Priors All the Way Down

Modern Machine Learning, From Foundations to Generative AI

A narrative summary of the book in progress

Kevin McAlister — Emory University — May 2026

A note to readers. This document is a narrative tour of a graduate textbook I am writing on modern machine learning. It is not the proposal and not the manuscript; it is the argument the book makes, in the order the book makes it, condensed enough that you can read it in an evening and tell me where I am wrong. The closing section names the questions I most want feedback on. I would rather have a sharp disagreement than a polite endorsement, and I am writing this with that in mind.

1. The frog problem

Open any introductory ML textbook to the chapter on logistic regression and you will find some version of the following exercise: train a binary classifier on CIFAR-10 to distinguish frogs from airplanes. The exercise is honest. Frogs and airplanes occupy genuinely different regions of color and texture space; even a linear classifier on raw pixels achieves something like 85–88% test accuracy. The student is meant to come away believing that the model has learned to distinguish frogs from airplanes.

I want to start the book by asking the reader to take that belief seriously. The classifier achieved 88% accuracy. What does it know about frogs?

The question is harder than it looks. The classifier cannot tell you what a frog looks like. It cannot generate a frog. It cannot recognize a frog photographed from above instead of from the side, or a frog drawn in ink, or a green plastic frog on a desk. It has no representation of frog-ness; it has a hyperplane that, on this dataset, separates the pixel patterns labeled "frog" from the pixel patterns labeled "airplane." Strip away the labels, hand it a fresh photo of a frog without telling it the photo contains a frog, and the model has nothing to say.

What the model does have is something else. It has a decision boundary that succeeds on this data because the prior structure of the problem—frogs live on the ground; airplanes live in the sky; ground is brown and green; sky is blue and white—was carried into the model by the dataset and the labels, not by anything the model itself learned. The 88% accuracy is real. The understanding is not.

This book is about the gap between those two things. The gap is not unique to logistic regression on CIFAR-10. It runs straight through the modern field. A 175-billion-parameter language model trained on most of the public internet is a more sophisticated version of the same situation: a model that succeeds on its training distribution because the prior structure of the problem was carried in by humans—who wrote the text, who curated the corpus, who designed the architecture, who chose the loss, who built the evaluation. The model inherits that structure. The credit for what it can do belongs to the structure, and so does the responsibility for what it cannot.

My claim, in one sentence, is that machine learning is the project of encoding human knowledge into computers, and the Bayesian framework is the only formalism honest enough to name the encodings being made. Architectures are priors. Losses are priors. Training data is a prior. Metrics are priors. The composition of components in a modern Gen AI system is a stack of priors layered together. The question is never whether the priors are present—they always are—but whose knowledge is encoded, with what care, and with what awareness of what the encoding costs.

That is the book. The rest of this document walks through how I argue it.

2. Why we need a different kind of textbook

The textbook market for modern machine learning is large. Goodfellow, Bengio, and Courville's Deep Learning is the canonical reference, increasingly dated. Murphy's two-volume Probabilistic Machine Learning is the comprehensive Bayesian toolkit. Bishop and Bishop's 2024 Deep Learning: Foundations and Concepts is the modern, mathematically careful, technique-organized text. Prince, Foster, and others occupy adjacent niches. Hardt and Recht's Patterns, Predictions, and Actions is the strongest existing engagement with the political economy of ML in textbook form.

Each of these books does what it does well. None of them, individually or collectively, gives a graduate student what I think a second course in modern machine learning needs to give them: a framework for reading the field. The reason is structural. These books are organized around the techniques the field has produced. The reader learns convolutional networks, attention, autoencoders, GANs, diffusion, transformers, RLHF—one chapter at a time, each chapter standing more or less on its own. The reader who finishes such a book is conversant with the techniques. They are not equipped to read a paper they have never seen before and ask, of any architectural choice in it, what the choice is committing to and what failure mode the commitment is answering. They are not equipped to read modern Gen AI capabilities as compositions of priors rather than as monolithic models. They are not equipped to evaluate the field's methodological claims about scaling, emergence, and architectural superiority against an honest framework for what those claims assume.

