a population objective based on a sum of two convolutions of ReLU functions. They show that this ReLU-based maximum score offers computational advantages for optimization using gradients and establish the convergence rate as well as the asymptotic normality of their ReLU-based maximum score reformulation.

4.5.3 Simulation Data Athey et al. (2021) argue that the credibility of Monte Carlo simulations to evaluate new methodologies would benefit if the design of the DGPs in the Monte Carlo simulation was not left to the researcher who is advocating a new methodology. Applications to real data are credible but have the issue that the DGP is not known. As a compromise, Athey et al. (2021) propose to use GANs, specifically Wasserstein GANs, see Section 3.3.2, to approximate the DGP of a real dataset. The estimated DGP can then be used to evaluate new methodologies with less concern that the researcher cherry-picked the DGP. As an added benefit, Xie et al. (2018) show that the estimated DGP satisfies a privacy guarantee so that it might be possible to share the estimated DGP even when a dataset is proprietary. More recently, diffusion-based methods to generate data have gained popularity. See Section 3.3.3 for an introduction and Tian and Shen (2026) for an application to generating data on a low-dimensional manifold in a high-dimensional space.

4.5.4 Dynamics Dynamic decision problems and dynamic games often feature high-dimensional state spaces and a curse of dimensionality in the sense of Bellman (1957). For example, Aguirregabiria and Nevo (2013) write

> *"The [...] 'curse of dimensionality' [is] present even in relatively simple models. [...] As a result, much of the recent work in structural econometrics in IO focuses on finding ways to make dynamic problems more tractable [...]."*

As neural nets can approximate multi-dimensional functions well in theory, they have been used to approximate value functions, going back to Tesauro (1995), as well as action-value functions, policy functions, and distributions of rewards and state transitions. Indeed, many recent successes in games with large state spaces have been documented in classical board games by Gelly and Silver (2008); Silver et al. (2016, 2018), in video games by Mnih et al. (2013); Vinyals et al. (2019), in chatbots by Ouyang et al. (2022), and in self-driving vehicles, as reviewed in Kiran et al. (2021). Due to space constraints, we cannot cover these approaches in much detail and refer the reader to the classical textbook of Sutton and Barto (2018), as well as the more compact introductions highlighting the links to structural estimation by Igami (2020); Iskhakov, Rust, and Schjerning (2020). Perhaps the successes of neural nets in the above applications are due to their architectures designed for state spaces that satisfy some lower-dimensional manifold hypotheses. In general, new neural net architectures imposing shape constraints based on economic theory or economic assumptions for specific contexts may be needed; see, e.g., Hendel and Nevo (2006); Gowrisankaran and Rysman (2012); Aguirregabiria and Ho (2012); Ifrach and Weintraub (2016); Fernández-Villaverde, Hurtado, and Nuño (2023); Han, Yang, and E (2026); Azinovic-Yang and Žemlička (2025).

5 Conclusion

We have introduced neural networks, discussed theoretical perspectives, and reviewed empirical applications in economics. We hope that the metaphor of neural networks as Lego structures inspires economists to combine different types of layers in novel ways to answer important questions. For example, multimodal learning that combines text and video data from a FOMC press conference looks promising. Our goal is to inspire economists to incorporate neural networks into their research. Yet, many aspects remain poorly

understood. We conclude with several open questions.

Economists who have access to large and accurate data sets can harness LLMs and other deep-learning tools to conduct more impactful research, as discussed by Varian (2014); Athey (2015); Mullainathan and Spiess (2017); Dell (2024). Economists can also apply NN architectures, including so-called physics-informed NNs, to accelerate the computation and simulation of complex high-dimensional dynamic equilibrium models, as reviewed in Fernández-Villaverde, Nuño, and Perla (2024). However, for researchers conducting structural estimation and inference with moderate and noisy data sets, care is needed when applying modern deep learning. For instance, current diffusion models are not effective for simulating economic structural models with latent heterogeneity and noisy data. As another example, many default NN packages are designed for regression and classification, that is, for prediction; they are therefore not yet directly suited to problems such as nonparametric regression with endogeneity in demand estimation.

The standard empirical practice in economics is to assume that the machine algorithms can find global solutions, and hence computational error is of a much smaller order than the statistical error (statistical bias + standard deviations) and can be ignored. Modern NNs and LLMs are nonlinear and high-dimensional, and computations are based on various versions of stochastic approximations (in particular, mini-batch versions of stochastic gradient descent). For single-hidden-layer neural networks, there are some recent results establishing global convergence of stochastic approximation algorithms, see, for example Zhong et al. (2017); Chen, Tamer, and Yao (2026). However, even for neural networks with two hidden layers, global optimization is already provably difficult. Consequently, statistical/econometric theories established based on finding global optimizers are likely to be poor approximations to empirical performance and should be regarded with reservations. Instead, inferential theory should take into account the approximation error (due to sieve or regularization), the statistical error and the computational error terms. Simon et al. (2026) argue that models from physics could help understand the learning process and thus the computational error terms.

Despite the successes of neural nets across many fields, especially in LLMs, they are not immune to error. The theoretical basis for their success remains only partially understood. Neither the Kolmogorov-Arnold representation view nor the flexible approximation view fully explains their remarkable empirical

performance; mathematically, many other sieve bases share similar high-level properties. Indeed, as discussed in Zhang et al. (2016), it is often unclear why one neural network trained on a particular data set and prediction task generalizes well while another does not. By the no-free-lunch theorem of Wolpert (1996); Wolpert and Macready (1997), it would be overly optimistic to expect neural networks to work well for all tasks and all data sets. The sub-manifold hypothesis suggests that seemingly high-dimensional objects such as text and images may lie on much lower-dimensional latent manifolds on which neural networks perform and generalize well. Unfortunately, without economic shape/sparsity restrictions, it is very difficult to nonparametrically recover the lower-dimensional latent manifold accurately from high-dimensional noisy data. So, while neural networks perform well in many prediction and classification tasks using large accurate data sets, they can perform poorly in generalization. One manifestation of these challenges is the prevalence of adversarial examples documented by Szegedy et al. (2013); Goodfellow, Shlens, and Szegedy (2014); Belkin, Hsu, and Mitra (2018).

As illustrated in Section 4.3: fine-tuned foundation models can sometimes outperform architectures specifically trained for a task. However, such models are prohibitively expensive to train for individual applied researchers, raising questions about the theoretical properties of networks trained on unknown data using unknown methods; see, for example, Springer et al. (2026) about the safety concerns of fine-tuning algorithms. In Section 4.4.3, we review some recent econometric literature on using DNN estimated variables in estimation of some policy functionals. Some approaches require strong assumptions, such as fast convergence rates of black-box LLM estimated embeddings. When large networks are trained on proprietary data with undisclosed methods, these assumptions are difficult to verify in practice.

This discussion should serve as a cautionary note: neural networks are not infallible machines. At the same time, we hope to provide a starting point for econometricians to deepen our understanding of fundamental theoretical aspects underlying the application of neural networks in economics. Simon et al. (2026) hosts a list of open questions related to the statistics of neural networks and their training. We encourage econometricians to contribute to these open questions — and, in doing so, to follow in the footsteps of Hal White.

References

Abbring, Jaap H and Øystein Daljord. 2020. "Identifying the Discount Factor in Dynamic Discrete Choice Models." *Quantitative Economics* 11 (2):471–501.

Abdar, Moloud, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya et al. 2021. "A Review of Uncertainty Quantification in Deep Learning: Techniques, Applications and Challenges." *Information Fusion* 76:243–297.

Abramitzky, Ran, Leah Boustan, Katherine Eriksson, James Feigenbaum, and Santiago Pérez. 2021. "Automated Linking of Historical Data." *Journal of Economic Literature* 59 (3):865–918.

Abudy, Matan, Orr Well, Emmanuel Chemla, Roni Katzir, and Nur Lan. 2025. "A Minimum Description Length Approach to Regularization in Neural Networks." *arXiv preprint arXiv:2505.13398* .

Acikalin, Utku U, Tolga Caskurlu, Gerard Hoberg, and Gordon M Phillips. 2022. "Intellectual Property Protection Lost and Competition: An Examination Using Machine Learning." .

Ackerberg, Daniel, Xiaohong Chen, and Jinyong Hahn. 2012. "A Practical Asymptotic Variance Estimator for Two-Step Semiparametric Estimators." *Review of Economics and Statistics* 94 (2):481–498.

Ackerberg, Daniel, Xiaohong Chen, Jinyong Hahn, and Zhipeng Liao. 2014. "Asymptotic Efficiency of Semiparametric Two-Step GMM." *Review of Economic Studies* 81 (3):919–943.

Ackley, David H, Geoffrey E Hinton, and Terrence J Sejnowski. 1985. "A Learning Algorithm for Boltzmann Machines." *Cognitive Science* 9 (1):147–169.

Adukia, Anjali, Alex Eble, Emileigh Harrison, Hakizumwami Birali Runesha, and Teodora Szasz. 2023. "What We Teach about Race and Gender: Representation in Images and Text of Children's Books." *The Quarterly Journal of Economics* 138 (4):2225–2285.

Aguirregabiria, Victor and Chun-Yu Ho. 2012. "A Dynamic Oligopoly Game of the US Airline Industry: Estimation and Policy Experiments." *Journal of Econometrics* 168 (1):156–173.

Aguirregabiria, Victor and Pedro Mira. 2007. "Sequential Estimation of Dynamic Discrete Games." *Econometrica* 75 (1):1–53.

Aguirregabiria, Victor and Aviv Nevo. 2013. "Recent Developments in Empirical IO: Dynamic Demand and Dynamic Games." In *Advances in Economics and Econometrics: Tenth World Congress*, edited by Daron Acemoglu, Manuel Arellano, and EddieEditors Dekel, Econometric Society Monographs. Cambridge University Press, 53—-122.

Aher, Gati V, Rosa I Arriaga, and Adam Tauman Kalai. 2023. "Using Large Language Models to Simulate Multiple Humans and Replicate Human Subject Studies." In *International Conference on Machine Learning*. PMLR, 337–371.

Ai, Chunrong and Xiaohong Chen. 2003. "Efficient estimation of Models with Conditional Moment Restrictions Containing Unknown Functions." *Econometrica* 71 (6):1795–1843.

———. 2007. "Estimation of Possibly Misspecified Semiparametric Conditional Moment Restriction Models with Different Conditioning Variables." *Journal of Econometrics* 141 (1):5–43.

———. 2012. "The Semiparametric Efficiency Bound for Models of Sequential Moment Restrictions Containing Unknown Functions." *Journal of Econometrics* 170 (2):442–457. Thirtieth Anniversary of Generalized Method of Moments.

Alain, Guillaume and Yoshua Bengio. 2014. "What Regularized Auto-Encoders Learn from the Data-Generating Distribution." *The Journal of Machine Learning Research* 15 (1):3563–3593.

Alexopoulos, Michelle, Xinfen Han, Oleksiy Kryvtsov, and Xu Zhang. 2024. "More Than Words: Fed Chairs' Communication During Congressional Testimonies." *Journal of Monetary Economics* 142:103515.

Allais, Maurice. 1953. "Le Comportement de l'Homme Rationnel Devant le Risque: Critique des Postulats et Axiomes de l'école Américaine." *Econometrica: Journal of the Econometric Society* :503–546.

Altonji, Joseph G and Lewis M Segal. 1996. "Small-Sample Bias in GMM Estimation of Covariance Structures." *Journal of Business & Economic Statistics* 14 (3):353–366.

Amos, Brandon, Lei Xu, and J Zico Kolter. 2017. "Input Convex Neural Networks." In *International Conference on Machine Learning*. PMLR, 146–155.

Andrews, Donald. 1994. "Asymptotics for Semiparametric Econometric Models Via Stochastic Equicontinuity." *Econometrica* 62 (1):43–72.

Angelopoulos, Anastasios N, Stephen Bates, Clara Fannjiang, Michael I Jordan, and Tijana Zrnic. 2023. "Prediction-Powered Inference." *Science* 382 (6671):669–674.

Arjovsky, Martin and Léon Bottou. 2017. "Towards Principled Methods For Training Generative Adversarial Networks." *arXiv preprint arXiv:1701.04862* .

