CMPUT 654: Mathematical Foundations of Modern AI Systems Fall 2026

Lecture 4: Sequence modeling, generation, and memory

September 10, 2026

Lecturer: Csaba Szepesvári *Note prepared by: Csaba Szepesvári*

Grayson Lee prepared a scribe of the September 10 lecture. Csaba Szepesvári prepared and substantially expanded the present notes.

About these lecture notes. We know how to fit a sequence model by next-token prediction. We now ask whether fitting successful examples makes the model generate successful sequences. The answer depends on how prediction errors affect later generation, what cost we care about, and what structure the model preserves. This leads from log loss to error recovery, reverse KL, and models with explicit memory.

Scope relative to class. Further questions and ideas arose while preparing these notes, so they extend and refine the discussion from the lecture. The lecture developed the positive-data fitting question, the binary error-propagation example, stability of Markov models, and the concentration-dependent reverse-KL bound. The state-space introduction and system-identification material are now collected in Lecture 5, which develops these topics in detail. The notes group the cost bounds first, then use exact fixed-KL cost intervals to develop two laws attaining the cost bounds. We reuse these laws when interpreting normalized loss, comparing fitting objectives, and introducing recovery. Repeated errors, parity, and the explicit modulus calculations are developed in exercises. A further section asks how parameter sharing affects learning as sequence length grows. The fitting and structural choices follow. Headings ending in a superscript asterisk* mark supplementary material: sharper comparisons between likelihood and cost and a finite-sample fitting guarantee. The exercises retain the calculations and proofs left to students. Longer proofs, a statistical obstruction to reverse-KL fitting, reverse-KL optimization details, and recovery for history-dependent generators are in the appendices.

Notation and reading guide. Tokens are $X_t \in [N]$. We use S_t for a state when discussing persistent and revisable memory. A vector $x \in \mathbb{R}^d$ is a column vector. For indices $a \le b$, write $X_{a:b} = (X_a, \dots, X_b)$ and $X_{<t} = X_{1:t-1}$; lowercase $x_{a:b}$ denotes a realized sequence. For a symbol u, $u^k = (u, \dots, u)$ is the length-k constant sequence, so $0^T = (0, \dots, 0)$, and juxtaposition denotes concatenation, as in $10^{T-1} = (1, 0, \dots, 0)$. Sequence notation is useful because the same object has two roles: it is an ordered tuple in $[N]^T$, and it is a time-indexed function on $[T]$. We use tuple notation when order and prefixes matter, and ordinary vector notation when applying linear algebra. For a sequence cost $c : [N]^T \to \mathbb{R}$ and a distribution P over $[N]^T$, $\mathbb{E}_P[c]$ abbreviates $\mathbb{E}[c(X_{1:T})]$ for $X_{1:T} \sim P$. Thus

$$\mathbb{E}_P[c] = \sum_{x_{1:T}} P(x_{1:T}) c(x_{1:T}).$$

By default, in what follows, P is the data law and Q, or Q_θ, is a model law. We use κ for a KL value or budget, s for a cost-discrepancy value or budget, and η for per-token accuracy, including the optimization tolerance introduced in Section 4.4.3. Logarithms are natural. Read through Section 4.6; the supplementary material can be read separately. Numbered endnotes, bibliographical notes, the glossary, exercises, appendices, and references follow the take-home messages in that order.

4.1 From next-token prediction to useful generation

Lecture 1 described transformers as maps from token prefixes to next-token distributions. Lecture 2 asked what structure allows learning to generalize. Lecture 3 explained why log loss is a natural objective for probability estimation. Pretraining brings these ideas together: shared parameters are fitted to predict tokens across many sequences and contexts. The prediction problem supplies a scalable way to learn from data.

Throughout the discrete discussion, sequences have fixed length T over a finite vocabulary $[N]$, and costs are real-valued. The finite space $[N]^T$ simplifies the arguments: every cost is bounded, and expectations are finite sums. Many results hold more generally; Appendix 4.A explains where finiteness is used and what can replace it. It is also easy to extend the results to allow variable length sequences as outlines in Lecture 1. For simplicity, we restrict our attention to the fixed length case.

Let $X_{1:T}^{(1)}, \ldots, X_{1:T}^{(n)}$ be independent training sequences from P, a distribution over $[N]^T$. Their tokens may have arbitrary dependence within each sequence. Empirical log-loss minimization chooses

$$\widehat{\theta} \in \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} -\log Q_\theta(X_{1:T}^{(i)}). \tag{4.1}$$

The chain rule of probabilities gives that for every $x_{1:T} \in [N]^T$,

$$Q_\theta(x_{1:T}) = \prod_{t=1}^{T} Q_\theta(x_t \mid x_{<t}), \qquad x_{<t} = (x_1, \ldots, x_{t-1}). \tag{4.2}$$

Every sequence law has this factorization. We call it the *autoregressive factorization*; restrictions on the conditional predictors, such as a transformer architecture or a finite context window, specify the model class. Taking logarithms in Eq. (4.2) and plugging in $X_{1:T}$ in place of $x_{1:T}$ gives

$$-\log Q_\theta(X_{1:T}) = \sum_{t=1}^{T} -\log Q_\theta(X_t \mid X_{<t}). \tag{4.3}$$

Consequently, next-token log-loss minimization, the workhorse of language model training, is sequence maximum likelihood.

When evaluating Eq. (4.3), the predictor receives the observed prefix, including the observed next input. Evaluating every factor on the observed prefix is *teacher forcing*: the student's cost is measured at the teacher's data. The metaphore comes from viewing the problem as a teacher generating data that a student needs to use to learn. During generation, the next input is instead sampled from the model:

$$\widetilde{X}_t \sim Q_\theta(\cdot \mid \widetilde{X}_{<t}). \tag{4.4}$$

The resulting sequence has joint law Q_θ. The same factorization supports both likelihood evaluation and sampling.

As explained earlier, the parameter $\widehat{\theta}$ from Eq. (4.1) is maximum likelihood parameter estimate: minimizing the negative log of a product maximizes that product, which, under independence, is the probability assigned to the data. For a fresh sequence $X_{1:T}$ with law P, the log-loss identity from Lecture 3 and the decomposition in Eq. (4.3) give

$$\begin{aligned} \mathbb{E}[-\log Q_\theta(X_{1:T})] &= H(P) + D(P, Q_\theta), \\ D(P, Q_\theta) &= \sum_{t=1}^{T} \mathbb{E}\big[D\big(P(\cdot \mid X_{<t}), Q_\theta(\cdot \mid X_{<t})\big)\big]. \end{aligned} \tag{4.5}$$

The entropy term is independent of θ. Thus minimizing population log loss minimizes KL; empirical log loss estimates this population objective from data. The second identity is the *KL chain rule*: sequence KL is the sum of the expected conditional KLs. Exercise 1 asks for its derivation.

In ML/statistics we distinguish between data and model laws. This leads to the following definition:

Definition 4.1.1 (Forward and reverse KL)**.** When P is a data law and Q is a model law, we call $D(P, Q)$ the *forward KL*, while $D(Q, P)$ is called the *reverse KL*.

With our finite spaces, we have

$$D(P, Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)} \quad \text{(the } \textit{forward KL}\text{)},$$
$$D(Q, P) = \sum_x Q(x) \log \frac{Q(x)}{P(x)} \quad \text{(the } \textit{reverse KL}\text{)}.$$

Note that above a zero numerator contributes zero; a positive numerator with zero denominator contributes $+\infty$.[1] The names refer to the roles of P, Q: forward KL averages under the data law, whereas reverse KL averages under the generated law.

As explained, $\widehat{\theta}$, is expected to approximately minimize $D(P, Q_\theta)$ in θ. The discrepancy $D(P, Q_{\widehat{\theta}}) - \inf_\theta D(P, Q_\theta)$ describes the effect of using a finite dataset in the lack of access to P (and a reliable way of minimizing $D(P, Q_\theta)$). Recall from Lecture 2 that this is expected to be controlled when n is large and things are "sufficiently nice". As such, we leave this question alone in this lecture and instead ask, would it be even beneficial (and to what degree) if we could find a model Q with a small population KL?

4.1.1 A task specifies which generations are useful

A prompt asks the model to continue in a particular way. Mathematically, *prompting* uses a conditional distribution $Q_\theta(\cdot \mid u)$, where u is the supplied prompt (we omit the full formal description of the setup, but here u is also a sequence in the typical language model training setting). For a mathematics task, a useful continuation might be a correct solution; for a knowledge question, an accurate answer. A *cost* $c(u, x_{1:T})$ assigns a numerical value to the generated response. Given a test-prompt distribution ν, the expected cost of a model Q_θ is

$$\mathcal{J}(\theta) = \mathbb{E}[c(U, \widetilde{X}_{1:T})], \qquad U \sim \nu, \quad \widetilde{X}_{1:T} \mid U \sim Q_\theta(\cdot \mid U). \tag{4.6}$$

A model's quality then is determined by $\mathcal{J}(\theta)$. The cost function is thus $c : [N]^T \to \mathbb{R}$. In what follows, without loss of generality, we assume that it is nonnegative valued. Sometimes (especially in RL) the task is formulated with rewards. These formulations are equivalent: Maximizing expected reward is equivalent to minimizing its negative.

A common approach to get a small-cost model is to collect low-cost (or even zero-cost) prompt–response pairs, and then find a model that minimizes the empirical log-loss, aiming to "mimick" the responses in the training data. Formally, solve

$$\widehat{\theta} \in \arg\min_\theta \frac{1}{n} \sum_{i=1}^{n} \sum_{t=1}^{T} -\log Q_\theta(X_t^{(i)} \mid U^{(i)}, X_{<t}^{(i)}). \tag{4.7}$$

In training large language models, often incremental methods are used, such as some form of gradient descent. Lectures 9–11 return to how gradient-based optimization selects among fitted solutions

and when local gradients lead to global convergence. When this is the case, typically people start gradient descent with a parameter θ_0 which is obtained during *pre-training* (as outlined in Lecture 1). Then, with the data as above, people continue gradient descent from θ_0 with the data as above. This practice is called *fine-tuning* by likelihood on *positive examples* (starting from a pretrained model parameters). The data is called "positive" because it will contain completed proofs (in math problems) or, more generally, answered deemed correct, with no examples in the data of incorrect responses.

For simplicity, in this lecture we do not consider the "continuing" aspect of the fine-tuning and will ask the equation whether minimizing log-loss with positive data will produce low-cost answers to new prompts. To isolate the connection between fitting and task performance, we set the prompts aside; keeping the prompts (conditioning) around would introduce extra notation, but otherwise, in general, changes nothing. There is one exception to this which will be noted when it arises.

With this, let is turn to formulating the specific question studied next. Let P be the law of positive sequences, Q the fitted law, and c the cost. One should generally think that $\mathbb{E}_P[c] = 0$; the results will aim at this case but will be more general. Recall from Eq. (4.5) that the population log loss is

$$\mathbb{E}_P[-\log Q] = H(P) + D(P, Q).$$

As explained earlier, one should think about log-loss minimization, or likelihood fitting as seeking a model Q that with small $D(P, Q)$. Yet, the task calls for small generated cost $\mathbb{E}_Q[c]$. Suppose fitting gives us $D(P, Q) \le \kappa$. The basic question we ask is how large can $\mathbb{E}_Q[c]$ be? In other words, is controlling KL this way sufficient to control generation cost? To what degree? What are the strengths and weaknesses of this approach? Naturally, we will continue with asking whether there is anything better.

4.2 How does KL control the task cost?

Define the *cost discrepancy*

$$d_c(P, Q) = |\mathbb{E}_P[c] - \mathbb{E}_Q[c]|$$

to measure the change in expected cost from the data law to the model law. For cost-free data laws, thank to the assumption that $c \ge 0$, $d_c(P, Q) = \mathbb{E}_Q[c]$, so a bound on this discrepancy answers our question.

We will compare four ways to turn KL control into cost control, starting with simpler and moving to more elaborate results. We will compare the resulting bounds, and ask which of them are sharp and when. A supplementary subsection asks for both the smallest and largest cost discrepancies at a fixed KL, relating these to moduli of continuity. We then compute these intervals in binary examples to see how differently models with the same likelihood can behave.

4.2.1 Use the cost range

Recall the definition of the total variation distance between probability laws: In what follows we fix a finite set $\mathcal{X}$ as the domain of the cost and the probability distributions P and Q.

Definition 4.2.1 (Total variation)**.** For laws P, Q over a finite set $\mathcal{X}$, their *total-variation distance* is

$$\|P - Q\|_{\mathrm{TV}} = \sup_{A \subseteq \mathcal{X}} |P(A) - Q(A)| = \frac{1}{2} \sum_{x \in \mathcal{X}} |P(x) - Q(x)|. \tag{4.8}$$

Lecture 3 explains why the same formula is a norm on signed measures and a metric on probability laws, and contrasts both with a pseudometric.

Proposition 4.2.2 (Total variation controls bounded costs)**.** *If* $0 \le c(x) \le C_{\max}$ *for every* $x \in \mathcal{X}$*, then*

$$|\mathbb{E}_Q[c] - \mathbb{E}_P[c]| \le C_{\max}\|P - Q\|_{\mathrm{TV}}. \tag{4.9}$$

This is Lecture 3, Exercise 4(c), after scaling the cost by $C_{\max}$. The exercise proves it from the positive and negative parts of the signed measure $Q - P$.

This bound is exact when maximizing over costs with the specified range: $c(x) = C_{\max}\mathbb{I}\{Q(x) > P(x)\}$ attains equality. Thus TV measures the largest expected-cost discrepancy among such costs.

Pinsker's inequality from Lecture 3 now gives

$$\mathbb{E}_Q[c] \le \mathbb{E}_P[c] + C_{\max}\sqrt{\frac{D(P,Q)}{2}}. \tag{4.10}$$

For any bounded cost, sufficiently small KL therefore gives sufficiently small cost discrepancy. If $\mathbb{E}_P[c] = 0$, a sufficient condition for $\mathbb{E}_Q[c] \le s$ is

$$D(P,Q) \le 2(s/C_{\max})^2.$$

Both the square-root dependence and its constant are necessary for general bounded costs.

Example 4.2.3 (A locally sharp fair-coin comparison)**.** Let $\mathcal{X} = \{0,1\}$,

$$P = \text{Bernoulli}(1/2), \qquad Q = \text{Bernoulli}(1/2+\delta), \qquad c(x) = x.$$

As $\delta \to 0$,

$$d_c(P,Q) = |\delta|, \qquad D(P,Q) = 2\delta^2 + O(\delta^4).$$

Pinsker's bound divided by the actual discrepancy tends to one: it is *locally sharp*.

For a count of token errors, $C_{\max} = T$, so the horizon also enters the conversion.

Recall that we are interested in the case when $\mathbb{E}_P c = 0$ or small, while $c \ge 0$. Can we improve this bound using this information?

4.2.2 Successful data give a sharper bound*

Let us start with the case when $\mathbb{E}_P c = 0$.

Then, the following result applies:

Proposition 4.2.4 (Forward KL and zero-cost data)**.** *If* $0 \le c \le C_{\max}$*,* $\mathbb{E}_P[c] = 0$ *then*

$$\mathbb{E}_Q[c] \le C_{\max}(1 - e^{-D(P,Q)}) \le C_{\max}D(P,Q). \tag{4.11}$$

This is an improvement over Eq. (4.10) when $D(P,Q) \le 1/2$ and the gap between them grows as the KL between P and Q approaches zero. For small KL, exploiting successful data replaces the square-root dependence from Pinsker by linear dependence.

Recording only whether an event occurred can only decrease KL. For an event certain under P, this gives a lower bound on its probability under Q. Recall that for $p, q \in [0,1]$, $\mathrm{kl}(p,q)$ denotes the KL between the Bernoulli distributions with parameters p and q.

Proposition 4.2.5 (From distribution KL to event probability KL)**.** *Let P, Q be distributions over the finite domain $\mathcal{X}$. Then, for any $A \subset \mathcal{X}$,*

$$\mathrm{kl}(P(A), Q(A)) \le D(P, Q)\,.$$

In particular, if $P(A) = 1$ then

$$Q(A) \ge e^{-D(P,Q)}\,.$$

Proof. Consider the transformation that maps any $x \in \mathcal{X}$ to 1 if $x \in A$ and to 0 otherwise (this is just the indicator if A). Apply the data processing inequality with this deterministic transformation to get the first inequality. Indeed, the push-forward of P (Q) under $\mathbb{I}_A$ is the Bernoulli distribution with parameter $P(A)$ (respectively, $Q(A)$). To get the second inequality, note that if $P(A) = 1$, $\mathrm{kl}(P(A), Q(A)) = \log \frac{P(A)}{Q(A)}$. Rearrangement then gives the second inequality. The general data-processing statement and its log-sum proof appear in Lecture 3, Proposition 3.4.2 and Exercise 4. □

With this we can return to the proof of Proposition 4.2.4.

Proof of Proposition 4.2.4. Let $A = \{x : c(x) = 0\}$. Note that since $c \ge 0$ and $\mathbb{E}_P[c] = 0$, $P(c = 0) = P(A) = 1$. Now, by algebra, using the result of the previous proposition and since $1 - e^{-u} \le u$ for $u \ge 0$,

$$\mathbb{E}_Q\, c \le C_{\max} Q(A^c) = C_{\max}(1 - Q(A)) \le C_{\max}(1 - e^{-D(P,Q)}) \le C_{\max} D(P, Q)\,. \qquad \square$$

The bound is sharp: Let c take on values of 0 and $C_{\max}$, but otherwise be arbitrary. Let P be as in the proposition. Let R be any probability distribution supported on $\{c > 0\} = \{c = C_{\max}\}$. For any $0 < a < 1$, choosing $Q = (1 - a)P + aR$, we have $D(P, Q) = -\log(1 - a)$ and $\mathbb{E}_Q[c] = aC_{\max}$. Hence, indeed

$$\mathbb{E}_Q[c] = aC_{\max} = C_{\max}(1 - e^{-D(P,Q)})\,.$$

Thus, equality holds in the first bound of Eq. (4.11).

4.2.3 Concentration and the direction of KL

Our next result strengthens the bound we obtained using Pinsker inequality when the costs under P concentrate around their mean, but the price is switching from forward to reverse KL:

Proposition 4.2.6 (Cost control from exponential moments)**.** *Let P, Q be distributions over a finite set $\mathcal{X}$ and let $c : \mathcal{X} \to \mathbb{R}$. Then.*

$$\mathbb{E}_Q[c] - \mathbb{E}_P[c] \le \inf_{\lambda > 0} \frac{D(Q, P) + \log \mathbb{E}_P[e^{\lambda(c - \mathbb{E}_P[c])}]}{\lambda}. \tag{4.12}$$

This reduces cost control to bounding an exponential moment and optimizing one positive scalar. The exponential-tilting proof is in Appendix 4.B. A quadratic bound on that logarithmic moment function gives a particularly simple answer.

Definition 4.2.7 (Subgaussian cost)**.** The cost c is v^2*-subgaussian under* P, for $v \ge 0$, if

$$\log \mathbb{E}_P\left[e^{\lambda(c - \mathbb{E}_P[c])}\right] \le \frac{\lambda^2 v^2}{2} \quad \text{for every } \lambda \in \mathbb{R}. \tag{4.13}$$

Here v measures a scale of fluctuations; v^2 is the variance proxy. We write $v_P(c)$ for the *optimal variance proxy*, which is also known as the subgaussian constant of c under P:

$$v_P(c) = \min\left\{a \geq 0 : \log \mathbb{E}_P[e^{\lambda(c-\mathbb{E}_P[c])}] \leq \frac{\lambda^2 a}{2} \text{ for every } \lambda \in \mathbb{R}\right\}.$$

The minimum is attained; a constant cost has $v_P(c) = 0$.[2]

Theorem 4.2.8 (Reverse KL and cost concentration)**.** *Suppose c satisfies Eq. (4.13). Then, interpreting $0 \cdot \infty = \infty$, we have*

$$|\mathbb{E}_Q[c] - \mathbb{E}_P[c]| \leq v\sqrt{2D(Q,P)}. \tag{4.14}$$

Proof. Substituting Eq. (4.13) into Eq. (4.12) gives

$$\inf_{\lambda>0}\left(\frac{D(Q,P)}{\lambda} + \frac{\lambda v^2}{2}\right) = v\sqrt{2D(Q,P)}.$$

Applying the same argument to $-c$ bounds the other direction. □

Interchanging P, Q and using the optimal proxies $v_P(c), v_Q(c)$ gives

$$d_c(P,Q) \leq \sqrt{2\min\{v_Q(c)D(P,Q),\, v_P(c)D(Q,P)\}}. \tag{4.15}$$

While the forward KL term refines the bound coming from Pinsker's inequality (i.e., Eq. (4.10)), when besides that P is supported on zero-cost sequences no cost information is available while fitting Q, the usefulness of this refinement is limited: $v_Q(c)$ may be as large as $C_{\max}^2/4$ and remains uncontrolled.

