6

Bayesian Learning

> *. . . the mathematical rules of probability theory are not merely rules for calculating frequencies of 'random variables'; they are also the unique rules for conducting inference (i.e. plausible reasoning) of any kind, and we shall apply them in full generality to that end.*
>
> E. T. Jaynes, explaining the theme of his book [50].

In the previous three chapters, we've spent a considerable amount of spacetime analyzing the ensemble of wide neural networks at initialization. In particular, through the $1/n$ expansion and deep asymptotic analysis, we've obtained a rather thorough understanding of the interplay between the architecture, width, depth, and initialization hyperparameters that together define the effective distribution of preactivations.

In this study, we've paid very careful attention to the *deep* of *deep learning* to the total neglect of the *learning*. But this is a *deep learning book*, not just a *deep book*. Thus, in this chapter we will begin to learn about learning and – if the titles of our chapters are any real guide to their contents – will continue learning about learning for the rest of the book.

We'll begin on our learning quest with a discussion of *Bayesian inference*, as it provides a natural framework for thinking about learning in general. We'll first explain in §6.1 the Bayesian approach to probability, in which probabilities are reinterpreted to represent the strength of our beliefs about the world according to different hypotheses. There, we'll learn that the rules of Bayesian inference – really the rules of logic extended to probabilistic reasoning – pick out a logically consistent way of incorporating newly observed information into the probabilistic models representing our hypotheses.

From §6.2 on out, we'll see why this simple yet powerful framework enables us to analyze and then understand how deep neural networks learn from observed data.

In §6.2.1, we'll detail how Bayesian model fitting works for neural networks. First, we'll reinterpret our well-studied effective preactivation distribution as a *prior* distribution, encoding our initial beliefs about the model outputs before observing any data. With this as a starting point, the rules of Bayesian inference then imply a learning algorithm for sharpening our beliefs so as to best fit our observations. The result

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

of inference – the *posterior* distribution – further lets us make Bayesian predictions on novel inputs whose outputs we haven't observed but need to infer. This naturally segues into a discussion of practical implementations: first we'll discuss approximation methods – giving a Bayesian interpretation to the gradient-based learning methods that we'll explore in the epochs following this chapter – and then we'll discuss an exact method on which the rest of the current chapter will be based.

In §6.2.2, we'll expand our horizons by contemplating the ultimate question of Life, the Universe, and Everything: Bayesian model comparison. We'll explain how to use Bayesian *evidence* to select between different plausible hypotheses, organized according to different choices of hyperparameters and network architectures, in order to pick the best ones. Bayesian model comparison also gives us a quantitative means to address *inductive biases*, the often hidden assumptions built into deep learning models. As a bonus, we'll further see how *Occam's razor* is automatically incorporated in the rules of Bayesian inference applied to such model comparison. With these tools, we can really begin to address one of the fundamental questions we posed at the beginning of the book: why do some neural network models perform so well while others fail?

These abstract discussions are then followed by an onslaught of concrete calculations for infinite- and finite-width neural networks in §6.3 and §6.4, respectively.

Some of these calculations reinforce the themes of the previous chapter. We'll first show that Bayesian model comparison prefers critical initialization hyperparameters, giving additional evidence for the *principle of criticality* (§6.3.1). We'll also illustrate another role of finite-width interactions. Specifically, the accumulation of correlated *fluctuations* induces an inductive bias for *neural association*, leading to a propensity for Hebbian learning – a learning principle inspired by biological neurons (§6.4.1).

Some of these calculations contrast qualitatively different characteristics of infinite- and finite-width models that are trained with exact Bayesian learning. Analyzing the posterior distribution of network outputs, we'll see that correlations among different components of the output are nonzero at finite width only, in two contrasting subsections (§6.3.2 ⊥ §6.4.2). The resulting expressions will also make it clear why – while theoretically quite tractable – exact Bayesian learning is impractical for any dataset of reasonable size. Next, analyzing the posterior distribution of hidden-layer representations, we'll see the absence/presence of representation learning at infinite/finite width (§6.3.3 ⊥ §6.4.3). Overall, this contrasting will provide a valuable blueprint for when we later consider infinite- and finite-width models trained with gradient-based learning (§10 ⊥ §∞).

6.1 Bayesian Probability

A Bayesian always starts with a **hypothesis** $\mathcal{H}$. Mathematically, a hypothesis is a mechanism for assigning numbers $p(A|\mathcal{H})$ to *statements* A about the world. These statements are logical propositions – such as "it will rain tomorrow" or "this image x contains a cat" or "the output value for this function $f(x)$ evaluated on an input x is z" – and these numbers $p(A|\mathcal{H})$ represent the relative plausibilities of those statements

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

according to the assumptions or model of the world summarized by the hypothesis $\mathcal{H}$. In the context of machine learning, $p(A|\mathcal{H})$ is often called a **probabilistic model**.

As this notation and discussion should make clear, these beliefs $p(A|\mathcal{H})$ are expressed in the language of probability. However, the *Bayesian* interpretation of the probability $p(A|\mathcal{H})$ subtlely differs from the *ensemble* interpretation that we gave in §2.3. Namely, rather than representing the statistics of a random variable – the relative frequency or chance of observing A, given the conditions $\mathcal{H}$ – this probability instead constitutes the strength of our belief in the proposition A according to the assumptions $\mathcal{H}$.[1] Further, with such a Bayesian perspective, all of probability theory and statistical inference can be uniquely derived as a consequence of logical constraints on these beliefs $p(A|\mathcal{H})$.[2] We'll next brief you through these constraints as they form the foundation of this chapter, but, as we have been using probabilities for quite a while now in this book, let us be brief.

Formally, the first logical constraint is known as the **product rule**,

$$p(A, B|\mathcal{H}) = p(A|B, \mathcal{H})\, p(B|\mathcal{H}) = p(B|A, \mathcal{H})\, p(A|\mathcal{H}), \tag{6.1}$$

where $p(A, B|\mathcal{H})$ represents a *joint* belief in both A *and* B according to the hypothesis $\mathcal{H}$, while $p(A|B, \mathcal{H})$ represents a *conditional* belief in A according to $\mathcal{H}$ *given* that B has been observed. The second logical constraint is known as the **sum rule**,

$$p(A|\mathcal{H}) = \sum_B p(A, B|\mathcal{H}), \tag{6.2}$$

and relates the joint belief in A and B to a marginal belief in just A.[3] Here, the symbol $\sum_B$ represents a sum over all the logically possible values of a discrete variable B, or for a continuous variable it represents an integral.[4] This sum rule in particular implies the

[1] The *ensemble* interpretation is often called **frequentist probability** when contrasted with **Bayesian probability**. In this book, we use the interpretation that is most appropriate for the particular problem under consideration: if we're instantiating models by randomly drawing parameters from an initialization distribution, it makes sense to analyze an ensemble; if we're making inferences based on a fixed hypothesis or comparing different hypotheses, it makes sense to adopt the Bayesian perspective.

[2] See Jaynes' book [50] for an extended development of this perspective to which our brief summary does not do justice.

[3] We essentially discussed this sum rule as (4.93) under §4.4 *Marginalization Rules*.

[4] Though (Bayesian) probably it's already clear if you've made it this deep in the book, as we cannot be (Bayesian) certain, let us clarify the meaning of the *statement* A inside the belief system $p(A|\mathcal{H})$. Sometimes a statement represents a *fixed* logical proposition, such as $A =$ "Schrödinger's cat is alive," with $p(A|\mathcal{H})$ encoding the plausibility of the cat's aliveness. Sometimes a statement represents a binary *variable*, such as $B =$ "the living status of Schrödinger's cat," which takes values in $\{\text{dead}, \text{alive}\}$ with $p(B|\mathcal{H})$ giving the distribution over the two binary outcomes. More generally, the statement can represent observable outcomes $\mathcal{O}$ of experiments – a.k.a. *observables* – with $p(\mathcal{O}|\mathcal{H})$ encoding our relative belief in the plausibilities of the different outcomes, where such observables can take on a discrete or continuous spectrum of values. Prominent examples of such general observables for us include the model parameters θ and preactivations $z^{(\ell)}$.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

normalization condition if we assign $p(C|\mathcal{H}) \equiv 1$ for the statement C that holds with *absolute certainty* according to $\mathcal{H}$:

$$\sum_B p(B|\mathcal{H}) = \sum_B p(C, B|\mathcal{H}) = p(C|\mathcal{H}) = 1. \tag{6.3}$$

With these rules in mind, after fixing a hypothesis, a Bayesian then gathers information in order to refine the plausibilities of different beliefs. For instance, after *observing* A, we may want to *update* our beliefs about B. Such **Bayesian inference** can be accomplished by noting that an algebraic rearrangement of the product rule (6.1) tells us how our beliefs should change as we condition on additional information A:

$$p(B|A, \mathcal{H}) = \frac{p(A|B, \mathcal{H})\, p(B|\mathcal{H})}{p(A|\mathcal{H})}. \tag{6.4}$$

This rearrangement is so important that it's given its own name, **Bayes' rule**, and even each individual factor of the equation is named as well:

- The factor $p(B|\mathcal{H})$ is called the **prior** of B, thusly named because it quantifies our belief in B *a priori*; that is, it encodes our belief in B based entirely on our model $\mathcal{H}$ before we observe any additional information.
- The factor $p(B|A, \mathcal{H})$ is called the **posterior** of B given A, thusly named because it quantifies our belief in B *a posteriori* upon learning A; that is, it encodes how our model $\mathcal{H}$ updates its belief in B after observing A.
- The factor $p(A|B, \mathcal{H})$ is called the **likelihood**. We'll elaborate more on its name and interpretation later in §6.2.1 where we talk about model *fitting*.
- The factor $p(A|\mathcal{H})$ is called the **evidence** for $\mathcal{H}$. We'll elaborate more on its name and interpretation later in §6.2.2 where we talk about model *comparison*.

Note that the posterior is automatically normalized:

$$\sum_B p(B|A, \mathcal{H}) = \sum_B \frac{p(A|B, \mathcal{H})\, p(B|\mathcal{H})}{p(A|\mathcal{H})} = \sum_B \frac{p(A, B|\mathcal{H})}{p(A|\mathcal{H})} = \frac{p(A|\mathcal{H})}{p(A|\mathcal{H})} = 1. \tag{6.5}$$

More importantly, Bayes' rule is the only logically consistent way to update a set of beliefs after making observations.

6.2 Bayesian Inference and Neural Networks

The Bayesian framework for inference can be used for building, updating, and reasoning with powerful probabilistic models of the world. Let's now see how we can apply the Bayesian framework to deep learning, first for model fitting (§6.2.1) and then for model comparison (§6.2.2).

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

6.2.1 Bayesian Model Fitting

For neural networks, it's most natural to begin by discussing the prior distribution $p(\theta|\mathcal{H})$ of the model parameters $\theta_\mu = \left\{b_i^{(\ell)}, W_{ij}^{(\ell)}\right\}$. This prior lets us quantify our initial beliefs about the particular values of the model parameters that determine our neural-network function approximator $f(x;\theta)$. The most common choice is to simply reinterpret the initialization distribution of the ensemble,

$$p(\theta|\mathcal{H}) \equiv \prod_{\ell=1}^{L} \left\{ \left[\prod_{i=1}^{n_\ell} p\left(b_i^{(\ell)}\right) \right] \left[\prod_{i=1}^{n_\ell} \prod_{j=1}^{n_{\ell-1}} p\left(W_{ij}^{(\ell)}\right) \right] \right\}, \tag{6.6}$$

as our Bayesian prior distribution. Here we recall that $p\left(b_i^{(\ell)}\right)$ and $p\left(W_{ij}^{(\ell)}\right)$ – given by (2.21) and (2.22) – are zero-mean Gaussian distributions with bias variance $C_b^{(\ell)}$ and weight variance $C_W^{(\ell)}/n_{\ell-1}$, respectively.

From the Bayesian perspective, these initialization hyperparameters are part of the hypothesis $\mathcal{H}$. This hypothesis $\mathcal{H}$ also contains our choice of architecture – MLP, CNN, transformer, etc. – as well as all the architecture hyperparameters within that architecture class – e.g., for MLPs we need to further select the depth L, the hidden-layer widths n_ℓ, and the activation function $\sigma(z)$. In short, $\mathcal{H}$ is for $\mathcal{H}$yperparameters.[5]

Here, we've taken familiar objects – the hyperparameters and the initialization distribution characterizing the frequency of potential network realizations – and interpreted them in the way of Bayes – as the hypothesis $\mathcal{H}$ and as the prior distribution $p(\theta|\mathcal{H})$ characterizing our initial beliefs about the value of the model parameters. Another familiar object, of course, is the distribution of ℓ-th-layer preactivations that we've spent the last three chapters evaluating explicitly. To give that a Bayesian interpretation, let us first denote by $z_{\mathcal{D}}^{(\ell)} \equiv \left\{z_{i;\delta}^{(\ell)}\right\}$ the set of ℓ-th-layer preactivations evaluated on inputs $x_{j;\delta} \in \mathcal{D}$ in some dataset $\mathcal{D}$. Then, the prior distribution over these ℓ-th-layer preactivations can be related to the prior distribution over the model parameters by

$$p\left(z_{\mathcal{D}}^{(\ell)}\middle|\mathcal{H}\right) = \int \left[\prod_{\mu=1}^{P} d\theta_\mu\right] p\left(z_{\mathcal{D}}^{(\ell)}, \theta\middle|\mathcal{H}\right) = \int \left[\prod_{\mu=1}^{P} d\theta_\mu\right] p\left(z_{\mathcal{D}}^{(\ell)}\middle|\theta, \mathcal{H}\right)\, p(\theta|\mathcal{H}), \tag{6.7}$$

where we've applied the sum rule (6.2) in the first equality and the product rule (6.1) in the second. This prior quantifies our initial beliefs about the different neural-network

[5] To be strict, we should have always conditioned on $C_b^{(\ell)}$, $C_W^{(\ell)}$, and n_ℓ whenever we discussed the initialization distribution: $p(\theta) \to p\left(\theta\middle|n_0, C_b^{(1)}, C_W^{(1)}, \ldots, n_{L-1}, C_b^{(L)}, C_W^{(L)}\right)$. Thankfully we've so far left, and will continue to leave, this type of detailed dependence implicit for notational simplicity. However, to underscore the importance of the hypothesis for Bayesian inference, in this chapter we *(i)* will leave the conditioning on the overall hypothesis $\mathcal{H}$ explicit until the end of §6.3.1 and at the same time *(ii)* will move the dependence of a dataset $\mathcal{D}$ to an overall subscript of the preactivations. As a particular example, the prior distribution of the ℓ-th-layer preactivations $p\left(z_{\mathcal{D}}^{(\ell)}\middle|\mathcal{H}\right)$, defined next paragraph in (6.7), is equivalent to what we've been denoting as $p\left(z^{(\ell)}\middle|\mathcal{D}\right)$ outside of this chapter.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

variables. More specifically, for a hidden layer ℓ, this distribution represents our beliefs about a particular *feature representation* of the input and, for the output layer L, this represents our initial beliefs about the behavior of the *function approximation* $f(x;\theta)$. More generally, for any neural-network observable $\mathcal{O} = \mathcal{O}(\theta)$, our prior beliefs are determined by

$$p(\mathcal{O}|\mathcal{H}) = \int \left[\prod_{\mu=1}^{P} d\theta_\mu\right] p(\mathcal{O}|\theta,\mathcal{H})\; p(\theta|\mathcal{H}). \tag{6.8}$$

To better illustrate what these formal expressions represent, let us take the network output $z_{\mathcal{D}}^{(L)}$ as an observable. Then, the prior distribution for the output layer $p\Big(z_{\mathcal{D}}^{(L)}\Big|\mathcal{H}\Big)$, (6.7), is the same distribution as the output distribution induced by the initialization ensemble, (2.35), *if and only if* we also pick the conditional distribution of the outputs given the parameters to be deterministic:

$$p\Big(z_{\mathcal{D}}^{(L)}\Big|\theta,\mathcal{H}\Big) = \prod_{i=1}^{n_L}\prod_{\delta\in\mathcal{D}} \delta\Big(z_{i;\delta}^{(L)} - f_i(x_\delta;\theta)\Big). \tag{6.9}$$

Here, $f_i(x_\delta;\theta)$ is an expression for the network output given in terms of the iteration equation that defines the MLP (2.5), while $z_{i;\delta}^{(L)}$ is interpreted as a random variable. The resulting prior distribution for the network outputs $p\Big(z_{\mathcal{D}}^{(L)}|\mathcal{H}\Big)$ then characterizes our overall initial belief about the joint set of output values for a given set of inputs $\mathcal{D}$ according to the hypothesis $\mathcal{H}$, instead of characterizing the relative frequency of such output values at initialization across different realizations of the model parameters. That said, operationally, the formalism developed in the previous chapters can be directly brought to bear on calculating with these beliefs.