Arjovsky, Martin, Soumith Chintala, and Léon Bottou. 2017. "Wasserstein Generative Adversarial Networks." In *International Conference on Machine Learning*. Pmlr, 214–223.

Armstrong, Timothy, Marinho Bertanha, and Han Hong. 2014. "A Fast Resample Method for Parametric and Semiparametric Models." *Journal of Econometrics* 179 (2):128–133.

Arora, Abhishek and Melissa Dell. 2023. "LinkTransformer: A Unified Package for Record Linkage with Transformer Language Models." *arXiv preprint arXiv:2309.00789* .

Arora, Sanjeev, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. 2017. "Generalization and equilibrium in generative adversarial nets (gans)." In *International conference on machine learning*. PMLR, 224–232.

Arora, Sanjeev, Andrej Risteski, and Yi Zhang. 2018. "Do GANs learn the distribution? Some Theory and Empirics." In *International conference on learning representations*.

Ash, Elliott and Stephen Hansen. 2023. "Text Algorithms in Economics." *Annual Review of Economics* 15 (1):659–688.

Asher, Sam, Tobias Lunt, Ryu Matsuura, and Paul Novosad. 2021. "Development Research at High Geographic Resolution: An Analysis of Night-Lights, Firms, and Poverty in India using the SHRUG Open Data Platform." *The World Bank Economic Review* 35 (4):845–871.

Athey, Susan. 2015. "Machine Learning and Causal Inference for Policy Evaluation." In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 5–6.

Athey, Susan, Herman Brunborg, Tianyu Du, Ayush Kanodia, and Keyon Vafa. 2024. "LABOR-LLM: Language-Based Occupational Representations with Large Language Models." *arXiv preprint arXiv:2406.17972* .

Athey, Susan, Raj Chetty, Guido W Imbens, and Hyunseung Kang. 2025. "The Surrogate Index: Combining Short-Term Proxies to Estimate Long-Term Treatment Effects More Rapidly and Precisely." *Review of Economic Studies* :rdaf087.

Athey, Susan, Guido W Imbens, Jonas Metzger, and Evan Munro. 2021. "Using Wasserstein Generative Adversarial Networks for the Design of Monte Carlo Simulations." *Journal of Econometrics* :105076.

Avivi, Hadar. 2024. "Are Patent Examiners Gender Neutral?" *Working Paper* .

Azinovic-Yang, Marlon and Jan Žemlička. 2025. "Deep Learning in the Sequence Space." *Working Paper* .

Ba, Jimmy and Rich Caruana. 2014. "Do Deep Nets Really Need to Be Deep?" *Advances in Neural Information Processing Systems* 27.

Ba, Jimmy Lei, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. "Layer Normalization." *arXiv preprint arXiv:1607.06450* .

Bahdanau, Dzmitry. 2014. "Neural Machine Translation by Jointly Learning to Align and Translate." *arXiv preprint arXiv:1409.0473* .

Bai, Yu, Fan Chen, Huan Wang, Caiming Xiong, and Song Mei. 2024. "Transformers as Statisticians: Provable in-Context Learning with in-Context Algorithm Selection." *Advances in Neural Information Processing Systems* 36.

Bajari, Patrick, Zhihao Cen, Victor Chernozhukov, Manoj Manukonda, Suhas Vijaykumar, Jin Wang, Ramon Huerta, Junbo Li, Ling Leng, George Monokroussos et al. 2023. "Hedonic Prices and Quality Adjusted Price Indices Powered by AI." *arXiv preprint arXiv:2305.00044* .

Baker, Scott R, Nicholas Bloom, and Steven J Davis. 2016. "Measuring Economic Policy Uncertainty." *The Quarterly Journal of Economics* 131 (4):1593–1636.

Balke, Neele, Stéphane Bonhomme, and Thibaut Lamadon. 2025. "How Variational Are Earnings Dynamics?" .

Bansal, Ravi, A Ronald Gallant, Robert Hussey, and George Tauchen. 1995. "Nonparametric Estimation of Structural Models for High-Frequency Currency Market Data." *Journal of Econometrics* 66 (1-2):251–287.

Bansal, Ravi, Ronald Gallant, Robert Hussey, and George Tauchen. 1993. "Computational Aspects of Nonparametric Simulation Estimation." In *Computational Techniques for Econometrics and Economic Analysis*. Springer, 3–22.

Barron, Andrew. 1991. "Complexity Regularization with Application to Artificial Neural Networks." In *Nonparametric Functional Estimation and Related Topics*. Springer, 561–576.

———. 1993. "Universal Approximation Bounds for Superpositions of a Sigmoidal Function." *IEEE Transactions on Information theory* 39 (3):930–945.

Barron, Andrew, Jorma Rissanen, and Bin Yu. 1998. "The Minimum Description Length Principle in Coding and Modeling." *IEEE Transactions on Information Theory* 44 (6):2743–2760.

Bartlett, Peter L and Shai Ben-David. 2002. "Hardness Results for Neural Network Approximation Problems." *Theoretical Computer Science* 284 (1):53–66.

Battaglia, Laura, Timothy Christensen, Stephen Hansen, and Szymon Sacher. 2024. "Inference for Regression with Variables Generated by AI or Machine Learning." *Working Paper* .

Baydin, Atilim Gunes, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. 2018. "Automatic Differentiation in Machine Learning: A Survey." *Journal of Machine Learning Research* 18 (153):1–43.

Belkin, Mikhail, Daniel Hsu, and Partha Mitra. 2018. "Overfitting or Perfect Fitting? Risk Bounds for Classification and Regression Rules that Interpolate."

Bellman, Richard. 1957. *Dynamic Programming*. Princeton University Press.

Beltagy, Iz, Matthew E Peters, and Arman Cohan. 2020. "Longformer: The Long-Document Transformer." *arXiv preprint arXiv:2004.05150* .

Beltratti, Andrea, Sergio Margarita, and Pietro Terna. 1996. *Neural Networks for Economic and Financial Modelling*. International Thomson Computer Press London, UK.

Bengio, Yoshua, Réjean Ducharme, and Pascal Vincent. 2000. "A Neural Probabilistic Language Model." *Advances in Neural Information Processing Systems* 13.

Bengio, Yoshua, Patrice Simard, and Paolo Frasconi. 1994. "Learning Long-Term Dependencies with Gradient Descent is Difficult." *IEEE Transactions on Neural Networks* 5 (2):157–166.

Bennett, Andrew and Nathan Kallus. 2023. "The Variational Method of Moments." *Journal of the Royal Statistical Society Series B: Statistical Methodology* 85 (3):810–841.

Berner, Julius, Philipp Grohs, Gitta Kutyniok, and Philipp Petersen. 2022. *The Modern Mathematics of Deep Learning*. Cambridge University Press, 1–111.

Berry, Steven, James Levinsohn, and Ariel Pakes. 1995. "Automobile Prices in Market Equilibrium." *Econometrica: Journal of the Econometric Society* :841–890.

Berry, Steven T and Philip A Haile. 2014. "Identification in Differentiated Products Markets Using Market Level Data." *Econometrica* 82 (5):1749–1797.

———. 2021. "Foundations of Demand Estimation." *Handbook of Industrial Organization* 4 (1):1–62.

Bhavan, Anjali, Pankaj Chauhan, Rajiv Ratn Shah et al. 2019. "Bagged Support Vector Machines for Emotion Recognition from Speech." *Knowledge-Based Systems* 184:104886.

Binette, Olivier and Rebecca C Steorts. 2022. "(Almost) All of Entity Resolution." *Science Advances* 8 (12):eabi8021.

Bishop, Christopher M and Hugh Bishop. 2023. *Deep Learning: Foundations and Concepts*. Springer Nature.

Blei, David M, Alp Kucukelbir, and Jon D McAuliffe. 2017. "Variational Inference: A Review for Statisticians." *Journal of the American Statistical Association* 112 (518):859–877.

Blelloch, Guy E. 1990. "Prefix Sums and Their Applications." *Technical Report* .

Blundell, Charles, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. 2015. "Weight Uncertainty in Neural Network." In *International conference on machine learning*. PMLR, 1613–1622.

Bochkarev, Vladimir V, Anna V Shevlyakova, and Valery D Solovyev. 2015. "The Average Word Length Dynamics as an Indicator of Cultural Changes in Society." *Social Evolution and History* 14 (2):153–175.

Bodéré, Pierre. 2023. "Dynamic Spatial Competition in Early Education: An Equilibrium Analysis of the Preschool Market in Pennsylvania." *Job Market Paper* .

Bommasani, Rishi, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill et al. 2021. "On the Opportunities and Risks of Foundation Models." *arXiv preprint arXiv:2108.07258* .

Bonhomme, Stéphane. 2021. "Teams: Heterogeneity, sorting, and complementarity." *arXiv preprint arXiv:2102.01802* .

Boskin, Michael J. 1974. "A Conditional Logit Model of Occupational Choice." *Journal of Political Economy* 82 (2, Part 1):389–398.

Boustan, Leah and Ran Abramitzky. 2026. "The Linking Revolution: New Insights from Historical Panel Data." *Working Paper* .

Brand, James, Ayelet Israeli, and Donald Ngwe. 2026. "Using LLMs for Market Research." *Working Paper* .

Brand, James and Adam Smith. 2025. "A Quasi-Bayes Approach to Nonparametric Demand Estimation with Economic Constraints." *Working Paper* .

Breunig, Christoph and Xiaohong Chen. 2024. "Adaptive, Rate-optimal Hypothesis Testing in Nonparametric IV Models." *Econometrica* 92 (6):2027–2067.

Bryan, Tom, Jacob Carlson, Abhishek Arora, and Melissa Dell. 2023. "EfficientOCR: An Extensible, Open-Source Package for Efficiently Digitizing World Knowledge." *arXiv preprint arXiv:2310.10050* .

Bursztyn, Leonardo, Thomas Chaney, Tarek A Hassan, and Aakaash Rao. 2024. "The Immigrant Next Door." *American Economic Review* 114 (2):348–384.

Bybee, J Leland. 2025. "The Ghost in the Machine: Generating Beliefs with Large Language Models." *Working Paper* .

Cardell, Scott, Wayne Joerding, and Ying Li. 1995. "Symmetry Constraints for Feedforward Network Models of Gradient Systems." *IEEE Transactions on Neural Networks* 6 (5):1249–1254.

Carlson, Jacob and Melissa Dell. 2025. "A Unifying Framework for Robust and Efficient Inference with Unstructured Data." *arXiv preprint arXiv:2505.00282* .

Cer, Daniel, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar et al. 2018. "Universal Sentence Encoder for English." In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 169–174.

Chamberlain, Gary. 1987. "Asymptotic Efficiency in Estimation with Conditional Moment Restrictions." *Journal of Econometrics* 34 (3):305–334.

Chan, Joshua CC and Xuewen Yu. 2022. "Fast and Accurate Variational Inference for Large Bayesian VARs with Stochastic Volatility." *Journal of Economic Dynamics and Control* 143:104505.

Charness, Gary and Matthew Rabin. 2002. "Understanding Social Preferences with Simple Tests." *The Quarterly Journal of Economics* 117 (3):817–869.

Chatterjee, Sourav and Timothy Sudijono. 2026. "Neural Networks Generalize on Low Complexity Data." *The Annals of Statistics* 54 (1):350–382.

Chava, Sudheer, Wendi Du, and Baridhi Malakar. 2021. "Do Managers Walk the Talk on Environmental and Social Issues?" *Georgia Tech Scheller College of Business Research Paper* .

Chava, Sudheer, Wendi Du, and Nikhil Paradkar. 2020. "More than Buzzwords? Firms' Discussions of Emerging Technologies in Earnings Conference Calls." In *Firms' Discussions of Emerging Technologies in Earnings Conference Calls*.

Chen, Jiafeng, Xiaohong Chen, and Elie Tamer. 2023. "Efficient Estimation of Average Derivatives in NPIV Models: Simulation Comparisons of Neural Network Estimators." *Journal of Econometrics* 235:1848–1875.

Chen, Sitan, Sinho Chewi, Holden Lee, Yuanzhi Li, Jianfeng Lu, and Adil Salim. 2023. "The Probability Flow ODE is Provably Fast." *Advances in Neural Information Processing Systems* 36:68552–68575.