The reverse KL part of the bound is more promising in that $v_P(c)$ is something we expect to be small. In particular, when $P(c = 0) = 1$, as noted beforehand, $v_P(c) = 0$. However, this immediately reveals a limitation: The bound tells us that for such P and c, the costs under P and Q either match perfectly (when $D(Q,P) < \infty$) or if the costs are different then $D(Q,P) = \infty$. This foreshadows the difficulty of controlling reverse KL, which we will discuss later.

4.2.4 Use the variability of the data's cost*

Can we use cost variation under P while keeping forward KL, the quantity likelihood fitting controls? As it turns out, the answer is yes. The proof of this is slightly more complicated than the previous ones. The proof centers the cost under P, expands $Q - P$ in powers of $\sqrt{Q} - \sqrt{P}$, and uses Cauchy–Schwarz to obtain a variance term instead of an exponential-moment bound. Write $\sigma_P(c)^2 = \mathbb{E}_P[(c - \mathbb{E}_P[c])^2]$ for the variance of the cost.

Proposition 4.2.9 (Cost variance and forward KL)**.** *If $0 \leq c \leq C_{\max}$ and $D(P,Q) < \infty$, then*

$$d_c(P,Q) \leq 2\sigma_P(c)\sqrt{D(P,Q)} + C_{\max}D(P,Q). \tag{4.16}$$

Appendix 4.C.1 gives the proof.

The combination of a variance-dependent square-root term and a range-dependent linear term is called *Bernstein form*. The square-root term uses the data's cost variation rather than its full range. When $\sigma_P(c) = 0$, this term vanishes, leaving a bound linear in KL. Note that this bound extends the zero-cost data bound Eq. (4.11). Indeed, when $c = 0$ P-almost surely, $\sigma_P(c) = 0$ and we

recover the linear bound in Eq. (4.11). The additive term accounts for changes in cost on outcomes to which Q assigns probability, including outcomes absent under P.

The variance-sensitive and subgaussian bounds are useful in different regimes. In Example 4.2.3, this bound is asymptotic to $\sqrt{2}|\delta|$, while Pinsker and the subgaussian bound are asymptotic to the actual discrepancy $|\delta|$. For rare errors, the variance-sensitive bound can instead give the correct order while the subgaussian bound is much larger; Appendix 4.C.2 works out both comparisons.

4.2.5 The play of cost when forward KL is controlled

A sharp upper bound tells us how large the cost discrepancy can be. To understand how much a KL value tells us about task performance, we need both the smallest and the largest cost discrepancies compatible with that value. The binary examples below will give exact intervals: at the same KL, some generators have much smaller cost than others. Calculating these fixed-KL intervals reveals the ambiguity that remains even after the upper bounds have been made sharp. If the interval is large, we can conclude that KL on its own does allow a "large play". *Knowing how large the play is exactly what allows us to think about whether it is worthwhile to go beyond controlling KL.* The rest of this section develops these ideas and connects them to standard concepts in mathematics.

Fix the data law P and cost c, and let Q range over all probability laws on $\mathcal{X}$. Define the two endpoints

$$\begin{aligned} f_P(\kappa) &= \min_{Q:\, D(P,Q)=\kappa} d_c(P,Q), \\ g_P(\kappa) &= \max_{Q:\, D(P,Q)=\kappa} d_c(P,Q). \end{aligned} \tag{4.17}$$

Here KL levels range over the attainable values in $[0, +\infty]$. The extrema exist in our finite setting; Appendix 4.A gives the compactness argument. These are the sharp bounds in

$$f_P\big(D(P,Q)\big) \le d_c(P,Q) \le g_P\big(D(P,Q)\big).$$

The preceding inequalities studied the upper endpoint. The lower endpoint answers an equally relevant question: *how small could the cost discrepancy be at this same KL?* A large gap between the endpoints means that likelihood alone leaves substantial uncertainty about task performance. Conversely, a small gap means no such wiggle room, which would mean that controlling KL alone could be sufficient to get a good control on the task cost.

From fixed values to error budgets. Usually we know an upper bound on a discrepancy, rather than its exact value. The corresponding sharp conversion functions are called *moduli of continuity*. At the fixed reference P, define

$$\omega_{D\to c}(\kappa; P) = \sup_{Q:\, D(P,Q)\le\kappa} d_c(P,Q). \tag{4.18}$$

This is the modulus of continuity of the cost discrepancy with respect to KL. It asks: *how large can the cost discrepancy be when KL is at most κ?* In the other direction, define

$$\omega_{c\to D}(s; P) = \sup_{Q:\, d_c(P,Q)\le s} D(P,Q). \tag{4.19}$$

This is the modulus of continuity of KL with respect to cost discrepancy. It asks: *how large can KL be when the cost discrepancy is at most s?* The arrows specify which discrepancy controls the other; both formulas use forward KL, $D(P,Q)$.

Proposition 4.2.10 (The sharp conversions from fixed-KL endpoints)**.** *For the fixed P, c,*

$$\begin{aligned}\omega_{D\to c}(\kappa; P) &= \sup_{\kappa' \le \kappa} g_P(\kappa'),\\ \omega_{c\to D}(s; P) &= \sup\{\kappa : f_P(\kappa) \le s\}.\end{aligned} \tag{4.20}$$

Both moduli are nondecreasing. They are the smallest nondecreasing functions giving, respectively, the bounds

$$d_c(P,Q) \le \omega_{D\to c}\big(D(P,Q); P\big), \qquad D(P,Q) \le \omega_{c\to D}\big(d_c(P,Q); P\big).$$

Proof. At KL level κ, the largest cost discrepancy is $g_P(\kappa)$, and some cost discrepancy is at most s exactly when $f_P(\kappa) \le s$. Taking the indicated suprema proves Eq. (4.20). Increasing either budget enlarges its feasible set, proving monotonicity. If a nondecreasing function g satisfies $d_c(P,Q) \le g(D(P,Q))$, then every Q with KL at most κ has discrepancy at most $g(\kappa)$. Taking the supremum proves $\omega_{D\to c}(\kappa; P) \le g(\kappa)$. The other direction is identical. □

Thus the upper endpoints determine the $D \to c$ modulus, while the lower endpoints determine the $c \to D$ modulus. In particular, if g_P is nondecreasing, it already equals $\omega_{D\to c}$. An increasing lower bound can be read in the opposite direction by inverting it.

Proposition 4.2.11 (Inverting a lower bound)**.** *Suppose $f : [0,+\infty] \to [0,+\infty]$ is strictly increasing and $f(D(P,Q)) \le d_c(P,Q)$ for every Q. Then*

$$\omega_{c\to D}(f(\kappa); P) \le \kappa.$$

If some Q has $D(P,Q) = \kappa$ and $d_c(P,Q) = f(\kappa)$, equality holds. In particular, when the sharp lower endpoint f_P is strictly increasing, $\omega_{c\to D}(s; P) = f_P^{-1}(s)$ for s in its range.

Proof. If $d_c(P,Q) \le f(\kappa)$, then $f(D(P,Q)) \le f(\kappa)$. Strict increase gives $D(P,Q) \le \kappa$. A law attaining the lower bound at κ also attains this KL budget, proving equality. □

This explains why we care about the converse modulus: it describes the lower-bound side of the same comparison. A small lower endpoint and a large upper endpoint allow models of very different task quality to have exactly the same KL. The examples below will compute both endpoints and exhibit such models.

Uniform comparisons over data laws. If the reference law is unknown, we may want a conversion valid for every P in a class $\mathcal{P}$. The *uniform moduli* are

$$\omega^{\mathcal{P}}_{D\to c}(\kappa) = \sup_{P\in\mathcal{P}} \omega_{D\to c}(\kappa; P), \qquad \omega^{\mathcal{P}}_{c\to D}(s) = \sup_{P\in\mathcal{P}} \omega_{c\to D}(s; P).$$

Enlarging $\mathcal{P}$ can only increase these worst-case conversion bounds. For example, with $0 \le c \le C_{\max}$, take $\mathcal{P}_\sigma = \{P : \sigma_P(c) \le \sigma\}$. Proposition 4.2.9 gives

$$\omega^{\mathcal{P}_\sigma}_{D\to c}(\kappa) \le 2\sigma\sqrt{\kappa} + C_{\max}\kappa.$$

This reformulates the earlier sharpness question: among all data laws whose cost variance is at most σ^2, how much can cost change under a KL budget κ? Throughout, Q still ranges over all laws.

The norm-comparison analogy. For two norms on $\mathbb{R}^d$, sharp bounds $\alpha\|z\|_a \le \|z\|_b \le \beta\|z\|_a$ place $\|z\|_b$ between αr and βr when $\|z\|_a = r$. The ratio β/α measures the multiplicative spread at this fixed input norm. For norms the sharp budget conversion is linear in the budget r: substituting $z = ru$ gives

$$\sup_{\|z\|_a \le r} \|z\|_b = r \sup_{\|u\|_a \le 1} \|u\|_b \qquad (r > 0).$$

Norms scale linearly under multiplication, which forces this form. Our cost–KL conversions instead involve functions such as $\sqrt{\kappa}$ and $1 - e^{-\kappa}$. The shared question is how tightly one measurement determines another. Equivalence of finite-dimensional norms also gives a positive lower endpoint when the fixed input norm is positive. The examples below show that the cost discrepancy can instead be zero at positive KL. Appendix 4.C.3 explains the terminology's connection to continuity and approximation.

We can ask the same questions for other probability metrics, such as TV: which cost discrepancies are compatible with a fixed distance, and what are the sharp conversion functions? The next section will examine TV alongside KL.

4.3 Examples of task cost control at fixed KL

We now use point-mass data distributions to illustrate how tightly fixing forward KL controls task cost. Upper bounds alone leave open whether the smallest and largest compatible costs are close; here we characterize the entire interval and show that they need not be. For small κ, its width is approximately κ times the off-data cost span, so whether it is large depends on the attainable κ, discussed in the next section.

4.3.1 The case of point-mass data distributions

We first calculate the full interval of expected costs at a fixed forward KL for data concentrated on a single successful outcome. Write $P = \delta_{x^\star}$, where $c(x^\star) = 0$. The KL constraint fixes $Q(x^\star)$; we can choose how to distribute the remaining probability.

Proposition 4.3.1 (Cost interval for a point-mass reference)**.** *Let $\mathcal{X}$ have at least two elements, fix $x^\star \in \mathcal{X}$, and let $c : \mathcal{X} \to \mathbb{R}$ satisfy $c(x^\star) = 0$. For $P = \delta_{x^\star}$ and $\kappa \ge 0$,*

$$\{\mathbb{E}_Q[c] : D(P, Q) = \kappa\} = (1 - e^{-\kappa}) \left[\min_{x \ne x^\star} c(x),\ \max_{x \ne x^\star} c(x) \right]. \tag{4.21}$$

The endpoints are attained by placing all probability outside $x^\star$ on a minimizing or maximizing outcome, respectively.

Proof. By algebra, the KL constraint is $-\log Q(x^\star) = \kappa$. Thus every admissible law has the form

$$Q = e^{-\kappa}\delta_{x^\star} + (1 - e^{-\kappa})R, \qquad R(x^\star) = 0. \tag{4.22}$$

Since $c(x^\star) = 0$, we have $\mathbb{E}_Q[c] = (1 - e^{-\kappa})\mathbb{E}_R[c]$, where we are free to choose R subject to $R(x^\star) = 0$. Point masses R at a cost minimizer or maximizer give the two endpoints. Mixtures of these point masses give every intermediate value. □

In words, the result states that KL fixes the mass $1 - e^{-\kappa}$ assigned away from the demonstrated outcome; its average cost can be anywhere between the smallest and largest costs available there.

We note in passing that total variation records the same mass:

$$\|P - Q\|_{\mathrm{TV}} = 1 - Q(x^\star) = 1 - e^{-\kappa}.$$

Thus all models in the interval also have the same TV. The width of the cost interval tells us exactly how much task performance can vary while both distributional discrepancies remain unchanged.

4.3.2 Sequence examples: incorrect steps and final answers

We now ask how models with the same sequence KL can differ in two assessments of a generated solution: its number of incorrect steps and the correctness of its final answer. Applying Proposition 4.3.1 to each cost gives the exact range of possible performance.

Now take $\mathcal{X} = \{0,1\}^T$, with $T \geq 2$, and set $x^\star = 0^T$. Think of this as a simplified model of generating a solution to a mathematical problem. For example, let a one record an error in a T-step argument, and let the final bit record whether the derived answer is incorrect. With this, there are two natural assessments of a derivation $x_{1:T} \in \{0,1\}^T$:

$$c_{\mathrm{cnt}}(x_{1:T}) = \sum_{t=1}^{T} x_t, \qquad c_{\mathrm{fin}}(x_{1:T}) = x_T. \tag{4.23}$$

The *error count* c_{cnt} measures how many steps are wrong. The *terminal cost* c_{fin} measures whether the final answer is wrong; earlier mistakes matter only through their effect on that answer. Under $P = \delta_{0^T}$, the data distribution has zero cost under either assessment.

Proposition 4.3.2 (Error count at fixed KL)**.** *For $P = \delta_{0^T}$ and every $\kappa \geq 0$,*

$$\{\mathbb{E}_Q[c_{\mathrm{cnt}}] : D(P,Q) = \kappa\} = [1 - e^{-\kappa},\, T(1 - e^{-\kappa})], \tag{4.24}$$

where Q ranges over all laws on $\{0,1\}^T$.

Proof. Outside 0^T, the smallest count is 1, attained at (say) 10^{T-1}, and the largest is T, attained at 1^T. Apply Proposition 4.3.1. □

At any positive KL, the largest possible expected count is T times the smallest. Even a sharp upper bound leaves this entire factor-T spread. A model that makes just one mistake on each erroneous sequence and a model that makes T mistakes can have exactly the same likelihood. This does not contradict the bounded-cost guarantee: c_{cnt} has range T, so its KL–TV bound also scales with T; instead, the comparison shows that likelihood need not distinguish recovery from persistence among generators with the same KL.

For terminal cost, an erroneous sequence can still end with a correct answer. This changes the lower endpoint to zero.

Proposition 4.3.3 (Terminal error at fixed KL)**.** *For $P = \delta_{0^T}$, $T \geq 2$, and every $\kappa \geq 0$,*

$$\{\mathbb{E}_Q[c_{\mathrm{fin}}] : D(P,Q) = \kappa\} = [0,\, 1 - e^{-\kappa}], \tag{4.25}$$

where Q ranges over all laws on $\{0,1\}^T$.

Proof. Outside 0^T, terminal cost has minimum 0, attained at 10^{T-1}, and maximum 1, attained at 1^T. Apply Proposition 4.3.1. □

At the same KL, one generator can always give the correct final answer, while another has error probability $1 - e^{-\kappa}$. Here the cost range is fixed at $[0, 1]$, independently of the horizon.

The interval widths are governed by $1 - e^{-\kappa}$, which approaches one for large κ. For small κ,

$$1 - e^{-\kappa} = \kappa + O(\kappa^2) \sim \kappa.$$

The expected count therefore ranges from approximately κ to $T\kappa$, and final-answer error ranges from zero to approximately κ. Small sequence KL gives small terminal error, but models with that same KL can differ throughout the permitted interval. At $\kappa = 1$, that interval extends from zero to approximately 0.632. Whether such a guarantee is useful depends on the task's required accuracy.

4.3.3 Attaining the cost bounds: recovery or persistence

We now construct two sequence laws that attain the lower and upper cost bounds in the preceding subsection. The construction in the proof of Proposition 4.3.1 suggests taking

$$\begin{aligned} Q_- &= e^{-\kappa}\delta_{0^T} + (1 - e^{-\kappa})\delta_{10^{T-1}}, \\ Q_+ &= e^{-\kappa}\delta_{0^T} + (1 - e^{-\kappa})\delta_{1^T}. \end{aligned} \tag{4.26}$$

The law Q_- attains the lowest cost, while Q_+ attains the highest cost for both cost functions considered in the preceding subsection, among all laws Q with $D(\delta_{0^T}, Q) = \kappa$.

Generation under these laws has a similar form. Both start with $X_1 \sim \text{Bernoulli}(1 - e^{-\kappa})$. Under Q_-, one continues with $X_{t+1} = 0$; under Q_+, one continues with $X_{t+1} = X_t$.

Under both laws, only the last bit is needed to generate the next one. This is the Markov property:

Definition 4.3.4 (Markov process)**.** A finite-state discrete-time *stochastic process* is a collection of random variables $(X_t)_{t=1}^T$ taking values in a finite set $\mathcal{X}$. It has the *Markov property*, and is called a *Markov process*, if for every $t < T$ there is a map $K_t : \mathcal{X}^2 \to [0, 1]$ satisfying $\sum_{y \in \mathcal{X}} K_t(x, y) = 1$ for every $x \in \mathcal{X}$, such that, for every $y \in \mathcal{X}$, $\mathbb{P}$-almost surely,

$$\mathbb{P}(X_{t+1} = y \mid X_1, \ldots, X_t) = K_t(X_t, y)\,.$$

The rule K_t is called the process's *transition kernel*. If the same kernel K is used at every time, the process is *time-homogeneous*.

After identifying $\mathcal{X}$ with $[d]$, we can represent K_t by a $d \times d$ nonnegative matrix whose (i, j)th entry is $K_t(i, j)$ and whose rows sum to one.

Both laws can be generated by time-homogeneous Markov processes. The following table gives one choice of transition kernel for each law, specified by $Q(X_{t+1} = 1 \mid X_t = x)$, along with the resulting costs:

	Q_-	Q_+
$Q(X_{t+1} = 1 \mid X_t = x)$	0	x
Expected error count	$1 - e^{-\kappa}$	$T(1 - e^{-\kappa})$
Final-answer error	0	$1 - e^{-\kappa}$

We can interpret the generation law of Q_- as recovering from an initial error, and that of Q_+ as persisting that error.

By the preceding results, the intervals also determine both moduli of continuity introduced in Section 4.2.5. Exercise 4 asks for these conversions and their interpretation: small expected error

count forces small KL here, while perfect final answers permit arbitrarily large KL. In addition, Exercise 9 compares the Pinsker and exact-TV bounds in a variant with an error opportunity at every step; the additional calculations make the same distinction between sharpening a guarantee and choosing a better generator.

4.4 What does likelihood fitting guarantee?

What does likelihood fitting guarantee about the generated solutions when data and computation are limited? We first ask how sequence length and parameter sharing affect learning accuracy, and why lower per-token loss need not mean better task performance. We then examine the accuracy required of the optimizer and how optimization error combines with finite-data error. Together, these comparisons identify what likelihood fitting controls and which differences between generators it leaves unresolved.

4.4.1 Repeated parameters and new parameters

Increasing the length of the training sequences gives us more observations, but fitting longer sequences requires predicting more tokens. Does the additional information compensate for the additional predictions? We compare cases where new positions reuse unknown parameters with cases where they introduce additional unknown parameters, then return to the resulting task-cost ranges.