Importantly, note that the deterministic conditional distribution for the output (6.9) is a part of our hypothesis within the Bayesian framework: according to the hypothesis $\mathcal{H}$, given the model parameters θ, the outputs are *definitely* the ones computed by the function $f(x;\theta)$. Another common hypothesis is the *uncertain hypothesis*

$$p\Big(z_{\mathcal{D}}^{(L)}\Big|\theta,\mathcal{H}\Big) = \prod_{i=1}^{n_L}\prod_{\delta\in\mathcal{D}} \left\{\frac{1}{\sqrt{2\pi\sigma_\varepsilon^2}}\exp\left[-\frac{1}{2\sigma_\varepsilon^2}\Big(z_{i;\delta}^{(L)} - f_i(x_\delta;\theta)\Big)^2\right]\right\}, \tag{6.10}$$

which reduces to the deterministic hypothesis (6.9) in the limit of zero variance and absolute certainty: $\sigma_\varepsilon^2 \to 0$.[6]

[6]This hypothesis is equivalent to injecting random noise ε_i with mean zero and variance σ_ε^2 into the network output. This in turn is tantamount to shifting the last-layer biases as $b_i^{(L)} \to b_i^{(L)} + \varepsilon_i$, and hence we can easily incorporate this in our analysis by shifting the final bias variance as $C_b^{(L)} \to C_b^{(L)} + \sigma_\varepsilon^2$. You should keep in mind, however, that ε_i is separate from the bias $b_i^{(L)}$ and is *not* a part of the adjustable model parameters θ_μ; instead, this noise is intended to embody an intrinsic uncertainty present in our observation of the model's output.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

Having now thoroughly discussed the prior, let's next consider the posterior. As we gather more information A about the true behavior of our desired function $f(x)$, we should update our beliefs about our probabilistic model for $f(x;\theta)$. In order to incorporate this information in a logically consistent manner, we should use Bayes' rule. Specifically, to update our belief about the model parameters, Bayes' rule (6.4) instructs us to use

$$p(\theta|A,\mathcal{H}) = \frac{p(A|\theta,\mathcal{H})\,p(\theta|\mathcal{H})}{p(A|\mathcal{H})}. \tag{6.11}$$

Here, to find the posterior distribution $p(\theta|A,\mathcal{H})$, the prior distribution $p(\theta|\mathcal{H})$ gets multiplied by the likelihood $p(A|\theta,\mathcal{H})$ of the model parameters θ for the observation of A, and divided by the evidence $p(A|\mathcal{H})$. Consequently, with such a posterior distribution of the model parameters, our beliefs about any neural-network observable $\mathcal{O}$ shifts from our prior $p(\mathcal{O}|\mathcal{H})$ (6.8) to a posterior with the insertion of A,

$$p(\mathcal{O}|A,\mathcal{H}) = \int \left[\prod_{\mu=1}^{P} d\theta_\mu\right] p(\mathcal{O}|\theta,\mathcal{H})\; p(\theta|A,\mathcal{H}). \tag{6.12}$$

These two equations (6.11) and (6.12) uniquely determine how new information A can be incorporated to change our beliefs about the value of any neural-network observable.

For function approximation tasks, such information often comes in the form of some dataset $\mathcal{A}$ containing observed input-output pairs:

$$A \equiv \{(x_{j;\tilde{\alpha}}, y_{i;\tilde{\alpha}})\}\,|_{\tilde{\alpha}\in\mathcal{A}}. \tag{6.13}$$

Here, each input $x_{j;\tilde{\alpha}} \in \mathcal{A}$ is paired with its corresponding true output $y_{i;\tilde{\alpha}} \equiv f_i(x_{\tilde{\alpha}})$ recorded from our desired function $f(x)$.[7] With our observation of the true values $y_{\mathcal{A}} \equiv \{y_{i;\tilde{\alpha}}\}$, the likelihood and evidence are then given by the conditional belief $p(y_{\mathcal{A}}|\theta,\mathcal{H})$ and the belief $p(y_{\mathcal{A}}|\mathcal{H})$ for outputs, respectively. Such beliefs appeared before when considering the prior distribution of the outputs, (6.7) with $\ell = L$, but are now evaluated on the *fixed* values $y_{\mathcal{A}}$ associated with the given inputs $x_{\mathcal{A}}$.

Before moving on, let us also mention one other common hypothesis for the network output, the *categorical hypothesis*, defined for each input x by

$$p(i|\theta,\mathcal{H}) \equiv \frac{\exp[f_i(x;\theta)]}{\sum_{j=1}^{n_L}\exp[f_j(x;\theta)]}. \tag{6.14}$$

This distribution is also sometimes known as the *softmax*. Here, instead of considering a continuous distribution over the n_L output *values* $z_i^{(L)}$, we consider a discrete distribution over output *classes* i, such as `dog` or `cat` or `car`; then, for such classification tasks, each number $p(i|\theta,\mathcal{H})$ quantifies our belief about how likely it is that the input x represents the class i. Functionally, the softmax can be thought of as a generalization of the logistic function (2.10) in the sense that it maps a vector of real numbers to a discrete probability distribution.

[7] For maximal disambiguation, in this chapter we'll use sample indices of the form $\tilde{\alpha}$ – the Greek letter alpha with a tilde on top – for elements of the dataset $\mathcal{A}$ corresponding to input-output pairs for which the *true* output values from $f(x)$ are observed.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

To develop some intuition for what this means, let's again take the deterministic hypothesis (6.9). In this case, the likelihood is given by

$$p(A|\theta,\mathcal{H}) \equiv p(y_{\mathcal{A}}|\theta,\mathcal{H}) = \prod_{\tilde{\alpha}\in\mathcal{A}} \prod_{i=1}^{n_L} \delta\Big(y_{i;\tilde{\alpha}} - f_i(x_{\tilde{\alpha}};\theta)\Big). \tag{6.15}$$

This likelihood quite explicitly restricts the model parameters to those *exactly* satisfying the constraints $f_i(x_{\tilde{\alpha}};\theta) = y_{i;\tilde{\alpha}}$ *fitting* our observations. Vice versa, the functions in our set that do *not* satisfy these constraints are completely thrown away from the posterior distribution, deemed *unlikely*. Note what has just happened. Naively, $p(y_{\mathcal{A}}|\theta,\mathcal{H})$ represents our beliefs about the output values $y_{\mathcal{A}}$, given that we set the parameters of our model to θ. However, here we *first* observed the true output values $y_{\mathcal{A}}$ and *then* interpreted $p(y_{\mathcal{A}}|\theta,\mathcal{H})$ in terms of how likely it is that the model parameters θ fit the observation A. This is the origin of the name "likelihood" and why the proper way to refer to it is "the likelihood of the model parameters θ for the observation A."

To develop even more intuition, it's customary to introduce the **negative log-likelihood** $\mathcal{L}_{\mathcal{A}}(\theta)$ – or *loss* – representation of the likelihood:

$$p(y_{\mathcal{A}}|\theta,\mathcal{H}) \equiv \exp[-\mathcal{L}_{\mathcal{A}}(\theta)]\,. \tag{6.16}$$

Here, by parameterizing the loss as a function of the parameters θ, we are emphasizing that it's the (negative log-)likelihood *of* the parameters.[8] For the uncertain hypothesis (6.10), the negative log-likelihood takes the form of the famous mean-squared-error or **MSE loss**:

$$\mathcal{L}_{\text{MSE}}(\theta) = \sum_{\tilde{\alpha}\in\mathcal{A}} \left\{ \frac{1}{2\sigma_\varepsilon^2}\big[f_i(x_{\tilde{\alpha}};\theta) - y_{i;\tilde{\alpha}}\big]^2 + \frac{1}{2}\log\Big(2\pi\sigma_\varepsilon^2\Big) \right\}. \tag{6.17}$$

In particular, as the network outputs $f_i(x_{\tilde{\alpha}};\theta)$ get closer to their target values $y_{i;\tilde{\alpha}}$, the MSE loss decreases and the likelihood increases.[9] As such, the loss is a natural measure of how well our model is approximating the true behavior of the function. Additionally, since the loss (6.17) involves an explicit sum over observations, as the number of observed input-output pairs $N_{\mathcal{A}}$ increases, the likelihood can dominate the prior; that is, if we gather enough information, eventually our prior beliefs can become entirely replaced by what we learned from our observations.

[8] While the likelihood function – and therefore the loss – is considered auxiliary from the perspective of function approximation, from the perspective of Bayesian inference, the form of the likelihood is considered to be part of the hypothesis, cf. the deterministic hypothesis (6.9) vs. the uncertain hypothesis (6.10).

[9] In the deterministic limit $\sigma_\varepsilon^2 \to 0$, the loss $\mathcal{L}_{\mathcal{A}}(\theta)$ would be infinite for functions that don't *exactly* fit all the constraints $f_i(x_{\tilde{\alpha}};\theta) = y_{i;\tilde{\alpha}}$ and negative infinite for those that do. Thus, the uncertain hypothesis softens these hard-fitting constraints of the deterministic hypothesis by relaxing the Dirac delta function distribution to a Gaussian distribution with a finite variance σ_ε^2.

When we consider the categorical hypothesis (6.14), the negative log-likelihood of the softmax distribution gives the *cross-entropy loss*. We'll address the consequences of these different choices of loss functions more systematically in §10.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

This is **Bayesian model fitting**: Bayesian inference (6.11) is used as a *learning algorithm* to increase the accuracy of a function approximation. It gives greater preference to the functions that better fit the constraints $f_i(x_{\tilde{\alpha}};\theta) = y_{i;\tilde{\alpha}}$ and penalizes the ones that don't. The posterior (6.11) is then updated to reflect a balance between this preference for fitting our observations and an adherence to our prior beliefs about the values the model parameters should take.

Ultimately, we want to use our fit Bayesian model to make **Bayesian predictions**. This is generically and abstractly embodied in (6.12). Specifically and concretely, for function approximation tasks we are most often interested in posterior beliefs about the network outputs $\mathcal{O} = z^{(L)}$, for which (6.12) reads

$$p\Big(z^{(L)}\Big|A,\mathcal{H}\Big) = \int \left[\prod_{\mu=1}^{P} d\theta_\mu\right] p\Big(z^{(L)}\Big|\theta,\mathcal{H}\Big)\; p(\theta|A,\mathcal{H}). \tag{6.18}$$

Once we have this distribution, then we can in particular use its mean as our prediction and its variance as our level of confidence. To compute any of these quantities, one way or another we need to perform a gigantic integral over the model parameters θ in order to properly weight our different beliefs. With that in mind, we'll now present two kinds of methods to tackle this model marginalization: *(i)* approximate methods based on saddle-point approximations and *(ii)* an exact method based on our effective theory approach.

Approximation Methods for Model Marginalization: MAP and MLE

One way to tackle such a gigantic integral is to presume that the integral measure, given by the posterior distribution $p(\theta|A,\mathcal{H})$ (6.11), is very concentrated around its *mode*:

$$\theta^\star_{\text{MAP}} \equiv \arg\max_\theta p(\theta|A,\mathcal{H}) = \arg\max_\theta \left[p(y_\mathcal{A}|\theta,\mathcal{H})\, p(\theta|\mathcal{H})\right]. \tag{6.19}$$

This maximum is known as the **maximum a posteriori** (MAP) estimate. After such a maximization, we can use the function $f(x;\theta^\star_{\text{MAP}})$ for tasks and more generally approximate the full posterior distribution $p(\mathcal{O}|A,\mathcal{H})$ (6.11) by the point estimate $\mathcal{O}(\theta^\star_{\text{MAP}})$. This notion of approximating a probability distribution with a single value of the random variable is known in statistics as a *point estimate* and in physics as a *saddle-point approximation*. Another commonly-used saddle is given by the maximum of the likelihood,

$$\theta^\star_{\text{MLE}} \equiv \arg\max_\theta p(y_\mathcal{A}|\theta,\mathcal{H}), \tag{6.20}$$

known as the **maximum likelihood estimation** (MLE) of the model parameters.

In terms of the negative log-likelihood $\mathcal{L}_\mathcal{A}(\theta)$, MLE is equivalent to the minimization of the loss

$$\theta^\star_{\text{MLE}} = \arg\min_\theta \mathcal{L}_\mathcal{A}(\theta)\,, \tag{6.21}$$

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

while the MAP estimate (6.19) is a joint minimization of the loss and the negative log of the prior,

$$\theta^{\star}_{\text{MAP}} = \arg\min_{\theta} \left[\mathcal{L}_{\mathcal{A}}(\theta) - \log p(\theta|\mathcal{H})\right]. \tag{6.22}$$

In particular, for a generic Gaussian prior of the form $p(\theta|\mathcal{H}) \propto \exp\big(-\sum_{\mu=1}^{P} a_\mu \theta_\mu^2\big)$, the negative-log prior acts as a *regularization* term of the form $\sum_{\mu=1}^{P} a_\mu \theta_\mu^2$ that has an effect of penalizing large parameter magnitudes. Since the loss grows extensively with the size of the dataset $\mathcal{A}$ while this regularization term stays constant, when we've made sufficiently many observations, we naively expect that the prior will eventually be overwhelmed by the likelihood and that the MAP and MLE estimates will become similar.

If we are to apply these approximation methods to wide neural networks, there are certain things we need to keep in mind.[10] First of all, there is actually no single optimal value for the maximum likelihood estimation $\theta^{\star}_{\text{MLE}}$. Instead, there is a continuum of such optima, and we still have to consider a distribution over them. Importantly, such a distribution over maxima depends critically on how the maxima are obtained. For instance, it depends on the way you initialize model parameters θ_{init}, the learning algorithm used to estimate these maxima – such as gradient descent vs. *stochastic* gradient descent – and the *training hyperparameters* controlling the learning algorithm. The study of this ensemble over optima and its dependence on the initialization and training hyperparameters will more or less be the focus of the remaining chapters §7–§∞.[11]

Exact Method for Model Marginalization: Effective Theory

For the prior (6.7), we know very well that it's possible to directly integrate out the model parameters through the use of a $1/n$ expansion. Such a gigantic marginalization

[10] In §10, we'll go through how all of this works in detail.

[11] Since those following chapters will unsentimentally drop our Bayesian lens, let's interpret these different methods with fresh Bayesian eyes here.

In the *impure* Bayesian approach – that is MLE – we have an initialization distribution $p(\theta_{\text{init}})$ but no prior distribution $p(\theta|\mathcal{H})$. By construction, the prior distribution does not enter into the estimate of the impure Bayesian (6.20), but the initialization distributions (2.21) and (2.22) enter into their code to give particular realizations of networks acting as the starting points for optimization and training. Thus, such an initialization distribution induces a distribution over the resulting MLE estimates.

In the *less impure* Bayesian approach – that is MAP – we have *both* an initialization distribution $p(\theta_{\text{init}})$ *and* a prior distribution $p(\theta|\mathcal{H})$. For the former, we again use the initialization distributions (2.21) and (2.22) to provide starting points for optimization; for the latter, we typically use a Gaussian prior $p(\theta|\mathcal{H}) \propto \exp\big(-\sum_{\mu=1}^{P} a_\mu \theta_\mu^2\big)$ which, as we said, serves as a regularization term when added to the optimization objective – the loss – as per (6.22).

In the *pure* Bayesian approach – which is the focus of the rest of this chapter – there is a prior distribution $p(\theta|\mathcal{H})$, but the initialization distribution $p(\theta_{\text{init}})$ isn't needed. Pure Bayesians always integrate. What we really did with (6.6) was pick a Gaussian prior over the parameters and then adopt the same conventions for the variances as we've been using for the initialization distribution (2.21) and (2.22). We'll see in the rest of the chapter why this is sensible.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

was the focus of §4, and in writing (6.7) we already reinterpreted our effective preactivation distribution at initialization as our prior beliefs about the preactivations. For the posterior, the only hypothetical worry would be that we'd need to carry out entirely different sets of integrals. We'll show next that there is no such need. Thus, in a very real sense the most painstaking theoretical part of Bayesian inference has already been taken care of for us!

Let's continue to suppose that we've made some observations A of the true outputs $y_{i;\tilde{\alpha}} \equiv f_i(x_{\tilde{\alpha}})$ of our function $f(x)$ for a given set of inputs $x_{\mathcal{A}}$ in a subsample $\mathcal{A}$ as defined by (6.13). We now want to incorporate what we've learned from these observations in order to update our beliefs about the output values $z_{\mathcal{B}}^{(L)} \equiv \left\{ z_{i;\dot{\beta}}^{(L)} \right\}$ for a potentially different set of inputs $x_{j;\dot{\beta}} \in \mathcal{B}$ in another subsample $\mathcal{B}$.[12] Beginning with the joint prior for the network outputs over the union of both subsamples $\mathcal{D} \equiv \mathcal{A} \cup \mathcal{B}$,

$$p\left(z_{\mathcal{D}}^{(L)} \middle| \mathcal{H}\right) \equiv p\left(z_{\mathcal{A}}^{(L)}, z_{\mathcal{B}}^{(L)} \middle| \mathcal{H}\right), \tag{6.23}$$

we can set $z_{\mathcal{A}}^{(L)} \to y_{\mathcal{A}}$ and use the product rule (6.1) to condition our beliefs about $z_{\mathcal{B}}^{(L)}$ on the observed *true* values $y_{\mathcal{A}}$:

$$p\left(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)} \middle| \mathcal{H}\right) = p\left(z_{\mathcal{B}}^{(L)} \middle| y_{\mathcal{A}}, \mathcal{H}\right) p\left(y_{\mathcal{A}} \middle| \mathcal{H}\right). \tag{6.24}$$

Then, rearranging terms like we are the Reverend Thomas Bayes, we get

$$p\left(z_{\mathcal{B}}^{(L)} \middle| y_{\mathcal{A}}, \mathcal{H}\right) = \frac{p\left(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)} \middle| \mathcal{H}\right)}{p\left(y_{\mathcal{A}} \middle| \mathcal{H}\right)}. \tag{6.25}$$

Since this iteration of Bayes' rule is so important, let us be verbose and crystal clear about its interpretation: the denominator $p(y_{\mathcal{A}}|\mathcal{H})$ is the prior for the network outputs given the inputs $x_{\mathcal{A}}$ in the subsample $\mathcal{A}$, evaluated on the *fixed* observed values $y_{\mathcal{A}}$, hence it is just a *number*; the numerator $p\left(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)} \middle| \mathcal{H}\right)$ is the prior for the network outputs given the inputs $x_{\mathcal{D}}$ in the joint dataset $\mathcal{D} \equiv \mathcal{A} \cup \mathcal{B}$, evaluated on the *fixed* observed values $y_{\mathcal{A}}$ but with the network outputs $z_{\mathcal{B}}^{(L)}$ still *variable*, hence it is a *function* of the $z_{\mathcal{B}}^{(L)}$.[13]

[12] For maximal disambiguation, in this chapter we'll use sample indices of the form $\dot{\beta}$ – the Greek letter beta with a dot on top – for elements of the dataset $\mathcal{B}$ corresponding to input-output pairs for which output values from $f(x)$ are *not* observed but need instead to be *inferred*.

[13] The reason we say *given the inputs* here is that technically we should be conditioning on $x_{\mathcal{A}}$ and $x_{\mathcal{B}}$ as well. In particular, while $y_{\mathcal{A}}$ is fixed and $z_{\mathcal{B}}^{(L)}$ is completely variable in the expression for the joint prior

$$p\left(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)} \middle| \mathcal{H}\right) \equiv p\left(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)} \middle| x_{\mathcal{D}}, \mathcal{H}\right), \tag{6.26}$$

the full set of inputs $x_{\mathcal{D}} \equiv x_{\mathcal{A}} \cup x_{\mathcal{B}}$ determines the *data-dependent couplings* $g_{(L)}$ and $v_{(L)}$ – or equivalently the metric $G^{(L)}$ and the four-point vertex $V^{(L)}$ – that parameterize the output distribution. We will see how this works in more detail in the following sections.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

The numerator and denominator combine to make the posterior on the left-hand side, which is thus a function of the random variable $z_{\mathcal{B}}^{(L)}$ encoding our posterior beliefs about the plausible values of the network outputs $z_{\mathcal{B}}^{(L)}$ for the inputs $x_{\mathcal{B}}$ in $\mathcal{B}$, updated with our observations about the true values $y_{\mathcal{A}}$ of the outputs for the inputs $x_{\mathcal{A}}$ in $\mathcal{A}$. In this way, rather than performing Bayesian inference to learn about the model parameters as a way of maintaining different beliefs about the different functions $f(x;\theta)$ in our flexible set, here we simply update our beliefs about the behavior of the function $f(x)$ directly.