Chen, Xi and William D Nordhaus. 2011. "Using Luminosity Data as a Proxy for Economic Statistics." *Proceedings of the National Academy of Sciences* 108 (21):8589–8594.

Chen, Xiao, Yu Chen, Zhouyu Shen, and Dacheng Xiu. 2025. "Recurrent Neural Networks for Nonlinear Time Series." *Chicago Booth Research Paper* (26-01).

Chen, Xiaohong. 1993. *Asymptotic Properties of Recursive M-Estimators in an Infinite-Dimensional Hilbert Space*. University of California, San Diego.

———. 2007. "Large Sample Sieve Estimation of Semi-Nonparametric Models." *Handbook of Econometrics* 6:5549–5632.

Chen, Xiaohong, Zengjing Chen, Wayne Yuan Gao, Xiaodong Yan, and Guodong Zhang. 2026. "Optimization via the Strategic Law of Large Numbers." *PNAS* 123 (4):2519845123.

Chen, Xiaohong, Zhenxiao Chen, and Wayne Yuan Gao. 2025. "Inference on Welfare and Value Functionals under Optimal Treatment Assignment." *Working Paper* .

Chen, Xiaohong, Wayne Yuan Gao, and Likang Wen. 2025. "ReLU-Based and DNN-Based Generalized Maximum Score Estimators." *Working Paper* .

Chen, Xiaohong, Han Hong, and Matthew Shum. 2007. "Nonparametric likelihood ratio model selection tests between parametric likelihood and moment condition models." *Journal of Econometrics* 141 (1):109–140.

Chen, Xiaohong, Han Hong, and Elie Tamer. 2005. "Measurement Error Models with Auxiliary Data." *The Review of Economic Studies* 72 (2):343–366.

Chen, Xiaohong, Han Hong, and Alessandro Tarozzi. 2008. "Semiparametric Efficiency in GMM Models with Auxiliary Data." *The Annals of Statistics* :808–843.

Chen, Xiaohong, Yuan Liao, and Weichen Wang. 2025. "Inference on Time Series Nonparametric Conditional Moment Restrictions using Nonlinear Sieves." *Journal of Econometrics* 249:105920.

Chen, Xiaohong and Zhipeng Liao. 2015. "Sieve Semiparametric Two-Step GMM under Weak Dependence." *Journal of Econometrics* 189 (1):163–186.

Chen, Xiaohong, Zhipeng Liao, and Yixiao Sun. 2014. "Sieve Inference on Possibly Misspecified Semi-Nonparametric Time Series Models." *Journal of Econometrics* 178:639–658.

Chen, Xiaohong, Oliver Linton, and Ingrid Van Keilegom. 2003. "Estimation of Semiparametric Models when the Criterion Function Is Not Smooth." *Econometrica* 71 (5):1591–1608.

Chen, Xiaohong, Ying Liu, Shujie Ma, and Zheng Zhang. 2024. "Causal Inference of General Treatment Effects using Neural Networks with a Diverging Number of Confounders." *Journal of Econometrics* 238 (1):105555.

Chen, Xiaohong and Sydney C Ludvigson. 2009. "Land of Addicts? An Empirical Investigation of Habit-Based Asset Pricing Models." *Journal of Applied Econometrics* 24 (7):1057–1093.

Chen, Xiaohong and Demian Pouzo. 2012. "Estimation of Nonparametric Conditional Moment Models with Possibly Nonsmooth Generalized Residuals." *Econometrica* 80 (1):277–321.

———. 2015. "Sieve Wald and QLR Inferences on Semi/Nonparametric Conditional Moment Models." *Econometrica* 83 (3):1013–1079. URL https://onlinelibrary.wiley.com/doi/abs/10.3982/ECTA10771.

Chen, Xiaohong, Jeffrey Racine, and Norman R Swanson. 2001. "Semiparametric ARX Neural-Network Models with an Application to Forecasting Inflation." *IEEE Transactions on neural networks* 12 (4):674–683.

Chen, Xiaohong and Andres Santos. 2018. "Overidentification in Regular Models." *Econometrica* 86 (5):1771–1817.

Chen, Xiaohong and Xiaotong Shen. 1998. "Asymptotic Properties of Sieve Extremum Estimates for Weakly Dependent Data with Applications." *Econometrica* 66 (2):289–314.

Chen, Xiaohong and Norman R Swanson. 2012. "Recent Advances and Future Directions in Causality, Prediction, and Specification Analysis: Essays in Honor of Halbert L. White Jr." .

———. 2014. "Causality, Prediction, and Specification Analysis: Recent Advances and Future Directions." *Journal of Econometrics* 182 (1):1–4.

Chen, Xiaohong, Elie Tamer, and Qingsong Yao. 2026. "Online Learning in Semiparametric Econometric Models." *Working Paper* .

Chen, Xiaohong and Halbert White. 1999. "Improved Rates and Asymptotic Normality for Nonparametric Neural Network Estimators." *IEEE Transactions on Information Theory* 45 (2):682–691.

Chernozhukov, Victor, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. 2018. "Double/Debiased Machine Learning for Treatment and Structural Parameters." *The Econometrics Journal* 21 (1):C1–C68.

Chernozhukov, Victor, Juan Carlos Escanciano, Hidehiko Ichimura, Whitney K. Newey, and James M. Robins. 2022. "Locally Robust Semiparametric Estimation." *Econometrica* 90 (4):1501–1535.

Chetverikov, Denis, Andres Santos, and Azeem M Shaikh. 2018. "The Econometrics of Shape Restrictions." *Annual Review of Economics* 10 (1):31–63.

Chiang, Harold D., Jack Collison, Lorenzo Magnolfi, and Christopher Sullivan. 2026. "Market Counterfactuals with Nonparametric Supply: An ML/AI Approach." *Working Paper* .

Cho, Kyunghyun, Bart van Merriënboer, Çağlar Gulçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. "Learning Phrase Representations Using RNN Encoder–Decoder for Statistical Machine Translation." In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1724–1734.

Chollet, François. 2017. "Xception: Deep Learning with Depthwise Separable Convolutions." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1251–1258.

Christen, Peter. 2011. "A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication." *IEEE Transactions on Knowledge and Data Engineering* 24 (9):1537–1555.

Christensen, Timothy and Giovanni Compiani. 2026. "From Unstructured Data to Demand Counterfactuals: Theory and Practice." *Working Paper* .

Cigliutti, Ignacio and Elena Manresa. 2022. "Adversarial Method of Moments." *Working Paper* .

Cireşan, Dan Claudiu, Ueli Meier, Luca Maria Gambardella, and Jürgen Schmidhuber. 2010. "Deep, Big, Simple Neural Nets for Handwritten Digit Recognition." *Neural Computation* 22 (12):3207–3220.

Clark, Kevin, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. "ELECTRA: Pre-Training Text Encoders as Discriminators Rather than Generators." *arXiv preprint arXiv:2003.10555* .

Clevert, Djork-Arné. 2015. "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)." *arXiv preprint arXiv:1511.07289* .

Compiani, Giovanni. 2022. "Market Counterfactuals and the Specification of Multiproduct Demand: A Nonparametric Approach." *Quantitative Economics* 13 (2):545–591.

Compiani, Giovanni, Ilya Morozov, and Stephan Seiler. 2025. "Demand Estimation with Text and Image Data." *Working Paper* .

Coppejans, M. 2004. "On Kolmogorov's Representation of Functions of Several Variables by Functions of one Variable." *Journal of Econometrics* 22 (3):580–615.

Corliss, George F. 1988. "Applications of Differentiation Arithmetic." In *Reliability in Computing*. Elsevier, 127–148.

Corradi, Valentina and Halbert White. 1995. "Regularized Neural Networks: Some Convergence Rate Results." *Neural Computation* 7 (6):1225–1244.

Corso, Gabriele, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Veličković. 2020. "Principal Neighbourhood Aggregation for Graph Nets." *Advances in Neural Information Processing Systems* 33:13260–13271.

Cowen, Tyler and Alexander T Tabarrok. 2023. "How to Learn and Teach Economics with Large Language Models, including GPT." .

Croft, Thomas A. 1978. "Nighttime Images of the Earth from Space." *Scientific American* 239 (1):86–101.

Curti, Filippo and Sophia Kazinnik. 2023. "Let's Face It: Quantifying the Impact of Nonverbal Communication in FOMC Press Conferences." *Journal of Monetary Economics* 139:110–126.

Cybenko, George. 1989. "Approximation by Superpositions of a Sigmoidal Function." *Mathematics of Control, Signals and Systems* 2 (4):303–314.

Dai, Zihang, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. "Transformer-XL: Attentive Language Models beyond a Fixed-Length Context." *arXiv preprint arXiv:1901.02860* .

Daskalakis, Constantinos, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. 2017. "Training GANs with Optimism." *arXiv preprint arXiv:1711.00141* .

Dave, Vachik S, Baichuan Zhang, Mohammad Al Hasan, Khalifeh AlJadda, and Mohammed Korayem. 2018. "A Combined Representation Learning Approach for Better Job and Skill Recommendation." In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 1997–2005.

De Bortoli, Valentin, James Thornton, Jeremy Heng, and Arnaud Doucet. 2021. "Diffusion Schrödinger Bridge with Applications to Score-Based Generative Modeling." *Advances in Neural Information Processing Systems* 34:17695–17709.

Deb, Nabarun and Tengyuan Liang. 2025. "No-Regret Generative Modeling via Parabolic Monge-Amp\ere PDE." *arXiv preprint arXiv:2504.09279* .

Dell, Melissa. 2024. "Deep Learning for Economists." .

Dell, Melissa, Jacob Carlson, Tom Bryan, Emily Silcock, Abhishek Arora, Zejiang Shen, Luca D'Amico-Wong, Quan Le, Pablo Querubin, and Leander Heldring. 2024. "American Stories: A Large-Scale Structured Text Dataset of Historical US Newspapers." *Advances in Neural Information Processing Systems* 36.

Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. 2009. "ImageNet: A large-scale hierarchical image database." In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." In *Proceedings of the 2019 Conference of the North American chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (long and short papers)*. 4171–4186.

DeVore, Ronald, Robert D Nowak, Rahul Parhi, and Jonathan W Siegel. 2025. "Weighted Variation Spaces and Approximation by Shallow ReLU networks." *Applied and Computational Harmonic Analysis* 74.

Dikkala, Nishanth, Greg Lewis, Lester Mackey, and Vasilis Syrgkanis. 2020. "Minimax Estimation of Conditional Moment Models." *Advances in Neural Information Processing Systems* 33:12248–12262.

Dobbie, Will and Crystal S Yang. 2021. "The US Pretrial System: Balancing Individual Rights and Public Interests." *Journal of Economic Perspectives* 35 (4):49–70.

Donaldson, Dave and Adam Storeygard. 2016. "The View from Above: Applications of satellite Data in Economics." *Journal of Economic Perspectives* 30 (4):171–198.

Dowling, Michael and Brian Lucey. 2023. "ChatGPT for (Finance) Research: The Bananarama Conjecture." *Finance Research Letters* 53:103662.

Dubé, Jean-Pierre and Sanjog Misra. 2023. "Personalized Pricing and Consumer Welfare." *Journal of Political Economy* 131 (1):131–189.

Dubin, David. 2004. "The Most Influential Paper Gerard Salton Never Wrote." *Library Trends* 52 (4):748–764.

Duchi, John, Elad Hazan, and Yoram Singer. 2011. "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization." *Journal of Machine Learning Research* 12 (7).

Dunn, Halbert L. 1946. "Record Linkage." *American Journal of Public Health and the Nations Health* 36 (12):1412–1416.

Dupuis, Kate and M Kathleen Pichora-Fuller. 2010. "Toronto Emotional Speech Set." .

Ebraheem, Muhammad, Saravanan Thirumuruganathan, Joyu Shafiq, Nan Tang, and Mourad Ouzzani. 2018. "Distributed Representations of Tuples for Entity Resolution." *Proceedings of the VLDB Endowment* 11 (11).