Fix a data law P and a model class $\mathcal{Q}$ over infinite token sequences. Write $P_{1:T}, Q_{1:T}$ for their length-T prefix laws. At each length T, a learner receives n independent training sequences $X_{1:T}^{(1)}, \dots, X_{1:T}^{(n)}$ and fits a model $\widehat{Q}^{(n,T)}$, a law over infinite token sequences. Define the *expected sequence KL for n training sequences of length T* as

$$\kappa_n(T) = \mathbb{E}\Big[D(P_{1:T}, \widehat{Q}_{1:T}^{(n,T)})\Big]. \tag{4.27}$$

The expectation averages over the training sample with law $P_{1:T}^{\otimes n}$ and any learner randomness. The fitted model can change with both n and T. We suppress the superscript (n,T) when the training sample size and length are clear, writing $\widehat{Q} = \widehat{Q}^{(n,T)}$. Subscripts such as $1:T$ indicate the prefix being evaluated. Unless specified otherwise, we keep n fixed as T increases.

For a fixed prediction task, additional training observations cannot worsen the best attainable statistical accuracy: a learner can ignore them. In particular, from n longer sequences it can keep only the shorter prefixes and run its original procedure. Here we also enlarge the prediction task. For an unchanged model Q, the KL chain rule in Eq. (4.5) gives

$$D(P_{1:T}, Q_{1:T}) \le D(P_{1:T+1}, Q_{1:T+1}).$$

Learning from the additional observations can improve the fitted model. We ask whether that improvement compensates for the new prediction terms.

For models fitted by iterative optimization, exploiting better statistical accuracy also calls for a sufficiently small optimization error. Thus the appropriate fitting tolerance depends on both the amount of data and the sequence length. Section 4.4.4 makes this balance explicit.

A simple estimation problem exhibits both possibilities. First suppose all tokens are independent Bernoulli(p) variables. Let P denote the resulting joint law. Estimate their common parameter using the smoothed frequency

$$\widehat{p} = \frac{1 + \sum_{i=1}^{n} \sum_{t=1}^{T} X_t^{(i)}}{nT + 2}.$$

(Smoothing avoids the infinite expected KL of the ordinary empirical frequency; see Exercise 14.) The fitted model $\widehat{Q}^{(n,T)}$ is the independent Bernoulli bit model with parameter $\widehat{p}$. For fixed $p \in (0,1)$, the large-sample calculation gives

$$\mathbb{E}[(\widehat{p}-p)^2] \sim \frac{p(1-p)}{nT}, \qquad \kappa_n(T) = T\,\mathbb{E}[\mathrm{kl}(p,\widehat{p})] \longrightarrow \frac{1}{2n} \quad (T\to\infty). \tag{4.28}$$

Exercise 14 asks you to prove this and gives hints on the calculation. Note that each new position supplies another observation of the same unknown parameter. The gain in estimation accuracy compensates for the longer prediction task, giving $\lim_{T\to\infty}\kappa_n(T) = 1/(2n)$.

At the other extreme, let the tokens be independent with unrelated parameters p_t. In particular, for $t \geq 1$, X_t is Bernoulli with parameter p_t. Estimate each p_t separately from the n training sequences, using

$$\widehat{p}_t = \frac{1+\sum_{i=1}^n X_t^{(i)}}{n+2}.$$

Each parameter still receives only n observations. For fixed n, each estimation risk remains positive; if $p_t \in [a,1-a]$ for a fixed $0<a<1/2$, their sum grows linearly with T. The large-n calculation identifies the leading coefficient: for fixed T and $p_1,\dots,p_T \in (0,1)$,

$$\kappa_n(T) = \sum_{t=1}^T \mathbb{E}[\mathrm{kl}(p_t,\widehat{p}_t)] \sim \frac{T}{2n} \quad (n\to\infty). \tag{4.29}$$

Thus the leading estimation cost grows with the number of positions. The calculation describes the same fitting method in two model classes: one shared parameter versus one parameter per position.

Partial sharing. Fix a known partition $G_1, G_2, \dots$ of the positive integer positions, ordered by first occurrence. Tokens are independent, with $\mathbb{P}(X_t=1)=p_j$ for $t \in G_j$. Let $d(T)$ be the number of groups represented among the first T positions, and let $m_j = |G_j \cap [T]|$. Pool all observations belonging to the same group and fit independent bits with the estimated probabilities

$$\widehat{p}_j = \frac{1+\sum_{i=1}^n \sum_{t\in G_j\cap[T]} X_t^{(i)}}{nm_j+2}. \tag{4.30}$$

The resulting sequence KL grows with the number of unknown parameters: each contributes approximately $1/(2n)$.

Proposition 4.4.1 (Sequence KL under parameter sharing)**.** *In the grouped independent-bit model, fit each probability by the pooled smoothed frequency from n independent length-T training sequences. Fix $0<a<1/2$ and suppose $p_j \in [a,1-a]$. Then*

$$\kappa_n(T) \asymp_a \frac{d(T)}{n}, \qquad n,T \geq 1. \tag{4.31}$$

Here $\asymp_a$ means upper and lower bounds by positive constant multiples, with constants depending only on a. If $n\min_{1\leq j\leq d(T)} m_j \to \infty$, then more precisely

$$\kappa_n(T) \sim \frac{d(T)}{2n}. \tag{4.32}$$

Each independently estimated parameter therefore contributes approximately $1/(2n)$, regardless of how many positions use it. To see the cancellation, group j supplies nm_j training observations and contributes m_j terms to the test-sequence KL:

$$\kappa_n(T) = \sum_{j=1}^{d(T)} m_j \, \mathbb{E}[\mathrm{kl}(p_j, \widehat{p}_j)] \approx \sum_{j=1}^{d(T)} m_j \frac{1}{2nm_j} = \frac{d(T)}{2n}.$$

The condition in the proposition says that every pooled sample size becomes large, so this approximation is accurate across all groups. Exercise 14 develops the proof; Appendix 4.D.2 gives the proof.

The two extremes are $d(T) = 1$ and $d(T) = T$. Intermediate sharing patterns give intermediate growth of the estimation cost. For example, using one new parameter for each successive block of lengths $1, 2, 3, \ldots$ gives $d(T)$ of order $\sqrt{T}$. A fixed finite collection of known candidate models offers another possibility: exact identification can become sufficiently reliable that $\kappa_n(T)$ tends to zero (even with n fixed). Exercise 20 asks for this calculation with two Bernoulli candidates.

These comparisons keep the number of independent sequences n fixed, allowing the training-token budget nT to grow. If instead we fix $L = nT$, write $\kappa'_L(T) := \kappa_{L/T}(T)$. In the fully shared model, substituting $n = L/T$ and $d(T) = 1$ into Eq. (4.32) gives

$$\kappa'_L(T) \sim \frac{1}{2(L/T)} = \frac{T}{2L} \qquad (L \to \infty). \tag{4.33}$$

Thus, at a fixed large training-token budget, increasing T worsens expected sequence KL, which grows approximately linearly. Exercise 17 also asks for the unshared comparison.

For the class with $p_j \in [a, 1-a]$, the order $d(T)/n$ is also *minimax optimal*: no learner improves its worst-case dependence on n and $d(T)$; see the bibliographical notes.

Proposition 4.4.2 (Learning rates and task-cost intervals)**.** *For any d with $1 \le d(T) \le T$, there exist a data law, terminal and counting costs, and estimator families such that, for all $n \ge 1, T \ge 2$, the estimators have common expected sequence KL $\kappa_n(T) \asymp d(T)/n$ and their expected costs range over exactly*

$$\left[0, 1 - e^{-\kappa_n(T)/2}\right] \quad \text{and} \quad \left[1 - e^{-\kappa_n(T)/2}, T(1 - e^{-\kappa_n(T)/2})\right],$$

respectively.

To get this result, combine the grouped Bernoulli construction with Section 4.3, using tokens $X_t = (U_t, V_t)$, with costs computed from $V_{1:T}$. Exercise 19 develops the proof.

What should we expect for transformers? In a mathematical solution, many steps apply familiar rules, while later steps may require combining definitions and intermediate results across a long prefix. The statistical question is how much useful shared structure training can discover. The computational question is whether the architecture can carry out the required combinations with its available depth, width, and memory. The Bernoulli examples isolate the statistical benefit when the sharing pattern is known. Exercise 21 studies a separate obstruction: a short copying rule whose required memory grows.

For mathematical reasoning, a plausible working hypothesis is substantial reuse together with additional demands as problems grow. The grouped model gives one way to quantify that hypothesis; the number $d(T)$ counts its independently estimated probabilities. Which sharing is useful and executable in a transformer is a question about the task and the learned computation. Exercise 23 studies a missed parity dependency as an exercise in model-class restrictions and optimization accuracy. Lecture 18 develops the statistical horizon comparison further.

4.4.2 The pitfalls of reporting per-token log loss

As sequence length grows, log loss per token can vanish while final-answer error stays constant. For the all-zero law P and persistent generator Q_+ from Eq. (4.26), with fixed $\kappa > 0$,

$$\frac{D(P_{1:T}, (Q_+)_{1:T})}{T} = \frac{\kappa}{T} \longrightarrow 0, \qquad \mathbb{E}_{(Q_+)_{1:T}}[c_{\text{fin}}] = 1 - e^{-\kappa}. \tag{4.34}$$

The lower per-token loss comes entirely from dividing the same sequence loss by a larger T. When comparing different sequence lengths, lower per-token log loss alone is therefore insufficient evidence of better final-answer accuracy.

4.4.3 Likelihood fitting at finite accuracy

We use loss per predicted token as the optimization objective. For n observed sequences, define empirical *per-token* log loss

$$\widehat{L}(Q) = \frac{1}{nT} \sum_{i=1}^{n} \sum_{t=1}^{T} -\log Q(X_t^{(i)} \mid X_{<t}^{(i)}).$$

This averages the losses of the nT token predictions. We use this normalization despite the warning in Section 4.4.2: per-token loss can improve as sequence length grows without task performance improving. There are two reasons for using it. First, when token losses have comparable scales, normalization removes the explicit factor of T that would otherwise multiply the objective and its gradient. For gradient descent, this avoids a corresponding adjustment of the stepsize and helps us use common optimization settings across sequence lengths. Second, log loss is conventionally reported per token.[3]

When an optimizer is run with finite compute, it generally does not return a model with the lowest possible empirical loss. The *optimization error* of its returned model $\widehat{Q}$ is

$$E(\widehat{Q}) := \widehat{L}(\widehat{Q}) - \inf_{Q \in \mathcal{Q}} \widehat{L}(Q). \tag{4.35}$$

We say that $\widehat{Q}$ satisfies the *optimization tolerance* $\eta \geq 0$ if $E(\widehat{Q}) \leq \eta$, equivalently,

$$\widehat{L}(\widehat{Q}) \leq \inf_{Q \in \mathcal{Q}} \widehat{L}(Q) + \eta. \tag{4.36}$$

Establishing whether a returned model satisfies this tolerance is a *certification question*. For example, suppose $\mathcal{Q} = \{Q_\theta : \theta \in \mathbb{R}^d\}$ and $f(\theta) = \widehat{L}(Q_\theta)$ is differentiable and strongly convex with known parameter $\mu > 0$. Then

$$E(Q_\theta) \leq \frac{\|\nabla f(\theta)\|_2^2}{2\mu},$$

so $\|\nabla f(\theta)\|_2 \leq \sqrt{2\mu\eta}$ certifies the tolerance (Exercise 27). A small gradient alone gives no such guarantee for a general nonconvex loss. We still wanto to study the consequences of positive optimization error even when its size is unknown to the optimizer because this helps with understanding the impact of putting more effort into optimization.

Because of the normalization by T, a per-token tolerance η only guarantees a per-sequence tolerance $T\eta$. Our earlier cost bounds motivate controlling the per-sequence error. Maintaining a fixed per-sequence tolerance as T increases requires increasingly accurate per-token optimization, which may require more compute. Whether a fixed number of passes through the dataset suffices depends on the optimization problem and how the number of updates scales with T.[4]

Positive optimization error can translate into poor task performance, just as sequence-level KL error can. Since η is measured per token, the relevant sequence-level quantity is $T\eta$. The simplest illustration uses the deterministic data law $P = \delta_{0^T}$ from Section 4.3.3: empirical and population log loss coincide, so there is no statistical error to account for. Reusing that example gives the following result.

Proposition 4.4.3 (Different task costs at the same optimization tolerance)**.** *Let $T \geq 2$. Take the laws Q_-, Q_+ in Eq. (4.26) with $\kappa = T\eta$, and let the model class contain P, Q_-, Q_+. For every sample of n sequences from $P = \delta_{0^T}$,*

$$\begin{aligned} \widehat{L}(P) = 0, \qquad \widehat{L}(Q_-) = \widehat{L}(Q_+) = \eta, \\ \mathbb{E}_{Q_-}[c_{\mathrm{fin}}] = 0, \qquad \mathbb{E}_{Q_+}[c_{\mathrm{fin}}] = 1 - e^{-T\eta}. \end{aligned} \tag{4.37}$$

Both generators satisfy Eq. (4.36).

Proof. Each data sequence is 0^T. Both generators assign it probability $e^{-T\eta}$, so their loss per token is η. Their terminal costs follow from Eq. (4.26). □

For small $T\eta$, the permitted final-answer error is approximately $T\eta$. For example, $\eta = 10^{-3}$ and $T = 1000$ permit anything from zero terminal error to about 0.632. More demonstrations of 0^T leave this choice unchanged.

If the class consists of Q_-, Q_+ at a fixed $\kappa > 0$, then the data law P is outside the class. Both generators minimize empirical and population log loss, yet their terminal costs are zero and $1 - e^{-\kappa}$. Thus, in addition to optimization error, *approximation error* can also leave a consequential choice between equally fitted models. Exercise 28 develops this comparison with repeated errors, and Exercise 23 studies a missed dependency in a parity task.

4.4.4 What does finite-sample likelihood fitting guarantee?*

Likelihood fitting gives a positive statistical guarantee on task performance. The following stylized result bounds the task cost of choosing a model with nearly minimal empirical log loss from a finite class. Allowing optimization slack, as in the previous subsection, lets us ask how optimization effort should depend on sample size.

Theorem 4.4.4 (Approximate maximum likelihood controls cost)**.** *Let $\mathcal{Q}$ be a finite class containing P, and observe n independent sequences with law P. Let $0 \leq c \leq C_{\max}$. Suppose $\widehat{Q} \in \mathcal{Q}$ satisfies Eq. (4.36) for $\eta \geq 0$. For $0 < \delta < 1$, put*

$$b = \frac{2\log(|\mathcal{Q}|/\delta)}{n} + T\eta.$$

With probability at least $1 - \delta$,

$$d_c(P, \widehat{Q}) \leq 2\sigma_P(c)\sqrt{b} + C_{\max} b. \tag{4.38}$$

In particular, if $\mathbb{E}_P[c] = 0$, then $\mathbb{E}_{\widehat{Q}}[c] \leq C_{\max} b$.

The proof in Appendix 4.D.1 uses the same Hellinger-to-cost inequality as Proposition 4.2.9 (Proposition 4.C.1). The earlier result bounds squared Hellinger distance by forward KL; here the likelihood argument bounds it directly by b. Forward KL itself need not be finite, so the earlier proposition alone does not give this theorem.

For zero-cost data and $0 \le c \le 1$, the theorem separates the two contributions to its guarantee:

$$\mathbb{E}_{\widehat{Q}}[c] \le \underbrace{\frac{2\log(|\mathcal{Q}|/\delta)}{n}}_{\text{statistical contribution}} + \underbrace{T\eta}_{\text{optimization contribution}}. \tag{4.39}$$

At fixed model class and confidence level, the statistical contribution is of order $1/n$ and remains in the bound even with exact optimization. If optimization effort is insufficient, the optimization contribution $T\eta$ may dominate. To keep it no larger than the statistical contribution, it suffices to achieve

$$\eta \le \frac{2\log(|\mathcal{Q}|/\delta)}{nT}. \tag{4.40}$$

Substitution into Eq. (4.39) gives $\mathbb{E}_{\widehat{Q}}[c] \le 4\log(|\mathcal{Q}|/\delta)/n$, with probability at least $1-\delta$. *Optimizing to statistical accuracy* means keeping the optimization contribution on the scale of that statistical term. Even at fixed sequence length, retaining this statistical rate calls for a decreasing per-token tolerance as the number of examples grows.

The example in Proposition 4.4.3 shows that the optimization contribution has the correct order when $T\eta$ is small.

Lecture 18 develops this comparison for imitation learning. The comparison in Section 4.4.1 fixes infinite-sequence models and varies the length of the observed prefixes. Its Bernoulli examples isolate parameter sharing; the present finite-class theorem also accounts for optimization tolerance. Fixing the number of sequences gives a different comparison from fixing the token budget.

4.5 Beyond likelihood fitting can we improve task performance?

The main conclusion of the previous section is that in terms of task performance, likelihood fitting may leave a large room for improvement. In this section we investigate how to decrease this room. For this, we consider two broad approaches. The first considers changing the objective to incorporate information about the consequences of generation, while the second one gives the models more structure to support recovery from errors. Recovery then raises a further question: which information should be forgotten, and which must be preserved?

4.5.1 Evaluate consequences of generation

Direct cost feedback. In some cases, the cost function is available for evaluation. That is, there is a cheap way to get the value of c at any sequence $x_{1:T}$. (The cost may be unavailable, for example, when evaluating $c(x_{1:T})$ requires a human and no human is available.) However, when c is available for measurement, this creates a new opportunity to get a good model. One can simply run an optimization process that aims to minimize

$$\min_{\theta} \mathbb{E}_{Q_\theta}[c]. \tag{4.41}$$

The procedures that aim to perform this optimization are discussed later. Lecture 19 studies preference feedback and the objectives constructed from it. Lecture 20 develops reinforcement learning with generated feedback, including the policy-gradient method and optimization with learned evaluators.

In practice, one often combines several data sources and objectives. Combining the objectives is coherent when they are aligned, as when teacher forcing with positive data is used alongside the optimization objective above. The reason to combine the objectives is to take advantage of their

complementary strengths. For example, consider minimizing the expected terminal cost c_{fin}. As we have seen, Q_+ and Q_- of Section 4.3.3 both have the same log-likelihood, while the expected cost of Q_- is zero, and the expected cost of Q_+ is $1 - e^{-\kappa}$. If the expected cost can be calculated up to sufficient accuracy (less than the performance gap between the models), the better, Q_- can be chosen, which is a large win compared to not using cost information.

Direct access to task cost does not imply that its Euclidean gradient gives the best training dynamics. A surrogate objective can preserve the desired limiting decision while giving a better-conditioned optimization problem. Changing the parameter-space geometry, as in natural gradient, can have the same purpose without replacing the task objective. In the one-step binary problem of Exercise 31, the exact expected cost is smooth and its gradient can be evaluated without sampling. Nevertheless, from a confidently wrong policy, ordinary gradient flow on that cost takes time exponential in the initial log-odds deficit to reach even odds. Log-loss gradient flow and natural-gradient flow take only linear time. Thus positive examples and direct cost feedback can have complementary roles: the cost specifies the desired behavior, while log loss can supply a useful training signal where the direct cost gradient is small. The example isolates this optimization effect from exploration, delayed credit, and action-sampling noise.

Predicting further ahead. Positive demonstrations can supervise predictions several steps into the future as well as predictions of the next token. The model must predict X_{t+k} from $X_{\le t}$, before the intervening tokens are supplied. For an observed sequence $x_{1:T}$ and a model with $Q(x_{1:T}) > 0$, define the *k-step prediction loss*, for $1 \le k \le T$, by

$$\mathcal{L}_k(Q) = -\sum_{t=0}^{T-k} \log Q(X_{t+k} = x_{t+k} \mid X_{\le t} = x_{\le t}). \tag{4.42}$$

The prefix is empty when $t = 0$. In particular, $\mathcal{L}_1(Q) = -\log Q(x_{1:T})$ is the usual sequence log loss. One can add $\lambda\mathcal{L}_k(Q)$, with $\lambda > 0$ and $k \ge 2$, to this objective and average over the demonstrations.