I want a textbook that gives the reader the framework. Not a vocabulary list. Not a survey. A unifying lens that organizes the technical content across the modern field, that reads architectures as commitments and failures as predicted consequences of those commitments, and that takes the political economy of contemporary ML seriously enough to engage with it at the same technical depth as the rest of the book.

That is what this book is. The framework is the Bayesian one—not as a practical toolkit, which is what Murphy already does well, but as an explanatory lens for what the rest of the field is doing whether the field admits it or not. The technical arc runs from the mathematical and methodological foundations through deep learning, foundation models, and the modern generative-AI stack. The framework holds the arc together.

3. The Bayesian framework as explanatory lens

The book's conceptual keystone is what I call the Bayesian Reveal: the observation that every parameterized model trained on data is doing Bayesian inference, whether the modeler invokes Bayes' rule or not. A frequentist using maximum likelihood on a parameterized family is using a flat prior on a hard architectural truncation. A regularized loss is a prior on parameters by way of a different name. An architecture is a prior on functions, obtained by pushing a prior on parameters through the parameter-to-function mapping the architecture defines. The trained model is a posterior sample whose prior is the convolution of architecture, optimizer, initialization, and any explicit regularization the modeler chose.

This reframing is mathematically uncontroversial. What is uncommon is treating it as the organizing perspective of an entire textbook rather than as a chapter on Bayesian deep learning that lives next to the other chapters. The framework gives the reader a vocabulary for asking what every paper is committing to, a criterion (the marginal likelihood) for whether the commitment was good, and a structural reading of why specific architectures succeed or fail.

The marginal likelihood as compass, not calculator

The standard objection to a Bayesian reading of deep learning is that the marginal likelihood is intractable for the models the field actually trains. Wilson and Izmailov, Lotfi et al., Immer et al., and the broader Bayesian deep learning literature have done excellent work on computing approximations to marginal likelihoods for trained networks; that work is technically demanding and remains open. I want the book to take a different posture. The marginal likelihood is not what we compute; it is what we navigate by. We cannot calculate the absolute altitude of any deep model's marginal likelihood, but we do not need the absolute altitude if we know which direction is up.

The framework gives us a topographical map. Placing prior mass where the answer lives—and denying it where human structural reasoning knows the answer cannot live—is a mathematically guaranteed direction of improvement, regardless of whether the integral closes. This is the move that converts a technically fraught quantity into an operationally useful one.

It also makes precise what the cost-asymmetry argument actually is. The standard reading of the curse of dimensionality is purely negative: volume grows exponentially, so data becomes sparse. Reading the curse the other way, hard architectural truncations—assignments of zero prior mass to impossible parameter configurations—eliminate exponentially vast regions of spurious volume. The same dimensionality that makes brute-force fitting hard makes structural elimination cheap. Architectural design is

mathematically hyper-efficient because the curse cuts both ways, and the framework makes the directional guarantee precise without ever asking the reader to compute an integral.

Two levers

The two-levers argument follows directly. Lever 1 is scaling: throw enough data at the likelihood to overwhelm the unconstrained prior volume, by way of the Bernstein-von Mises asymptotic, at a cost in compute, electricity, water, and the carbon footprint of the data centers that produce them. Lever 2 is prior-improvement: shrink the prior volume geometrically by encoding human observations as architectural truncations, at a cost in research labor. Both produce capability gains. The framework gives the technical reason why the second is so much cheaper than the first when the structure is available to encode.

This is the move that makes the cost asymmetry a technical claim rather than a normative claim. When prior-improvement is available, choosing scaling over it is not methodologically neutral. It is a decision with consequences that the framework lets us name.

4. Representation and the manifold

The Bayesian framework gives us the language for talking about priors, but priors do not by themselves explain why deep learning works on real data. The empirical fact that deep models can be trained to fit images, text, audio, and proteins—despite the curse of dimensionality's warnings against any such thing—requires a separate observation. That observation is the manifold hypothesis: real high-dimensional data does not fill its ambient space; it lives on or near a low-dimensional manifold whose intrinsic dimension is much smaller than the ambient dimension.