Egami, Naoki, Musashi Hinck, Brandon Stewart, and Hanying Wei. 2023. "Using imperfect surrogates for downstream inference: Design-based supervised learning for social science applications of large language models." *Advances in Neural Information Processing Systems* 36:68589–68601.

Einav, Liran and Jonathan Levin. 2014. "Economics in the Age of Big Data." *Science* 346 (6210):1243089.

Engelberg, Joseph, Asaf Manela, William Mullins, and Luka Vulicevic. 2025. "Entity Neutering." *Available at SSRN* .

Eriksson, Katherine, Gregory Niemesh, Myera Rashid, and Jacqueline Craig. 2026. "Marriage and the Intergenerational Mobility of Women: Evidence from Marriage Certificates 1850-1920." .

Fan, Jianqing, Yihong Gu, and Ximing Li. 2025. "Optimal Estimation of a Factorizable Density Using Diffusion Models with ReLU Neural Networks." *arXiv preprint arXiv:2510.03994* .

Farrell, Max H, Tengyuan Liang, and Sanjog Misra. 2021a. "Deep Learning for Individual Heterogeneity: An Automatic Inference Framework." .

———. 2021b. "Deep Neural Networks for Estimation and Inference." *Econometrica* 89 (1):181–213.

Fellegi, Ivan P and Alan B Sunter. 1969. "A Theory for Record Linkage." *Journal of the American Statistical Association* 64 (328):1183–1210.

Fernández-Villaverde, Jesús, Galo Nuño, and Jesse Perla. 2024. "Taming the Curse of Dimensionality: Quantitative Economics with Deep Learning." .

Fernández-Villaverde, Jesús, Samuel Hurtado, and Galo Nuño. 2023. "Financial Frictions and the Wealth Distribution." *Econometrica* 91 (3):869–901.

Froese, Vincent and Christoph Hertrich. 2023. "Training Neural Networks is NP-Hard in Fixed Dimension." *Advances in Neural Information Processing Systems* 36:44039–44049.

Fudenberg, Drew and Annie Liang. 2019. "Predicting and Understanding Initial Play." *American Economic Review* 109 (12):4112–4141.

Fukushima, Kunihiko. 1975. "Cognitron: A Self-Organizing Multilayered Neural Network." *Biological Cybernetics* 20 (3):121–136.

Gal, Yarin and Zoubin Ghahramani. 2016. "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning." In *International Conference on Machine Learning*. PMLR, 1050–1059.

Gallant, A Ronald and George Tauchen. 1996. "Which Moments to Match?" *Econometric Theory* 12 (4):657–681.

Gallant, A Ronald and Halbert White. 1992. "On Learning the Derivatives of an Unknown Mapping with Multilayer Feedforward Networks." *Neural Networks* 5 (1):129–138.

Gallant, Ronald and Jonathan R Long. 1997. "Estimating Stochastic Differential Equations Efficiently by Minimum Chi-Squared." *Biometrika* 84 (1):125–141.

Gao, Wayne, Sukjin Han, and Annie Liang. 2026. "How Well Do LLMs Predict Human Behavior? A Measure of their Pretrained Knowledge." *Working Paper* .

Gao, Yuanbo, Baobin Li, Ning Wang, and Tingshao Zhu. 2017. "Speech Emotion Recognition Using Local and Global Features." In *Brain Informatics: International Conference, BI 2017, Beijing, China, November 16-18, 2017, Proceedings*. Springer, 3–13.

Gelly, Sylvain and David Silver. 2008. "Achieving Master Level Play in 9 x 9 Computer Go." In *AAAI*, vol. 8. 1537–1540.

Gentzkow, Matthew, Bryan Kelly, and Matt Taddy. 2019. "Text as Data." *Journal of Economic Literature* 57 (3):535–574.

Ghosh, Tilottama, Sharolyn Anderson, Rebecca L Powell, Paul C Sutton, and Christopher D Elvidge. 2009. "Estimation of Mexico's Informal Economy and Remittances Using Nighttime Imagery." *Remote Sensing* 1 (3):418–444.

Ghosh, Tilottama, Rebecca L Powell, Christopher D Elvidge, Kimberly E Baugh, Paul C Sutton, and Sharolyn Anderson. 2010. "Shedding Light on the Global Distribution of Economic Activity." *The Open Geography Journal* 3 (1):147–160.

Glasserman, Paul and Caden Lin. 2023. "Assessing Look-Ahead Bias in Stock Return Predictions Generated by GPT Sentiment Analysis." *arXiv preprint arXiv:2309.17322* .

Glorot, Xavier and Yoshua Bengio. 2010. "Understanding the Difficulty of Training Deep Feedforward Neural Networks." In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 249–256.

Glorot, Xavier, Antoine Bordes, and Yoshua Bengio. 2011. "Deep Sparse Rectifier Neural Networks." In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 315–323.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. "Generative Adversarial Networks." *Communications of the ACM* 63 (11):139–144.

Goodfellow, Ian J, Jonathon Shlens, and Christian Szegedy. 2014. "Explaining and Harnessing Adversarial Examples." *arXiv preprint arXiv:1412.6572* .

Gorodnichenko, Yuriy, Tho Pham, and Oleksandr Talavera. 2023. "The Voice of Monetary Policy." *American Economic Review* 113 (2):548–584.

Gottschling, Andreas Peter. 1997. *Three essays in neural networks and financial prediction*. University of California, San Diego.

Gowrisankaran, Gautam and Marc Rysman. 2012. "Dynamics of Consumer Demand for New Durable Goods." *Journal of Political Economy* 120 (6):1173–1219.

Graham, Bryan S, Cristine Campos de Xavier Pinto, and Daniel Egel. 2016. "Efficient Estimation of Data Combination Models by the Method of Auxiliary-to-Study Tilting (AST)." *Journal of Business & Economic Statistics* 34 (2):288–301.

Graves, Alex. 2011. "Practical Variational Inference for Neural Networks." *Advances in Neural Information Processing Systems* 24.

Greff, Klaus, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. 2016. "LSTM: A Search Space Odyssey." *IEEE Transactions on Neural Networks and Learning Systems* 28 (10):2222–2232.

Gu, Albert and Tri Dao. 2023. "Mamba: Linear-Time Sequence Modeling with Selective State Spaces." *arXiv preprint arXiv:2312.00752* .

Gu, Albert, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. 2020. "HiPPO: Recurrent Memory with Optimal Polynomial Projections." *Advances in Neural Information Processing Systems* 33:1474–1487.

Gu, Albert, Karan Goel, and Christopher Ré. 2021. "Efficiently Modeling Long Sequences with Structured State Spaces." *arXiv preprint arXiv:2111.00396* .

Guenter, Brian. 2007. "Efficient Symbolic Differentiation for Graphics Applications." In *ACM SIGGRAPH 2007 papers*. 108–es.

Gulrajani, Ishaan, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. "Improved Training of Wasserstein GANs." *Advances in neural information processing systems* 30.

Guo, Daya, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi et al. 2025. "Deepseek-r1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning." *arXiv preprint arXiv:2501.12948* .

Hall, Ola, Francis Dompae, Ibrahim Wahab, and Fred Mawunyo Dzanku. 2023. "A Review of Machine Learning and Satellite Imagery for Poverty Prediction: Implications for Development Research and Applications." *Journal of International Development* 35 (7):1753–1768.

Han, Changho. 1999. *Multi-variate estimation and forecasting with artificial neural networks*. University of California, San Diego.

Han, Jiequn, Yingzhou Li, Lin Lin, Jianfeng Lu, and Linfeng Zhang. 2022. "Universal Approximation of Symmetric and Anti-symmetric Functions." *Communications in Mathematical Sciences* 20 (5):1397–1408.

Han, Jiequn, Yucheng Yang, and Weinan E. 2026. "DeepHAM: A Global Solution Method For Heterogeneous Agent Models With Aggregate Shocks." *Quantitative Economics* 17 (2):297–341.

Han, Sukjin and Kyungho Lee. 2025. "Copyright and Competition: Estimating Supply and Demand with Unstructured Data." *Working Paper* .

Han, Sukjin, Eric H Schulman, Kristen Grauman, and Santhosh Ramakrishnan. 2021. "Shapes as Product Differentiation: Neural Network Embedding in the Analysis of Markets for Fonts." *arXiv preprint arXiv:2107.02739* .

Handlan, Amy and Haoyu Sheng. 2023. *Gender and Tone in Recorded Economics Presentations: Audio Analysis with Machine Learning*.

Hansen, Anne Lundgaard, John J Horton, Sophia Kazinnik, Daniela Puzzello, and Ali Zarifhonarvar. 2026. "Simulating the Survey of Professional Forecasters." *Working Paper* .

Hansen, Lars Peter. 1982. "Large Sample Properties of Generalized Method of Moments Estimators." *Econometrica: Journal of the Econometric Society* :1029–1054.

Hansen, Stephen, Peter John Lambert, Nicholas Bloom, Steven J Davis, Raffaella Sadun, and Bledi Taska. 2023. "Remote Work Across Jobs, Companies, and Space." .

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification." In *Proceedings of the IEEE International Conference on Computer Vision*. 1026–1034.

———. 2016. "Deep Residual Learning for Image Recognition." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.

He, Songrun, Linying Lv, Asaf Manela, and Jimmy Wu. 2025. "Chronologically Consistent Large Language Models." *arXiv preprint arXiv:2502.21206* .

Hebb, DO. 1949. "The Organization of Behavior. A Neuropsychological Theory." .

Hendel, Igal and Aviv Nevo. 2006. "Measuring the Implications of Sales and Consumer Inventory Behavior." *Econometrica* 74 (6):1637–1673.

Henderson, J Vernon, Adam Storeygard, and David N Weil. 2012. "Measuring Economic Growth from Outer Space." *American Economic Review* 102 (2):994–1028.

Hendrycks, Dan and Kevin Gimpel. 2016. "Gaussian Error Linear Units (GELUs)." *arXiv preprint arXiv:1606.08415* .

Herbrich, Ralf, Max Keilbach, Thore Graepel, Peter Bollmann-Sdorra, and Klaus Obermayer. 1999. "Neural Networks in Economics: Background, Applications and New Developments." In *Computational Techniques for Modelling Learning in Economics*. Springer, 169–196.

Hernández-Lobato, José Miguel and Ryan Adams. 2015. "Probabilistic Backpropagation for Scalable Learning of Bayesian Neural Networks." In *International Conference on Machine Learning*. PMLR, 1861–1869.

Hinton, Geoffrey, Niash Srivastava, and Kevin Swersky. 2012. "Lecture 6.5-RMSprop: Divide the Gradient by a Running Average of Its Recent Magnitude." *COURSERA: Neural Networks for Machine Learning* 4 (2):26.

Hinton, Geoffrey E and Ruslan R Salakhutdinov. 2006. "Reducing the Dimensionality of Data with Neural Networks." *Science* 313 (5786):504–507.

Ho, Jonathan, Ajay Jain, and Pieter Abbeel. 2020. "Denoising Diffusion Probabilistic Models." *Advances in Neural Information Processing Systems* 33:6840–6851.

Hochreiter, Sepp. 1991. "Untersuchungen zu Dynamischen Neuronalen Netzen." *Diploma, Technische Universität München* 91 (1):31.

Hochreiter, Sepp and Jürgen Schmidhuber. 1997. "Long Short-Term Memory." *Neural Computation* 9 (8):1735–1780.

Hopfield, John. 1982. "Neural Networks and Physical Systems with Emergent Collective Computational Abilities." *Proceedings of the National Academy of Sciences* 79 (8):2554–2558.

———. 1984. "Neurons with Graded Response have Collective Computational Properties like those of Two-State Neurons." *Proceedings of the National Academy of Sciences* 81 (10):3088–3092.

Hornik, Kurt, Maxwell Stinchcombe, and Halbert White. 1989. "Multilayer Feedforward Networks are Universal Approximators." *Neural Networks* 2:359–366.

———. 1990. "Universal Approximation of an Unknown Mapping and Its Derivatives Using Multilayer Feedforward Networks." *Neural Networks* 3 (5):551–560.

Hornik, Kurt, Maxwell Stinchcombe, Halbert White, and Peter Auer. 1994. "Degree of Approximation Results for Feedforward Networks Approximating Unknown Mappings and their Derivatives." *Neural Computation* 6 (6):1262–1275.