For the laws in Section 4.3.3, evaluated on 0^T,

$$\mathcal{L}_2(Q_-) = 0, \qquad \mathcal{L}_2(Q_+) = \kappa. \tag{4.43}$$

Their next-token losses agree. At $\kappa > 0$, any positive weight on $\mathcal{L}_2$ therefore favors Q_-, which recovers from its initial error, over Q_+, which preserves it. The calculation is in Exercise 32.[5]

TODO: Find substantive conditions under which adding a short-lookahead loss improves task performance over next-token fitting, beyond the Q_-, Q_+ example. A bound in terms of achieved multistep loss alone does not establish this comparison.

The computational challenge is to evaluate and optimize these losses. For a general autoregressive model,

$$Q_\theta(X_{t+k} = a \mid x_{\le t}) = \sum_{y_{1:k-1} \in [N]^{k-1}} \left[\prod_{j=1}^{k-1} Q_\theta(y_j \mid x_{\le t}, y_{<j})\right] \cdot Q_\theta(a \mid x_{\le t}, y_{1:k-1}). \tag{4.44}$$

Direct evaluation sums over N^{k-1} possible intervening sequences; for a large vocabulary, even $k = 2$ can be costly. Supplying the observed intervening tokens instead would return to a next-token prediction problem. An auxiliary prediction head can instead be trained to predict the future token directly from the prefix, without evaluating this sum. Its predictions need not agree with the generator's k-step marginals.[6]

The intervening tokens are latent variables in this prediction problem: Eq. (4.44) is a conditional likelihood obtained by summing them out. Lecture 5 returns to the same operation for hidden states. Exercise 33 develops the calculation and shows how Markov structure makes it efficient.

Fit reverse KL. Forward KL penalizes failing to cover outcomes that are likely under P. Reverse KL penalizes generating outcomes to which P assigns too little probability. When a model cannot reproduce P, this distinction matters: we may prefer to generate a smaller variety of successful responses if covering more responses would also introduce failures.

To quantify the penalty on failures (positive costs), let $B = \{x : c(x) > 0\}$, $\varepsilon = P(B) \in (0, 1)$, and $q = Q(B)$. Data processing gives

$$D(Q, P) \geq \mathrm{kl}(q, \varepsilon) \geq q \log(1/\varepsilon) - \log 2. \tag{4.45}$$

Thus, reverse KL increases at least at a linear speed with the failure probability $q = Q(B)$ with steepness $\log(1/\varepsilon)$, which diverges as $\varepsilon \to 0$. Conversely, a bounded reverse KL forces failures to become rare when they are rare under P. Theorem 4.2.8 gives a related cost guarantee using a subgaussian variance proxy; Eq. (4.45) directly controls failure probability.

Finite reverse KL requires staying in the support of P. When $P(B) = 0$, finite reverse KL implies zero task cost. But finite reverse KL can also exclude probability laws that have zero task cost: Q_- has zero terminal cost but infinite reverse KL from $P = \delta_{0^T}$ (just like Q_+). Exercise 35 develops the failure bound and the complementary calculations for adding unsuccessful outcomes and omitting successful ones.

The information needed for reverse-KL fitting. Forward KL averages conditional discrepancies at data prefixes. Reversing its arguments averages them at generated prefixes:

$$D(Q, P) = \sum_{t=1}^{T} \mathbb{E}\Big[D\big(Q(\cdot \mid \widetilde{X}_{<t}), P(\cdot \mid \widetilde{X}_{<t})\big)\Big], \qquad \widetilde{X}_{1:T} \sim Q. \tag{4.46}$$

Thus the generator's own erroneous histories contribute to the objective

$$D(Q_\theta, P) = \mathbb{E}[\log Q_\theta(\widetilde{X}) - \log P(\widetilde{X})], \qquad \widetilde{X} \sim Q_\theta. \tag{4.47}$$

For the unknown data law P, the probabilities $P(\widetilde{X})$ are unavailable. Thus this expression does not directly provide an objective we can evaluate from the training data. A *variational representation* avoids requiring these probabilities and leads to an *adversarial* fitting method. For $X \sim P$ and $Y \sim Q$ on the finite sequence space $\mathcal{X} = [N]^T$,

$$D(Q, P) = \sup_{g:\mathcal{X}\to\mathbb{R}} \left\{1 + \mathbb{E}[g(Y)] - \mathbb{E}[e^{g(X)}]\right\}. \tag{4.48}$$

The function g is called a *critic*. When P, Q are positive, the maximizer is $g^*(x) = \log(Q(x)/P(x))$. Exercise 38 asks for the proof. Both expectations can be estimated from samples, without evaluating $P(x)$. This suggests alternating updates: fit the critic to maximize the expression, then fit the generator to minimize it with the critic fixed.

Restricting the critic to a tractable class yields only a lower bound on $D(Q, P)$, and fitting it from finite data requires generalization. Thus a small fitted objective does not certify small reverse KL. The support requirement remains: if Q assigns mass where P is zero, the unrestricted supremum is infinite; a restricted critic may fail to detect this.[7]

Direct reverse-KL fitting is nevertheless useful for model compression through *model distillation*: a smaller student approximates an existing teacher whose probabilities can be evaluated. Here the target is the teacher's distribution. Both forward and reverse KL are used for model distillation.[8]

4.5.2 Recovery at unchanged likelihood

Suppose the task is to produce a correct final answer at position T. For a fixed question, let $c(x) = \mathbf{1}\{x \text{ is incorrect}\}$. The expected cost $\mathbb{E}_Q[c(X_T)]$ depends only on the distribution of X_T. This distribution is the *marginal law at position* T: we obtain it from the joint law by summing over the preceding tokens. More generally, let μ_t, ν_t denote the marginals under P, Q, respectively:

$$\mu_t(x) = P(X_t = x), \qquad \nu_t(x) = Q(X_t = x)\,, \qquad t \in [T]\,.$$

By Proposition 4.2.2, we have

$$|\mathbb{E}_P[c(X_T)] - \mathbb{E}_Q[c(X_T)]| \le \|\mu_T - \nu_T\|_{\mathrm{TV}}.$$

If the data always end with a correct answer, $\mathbb{E}_P[c(X_T)] = 0$, so controlling this marginal discrepancy controls the generated answer's error probability.

An early mistake can be corrected before the final answer. We will quantify this recovery when the fitted process Q is Markov, and then exhibit improved final-answer accuracy at unchanged likelihood. The same argument controls cumulative costs $\sum_{t=1}^{T} c_t(X_t)$; Exercise 42 asks for the bound. Appendix 4.F extends the result to history-dependent generators by controlling how long a changed token affects later predictions.

For a law μ and kernel K, write

$$(\mu K)(y) = \sum_x \mu(x) K(x, y).$$

Let Q have transition kernels $\widehat{K}_t$, so $\nu_{t+1} = \nu_t \widehat{K}_t$. The data law P can depend on the full history. We measure local prediction error at its prefixes, as in likelihood fitting.

Theorem 4.5.1 (Recovery and final-answer accuracy)**.** *Suppose that, for $1 \le t < T$, the fitted kernels satisfy*

$$\|\alpha \widehat{K}_t - \beta \widehat{K}_t\|_{\mathrm{TV}} \le \rho \|\alpha - \beta\|_{\mathrm{TV}} \quad \textit{for all laws } \alpha, \beta, \qquad 0 \le \rho \le 1, \tag{4.49}$$

and their local errors under P satisfy

$$\mathbb{E}_P\Big[\|P(\cdot \mid X_{\le t}) - \widehat{K}_t(X_t, \cdot)\|_{\mathrm{TV}}\Big] \le \varepsilon_{t+1}. \tag{4.50}$$

Write $e_t = \|\mu_t - \nu_t\|_{\mathrm{TV}}$. For every $c : [N] \to [0, 1]$,

$$|\mathbb{E}_P[c(X_T)] - \mathbb{E}_Q[c(X_T)]| \le e_T \le \rho^{T-1} e_1 + \sum_{t=2}^{T} \rho^{T-t} \varepsilon_t. \tag{4.51}$$

If $\rho < 1$, $e_1 \le \varepsilon$, and $\varepsilon_t \le \varepsilon$ for all $2 \le t \le T$, then

$$\max_{1 \le t \le T} |\mathbb{E}_P[c(X_t)] - \mathbb{E}_Q[c(X_t)]| \le \frac{\varepsilon}{1 - \rho}. \tag{4.52}$$

Proof. The laws μ_{t+1} and $\mu_t \widehat{K}_t$ average the true and fitted next-token conditionals over the same data prefixes:

$$\mu_{t+1} = \mathbb{E}_P[P(\cdot \mid X_{\le t})], \qquad \mu_t \widehat{K}_t = \mathbb{E}_P[\widehat{K}_t(X_t, \cdot)].$$

Their TV distance is therefore at most ε_{t+1}. Insert $\mu_t \widehat{K}_t$ between μ_{t+1} and $\nu_{t+1} = \nu_t \widehat{K}_t$:

$$e_{t+1} \le \|\mu_{t+1} - \mu_t \widehat{K}_t\|_{\mathrm{TV}} + \|\mu_t \widehat{K}_t - \nu_t \widehat{K}_t\|_{\mathrm{TV}} \le \varepsilon_{t+1} + \rho e_t.$$

Iteration and the TV cost bound give Eq. (4.51). Under the uniform error bounds, $e_t \le \varepsilon \sum_{j=0}^{t-1} \rho^j \le \varepsilon/(1 - \rho)$. Apply Proposition 4.2.2 to c at each time and take the maximum. □

In the final-answer bound, the local error at time t is multiplied by ρ^{T-t}. Recovery reduces the effect of early errors; a mistake in the final prediction has no remaining steps in which to be corrected. With repeated local errors of size at most ε, their combined effect is bounded by $\varepsilon/(1-\rho)$, independently of the horizon. The argument also extends to other distances between distributions, such as Wasserstein-1 distance with Lipschitz costs and contraction in that distance.

We can check contraction by comparing the next-token laws at different current states. For a finite-state kernel, the smallest possible contraction factor is the *Dobrushin ergodicity coefficient*, given by

$$\rho(\widehat{K}) = \max_{x,x'} \|\widehat{K}(x,\cdot) - \widehat{K}(x',\cdot)\|_{\mathrm{TV}}. \tag{4.53}$$

Exercise 39 proves this characterization.

Recovery in the binary example. We now improve the persistent law Q_+ by giving each error a chance to be corrected. Here $c(x) = x$, so $c(X_T) = c_{\mathrm{fin}}(X_{1:T})$ is the final-answer error. The per-position costs $c_t(x) = x$ give c_{cnt}.

Fix $P = \delta_{0^T}$ and $\kappa \geq 0$. Let Q_r use the same first-bit distribution as Q_- and Q_+, followed by

$$Q_r(X_1 = 1) = 1 - e^{-\kappa},$$
$$Q_r(X_{t+1} = 1 \mid X_t = 0) = 0, \qquad Q_r(X_{t+1} = 0 \mid X_t = 1) = r, \quad 1 \leq t < T,$$

where $0 \leq r \leq 1$. Once an error is corrected, all later bits are zero. The choices $r = 0$ and $r = 1$ give Q_+ and Q_-, respectively.

Proposition 4.5.2 (Recovery improves final-answer accuracy at unchanged likelihood)**.** *For this family and $T \geq 2$,*

$$D(P, Q_r) = \kappa, \qquad \mathbb{E}_{Q_r}[c_{\mathrm{fin}}] = (1 - e^{-\kappa})(1 - r)^{T-1}. \tag{4.54}$$

For $r > 0$,

$$\mathbb{E}_{Q_r}[c_{\mathrm{cnt}}] = (1 - e^{-\kappa})\frac{1 - (1-r)^T}{r} \leq \frac{1 - e^{-\kappa}}{r}. \tag{4.55}$$

At $r = 0$, the expected count is $T(1 - e^{-\kappa})$.

Proof. The all-zero path has probability $e^{-\kappa}$, independently of r. Also,

$$Q_r(X_t = 1) = (1 - e^{-\kappa})(1 - r)^{t-1},$$

since the first bit must be wrong and each subsequent correction must fail. Taking $t = T$ gives terminal error; summing over t gives the expected count. □

At a fixed positive recovery probability, terminal error decays geometrically and expected error count stays bounded as the horizon grows. An erroneous run has geometric duration with mean $1/r$ before truncation at the horizon. Likelihood constrains the probability of the initial error; the probability r of returning to zero determines how likely it is to persist until the final answer.[9]

For Q_r, the two rows differ by $1 - r$, so its kernels after the first token have contraction factor $1 - r$. The initial discrepancy is $e_1 = 1 - e^{-\kappa}$, and all subsequent local errors under P are zero. The theorem therefore gives

$$e_T \leq (1 - r)^{T-1}(1 - e^{-\kappa}),$$

which is exactly the terminal cost in Eq. (4.54). Exercise 39 adds fresh errors at every step and computes the resulting long-run error level. The next question is how to obtain recovery while retaining information the task needs.

4.5.3 What should be forgotten?

Random resets give a simple mathematical way to enforce forgetting. For any kernel K and reset law π_{reset}, the kernel

$$K_r(x, \cdot) = (1 - r)K(x, \cdot) + r\pi_{\text{reset}}(\cdot)$$

has contraction factor at most $1 - r$. A reset makes the two initial states use the same distribution. But forgetting everything can erase exactly the information a task needs. A proof generator must retain the theorem being proved and useful facts established earlier.

Limited context and contraction are different properties. The rule $X_{t+1} = X_t$ uses only one token of context and preserves its initial bit forever. During generation, recent tokens can keep transmitting old information. Restricting a transformer's accessible prefix therefore does not establish Eq. (4.49).

For prompting, a sliding window creates a more immediate difficulty: a fact in the original prompt can leave the accessible context before the model has carried it forward. Subsequent predictions then have no direct access to it. Generated reminders can preserve information, as in the copying example, but can also introduce errors into what is retained. Keeping the original prompt or verified task information separately provides a more reliable source than repeatedly reconstructing it from generated text.

A more useful design separates persistent task information M from a revisable working state R_t:

$$S_t = (M, R_t), \qquad R_{t+1} = g_M(R_t, \Xi_{t+1}). \tag{4.56}$$

Conditional on the same M, the working dynamics can forget perturbations while M retains the problem and trusted information. For example, in proof generation this suggests retaining the problem statement and checked lemmas while restarting a failed attempt. The mathematical question is which state differences can be made contractive without destroying information the task needs. Lecture 5 develops state-space models as an explicit way to express this separation.

4.6 Take-home messages

1. Next-token log-loss minimization is sequence maximum likelihood. Its population excess loss is forward sequence KL.

2. Successful demonstrations specify a distribution to imitate. The task objective is the expected cost of generated sequences. TV and KL bounds connect these objectives by limiting the expected cost discrepancy.

3. At the same forward KL κ and TV, the laws Q_- and Q_+ have expected counts $1 - e^{-\kappa}$ and $T(1 - e^{-\kappa})$, and terminal error probabilities zero and $1 - e^{-\kappa}$. Approximate likelihood fitting can leave this choice unresolved: a per-token tolerance η permits per-sequence excess $T\eta$.

4. Longer training sequences provide more information, while longer predictions may require learning additional structure. In the grouped Bernoulli model, $d(T)$ unknown shared probabilities give expected sequence KL at most $d(T)/n$. The persistent law Q_+ shows that a vanishing average-loss gap can coexist with unchanged task error as the horizon grows.

5. Recovery attenuates an initial error; continued prediction errors contribute according to how long their effects persist. Contraction bounds both terms. Useful task memory may still need to persist.

6. Forward KL penalizes failing to cover likely data outcomes; reverse KL penalizes assigning too much probability to outcomes rare under the target. Adversarial fitting uses samples from both laws, but a small objective with a restricted critic does not certify small reverse KL.

7. Multi-step objectives integrate over missing intermediate tokens. Direct cost optimization instead uses evaluations on model generations.

Endnotes

1. **Conventions at zero.** These conventions are the continuous extensions of the nonzero formula and express the infinite log loss incurred when a positive-probability outcome is assigned probability zero; Lecture 3, Exercise 2(e), verifies both limits.

2. **Variance proxy and variance.** How does the optimal proxy compare with the variance? Write $\sigma_P(c)^2 = \mathbb{E}_P[(c - \mathbb{E}_P[c])^2]$. Then

$$\sigma_P(c)^2 \le v_P(c) \le \frac{C_{\max}^2}{4} \qquad \text{when } 0 \le c \le C_{\max}.$$

   The lower bound follows by expanding Eq. (4.13) at $\lambda = 0$; the upper bound is Hoeffding's lemma. Thus $\sqrt{v_P(c)}$, the subgaussian fluctuation scale, lies between $\sigma_P(c)$ and $C_{\max}/2$. For rare errors the proxy can greatly exceed the variance; Appendix 4.C.2 gives a Bernoulli example. The bounds and existence of the minimum are proved in Appendix 4.B.2.

3. **Unequal response lengths.** For response y_i to prompt u_i, let T_i count its predicted tokens and put $\ell_i(Q) = -\log Q(y_i \mid u_i)$. Two averaging conventions are

$$\widehat{L}_{\text{token}}(Q) = \frac{\sum_i \ell_i(Q)}{\sum_i T_i}, \qquad \widehat{L}_{\text{response}}(Q) = \frac{1}{n}\sum_i \frac{\ell_i(Q)}{T_i}.$$

   The first weights predicted tokens equally; the second weights each response's average loss equally. Prompt and padding tokens excluded from the loss are also excluded from T_i. Both conventions reduce to the formula in Section 4.4.3 when every $T_i = T$. Both are used in supervised fine-tuning; see the bibliographical notes.

4. **Fixed passes and finer accuracy.** Consider smooth, strongly convex per-token objectives with constants uniform in n, T, bounded initial gaps, and unbiased stochastic gradients with uniformly bounded variance. With suitable decreasing stepsizes, stochastic gradient descent gives expected optimization error $O(1/K)$ after K updates. If batches contain a fixed number of uniformly sampled token losses, $K = \Theta(nT)$ updates process a fixed multiple of the dataset's token count and give expected per-sequence error $O(1/n)$. This counts token-loss evaluations; their computational cost can depend on context length. The bound averages over optimizer randomness and does not certify an individual run. Full-gradient descent uses one pass per update, so fixed passes give fixed K. Even for $f(\theta) = \theta^2/2$, starting at $\theta_0 = 1$ with stepsize $1/2$ gives error 2^{-2K-1}, independent of T; reaching s/T for fixed $s > 0$ requires order $\log T$ updates. Thus passes alone do not determine the achieved tolerance. The stochastic rate also need not hold for a general neural-network loss. Lecture 10 returns to when local gradients ensure global convergence; see the bibliographical notes for the assumptions and rate.

5. **A separate benefit under misspecification.** A finite model class may contain a law with the correct marginal of $Y = f(X_{1:T})$, even when no model matches the full sequence law. The finite-class guarantee in Theorem 4.4.4 then applies to fitting the marginal likelihood of Y and any cost determined by Y. For a Markov generator and $T = mk$, the likelihood of $Y = (X_k, X_{2k}, \dots, X_T)$ is the product of the k-step predictions at starts $0, k, \dots, T - k$. The guarantee therefore applies to this subsampled likelihood objective.

6. **Auxiliary prediction heads and consistency.** A separate head $h_{\theta,k}(a \mid x_{\le t})$ can be trained directly on observed prefix–target pairs. This is auxiliary supervision, used in multi-token prediction. To represent the original generator's k-step marginal, the head must satisfy

$$h_{\theta,k}(a \mid x_{\le t}) = Q_\theta(X_{t+k} = a \mid x_{\le t}).$$

   Sharing parameters does not enforce this equality. A penalty can encourage agreement without enforcing it; fitting a head to a fixed teacher's predictions is a form of *distillation*. For a fixed generator, future-token samples from its rollouts provide training targets without enumerating all intervening sequences, at the cost of sampling error. Exact agreement remains a separate requirement, and accurate auxiliary heads alone do not guarantee accurate generation. See the bibliographical notes for examples and references.