In this presentation of Bayes' rule (6.25), the marginalization over all the model parameters already occurred in our transition from (6.6), the prior over the parameters, to (6.7), the prior over the preactivations. The resulting posterior (6.25) is in fact exactly equivalent to what you'd get by explicitly doing a marginalization over a posterior distribution of the model parameters, e.g., as in (6.12). To see why, consider the following set of manipulations:

$$\begin{aligned} p\left(z_{\mathcal{B}}^{(L)}\middle|y_{\mathcal{A}},\mathcal{H}\right) &= \int\left[\prod_{\mu=1}^{P} d\theta_\mu\right] p\left(z_{\mathcal{B}}^{(L)},\theta\middle|y_{\mathcal{A}},\mathcal{H}\right) = \int\left[\prod_{\mu=1}^{P} d\theta_\mu\right] p\left(z_{\mathcal{B}}^{(L)}\middle|\theta,\mathcal{H}\right)\; p(\theta|y_{\mathcal{A}},\mathcal{H}) \\ &= \int\left[\prod_{\mu=1}^{P} d\theta_\mu\right] p\left(z_{\mathcal{B}}^{(L)}\middle|\theta,\mathcal{H}\right)\left[\frac{p(y_{\mathcal{A}}|\theta,\mathcal{H})\,p(\theta|\mathcal{H})}{p(y_{\mathcal{A}}|\mathcal{H})}\right] \\ &= \frac{1}{p(y_{\mathcal{A}}|\mathcal{H})}\int\left[\prod_{\mu=1}^{P} d\theta_\mu\right] p\left(y_{\mathcal{A}},z_{\mathcal{B}}^{(L)}\middle|\theta,\mathcal{H}\right)\; p(\theta|\mathcal{H}) \\ &= \frac{1}{p(y_{\mathcal{A}}|\mathcal{H})}\int\left[\prod_{\mu=1}^{P} d\theta_\mu\right] p\left(y_{\mathcal{A}},z_{\mathcal{B}}^{(L)},\theta\middle|\mathcal{H}\right) = \frac{p\left(y_{\mathcal{A}},z_{\mathcal{B}}^{(L)}\middle|\mathcal{H}\right)}{p\left(y_{\mathcal{A}}\middle|\mathcal{H}\right)}. \end{aligned} \tag{6.27}$$

The only nontrivial step is in the third line, where we reversed the factorization,

$$p\left(z_{\mathcal{A}}^{(L)},z_{\mathcal{B}}^{(L)}\middle|\theta,\mathcal{H}\right) = p\left(z_{\mathcal{A}}^{(L)}\middle|\theta,\mathcal{H}\right)\; p\left(z_{\mathcal{B}}^{(L)}\middle|\theta,\mathcal{H}\right), \tag{6.28}$$

and evaluated at $z_{\mathcal{A}}^{(L)} \to y_{\mathcal{A}}$. This factorization (6.28) says that the network outputs are *conditionally independent*, given the parameters. This is a consequence of the fact that – for a fixed set of network parameters – the output on an example $x_{\tilde{\alpha}}$ is entirely independent from the output evaluated on any other example $x_{\dot{\beta}}$, which is manifestly true for all the hypotheses that we mentioned. (If it were not, neural networks would be pretty useless in practice.) The use of Bayes' rule for the model parameters in the square brackets in the second line also makes manifest the connection between *Bayesian model fitting* (6.11) on the one hand and *Bayesian prediction* (6.12) on the other hand.

As we already alluded to, this exact method for model marginalization is closely connected with our effective theory approach to understanding neural networks. In particular, while the model parameters are always part of the definition of our neural networks, we've always had to integrate them out in the process of determining the distribution over the network outputs. In this way, our effective theory of deep learning

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

has always worked directly with the entire ensemble of network functions implied by the initialization distribution of the parameters (6.6) rather than with any *particular* network. Up until now, we've motivated this ensemble approach via the *principle of typicality*, in which we use the ensemble to analyze how a typical realization is likely to behave.[14] Here we have a slightly different interpretation: rather than trying to make the ensemble describe a typical network, we actually want to consider the posterior predictions across the full set of potential networks, each weighted according to our posterior beliefs about how plausible those predictions are.

Now, after a brief detour into Bayesian model comparison, much of the focus of §6.3 and §6.4 will be the explicit evaluation of these Bayesian predictions (6.25) for infinite- and finite-width MLPs, respectively.

6.2.2 Bayesian Model Comparison

In the context of Bayesian model fitting and Bayesian prediction, the *evidence* $p(y_{\mathcal{A}}|\mathcal{H})$ has thus far played essentially no role. In the context of our approximation methods, MAP and MLE and their respective maximizations (6.19) and (6.20), the value of the argument maximization is strictly independent of the evidence, since it doesn't depend on the model parameters. In the context of our exact method for Bayesian prediction, the evidence is simply the normalization factor of the posterior, which is trivial for us to compute.

To actually see the role of the evidence in action, *you mustn't be afraid to dream a little bigger, darling*. That is, rather than being fixated on a single hypothesis $\mathcal{H}$, we instead consider a multitude of different hypotheses $\mathcal{H}_a$ as possible explanations for our data. This is the essence of **Bayesian model comparison**: using the *evidence* to weigh the plausibility of different probabilistic models as explanations for all of our observations. In the context of deep learning, this corresponds to comparing our relative beliefs in the different modeling choices encapsulated in each $\mathcal{H}_a$ – i.e., comparing different hyperparameter settings – and determining which modeling choice provides the best description of our observations $y_{\mathcal{A}}$.

To begin, let us again use Bayes' rule – this time on the evidence – to invert the conditioning as

$$p(\mathcal{H}_a|y_{\mathcal{A}}) = \frac{p(y_{\mathcal{A}}|\mathcal{H}_a)\, p(\mathcal{H}_a)}{p(y_{\mathcal{A}})}. \tag{6.29}$$

In this form, the posterior $p(\mathcal{H}_a|y_{\mathcal{A}})$ on the left-hand side encodes our updated beliefs in the plausibility of the different hypotheses $\mathcal{H}_a$ – the different hyperparameter settings – given our observation $y_{\mathcal{A}}$, while the prior $p(\mathcal{H}_a)$ on the right-hand side encodes our initial beliefs about these hypotheses. Amusingly, the *old evidence* $p(y_{\mathcal{A}}|\mathcal{H}_a)$ for the hypothesis $\mathcal{H}_a$ from our Bayesian model fitting now appears as the *new likelihood* $p(y_{\mathcal{A}}|\mathcal{H}_a)$ of the

[14] In §8 and onwards, we'll see how this principle is manifested in neural networks *trained* via gradient-based learning.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

hypothesis $\mathcal{H}_a$ for the observation $y_\mathcal{A}$ in the context of Bayesian model comparison. Lastly, the *new evidence* $p(y_\mathcal{A})$ is just a normalization factor that we can safely ignore.[15]

To see how the model comparison works, let's use (6.29) to compare two different hypothesis, $\mathcal{H}_1$ and $\mathcal{H}_2$, in order to determine which is a better fit for our observations. Since our *relative beliefs* are all that matter, let's take the ratio of the two posteriors,

$$\frac{p(\mathcal{H}_1|y_\mathcal{A})}{p(\mathcal{H}_2|y_\mathcal{A})} = \left[\frac{p(y_\mathcal{A}|\mathcal{H}_1)}{p(y_\mathcal{A}|\mathcal{H}_2)}\right] \frac{p(\mathcal{H}_1)}{p(\mathcal{H}_2)}, \tag{6.30}$$

from which we see that the irrelevant normalization factor $p(y_\mathcal{A})$ simply drops out. Here, the ratio in the square brackets is sometimes given the name the **Bayes' factor**, which in turn multiplies the ratio of our prior beliefs. In particular, the Bayes' factor contains all of the observation dependence and characterizes how we should update our relative prior beliefs in each hypothesis given the new data $y_\mathcal{A}$. A ratio greater than one indicates that the model specified by hypothesis $\mathcal{H}_1$ is favored, while a ratio less than one indicates that the model specified by hypothesis $\mathcal{H}_2$ is favored. In this way, the old evidence – i.e., the new likelihood – $p(y_\mathcal{A}|\mathcal{H}_a)$ can be very useful, indeed.

Occam's Razor

In order to further elaborate on the mechanism behind Bayesian model comparison (6.30), let us pick up **Occam's razor** [51], which is the famous *principle of sparsity*. It says that we should favor the simplest hypothesis that fits all the observations. In the context of machine learning and parameterized probabilistic modeling, this principle is often intended as a heuristic that guides us to favor models with fewer parameters, all else being equal. The intuitive explanation for this heuristic is that models with more parameters have greater flexibility to fit the observed data, making them more likely to *overfit* and less likely to *generalize* to explain new observations.[16]

[15] Unless, of course, we aren't afraid to dream even bigger. If we did – *narrator:* they won't – we'd need to introduce a *meta hypothesis*, $\mathcal{G}$, that encodes our prior beliefs about different hyperparameter configurations $p(\mathcal{H}_a|\mathcal{G})$. This is sometimes called *Bayesian hierarchical modeling*. In this case, Bayesian model comparison in terms of this even grander evidence $p(y_\mathcal{A}) \rightarrow p(y_\mathcal{A}|\mathcal{G})$ in principle involves integrating over *all* the probabilistic models as $p(y_\mathcal{A}|\mathcal{G}) = \sum_a p(y_\mathcal{A}|\mathcal{H}_a)\ p(\mathcal{H}_a|\mathcal{G})$, i.e., any and all hypotheses $\mathcal{H}_a$ that are encoded by $\mathcal{G}$. The distinction between the meta hypothesis $\mathcal{G}$ and hypotheses $\mathcal{H}_a$ is somewhat arbitrary, however; for instance, we could put into $\mathcal{G}$ our overall choice of architecture – e.g., MLP, CNN, transformer – and then let $\mathcal{H}_a$ index the different settings of $\mathcal{H}$yperparameters. Then, recursing again, a Bayesian model comparison over $\mathcal{G}$ would be a weighted evaluation of the best architecture for the data, taking into account all possible settings of the hyperparameters for those architectures.

[16] It's natural to wonder here how to interpret this overfitting in light of the fact that we've actually integrated out all our parameters! (In the machine learning literature, such ensembles are sometimes called *nonparametric models*, though we really do not like such terminology, given the following explanation.) The culprit for this potential confusion is the overloaded usage of the word *parameter*. To illustrate this with the extreme, let's consider the infinite-width limit. Despite formally starting with an infinite number of model parameters – giving a model that is naively very *overparameterized*, to say the least – the effective theory of the output distribution is completely characterized by the kernel $K^{(L)}$,

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

Naively, Bayesian model comparison (6.29) seems to give us a very natural way to implement this razor: we can *subjectively* adjust the ratio of our prior beliefs $p(\mathcal{H}_1)/p(\mathcal{H}_2)$ to explicitly favor the simpler hypothesis, a priori penalizing more complicated models. However, as MacKay [52] points out:

> *Coherent [Bayesian] inference embodies Occam's razor automatically and quantitatively.*

That is, Occam's razor is *objectively* built into Bayesian model comparison (6.30) through the Bayes' factor.[17]

To understand why, note that the prior distribution $p\left(z_{\mathcal{A}}^{(L)}\middle|\mathcal{H}_a\right)$ needs to be normalized. This means that for a given hypothesis $\mathcal{H}_a$ to be complicated enough to explain an overwhelmingly wide variety of *potential* observations $z_{\mathcal{A}}^{(L)}$, it must have small support on any *particular* observation $y_{\mathcal{A}}$. Hence the evidence $p(y_{\mathcal{A}}|\mathcal{H}_a)$ for such a hypothesis will be small regardless of which actual observation we make. In contrast, if the hypothesis is very simple, the prior $p\left(z_{\mathcal{A}}^{(L)}\middle|\mathcal{H}_a\right)$ will make a constrained set of predictions, but make them strongly, by concentrating its support on only a few plausible outcomes. Thus, the simplest models that still correctly predict the observation $y_{\mathcal{A}}$ are naturally preferred by the Bayes' factor $p(y_{\mathcal{A}}|\mathcal{H}_1)/p(y_{\mathcal{A}}|\mathcal{H}_2)$ alone. In addition, the more observations we make that are correctly predicted, the more the Bayes' factor will amplify this preference for simpler models that still fit.[18]

Since the Bayes' factor automatically and objectively implements Occam's razor, there's no need to subjectively express a preference for simpler models using the prior over hypotheses $p(\mathcal{H}_a)$. This means that for a discrete set of hypotheses $\{\mathcal{H}_a\}$, we can choose the prior distribution to be uniform, giving equal a priori preference to any particular hypothesis $\mathcal{H}_a$ regardless of its complexity. With this choice, our Bayesian model comparison is completely characterized by the Bayes' factor:

$$\frac{p(\mathcal{H}_1|y_{\mathcal{A}})}{p(\mathcal{H}_2|y_{\mathcal{A}})} = \frac{p(y_{\mathcal{A}}|\mathcal{H}_1)}{p(y_{\mathcal{A}}|\mathcal{H}_2)}. \tag{6.31}$$

Thus, we should really think of Occam's razor as the *inductive bias* of Bayesian inference applied to model comparison.

---

which can be described by a finite number of *data-dependent couplings* $\sim N_D^2$. Thus, from the macroscopic perspective of Bayesian model comparison, it's these couplings that control the model complexity and not what we usually call the parameters, the tunable weights and biases. We will discuss this further and in greater detail in Epilogue ε, and in particular we'll highlight how the $1/n$ expansion for finite-width networks leads to a sequence of effective theories with increasing complexity.

[17] See MacKay's excellent exposition [52] for further details and examples, with a particular emphasis on (pre-deep-learning-era) neural networks.

[18] This is analogous to the way the likelihood factor will dominate the prior as observations accumulate when Bayesian model fitting.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

Inductive Bias

Given our last statement, we should clarify something that we've been informally referring to since §2.1 but now are finally ready to formally address: **inductive bias**.

Way back in §2.1, inductive biases were introduced as something implicit that is built into a neural network architecture in order that the set of functions $\{f(x;\theta)\}$ may better represent the properties of a particular dataset $\mathcal{D}$ and the function approximation task at hand. From the Bayesian perspective, inductive biases represent the a priori assumptions made about the desired function $f(x)$ before any observations are made. More broadly, both hypotheses and learning algorithms may have their own set of inductive biases; e.g., we've just pointed out that Occam's razor is an inductive bias of Bayesian inference.

Throughout §6.3 and §6.4, we'll encounter various inductive biases while performing concrete calculations for infinite- and finite-width MLPs. Here, let's consider a very simple example for illustration: suppose that a Bayesian firmly believes with *absolute certainty* that a statement B is false such that their hypothesis $\mathcal{H}_{\overline{B}}$ assigns an a priori probability of zero to this belief as $p(B|\mathcal{H}_{\overline{B}}) = 0$; then, via Bayes' rule (6.4), there's no way that the posterior on B can be updated to be anything other than zero, even if the Bayesian gathers some new information A that would serve as positive evidence for B. In this case, $\mathcal{H}_{\overline{B}}$ is clearly a bad hypothesis; its inductive bias is leading to an absurdly stubborn set of beliefs. Alternatively, if B turns out to be actually false, $\mathcal{H}_{\overline{B}}$ is a good hypothesis because it can then assign more probability to other plausibly true statements. As this *gedanken inference* illustrates, the advantage and disadvantage of an inductive bias depends on the ground truth.

Returning to our initial example in §2.1 of the inductive bias of different neural-network architectures, the advantage of one architecture over another is a highly data- and task-dependent question. In principle, we could use Bayesian model comparison (6.30) to directly compare these different architectures – MLPs, CNNs, and transformers – for different sets of observations $y_{\mathcal{A}}$ if only we knew how to compute the evidence $p(y_{\mathcal{A}}|\mathcal{H})$ for those architectures.[19] The formalism of our effective theory of deep learning as laid out in the earlier chapters is precisely a blueprint for computing such factors for different architectures as a function of a particular dataset. We encourage you to give it a try.

[19]Recall from §2.1 that CNNs (2.8) are designed to capitalize on the fact that computer vision data organizes useful information in a spatially local, translationally invariant manner. Incorporating this property into the architecture design is an inductive bias of the CNN; in particular, the assumption is that a cat is still a `cat`, even if it's shifted *up up down down left right left right BA*. The advantage of such an inductive bias as compared to MLPs should be directly encoded in a Bayes' factor $p\big(y_{\mathcal{A}}\big|\mathcal{H}_{\text{CNN}}\big)\,/\,p\big(y_{\mathcal{A}}\big|\mathcal{H}_{\text{MLP}}\big)$. This ratio should presumably be greater than one for any dataset with desired outputs $y_{\mathcal{A}}$ for which the assumption of spatial locality is a useful inductive bias.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

6.3 Bayesian Inference at Infinite Width

In this section, we'll give three lessons on Bayesian learning in the infinite-width limit. First, we'll calculate the evidence $p(y_{\mathcal{A}}|\mathcal{H})$ and see that Bayesian model comparison prefers criticality for sufficiently deep networks (§6.3.1). Then, we'll calculate the posterior distribution for the network outputs $p\left(z_{\mathcal{B}}^{(L)}|y_{\mathcal{A}},\mathcal{H}\right)$ and see that different output components are completely independent in this limit (§6.3.2). Finally, we'll calculate the posterior distribution of preactivations in the *penultimate* layer $p\left(z_{\mathcal{D}}^{(L-1)}|y_{\mathcal{A}},\mathcal{H}\right)$ and show that it's identical to the penultimate prior distribution $p\left(z_{\mathcal{D}}^{(L-1)}|\mathcal{H}\right)$, thus implying that such infinitely-wide networks lack representation learning (§6.3.3).