Horowitz, Joel L and Enno Mammen. 2007. "Rate-Optimal Estimation for a General Class of Nonparametric Regression Models with Unknown Link Functions." *The Annals of Statistics* :2589–2619.

Horton, John J. 2023a. "Large Language Models as Simulated Economic Agents: What Can We Learn From Homo Silicus?" .

———. 2023b. "Price Floors and Employer Preferences: Evidence from a Minimum Wage Experiment." *Working Paper* .

Hu, Allen and Song Ma. 2021. "Persuading Investors: A Video-Based Study." .

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. "Densely Connected Convolutional Networks." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4700–4708.

Huang, Luna Yue, Solomon M Hsiang, and Marco Gonzalez-Navarro. 2021. "Using Satellite Imagery and Deep Learning to Evaluate the Impact of Anti-Poverty Programs." .

Ifrach, Bar and Gabriel Y. Weintraub. 2016. "A Framework for Dynamic Oligopoly in Concentrated Industries." *The Review of Economic Studies* 84 (3):1106–1150.

Igami, Mitsuru. 2020. "Artificial Intelligence as Structural Estimation: Deep Blue, Bonanza, and AlphaGo." *The Econometrics Journal* 23 (3):S1–S24.

Ioffe, Sergey and Christian Szegedy. 2015. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift." In *International Conference on Machine Learning*. pmlr, 448–456.

Iskhakov, Fedor, John Rust, and Bertel Schjerning. 2020. "Machine Learning and Structural Econometrics: Contrasts and Synergies." *The Econometrics Journal* 23 (3):S81–S124.

Ismayilova, Aysu and Vugar E Ismailov. 2024. "On the Kolmogorov Neural Networks." *Neural Networks* 176:106333.

Ivakhnenko, Alexey and Valentin Lapa. 1966. "Cybernetic Predicting Devices." Tech. rep.

Jabbar, Abdul, Xi Li, and Bourahla Omar. 2021. "A Survey on Generative Adversarial Networks: Variants, Applications, and Training." *ACM Computing Surveys (CSUR)* 54 (8):1–49.

Jarrett, Kevin, Koray Kavukcuoglu, Marc'Aurelio Ranzato, and Yann LeCun. 2009. "What is the Best Multi-Stage Architecture for Object Recognition?" In *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2146–2153.

Jean, Neal, Marshall Burke, Michael Xie, W Matthew Davis, David B Lobell, and Stefano Ermon. 2016. "Combining Satellite Imagery and Machine Learning to Predict Poverty." *Science* 353 (6301):790–794.

Jean, Sébastien, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2014. "On Using Very Large Target Vocabulary for Neural Machine Translation." *arXiv preprint arXiv:1412.2007* .

Ji, Wenlong, Weizhe Yuan, Emily Getzen, Kyunghyun Cho, Michael I. Jordan, Song Mei, Jason E Weston, Weijie J. Su, Jing Xu, and Linjun Zhang. 2025. "An Overview of Large Language Models for Statisticians."

Joerding, Wayne and Ying Li. 1994. "Global Estimation of Feedforward Networks With A Priori Constraints." *Computational Economics* 7 (2):73–87.

Joerding, Wayne and Jack Meador. 1991. "Encoding A Priori Information in Feedforward Networks." *Neural Networks* 4 (6):847–856.

Johnson, Jeff, Matthijs Douze, and Hervé Jégou. 2019. "Billion-Scale Similarity Search with GPUs." *IEEE Transactions on Big Data* 7 (3):535–547.

Jordà, Òscar. 2005. "Estimation and Inference of Impulse Responses by Local Projections." *American Economic Review* 95 (1):161–182.

Kahneman, Daniel, Jack L Knetsch, and Richard Thaler. 1986. "Fairness as a Constraint on Profit Seeking: Entitlements in the Market." *American Economic Review* 76 (4):728–741.

Kahneman, Daniel, Jack L Knetsch, and Richard H Thaler. 1991. "Anomalies: The Endowment Effect, Loss Aversion, and Status Quo Bias." *Journal of Economic Perspectives* 5 (1):193–206.

Kahneman, Daniel and Amos Tversky. 1979. "Prospect Theory: An Analysis of Decision under Risk." *Econometrica* 47 (2):263–292.

Kaji, Tetsuya, Elena Manresa, and Guillaume Pouliot. 2023. "An Adversarial Approach to Structural Estimation." *Econometrica* 91 (6):2041–2063.

Kaji, Tetsuya, Elena Manresa, and Guillaume A Pouliot. 2021. "Adversarial Inference Is Efficient." In *AEA Papers and Proceedings*, vol. 111. 621–625.

Kamstra, Mark Jack. 1992. *A Collection of Papers in Econometric Theory*. University of California, San Diego.

Keskar, Nitish Shirish, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. 2017. "On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima." *International Conference on Learning Representations* .

Khemani, Bharti, Shruti Patil, Ketan Kotecha, and Sudeep Tanwar. 2024. "A Review of Graph Neural Networks: Concepts, Architectures, Techniques, Challenges, Datasets, Applications, and Future Directions." *Journal of Big Data* 11 (1):18.

Kim, H. and J.-S. Lee. 2024. "Scalable monotonic neural networks." *The Twelfth International Conference on Learning Representations* URL https://openreview. net/forum?id=DjIsNDEOYX.

Kingma, Diederik P and Jimmy Ba. 2014. "Adam: A Method for Stochastic Optimization." *arXiv preprint arXiv:1412.6980* .

Kingma, Diederik P and Max Welling. 2013. "Auto-Encoding Variational Bayes." *arXiv preprint arXiv:1312.6114* .

Kiran, B Ravi, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A Al Sallab, Senthil Yogamani, and Patrick Pérez. 2021. "Deep Reinforcement Learning for Autonomous Driving: A Survey." *IEEE Transactions on Intelligent Transportation Systems* 23 (6):4909–4926.

Klambauer, Günter, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. 2017. "Self-Normalizing Neural Networks." *Advances in Neural Information Processing Systems* 31.

Koffi, Marlène and Matt Marx. 2023. "Cassatts in the Attic." .

Kohler, Michael and Sophie Langer. 2021. "On the rate of convergence of fully connected deep neural network regression estimates." *Ann. Stat.* 49:2231–2249.

Koppelman, Frank S and Chieh-Hua Wen. 2000. "The Paired Combinatorial Logit Model: Properties, Estimation and Application." *Transportation Research Part B: Methodological* 34 (2):75–89.

Korinek, Anton. 2023. "Generative AI for Economic Research: Use Cases and Implications for Economists." *Journal of Economic Literature* 61 (4):1281–1317.

———. 2024a. "LLMs Learn to Collaborate and Reason: December 2024 Update to 'Generative AI for Economic Research: Use Cases and Implications for Economists', published in the Journal of Economic 61 (4)." .

———. 2024b. "LLMs Level Up—Better, Faster, Cheaper: June 2024 Update to Section 3 of 'Generative AI for Economic Research: Use Cases and Implications for Economists', published in the Journal of Economic 61 (4)." .

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. 2012. "Imagenet Classification with Deep Convolutional Neural Networks." *Advances in Neural Information Processing Systems* 25.

Krusell, Per and Anthony A Smith, Jr. 1998. "Income and Wealth Heterogeneity in the Macroeconomy." *Journal of Political Economy* 106 (5):867–896.

Kuan, Chung-Ming. 1989. *Estimation of Neural Network Models*. University of California, San Diego.

Kuan, Chung-Ming and Halbert White. 1994. "Artificial Neural Networks: An Econometric Perspective." *Econometric Reviews* 13 (1):1–91.

Kuhn, Thomas S. 1962. *The Structure of Scientific Revolutions*. University of Chicago Press.

Lai, Zehua, Lek-Heng Lim, and Yucong Liu. 2024. "Attention is a Smoothed Cubic Spline." *arXiv preprint arXiv:2408.09624* .

Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell. 2017. "Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles." *Advances in neural information processing systems* 30.

Lan, Zhenzhong, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. "ALBERT: A Lite BERT for Self-Supervised Learning of Language Representations." *arXiv preprint arXiv:1909.11942* .

Lapakko, David. 2007. "Communication is 93% Nonverbal: An Urban Legend Proliferates." *Communication and Theater Association of Minnesota Journal* 34 (1):2.

Laue, Soeren. 2019. "On the Equivalence of Automatic and Symbolic Differentiation." *arXiv preprint arXiv:1904.02990* .

LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. "Gradient-Based Learning Applied to Document Recognition." *Proceedings of the IEEE* 86 (11):2278–2324.

Lee, Lung-fei and Jungsywan H Sepanski. 1995. "Estimation of Linear and Nonlinear Errors-in-Variables Models Using Validation Data." *Journal of the American Statistical Association* 90 (429):130–140.

Lee, Tae-Hwy, Halbert White, and Clive WJ Granger. 1993. "Testing for Neglected Nonlinearity in Time Series Models: A Comparison of Neural Network Methods and Alternative Tests." *Journal of Econometrics* 56 (3):269–290.

Leshno, Moshe, Vladimir Ya Lin, Allan Pinkus, and Shimon Schocken. 1993. "Multilayer Feedforward Networks with a Nonpolynomial Activation Function Can Approximate Any Function." *Neural Networks* 6 (6):861–867.

Leung, Michael P. 2022. "Causal Inference under Approximate Neighborhood Interference." *Econometrica* 90 (1):267–293.

Leung, Michael P. and Pantelis Loupos. 2024. "Graph Neural Networks for Causal Inference under Network Confounding." *arXiv preprint arXiv:2211.07823* .

Lewis, Greg and Vasilis Syrgkanis. 2018. "Adversarial Generalized Method of Moments." *arXiv preprint arXiv:1803.07164* .

Lewis, Mike, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. "BART: Denoising Sequence-to-Sequence Pre-Training for Natural Language Generation, Translation, and Comprehension." *arXiv preprint arXiv:1910.13461* .

Li, Liangyue, How Jing, Hanghang Tong, Jaewon Yang, Qi He, and Bee-Chung Chen. 2017. "NEMO: Next Career Move Prediction with Contextual Embedding." In *Proceedings of the 26th International Conference on World Wide Web Companion*. 505–513.

Li, Yuliang, Jinfeng Li, Yoshi Suhara, AnHai Doan, and Wang-Chiew Tan. 2023. "Effective Entity Matching with Transformers." *The VLDB Journal* 32 (6):1215–1235.

Li, Zihao, Hui Yuan, Kaixuan Huang, Chengzhuo Ni, Yinyu Ye, Minshuo Chen, and Mengdi Wang. 2024. "Diffusion Model for Data-Driven Black-Box Optimization."

Liang, Tengyuan. 2021. "How Well Generative Adversarial Networks Learn Distributions." *Journal of Machine Learning Research* 22 (228):1–41.

Liang, Tengyuan, Kulunu Dharmakeerthi, and Takuya Koriyama. 2024. "Denoising Diffusions with Optimal Transport: Localization, Curvature, and Multi-Scale Complexity." *arXiv preprint arXiv:2411.01629* .

Liem, Cynthia CS, Markus Langer, Andrew Demetriou, Annemarie MF Hiemstra, Achmadnoer Sukma Wicaksana, Marise Ph Born, and Cornelius J König. 2018. "Psychology Meets Machine Learning: Interdisciplinary Perspectives on Algorithmic Job Candidate Screening." *Explainable and Interpretable Models in Computer Vision and Machine Learning* :197–253.

Likitha, MS, Sri Raksha R Gupta, K Hasitha, and A Upendra Raju. 2017. "Speech Based Human Emotion Recognition Using MFCC." In *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*. IEEE, 2257–2260.

Lin, Chien-Fu Jeff. 1992. *The Econometrics of Structural Change, Neural Network and Panel Data Analysis*. University of California, San Diego.

Lin, Min, Qiang Chen, and Shuicheng Yan. 2013. "Network In Network." *arXiv preprint arXiv:1312.4400* .

Linnainmaa, Seppo. 1970. "Algoritmin kumulatiivinen pyöristysvirhe yksittäisten pyöristysvirheiden Taylor-kehitelmänä." *Finnish. University of Helsinki* .