7. **Learning support from samples.** The empirical data law gives infinite reverse KL to any generator that leaves the training set. Exact support recovery can itself require many demonstrations, even for two candidate rules; Appendix 4.D.3 gives the calculation.

8. **Distillation objectives and training data.** Forward KL is the conventional soft-target distillation objective: the student fits the teacher's output probabilities using cross-entropy. For language models, distinguish the source of prefixes from the source of response targets. Token-level distillation matches the teacher's next-token distribution at prefixes from a text or fine-tuning dataset. Sequence-level forward KL can be fitted by sampling responses from the teacher and maximizing their likelihood under the student. Sequence-level reverse KL instead uses student-generated responses and teacher probability evaluations.

   Practical methods modify these schemes. MiniLLM uses instruction-data prompts, student–teacher mixed generation, supervised fine-tuning for initialization, and an auxiliary language-modeling loss on a pretraining corpus. The teacher's original pretraining dataset is not required. See the bibliographical notes for sources.

9. **Which costs benefit from recovery?** For $c_{\text{any}}(x_{1:T}) = \mathbf{1}\{\exists t : x_t = 1\}$, $\mathbb{E}_{Q_r}[c_{\text{any}}] = 1 - e^{-\kappa}$ for every r. Correcting a mistake improves the final answer and limits the error count, while leaving unchanged whether a mistake ever occurred.

Bibliographical notes

The probability and KL chain rules continue Lecture 3. The cost bounds and binary calculations here follow directly from those identities. The zero-cost forward-KL refinement makes explicit information that a general Pinsker conversion discards.

Likelihood, task discrepancy, and imitation learning. Proposition 4.2.9 has Bernstein form, with a variance-dependent square-root term and a range-dependent linear term. Foster et al. [2024, Theorem 3.1] obtain this form using trajectory-level Hellinger distance and expert-action variance. Their lower-order term retains the total-return range R, up to logarithms. The technical gain in their Section 3.2 is to avoid an extra horizon factor: summing H advantages (changes in expected remaining return caused by an action) gives a naive range coefficient HR, which their concentration and stopping-time argument reduces to order R up to logarithms. Their Proposition 3.1 also bounds expert-action variance by the variance of the total return; a deterministic expert has zero action variance even in a random environment.

Here $\sigma_P(c)$ measures the variation of the specified trajectory cost directly. For our binary examples it is zero, and the sharp zero-cost bound already uses $C_{\max} = T$ for the count and $C_{\max} = 1$ for terminal error. Wang et al. [2025] survey variance-sensitive Hellinger bounds and the role of loss functions in decision making.

The finite-class likelihood proof in Appendix 4.D.1 makes the optimization tolerance explicit. Lecture 18 develops the imitation-learning results, including the expert variance, parameter sharing, misspecification, and the sample-budget meaning of horizon dependence.

The comparison-rule example in Appendix 4.D.3 adapts the rare-disagreement construction of György et al. [2025, Theorem 3.2] to conditional reverse KL. Its exact two-rule failure probability is proved here.

Minimax optimality of the parameter-sharing rate. The rate $d(T)/n$ in Proposition 4.4.1 is optimal up to constants depending on a. Write P_p for the length-T grouped Bernoulli law with parameter vector p. Then

$$\inf_{\widehat{Q}} \sup_{p \in [a,1-a]^{d(T)}} \mathbb{E}[D(P_p, \widehat{Q})] \asymp_a \frac{d(T)}{n},$$

where the infimum ranges over all learners that use n independent training sequences to output a probability law on length-T sequences. The upper bound follows from Proposition 4.4.1. For the lower bound, put independent two-point priors on the parameters, with the two values near $1/2$ separated by a small multiple of $(nm_j)^{-1/2}$, depending only on a. At a test position in group j, the training data and preceding test tokens contain at most $(n+1)m_j \le 2nm_j$ observations of that parameter. The two values remain hard to distinguish: Le Cam's testing bound and Pinsker's inequality give expected conditional KL at least a constant multiple of $1/(nm_j)$. Summing over the m_j positions per group and then over groups, using the KL chain rule, gives $d(T)/n$. Yu [1997] explains the two-point and coordinatewise testing methods underlying this argument.

Loss normalization in supervised fine-tuning. Yao et al. [2025, Section 4.3, Eq. (6)] distinguish averaging all predicted token losses from averaging separately normalized response losses, and compare these choices in supervised fine-tuning. Hugging Face [2024] explain how to implement token averaging across gradient-accumulation batches: sum the losses and divide by the total number of prediction targets included in the loss.

Statistical accuracy, optimization, and scaling laws. The strong-convexity certificate in Section 4.4.3 follows by minimizing the quadratic lower bound in the definition of strong convexity; Exercise 27 gives the calculation. Boyd and Vandenberghe [2004, Section 9.1.2] discuss this stopping criterion. Bottou et al. [2018, Theorem 4.7] give the $O(1/K)$ expected optimization rate used in Endnote 4, under smoothness, strong convexity, controlled gradient noise, and suitable decreasing stepsizes. Their Section 4.3 treats the weaker stationarity guarantees available for general nonconvex objectives.

Bottou and Bousquet [2007, Sections 2–3] study the approximation–estimation–optimization tradeoff. Balancing optimization tolerance against estimation error gives the statistical-accuracy principle used in Eq. (4.40). As data improve the statistical accuracy, finer optimization becomes worthwhile.

The connection to language-model scaling laws is explicit in Hoffmann et al. [2022, Section 3.3 and Appendix D.2]. Renaming their variables, they fit the empirical relation

$$L(m, B) \approx L_\infty + a\, m^{-\alpha} + b\, B^{-\beta},$$

where m is parameter count, B is the number of training tokens, and the constants are fitted across training runs. Their last term combines finite-data effects and incomplete optimization during a single pass. Increasing the training-token budget supplies both observations and gradient updates. Under a fixed compute budget, Kaplan et al. [2020] find a regime in which a larger model stopped before convergence outperforms smaller models trained longer. This allocates computation between model size and fitting accuracy. These empirical laws describe average log loss. The cost comparisons in Section 4.3 ask what task-performance variation remains at a given likelihood, and hence what additional information is useful for choosing a generator.

Direct cost gradients and finite training budgets. Osband [2026] compare exact expected-accuracy policy gradient with cross-entropy in full-information classification. If p is the probability assigned to the correct label and z is its log-odds, the two update weights on ∇z are $p(1-p)$ and $1-p$, respectively. They interpret cross-entropy as cumulative future error along a unit-speed increase in z, then truncate this integral at an estimate of the learning budget that remains. Their finite-horizon allocation guarantees concern a simplified model with independently movable example gaps; their neural-network results are empirical, and they do not establish the same conclusion for sampled-action reinforcement learning. Exercise 31 isolates the corresponding single-example gradient slowdown and compares it with natural gradient.

Multi-token prediction and auxiliary heads. Gloeckle et al. [2024, Section 2] train several future-token heads on a shared transformer representation and report improved downstream performance in their experiments. These are separate prediction heads; the architecture does not enforce equality with the autoregressive model's multistep marginals. The exact loss in Eq. (4.42) and this practical auxiliary objective therefore need to be distinguished. Hinton et al. [2015, Section 2] describe distillation by training a student on a teacher's predictive probabilities. Using generator rollouts to train a future-token head is a sampling-based version of this prediction-matching idea.

Stability and concentration. Gaubert and Qu [2015, Section 1] recall the Dobrushin ergodicity coefficient and its characterization as the optimal TV contraction factor. Rudolf and Schweizer [2018] develop perturbation bounds for Markov chains using contraction. Theorem 4.5.1 is the elementary finite-state TV argument. The history-dependent comparison in Appendix 4.F follows by replacing conditionals one at a time and coupling the resulting continuations. Doukhan and Wintenberger [2007, Section 3.1] use related summable Lipschitz coefficients to study processes with infinite memory. Raginsky and Sason [2014, Sections 2.2.1 and 3.4.3] give Hoeffding's lemma and the relative-entropy variational inequality. Our exponential-tilting calculation applies these ideas to a specified sequence cost. Proposition 4.B.1 identifies its exact reverse-KL conversion by constructing extremizing tilted laws. The single-bit comparison in Appendix 4.C.2 uses the sharp Bernoulli subgaussian proxy from Schlemm [2014].

Moduli of continuity and approximation. Section 4.2.5 applies the usual quantitative continuity question to KL and expected cost. The fixed-KL endpoint calculations give both conversion directions directly. For the approximation-theory connection in Appendix 4.C.3, Drucker and de Wolf [2011, Theorem 1] state Jackson's theorem in the form used here and give proofs. Its modulus controls the uniform polynomial approximation error as a function of degree.

Reverse-KL fitting. Hinton et al. [2015, Section 2] use cross-entropy with teacher probabilities, equivalent to forward KL, for model compression through distillation. Gu et al. [2024, Sections 1–2]

motivate reverse-KL distillation by prioritizing plausible student generations when the student cannot represent the full teacher distribution. Their MiniLLM algorithm uses policy gradients with practical stabilization; its reported benefits are empirical. Their Sections 2.2–3.1 describe the generation scheme and training data summarized in the distillation endnote. Nowozin et al. [2016, Sections 2.2–2.3] develop adversarial fitting through divergence lower bounds. Specializing their representation to reverse KL and writing their critic as $-e^g$ gives Eq. (4.48). Exercise 38 asks for a direct proof. Appendix 4.E derives the gradient for an evaluable target and the cost-defined Gibbs target.

Glossary

Sequence model; autoregressive factorization
A joint law on token sequences, expressed as a product of next-token conditionals. Every finite sequence law has such a factorization.

Maximum likelihood; teacher forcing
Maximum likelihood minimizes empirical negative log probability. Teacher forcing evaluates each conditional using the observed prefix. Generation instead feeds sampled tokens back into the predictor.

Positive-data fine-tuning
Continuing model fitting on examples selected to demonstrate desired behavior. The idealization here uses sequences of cost zero.

Cost and reward
A numerical assessment of a generated response or sequence. The task objective is its expected cost, or expected reward.

Cost discrepancy; moduli of continuity
The pseudometric $d_c(P, Q) = |\mathbb{E}_P[c] - \mathbb{E}_Q[c]|$ compares one expected cost. The modulus $\omega_{D\to c}$ asks how much cost discrepancy a KL budget permits. The converse $\omega_{c\to D}$ asks how large KL can be at a given cost-discrepancy budget. The upper and lower endpoints of the fixed-KL cost ranges determine these two moduli. Taking the supremum over reference laws gives a uniform conversion.

Hellinger distance (appendix)
A distributional distance used in the proofs of the variance-sensitive cost and finite-sample likelihood bounds; defined in Appendix 4.C.1.

Optimization error and tolerance
The error $E(\widehat{Q})$ is the model's excess empirical per-token loss above the infimum in the class. A tolerance η bounds this error and permits per-sequence excess $T\eta$. The average can place this excess unevenly across positions.

Total variation
$\|P - Q\|_{\mathrm{TV}}$ is the largest discrepancy in event probabilities. For costs in $[0, C_{\max}]$, it bounds expectation discrepancy multiplied by $C_{\max}$.

Forward and reverse KL
$D(P, Q)$ averages log density ratios under the data law; $D(Q, P)$ averages them under the model law. The sequence chain rules inherit these different prefix distributions.

Markov process; transition kernel
A process whose current state determines the conditional next-state law. $K(x, y)$ gives its transition probabilities.

Marginal law
The distribution of a selected part of a random sequence. The marginal at position t assigns probability $P(X_t = x)$ to x, summing over the other positions.

Dobrushin ergodicity coefficient; contraction margin
The coefficient $\rho(K)$ is the largest TV distance between two rows of K, and its optimal TV contraction factor. If a kernel contracts TV by at most $\rho < 1$, its margin is $1 - \rho$. The local-error bound is amplified by its reciprocal.

Recovery
In the binary example, the probability of returning from an erroneous state to zero. It controls

the duration of an erroneous run.

Subgaussian cost

A cost whose centered log exponential moment is at most $\lambda^2 v^2/2$. The quantity v^2 is a variance proxy; $v_P(c)$ is the smallest valid proxy under P. It satisfies $\sigma_P(c)^2 \le v_P(c) \le C_{\max}^2/4$ for $0 \le c \le C_{\max}$.

k-step prediction loss

$\mathcal{L}_k(Q)$ scores a token k positions ahead from the observed prefix, summing over the model's intervening tokens.

Critic; adversarial fitting

A critic scores sequences to maximize a divergence lower bound. Adversarial fitting alternates critic maximization with generator minimization; a restricted critic may leave a gap to the full divergence.

4.7 Exercises

These exercises follow the lecture from KL and task cost to sequence length, fitting, recovery, and persistent memory.

General instructions. Justify every answer, including yes/no answers, even when the question does not explicitly ask for a justification.

Exercise profiles. Filled mountains show difficulty on a four-mountain scale: one is a direct application, two uses a familiar method, three combines ideas, and four is a substantial challenge. Filled cogs show technical load on a three-cog scale: one is light, two is sustained, and three is dense. Filled hourglasses show estimated time: one is under 20 minutes, two is 20–45 minutes, three is 45–90 minutes, and four is more than 90 minutes. Pale pictograms show the unused part of each scale. The work row uses the canonical order proof, calculation, construction, computation, and interpretation; pale icons mark work types not used. ◎ marks the intended learning outcome. Time estimates cover the non-optional parts. Ratings are not prerequisites or grading weights. A superscript asterisk marks an exercise based on supplementary lecture material; it is not a difficulty rating and does not make the exercise optional when assigned.

Exercise 1: The KL chain rule.

◎ Derive KL chain rules for sequences and prompts and identify which prefix law weights each direction.

Let P, Q be laws on $[N]^T$, with $Q(x) > 0$ whenever $P(x) > 0$.

(a) Use the probability chain rule to write $\log(P(X_{1:T})/Q(X_{1:T}))$ as a sum.

(b) Take expectation under P, and then condition on $X_{<t}$ in each term, to prove the KL chain rule in Eq. (4.5).

(c) State the reversed identity, with its own support condition. Which prefix distribution appears in each direction?

(d) Let P, Q now be laws on $\mathcal{U} \times [N]^T$, where $\mathcal{U}$ is a finite prompt set, with the same prompt marginal μ. Prove

$$D(P, Q) = \sum_{u:\, \mu(u)>0} \mu(u) D(P(\cdot \mid u), Q(\cdot \mid u)).$$

Exercise 2: Zero-cost data and forward KL.

◎ Derive a sharp forward-KL bound on generated cost when the data cost is zero.

Let P, Q be laws on a finite set $\mathcal{X}$, and let $c : \mathcal{X} \to [0, C_{\max}]$, with $C_{\max} > 0$ and $\mathbb{E}_P[c] = 0$.

(a) Prove

$$\mathbb{E}_Q[c] \le C_{\max}(1 - e^{-D(P,Q)}) \le C_{\max} D(P, Q).$$

(b) Let $\kappa \ge 0$ and let R be supported on $\{x : c(x) = C_{\max}\}$. Show that $Q = e^{-\kappa} P + (1 - e^{-\kappa}) R$ attains the bound.

(c) Can a smaller bound hold for all such laws and costs using only $C_{\max}$ and $D(P, Q)$? Explain using part (b).

Hints For (a), take $A = \{x : c(x) = 0\}$ and compare $P(A)$ with $Q(A)$.

Exercise 3: Concentration and reverse KL.

Use exponential tilting to connect reverse KL with concentration and solve the minimum-reverse-KL problem under an expected-count constraint.

Use the notation of Section 4.2.3.

(a) Prove by exponential tilting that

$$\mathbb{E}_Q[c] - \mathbb{E}_P[c] \le \inf_{\lambda>0} \frac{D(Q,P) + \log \mathbb{E}_P[e^{\lambda(c-\mathbb{E}_P[c])}]}{\lambda}.$$

Then suppose $\log \mathbb{E}_P[e^{\lambda(c-\mathbb{E}_P[c])}] \le \lambda^2 v^2/2$ for every $\lambda \in \mathbb{R}$, and deduce

$$|\mathbb{E}_Q[c] - \mathbb{E}_P[c]| \le v\sqrt{2D(Q,P)}.$$

Include the case $v = 0$.

(b) Let $c = T^{-1}\sum_{t=1}^{T} f_t(X_t)$, with independent $[0,1]$-valued summands under P. Prove $v_P(c) \le 1/(4T)$, and deduce, for every law Q,

$$|\mathbb{E}_Q[c] - \mathbb{E}_P[c]| \le \sqrt{\frac{D(Q,P)}{2T}}. \tag{4.57}$$

What growth of reverse KL still permits vanishing discrepancy?

(c) Let $P_\delta = \text{Bernoulli}(\delta)^{\otimes T}$, $0 < \delta < 1$, and set $m = \mathbb{E}_Q[c_{\text{cnt}}]/T$ for the error-count cost. Prove

$$D(Q, P_\delta) = T\ \text{kl}(m,\delta) + D(Q, P_m),$$

where $P_m = \text{Bernoulli}(m)^{\otimes T}$ and $\text{kl}(a,b) = a\log(a/b) + (1-a)\log((1-a)/(1-b))$. Treat $m = 0, 1$ as well.

(d) Which generator has the smallest reverse KL among laws with a specified expected count?

Hints For (a), normalize $P(x)e^{\lambda(c(x)-\mathbb{E}_P[c])}$ to obtain a probability law, then use nonnegativity of KL. For (b), apply Hoeffding's lemma to each independent summand.

Task discrepancy conversions

Exercise 4: Task discrepancy as a pseudometric.*

Prove that a cost-induced discrepancy is a pseudometric and show why it need not distinguish laws.

For probability laws P, Q on $\mathcal{X}$ and a cost $c : \mathcal{X} \to \mathbb{R}$, define

$$d_c(P,Q) = |\mathbb{E}_P[c] - \mathbb{E}_Q[c]|.$$

Prove that d_c is a pseudometric. Give distinct laws with $d_c(P,Q) = 0$, first for the binary count and then for another cost.

Exercise 5: Forward discrepancy conversion.*

Compute sharp forward-KL-to-cost moduli and identify the laws attaining them.

Use the definitions in Section 4.2.5. Let $P = \delta_{0^T}$, $T \ge 2$, with $c_{\text{cnt}}(x) = \sum_{t=1}^{T} x_t$ and $c_{\text{fin}}(x) = x_T$. Use the fixed-KL intervals in Eqs. (4.24) and (4.25) to prove

$$\omega_{D\to c_{\text{cnt}}}(\kappa; P) = T(1 - e^{-\kappa}), \tag{4.58}$$

$$\omega_{D\to c_{\text{fin}}}(\kappa; P) = 1 - e^{-\kappa}, \qquad \kappa \ge 0. \tag{4.59}$$

Which of Q_- and Q_+ attains these bounds?

Exercise 6: Reverse discrepancy conversion.*

Determine when small task cost controls forward KL and derive the sharp fixed-cost interval.

Use the definitions in Section 4.2.5. Let $P = \delta_{0^T}$, $T \geq 2$, with $c_{\mathrm{cnt}}(x) = \sum_{t=1}^{T} x_t$ and $c_{\mathrm{fin}}(x) = x_T$.

(a) Use the fixed-KL intervals in Eqs. (4.24) and (4.25) to calculate the reverse conversions:

$$\omega_{c_{\mathrm{cnt}} \to D}(s; P) = \begin{cases} -\log(1-s), & 0 \leq s < 1, \\ +\infty, & s \geq 1, \end{cases} \tag{4.60}$$

$$\omega_{c_{\mathrm{fin}} \to D}(s; P) = +\infty, \qquad s \geq 0. \tag{4.61}$$

Exhibit laws attaining each finite bound and laws with infinite KL where appropriate. Explain why small expected count forces small KL, whereas perfect final answers allow arbitrarily large KL.