A 256×256 RGB image lives in a 196,608-dimensional space. The space of natural images is unimaginably smaller than that. Most points in the ambient space are noise; the image manifold is a wisp of structure threading through an ocean of nothing. Deep learning works because the field figured out how to design architectures whose function classes, when truncated by depth and connectivity, naturally concentrate their representational capacity along that manifold.

This is what representation learning is. Every architectural choice—convolutional locality, attention's flexibility, autoencoder bottlenecks, discrete codebooks, position encodings—is a decision about how to sculpt the model's representational volume to align with where the data manifold actually is. A successful architecture is one whose hard truncations cleanly eliminate the regions of function space that cannot describe the data, leaving the model's capacity available for the regions that can.

The reason this matters for the book is that representation precedes generation. You cannot generate samples from a manifold you have not learned to represent. Useful generation requires good representation—realistic, low-dimensional, manifold-respecting representation—and good representation is achievable only through strong priors. This thread is what makes the generative-ML half of the book a single argument rather than a list of techniques. Each architecture in the generative lineage is a different attempt to sculpt a representational volume that supports generation; each failure is a failure of the underlying prior to constrain that volume well enough.

The thread also explains generative pathology directly. Generative modeling pushes the volume-allocation framework to its absolute limit by shifting the volume of interest from parameter space into the data space itself. Mapping a simple prior distribution onto a complex data manifold is extraordinarily difficult; relying on weak, unconstrained architectures leaves vast empty voids in the probability volume. Hallucination, mode collapse, demographic stereotyping, and aesthetic homogenization are not separate failure modes of separate architectures. They are the same structural pathology, the math functioning exactly as designed, smoothly interpolating across the un-pinned voids the

loose prior left behind. Existing textbooks treat these failures as separate bugs to be addressed by separate engineering. The framework reads them as one phenomenon, predicts them, and identifies the only structural solution: shrink the prior’s volume so the un-pinned regions vanish.

5. The generative lineage as motivated failures

The pedagogical centerpiece of the book is a sequence of eight chapters in which I develop generative modeling for images as one continuous argument. Each architecture in the sequence is motivated by the previous one's specific structural failure; each failure is diagnostic of a specific prior's inadequacy; each fix is a sharper prior. I want the reader to come away with two things: the technical content (they can now build any of these models), and the skill of reading architectures as commitments (they can now read any architecture they encounter in the wild).

The sequence:

The autoencoder fails because a deterministic encoder defines a Dirac-delta posterior over the latent. The latent space is pinned to single points by training data and contains no probability mass anywhere else. Generation requires sampling from the latent; sampling from a Dirac delta is undefined. The autoencoder learned a useful representation but cannot generate. This is mass collapse.

The VAE fixes mass collapse by making the encoder probabilistic—outputting a distribution over the latent rather than a point—and regularizing toward a tractable prior. We can now sample. But the canonical pixel-MSE reconstruction loss is a Gaussian likelihood with constant variance, which assumes pixel-space distances measure perceptual similarity. They do not. The model produces blurry samples because pixel-averaged blur is the MSE-optimal response to uncertainty, and the prior we put on reconstruction quality was wrong.

The GAN fixes the loss problem by replacing the explicit reconstruction objective with an adversarial one: a discriminator learns what real samples look like, and the generator learns to produce samples the discriminator cannot distinguish. No more pixel-space prior. Samples become sharp. But the framework now optimizes a minimax game with no fixed objective, training is unstable, mode collapse is endemic, and the structural reason is that the implicit prior on the data distribution is too unconstrained for the optimization dynamics to converge reliably.

The VAE-GAN fixes the GAN's instability by combining the VAE's structured latent with the GAN's discriminator-based loss. Reconstruction is now adversarial rather than pixel-MSE; the latent retains the VAE's probabilistic structure. Samples are sharper than the VAE's and more stable than the GAN's. Residual blur and limited expressiveness remain, because the continuous latent still has limited capacity to encode the fine structure of natural images.

The VQ-VAE fixes the latent-expressiveness problem by replacing the continuous latent with a discrete codebook. Each spatial position in the latent grid is quantized to one of K learned

codes. The discrete representation can encode fine structure that a continuous latent cannot, and the codebook becomes a learned vocabulary of visual primitives. Crucially, discrete latents are tokens. Once images are tokens, you can train an autoregressive transformer over them. This is the bridge to DALL-E 1.