Liu, Ziming, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y. Hou, and Max Tegmark. 2025. "KAN: Kolmogorov-Arnold Networks."

Livingstone, Steven R and Frank A Russo. 2018. "The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A Dynamic, Multimodal Set of Facial and Vocal Expressions in North American English." *PloS one* 13 (5):e0196391.

Loaiza-Ganem, Gabriel, Valentin Villecroze, and Yixin Wang. 2025. "Deep Ensembles Secretly Perform Empirical Bayes." *arXiv preprint arXiv:2501.17917* .

Loaiza-Maya, Rubén and Didier Nibbering. 2023. "Fast Variational Bayes Methods for Multinomial Probit Models." *Journal of Business & Economic Statistics* 41 (4):1352–1363.

Lopez-Lira, Alejandro, Yuehua Tang, and Mingyin Zhu. 2025. "The Memorization Problem: Can We Trust LLMs' Economic Forecasts?" *arXiv preprint arXiv:2504.14765* .

Loshchilov, Ilya and Frank Hutter. 2017. "Decoupled Weight Decay Regularization." *arXiv preprint arXiv:1711.05101* .

Lu, Jianfeng, Zuowei Shen, Haizhao Yang, and Shijun Zhang. 2021. "Deep network approximation for smooth functions." *SIAM J. Math. Anal.* 53:5465–5506.

Ludwig, Jens and Sendhil Mullainathan. 2022. "Algorithmic Behavioral Science: Machine Learning as a Tool for Scientific Discovery." *Chicago Booth Research Paper* (22-15).

Ludwig, Jens, Sendhil Mullainathan, and Ashesh Rambachan. 2025. "Large Language Models: An Applied Econometric Framework." .

Maas, Andrew L, Awni Y Hannun, Andrew Y Ng et al. 2013. "Rectifier Nonlinearities Improve Neural Network Acoustic Models." In *Proc. ICML*, vol. 30. Atlanta, GA, 3.

Maclaurin, Dougal. 2016. *Modeling, Inference and Optimization with Composable Differentiable Procedures*. Ph.D. thesis, Harvard University.

Magnolfi, Lorenzo, Jonathon McClure, and Alan T Sorensen. 2022. *Triplet Embeddings for Demand Estimation*. SSRN.

Makovoz, Y. 1996. "Random approximants and neural networks." *Journal of Approximation Theory* 85:98–109.

Manski, Charles. 1975. "Maximum Score Estimation of the Stochastic Utility Model of Choice." *Journal of Econometrics* 3 (3):205–228.

Manski, Charles F. 1985. "Semiparametric Analysis of Discrete Response: Asymptotic Properties of the Maximum Score Estimator." *Journal of Econometrics* 27 (3):313–333.

———. 1987. "Semiparametric Analysis of Random Effects Linear Models from Binary Panel Data." *Econometrica: Journal of the Econometric Society* :357–362.

Matzkin, Rosa L. 1994. "Restrictions of Economic Theory in Nonparametric Methods." *Handbook of Econometrics* 4:2523–2558.

McCulloch, Warren S and Walter Pitts. 1943. "A Logical Calculus of the Ideas Immanent in Nervous Activity." *The Bulletin of Mathematical Biophysics* 5:115–133.

McFadden, Daniel. 1989. "A Method of Simulated Moments for Estimation of Discrete Response Models without Numerical Integration." *Econometrica: Journal of the Econometric Society* :995–1026.

———. 2001. "Economic Choices." *The American Economic Review* :351–378.

Mehrabian, Albert. 1971. "Implicit Communication of Emotions and Attitudes."

Mehrabian, Albert and Susan R Ferris. 1967. "Inference of Attitudes from Nonverbal Communication in two Channels." *Journal of Consulting Psychology* 31 (3):248.

Mehrabian, Albert and Morton Wiener. 1967. "Decoding of Inconsistent Communications." *Journal of Personality and Social Psychology* 6 (1):109.

Mehrabian, Ali, Ehsan Hoseinzade, Mahdi Mazloum, and Xiaohong Chen. 2025. "Mamba Meets Financial Markets: A Graph-Mamba Approach for Stock Price Prediction." *ICASSP* .

Mele, Angelo and Lingjiong Zhu. 2023. "Approximate Variational Estimation for a Model of Network Formation." *Review of Economics and Statistics* 105 (1):113–124.

Menzel, Konrad. 2021. "Structural Sieves." *arXiv preprint arXiv:2112.01377* .

Mihalcea, Rada and Paul Tarau. 2004. "TextRank: Bringing Order into Text." In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. 404–411.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. "Efficient Estimation of Word Representations in Vector Space." *arXiv preprint arXiv:1301.3781* .

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. "Distributed Representations of Words and Phrases and Their Compositionality." *Advances in Neural Information Processing Systems* 26.

Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. "Playing Atari with Deep Reinforcement Learning." *arXiv preprint arXiv:1312.5602* .

Morgan, Nelson and Hervé Bourlard. 1989. "Generalization and Parameter Estimation in Feedforward Nets: Some Experiments." *Advances in Neural Information Processing Systems* 2.

Mudgal, Sidharth, Han Li, Theodoros Rekatsinas, AnHai Doan, Youngchoon Park, Ganesh Krishnan, Rohit Deep, Esteban Arcaute, and Vijay Raghavendra. 2018. "Deep Learning for Entity Matching: A Design Space Exploration." In *Proceedings of the 2018 International Conference on Management of Data*. 19–34.

Mullainathan, Sendhil and Ashesh Rambachan. 2024. "From Predictive Algorithms to Automatic Generation of Anomalies." .

Mullainathan, Sendhil and Jann Spiess. 2017. "Machine Learning: An Applied Econometric Approach." *Journal of Economic Perspectives* 31 (2):87–106.

Murty, Katta G and Santosh N Kabadi. 1987. "Some NP-Complete Problems in Quadratic and Nonlinear Programming." *Mathematical Programming* 39 (2):117–129.

Nagel, Rosemarie. 1995. "Unraveling in Guessing Games: An Experimental Study." *American Economic Review* 85 (5):1313–1326.

Nair, Vinod and Geoffrey E Hinton. 2010. "Rectified Linear Units Improve Restricted Boltzmann Machines." In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. 807–814.

Nesterov, Yurii Evgen'evich. 1983. "A Method of Solving a Convex Programming Problem with Convergence Rate O(k^2)." In *Doklady Akademii Nauk*, vol. 269. Russian Academy of Sciences, 543–547.

Newey, Whitney K. 1990. "Efficient Instrumental Variables Estimation of Nonlinear Models." *Econometrica: Journal of the Econometric Society* :809–837.

Newey, Whitney K and James L Powell. 2003. "Instrumental Variable Estimation of Nonparametric Models." *Econometrica* 71 (5):1565–1578.

Newey, W.K. 1994. "The Asymptotic Variance of Semiparametric Estimators." *Econometrica* 62 (6):1349–1382.

Nosratabadi, Saeed, Amirhosein Mosavi, Puhong Duan, Pedram Ghamisi, Ferdinand Filip, Shahab S Band, Uwe Reuter, Joao Gama, and Amir H Gandomi. 2020. "Data Science in Economics: Comprehensive Review of Advanced Machine Learning and Deep Learning Methods." *Mathematics* 8 (10):1799.

Oh, Kyoung-Su and Keechul Jung. 2004. "GPU Implementation of Neural Networks." *Pattern Recognition* 37 (6):1311–1314.

Orvieto, Antonio, Samuel L Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. 2023. "Resurrecting Recurrent Neural Networks for Long Sequences." In *International Conference on Machine Learning*. PMLR, 26670–26698.

Ouyang, Long, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. "Training Language Models to Follow Instructions with Human Feedback." In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22. Red Hook, NY, USA: Curran Associates Inc.

Pakes, Ariel and David Pollard. 1989. "Simulation and the Asymptotics of Optimization Estimators." *Econometrica: Journal of the Econometric Society* :1027–1057.

Pan, Yixiong, Peipei Shen, and Liping Shen. 2012. "Speech Emotion Recognition Using Support Vector Machine." *International Journal of Smart Home* 6 (2):101–108.

Park, Joon Sung, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. "Generative Agents: Interactive Simulacra of Human Behavior." In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–22.

Paszke, Adam, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. "Automatic Differentiation in PyTorch." *NIPS 2017 Autodiff Workshop* .

Peeters, Ralph and Christian Bizer. 2021. "Dual-Objective Fine-Tuning of BERT for Entity Matching." *Proceedings of the VLDB Endowment* 14:1913–1921.

———. 2023. "Using ChatGPT for Entity Matching." In *European Conference on Advances in Databases and Information Systems*. Springer, 221–230.

Peterson, Joshua C, David D Bourgin, Mayank Agrawal, Daniel Reichman, and Thomas L Griffiths. 2021. "Using Large-Scale Experiments and Machine Learning to Discover Theories of Human Decision-Making." *Science* 372 (6547):1209–1214.

Plutowski, Mark and Halbert White. 1993. "Selecting Concise Training Sets from Clean Data." *IEEE Transactions on Neural Networks* 4 (2):305–318.

Pollmann, Michael. 2020. "Causal Inference for Spatial Treatments." *arXiv preprint arXiv:2011.00373* .

Polyak, Boris T. 1964. "Some Methods of Speeding up the Convergence of Iteration Methods." *USSR Computational Mathematics and Mathematical Physics* 4 (5):1–17.

Prentice, Ross L. 1989. "Surrogate Endpoints in Clinical Trials: Definition and Operational Criteria." *Statistics in Medicine* 8 (4):431–440.

Primpeli, Anna, Ralph Peeters, and Christian Bizer. 2019. "The WDC Training Dataset and Gold Standard for Large-Scale Product Matching." In *Companion Proceedings of The 2019 World Wide Web Conference*. 381–386.

Quass, Dallan and Paul Starkey. 2003. "A Comparison of Fast Blocking Methods for Record Linkage." In *of: Proceedings of the KDD 2003 Workshop on Data Cleaning, Record Linkage and Object Consolidation. Washington, DC, USA*. 40–42.

Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer." *Journal of Machine Learning Research* 21 (140):1–67.

Raissi, Maziar, Paris Perdikaris, and George E Karniadakis. 2019. "Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations." *Journal of Computational Physics* 378:686–707.

Ramachandran, Prajit, Barret Zoph, and Quoc V Le. 2017. "Searching for Activation Functions." *arXiv preprint arXiv:1710.05941* .

Rambachan, Ashesh, Rahul Singh, and Davide Viviano. 2024. "Program Evaluation with Remotely Sensed Outcomes." *arXiv preprint arXiv:2411.10959* .

Ramesh, Aditya, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. "Hierarchical Text-Conditional Image Generation with CLIP Latents."

Reimers, Nils and Iryna Gurevych. 2019. "Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks." *arXiv preprint arXiv:1908.10084* .

Rezende, Danilo Jimenez, Shakir Mohamed, and Daan Wierstra. 2014. "Stochastic Backpropagation and Approximate Inference in Deep Generative Models." In *International Conference on Machine Learning*. PMLR, 1278–1286.

Ritter, Hippolyt, Aleksandar Botev, and David Barber. 2018. "A Scalable Laplace Approximation for Neural Networks." In *International Conference on Learning Representations*.

Robbins, Herbert and Sutton Monro. 1951. "A Stochastic Approximation Method." *The Annals of Mathematical Statistics* :400–407.

Rombach, Robin, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. "High-Resolution Image Synthesis with Latent Diffusion Models." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10684–10695.

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

Ruan, Xinrui, Xinwei Ma, Yingfei Wang, Waverly Wei, and Jingshen Wang. 2025. "Can language models boost the power of randomized experiments without statistical bias?" *arXiv:2510.05545* .

Ruiz, Francisco JR, Susan Athey, and David M Blei. 2020. "SHOPPER." *The Annals of Applied Statistics* 14 (1):1–27.

Rumelhart, David E, Geoffrey E Hinton, and Ronald J Williams. 1986. "Learning Representations by Back-Propagating Errors." *nature* 323 (6088):533–536.