Before we begin, let's start with some reminiscence, recast through the lens of our new Bayesian glasses. In the infinite-width limit, the prior distribution over the network outputs is given by a simple zero-mean Gaussian distribution

$$p\left(z_{\mathcal{D}}^{(L)}\middle|\mathcal{H}\right) = \frac{1}{\sqrt{|2\pi K|^{n_L}}} \exp\left(-\frac{1}{2}\sum_{i=1}^{n_L}\sum_{\delta_1,\delta_2\in\mathcal{D}} K^{\delta_1\delta_2} z_{i;\delta_1}^{(L)} z_{i;\delta_2}^{(L)}\right), \tag{6.32}$$

with the variance $K_{\delta_1\delta_2} \equiv K_{\delta_1\delta_2}^{(L)} = K^{(L)}(x_{\delta_1}, x_{\delta_2})$ given by the kernel at the output layer – here with the layer index dropped – depending explicitly on pairs of inputs x_{δ_1} and x_{δ_2} from the dataset $\mathcal{D}$ and implicitly on the $\mathcal{H}$yperparameters C_b and C_W. Also recall that, as per our *general relativistic* conventions, the matrix $K^{\delta_1\delta_2}$ is the *inverse* of the covariance matrix $K_{\delta_1\delta_2}$

$$\sum_{\delta_2\in\mathcal{D}} K^{\delta_1\delta_2} K_{\delta_2\delta_3} = \delta^{\delta_1}{}_{\delta_3}, \tag{6.33}$$

where we are entertained by – but also apologize for – the collision of sample indices δ_1, δ_2 with the overall Kronecker delta, and further recall that $|2\pi K|$ is the determinant of the $N_{\mathcal{D}}$-by-$N_{\mathcal{D}}$ matrix $(2\pi K)_{\delta_1\delta_2}$.

6.3.1 The Evidence for Criticality

As we elaborated on in the last section, the evidence is just the prior distribution for the network outputs evaluated on the observed true output values $y_{i;\tilde{\alpha}}$ given the inputs $x_{i;\tilde{\alpha}}$ in the subsample $\mathcal{A}$:

$$p(y_{\mathcal{A}}|\mathcal{H}) = \frac{1}{\sqrt{\left|2\pi\widetilde{K}\right|^{n_L}}} \exp\left(-\frac{1}{2}\sum_{i=1}^{n_L}\sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} \widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_1} y_{i;\tilde{\alpha}_2}\right). \tag{6.34}$$

Here we've put tildes both on the sample indices $\tilde{\alpha}$ and on the kernel as well, $\widetilde{K}_{\tilde{\alpha}_1\tilde{\alpha}_2}$, in order to indicate that it's an $N_{\mathcal{A}}$-by-$N_{\mathcal{A}}$ submatrix built from the pairs of inputs

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

$(x_{\tilde{\alpha}_1}, x_{\tilde{\alpha}_2})$ in the subsample $\mathcal{A}$ of size $N_{\mathcal{A}}$. Importantly, this means that the inverse $\widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2}$ is taken with respect to the samples in the set $\mathcal{A}$ *only*,

$$\sum_{\tilde{\alpha}_2 \in \mathcal{A}} \widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2} \widetilde{K}_{\tilde{\alpha}_2\tilde{\alpha}_3} = \delta^{\tilde{\alpha}_1}_{\tilde{\alpha}_3}, \tag{6.35}$$

and in particular that $\widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2} \neq K^{\tilde{\alpha}_1\tilde{\alpha}_2}$. In other words, $\widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2}$ is *not* the same as the inverse of the kernel $K^{\delta_1\delta_2}$ on the whole dataset $\mathcal{D}$ (6.33) evaluated on the sample indices $\delta_1 = \tilde{\alpha}_1$ and $\delta_2 = \tilde{\alpha}_2$; if you'd like, flip ahead and cf. (6.53). Accordingly, the determinant $\left|2\pi\widetilde{K}\right|$ is also computed from this $N_{\mathcal{A}}$-by-$N_{\mathcal{A}}$ submatrix. The usefulness of this notation and the essentialness of this distinction will become clearer when we consider the posterior in §6.3.2.

Before we analyze the evidence (6.34) in detail, we should establish our space of hypotheses. Considering MLP architectures in the infinite-width limit, there are only three hyperparameters of relevance, the bias variance and rescaled weight variance, C_b and C_W, and the depth L. In principle, each combination of these three hyperparameters is a different hypothesis. However, in the asymptotic limit of large depth $L \gg 1$, we know from our discussion in §3.2 and our analysis in §5 that generically the kernel recursion will either exponentially lead to a *trivial fixed point* at zero $K^\star = 0$ or at infinity $K^\star = \infty$, or slowly approach a *nontrivial fixed point* at criticality.[20] Thus, for deep networks, Bayesian model comparison essentially reduces to the comparison of three different hypotheses, $\mathcal{H}_0$, $\mathcal{H}_\infty$, and $\mathcal{H}_{\text{critical}}$, corresponding to the two trivial fixed points and the one nontrivial fixed point, respectively.

Having established our space of hypotheses, let's first see how Bayesian model comparison works when we have only a single input x. In this case the kernel is just a scalar, and the evidence is simply given by

$$p(y|\mathcal{H}) = \frac{1}{\left(2\pi\widetilde{K}\right)^{\frac{n_L}{2}}} \exp\left(-\frac{1}{2\widetilde{K}} \sum_{i=1}^{n_L} y_i^2\right). \tag{6.36}$$

Here, the output norm $\sum_{i=1}^{n_L} y_i^2$ is fixed by a given function approximation task.[21] Thus all the dependence on the hyperparameters $\mathcal{H}$ is encoded in a single number: $\widetilde{K}$.

Let's start with $\mathcal{H}_\infty$, for which $\widetilde{K} \to \infty$. In this case, the argument of the exponential in (6.36) vanishes, and thus the exponential evaluates to unity, while the normalization factor in front vanishes. Therefore, the evidence will vanish polynomially:

$$p(y|\mathcal{H}_\infty) = \lim_{\widetilde{K}\to\infty} \frac{1}{\left(2\pi\widetilde{K}\right)^{\frac{n_L}{2}}} = 0. \tag{6.37}$$

[20] Yes, we know: for some activation functions there exist hyperparameter settings that lead to trivial fixed points at nonzero values of the kernel $K^\star \neq 0$. We'll eventually consider – and make a case against – such hypotheses as well, though only in a future footnote, 23, and only after first considering the details of the two-input evidence.

[21] Many common datasets for *classification* tasks employ "one-hot" true outputs in which all but one component y_i of a particular output are zero, and the remaining single component – corresponding to the *correct* class – is equal to one. For such datasets, the output norm is trivial, $\sum_{i=1}^{n_L} y_i^2 = 1$.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

In fact, in this limit the output distribution becomes an (unnormalizable) uniform distribution over all possible output norms. Next, let's consider $\mathcal{H}_0$ with $\widetilde{K} \to 0$. In this case, while the normalization factor grows polynomially, the argument in the exponent approaches negative infinity. Thus, the evidence approaches zero exponentially quickly:

$$p(y|\mathcal{H}_0) = \lim_{\widetilde{K}\to 0} \exp\left[-\frac{1}{2\widetilde{K}}\sum_{i=1}^{n_L} y_i^2 + O\Big(\log \widetilde{K}\Big)\right] = 0. \tag{6.38}$$

Indeed, recalling (2.30), the evidence (6.36) in this limit becomes a Dirac delta function,

$$p(y|\mathcal{H}_0) = \prod_{i=1}^{n_L} \delta(y_i)\,, \tag{6.39}$$

which is a fairly useless hypothesis unless all of the true outputs are the zero vector. Therefore, for generic nonzero and finite output values, the maximal evidence should lie between these two extrema. Specifically, seen as a function of $\widetilde{K}$, the evidence (6.36) peaks at $\widetilde{K} = \widetilde{K}^{(L)}(x,x) \equiv \sum_{i=1}^{n_L} y_i^2/n_L$. Our remaining hypothesis, criticality ($\mathcal{H}_{\text{critical}}$), comes the closest to realizing this maximum.

To reiterate, for a single input we just need the kernel $\widetilde{K}$ to be of order one. For deep neural networks, this is precisely the condition that we imposed in order to avoid the *exploding and vanishing kernel problem* for a single input, which we satisfied with the parallel susceptibility condition $\chi_\parallel(K^\star) = 1$. Physically, the exploding kernel gives a very flat distribution spread over a big range of output norms, yielding insubstantial evidence for any particular output norm; the vanishing kernel gives sharp support for the zero norm (6.39) and no support anywhere else. Clearly the Bayes' factor (6.31) will prefer any hypothesis that gives more focused support over reasonable output norms. In the language of our Occam's razor discussion, $\mathcal{H}_\infty$ is too complex, predicting every possible norm, while $\mathcal{H}_0$ is too simple, predicting only one particular norm. The only hypothesis that gives a finite and nonzero $\widetilde{K}$ in the deep asymptotic regime is $\mathcal{H}_{\text{critical}}$, whereat the initialization hyperparameters are tuned to satisfy $\chi_\parallel(K^\star) = 1$.[22]

Now that we see how this works, let's extend our analysis of the evidence to two inputs, with $\tilde{\alpha} = \pm$. Intuitively, we expect to find the perpendicular susceptibility condition $\chi_\perp(K^\star) = 1$ and thus demonstrate a conclusive preference for the criticality hypothesis $\mathcal{H}_{\text{critical}}$. To rediscover $\chi_\perp(K^\star) = 1$, it will be sufficient to consider the case where both inputs have the same norm:

$$\sum_{i=1}^{n_0} x_{i;+}^2 = \sum_{i=1}^{n_0} x_{i;-}^2. \tag{6.40}$$

[22] N.B. polynomially vanishing kernels give finite evidence for all practical depths. To be very pedantic about this, for such kernels – for instance, for the `tanh` – for absurdly deep networks, the truly Bayesian-optimal C_W would be ever so slightly above its critical value.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

Then, recalling our decomposition into the $\gamma^{[a]}_{\tilde{\alpha}_1\tilde{\alpha}_2}$ basis (5.15), we can write the kernel as

$$\widetilde{K}_{\tilde{\alpha}_1\tilde{\alpha}_2} = \begin{pmatrix} \widetilde{K}_{[0]} + \widetilde{K}_{[2]} & \widetilde{K}_{[0]} - \widetilde{K}_{[2]} \\ \widetilde{K}_{[0]} - \widetilde{K}_{[2]} & \widetilde{K}_{[0]} + \widetilde{K}_{[2]} \end{pmatrix}, \tag{6.41}$$

where we've used the fact that $\widetilde{K}_{[1]} = 0$ when both inputs have the same norm (6.40).

In this basis, the determinant is given by $\left|2\pi\widetilde{K}\right| = 16\pi^2\widetilde{K}_{[0]}\widetilde{K}_{[2]}$, and the inverse of the kernel is given by

$$\widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2} = \frac{1}{4\widetilde{K}_{[0]}\widetilde{K}_{[2]}} \begin{pmatrix} \widetilde{K}_{[0]} + \widetilde{K}_{[2]} & -\widetilde{K}_{[0]} + \widetilde{K}_{[2]} \\ -\widetilde{K}_{[0]} + \widetilde{K}_{[2]} & \widetilde{K}_{[0]} + \widetilde{K}_{[2]} \end{pmatrix}, \tag{6.42}$$

which in turn lets us evaluate the argument of the exponential in (6.34) as

$$\begin{aligned} \sum_{i=1}^{n_L} \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2=\pm} \widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_1} y_{i;\tilde{\alpha}_2} &= \sum_{i=1}^{n_L} \frac{1}{4\widetilde{K}_{[0]}\widetilde{K}_{[2]}} \left[\widetilde{K}_{[2]} \left(y_{i;+} + y_{i;-}\right)^2 + \widetilde{K}_{[0]} \left(y_{i;+} - y_{i;-}\right)^2\right] \\ &= \frac{\mathbb{Y}_{[0]}}{\widetilde{K}_{[0]}} + \frac{\mathbb{Y}_{[2]}}{\widetilde{K}_{[2]}}, \end{aligned} \tag{6.43}$$

where in the last equality we introduced the components

$$\mathbb{Y}_{[0]} \equiv \sum_i^{n_L} \left(\frac{y_{i;+} + y_{i;-}}{2}\right)^2, \qquad \mathbb{Y}_{[2]} \equiv \sum_i^{n_L} \left(\frac{y_{i;+} - y_{i;-}}{2}\right)^2. \tag{6.44}$$

All together, this gives a simple expression for the two-input evidence,

$$\begin{aligned} p(y_+, y_-|\mathcal{H}) &= \left(16\pi^2\widetilde{K}_{[0]}\widetilde{K}_{[2]}\right)^{-\frac{n_L}{2}} \exp\left(-\frac{\mathbb{Y}_{[0]}}{2\widetilde{K}_{[0]}} - \frac{\mathbb{Y}_{[2]}}{2\widetilde{K}_{[2]}}\right) \\ &= \left[\left(4\pi\widetilde{K}_{[0]}\right)^{-\frac{n_L}{2}} \exp\left(-\frac{\mathbb{Y}_{[0]}}{2\widetilde{K}_{[0]}}\right)\right] \times \left[\left(4\pi\widetilde{K}_{[2]}\right)^{-\frac{n_L}{2}} \exp\left(-\frac{\mathbb{Y}_{[2]}}{2\widetilde{K}_{[2]}}\right)\right]. \end{aligned} \tag{6.45}$$

Now, let's consider a generic pair of input-output pairs (x_+, y_+) and (x_-, y_-) for which both the average and the difference of the true outputs, $\mathbb{Y}_{[0]}$ and $\mathbb{Y}_{[2]}$ (6.44), are nonzero and of order one. Then, running the same argument as we did for the single-input evidence, we prefer a hypothesis that comes as close as possible to having both $\widetilde{K}_{[0]} \approx \mathbb{Y}_{[0]}/n_L = O(1)$ – from maximizing the object in the first square brackets of (6.45) – and $\widetilde{K}_{[2]} \approx \mathbb{Y}_{[2]}/n_L = O(1)$ – from maximizing the object in the second square brackets of (6.45). And, as we learned in §5, to keep both $\widetilde{K}_{[0]}$ *and* $\widetilde{K}_{[2]}$ of order one, we need to set both the critical parallel susceptibility condition $\chi_\parallel(K^\star) = 1$ *and* the critical perpendicular susceptibility condition $\chi_\perp(K^\star) = 1$.[23] Therefore, with

[23] Finally, let's consider the trivial fixed points with nonzero kernel values $K^\star \neq 0$. (This can occur, e.g., for the $K^\star = 0$ universality class, for which there exist fixed points $K^\star$ that have $\chi_\perp(K^\star) = 1$ but

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

this *evidence for criticality*, Bayesian model comparison demonstrates a full preference for $\mathcal{H}_{\text{critical}}$.[24]

Programming note: since conditioning on $\mathcal{H}$ is so deeply ingrained in our minds by now, for notational simplicity we'll re-start the suppression of this conditioning from here on out.

6.3.2 Let's Not Wire Together

Now, let's work out the full posterior distribution (6.25) at infinite width.[25] As we already have an expression for the evidence $p(y_{\mathcal{A}})$ (6.34) in the denominator, let's focus on the joint distribution $p\left(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)}\right)$ in the numerator. Recall also that to discuss the posterior we need to partition the data into two subsamples, $\mathcal{D} \equiv \mathcal{A} \cup \mathcal{B}$, one for which we have observed the true output values $y_{\mathcal{A}}$ and the other for which we are going to infer the output values.

$\chi_{\parallel}(K^{\star}) < 1$.) For this analysis, we need to relax the same-norm condition (6.40) and consider the most general form of the two-input kernel. Projecting the kernel into the $\gamma^{[a]}_{\tilde{\alpha}_1\tilde{\alpha}_2}$ basis (5.15) as

$$\widetilde{K}_{\tilde{\alpha}_1\tilde{\alpha}_2} = \begin{pmatrix} \widetilde{K}_{[0]} + \widetilde{K}_{[1]} + \widetilde{K}_{[2]} & \widetilde{K}_{[0]} - \widetilde{K}_{[2]} \\ \widetilde{K}_{[0]} - \widetilde{K}_{[2]} & \widetilde{K}_{[0]} - \widetilde{K}_{[1]} + \widetilde{K}_{[2]} \end{pmatrix} , \tag{6.46}$$

we can similarly use (5.20) to decompose the *output matrix*, $\mathbb{Y}_{\tilde{\alpha}_1\tilde{\alpha}_2} \equiv \sum_{i=1}^{n_L} y_{i;\tilde{\alpha}_1} y_{i;\tilde{\alpha}_2}$, into components

$$\mathbb{Y}_{[0]} = \sum_i \left(\frac{y_{i;+} + y_{i;-}}{2}\right)^2 , \quad \mathbb{Y}_{[1]} = \frac{\sum_i y_{i;+}^2 - \sum_i y_{i;-}^2}{2} , \quad \mathbb{Y}_{[2]} = \sum_i \left(\frac{y_{i;+} - y_{i;-}}{2}\right)^2 . \tag{6.47}$$

Then, a quick calculation shows that the evidence evaluates to

$$p(y_+, y_-|\mathcal{H}) = \left[4\pi^2(4\widetilde{K}_{[0]}\widetilde{K}_{[2]} - \widetilde{K}_{[1]}^2)\right]^{-\frac{n_L}{2}} \exp\left[-\frac{(4\widetilde{K}_{[0]}\mathbb{Y}_{[2]} + 4\widetilde{K}_{[2]}\mathbb{Y}_{[0]} - 2\widetilde{K}_{[1]}\mathbb{Y}_{[1]})}{2(4\widetilde{K}_{[0]}\widetilde{K}_{[2]} - \widetilde{K}_{[1]}^2)}\right] . \tag{6.48}$$

Now, we see from this expression that a hypothesis with $\widetilde{K}_{[1]}\mathbb{Y}_{[1]} > 0$ has improved evidence compared to the one with nonpositive $\widetilde{K}_{[1]}\mathbb{Y}_{[1]}$. In particular, if a fixed point is trivial, then the parallel perturbation $\widetilde{K}_{[1]}$ always vanishes exponentially, even if the fixed-point value of the kernel is nonvanishing, $K^{\star} \neq 0$. Thus, such a hypothesis will be disfavored compared to $\mathcal{H}_{\text{critical}}$, completing our argument.