DALL-E 1 trains a transformer over tokenized text concatenated with tokenized images, generates image tokens from text tokens, and decodes the image tokens back to pixels through the VQ-VAE's decoder. This is the first widely-successful text-to-image system, and it is structurally a composition of four priors: the VQ-VAE's discrete-token prior on images, the transformer's sequence-modeling prior on tokens, BPE's subword prior on text, and the dataset's caption-alignment prior on text-image pairs. Each component carries multiple priors. The system's capability is the aggregate.

Diffusion comes next, and it changes the framing significantly enough that I treat it as its own section. The reverse-diffusion process is a different way of generating samples from a prior, with different stability properties and a different relationship to the data manifold. Latent diffusion (Stable Diffusion) composes diffusion with the VAE-GAN encoder lineage developed in Section 8, and the modern compositional pattern—pretrained encoder + generation engine + conditioning bridge—emerges as the natural endpoint of the sequence I have just walked through.

Existing textbooks treat these architectures as a list of techniques: here is the VAE, here is the GAN, here is the diffusion model. Pulling the lineage out as a sequence of motivated failures is pedagogically valuable on its own. It is also where the reader most directly learns the skill the rest of the book wants them to take away: read any architecture as encoding a specific commitment, and read any failure as a predicted consequence of a commitment too loose to constrain the relevant volume.

The book's treatment also gives substantial space to the practical realities that determine whether a generative model is actually usable. A pixel-MSE VAE produces blurry samples and is essentially unusable for image generation. The path from "VAE you saw in a textbook" to "VAE you might actually train" requires the perceptual loss, the adversarial loss, KL annealing, free bits, posterior-collapse mitigations, GroupNorm in the encoder, and a half-dozen other choices. These are not implementation details. They are the substantive content of how the field made VAEs work, and they are where the framework most directly meets practice. Existing textbooks largely treat them as optional polish; the book treats them as essential.

6. Hallucination as marginal-likelihood pathology

The framework's most legible payoff on a topic readers already care about is its reading of hallucination in large language models. The standard discourse treats hallucination as a malfunction to be engineered away: better data, better training, better grounding, better RLHF. I argue that hallucination is the framework functioning exactly as predicted, and that calling it a malfunction is a failure to take the framework seriously.

The argument is direct. An attention-only transformer commits to almost nothing about data structure. Attention is the most flexible operator the field has standardized on; the prior it imposes on the function space is enormous. The training data—even at internet scale, even now that the field has effectively exhausted the supply of high-quality public text—cannot pin down the function within that volume. The model produces samples from the un-pinned regions. Those samples are fluent because fluency was well-pinned by the data; they are factually arbitrary because facticity was not. The model is not malfunctioning. The model is correctly producing samples from the posterior the architecture and data jointly define, and the posterior is loose where the data did not constrain it.

This is what I call the shocked-Pikachu argument. We made the architectural choice that requires astronomical data to identify the right function. We could not possibly have provided that data. The model produces samples from the un-pinned regions. We are surprised. The framework predicted exactly this; we are surprised because we were not using the framework.

The chapter then derives why scale-up cannot fix the problem. More data of the same kind reinforces the same distribution and leaves the prior-volume problem untouched; the data-exhaustion situation means there is no more high-quality data to add; we are at the structural limit of what Lever 1 can accomplish. The field's mitigations—RAG, grounding, calibration, constitutional methods, chain-of-thought, tool use—are all implicit fixes that constrain outputs without changing the prior. They compensate for the prior's looseness rather than tightening it. The structurally honest path forward is architectural change that produces a lower-volume prior: diffusion language models, energy-based models, hybrid architectures with structural priors built in.

I think this is the chapter that will most clearly demonstrate to readers that the framework is not a methodological commentary attached to a textbook. It is a predictive instrument that tells you, before you train the model, what the model will and will not be able to do.