Salton, Gerard, Anita Wong, and Chung-Shu Yang. 1975. "A Vector Space Model for Automatic Indexing." *Communications of the ACM* 18 (11):613–620.

Samuelson, William and Richard Zeckhauser. 1988. "Status Quo Bias in Decision Making." *Journal of Risk and Uncertainty* 1:7–59.

Sanh, Victor, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. "DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter." *arXiv preprint arXiv:1910.01108* .

Saremi, Saeed and Aapo Hyvärinen. 2019. "Neural Empirical Bayes." *Journal of Machine Learning Research* 20 (181):1–23.

Sarkar, Suproteem K. 2024. "StoriesLM: A Family of Language Models with Time-Indexed Training Data." *Available at SSRN 4881024* .

Sarkar, Suproteem K and Keyon Vafa. 2024. "Lookahead Bias in Pretrained Language Models." In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*.

Scarselli, Franco, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2008. "The Graph Neural Network Model." *IEEE transactions on neural networks* 20 (1):61–80.

Schennach, Susanne M. 2016. "Recent Advances in the Measurement Error Literature." *Annual Review of Economics* 8 (1):341–377.

Schmidhuber, Jürgen. 2015. "Deep Learning in Neural Networks: An Overview." *Neural Networks* 61:85–117.

Schmidt-Hieber, J. 2020. "Nonparametric Regression Using Deep Neural Networks With Relu Activation Function." *Annals of Statistics* 137:119–126.

———. 2021. "The Kolmogorov–Arnold representation theorem revisited." *Neural Networks* 137:119–126.

Schroff, Florian, Dmitry Kalenichenko, and James Philbin. 2015. "FaceNet: A Unified Embedding for Face Recognition and Clustering." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 815–823.

Schölkopf, Bernhard, Chris Burges, and Vladimir Vapnik. 1995. "Extracting Support Data for a Given Task." In *Proceedings, First International Conference on Knowledge Discovery & Data Mining publisher=AAAI Press, Menlo Park, CA*. 252–257.

Sejnowski, Terrence J. 2018. *The Deep Learning Revolution*. MIT press.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016. "Neural Machine Translation of Rare Words with Subword Units." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1715–1725.

Shah, Agam, Suvan Paturi, and Sudheer Chava. 2023. "Trillion Dollar Words: A New Financial Dataset, Task & Market Analysis." *arXiv preprint arXiv:2305.07972* .

Shen, X. and W.H. Wong. 1994. "Convergence rates of sieve estimates." *Annals of Statistics* 22 (3):580–615.

Shen, Xiaotong. 1997. "On Methods of Sieves and Penalization." *The Annals of Statistics* 25 (6):2555–2591.

Shen, Zejiang, Ruochen Zhang, Melissa Dell, Benjamin Charles Germain Lee, Jacob Carlson, and Weining Li. 2021. "LayoutParser: A Unified Toolkit for Deep Learning Based Document Image Analysis." In *Document Analysis and Recognition–ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part I 16*. Springer, 131–146.

Siegel, Jonathan W. 2023. "Optimal Approximation Rates for Deep ReLU Neural Networks on Sobolev and Besov Spaces." *J. Mach. Learn. Res.* 24 (1).

Silver, David, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot et al. 2016. "Mastering the Game of Go with Deep Neural Networks and Tree Search." *Nature* 529 (7587):484–489.

Silver, David, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel et al. 2018. "A General Reinforcement Learning Algorithm that Masters Chess, Shogi, and Go through Self-Play." *Science* 362 (6419):1140–1144.

Simon, Jamie, Daniel Kunin, Alexander Atanasov, Enric Boix-Adserà, Blake Bordelon, Jeremy Cohen, Nikhil Ghosh, Florentin Guth, Arthur Jacot, Mason Kamb, Dhruva Karkada, Eric Michaud, Berkan Ottlik, and Joseph Turnbull. 2026. "There Will Be a Scientific Theory of Deep Learning." *Working Paper* .

Simonyan, Karen and Andrew Zisserman. 2014. "Very Deep Convolutional Networks for Large-Scale Image Recognition." *arXiv preprint arXiv:1409.1556* .

Singer, Uriel, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. 2022. "Make-A-Video: Text-to-Video Generation without Text-Video Data."

Small, Kenneth A. 1987. "A Discrete Choice Model for Ordered Alternatives." *Econometrica: Journal of the Econometric Society* :409–424.

Sohl-Dickstein, Jascha, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. "Deep Unsupervised Learning Using Nonequilibrium Thermodynamics." In *International Conference on Machine Learning*. PMLR, 2256–2265.

Springer, Max, Chung Lee, Blossom Metevier, Jane Castleman, Bohdan Turbal, Hayoung Jung, Zeyu Shen, and Aleksandra Korolova. 2026. "The Geometry of Alignment Collapse: When Fine-Tuning Breaks Safety." *arXiv:2602.15799* .

Spärck Jones, Karen. 1972. "A Statistical Interpretation of Term Specificity and its Application in Retrieval." *Journal of Documentation* 28 (1):11–21.

Srebrovic, Rob and Jay Yonamine. 2020. "Leveraging the BERT algorithm for Patents with TensorFlow and BigQuery." *Google White Paper* .

Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting." *The Journal of Machine Learning Research* 15 (1):1929–1958.

Stahl II, Dale O and Paul W Wilson. 1994. "Experimental Evidence on Players' Models of Other Players." *Journal of Economic Behavior & Organization* 25 (3):309–327.

Steinkraus, Dave, Ian Buck, and Patrice Y Simard. 2005. "Using GPUs for Machine Learning Algorithms." In *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*. IEEE, 1115–1120.

Stephanovitch, Arthur, Ugo Tanielian, Benoit Cadre, Nicolas Klutchnikoff, and Gerard Biau. 2024. "Optimal 1-Wasserstein Distance for WGANs." *Bernoulli* 30 (4):2955–2978.

Su, Weijie. 2025. "Do Large Language Models (Really) Need Statistical Foundations?"

Supreme Court of the United States. 2014. "Alice Corp. v. CLS Bank Int'l." 573 U.S. 208. Supreme Court decision.

Sutskever, I. 2014. "Sequence to Sequence Learning with Neural Networks." *arXiv preprint arXiv:1409.3215* .

Sutskever, Ilya, James Martens, George Dahl, and Geoffrey Hinton. 2013. "On the Importance of Initialization and Momentum in Deep Learning." In *International Conference on Machine Learning*. PMLR, 1139–1147.

Sutton, Paul C and Robert Costanza. 2002. "Global Estimates of Market and Non-Market Values Derived from Nighttime Satellite Imagery, Land Cover, and Ecosystem Service Valuation." *Ecological Economics* 41 (3):509–527.

Sutton, Richard S and Andrew G Barto. 2018. *Reinforcement Learning: An Introduction*. MIT Press.

Swanson, Eric T. 2021. "Measuring the Effects of Federal Reserve Forward Guidance and Asset Purchases on Financial Markets." *Journal of Monetary Economics* 118:32–53.

Swanson, Norman R and Halbert White. 1997. "A Model Selection Approach to Real-time Macroeconomic Forecasting using Linear Models and Artificial Neural Networks." *Review of Economics and Statistics* 79 (4):540–550.

Sweeting, Andrew. 2013. "Dynamic Product Positioning in Differentiated Product Markets: The Effect of Fees for Musical Performance Rights on the Commercial Radio Industry." *Econometrica* 81 (5):1763–1803.

Szasz, Teodora, Emileigh Harrison, Ping-Jung Liu, Ping-Chang Lin, Hakizumwami Birali Runesha, and Anjali Adukia. 2022. "Measuring Representation of Race, Gender, and Age in Children's Books: Face Detection and Feature Classification in Illustrated Images." In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 462–471.

Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. "Going Deeper with Convolutions." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1–9.

Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. "Rethinking the Inception Architecture for Computer Vision." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2818–2826.

Szegedy, Christian, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. "Intriguing Properties of Neural Networks." *arXiv preprint arXiv:1312.6199* .

Tan, Mingxing and Quoc Le. 2019. "Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks." In *International Conference on Machine Learning*. PMLR, 6105–6114.

Tay, Yi, Mostafa Dehghani, Vinh Q Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Siamak Shakeri, Dara Bahri, Tal Schuster et al. 2022. "UL2: Unifying Language Learning Paradigms." *arXiv preprint arXiv:2205.05131* .

Team, Kimi, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, Chuning Tang, Congcong Wang, Dehao Zhang, Enming Yuan, Enzhe Lu, Fengxiang Tang, Flood Sung, Guangda Wei, Guokun Lai, Haiqing Guo, Han Zhu, Hao Ding, Hao Hu, Hao Yang, Hao Zhang, Haotian Yao, Haotian Zhao, Haoyu Lu, Haoze Li, Haozhen Yu, Hongcheng Gao, Huabin Zheng, Huan Yuan, Jia Chen, Jianhang Guo, Jianlin Su, Jianzhou Wang, Jie Zhao, Jin Zhang, Jingyuan Liu, Junjie Yan, Junyan Wu, Lidong Shi, Ling Ye, Longhui Yu, Mengnan Dong, Neo Zhang, Ningchen Ma, Qiwei Pan, Qucheng Gong, Shaowei Liu, Shengling Ma, Shupeng Wei, Sihan Cao, Siying Huang, Tao Jiang, Weihao Gao, Weimin Xiong, Weiran He, Weixiao Huang, Wenhao Wu, Wenyang He, Xianghui Wei, Xianqing Jia, Xingzhe Wu, Xinran Xu, Xinxing Zu, Xinyu Zhou, Xuehai Pan, Y. Charles, Yang Li, Yangyang Hu, Yangyang Liu, Yanru Chen, Yejie Wang, Yibo Liu, Yidao Qin, Yifeng Liu, Ying Yang, Yiping Bao, Yulun Du, Yuxin Wu, Yuzhi Wang, Zaida Zhou, Zhaoji Wang, Zhaowei Li, Zhen Zhu, Zheng Zhang, Zhexu Wang, Zhilin Yang, Zhiqi Huang, Zihao Huang, Ziyao Xu, and Zonghan Yang. 2025. "Kimi k1.5: Scaling Reinforcement Learning with LLMs."

Tesauro, Gerald. 1995. "Temporal Difference Learning and TD-Gammon." *Communications of the ACM* 38 (3):58–68.

Tian, Xinyu and Xiaotong Shen. 2026. "Manifold-Aligned Generative Transport." *Working Paper* .

Tjuatja, Lindia, Valerie Chen, Tongshuang Wu, Ameet Talwalkwar, and Graham Neubig. 2024. "Do LLMs Exhibit Human-like Response Biases? A Case Study in Survey Design." *Transactions of the Association for Computational Linguistics* 12:1011–1026.

Touvron, Hugo, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton, Moya Chen Ferrer, Guillem Cucurull, David Esiobu, Jude Fernandes, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Sub-

ramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. "Llama 2: Open Foundation and Fine-Tuned Chat Models." *arXiv preprint arXiv:2307.09288* .

Turing, AM. 1950. "Computing Machinery and Intelligence." *Mind* 59:433–460.

Tversky, Amos and Daniel Kahneman. 1992. "Advances in Prospect Theory: Cumulative Representation of Uncertainty." *Journal of Risk and Uncertainty* 5:297–323.

Upasani, Siddhant, Noorul Amin, Sahil Damania, Ayush Jadhav, and AM Jagtap. 2020. "Automatic Summary Generation Using TextRank Based Extractive Text Summarization Technique." *International Research Journal of Engineering and Technology (IRJET) e-ISSN* :2395–0056.

USPTO, United States Patent & Trademark Office. 2019. "Progress and Potential: A Profile of Women Inventors on U.S. Patents." *Report* .

Vafa, Keyon, Emil Palikot, Tianyu Du, Ayush Kanodia, Susan Athey, and David Blei. 2024. "CAREER: A Foundation Model for Labor Sequence Data." *Transactions on Machine Learning Research* .

Varian, Hal R. 2014. "Big Data: New Tricks for Econometrics." *Journal of Economic Perspectives* 28 (2):3–28.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention Is All You Need." *Advances in Neural Information Processing Systems* 30.