It should be noted that for this distinction to matter, we must have a nonzero $\mathbb{Y}_{[1]}$, meaning $\sum_i y_{i;+}^2 \neq \sum_i y_{i;-}^2$. For networks used as generic function approximators – or for tasks where the network outputs are general and used downstream for other tasks – this may matter. For deep-learning tasks where all the true outputs have the same norm, this may not matter.

[24] Technically, what we've shown here is a preference for criticality in the Bayesian prior distribution. In §9.4, we'll also find a natural preference for criticality in the initialization distribution, by showing that such a tuning is necessary for controlling the exploding and vanishing *gradient* problem that arises with gradient-based learning.

[25] The form of this distribution was first worked out by Williams in [53] for one-hidden-layer networks.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

With such a data partitioning in mind, we can write out the joint distribution as

$$p\Big(y_{\mathcal{A}}, z^{(L)}_{\mathcal{B}}\Big) \tag{6.49}$$
$$= \frac{1}{\sqrt{|2\pi K|^{n_L}}} \exp\Bigg[-\frac{1}{2}\sum_{i=1}^{n_L}\Bigg(\sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} K^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_1} y_{i;\tilde{\alpha}_2} + \sum_{\tilde{\alpha}_1\in\mathcal{A},\dot{\beta}_2\in\mathcal{B}} K^{\tilde{\alpha}_1\dot{\beta}_2} y_{i;\tilde{\alpha}_1} z^{(L)}_{i;\dot{\beta}_2}$$
$$+ \sum_{\dot{\beta}_1\in\mathcal{B},\tilde{\alpha}_2\in\mathcal{A}} K^{\dot{\beta}_1\tilde{\alpha}_2} z^{(L)}_{i;\dot{\beta}_1} y_{i;\tilde{\alpha}_2} + \sum_{\dot{\beta}_1,\dot{\beta}_2\in\mathcal{B}} K^{\dot{\beta}_1\dot{\beta}_2} z^{(L)}_{i;\dot{\beta}_1} z^{(L)}_{i;\dot{\beta}_2} \Bigg)\Bigg],$$

where $K^{\tilde{\alpha}_1\tilde{\alpha}_2}$, $K^{\tilde{\alpha}_1\dot{\beta}_2}$, $K^{\dot{\beta}_1\tilde{\alpha}_2}$, and $K^{\dot{\beta}_1\dot{\beta}_2}$ are the blocks of

$$K^{\delta_1\delta_2} \equiv \begin{pmatrix} K^{\tilde{\alpha}_1\tilde{\alpha}_2} & K^{\tilde{\alpha}_1\dot{\beta}_2} \\ K^{\dot{\beta}_1\tilde{\alpha}_2} & K^{\dot{\beta}_1\dot{\beta}_2} \end{pmatrix}, \tag{6.50}$$

which is the inverse of the whole $N_{\mathcal{D}}$-by-$N_{\mathcal{D}}$ kernel matrix,

$$K_{\delta_1\delta_2} = \begin{pmatrix} \widetilde{K}_{\tilde{\alpha}_1\tilde{\alpha}_2} & K_{\tilde{\alpha}_1\dot{\beta}_2} \\ K_{\dot{\beta}_1\tilde{\alpha}_2} & K_{\dot{\beta}_1\dot{\beta}_2} \end{pmatrix}. \tag{6.51}$$

To make progress, we need to relate the submatrices in the inverse (6.50) to the submatrices in the kernel decomposition (6.51), since, recalling

$$K_{\delta_1\delta_2} \equiv \frac{1}{n_L}\sum_{i}^{n_L} \mathbb{E}\left[z^{(L)}_i(x_{\delta_1})\, z^{(L)}_i(x_{\delta_2}) \right] + O\Big(\frac{1}{n}\Big), \tag{6.52}$$

it's these blocks that are naturally defined in terms of the data.[26]

Explicitly inverting $K_{\delta_1\delta_2}$ according to the inverse formula (6.33), we find that the submatrices of (6.50) can be defined in terms of the blocks of the kernel (6.51) and the inverse submatrix $\widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2}$ on $\mathcal{A}$ as

$$K^{\tilde{\alpha}_1\tilde{\alpha}_2} \equiv \widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2} + \sum_{\tilde{\alpha}_3,\tilde{\alpha}_4\in\mathcal{A}} \sum_{\dot{\beta}_3,\dot{\beta}_4\in\mathcal{B}} \widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_3} K_{\tilde{\alpha}_3\dot{\beta}_3} \mathbb{K}^{\dot{\beta}_3\dot{\beta}_4} K_{\dot{\beta}_4\tilde{\alpha}_4} \widetilde{K}^{\tilde{\alpha}_4\tilde{\alpha}_2}, \tag{6.53}$$

$$K^{\tilde{\alpha}_1\dot{\beta}_2} \equiv -\sum_{\tilde{\alpha}_3\in\mathcal{A}} \sum_{\dot{\beta}_3\in\mathcal{B}} \widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_3} K_{\tilde{\alpha}_3\dot{\beta}_3} \mathbb{K}^{\dot{\beta}_3\dot{\beta}_2}, \tag{6.54}$$

$$K^{\dot{\beta}_1\tilde{\alpha}_2} \equiv -\sum_{\tilde{\alpha}_3\in\mathcal{A}} \sum_{\dot{\beta}_3\in\mathcal{B}} \mathbb{K}^{\dot{\beta}_1\dot{\beta}_3} K_{\dot{\beta}_3\tilde{\alpha}_3} \widetilde{K}^{\tilde{\alpha}_3\tilde{\alpha}_2}, \tag{6.55}$$

$$K^{\dot{\beta}_1\dot{\beta}_2} \equiv \mathbb{K}^{\dot{\beta}_1\dot{\beta}_2}, \tag{6.56}$$

[26] As we explained before, the over-tilde on $\widetilde{K}_{\tilde{\alpha}_1\tilde{\alpha}_2}$ indicates that it's a submatrix of the kernel evaluated on samples in the set $\mathcal{A}$, only. The inverse of that block was defined explicitly in (6.35) and is symbolized as $\widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2}$. Also note that the symmetry of the full kernel, $K_{\delta_1\delta_2} = K_{\delta_2\delta_1}$, endows a similar set of symmetries on the submatrices: $\widetilde{K}_{\tilde{\alpha}_1\tilde{\alpha}_2} = \widetilde{K}_{\tilde{\alpha}_2\tilde{\alpha}_1}$, $K_{\dot{\beta}_1\dot{\beta}_2} = K_{\dot{\beta}_2\dot{\beta}_1}$, and $K_{\dot{\beta}\tilde{\alpha}} = K_{\tilde{\alpha}\dot{\beta}}$.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

where we've had to introduce (and name a posteriori) the *posterior covariance*,

$$\mathbb{K}_{\dot\beta_1\dot\beta_2} \equiv K_{\dot\beta_1\dot\beta_2} - \sum_{\tilde\alpha_3,\tilde\alpha_4\in\mathcal{A}} K_{\dot\beta_1\tilde\alpha_3}\widetilde{K}^{\tilde\alpha_3\tilde\alpha_4}K_{\tilde\alpha_4\dot\beta_2}. \tag{6.57}$$

The expression for (6.56) is defined implicitly by taking the inverse of (6.57):

$$\sum_{\dot\beta_2\in\mathcal{B}} \mathbb{K}^{\dot\beta_1\dot\beta_2}\,\mathbb{K}_{\dot\beta_2\dot\beta_3} = \delta^{\dot\beta_1}_{\ \dot\beta_3}. \tag{6.58}$$

Since these are essential relations, let us check all the components of the inverse formula (6.33), one by one. Firstly, considering the $\delta^{\delta_1}_{\ \delta_3} \to \delta^{\tilde\alpha_1}_{\ \tilde\alpha_3}$ component, we see

$$\begin{aligned}\sum_{\delta_2\in\mathcal{D}} K^{\tilde\alpha_1\delta_2}K_{\delta_2\tilde\alpha_3} &= \sum_{\tilde\alpha_2\in\mathcal{A}} K^{\tilde\alpha_1\tilde\alpha_2}K_{\tilde\alpha_2\tilde\alpha_3} + \sum_{\dot\beta_2\in\mathcal{B}} K^{\tilde\alpha_1\dot\beta_2}K_{\dot\beta_2\tilde\alpha_3} \\ &= \sum_{\tilde\alpha_2\in\mathcal{A}} \widetilde{K}^{\tilde\alpha_1\tilde\alpha_2}\widetilde{K}_{\tilde\alpha_2\tilde\alpha_3} = \delta^{\tilde\alpha_1}_{\ \tilde\alpha_3},\end{aligned} \tag{6.59}$$

where in the first line we decomposed the sum over $\delta_2 \in \mathcal{D}$ into separate sums over $\tilde\alpha_2 \in \mathcal{A}$ and over $\dot\beta_2 \in \mathcal{B}$ according to our partitioning $\mathcal{D} = \mathcal{A} \cup \mathcal{B}$; then in going to the second line we plugged in our expressions for the inverse blocks (6.53) and (6.54); and finally in the last step we used the fact that $\widetilde{K}^{\tilde\alpha_1\tilde\alpha_2}$ is the inverse of the submatrix $\widetilde{K}_{\tilde\alpha_1\tilde\alpha_2}$ (6.35). Secondly, considering the $\delta^{\delta_1}_{\ \delta_3} \to \delta^{\dot\beta_1}_{\ \dot\beta_3}$ component, we see

$$\begin{aligned}&\sum_{\delta_2\in\mathcal{D}} K^{\dot\beta_1\delta_2}K_{\delta_2\dot\beta_3} = \sum_{\tilde\alpha_2\in\mathcal{A}} K^{\dot\beta_1\tilde\alpha_2}K_{\tilde\alpha_2\dot\beta_3} + \sum_{\dot\beta_2\in\mathcal{B}} K^{\dot\beta_1\dot\beta_2}K_{\dot\beta_2\dot\beta_3} \\ &= \sum_{\dot\beta_2\in\mathcal{B}} \mathbb{K}^{\dot\beta_1\dot\beta_2}\left(K_{\dot\beta_2\dot\beta_3} - \sum_{\tilde\alpha_3,\tilde\alpha_2\in\mathcal{A}} K_{\dot\beta_2\tilde\alpha_3}\widetilde{K}^{\tilde\alpha_3\tilde\alpha_2}K_{\tilde\alpha_2\dot\beta_3}\right) = \sum_{\dot\beta_2\in\mathcal{B}} \mathbb{K}^{\dot\beta_1\dot\beta_2}\mathbb{K}_{\dot\beta_2\dot\beta_3} = \delta^{\dot\beta_1}_{\ \dot\beta_3}\,,\end{aligned} \tag{6.60}$$

where as before in the first line we decomposed the sum over $\delta_2 \in \mathcal{D}$ into separate sums over $\tilde\alpha_2 \in \mathcal{A}$ and over $\dot\beta_2 \in \mathcal{B}$ according to our partitioning $\mathcal{D} = \mathcal{A} \cup \mathcal{B}$; then in going to the second line we plugged in our expressions for the inverse blocks (6.55) and (6.56); and finally, identifying the expression in the parenthesis as the definition of the posterior covariance $\mathbb{K}_{\dot\beta_1\dot\beta_2}$ (6.57), we get the final result since $\mathbb{K}^{\dot\beta_1\dot\beta_2}$ is the inverse of the posterior covariance (6.58). Lastly, we consider the off-diagonal block:

$$\begin{aligned}&\sum_{\delta_2\in\mathcal{D}} K^{\tilde\alpha_1\delta_2}K_{\delta_2\dot\beta_3} = \sum_{\tilde\alpha_2\in\mathcal{A}} K^{\tilde\alpha_1\tilde\alpha_2}K_{\tilde\alpha_2\dot\beta_3} + \sum_{\dot\beta_2\in\mathcal{B}} K^{\tilde\alpha_1\dot\beta_2}K_{\dot\beta_2\dot\beta_3} \\ &= \sum_{\tilde\alpha_2\in\mathcal{A},\dot\beta_2\in\mathcal{B}} \widetilde{K}^{\tilde\alpha_1\tilde\alpha_2}K_{\tilde\alpha_2\dot\beta_2}\left(\delta^{\dot\beta_2}_{\ \dot\beta_3} + \sum_{\tilde\alpha_3,\tilde\alpha_4\in\mathcal{A}}\sum_{\dot\beta_4\in\mathcal{B}} \mathbb{K}^{\dot\beta_2\dot\beta_4}K_{\dot\beta_4\tilde\alpha_4}\widetilde{K}^{\tilde\alpha_4\tilde\alpha_3}K_{\tilde\alpha_3\dot\beta_3} - \sum_{\dot\beta_4\in\mathcal{B}} \mathbb{K}^{\dot\beta_2\dot\beta_4}K_{\dot\beta_4\dot\beta_3}\right) \\ &= \sum_{\tilde\alpha_2\in\mathcal{A},\dot\beta_2\in\mathcal{B}} \widetilde{K}^{\tilde\alpha_1\tilde\alpha_2}K_{\tilde\alpha_2\dot\beta_2}\left(\delta^{\dot\beta_2}_{\ \dot\beta_3} - \sum_{\dot\beta_4\in\mathcal{B}} \mathbb{K}^{\dot\beta_2\dot\beta_4}\mathbb{K}_{\dot\beta_4\dot\beta_3}\right) = 0\,,\end{aligned} \tag{6.61}$$

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

Here, we follow the same pattern as before: *(i)* decomposing the sum according to the partitioning $\mathcal{D} = \mathcal{A} \cup \mathcal{B}$, *(ii)* plugging in expressions for inverse blocks (6.53) and (6.54), and *(iii)* using the posterior covariance (6.57) and the inverse equation (6.58). Everything checks out.

Now that we have some confidence in our inversions, let's plug our expressions for these submatrices (6.53)–(6.56) into the joint prior (6.49). Since the posterior (6.25) is only a function of the outputs $z_{\mathcal{B}}^{(L)}$, we can make things easier by limiting our focus to the $z_{\mathcal{B}}^{(L)}$ dependence only, ignoring the $y_{\mathcal{A}}$ terms independent of $z_{\mathcal{B}}^{(L)}$ and ignoring the normalization factor:

$$p\Big(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)}\Big) \propto \exp\Bigg[-\frac{1}{2}\sum_{i=1}^{n_L} \sum_{\dot{\beta}_1,\dot{\beta}_2 \in \mathcal{B}} \mathbb{K}^{\dot{\beta}_1\dot{\beta}_2} z_{i;\dot{\beta}_1}^{(L)} z_{i;\dot{\beta}_2}^{(L)} + \sum_{i=1}^{n_L} \sum_{\dot{\beta}_1 \in \mathcal{B}, \tilde{\alpha}_1 \in \mathcal{A}} z_{i;\dot{\beta}_1}^{(L)} \left(\sum_{\tilde{\alpha}_2 \in \mathcal{A}, \dot{\beta}_2 \in \mathcal{B}} \mathbb{K}^{\dot{\beta}_1\dot{\beta}_2} K_{\dot{\beta}_2\tilde{\alpha}_2} \widetilde{K}^{\tilde{\alpha}_2\tilde{\alpha}_1} \right) y_{i;\tilde{\alpha}_1} \Bigg]. \tag{6.62}$$

At this point you know what to do: completing the square – as should be second nature to you by now – and ignoring the new $z_{\mathcal{B}}^{(L)}$-independent additive constant in the exponential, you get

$$p\Big(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)}\Big) \propto \exp\Bigg[-\frac{1}{2}\sum_{i=1}^{n_L} \sum_{\dot{\beta}_1,\dot{\beta}_2 \in \mathcal{B}} \mathbb{K}^{\dot{\beta}_1\dot{\beta}_2} \left(z_{i;\dot{\beta}_1}^{(L)} - \sum_{\tilde{\alpha}_3,\tilde{\alpha}_4 \in \mathcal{A}} K_{\dot{\beta}_1\tilde{\alpha}_3} \widetilde{K}^{\tilde{\alpha}_3\tilde{\alpha}_4} y_{i;\tilde{\alpha}_4} \right) \times \left(z_{i;\dot{\beta}_2}^{(L)} - \sum_{\tilde{\alpha}_5,\tilde{\alpha}_6 \in \mathcal{A}} K_{\dot{\beta}_2\tilde{\alpha}_5} \widetilde{K}^{\tilde{\alpha}_5\tilde{\alpha}_6} y_{i;\tilde{\alpha}_6} \right) \Bigg]. \tag{6.63}$$

This distribution (6.63) is still Gaussian, with a variance given by the posterior covariance $\mathbb{K}_{\dot{\beta}_1\dot{\beta}_2}$ and a *nonzero* posterior mean:

$$m_{i;\dot{\beta}}^{\infty} \equiv \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2 \in \mathcal{A}} K_{\dot{\beta}\tilde{\alpha}_1} \widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_2}. \tag{6.64}$$

Here, the superscript ∞ is used to remind us that we're in the infinite-width limit. Finally, we realize that the posterior distribution (6.25) is proportional to the joint prior (6.63),

$$p\Big(z_{\mathcal{B}}^{(L)} \Big| y_{\mathcal{A}}\Big) \propto p\Big(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)}\Big), \tag{6.65}$$

and that the posterior distribution is automatically normalized (6.5) as a function of the variable $z_{\mathcal{B}}^{(L)}$. Thus, computing the normalization factor for (6.63) – or really just writing it down, since at this point you know by heart how to normalize any Gaussian distribution – we get the posterior at infinite width:

$$p\Big(z_{\mathcal{B}}^{(L)} \Big| y_{\mathcal{A}}\Big) = \frac{1}{\sqrt{|2\pi\mathbb{K}|^{n_L}}} \exp\left[-\frac{1}{2}\sum_{i=1}^{n_L} \sum_{\dot{\beta}_1,\dot{\beta}_2 \in \mathcal{B}} \mathbb{K}^{\dot{\beta}_1\dot{\beta}_2} \left(z_{i;\dot{\beta}_1}^{(L)} - m_{i;\dot{\beta}_1}^{\infty} \right) \left(z_{i;\dot{\beta}_2}^{(L)} - m_{i;\dot{\beta}_2}^{\infty} \right) \right]. \tag{6.66}$$

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

The *posterior mean* $m^{\infty}_{i;\dot{\beta}}$ represents our updated belief about the expected network output for the input $x_{j;\dot{\beta}} \in \mathcal{B}$ after incorporating information about the true outputs $y_{\mathcal{A}}$ for all the inputs $x_{j;\tilde{\alpha}} \in \mathcal{A}$; as such, it is explicitly a function of the true input-output pairs $x_{\mathcal{A}}$ and $y_{\mathcal{A}}$ in the subsample $\mathcal{A}$, as we see in (6.64). Importantly, our expected predictions were a priori zero – indicating an inductive bias toward vanishing outputs on average – and now a posteriori our predictions are shifted to something nonzero. Such a nonzero posterior mean is a signature that learning is (finally!) happening. In addition, the posterior covariance $\mathbb{K}_{\dot{\beta}_1\dot{\beta}_2}$ encodes the confidence interval: the smaller the covariance is, the more sharply peaked the posterior is around its mean, and the more confident the model is about its predictions.