(b) Fix an expected count $s \in [0, 1)$. Prove the sharp bounds

$$-\log(1 - s/T) \leq D(P, Q) \leq -\log(1 - s).$$

Give a law attaining each endpoint and relate them to part (a).

Exercise 7: Normalization and discrepancy scale.*

Determine how cost normalization changes conversion moduli without changing their extremal ratio.

Let $P = \delta_{0^T}$, $T \geq 2$, with $c_{\mathrm{cnt}}(x) = \sum_{t=1}^{T} x_t$. Replace the count by the average count c_{cnt}/T. Show that

$$\omega_{D \to c_{\mathrm{cnt}}/T}(\kappa; P) = 1 - e^{-\kappa}$$

and

$$\omega_{c_{\mathrm{cnt}}/T \to D}(s; P) = \begin{cases} -\log(1 - Ts), & 0 \leq s < 1/T, \\ +\infty, & s \geq 1/T. \end{cases}$$

Show that normalization divides both fixed-KL cost endpoints by T and therefore leaves their ratio equal to T for every $\kappa > 0$.

Exercise 8: The variance bound at zero data variance.*

Construct a zero-data-variance example and identify the term that controls positive generated cost.

Construct probability laws P, Q and a cost $c : \mathcal{X} \to [0, C_{\max}]$ such that $\mathrm{Var}_P(c) = 0$ but $|\mathbb{E}_P[c] - \mathbb{E}_Q[c]| > 0$. In the bound

$$|\mathbb{E}_P[c] - \mathbb{E}_Q[c]| \leq 2\sqrt{\mathrm{Var}_P(c) D(P, Q)} + C_{\max} D(P, Q),$$

identify the term that accounts for the positive discrepancy.

Repeated and persistent errors

Exercise 9: Repeated errors and sequence KL.

Relate constant positionwise log loss to sequence KL and the distribution of the first error.

Compare isolated errors with errors that persist. Fix $T \geq 2$, $P = \delta_{0^T}$, $0 < \varepsilon < 1$, and $X_0 = 0$. Use $c_{\mathrm{cnt}}(x) = \sum_{t=1}^{T} x_t$ and $c_{\mathrm{fin}}(x) = x_T$. The Markov generators $Q_{\mathrm{ind}}, Q_{\mathrm{bad}}$ use

	$Q(X_{t+1}=1 \mid X_t=0)$	$Q(X_{t+1}=1 \mid X_t=1)$
Q_{ind}	ε	ε
Q_{bad}	ε	1

$(0 \leq t < T)$.

For P, use the kernel that returns zero from either state. Show that both generators have conditional KL $-\log(1-\varepsilon)$ at every data prefix, and hence

$$D(P, Q_{\mathrm{ind}}) = D(P, Q_{\mathrm{bad}}) = T \log \frac{1}{1-\varepsilon}. \tag{4.62}$$

Compare this constant positionwise loss with the loss profile of Q_- and Q_+. Compute the probability that the first one occurs at position s, and the probability of no ones.

Exercise 10: Persistent errors accumulate.

Derive exact and asymptotic error counts and explain the quadratic accumulation caused by persistent errors.

Use the setting and Markov generators from Exercise 9.

(a) Prove

$$\mathbb{E}_{Q_{\mathrm{ind}}}[c_{\mathrm{cnt}}] = T\varepsilon, \tag{4.63}$$

$$\mathbb{E}_{Q_{\mathrm{bad}}}[c_{\mathrm{cnt}}] = \sum_{t=1}^{T}[1-(1-\varepsilon)^t] = T - \frac{(1-\varepsilon)[1-(1-\varepsilon)^T]}{\varepsilon}. \tag{4.64}$$

Obtain the second identity both from marginal error probabilities and by conditioning on the first error position. Also prove

$$\mathbb{E}_{Q_{\mathrm{ind}}}[c_{\mathrm{fin}}] = \varepsilon, \qquad \mathbb{E}_{Q_{\mathrm{bad}}}[c_{\mathrm{fin}}] = 1-(1-\varepsilon)^T. \tag{4.65}$$

(b) Prove the finite bounds

$$\frac{\varepsilon T(T+1)}{2} - \frac{\varepsilon^2 T(T+1)(T-1)}{6} \leq \mathbb{E}_{Q_{\mathrm{bad}}}[c_{\mathrm{cnt}}] \leq \frac{\varepsilon T(T+1)}{2}.$$

Deduce $\mathbb{E}_{Q_{\mathrm{bad}}}[c_{\mathrm{cnt}}] \sim \varepsilon T(T+1)/2$ when $T\varepsilon \to 0$. Explain why the expected count can grow quadratically with T in this regime.

Exercise 11: What total-variation bounds discard.

Compare exact accumulated costs with total-variation and Pinsker bounds and identify the missing duration information.

Use the setting and Markov generators from Exercise 9. Show that both sequence TVs equal $1-(1-\varepsilon)^T$. For either generator, verify the bounds

$$\mathbb{E}_Q[c_{\mathrm{cnt}}] \leq T[1-(1-\varepsilon)^T], \qquad \text{using exact TV,} \tag{4.66}$$

$$\mathbb{E}_Q[c_{\mathrm{cnt}}] \leq T \min\left\{1, \sqrt{\frac{T}{2}\log\frac{1}{1-\varepsilon}}\right\}, \qquad \text{using Pinsker.} \tag{4.67}$$

Compare these with the exact counts at $T = 1000, \varepsilon = 10^{-3}$, and their leading terms when $T\varepsilon \to 0$. Which information about error duration does the TV-to-count bound discard?

Exercise 12: Terminal cost at fixed KL.

Show that models with equal sequence KL can have terminal costs differing by a factor of order T.

Use the setting and Markov generators from Exercise 9. Set $\varepsilon = 1 - e^{-\kappa/T}$ for fixed $\kappa > 0$. Compute the two terminal costs and the ratio $\mathbb{E}_{Q_{\mathrm{bad}}}[c_{\mathrm{fin}}]/\mathbb{E}_{Q_{\mathrm{ind}}}[c_{\mathrm{fin}}]$. Explain why this ratio is of order T even though the terminal cost range is fixed. Which model attains the terminal-cost upper endpoint?

Exercise 13: Positionwise and average KL bounds.

Distinguish uniform positionwise KL control from average sequence-KL control by deriving their consequences for accumulated and terminal costs.

Let $P = \delta_{0^T}$, $T \geq 2$, and use the counting and terminal costs from Exercise 9. Compare a bound on each next-token KL with a bound on their average. Fix $\eta \geq 0$.

(a) Suppose a generator Q satisfies

$$\mathbb{E}\big[D\big(P(\cdot \mid X_{<t}), Q(\cdot \mid X_{<t})\big)\big] \leq \eta \quad (t = 1, \dots, T), \qquad X_{1:T} \sim P.$$

Show that $Q(X_{1:t} = 0^t) \geq e^{-t\eta}$, and deduce

$$\mathbb{E}_Q[c_{\mathrm{cnt}}] \leq \sum_{t=1}^{T} (1 - e^{-t\eta}), \qquad \mathbb{E}_Q[c_{\mathrm{fin}}] \leq 1 - e^{-T\eta}. \tag{4.68}$$

Show that Q_{bad} attains both bounds when $\varepsilon = 1 - e^{-\eta}$.

(b) Suppose instead that only $D(P, Q)/T \leq \eta$. Use Eqs. (4.24) and (4.25) to obtain the corresponding bounds. Compare their small-$T\eta$ leading terms with Eq. (4.68). What per-token accuracy suffices to guarantee terminal error at most $s \in (0, 1)$?

Parameter sharing

Exercise 14: Estimating shared parameters.*

Show why an unsmoothed empirical Bernoulli estimate has infinite expected KL and prove a finite-sample bound for add-one smoothing.

Use the independent-bit models and pooled estimator from Section 4.4.1. All expectations below average over the training observations. Fix $0 < a < 1/2$. Let $B \sim \mathrm{Binomial}(L, p)$, with $L \geq 1$. For $0 < p < 1$, show that the ordinary empirical frequency $\bar{p} = B/L$ satisfies $\mathbb{E}[\mathrm{kl}(p, \bar{p})] = \infty$. Explain why adding one count of each bit gives $\widehat{p} = (B+1)/(L+2)$, and why this assigns positive probability to both bits. To obtain a finite-sample bound for this choice, prove that, for $0 < p < 1$,

$$\mathbb{E}\left[\frac{1}{B+1}\right] = \frac{1 - (1-p)^{L+1}}{(L+1)p}.$$

Apply the same identity to $L - B$, then use Jensen's inequality for the logarithm to show

$$\mathbb{E}[\mathrm{kl}(p, \widehat{p})] \leq \log \frac{L+2}{L+1} \leq \frac{1}{L+1}.$$

Verify the bound directly for $p = 0, 1$.

Hints Use $1/(B+1) = \int_0^1 u^B \, du$.

Exercise 15: Finite-sample benefit of parameter sharing.*

Prove matching-order finite-sample sequence-KL bounds and explain why group size cancels under pooled estimation.

Use the independent-bit models and pooled estimator from Section 4.4.1. All expectations below average over the training observations. Fix $0 < a < 1/2$. Apply Exercise 14 with $L = nm_j$ to bound $\kappa_n(T)$ from above. For the lower bound, calculate

$$\mathbb{E}[(\widehat{p} - p)^2] = \frac{Lp(1-p) + (1-2p)^2}{(L+2)^2}.$$

Use Pinsker's inequality and $L/(L+2) \geq 1/3$ to show that

$$\frac{2a(1-a)}{9}\frac{d(T)}{n} \leq \kappa_n(T) \leq \frac{d(T)}{n} \qquad \text{when } p_j \in [a, 1-a].$$

Explain why group sizes cancel from both bounds.

Exercise 16: The asymptotic constant for pooled estimation.*

Derive the uniform local expansion that gives the asymptotically sharp $d(T)/(2n)$ sequence-KL rate.

Use the independent-bit models and pooled estimator from Section 4.4.1. All expectations below average over the training observations. Fix $0 < a < 1/2$. For fixed $p \in (0,1)$, prove $L\,\mathbb{E}[\mathrm{kl}(p, \widehat{p})] \to 1/2$ as $L \to \infty$. Deduce Eqs. (4.28) and (4.29), keeping track of which of n, T is fixed in each limit. Make the expansion uniform for $p \in [a, 1-a]$ and prove Eq. (4.32) when $n \min_j m_j \to \infty$.

HINTS Expand $\mathrm{kl}(p, q)$ near $q = p$ and use a binomial tail bound for the remaining event.

Sequence length and token budgets

Exercise 17: Sequence length and the training budget.*

Compare shared, unshared, and intermediate parameter-growth patterns as sequence length increases at fixed sample count.

Use the grouped Bernoulli model and smoothed estimator from Section 4.4.1. Assume $p_j \in [a, 1-a]$, with $0 < a < 1/2$.

(a) Keep n fixed and suppose $p_t \in [a, 1-a]$. Use Exercise 15 to give explicit upper and lower bounds proportional to T/n for the unshared model's expected sequence KL. Compare this growth with the shared model's limit in Eq. (4.28).

(b) Use one new parameter for each successive block of lengths $1, 2, 3, \ldots$. Compute $d(T)$ at complete-block endpoints and interpret Eq. (4.31). Use the explicit remainder derived in Exercise 16 to determine the leading term even though the smallest group has size one. How can normalized KL improve while sequence KL grows?

Exercise 18: Sequence length at a fixed token budget.*

Determine how allocating a fixed training-token budget across longer sequences changes sequence KL and bounded-cost guarantees.

Use the grouped Bernoulli model and smoothed estimator from Section 4.4.1. Assume $p_j \in [a, 1-a]$, with $0 < a < 1/2$. Fix the total training-token budget $L = nT$ and use $\kappa'_L(T) = \kappa_{L/T}(T)$ to repeat the shared and unshared leading-order comparisons. Explain why increasing T worsens expected sequence KL in the shared model. Compare the numbers of observations per parameter in the two models. Translate the expected-KL bounds into bounds on the discrepancy of a cost in $[0, 1]$.

Exercise 19: Task costs at a given learning rate.*

Construct estimators with the same expected sequence-KL rate whose attainable task costs fill sharply different intervals.

Use the following construction to prove the stated KL rate and cost intervals. At each horizon T, partition the first T positions into $\lceil d(T) \rceil$ nonempty groups. Let the U_t be independent fair bits and $V_t = 0$ under P. Use terminal cost V_T and counting cost $\sum_{t=1}^{T} V_t$. Write P_T and P_T^U for the laws of $(U, V)_{1:T}$ and $U_{1:T}$, respectively. If $\widehat{P}_T^U$ is the pooled smoothed estimate of the law of $U_{1:T}$, set

$$r_T = \mathbb{E}\Big[D(P_T^U, \widehat{P}_T^U)\Big]$$

and, for any law R supported on $v_{1:T} \neq 0^T$, define

$$\widehat{Q}_R = e^{-r_T}(\widehat{P}_T^U \otimes \delta_{0^T}) + (1 - e^{-r_T})R.$$

(a) Show that r_T is determined by n, T and the chosen partition, so the mixture weight can be used by the estimator.

(b) Compute $\mathbb{E}[D(P_T, \widehat{Q}_R)]$ and show that it has order $d(T)/n$, independently of R.

(c) Writing $\kappa_n(T) = \mathbb{E}[D(P_T, \widehat{Q}_R)]$, show that the full attainable intervals for the expected terminal and counting costs as R varies are

$$[0, 1 - e^{-\kappa_n(T)/2}] \quad \text{and} \quad [1 - e^{-\kappa_n(T)/2}, T(1 - e^{-\kappa_n(T)/2})],$$

respectively. Identify laws attaining the endpoints.

Exercise 20: Identifying one of two Bernoulli models.*

Use an exponential error bound to show that two separated Bernoulli sequence models can be identified exactly even when the number of training sequences is fixed.

The data law is one of two infinite i.i.d. Bernoulli processes, with parameter $p \in \{1/4, 3/4\}$. Observe n independent length-T sequences. Choose $\widehat{p} = 3/4$ if the empirical mean of their nT bits is at least $1/2$, and choose $1/4$ otherwise. Let $\widehat{Q}$ be the i.i.d. model with parameter $\widehat{p}$.

(a) Prove that the probability of choosing the wrong parameter is at most $(\sqrt{3}/2)^{nT} \leq e^{-nT/8}$.

(b) Compute the sequence KL when the selected parameter is wrong, and deduce

$$\kappa_n(T) \leq \frac{T \log 3}{2} e^{-nT/8} \longrightarrow 0 \qquad (T \to \infty,\ n \text{ fixed}).$$

(c) Compare this with estimating an arbitrary $p \in (0, 1)$ in Eq. (4.28). What allows the two-candidate learner to identify the entire model exactly?

HINTS For (a), let B be the number of observed ones and apply Markov's inequality to $3^{B - nT/2}$ when $p = 1/4$.

Copying and memory

Exercise 21: Copying with a bounded context.*

Derive the irreducible log loss of bounded-context prediction on delayed copies and construct an optimal predictor.

Let P generate $B_1B_1B_2B_2\cdots$, where the blocks B_m are independent and uniform on $\{0,1\}^m$. For an integer $w \geq 0$, let $\mathcal{Q}_w$ be all infinite-sequence models whose next-token distribution can depend on the position and the last w tokens only.

(a) In a copied block of length $m > w$, show that the required bit is fair given the available context, although the full prefix determines it. Deduce expected conditional KL at least $\log 2$ at each such position.

(b) At $T = M(M+1)$, for $M > w$, prove

$$\inf_{Q \in \mathcal{Q}_w} D(P_{1:T}, Q_{1:T}) = \frac{\log 2}{2}\big(T - w(w+1)\big). \tag{4.69}$$

Specify a predictor attaining equality.

Exercise 22: Memory required for exact copying.*

Distinguish the short description of a copying rule from the growing context or state information required to execute it.

Use the copying process and bounded-context models from Exercise 21.

(a) Explain why more training data cannot remove this error for a fixed w. How must w grow to predict every copied bit exactly?

(b) Show that an exact recurrent generator must distinguish 2^m states after generating a fresh length-m block, and hence needs at least m bits of state information. Compare the description length of the copying rule with its memory requirement. Relate the distinction to recalling earlier definitions or intermediate results in a mathematical argument.

A missed final-position dependency

Exercise 23: A missed dependency at the final position.*

Show that sequence KL divided by sequence length can vanish while final-answer error stays fixed.

Let $T \geq 2$, let $\mathcal{X} = \{0,1\}^T$, and let P_{par} and Q_{coin} be probability mass functions on $\mathcal{X}$. Under P_{par}, the first $T-1$ bits are independent fair coins and the last is their parity: $X_T = (\sum_{t<T} X_t) \bmod 2$. Under Q_{coin}, all T bits are independent fair coins. Define the task cost $c_{\text{par}} : \mathcal{X} \to \{0,1\}$ by

$$c_{\text{par}}(x_{1:T}) = \mathbb{I}\left\{x_T \neq \left(\sum_{t<T} x_t\right) \bmod 2\right\}.$$

(a) Show that for every $t < T$ and every $x_{<t} \in \{0,1\}^{t-1}$,

$$P_{\text{par}}(\cdot \mid x_{<t}) = Q_{\text{coin}}(\cdot \mid x_{<t}) = \text{Bernoulli}(1/2).$$

Show also that for every $x_{<T} \in \{0,1\}^{T-1}$,

$$P_{\text{par}}(\cdot \mid x_{<T}) = \delta_{\text{par}(x_{<T})}, \qquad Q_{\text{coin}}(\cdot \mid x_{<T}) = \text{Bernoulli}(1/2),$$

where $\text{par}(x_{<T}) = (\sum_{t<T} x_t) \bmod 2$.

(b) Use the KL chain rule to show that

$$D(P_{\text{par}}, Q_{\text{coin}}) = \log 2, \qquad \frac{D(P_{\text{par}}, Q_{\text{coin}})}{T} = \frac{\log 2}{T}.$$

(c) Derive

$$\mathbb{E}_{P_{\mathrm{par}}}[c_{\mathrm{par}}] = 0, \qquad \mathbb{E}_{Q_{\mathrm{coin}}}[c_{\mathrm{par}}] = \frac{1}{2},$$

and verify the equality

$$\mathbb{E}_{Q_{\mathrm{coin}}}[c_{\mathrm{par}}] = 1 - e^{-D(P_{\mathrm{par}}, Q_{\mathrm{coin}})}.$$

(d) For this part, let $\mathcal{X}$ be any finite alphabet, and let P, Q be probability distributions on $\mathcal{X}^+ = \cup_{t \geq 1} \mathcal{X}^t$, the set of finite sequences taking values in $\mathcal{X}$. Let $c : \mathcal{X}^+ \to [0, 1]$. Assume that $\mathbb{E}_P[c] = 0$. For $x \in \mathcal{X}^+$, let $|x|$ denote its length. Assume that $P(\mathcal{X}^T) > 0$ and $Q(\mathcal{X}^T) > 0$ for every $T \geq 1$. For $x \in \mathcal{X}^T$, define the conditional probability mass functions

$$P_T(x) = \frac{P(x)}{P(\mathcal{X}^T)}, \qquad Q_T(x) = \frac{Q(x)}{Q(\mathcal{X}^T)}.$$

Show that $\mathbb{E}_{P_T}[c] = 0$ for every $T \geq 1$. Prove or disprove the following statement: If $D(P_T, Q_T)/T \to 0$ as $T \to \infty$ then $\lim_{T \to \infty} \mathbb{E}_{Q_T}[c] = 0$. That is, if the average KL between P_T and Q_T vanishes, the cost under Q_T vanishes.

Exercise 24: Fitting parity with independent-bit models.*

Compare fitting the parity law directly with fitting it using independent-bit models.