Veitch, Victor, Dhanya Sridhar, and David Blei. 2020. "Adapting Text Embeddings for Causal Inference." In *Conference on Uncertainty in Artificial Intelligence*. PMLR, 919–928.

Veitch, Victor, Yixin Wang, and David Blei. 2019. "Using Embeddings to Correct for Unobserved Confounding in Networks." *Advances in Neural Information Processing Systems* 32.

Vincent, Pascal. 2011. "A Connection between Score Matching and Denoising Autoencoders." *Neural Computation* 23 (7):1661–1674.

Vinyals, Oriol, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, Junhyuk Oh, Dan Horgan, Manuel Kroiss, Ivo Danihelka, Aja Huang, Laurent Sifre, Trevor Cai, John P Agapiou, Max Jaderberg, Alexander S Vezhnevets, Rémi Leblond, Tobias Pohlen, Valentin Dalibard, David Budden, Yury Sulsky, James Molloy, Tom L Paine, Caglar Gulcehre, Ziyu Wang, Tobias Pfaff, Yuhuai Wu, Roman Ring, Dani Yogatama, Dario Wünsch, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy Lillicrap, Koray Kavukcuoglu, Demis Hassabis, Chris Apps, and David Silver. 2019. "Grandmaster Level in StarCraft II Using Multi-Agent Reinforcement Learning." *Nature* 575 (7782):350–354.

Von Neumann, John and Oskar Morgenstern. 1944. *Theory of Games and Economic Behavior*. Princeton University Press.

Von Thun, Friedemann Schulz. 1981. *Miteinander Reden 1: Störungen und Klärungen: Allgemeine Psychologie der Kommunikation*. Reinbek bei Hamburg.

Wang, Siruo, Tyler H McCormick, and Jeffrey T Leek. 2020. "Methods for Correcting Inference Based on Outcomes Predicted by Machine Learning." *Proceedings of the National Academy of Sciences* 117 (48):30266–30275.

Wang, Yike, Chris Gu, and Taisuke Otsu. 2024. "Graph Neural Networks: Theory for Estimation with Application on Network Heterogeneity." *arXiv preprint arXiv:2401.16275* .

Wang, Yixuan, Jonathan W Siegel, Ziming Liu, and Thomas Y Hou. 2025. "On the Expressiveness and Spectral Bias of KANs." *International Conference on Learning Representations* .

Wei, Yanhao and Zhenling Jiang. 2025. "Estimating Parameters of Structural Models Using Neural Networks." *Marketing Science* 44 (1):102–128.

Welling, Max and Yee W Teh. 2011. "Bayesian Learning via Stochastic Gradient Langevin Dynamics." In *Proceedings of the 28th international conference on machine learning (ICML-11)*. 681–688.

Wen, Xumeng, Zihan Liu, Shun Zheng, Shengyu Ye, Zhirong Wu, Yang Wang, Zhijian Xu, Xiao Liang, Junjie Li, Ziming Miao et al. 2025. "Reinforcement Learning with Verifiable Rewards Implicitly Incentivizes Correct Reasoning in Base LLMs." *arXiv preprint arXiv:2506.14245* .

Werbos, Paul. 1974. "Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences." *PhD Dissertation, Harvard University* .

———. 1982. "Applications of Advances in Nonlinear Sensitivity Analysis." In *Systems Modeling and Optimization: Proceedings of the 10th IFIP Conference*, edited by Drenick, R. and Kozin, F. New York: Springer-Verlag, 762–777.

White, Halbert. 1987. "Some Asymptotic Results for Back-Propagation." In *Proceedings of the First Annual IEEE Conference on Neural Networks*, vol. 3. 261–66.

———. 1988. "Economic Prediction using Neural Networks: The Case of IBM Daily Stock Returns." In *ICNN*, vol. 2. 451–458.

———. 1989. "Some Asymptotic Results for Learning in Single Hidden-Layer Feedforward Network Models." *Journal of the American Statistical Association* 84 (408):1003–1013.

White, Halbert and Jeffrey Wooldridge. 1991. "Some Results for Sieve Estimation with Dependent Observations." In *Nonparametric and Semiparametric Methods in Econometrics and Statistics*. Cambridge University Press.

Wilson, D Randall. 2011. "Beyond Probabilistic Record Linkage: Using Neural Networks and Complex Features to Improve Genealogical Record Linkage." In *The 2011 International Joint Conference on Neural Networks*. IEEE, 9–14.

Wilson, D Randall and Tony R Martinez. 2003. "The General Inefficiency of Batch Training for Gradient Descent Learning." *Neural Networks* 16 (10):1429–1451.

Wolpert, David H. 1996. "The Lack of A Priori Distinctions Between Learning Algorithms." *Neural Computation* 8 (7):1341–1390.

Wolpert, David H and William G Macready. 1997. "No Free Lunch Theorems for Optimization." *IEEE Transactions on Evolutionary Computation* 1 (1):67–82.

Wu, Jing Cynthia, Jin Xi, and Shihan Xie. 2025. "LLM Survey Framework: Coverage, Reasoning, Dynamics, Identification." *NBER Working Paper* .

Wu, Jing Cynthia and Fan Dora Xia. 2016. "Measuring the Macroeconomic Impact of Monetary Policy at the Zero Lower Bound." *Journal of Money, Credit and Banking* 48 (2-3):253–291.

Wu, Lingfei, Peng Cui, Jian Pei, Liang Zhao, and Xiaojie Guo. 2022. "Graph Neural Networks: Foundation, Frontiers and Applications." In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*. 4840–4841.

Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey et al. 2016. "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation." *arXiv preprint arXiv:1609.08144* .

Xie, Liyang, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. 2018. "Differentially Private Generative Adversarial Network." *arXiv preprint arXiv:1802.06739* .

Xin, Yi, Siqi Luo, Haodi Zhou, Junlong Du, Xiaohong Liu, Yue Fan, Qing Li, and Yuntao Du. 2024. "Parameter-Efficient Fine-Tuning for Pre-Trained Vision Models: A Survey." *arXiv preprint arXiv:2402.02242* .

Xu, Keyulu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. "How Powerful Are Graph Neural Networks?" *arXiv preprint arXiv:1810.00826* .

Yang, Greg. 2019. "Wide Feedforward or Recurrent Neural Networks of any Architecture are Gaussian Processes." *Advances in Neural Information Processing Systems* 32.

Yang, Ling, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. 2024. "Diffusion Models: A Comprehensive Survey of Methods and Applications."

Yang, Yunfei. 2025. "On the optimal approximation of Sobolev and Besov functions using deep ReLU neural networks." *Applied and Computational Harmonic Analysis* 79:101797.

Yeh, Christopher, Anthony Perez, Anne Driscoll, George Azzari, Zhongyi Tang, David Lobell, Stefano Ermon, and Marshall Burke. 2020. "Using Publicly Available Satellite Imagery and Deep Learning to Understand Economic Well-Being in Africa." *Nature Communications* 11 (1):2583.

You, Yang, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. 2019. "Large Batch Optimization for Deep Learning: Training BERT in 76 Minutes." *arXiv preprint arXiv:1904.00962* .

Zarifhonarvar, Ali. 2026. "Generating Inflation Expectations with Large Language Models." *Journal of Monetary Economics* 157:103859.

Zhang, Chiyuan, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2016. "Understanding Deep Learning Requires Rethinking Generalization." *arXiv preprint arXiv:1611.03530* .

Zhang, Denghui, Junming Liu, Hengshu Zhu, Yanchi Liu, Lichen Wang, Pengyang Wang, and Hui Xiong. 2019. "Job2Vec: Job Title Benchmarking with Collective Multi-View Representation Learning." In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2763–2771.

Zhang, Walter. 2023. "Optimal Comprehensible Targeting." *Working Paper* .

Zheng, Yuanhang, Zeshui Xu, and Anran Xiao. 2023. "Deep Learning in Economics: A Systematic and Critical Review." *Artificial Intelligence Review* 56 (9):9497–9539.

Zhong, Kai, Zhao Song, Prateek Jain, Peter L Bartlett, and Inderjit S Dhillon. 2017. "Recovery Guarantees for One-Hidden-Layer Neural Networks." In *International Conference on Machine Learning*. PMLR, 4140–4149.

Zhou, Ding-Xuan. 2020. "Universality of Deep Convolutional Neural Networks." *Applied and Computational Harmonic Analysis* 48 (2):787–794.

Zhu, Yuchen, Yufeng Zhang, Zhaoran Wang, Zhuoran Yang, and Xiaohong Chen. 2024. "A Mean-Field Analysis of Neural Stochastic Gradient Descent-Ascent for Functional Minimax Optimization." *Journal of Machine Learning Research* 25:1–62.

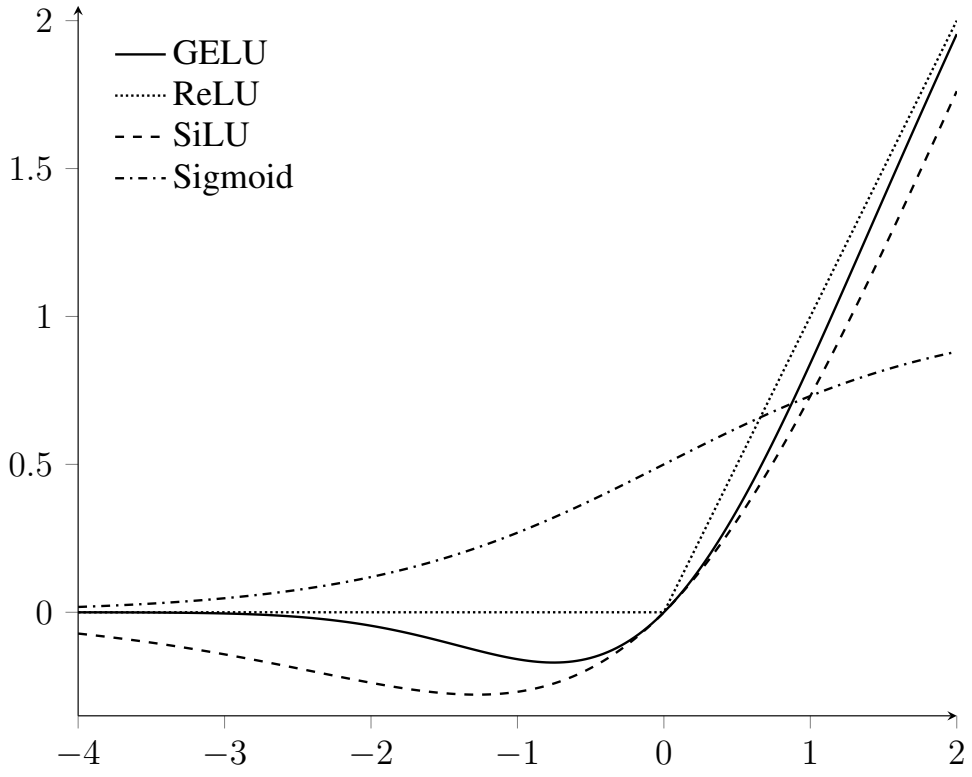
A On Activation Functions

Common choices are sigmoid activation, popularized by Rumelhart, Hinton, and Williams (1986), ReLU activation, due to Fukushima (1975); Jarrett et al. (2009); Nair and Hinton (2010); Glorot, Bordes, and Bengio (2011), the SiLU/Swish of Ramachandran, Zoph, and Le (2017), and Gaussian error linear units, defined as follows: of Hendrycks and Gimpel (2016)

$$
\begin{aligned}
f^{\text{sigmoid}}(z) &= 1/(1+\exp(-z)),\\
f^{\text{ReLU}}(z) &= \max\{0, z\},\\
f^{\text{SiLU}}(z) &= z f^{\text{sigmoid}}(z),\\
f^{\text{GELU}}(z) &= z\Phi(z).
\end{aligned}
$$

They are sometimes parametrized with additional learnable parameters. Other activation functions have been

Figure 3: Common Activation Functions



proposed, for example the hyperbolic tangent of LeCun et al. (1998), Leaky ReLU of Maas et al. (2013),

parametric ReLU of He et al. (2015), exponential linear units of Clevert (2015) and scaled exponential linear units of Klambauer et al. (2017). It is important that these functions be nonlinear.