Practically speaking, note that in order to compute the mean prediction $m^{\infty}_{i;\dot{\beta}_1}$ according to its definition (6.64), we'd in principle need to invert – and then represent – the $N_{\mathcal{A}}$-by-$N_{\mathcal{A}}$ submatrix $\widetilde{K}_{\tilde{\alpha}_1\tilde{\alpha}_2}$. As the size of our observations $N_{\mathcal{A}}$ grows, the computational cost of such an inversion grows very fast.[27] This hidden catch is why – though theoretically quite elegant – (at least any naive implementation of) Bayesian learning is not practical for large datasets. Instead, for this reason we will essentially need to rely on approximation methods for model fitting, such as MLE (6.20). We'll comment more on this in the next chapter (§7).

Theoretically *and* practically speaking, there is another serious issue with the infinite-width posterior mean. Looking at its expression (6.64), we see that the mean prediction on the output component i is entirely independent from the observations $y_{j;\alpha}$ that we made on the other components with $j \neq i$. Thus, our updated best estimate of these different output components are entirely uncorrelated, though in principle, observations of different components j may contain very useful information about a given component i.[28] In fact, we see from (6.66) that the posterior distribution actually factorizes as

$$p\Big(z^{(L)}_{i;\mathcal{B}}, z^{(L)}_{j;\mathcal{B}}\Big|y_{i;\mathcal{A}}, y_{j;\mathcal{A}}\Big) = p\Big(z^{(L)}_{i;\mathcal{B}}\Big|y_{i;\mathcal{A}}\Big)\; p\Big(z^{(L)}_{j;\mathcal{B}}\Big|y_{j;\mathcal{A}}\Big)\;, \qquad (i \neq j), \tag{6.67}$$

meaning that the different output components are entirely statistically independent.[29]

[27] For instance, the computational cost of *Gauss–Jordan elimination* scales as $\sim N^3_{\mathcal{A}}$ and requires us to represent the $N_{\mathcal{A}} \times N_{\mathcal{A}}$-dimensional inverse in memory. Things can be improved a bit by realizing that to compute the posterior mean we only really require the *matrix–vector product* of the inverse with the observations: $\sum_{\tilde{\alpha}_2 \in \mathcal{A}} \widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_2}$. However, such an improvement is still not really sufficient for Bayesian learning to compete practically with gradient-based learning for large datasets $\mathcal{A}$.

[28] The concept of *knowledge distillation* [54] is predicated on this principle of correlations among the output components. For example, if a network is trying to classify images of hand-written digits, a certain example of a "2" may be more "7"-like or more "3"-like. Such feature information is quite useful, especially if the output of the network is used downstream for some other task.

[29] To be FAIR, the issue is with the infinite-width limit itself, as different output components are also decorrelated for infinite-width networks trained with gradient-based learning (§10).

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

We can trace this independence back to a similar property of the infinite-width prior distribution

$$p\left(z_{i;\mathcal{A}}^{(L)}, z_{j;\mathcal{A}}^{(L)}\right) = p\left(z_{i;\mathcal{A}}^{(L)}\right) p\left(z_{j;\mathcal{A}}^{(L)}\right) , \qquad (i \neq j), \tag{6.68}$$

a property that we've recognized for a while now – see, e.g., (5.106). Thus, with Bayesian learning, output features do not *wire together*: recalling our discussion of inductive bias before (§6.2.2), we see that the prior endows on the posterior an absurdly stubborn set of beliefs, namely that the components of the output are completely independent with *absolute certainty*. Such an inductive bias is incurable by any amount of learning, regardless of how large the set of observations $\mathcal{A}$ is; the inductive bias of this prior can never be overwhelmed in the infinite-width limit.

Luckily, this state of affairs is completely curable – for both learning algorithms, Bayesian learning and gradient-based learning – by backing off of the infinite-width limit and working with finite-width networks ... the actual kinds of networks that are used in practice.

6.3.3 Absence of Representation Learning

Considering the independence of the different components of the output in the posterior, a natural follow-up question is whether or not Bayesian learning at infinite width enables representation learning. Here, we will show decisively that it does *not*.

As a representative avatar of this question, let's compute the *posterior* distribution of preactivations in the penultimate layer $\ell = L - 1$ on the full set of samples $\mathcal{D}$, given observations $y_{\mathcal{A}}$:

$$p\left(z_{\mathcal{D}}^{(L-1)} \middle| y_{\mathcal{A}}\right) = \frac{p\left(y_{\mathcal{A}} \middle| z_{\mathcal{D}}^{(L-1)}\right) p\left(z_{\mathcal{D}}^{(L-1)}\right)}{p(y_{\mathcal{A}})}. \tag{6.69}$$

This is an application of Bayes' rule (6.4), following from applying the product rule (6.1) to the joint distribution $p\left(y_{\mathcal{A}}, z_{\mathcal{D}}^{(L-1)}\right)$ between the observations $y_{\mathcal{A}}$ and the penultimate preactivations $z_{\mathcal{D}}^{(L-1)}$. Here, the likelihood $p\left(y_{\mathcal{A}} | z_{\mathcal{D}}^{(L-1)}\right)$ is the conditional distribution $p\left(z_{\mathcal{A}}^{(L)} | z_{\mathcal{D}}^{(L-1)}\right)$ evaluated on our set of observations $z_{\mathcal{A}}^{(L)} \to y_{\mathcal{A}}$.

We already know the form of this conditional distribution, as it is the same object (4.69) that we needed in order to work out the layer-to-layer RG flow of the preactivations. In general, this distribution involves the *stochastic* metric $\widehat{G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2} = \widehat{G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}\left(z_{\mathcal{D}}^{(L-1)}\right)$. However, in the infinite-width limit the metric is entirely *deterministic* $\widehat{G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2} \to G^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}$, with no dependence at all on the penultimate-layer preactivations $z_{\mathcal{D}}^{(L-1)}$. Thus, the likelihood at infinite width – swapping the deterministic metric for the kernel – is given by

$$p\left(y_{\mathcal{A}} \middle| z_{\mathcal{D}}^{(L-1)}\right) = \frac{1}{\sqrt{\left|2\pi \widetilde{K}^{(L)}\right|^{n_L}}} \exp\left(-\frac{1}{2}\sum_{i=1}^{n_L} \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2 \in \mathcal{A}} \widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_2}_{(L)} y_{i;\tilde{\alpha}_1} y_{i;\tilde{\alpha}_2}\right) = p(y_{\mathcal{A}}) , \tag{6.70}$$

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

and our expression for the posterior of the penultimate layer (6.69) reduces to the prior:

$$p\left(z_{\mathcal{D}}^{(L-1)}\middle|y_{\mathcal{A}}\right) = p\left(z_{\mathcal{D}}^{(L-1)}\right). \tag{6.71}$$

Since the posterior equals the prior, our observation of $y_{\mathcal{A}}$ had no consequence on the penultimate-layer representation; thus, we conclude that there is no representation learning at infinite width.

This lack of representation learning stems from the lack of interlayer correlation in the joint distribution $p\left(z_{\mathcal{D}}^{(\ell)}, z_{\mathcal{D}}^{(\ell+1)}\right)$ at infinite width, and thus it persists for all hidden layers with $\ell < L$. This is another bad inductive bias of the infinite-width hypotheses: regardless of the set of observations $y_{\mathcal{A}}$ that we make, there's no amount of new information that will allow the network to update its representations in the hidden layers $\ell < L$.

This state of affairs is somewhat tragic as the whole point of having many layers – in fact, the main motivation given for deep learning on the whole – is the learning of complex representations in those hidden layers. As we will see next, we can solve this lack of representation learning – as well as the lack of *wiring together* in the output – by backing off the infinite-width limit and looking at finite-width effects.[30]

6.4 Bayesian Inference at Finite Width

In this section, we'll give three lessons on Bayesian learning at finite width. To begin, we'll show that finite-width neural networks are automatically endowed with an inductive bias for neural association due to non-Gaussian interactions between neurons, leading to a natural predisposition toward Hebbian learning (§6.4.1). With that in mind, we'll in turn demonstrate how such learning works by first calculating the mean of the posterior distribution for the network outputs $p\left(z_{\mathcal{B}}^{(L)}|y_{\mathcal{A}}\right)$ – showing how *intralayer* neural interactions in the prior give rise to nontrivial correlations among the components of the output (§6.4.2) – and then calculating the posterior distribution of preactivations in the penultimate layer $p\left(z_{\mathcal{B}}^{(L-1)}|y_{\mathcal{A}}\right)$ – showing how *interlayer* interactions give rise to a nonzero shift between prior and posterior, thus signaling the presence of representation learning at finite width (§6.4.3).

6.4.1 Hebbian Learning, Inc.

In this subsection, we'll see that finite-width neural networks have an inductive bias that facilitates *neural association*. To explain **Hebbian learning**, let's begin first with a few words from our honorary guest speaker, Donald Hebb:

[30] In §10, we will also show that the same lack of representation learning occurs for an ensemble of infinite-width networks that are (theoretically) trained with gradient-based learning. This issue is also resolved (practically) in §11 by going to finite width.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

> *The general idea is an old one, that any two cells or systems of cells that are repeatedly active at the same time will tend to become "associated," so that activity in one facilitates activity in the other.*
>
> Donald Hebb, in his 1949 classic *The Organization of Behavior* [55].

(*Applause.*)

> *Thank you very much.*
>
> Donald Hebb, apocryphal.

While Hebb was originally thinking about biological neurons, Hebbian learning has become a popular guiding principle for systems of artificial neurons as well. We've actually already seen this inductive bias for neural association any of the numerous times we've discussed the presence of neural interactions in the finite-width prior distribution. To make this manifest, we're now going to explicitly determine the neural influence of one preactivation on another in our effective preactivation distribution at initialization.

Concretely, let's suppose that a single input x is fed into a network, and we've checked that at layer ℓ the value of the first preactivation $z_1^{(\ell)} = \check{z}_1^{(\ell)}$ is larger than typical; given this atypical value $\check{z}_1^{(\ell)}$, we can then ask whether the second preactivation $z_2^{(\ell)}$ is likely to be atypically large. This kind of neural association or influence is encoded in the conditional distribution

$$p\left(z_2^{(\ell)} \middle| \check{z}_1^{(\ell)}\right) = \frac{p\left(\check{z}_1^{(\ell)}, z_2^{(\ell)}\right)}{p\left(\check{z}_1^{(\ell)}\right)}. \tag{6.72}$$

Note that at infinite width, $p\left(z_2^{(\ell)} \middle| \check{z}_1^{(\ell)}\right) = p\left(z_2^{(\ell)}\right)$ due to the factorization of the prior on neurons (5.106), and so we see right away that there is a complete absence of neural association in such a limit.

To compute this association for finite-width networks, recall from §4.4 the action representation (4.97) for a distribution over m neurons,

$$p(z_1, \ldots, z_m) \propto \exp\left(-\frac{g_m}{2}\sum_{i=1}^{m} z_i^2 + \frac{v}{8}\sum_{i,j=1}^{m} z_i^2 z_j^2\right), \tag{6.73}$$

where we have temporarily dropped layer indices from the variables and couplings. Here, the quadratic coupling g_m is given implicitly by the expression (4.102),

$$\frac{1}{g_m} = G^{(\ell)} - \frac{(m+2)}{2n_{\ell-1}}\frac{V^{(\ell)}}{G^{(\ell)}} + O\left(\frac{1}{n^2}\right), \tag{6.74}$$

and we have emphasized the dependence of the coupling on m; similarly, the quartic coupling is given by (4.103),

$$v = \frac{1}{n_{\ell-1}}\frac{V^{(\ell)}}{\left(G^{(\ell)}\right)^4} + O\left(\frac{1}{n^2}\right), \tag{6.75}$$

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

which is independent of m to this order in $1/n$. Evaluating the action representation (6.73) on $m = 1$ and $m = 2$ neurons and plugging the resulting distributions into our expression for the conditional distribution (6.72), we get

$$p(z_2|\check{z}_1) \propto \exp\left[-\frac{g_2}{2}z_2^2 + \frac{v}{8}\left(z_2^4 + 2z_2^2\check{z}_1^2\right)\right], \tag{6.76}$$

where, similar to the last section, for such a conditional distribution we only need to keep track of the terms in the action that depend on z_2.

Now that we have a conditional distribution, let's evaluate some conditional expectations. Since this distribution is manifestly even in z_2, i.e., invariant under a sign flip $z_2 \leftrightarrow -z_2$, all the odd-point correlators vanish, including the conditional mean. This means that the first nontrivial observable is the two-point correlator or conditional variance:

$$\begin{aligned}\int dz_2\; p(z_2|\check{z}_1)\, z_2^2 &= \frac{\int dz_2\;\exp\left[-\frac{g_2}{2}z_2^2 + \frac{v}{8}\left(z_2^4 + 2z_2^2\check{z}_1^2\right)\right] z_2^2}{\int dz_2\;\exp\left[-\frac{g_2}{2}z_2^2 + \frac{v}{8}\left(z_2^4 + 2z_2^2\check{z}_1^2\right)\right]} \\ &= \frac{\int dz_2\; e^{-\frac{g_2 z_2^2}{2}}\left[z_2^2 + \frac{v}{8}\left(z_2^6 + 2z_2^4\check{z}_1^2\right) + O(v^2)\right]}{\int dz_2\; e^{-\frac{g_2 z_2^2}{2}}\left[1 + \frac{v}{8}\left(z_2^4 + 2z_2^2\check{z}_1^2\right) + O(v^2)\right]} \\ &= \frac{g_2^{-1} + \frac{v}{8}\left(15g_2^{-3} + 6g_2^{-2}\check{z}_1^2\right)}{1 + \frac{v}{8}\left(3g_2^{-2} + 2g_2^{-1}\check{z}_1^2\right)} + O\left(v^2\right) \\ &= g_2^{-1} + \frac{v}{2}g_2^{-2}\left(3g_2^{-1} + \check{z}_1^2\right) + O\left(v^2\right).\end{aligned} \tag{6.77}$$

Above, on the first line we used (6.76) in the numerator and at the same time computed its normalization in the denominator, on the second line we expanded both the numerator and denominator in v, on the third line we computed the single-variable Gaussian integrals, and on the final line we expanded the denominator in v. Plugging in our expressions for the quadratic coupling (6.74) and the quartic coupling (6.75) and reimplementing layer indices, we find

$$\int dz_2^{(\ell)}\; p\left(z_2^{(\ell)}\middle|\check{z}_1^{(\ell)}\right)\left(z_2^{(\ell)}\right)^2 = G^{(\ell)} + \frac{1}{2}\left[\left(\check{z}_1^{(\ell)}\right)^2 - G^{(\ell)}\right]\left[\frac{V^{(\ell)}}{n_{\ell-1}\left(G^{(\ell)}\right)^2}\right] + O\left(\frac{1}{n^2}\right). \tag{6.78}$$

In passing, note for later that this result holds for any distinct pair of neurons by replacing neural indices as $1, 2 \to i_1, i_2$, with $i_1 \neq i_2$.

This conditional variance (6.78) embodies some really interesting physics. If the observed value $\left(\check{z}_1^{(\ell)}\right)^2$ is larger/smaller than its expected value $\mathbb{E}\left[\left(z_1^{(\ell)}\right)^2\right] = G^{(\ell)}$, then the variance of $z_2^{(\ell)}$ will itself be larger/smaller than is typical. Thus, $z_1^{(\ell)}$ and $z_2^{(\ell)}$ correlate

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

their atypical firing.[31] This effect is proportional to the normalized four-point vertex in the second square brackets of (6.78), which as we know from (5.128) and (5.129) is proportional to ℓ/n across our universality classes when at criticality. In other words, deeper layers have an inductive bias to build more neural associations. Moreover, the presence of these associations is mediated by the interactions in the effective action induced at finite width *only*. As we will soon show, nontrivial representation learning is a direct descendant of such associations.

Note that this result should be interpreted as a *propensity* for atypicality rather than a *guarantee*. Since the conditional variance (6.78) applies to any pair of neurons, conditioned on a particular neuron i_* having a larger/smaller norm than expected, then all of the other neurons with $i \neq i_*$ are more likely to have a larger/smaller norm, though not all will. In a given realization of a network in practice, the ones that happen to have a larger/smaller norm are the ones that are more likely to develop a correlation with i_* as learning progresses.

Hebbian learning is often summarized by the following slogan: *neurons that fire together, wire together*. What we see here is that conditioned on an atypical firing $\check{z}_1$, another preactivation, e.g., z_2, is much more likely to have an atypical firing itself. This propensity of finite-width networks to *fire together* is an inductive bias of our prior beliefs before Bayesian learning as well as of our initialization distribution before gradient-based learning. To understand the *wire together* part, let's now consider the Bayesian posterior.[32]

6.4.2 Let's Wire Together

Let's start with some more reminiscing through our now well-adjusted Bayesian lens. Recall from (4.80) that the prior distribution over preactivations is nearly-Gaussian at large-but-finite width:

[31] You may or may not recall from footnote 8 in §1.2 that having a nontrivial connected four-point correlator serves as a measure of the potential for outliers. In statistics, for single-variable distributions this is called the *excess kurtosis*; here, we see a multi-neuron generalization (which apparently can be called the *cokurtosis*). In particular, observing an outlying value $z_1^{(\ell)} = \check{z}_1^{(\ell)}$ implies that we are more likely to see outlying values for $z_2^{(\ell)}$ as well. At the end of Appendix A, we'll provide an information-theoretic reformulation of this phenomenon that will also shed further light on how deep a network should be in order to best take advantage of it.