Let $T \geq 2$, let $\mathcal{X} = \{0, 1\}^T$, and define

$$\mathrm{par}(x_{<T}) = \left(\sum_{t=1}^{T-1} x_t \right) \bmod 2, \qquad x_{<T} \in \{0, 1\}^{T-1}.$$

For $t \in [T]$, let $X_t : \mathcal{X} \to \{0, 1\}$ be the t-th coordinate map, and write $X_{<T} = (X_1, \ldots, X_{T-1})$. Let P_{par} be the probability mass function on $\mathcal{X}$ under which the first $T - 1$ bits are independent fair coins and the last bit is their parity. Let Q_{coin} be the probability mass function on $\mathcal{X}$ under which all T bits are independent fair coins. Fix $n \geq 1$ and points $x^{(1)}, \ldots, x^{(n)} \in \mathcal{X}$ satisfying $P_{\mathrm{par}}(x^{(i)}) > 0$ for every $i \in [n]$. For any probability mass function Q on $\mathcal{X}$, define its empirical log loss per token on these points by

$$\widehat{L}_n(Q) = -\frac{1}{nT} \sum_{i=1}^{n} \log Q(x^{(i)}).$$

(a) Show that

$$\widehat{L}_n(P_{\mathrm{par}}) = \frac{T-1}{T} \log 2, \qquad \widehat{L}_n(Q_{\mathrm{coin}}) = \log 2, \qquad \widehat{L}_n(Q_{\mathrm{coin}}) - \widehat{L}_n(P_{\mathrm{par}}) = \frac{\log 2}{T}.$$

Verify that for every $x_{<T} \in \{0, 1\}^{T-1}$,

$$P_{\mathrm{par}}(X_T = \mathrm{par}(x_{<T}) \mid X_{<T} = x_{<T}) = 1, \qquad Q_{\mathrm{coin}}(X_T = \mathrm{par}(x_{<T}) \mid X_{<T} = x_{<T}) = \frac{1}{2}.$$

(b) Let $\mathcal{Q}_{\mathrm{ind}}$ be the class of independent-bit laws

$$Q_q = \bigotimes_{t=1}^{T} \mathrm{Bernoulli}(q_t), \qquad q = (q_1, \ldots, q_T) \in [0, 1]^T.$$

First verify that $Q_{\mathrm{coin}} = Q_{(1/2, \ldots, 1/2)}$ belongs to $\mathcal{Q}_{\mathrm{ind}}$. Then derive

$$\mathbb{E}_{P_{\mathrm{par}}}[-\log Q_q] = \sum_{t=1}^{T} \left[-\frac{1}{2} \log q_t - \frac{1}{2} \log(1 - q_t) \right]$$

and show that the unique minimizer of the population sequence log loss over $\mathcal{Q}_{\mathrm{ind}}$ is Q_{coin}.

Interpretation. Part (a) shows that one completely missed final-position dependency contributes only $(\log 2)/T$ to average per-token log loss, although the independent model assigns only probability $1/2$ to the correct final bit. Part (b) identifies the role of model misspecification: the best independent-bit law matches every one-coordinate marginal and is Q_{coin}, but no law in this class represents the parity dependency. Thus optimal fitting within a factorized class need not recover a sequence-level dependency.

Exercise 25: Concentrating KL at the final position.*

◎ Show that an average KL bound permits the entire discrepancy to occur at one position.

If $D(P,Q)/T \le \eta$, what is the largest possible final-position contribution

$$\mathbb{E}\big[D\big(P(\cdot \mid X_{<T}), Q(\cdot \mid X_{<T})\big)\big], \qquad X_{1:T} \sim P,$$

in Eq. (4.5)? Give an example attaining it. Compare the information supplied by the average and by the individual positionwise losses.

Exercise 26: Parity as a Markov state.*

◎ Represent parity as a Markov state and locate the final error in the stability bound.

Use the parity law P_{par}, independent fair-coin law Q_{coin}, and cost c_{par} from Exercise 23. Define the running parity $Y_t = (\sum_{j=1}^{t} X_j) \bmod 2$. Describe its Markov kernels under P_{par} and Q_{coin}. Apply Theorem 4.5.1 to this state, for which $c_{\mathrm{par}} = Y_T$. Show that the fitted kernels have contraction factor zero, all earlier local discrepancies vanish, and the final local discrepancy is $1/2$. Why does contraction leave this final error unchanged?

Exercise 27: Certifying an optimization tolerance.*

◎ Use strong convexity to certify an optimization gap from the gradient norm and explain why stationarity alone is insufficient.

Let $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable, with $f_\star = \inf_{v \in \mathbb{R}^d} f(v) > -\infty$. For $f(\theta) = \widehat{L}(Q_\theta)$ and $\mathcal{Q} = \{Q_\theta : \theta \in \mathbb{R}^d\}$, the gap $f(\theta) - f_\star$ is the optimization error in Eq. (4.35).

(a) Suppose f is *μ-strongly convex*, meaning that for a known $\mu > 0$ and every $\theta, v \in \mathbb{R}^d$,

$$f(v) \ge f(\theta) + \nabla f(\theta)^\top (v - \theta) + \frac{\mu}{2}\|v - \theta\|_2^2.$$

Prove

$$f(\theta) - f_\star \le \frac{\|\nabla f(\theta)\|_2^2}{2\mu}.$$

For a desired per-sequence tolerance $s > 0$, give a sufficient gradient-norm threshold that certifies $T\big(f(\theta) - f_\star\big) \le s$.

(b) Take $d = 1$ and $f(\theta) = (\theta^2 - 1)^2$. Compute $f'(0)$ and $f(0) - f_\star$. Explain why a small gradient alone does not certify a small optimization error for a general loss.

Hints For (a), minimize the quadratic lower bound over $v - \theta$.

Likelihood fitting, task cost, and support

Exercise 28: Approximate likelihood fits and task cost.*

Quantify how a small likelihood optimization gap can permit different terminal and accumulated task costs.

Use the repeated-error generators from Exercise 9 and all-zero training sequences.

(a) Compute both empirical per-token losses. Include $P = \delta_{0^T}$ in the model class and show that $\varepsilon = 1 - e^{-\eta}$ makes both generators acceptable at optimization tolerance $\eta > 0$.

(b) Compute their terminal and counting costs and their leading terms as $T\eta \to 0$. Give a sufficient order of η in T to keep each expected count bounded.

Exercise 29: Exact fitting and finite-class guarantees.*

Separate empirical minimization from task quality and track how finite-class guarantees scale with sequence length and token budget.

Use the repeated-error generators from Exercise 9 and all-zero training sequences.

(a) Remove P and use a finite collection of paired generators at distinct positive values of ε. Identify all empirical minimizers. Explain why exact fitting leaves a tie between models of different task cost.

(b) Specialize Theorem 4.4.4 to zero-cost data, first for $c \in [0, T]$ and then for $c \in [0, 1]$. Express each guarantee using the number of sequences n, then using the token budget $L = nT$.

Exercise 30: Chosen prompts and conditional support.*

Compare one deliberately chosen distinguishing prompt with the random demonstrations needed to identify the correct rule.

In Proposition 4.D.1, choose a prompt on which the two candidate rules disagree. Show that its expert label identifies the correct rule. Compare the number of such queries with the number of random demonstrations needed for failure probability at most $\delta \in (0, 1/2)$.

Exercise 31: Direct cost, log loss, and natural gradient.

Compare three optimization dynamics and identify why an exact smooth task-cost gradient can provide a poor training signal.

Consider a one-step binary policy. The correct action is 1, and the policy samples $A \in \{0, 1\}$ with

$$\pi_z(1) = p(z) := \sigma(z) = \frac{1}{1 + e^{-z}}, \qquad \pi_z(0) = 1 - p(z).$$

The task cost is $c(A) = \mathbb{I}\{A = 0\}$, so its expectation is $C(z) = \mathbb{E}[c(A)] = 1 - p(z)$. The correct action is known, so every gradient below is evaluated exactly. Define the log-loss surrogate

$$L(z) = -\log p(z).$$

For $a > 0$, initialize $z(0) = -a$. For a differentiable loss $f(z)$, its *gradient flow* is the continuous-time rule

$$\dot{z}(t) = -f'(z(t)).$$

For this policy, define the scalar Fisher information by

$$F(z) = \mathbb{E}_{A \sim \pi_z}\left[\left(\frac{\partial}{\partial z} \log \pi_z(A)\right)^2\right].$$

For a differentiable expected reward $J(z)$, its *natural-gradient flow* for maximizing J is

$$\dot{z}_{\mathrm{NG}}(t) = F(z_{\mathrm{NG}}(t))^{-1} J'(z_{\mathrm{NG}}(t)).$$

(a) Use this definition to derive the gradient flows for minimizing C and L:

$$\dot{z}_C = p(z_C)(1 - p(z_C)), \qquad \dot{z}_L = 1 - p(z_L).$$

(b) Let $\tau_C(a)$ and $\tau_L(a)$ be the first times at which the corresponding flow reaches $z = 0$. Show that

$$\tau_C(a) = e^a - e^{-a} + 2a, \qquad \tau_L(a) = a + 1 - e^{-a}.$$

Deduce that $\tau_C(a) \sim e^a$ and $\tau_L(a) \sim a$ as $a \to \infty$.

(c) Show that the Fisher information of the Bernoulli policy is

$$F(z) = p(z)(1 - p(z)).$$

For the expected reward $J(z) = p(z)$, show that its natural-gradient flow is $\dot{z}_{\mathrm{NG}}(t) = 1$, and hence that the flow reaches $z = 0$ at time a.

(d) Explain why this example is an optimization failure rather than a nonsmooth-loss, exploration, delayed-credit, or gradient-estimation failure. Your explanation must also state why multiplying the direct-cost gradient by one fixed positive constant cannot remove its exponential dependence on a, whereas log loss and natural gradient use state-dependent rescaling.

HINTS For (b), if $\dot{z} = g(z) > 0$, separate variables to express the hitting time as an integral of $1/g(z)$.

Exercise 32: Two-step prediction and recovery.

Compute how a two-step loss distinguishes generators tied under next-token loss and selects the lower-cost generator.

Use the losses in Eq. (4.42).

(a) For Q_-, Q_+ from Section 4.3.3, with $T \geq 2$ and $\kappa > 0$, compute $\mathcal{L}_1$ and $\mathcal{L}_2$ on 0^T.

(b) Which model minimizes $\mathcal{L}_1 + \lambda \mathcal{L}_2$ over this pair when $\lambda > 0$, and by how much is its loss smaller? Compare the models' expected terminal costs $\mathbb{E}_Q[c_{\mathrm{fin}}]$.

Exact and auxiliary multistep prediction

Exercise 33: Computing further-ahead predictions.

Derive the exponential marginalization cost and use Markov structure to replace it with a dynamic program.

Fix an autoregressive model Q_θ, a prefix $x_{\leq t}$, $k \geq 2$ with $t + k \leq T$, and $a \in [N]$.

(a) Derive

$$Q_\theta(X_{t+k} = a \mid X_{\leq t} = x_{\leq t}) = \sum_{y_{1:k-1} \in [N]^{k-1}} \left[\prod_{j=1}^{k-1} Q_\theta(y_j \mid x_{\leq t}, y_{<j}) \right] Q_\theta(a \mid x_{\leq t}, y_{1:k-1}).$$

Count its terms and the number of conditional evaluations if each term is evaluated separately. Explain why sharing prefixes still leaves exponentially many distinct prefixes in a general model.

(b) Show that supplying the observed intervening tokens trains

$$Q_\theta(X_{t+k} = x_{t+k} \mid X_{\le t+k-1} = x_{\le t+k-1})$$

instead of the marginalized conditional probability in part (a).

(c) Suppose $t \ge 1$ and the model is first-order Markov on $[N]$ with known transition matrix K. Give a recursion that computes the entire k-step distribution in $O(kN^2)$ arithmetic operations. Why can this recursion merge histories?

Exercise 34: Consistency of a future-token head.

◎ State the consistency condition for an auxiliary future-token head and construct a counterexample to consistency from data fit alone.

Fix a prefix $x_{\le t}$ and an integer $k \ge 2$. A separate head $h_{\theta,k}(\cdot \mid x_{\le t})$ can be trained from pairs $(X_{\le t}, X_{t+k})$. State the additional equality needed for its probabilities to be the original sequence model's k-step marginals. Does perfect prediction on the observed prefix–target pairs imply this equality? Give an example or a proof.

Errors detected by the two KL directions

Exercise 35: Added failures and forward KL.

◎ Show that small forward KL can coexist with costly added mass and infinite reverse KL.

Let P, R be laws on a finite domain $\mathcal{X}$. For $0 < \varepsilon < 1$, set $Q = (1-\varepsilon)P + \varepsilon R$. For $X \sim P$, $Z \sim R$, and $Y \sim Q$, prove

$$D(P, Q) \le -\log(1-\varepsilon), \qquad \mathbb{E}[c(Y)] = (1-\varepsilon)\mathbb{E}[c(X)] + \varepsilon\mathbb{E}[c(Z)]. \tag{4.70}$$

If $c = 0$ under P and $c = C_{\max} > 0$ under R, compute the generated cost and both KL directions exactly.

Exercise 36: Omitted outcomes and reverse KL.

◎ Quantify how reverse KL permits omitting successful outcomes while forward KL detects the support loss.

Let P be a law on a finite domain $\mathcal{X}$. For $0 < P(A) < 1$, define the conditional law

$$Q(x) = \frac{P(x)\mathbb{I}\{x \in A\}}{P(A)}, \qquad x \in \mathcal{X}.$$

Prove

$$D(Q, P) = -\log P(A), \qquad D(P, Q) = +\infty. \tag{4.71}$$

What fraction of the target probability can be omitted with reverse KL $\log 2$? If P is uniform on $M \ge 2$ zero-cost outcomes, compute reverse KL and task cost for an arbitrary Q supported there.

Hints Use the disjoint-support mixture calculation following Proposition 4.2.4.

Exercise 37: Rare failures and reverse KL.

◎ Use data processing to convert reverse KL into a failure-probability bound and analyze its rare-event limit.

Let P, Q be laws on a finite domain $\mathcal{X}$, let $B = \{x : c(x) > 0\}$, $P(B) = \varepsilon \in (0, 1)$, and $Q(B) = q$. Prove

$$D(Q, P) \ge \mathrm{kl}(q, \varepsilon) \ge q \log(1/\varepsilon) - \log 2,$$

and deduce that $D(Q,P) \le K$ implies

$$q \le \min\left\{1, \frac{K+\log 2}{\log(1/\varepsilon)}\right\}.$$

For fixed $q \in (0,1)$, compare the limits of $\mathrm{kl}(q,\varepsilon)$ and $\mathrm{kl}(\varepsilon,q)$ as $\varepsilon \to 0$. Explain what changes when $P(B) = 0$.

Hints Apply data processing to $x \mapsto \mathbb{I}\{x \in B\}$, then use the entropy bound for a bit.

Exercise 38: Variational representation of reverse KL.

Derive the reverse-KL variational representation by pointwise optimization, including its support boundary cases.

Let P, Q be laws on a finite space $\mathcal{X}$, with $X \sim P$ and $Y \sim Q$. Prove the identity in Eq. (4.48):

$$D(Q,P) = \sup_{g:\mathcal{X}\to\mathbb{R}} \left\{1 + \mathbb{E}[g(Y)] - \mathbb{E}[e^{g(X)}]\right\}.$$

Include the cases where either law has zero probabilities. When P, Q are strictly positive, show that the supremum is attained by $g^*(x) = \log(Q(x)/P(x))$.

Hints Optimize each value $g(x)$ separately.

Contraction and error recovery

Exercise 39: The Dobrushin contraction coefficient.

Prove that the Dobrushin coefficient is the optimal total-variation contraction factor.

Let $\widehat{K}$ be a Markov kernel on a finite state space. Prove that the smallest $\rho \ge 0$ satisfying

$$\|\alpha\widehat{K} - \beta\widehat{K}\|_{\mathrm{TV}} \le \rho\,\|\alpha - \beta\|_{\mathrm{TV}}$$

for all probability laws α, β is

$$\rho(\widehat{K}) = \max_{x,x'} \|\widehat{K}(x,\cdot) - \widehat{K}(x',\cdot)\|_{\mathrm{TV}}.$$

Hints Normalize the positive and negative parts of $\alpha - \beta$ to obtain two probability laws.

Exercise 40: Recovery from one initial error.

Use contraction to derive the exact decay of one initial error and its accumulated cost.

For the recovery family Q_r, show that the contraction factor is $1 - r$ and that the marginal error probability is

$$e_t = (1 - e^{-\kappa})(1-r)^{t-1}.$$

Deduce

$$\mathbb{E}_{Q_r}[c_{\mathrm{fin}}] = (1 - e^{-\kappa})(1-r)^{T-1}$$

and

$$\mathbb{E}_{Q_r}[c_{\mathrm{cnt}}] = \begin{cases} (1-e^{-\kappa})[1-(1-r)^T]/r, & r > 0, \\ T(1-e^{-\kappa}), & r = 0. \end{cases}$$

Verify that the stability recurrence is an equality for this family.

Exercise 41: Fresh errors under contraction.

Derive the long-run balance between newly introduced errors and their correction.

This exercise compares recovery from an initial error with continual correction of new errors. Let $Q_{\varepsilon,r}$, with $X_0 = 0$, use

$$Q_{\varepsilon,r}(X_{t+1} = 1 \mid X_t = 0) = \varepsilon, \qquad Q_{\varepsilon,r}(X_{t+1} = 0 \mid X_t = 1) = r,$$

where $0 < \varepsilon < 1$, $0 \le r \le 1$, and $0 \le t < T$.

(a) For $p_t = Q_{\varepsilon,r}(X_t = 1)$, prove

$$p_0 = 0, \qquad p_{t+1} = \varepsilon + (1 - \varepsilon - r)p_t, \qquad p_t = \frac{\varepsilon}{\varepsilon + r}[1 - (1 - \varepsilon - r)^t]. \tag{4.72}$$

Identify the choices of r giving the two models in Exercise 9.

(b) Compute the contraction factor $|1 - \varepsilon - r|$. For $0 \le r \le 1 - \varepsilon$, prove

$$\mathbb{E}_{Q_{\varepsilon,r}}[c_{\text{fin}}] \le \frac{\varepsilon}{\varepsilon + r}, \qquad \frac{1}{T}\mathbb{E}_{Q_{\varepsilon,r}}[c_{\text{cnt}}] \le \frac{\varepsilon}{\varepsilon + r}. \tag{4.73}$$

Recover these bounds from Theorem 4.5.1, using $e_1 = \varepsilon$ and local error ε at every subsequent step. At $r = 0$, why does the positive contraction margin still give the bound one? How large must r be relative to ε for the limiting error probability to be at most $s \in (0, 1)$?

Exercise 42: Cumulative cost under recovery.

Bound cumulative cost discrepancy by summing marginal recovery bounds and reversing the resulting geometric sums.

Use the notation and assumptions of Theorem 4.5.1, with $\rho < 1$. For $c_t : [N] \to [0, 1]$, define

$$c_{\text{sum}}(x_{1:T}) = \sum_{t=1}^{T} c_t(x_t).$$

Let $X_{1:T} \sim P$ and $Y_{1:T} \sim Q$. Prove

$$|\mathbb{E}[c_{\text{sum}}(X_{1:T})] - \mathbb{E}[c_{\text{sum}}(Y_{1:T})]| \le \frac{e_1 + \sum_{t=2}^{T} \varepsilon_t}{1 - \rho}. \tag{4.74}$$

Hints Sum the marginal bounds before bounding the geometric sums.

Exercise 43: Forgetting errors and retaining information.

Distinguish contractive working-state dynamics from persistent task information.

Use the reset mixture and persistent-state model from Section 4.5.3.

(a) Prove the contraction bound for the reset mixture K_r. Construct a process whose next-token law uses only the last token and whose initial bit can be recovered from every later token.