7. The stack of priors and the cost of scale

Modern Gen AI is not single models. It is compositions. Stable Diffusion is five components stacked: a VAE-GAN encoder/decoder, a CLIP text encoder, a diffusion U-Net or DiT, a sampler, and classifier-free guidance. LLaVA stacks a vision encoder under a language model with a learned projection bridge. ChatGPT stacks a base language model under RLHF under a tool-use harness under a system prompt. Each component carries multiple priors; the composition's capability is the aggregate; when the composition succeeds, every component contributed; when it fails, every component is implicated.

The reading matters because the field talks about modern systems as if they were monolithic. "The model learned X." "The system understands Y." The composition obscures the source of capabilities and failures unless the reader traces explicitly. When demographic stereotyping shows up in Stable Diffusion outputs, no single component is the source; the composition propagates a bias originating in the training data and preserved at every subsequent layer. When prompt-following works well, the capability comes from CLIP's caption-curated semantic alignment composed with the diffusion model's cross-attention to CLIP's embeddings composed with the CFG strength composed with the sampler's numerical fidelity. The framework gives the reader the vocabulary to ask, of any modern system, what each component is committing to and which component is responsible for any given behavior.

This is also where the political-economic argument lands. Once we read modern Gen AI as compositions of priors, the methodological choices that produced the dominant compositional pattern come into view. Why did the field standardize on attention-only transformers for sequence modeling, when 1D convolutional architectures (WaveNet, ByteNet, ConvS2S) achieved substantial fractions of attention-based performance at O(N) rather than $O(N^2)$ cost? The answer, when you trace it, is not that attention is methodologically superior—it is that attention parallelized better on the GPU stack the field had standardized on, and the cost asymmetry between architectural research and compute-scaling tilted decisively toward scaling once the compute was available. Why did the field invest astronomical resources in scaling autoregressive transformers when alternative architectures with stronger structural priors were available? Same answer.

The Bitter Lesson reading this produces is sharper than the standard one. Sutton's claim, in its narrow form, is correct: methods that scale with compute eventually overtake methods that do not. But the claim has been overgeneralized into a methodological prescription that ignores the cost side of the ledger. Both levers produce capability. They cost differently. Lever 1 (scaling) costs compute, electricity, water, carbon, and the geographic concentration of physical infrastructure that imposes specific costs on specific communities. Lever 2 (prior-improvement) costs research labor. The math says they are equivalent in capability

when both are available. The world says one of them costs people their water, their air, and their livelihoods. The field's apparent agnosticism between the levers is not actually agnostic; it is a choice with consequences, and the consequences are not borne by the people making the choice.

I want the book to make this argument carefully, not as a polemic but as the natural extension of the technical framework. By the time the reader reaches the closing chapters, they have ten sections of mathematical machinery for evaluating the claim. They are equipped to agree, to disagree, or to refine. What the book asks of them is to not be agnostic.

8. What the book asks of its reader

The closing chapter does not lecture the reader on what to do. It asks the reader to do three things and trusts them to act on the framework as they see fit.

First, see ML as priors. There is no level of a modern ML system at which the priors stop. Architecture is a prior; loss is a prior; training data is a prior; metric is a prior. When you read a paper, ask what it is committing to. When you build a system, ask what you are committing to. When you evaluate a result, ask what the prior the system encodes was, and whether the data could have constrained the function within that prior's volume.

Second, see compositions clearly. Modern Gen AI is not single models; it is stacks of priors. Trace the composition. When something works, identify which components contributed. When something fails, identify which components are implicated. Do not let the field's monolithic vocabulary obscure the structural reading.

Third, take the cost asymmetry seriously. When prior-improvement is available, the choice between it and scaling is not methodologically neutral. The framework gives you the vocabulary to ask what each lever costs, and what is gained that could not have been gained the other way. You will not always conclude that prior-improvement is the right choice. But you will not be able to conclude that scaling is the right choice without having examined the alternative, and the field, on aggregate, has not examined the alternative as carefully as the cost asymmetry warrants.

The framework is the reader's now. The book is a snapshot; the specific systems it discusses will be obsolete within a few years; the framework is meant to outlast them. A reader who finishes the book should be equipped to apply the framework to systems that did not exist when the book was written, and to evaluate methodological claims the field has not yet made. That is what I am trying to give them. Whether the book actually succeeds at giving it to them is a question I would like your help in answering.