[32] For some models of artificial neurons – such as the Hopfield network – Hebbian learning is often added in *by hand*. For instance, one learning rule for such networks that explicitly implements the Hebbian principle is updating the weights connecting two neurons i and j as $W_{ij} \propto z_i(x) z_j(x)$ when observing activities $z_i(x)$ and $z_j(x)$ for a given input x.

In contrast, any finite-width feedforward neural network should automatically incorporate Hebbian learning *by nature*. To underscore this point further, in §∞ we'll perform an analogous computation for a gradient-descent update. Since the prior has the same form as the initialization distribution, we expect that all learned finite-width networks will inc. the Hebbian learning principle automatically, regardless of whether that learning is Bayesian or gradient-based.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

$$p\Big(z_{\mathcal{D}}^{(L)}\Big) \propto \exp\Bigg[-\frac{1}{2}\sum_{j=1}^{n_L}\sum_{\delta_1,\delta_2\in\mathcal{D}} g^{\delta_1\delta_2} z_{j;\delta_1}^{(L)} z_{j;\delta_2}^{(L)} + \frac{1}{8}\sum_{j,k=1}^{n_L}\sum_{\delta_1,\ldots,\delta_4\in\mathcal{D}} v^{(\delta_1\delta_2)(\delta_3\delta_4)} z_{j;\delta_1}^{(L)} z_{j;\delta_2}^{(L)} z_{k;\delta_3}^{(L)} z_{k;\delta_4}^{(L)} + \cdots \Bigg]. \tag{6.79}$$

As a reminder, the quadratic coupling $g^{\delta_1\delta_2} \equiv g_{(L)}^{\delta_1\delta_2}$ (4.81) and the quartic coupling $v^{(\delta_1\delta_2)(\delta_3\delta_4)} \equiv v_{(L)}^{(\delta_1\delta_2)(\delta_3\delta_4)}$ (4.82) depend explicitly on groups of inputs from the dataset $\mathcal{D}$ and implicitly on the $\mathcal{H}$yperparameters C_b and C_W, the widths n_ℓ, and the depth L. As a consequence of the nonzero *intralayer* interaction between different output preactivations in the prior, there will be nonvanishing correlations between the components of the network outputs in the posterior.

As we did at infinite width, we'll start with the prior distribution (6.79) and then obtain the posterior distribution $p\Big(z_{\mathcal{B}}^{(L)}|y_{\mathcal{A}}\Big) \propto p\Big(y_{\mathcal{A}}, z_{\mathcal{B}}^{(L)}\Big)$ by plugging in our observations $z_{i;\tilde{\alpha}}^{(L)} \to y_{i;\tilde{\alpha}}$ and keeping track of the dependence on the remaining variables $z_{i;\dot{\beta}}^{(L)}$. For the quadratic term in the action, with exactly the same set of manipulations as we did in the infinite-width limit (§6.3.2), replacing the inverse kernel with the quadratic coupling at finite width $K^{\delta_1\delta_2} \to g^{\delta_1\delta_2}$, we find

$$\begin{aligned} &\frac{1}{2}\sum_{j=1}^{n_L}\sum_{\delta_1,\delta_2\in\mathcal{D}} g^{\delta_1\delta_2} z_{j;\delta_1}^{(L)} z_{j;\delta_2}^{(L)}\Big|_{z_{i;\tilde{\alpha}}^{(L)}=y_{i;\tilde{\alpha}}} \\ &\quad = \text{constant} + \frac{1}{2}\sum_{j=1}^{n_L}\sum_{\dot{\beta}_1,\dot{\beta}_2\in\mathcal{B}} \mathbb{G}^{\dot{\beta}_1\dot{\beta}_2}\left(z_{j;\dot{\beta}_1}^{(L)} - m_{j;\dot{\beta}_1}\right)\left(z_{j;\dot{\beta}_2}^{(L)} - m_{j;\dot{\beta}_2}\right), \end{aligned} \tag{6.80}$$

with the *naive* posterior mean

$$m_{i;\dot{\beta}} \equiv \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} g_{\dot{\beta}\tilde{\alpha}_1}\widetilde{g}^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_2} \tag{6.81}$$

and the *naive* posterior covariance

$$\mathbb{G}_{\dot{\beta}_1\dot{\beta}_2} \equiv g_{\dot{\beta}_1\dot{\beta}_2} - \sum_{\tilde{\alpha}_3,\tilde{\alpha}_4\in\mathcal{A}} g_{\dot{\beta}_1\tilde{\alpha}_3}\widetilde{g}^{\tilde{\alpha}_3\tilde{\alpha}_4} g_{\tilde{\alpha}_4\dot{\beta}_2}. \tag{6.82}$$

We say *naive* here because there are additional corrections we need to consider coming from the the quartic term in the action. Let's see explicitly how this works for the posterior mean.

Given the observed true outputs $y_{i;\tilde{\alpha}}$ and the quadratic term (6.80) centered at the naive posterior mean $m_{i;\dot{\beta}}$, it is natural to center ourselves at

$$\Phi_{i;\delta} \equiv \Big(y_{i;\tilde{\alpha}}\ ,\ m_{i;\dot{\beta}}\Big) = \Big(y_{i;\tilde{\alpha}}\ ,\ \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} g_{\dot{\beta}\tilde{\alpha}_1}\widetilde{g}^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_2}\Big) \tag{6.83}$$

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

and define a fluctuating variable $w_{i;\dot\beta} \equiv z^{(L)}_{i;\dot\beta} - m_{i;\dot\beta}$ so that we can plug the decomposition

$$z^{(L)}_{i;\delta} = \left(z^{(L)}_{i;\tilde\alpha}\ ,\ z^{(L)}_{i;\dot\beta}\right) \to \left(y_{i;\tilde\alpha}\ ,\ m_{i;\dot\beta} + w_{i;\dot\beta}\right) = \Phi_{i;\delta} + \left(0,\ w_{i;\dot\beta}\right) \tag{6.84}$$

into the action (6.79), thus making the partitioning into subsamples $\mathcal{D} = \mathcal{A} \cup \mathcal{B}$ manifest. In terms of this fluctuation, the quadratic term (6.80) takes the form

$$\text{constant} + \frac{1}{2}\sum_{j=1}^{n_L} \sum_{\dot\beta_1,\dot\beta_2 \in \mathcal{B}} \mathbb{G}^{\dot\beta_1\dot\beta_2} w_{j;\dot\beta_1} w_{j;\dot\beta_2}, \tag{6.85}$$

and the quartic term can be evaluated as

$$\begin{aligned}
\mathcal{Q}(w) \equiv & \left[\frac{1}{8}\sum_{j,k=1}^{n_L} \sum_{\delta_1,\ldots,\delta_4 \in \mathcal{D}} v^{(\delta_1\delta_2)(\delta_3\delta_4)} z^{(L)}_{j;\delta_1} z^{(L)}_{j;\delta_2}\ z^{(L)}_{k;\delta_3} z^{(L)}_{k;\delta_4}\right]\Bigg|_{z^{(L)}_{i;\tilde\alpha}=\Phi_{i;\tilde\alpha};\ z^{(L)}_{i;\dot\beta}=\Phi_{i;\dot\beta}+w_{i;\dot\beta}} \\
= & \ \text{constant} + \frac{4}{8}\sum_j \sum_{\dot\beta_1 \in \mathcal{B}} w_{j;\dot\beta_1} \left(\sum_k \sum_{\delta_1,\delta_2,\delta_3 \in \mathcal{D}} v^{(\dot\beta_1\delta_1)(\delta_2\delta_3)} \Phi_{j;\delta_1}\Phi_{k;\delta_2}\Phi_{k;\delta_3}\right) \\
& + \frac{2}{8}\sum_j \sum_{\dot\beta_1,\dot\beta_2 \in \mathcal{B}} w_{j;\dot\beta_1} w_{j;\dot\beta_2} \left(\sum_k \sum_{\delta_1,\delta_2 \in \mathcal{D}} v^{(\dot\beta_1\dot\beta_2)(\delta_1\delta_2)} \Phi_{k;\delta_1}\Phi_{k;\delta_2}\right) \\
& + \frac{4}{8}\sum_{j,k} \sum_{\dot\beta_1,\dot\beta_2 \in \mathcal{B}} w_{j;\dot\beta_1} w_{k;\dot\beta_2} \left(\sum_{\delta_1,\delta_2 \in \mathcal{D}} v^{(\dot\beta_1\delta_1)(\dot\beta_2\delta_2)} \Phi_{j;\delta_1}\Phi_{k;\delta_2}\right) \\
& + \frac{4}{8}\sum_{j,k} \sum_{\dot\beta_1,\dot\beta_2,\dot\beta_3 \in \mathcal{B}} w_{j;\dot\beta_1} w_{j;\dot\beta_2} w_{k;\dot\beta_3} \left(\sum_{\delta_1 \in \mathcal{D}} v^{(\dot\beta_1\dot\beta_2)(\dot\beta_3\delta)} \Phi_{k;\delta_1}\right) \\
& + \frac{1}{8}\sum_{j,k} \sum_{\dot\beta_1,\ldots,\dot\beta_4 \in \mathcal{B}} w_{j;\dot\beta_1} w_{j;\dot\beta_2}\ w_{k;\dot\beta_3} w_{k;\dot\beta_4} \left(v^{(\dot\beta_1\dot\beta_2)(\dot\beta_3\dot\beta_4)}\right).
\end{aligned} \tag{6.86}$$

Given all these expressions, we can finally determine the true posterior mean by computing the following expectation:

$$\begin{aligned}
\int dz^{(L)}_{\mathcal{B}}\, p\left(z^{(L)}_{\mathcal{B}}\middle|y_{\mathcal{A}}\right) z^{(L)}_{i;\dot\beta} &= \int dz^{(L)}_{\mathcal{B}} \frac{p\left(y_{\mathcal{A}}, z^{(L)}_{\mathcal{B}}\right)}{p\left(y_{\mathcal{A}}\right)} z^{(L)}_{i;\dot\beta} \\
&= m_{i;\dot\beta} + \int dw_{\mathcal{B}}\, \frac{p(y_{\mathcal{A}},\ m_{\mathcal{B}} + w_{\mathcal{B}})}{p(y_{\mathcal{A}})} w_{i;\dot\beta} \\
&= m_{i;\dot\beta} + \frac{\left\langle\!\left\langle w_{i;\dot\beta} e^{\mathcal{Q}(w)}\right\rangle\!\right\rangle_{\mathbb{G}}}{\left\langle\!\left\langle e^{\mathcal{Q}(w)}\right\rangle\!\right\rangle_{\mathbb{G}}} \\
&= m_{i;\dot\beta} + \frac{\left\langle\!\left\langle w_{i;\dot\beta}\left[1 + \mathcal{Q}(w)\right]\right\rangle\!\right\rangle_{\mathbb{G}}}{\left\langle\!\left\langle 1 + \mathcal{Q}(w)\right\rangle\!\right\rangle_{\mathbb{G}}} + O\left(v^2\right) \\
&= m_{i;\dot\beta} + \left\langle\!\left\langle w_{i;\dot\beta}\mathcal{Q}(w)\right\rangle\!\right\rangle_{\mathbb{G}} + O\left(v^2\right),
\end{aligned} \tag{6.87}$$

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

where on the first line we used Bayes' rule for the posterior (6.25), on the second line we inserted our decomposition (6.84) in two places, on the third line we separated out the quartic term in order to rewrite the posterior expectation as a Gaussian expectation with respect to the naive posterior covariance $\mathbb{G}$ divided by the distribution's normalization, on the fourth line we expanded the exponential, and on the final line we used the fact that the fluctuation has zero mean $\left\langle\!\left\langle w_{i;\dot{\beta}} \right\rangle\!\right\rangle_{\mathbb{G}} = 0$ in Gaussian expectation. We can now evaluate the remaining Gaussian expectation by plugging in our expression for the quartic term (6.86) and making Wick contractions:

$$
\begin{aligned}
& m_{i;\dot{\beta}} + \left\langle\!\left\langle w_{i;\dot{\beta}} \mathcal{Q}\left(w\right) \right\rangle\!\right\rangle_{\mathbb{G}} \\
& \quad = m_{i;\dot{\beta}} + \frac{1}{2} \sum_{\dot{\beta}_1 \in \mathcal{B}} \mathbb{G}_{\dot{\beta}\dot{\beta}_1} \left(\sum_{k} \sum_{\delta_1,\delta_2,\delta_3 \in \mathcal{D}} v^{(\dot{\beta}_1\delta_1)(\delta_2\delta_3)} \Phi_{i;\delta_1} \Phi_{k;\delta_2} \Phi_{k;\delta_3} \right) \\
& \qquad + \frac{1}{2} \sum_{\dot{\beta}_1,\dot{\beta}_2,\dot{\beta}_3 \in \mathcal{B}} \left(n_L \mathbb{G}_{\dot{\beta}_1\dot{\beta}_2} \mathbb{G}_{\dot{\beta}\dot{\beta}_3} + 2 \mathbb{G}_{\dot{\beta}\dot{\beta}_1} \mathbb{G}_{\dot{\beta}_2\dot{\beta}_3} \right) \left(\sum_{\delta_1 \in \mathcal{D}} v^{(\dot{\beta}_1\dot{\beta}_2)(\dot{\beta}_3\delta_1)} \Phi_{i;\delta_1} \right) .
\end{aligned}
\tag{6.88}
$$

Thus, we see that the naive posterior mean (6.81) is further corrected by a number of v-dependent terms.

To extract some physics from this complicated expression, note from the definition of Φ (6.83) that the i-th component of $\Phi_{i;\delta}$ depends on the i-th component of our observation $y_{i;\tilde{\alpha}}$. This in particular means that the term above $\propto \sum_k \Phi_{i;\delta_1} \Phi_{k;\delta_2} \Phi_{k;\delta_3}$ does incorporate information from all of the components of the observed true outputs. In other words, information from the k-th component of the observed outputs successfully influences the posterior mean prediction on the i-th component for $i \neq k$. This means that at finite width we have a dependence among the components of the posterior outputs,

$$
p\Big(z^{(L)}_{i;\mathcal{B}}, z^{(L)}_{k;\mathcal{B}} \Big| y_{i;\mathcal{A}}, y_{k;\mathcal{A}}\Big) \neq p\Big(z^{(L)}_{i;\mathcal{B}} \Big| y_{i;\mathcal{A}}\Big)\, p\Big(z^{(L)}_{k;\mathcal{B}} \Big| y_{k;\mathcal{A}}\Big) \, . \tag{6.89}
$$

This property of the posterior distribution descends from the nontrivial *fire-together* inductive bias $p\Big(z^{(L)}_i \Big| z^{(L)}_k\Big)$ present in the finite-width prior as discussed in §6.4.1. The dependence among the components of the posterior outputs (6.89) is a signature of our posterior beliefs' learning to *wire together*, and we will see a further manifestation of this when we again consider representation learning in the next section.

Before we move on, we should address practical matters. Practically speaking, it is even more computationally infeasible to evaluate the finite-width predictions of Bayesian learning (6.87) than it was at infinite width. In particular, evaluating the quartic coupling involves first *representing* the four-point vertex – an $N_{\mathcal{A}} \times N_{\mathcal{A}} \times N_{\mathcal{A}} \times N_{\mathcal{A}}$-dimensional tensor – and then multiply contracting it with inverse kernels. Thus, both the cost of computation and the memory requirements of Bayesian learning grow terrifyingly quickly with our observations, i.e., with size of our dataset $\mathcal{A}$. However, please *Don't Panic*: we are getting ever closer to the point where we can show you how gradient-based learning resolves all these practical difficulties at finite width.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

6.4.3 Presence of Representation Learning

The fact that the individual components of the finite-width posterior mean prediction can incorporate information from our observations of the other components is suggestive of the idea that these observations might also be used to build up representations in the hidden layers. Here we will show that such *representation learning* actually does occur at finite width as a direct consequence of the nonzero *interlayer* interactions.

Analogous to our parallel subsection at infinite width (§6.3.3), we can investigate representation learning by considering the posterior distribution in the penultimate layer $\ell = L-1$ on the full set of samples $\mathcal{D}$, given observations $y_{\mathcal{A}}$. In particular, to show how the *features* of the penultimate-layer representation evolve, our goal will be to compute the change in the expectation of a penultimate-layer observable $\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)$ taken with respect to the posterior as compared to the expectation taken with respect to the prior,

$$\overline{d\mathcal{O}} \equiv \int dz_{\mathcal{D}}^{(L-1)} p\left(z_{\mathcal{D}}^{(L-1)}\middle|y_{\mathcal{A}}\right) \mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right) - \int dz_{\mathcal{D}}^{(L-1)} p\left(z_{\mathcal{D}}^{(L-1)}\right) \mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right), \tag{6.90}$$

where $p\left(z_{\mathcal{D}}^{(L-1)}\right)$ and $p\left(z_{\mathcal{D}}^{(L-1)}\middle|y_{\mathcal{A}}\right)$ are the prior and posterior distributions, respectively. This expectation difference was strictly zero in the infinite-width limit since the penultimate-layer posterior was exactly equal to the penultimate-layer prior (6.71). A nonvanishing difference, in contrast, will mean that the penultimate-layer preactivations are being updated after making observations $y_{\mathcal{A}}$. Such an *update* is a direct avatar of representation learning.

As before, by Bayes' rule we can write the posterior distribution of the penultimate preactivations $z_{\mathcal{D}}^{(L-1)}$ given our observations $y_{\mathcal{A}}$ as

$$p\left(z_{\mathcal{D}}^{(L-1)}\middle|y_{\mathcal{A}}\right) = \frac{p\left(y_{\mathcal{A}}\middle|z_{\mathcal{D}}^{(L-1)}\right) p\left(z_{\mathcal{D}}^{(L-1)}\right)}{p(y_{\mathcal{A}})}. \tag{6.91}$$

Just as before, the likelihood $p\left(y_{\mathcal{A}}\middle|z_{\mathcal{D}}^{(L-1)}\right)$ is the conditional distribution $p\left(z_{\mathcal{A}}^{(L)}\middle|z_{\mathcal{D}}^{(L-1)}\right)$ evaluated on our set of observations $z_{\mathcal{A}}^{(L)} \to y_{\mathcal{A}}$. With this expression for the posterior (6.91), we can express the update $\overline{d\mathcal{O}}$ after Bayesian learning as

$$\overline{d\mathcal{O}} = \mathbb{E}\left[\frac{p\left(y_{\mathcal{A}}\middle|z_{\mathcal{D}}^{(L-1)}\right)}{p(y_{\mathcal{A}})}\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\right] - \mathbb{E}\left[\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\right]. \tag{6.92}$$

As always, the full expectation $\mathbb{E}[\,\cdot\,]$ is to be evaluated with respect to the *prior* or initialization distribution $p\left(z_{\mathcal{D}}^{(L-1)}\right)$; all learning will always be represented explicitly with the insertion of other factors as we did above.

Let's now determine how this insertion, the likelihood-to-evidence ratio

$$\frac{p\left(y_{\mathcal{A}}\middle|z_{\mathcal{D}}^{(L-1)}\right)}{p(y_{\mathcal{A}})}, \tag{6.93}$$

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

depends on the preactivations $z_{\mathcal{D}}^{(L-1)}$. As we pointed out when working through the infinite-width example, we already worked out the form of this likelihood in (4.69) as the conditional distribution between layers. In our current context and notation, the likelihood reads

$$p\left(y_{\mathcal{A}}\middle|z_{\mathcal{D}}^{(L-1)}\right) = \frac{1}{\sqrt{\left|2\pi\widehat{G}^{(L)}\right|^{n_L}}} \exp\left(-\frac{1}{2}\sum_{i=1}^{n_L}\sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} \widehat{G}^{\tilde{\alpha}_1\tilde{\alpha}_2}_{(L)} y_{i;\tilde{\alpha}_1} y_{i;\tilde{\alpha}_2}\right), \tag{6.94}$$

where as a reminder the *stochastic metric* $\widehat{G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2} = \widehat{G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}\left(z_{\mathcal{A}}^{(L-1)}\right)$ depends explicitly on the preactivations in the penultimate layer $z_{i;\mathcal{A}}^{(L-1)}$.[33] Thus, the stochastic metric acts as a coupling here, inducing interlayer interactions between the $(L-1)$-th-layer preactivations and the observations $y_{\mathcal{A}}$. As we will see, this endows the updated distribution over $z_{\mathcal{D}}^{(L-1)}$ with a dependence on $y_{\mathcal{A}}$.

As should be fairly familiar at this point, we can decompose the stochastic metric into a mean and a fluctuation,

$$\widehat{G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2} \equiv G^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2} + \widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}, \tag{6.95}$$

in terms of which the likelihood (6.94) can be Taylor-expanded à la Schwinger–Dyson as we did before in (4.56) and (4.57). At first nontrivial order, we find for the likelihood-to-evidence ratio (6.93)

$$\begin{aligned}\frac{p\left(y_{\mathcal{A}}\middle|z_{\mathcal{D}}^{(L-1)}\right)}{p(y_{\mathcal{A}})} &= \frac{1}{p(y_{\mathcal{A}})\sqrt{\left|2\pi G^{(L)}\right|^{n_L}}} \\ &\quad\times\left[1+\frac{1}{2}\sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4\in\mathcal{A}} \widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2} G^{\tilde{\alpha}_1\tilde{\alpha}_3}_{(L)} G^{\tilde{\alpha}_2\tilde{\alpha}_4}_{(L)} \sum_{i=1}^{n_L}\left(y_{i;\tilde{\alpha}_3}y_{i;\tilde{\alpha}_4} - G^{(L)}_{\tilde{\alpha}_3\tilde{\alpha}_4}\right) + O\left(\Delta^2\right)\right].\end{aligned} \tag{6.96}$$

Here, the prefactor before the square brackets is constant with respect to the variables $z_{\mathcal{D}}^{(L-1)}$, and so all of the relevant dependence needed to evaluate update $\overline{d\mathcal{O}}$ (6.92) is contained implicitly in the metric fluctuation $\widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}$. We can thus compute a posterior expectation – i.e., the first expectation in (6.92) – of any observable by integrating against the quantity in the square brackets, so long as we also divide by an integral of

[33] Strictly speaking, we should really denote the stochastic metric here as $\widehat{\widetilde{G}}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}$ to indicate that we're focusing on the $N_{\mathcal{A}}$-by-$N_{\mathcal{A}}$ submatrix of the full stochastic metric on $\mathcal{D}$, $\widehat{G}^{(L)}_{\delta_1\delta_2}$. It's the matrix inverse of this submatrix $\widehat{\widetilde{G}}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}$ – and not the $(\tilde{\alpha}_1,\tilde{\alpha}_2)$ block of the inverse of the full matrix $\widehat{G}^{(L)}_{\delta_1\delta_2}$ – that appears in (6.94). Since this tilde-with-a-hat looks ridiculous – and since we are already heavily overburdened on the notational front – if you promise to keep this caveat in mind, we'll do everyone a favor and temporarily suppress this tilde.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

"1" against the same quantity in order to properly normalize. With this by-now familiar trick in mind, we can rewrite the posterior expectation as

$$\frac{\mathbb{E}\left\{\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\left[1+\frac{1}{2}\sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4}\widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}G^{\tilde{\alpha}_1\tilde{\alpha}_3}_{(L)}G^{\tilde{\alpha}_2\tilde{\alpha}_4}_{(L)}\sum_i\left(y_{i;\tilde{\alpha}_3}y_{i;\tilde{\alpha}_4}-G^{(L)}_{\tilde{\alpha}_3\tilde{\alpha}_4}\right)+O(\Delta^2)\right]\right\}}{\mathbb{E}\left[1+\frac{1}{2}\sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4}\widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}G^{\tilde{\alpha}_1\tilde{\alpha}_3}_{(L)}G^{\tilde{\alpha}_2\tilde{\alpha}_4}_{(L)}\sum_i\left(y_{i;\tilde{\alpha}_3}y_{i;\tilde{\alpha}_4}-G^{(L)}_{\tilde{\alpha}_3\tilde{\alpha}_4}\right)+O(\Delta^2)\right]} \tag{6.97}$$

$$\begin{aligned}&=\mathbb{E}\left[\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\right]\\&\quad+\frac{1}{2}\sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4\in\mathcal{A}}\mathbb{E}\left[\widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\right]G^{\tilde{\alpha}_1\tilde{\alpha}_3}_{(L)}G^{\tilde{\alpha}_2\tilde{\alpha}_4}_{(L)}\sum_i\left(y_{i;\tilde{\alpha}_3}y_{i;\tilde{\alpha}_4}-G^{(L)}_{\tilde{\alpha}_3\tilde{\alpha}_4}\right)+O\left(\frac{1}{n^2}\right),\end{aligned}$$

where the details of what we actually did are hidden in this here footnote.[34] We see that the first term is just the prior expectation, while the second term expresses the update $\overline{d\mathcal{O}}$ (6.90). Finally, taking only the leading finite-width corrections at order $1/n$ and restoring the tildes to correctly represent the submatrices on $\mathcal{A}$ alone, we can write down a very general expression for the update to any penultimate-layer observable at leading nontrivial order in $1/n$:

$$\overline{d\mathcal{O}}=\frac{1}{2}\sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4\in\mathcal{A}}\mathbb{E}\left[\widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\right]\widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_3}_{(L)}\widetilde{K}^{\tilde{\alpha}_2\tilde{\alpha}_4}_{(L)}\sum_i^{n_L}\left(y_{i;\tilde{\alpha}_3}y_{i;\tilde{\alpha}_4}-\widetilde{K}^{(L)}_{\tilde{\alpha}_3\tilde{\alpha}_4}\right). \tag{6.98}$$

Again, please be careful and remember that the $\mathbb{E}[\,\cdot\,]$ in (6.98) is to be evaluated with respect to the prior distribution $p\left(z_{\mathcal{D}}^{(L-1)}\right)$. Note also that the lone expectation in the update (6.98) is just the covariance of the stochastic metric with the observable:

$$\mathbb{E}\left[\widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\right]=\mathbb{E}\left[\widehat{G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\right]-\mathbb{E}\left[\widehat{G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}\right]\mathbb{E}\left[\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\right]. \tag{6.99}$$

As we addressed in that rugly footnote, for a general order-one observable, this covariance is $1/n$-suppressed but nonzero. Thus, we see that at large-but-finite width ($1\ll n<\infty$), such observables get updated: representations are learned.

[34] The reason that we treated the additional $O\left(\Delta^2\right)$ pieces as $O\left(1/n^2\right)$ is hidden under the rug in the main body. To peak under that rug, first let us schematically express the likelihood-to-evidence ratio (6.96) as constant $\times\left[1+\sharp_1\Delta G+\sharp_2(\Delta G)^2+O\left(\Delta^3\right)\right]$. Then, the posterior expectation becomes

$$\begin{aligned}\frac{\mathbb{E}\left\{\mathcal{O}\left[1+\sharp_1\Delta G+\sharp_2(\Delta G)^2+O\left(\Delta^3\right)\right]\right\}}{\mathbb{E}\left[1+\sharp_1\Delta G+\sharp_2(\Delta G)^2+O(\Delta^3)\right]}&=\frac{\mathbb{E}[\mathcal{O}]+\sharp_1\mathbb{E}[\mathcal{O}\Delta G]+\sharp_2\mathbb{E}\left[(\Delta G)^2\,\mathcal{O}\right]+O\left(1/n^2\right)}{1+\sharp_2\mathbb{E}\left[(\Delta G)^2\right]+O(1/n^2)}\\&=\mathbb{E}[\mathcal{O}]+\sharp_1\mathbb{E}[\mathcal{O}\Delta G]+\sharp_2\left\{\mathbb{E}\left[(\Delta G)^2\,\mathcal{O}\right]-\mathbb{E}\left[(\Delta G)^2\right]\mathbb{E}[\mathcal{O}]\right\}+O\left(1/n^2\right).\end{aligned} \tag{6.100}$$

Decomposing the observable into a mean and a fluctuation as $\mathcal{O}=\mathbb{E}[\mathcal{O}]+\Delta\mathcal{O}$, we see that the term proportional to the coefficient $\sharp_2$ is $\mathbb{E}\left[O\left(\Delta^3\right)\right]=O\left(1/n^2\right)$ and thus can be neglected, while the leading finite-width correction cannot be neglected: $\sharp_1\mathbb{E}[\mathcal{O}\Delta G]=\sharp_1\mathbb{E}[\Delta\mathcal{O}\Delta G]=O(1/n)$.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

In order to see how this works, let's consider a concrete example. The simplest observable turns out to be the average norm of the activations,

$$\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right) \equiv \frac{1}{n_{L-1}} \sum_{j=1}^{n_{L-1}} \sigma_{j;\delta_1}^{(L-1)} \sigma_{j;\delta_2}^{(L-1)}, \tag{6.101}$$

which we can decompose in terms of a mean and a fluctuation as

$$\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right) = \mathbb{E}\left[\mathcal{O}\left(z_{\mathcal{D}}^{(L-1)}\right)\right] + \frac{1}{C_W^{(L)}} \widehat{\Delta G}_{\delta_1\delta_2}^{(L)}, \tag{6.102}$$

if we also recall the explicit form of the metric fluctuation (4.74)

$$\widehat{\Delta G}_{\tilde{\alpha}_1\tilde{\alpha}_2}^{(L)} = C_W^{(L)} \frac{1}{n_{L-1}} \sum_{j=1}^{n_{L-1}} \left(\sigma_{j;\tilde{\alpha}_1}^{(L-1)} \sigma_{j;\tilde{\alpha}_2}^{(L-1)} - \mathbb{E}\left[\sigma_{j;\tilde{\alpha}_1}^{(L-1)} \sigma_{j;\tilde{\alpha}_2}^{(L-1)}\right]\right). \tag{6.103}$$

Then, plugging into our expression for the leading-order finite-width update (6.98), we find

$$\begin{aligned} \overline{d\mathcal{O}} &= \frac{1}{2C_W^{(L)}} \sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4 \in \mathcal{A}} \mathbb{E}\left[\widehat{\Delta G}_{\delta_1\delta_2}^{(L)} \widehat{\Delta G}_{\tilde{\alpha}_1\tilde{\alpha}_2}^{(L)}\right] \widetilde{K}_{(L)}^{\tilde{\alpha}_1\tilde{\alpha}_3} \widetilde{K}_{(L)}^{\tilde{\alpha}_2\tilde{\alpha}_4} \sum_i \left(y_{i;\tilde{\alpha}_3} y_{i;\tilde{\alpha}_4} - \widetilde{K}_{\tilde{\alpha}_3\tilde{\alpha}_4}^{(L)}\right) \\ &= \frac{1}{2n_{L-1}C_W^{(L)}} \sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4 \in \mathcal{A}} V_{(\delta_1\delta_2)(\tilde{\alpha}_1\tilde{\alpha}_2)}^{(L)} \widetilde{K}_{(L)}^{\tilde{\alpha}_1\tilde{\alpha}_3} \widetilde{K}_{(L)}^{\tilde{\alpha}_2\tilde{\alpha}_4} \sum_i \left(y_{i;\tilde{\alpha}_3} y_{i;\tilde{\alpha}_4} - \widetilde{K}_{\tilde{\alpha}_3\tilde{\alpha}_4}^{(L)}\right), \end{aligned} \tag{6.104}$$

where to go to the second line we used the definition of the four-point vertex in terms of the two-point function of the metric fluctuation (4.76). As this vertex characterizes the non-Gaussianity of the output distribution, we see explicitly here how interactions are mediating updates to the penultimate-layer activations. In addition, the leading factor of $1/n_{L-1}$ makes it clear that this update is a finite-width effect. Further, the term in the last parenthesis shows that the update depends explicitly on the difference between our observations of the outputs, $y_{i;\tilde{\alpha}_3} y_{i;\tilde{\alpha}_4}$, and our prior expectations of them, $\mathbb{E}\left[z_{i;\tilde{\alpha}_3}^{(L)} z_{i;\tilde{\alpha}_4}^{(L)}\right] \equiv \widetilde{K}_{\tilde{\alpha}_3\tilde{\alpha}_4}^{(L)} + O(1/n)$. This means that the observations are in fact propagating *backward* to induce changes in the hidden-layer representations.[35]

[35] This kind of backward-propagation, or *backpropagation*, if you will, persists further into the shallower hidden layers as well. However, in the $(L-2)$-th layer, the posterior update turns out to be of order $O\left(1/n^2\right)$. Intuitively this makes sense because the change in the representation in the penultimate layer $(L-1)$ is already down by a factor of $1/n$, and it gets further suppressed due to the $1/n$-suppression of the interlayer interaction in going back to the $(L-2)$-th layer.

Mathematically, we can consider the update to an $(L-2)$-th-layer observable $\mathcal{O}\left(z_{\mathcal{D}}^{(L-2)}\right)$ as

$$\overline{d\mathcal{O}} \equiv \int dz_{\mathcal{D}}^{(L-2)} p\left(z_{\mathcal{D}}^{(L-2)} \middle| y_{\mathcal{A}}\right) \mathcal{O}\left(z_{\mathcal{D}}^{(L-2)}\right) - \int dz_{\mathcal{D}}^{(L-2)} p\left(z_{\mathcal{D}}^{(L-2)}\right) \mathcal{O}\left(z_{\mathcal{D}}^{(L-2)}\right). \tag{6.105}$$

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press

Although perhaps not practically useful, this Bayesian analysis of representation learning at finite width will serve as a theoretically useful blueprint for studying a similar type of representation learning that occurs with gradient-based learning at finite width in §11. Now, with all these allusions to gradient-based learning having accrued with interest, you must be really excited to flip the page to the next chapter!

Through the chain of Bayes', sum, and product rules, the posterior insertion in this formula is given in terms of the following marginalization:

$$p\Big(z_{\mathcal{D}}^{(L-2)}\Big|y_{\mathcal{A}}\Big)=\frac{p\Big(y_{\mathcal{A}}\Big|z_{\mathcal{D}}^{(L-2)}\Big)\,p\Big(z_{\mathcal{D}}^{(L-2)}\Big)}{p(y_{\mathcal{A}})}=\int dz_{\mathcal{D}}^{(L-1)}\frac{p\Big(y_{\mathcal{A}}\Big|z_{\mathcal{D}}^{(L-1)}\Big)}{p(y_{\mathcal{A}})}p\Big(z_{\mathcal{D}}^{(L-1)},z_{\mathcal{D}}^{(L-2)}\Big)\,. \tag{6.106}$$

From here, through the same set of manipulations that led to the update equation for the penultimate layer (6.98), we get

$$\overline{d\mathcal{O}}=\frac{1}{2}\sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4\in\mathcal{A}}\mathbb{E}\left[\widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}\Big(z_{\mathcal{D}}^{(L-1)}\Big)\,\mathcal{O}\Big(z_{\mathcal{D}}^{(L-2)}\Big)\right]\widetilde{K}^{\tilde{\alpha}_1\tilde{\alpha}_3}_{(L)}\widetilde{K}^{\tilde{\alpha}_2\tilde{\alpha}_4}_{(L)}\sum_{i}^{n_L}\Big(y_{i;\tilde{\alpha}_3}y_{i;\tilde{\alpha}_4}-\widetilde{K}^{(L)}_{\tilde{\alpha}_3\tilde{\alpha}_4}\Big)+O\Big(\frac{1}{n^2}\Big)\,. \tag{6.107}$$

Thus, to show that this change is of order $O\big(1/n^2\big)$, we need to show that the *interlayer correlation*,

$$\mathbb{E}\left[\widehat{\Delta G}^{(L)}_{\tilde{\alpha}_1\tilde{\alpha}_2}\Big(z_{\mathcal{D}}^{(L-1)}\Big)\,\mathcal{O}\Big(z_{\mathcal{D}}^{(L-2)}\Big)\right], \tag{6.108}$$

is of order $O\big(1/n^2\big)$. This can be most swiftly carried out in the future, first by the application of the formula (8.54) with $\ell=L-2$ and then with the associated trickery (8.70). If you are up for a challenge, please flip forward and write a note next to (8.70) reminding yourself to come back to footnote 35 in §6.4.3. *Spoiler alert:* you should in fact find that (6.108) is of order $O\big(1/n^2\big)$.

https://doi.org/10.1017/9781009023405.008 Published online by Cambridge University Press