9. What I would most like feedback on

Below are the specific questions where I think the book is most exposed to error and where I would most value pushback. They are not in order of importance.

Is the framework load-bearing or decorative?

The book's organizing claim is that the Bayesian framework as explanatory lens does real work—that the reader who reads the book through the framework comes away with something they would not have come away with otherwise. The hardest version of the criticism is that the framework is a coat of paint over a conventional ML textbook: pleasant to look at, easy to remove, not actually changing the structure underneath. I think the framework is load-bearing in at least four places (the marginal-likelihood-as-compass move, the representation-requires-strong-priors thread, the hallucination-as-pathology reading, the stack-of-priors composition reading). Are these four enough? Are there places I think the framework is doing work where it is actually decorative?

Is the cost-asymmetry argument well-grounded?

The argument that scaling and prior-improvement are equivalent in capability and asymmetric in cost rests on technical premises (the marginal likelihood as the right criterion, BvM as the asymptotic that scaling relies on, hard architectural truncation as the geometric alternative) and empirical premises (that prior-improvement is broadly available across the field's open problems, that the cost asymmetry is as sharp as I argue, that Sutton's Bitter Lesson generalizes only in the narrow form I grant). The technical premises I am confident in. The empirical premises are where the argument is most exposed. Where is the strongest counterargument I have not engaged with? Specifically: empirical scaling laws (Kaplan, Hoffmann/Chinchilla, recent compute-optimal results) suggest that for some classes of problem, scale produces capability gains that don't correspond to better priors in any meaningful sense. The framework can absorb this (the priors were always implicit in architecture and training procedure; scale just calibrated within them), but the absorption looks unfalsifiable to an unconvinced reader. How would you sharpen the response?

Is the historical-as-pedagogical sequence right?

Section 5 of this document walks through the autoencoder → VAE → GAN → VAE-GAN → VQ-VAE → DALL-E 1 → diffusion lineage as one continuous argument. I think this sequence is the strongest single piece of pedagogy in the book. I also recognize that I am claiming a specific reading of the field's history, and that other plausible orderings exist (energy-based models, normalizing flows, score-matching as a parallel lineage rather than as branches off the main line). Have I gotten the history right? Are there architectures or transitions I am

underweighting? Are there transitions I am overstating as motivated when they were actually independent developments?

Is the political-economic content landing in the right register?

The closing chapters argue that the field's methodological choices have real distributional consequences and that the framework gives us the vocabulary to evaluate them. I have tried to make this an extension of the technical argument rather than a separate critical posture, and I have tried to do it without overclaiming. The risk is real that the book is read as "the political book" rather than as "the technical book with political content." Two specific worries. First: am I making the argument too strongly, in a way that will alienate readers who would otherwise be receptive? Second: am I making it too gently, in a way that produces another textbook that gestures at ethics in a final chapter and doesn't change anyone's practice? I genuinely don't know which side of this I am on, and I would value an outside read.

Who is the audience, really?

I have been writing as if the audience is graduate students and advanced undergraduates with mathematical maturity at the level of a strong undergraduate and prior exposure to ML at the level of ISL—the floor—with the ability to follow ESL at the level it is presented as the ceiling. Section 2 of the book is meant to be the bridge for readers who are at ISL-level readiness and need to get to ESL-level fluency. Is this the right audience? Is Section 2 the right way to handle the bridge, or should it migrate online? Are there audiences I am underserving (early-career researchers, working practitioners, faculty developing courses) that the book could serve with relatively small adjustments?

Is anything important missing?

The book covers modern ML in the deep-learning + foundation-model paradigm. RL gets RLHF in the foundation-models section but no deeper treatment. Causal inference is barely present. Classical methods (kernel methods, graphical models, sequential Monte Carlo) appear only insofar as they help the framework. Some of these absences are honest scoping; some of them might be mistakes. What would you not have left out?

Is there a stronger version of the argument I am not seeing?

The hardest feedback to give and the most valuable: where would a smarter version of me have made this argument differently? What is the best book this could have been that this draft is not?

§

Thank you for reading. The book is a draft and your engagement with it is what will make it better than I could make it alone.