(b) In Eq. (4.56), suppose the conditional working-state kernels contract uniformly with factor $\rho < 1$. Explain why contraction applies when comparing equal M, while different persistent M's can remain distinguishable indefinitely.

4.A Where finiteness is used

The finite sequence space guarantees bounded costs and permits term-by-term differentiation. This section records which properties the final set of results uses, so we can also apply them to other probability spaces. The remaining appendices follow the order in which their topics first require deferred arguments in the main text: reverse-KL concentration, forward-KL variance control, finite-sample fitting, reverse-KL optimization, and history-dependent recovery.

Distributional comparisons. Data processing, the TV bound (Proposition 4.2.2), Pinsker's inequality, the zero-cost bound (Proposition 4.2.4), the exponential-moment bound (Proposition 4.2.6), and the variance-sensitive bounds (Propositions 4.2.9 and 4.C.1) hold on general probability spaces with the same bounded-cost assumptions. Their proofs use integrals of densities in place of sums. The mixture and conditioning identities in Exercise 35 also extend, with integrable costs. The support requirement becomes *absolute continuity*: finite $D(Q, P)$ requires $P(A) = 0 \Rightarrow Q(A) = 0$ for every measurable event A. The Markov stability argument in Theorem 4.5.1 uses only the stated contraction and expected local-error bounds. On general state spaces, the same proof applies when the conditional laws exist.

Extrema at fixed KL. For fixed P, each nonempty KL level set in Eq. (4.17) is compact. Indeed, the probability simplex is compact, and $D(P, Q)$ is continuous wherever $Q(x) > 0$ on the support of P. It tends to infinity if one of those probabilities tends to zero. Hence every finite level set is closed. The infinite level set is the finite union of the closed faces $\{Q : Q(x) = 0\}$ with $P(x) > 0$. Since $d_c(P, Q)$ is continuous in Q, its minimum and maximum on each level are attained. This justifies both endpoint formulas in Proposition 4.2.10. On other spaces, compact level sets and continuity of the cost discrepancy give the same conclusion. The budget definitions of the moduli, their monotonicity, and the lower-bound inversion implication use only well-defined discrepancies; identifying an exact converse modulus from lower endpoints additionally uses attainment of those endpoints.

Boundedness and exponential moments. Every real-valued cost on $[N]^T$ is bounded, so its exponential moments exist for every λ and it has a finite subgaussian proxy. Bounded costs on any probability space have these same properties. For an unbounded cost, Proposition 4.2.6 still holds when $\mathbb{E}_P[|c|], \mathbb{E}_Q[|c|] < \infty$, using those $\lambda > 0$ for which $\mathbb{E}_P[e^{\lambda c}] < \infty$. The subgaussian condition in Theorem 4.2.8 supplies the needed exponential moments in both directions. Attainment of the optimal proxy requires only that a finite valid proxy exist; Appendix 4.B.2 proves this directly. The entropy decomposition in Eq. (4.5) uses finite $H(P)$; for countable alphabets, $H(P) < \infty$ keeps that decomposition meaningful for comparing log loss with KL.

Differentiating expectations. Finite sums justify the reverse-KL gradient in Appendix 4.E. For general densities, sufficient conditions are differentiability on a fixed support and integrable bounds, uniform in a neighborhood of the parameter, on the derivatives of the density and the objective integrand. Dominated convergence then permits differentiation under the integral. For the exponential tilts in Appendix 4.B, boundedness of c supplies such bounds on every bounded interval of λ.

The maximum-cost endpoint in the exact modulus. The proof of Proposition 4.B.1 uses finiteness to obtain a maximum cost m with $P(c = m) > 0$. For a bounded cost on a general

space, take m to be its *essential supremum* under P: the smallest number with $P(c \le m) = 1$. If $P(c = m) > 0$, the same endpoint argument applies. If $P(c = m) = 0$, the tilted KL increases to infinity: the tilted laws concentrate on sets $\{c > m - \varepsilon\}$ whose P-probabilities decrease to zero. Every finite positive KL budget is therefore reached by a finite tilt. Thus the exact-modulus identity still holds for bounded costs; finiteness simplifies its endpoint proof.

Finite model classes. The finite-class guarantee in Theorem 4.4.4 uses a union bound over models, producing $\log |\mathcal{Q}|$; its likelihood-ratio argument also works for general observation spaces with densities. Finite-dimensional state representations are treated in Lecture 5.

4.B Reverse KL and exponential tilting

4.B.1 From exponential moments to subgaussian control

Proposition 4.2.6 (Cost control from exponential moments)**.** *Let P, Q be distributions over a finite set $\mathcal{X}$ and let $c : \mathcal{X} \to \mathbb{R}$. Then.*

$$\mathbb{E}_Q[c] - \mathbb{E}_P[c] \le \inf_{\lambda>0} \frac{D(Q,P) + \log \mathbb{E}_P[e^{\lambda(c-\mathbb{E}_P[c])}]}{\lambda}. \tag{4.12}$$

Proof. The basic inequality uses the full exponential-moment function of the cost. Put $\psi(\lambda) = \log \mathbb{E}_P[e^{\lambda(c-\mu)}]$. Define the tilted law by

$$P_\lambda(x) = P(x)e^{\lambda(c(x)-\mathbb{E}_P\, c)-\psi(\lambda)}.$$

The normalizing term $\psi(\lambda)$ makes P_λ a probability law. Writing $\kappa = D(Q,P)$, we have

$$0 \le D(Q,P_\lambda) = \kappa - \lambda(\mathbb{E}_Q[c] - \mathbb{E}_P[c]) + \psi(\lambda).$$

Consequently,

$$\mathbb{E}_Q[c] - \mathbb{E}_P[c] \le \inf_{\lambda>0} \frac{\kappa + \psi(\lambda)}{\lambda},$$

which proves Proposition 4.2.6. □

Under the subgaussian condition,

$$\mathbb{E}_Q[c] - \mathbb{E}_P[c] \le \inf_{\lambda>0} \left(\frac{\kappa}{\lambda} + \frac{\lambda v^2}{2} \right) = v\sqrt{2\kappa}.$$

Apply the same argument to $-c$ for the other direction. If $v = 0$, take $\lambda \to \infty$ in each direction.

For completeness, the exponential-moment bound for averages follows from Hoeffding's lemma. If $Y \in [0,1]$, let $h(s) = \log \mathbb{E}[e^{s(Y-\mathbb{E}[Y])}]$. Then $h(0) = h'(0) = 0$, and $h''(s)$ is the variance of Y under its tilted law. Every law on $[0,1]$ has variance at most $1/4$, giving $h(s) \le s^2/8$. For independent $f_t(X_t) \in [0,1]$, factor the expectation of the product (Lecture 2, Exercise 27(a)) to obtain

$$\begin{aligned}
&\log \mathbb{E} \exp\left[\frac{\lambda}{T} \sum_{t=1}^{T} (f_t(X_t) - \mathbb{E}[f_t(X_t)]) \right] \\
&\quad = \sum_{t=1}^{T} \log \mathbb{E} \exp\left[\frac{\lambda}{T} (f_t(X_t) - \mathbb{E}[f_t(X_t)]) \right] \\
&\quad \le T \, \frac{(\lambda/T)^2}{8} = \frac{\lambda^2}{8T}.
\end{aligned}$$

Hence the variance proxy is $v^2 = 1/(4T)$.

4.B.2 Optimal variance proxies

The smallest valid proxy is attained whenever any finite proxy exists, even on an infinite probability space. To see this, let c be an integrable subgaussian cost and write $\psi(\lambda) = \log \mathbb{E}_P[e^{\lambda(c-\mathbb{E}_P[c])}]$. The constraints defining a valid proxy give the identity

$$v_P(c) = \sup_{\lambda \neq 0} \frac{2\psi(\lambda)}{\lambda^2}.$$

This value satisfies every constraint, so the set of valid proxies is exactly $[v_P(c), \infty)$.

For the bounded costs used here, expansion at zero gives

$$\psi(\lambda) = \frac{\lambda^2 \sigma_P(c)^2}{2} + o(\lambda^2),$$

and hence $\sigma_P(c)^2 \le v_P(c)$. Scaling Hoeffding's lemma from $[0,1]$ to $[0, C_{\max}]$ gives $\psi(\lambda) \le \lambda^2 C_{\max}^2/8$, so $v_P(c) \le C_{\max}^2/4$.

4.B.3 The exact reverse-KL cost modulus

How much can the expected cost increase if the reverse KL is at most κ? Retaining the full exponential-moment function answers this question exactly when we allow all model laws.

Proposition 4.B.1 (Exact reverse-KL cost conversion)**.** *Fix a law P and a cost c. Let $\psi(\lambda) = \log \mathbb{E}_P[e^{\lambda(c - \mathbb{E}_P[c])}]$. For every $\kappa \ge 0$,*

$$\sup_{Q:\, D(Q,P) \le \kappa} (\mathbb{E}_Q[c] - \mathbb{E}_P[c]) = \inf_{\lambda > 0} \frac{\kappa + \psi(\lambda)}{\lambda}. \tag{4.75}$$

The supremum ranges over all probability laws Q.

Proof. Equation (4.12) gives the upper bound. We construct laws attaining it. Put $\mu = \mathbb{E}_P[c]$ and use the tilted laws $P_\lambda(x) = P(x) e^{\lambda(c(x) - \mu) - \psi(\lambda)}$. Differentiating ψ gives

$$\psi'(\lambda) = \mathbb{E}_{P_\lambda}[c] - \mu, \qquad D(P_\lambda, P) = \lambda \psi'(\lambda) - \psi(\lambda).$$

If c is constant on $\operatorname{supp}(P)$, both sides of Eq. (4.75) are zero. Otherwise let

$$m = \max_{x:\, P(x) > 0} c(x), \qquad \mathcal{A} = \{x : P(x) > 0,\ c(x) = m\}, \qquad \kappa_* = -\log P(\mathcal{A}) > 0.$$

As λ increases from zero to infinity, the tilted KL increases continuously from zero to κ_*. Indeed, its derivative is $\lambda \psi''(\lambda) > 0$, and $P_\lambda \to P(\cdot \mid \mathcal{A})$. For $0 < \kappa < \kappa_*$, choose $\lambda > 0$ with $D(P_\lambda, P) = \kappa$. Then

$$\mathbb{E}_{P_\lambda}[c] - \mu = \psi'(\lambda) = \frac{\kappa + \psi(\lambda)}{\lambda},$$

so this law attains the upper bound.

For $\kappa \ge \kappa_*$, the law $P(\cdot \mid \mathcal{A})$ is feasible and has cost m, the largest possible cost on the support of P. Also $(\kappa + \psi(\lambda))/\lambda \to m - \mu$ as $\lambda \to \infty$, proving equality in this case. For $\kappa = 0$, the only feasible law is P, and $\psi(\lambda)/\lambda \to 0$ as $\lambda \downarrow 0$. □

For the absolute discrepancy, apply the proposition to both c and $-c$:

$$\sup_{Q:\, D(Q,P) \le \kappa} d_c(P, Q) = \max_{s \in \{-1, 1\}} \inf_{\lambda > 0} \frac{\kappa + \psi(s\lambda)}{\lambda}.$$

This is the reverse-KL counterpart of the forward-KL modulus in Eq. (4.18). The subgaussian conversion replaces $\psi(s\lambda)$ by $\lambda^2 v_P(c)/2$. Any slack introduced at that step comes from summarizing the cost's full exponential-moment function by one quadratic bound.

4.B.4 A full-support version of the binary comparison

The infinite reverse KLs in the exact all-zero example are caused by its support. A small perturbation gives finite values and still distinguishes the generators. Take $P_\delta = \mathrm{Bernoulli}(\delta)^{\otimes T}$; compare Q_{ind} from Exercise 9 with $Q_{\varepsilon,r}$ from Exercise 39. For $0 < \delta, \varepsilon, r < 1$, all three laws have full support. With $\mathrm{kl}(a, b) = a \log(a/b) + (1-a)\log((1-a)/(1-b))$, the chain rule gives

$$D(P_\delta, Q_{\varepsilon,r}) = \mathrm{kl}(\delta, \varepsilon) + (T-1)\big[(1-\delta)\,\mathrm{kl}(\delta, \varepsilon) + \delta\,\mathrm{kl}(\delta, 1-r)\big],$$
$$D(Q_{\varepsilon,r}, P_\delta) = \sum_{t=1}^{T}\big[(1-p_{t-1})\,\mathrm{kl}(\varepsilon, \delta) + p_{t-1}\,\mathrm{kl}(1-r, \delta)\big].$$

Here p_t is the generated probability of a one, given by Eq. (4.72). For the independent law the two KLs are $T\,\mathrm{kl}(\delta, \varepsilon)$ and $T\,\mathrm{kl}(\varepsilon, \delta)$. At $T = 1000, \varepsilon = 10^{-3}, \delta = r = 10^{-6}$:

Q	$D(P_\delta, Q)$	$D(Q, P_\delta)$	$\mathbb{E}_Q[c_{\mathrm{cnt}}]$
Q_{ind}	0.993	5.909	1
$Q_{\varepsilon,r}$	1.006	5082.198	368.224

The expected data cost is 0.001. Forward KL weights the transition out of one by the tiny data probability δ; reverse KL weights it by the much larger generated probability p_{t-1}.

4.C The variance-sensitive bound and comparisons

4.C.1 Cost variance and Hellinger distance

The *Hellinger distance* between two laws P, Q is

$$h(P, Q)^2 = \sum_x (\sqrt{P(x)} - \sqrt{Q(x)})^2.$$

We use the normalization $0 \le h(P, Q)^2 \le 2$. Expanding the square and using $\sum_x P(x) = \sum_x Q(x) = 1$ gives

$$\sum_x \sqrt{P(x)Q(x)} = 1 - \frac{h(P, Q)^2}{2}. \tag{4.76}$$

The next bound uses this distance to control the cost discrepancy.

Proposition 4.C.1 (Hellinger distance and cost variance)**.** *If $0 \le c \le C_{\max}$, then*

$$d_c(P, Q) \le 2\sigma_P(c)h(P, Q) + C_{\max}h(P, Q)^2. \tag{4.77}$$

Moreover, $h(P, Q)^2 \le D(P, Q)$.

Proof. Put $\Delta(x) = \sqrt{Q(x)} - \sqrt{P(x)}$ and $\bar{c} = c - \mathbb{E}_P[c]$. Expand

$$\mathbb{E}_Q[c] - \mathbb{E}_P[c] = 2\sum_x \bar{c}(x)\sqrt{P(x)}\Delta(x) + \sum_x \bar{c}(x)\Delta(x)^2.$$

Cauchy–Schwarz bounds the first term in absolute value by $2\sigma_P(c)h(P, Q)$; $|\bar{c}| \le C_{\max}$ bounds the second.

Let $X \sim P$ and $\kappa = D(P,Q) < \infty$. Jensen's inequality for exp gives

$$\sum_x \sqrt{P(x)Q(x)} = \mathbb{E}\left[\exp\left(-\frac{1}{2}\log\frac{P(X)}{Q(X)}\right)\right] \geq \exp\left(-\frac{1}{2}\mathbb{E}\left[\log\frac{P(X)}{Q(X)}\right]\right) = e^{-\kappa/2}.$$

Now Eq. (4.76) and $1 - e^{-u} \leq u$ give $h(P,Q)^2 \leq 2(1 - e^{-\kappa/2}) \leq \kappa$. □

Substituting $h(P,Q)^2 \leq D(P,Q)$ into Eq. (4.77) proves the following restated result.

Proposition 4.2.9 (Cost variance and forward KL)**.** *If $0 \leq c \leq C_{\max}$ and $D(P,Q) < \infty$, then*

$$d_c(P,Q) \leq 2\sigma_P(c)\sqrt{D(P,Q)} + C_{\max}D(P,Q). \tag{4.16}$$

Proof. Apply Eq. (4.77) and $h(P,Q)^2 \leq D(P,Q)$. Since $h(P,Q) \leq \sqrt{D(P,Q)}$, the stated bound follows. □

4.C.2 Which bound is stronger? Two single-bit comparisons

One bit suffices to show that either the variance-sensitive or the subgaussian bound can be smaller. Throughout this subsection, $c(x) = x$ on $\{0,1\}$, so $C_{\max} = 1$.

Proposition 4.C.2 (Comparison near a fair coin and near zero cost)**.** *(a) Suppose*

$$P = \text{Bernoulli}(1/2), \qquad Q = \text{Bernoulli}(1/2 + \delta), \qquad 0 < |\delta| < 1/2.$$

As $\delta \to 0$, the Pinsker and subgaussian bounds are asymptotic to $d_c(P,Q) = |\delta|$, while the variance-sensitive bound is asymptotic to $\sqrt{2}|\delta|$.

(b) Suppose

$$P = \text{Bernoulli}(\varepsilon^2), \qquad Q = \text{Bernoulli}(\varepsilon), \qquad 0 < \varepsilon < 1.$$

As $\varepsilon \downarrow 0$, the actual discrepancy and the variance-sensitive bound are both asymptotic to ε. The Pinsker bound is asymptotic to $\sqrt{\varepsilon/2}$, and the subgaussian bound is asymptotic to $\sqrt{\varepsilon/\log(1/\varepsilon)}$.

Proof. For (a), direct expansion gives

$$D(P,Q) = -\tfrac{1}{2}\log(1 - 4\delta^2) = 2\delta^2 + O(\delta^4),$$
$$D(Q,P) = 2\delta^2 + O(\delta^4).$$

Also $\sigma_P(c) = 1/2$ and $v_P(c) = 1/4$: every variance proxy is at least the variance, and Hoeffding's lemma gives equality for a fair coin. Thus the reverse-KL term of Eq. (4.15) is asymptotic to $|\delta|$. The minimum of its two terms is between the actual discrepancy and this term, so it has the same asymptotic value. Substitution into Eqs. (4.10) and (4.16) gives the other two comparisons.

For (b), the Bernoulli KL formula gives

$$d_c(P,Q) = \varepsilon - \varepsilon^2 \sim \varepsilon,$$
$$D(P,Q) \sim \varepsilon, \qquad D(Q,P) \sim \varepsilon\log(1/\varepsilon),$$
$$\sigma_P(c) = \varepsilon\sqrt{1 - \varepsilon^2} \sim \varepsilon.$$

The variance-sensitive bound is therefore $2\sigma_P(c)\sqrt{D(P,Q)} + D(P,Q) \sim \varepsilon$. To evaluate the subgaussian bound, we use the sharp Bernoulli proxy [Schlemm, 2014, Eq. (1.1) and Theorem 1.1]:

$$v_{\mathrm{Bernoulli}(p)}(c) = \frac{1-2p}{2\log((1-p)/p)}, \qquad 0 < p < 1, \quad p \neq \tfrac{1}{2},$$

with value $1/4$ at $p = 1/2$. In particular,

$$v_Q(c) \sim \frac{1}{2\log(1/\varepsilon)}, \qquad v_P(c) \sim \frac{1}{4\log(1/\varepsilon)}.$$

Consequently,

$$\sqrt{2v_Q(c)D(P,Q)} \sim \sqrt{\frac{\varepsilon}{\log(1/\varepsilon)}},$$
$$\sqrt{2v_P(c)D(Q,P)} \sim \sqrt{\frac{\varepsilon}{2}}.$$

The forward-KL term is the smaller one for sufficiently small ε. Pinsker gives $\sqrt{D(P,Q)/2} \sim \sqrt{\varepsilon/2}$. □

Near a fair coin, Pinsker and the subgaussian inequality give the correct leading constant. Near zero cost, the variance-sensitive bound uses the small data variance to give the correct leading order, whereas even the optimal subgaussian proxies give a much larger bound. The first example requires square-root dependence on KL in a general cost bound. The zero-cost equality example in Section 4.2 requires linear dependence when the data variance vanishes. These explain the two terms in Eq. (4.16).

4.C.3 Moduli of continuity in analysis and approximation

A modulus of continuity quantifies how much a function's output can change when its input changes by at most a specified amount. For a function $h : [0,1] \to \mathbb{R}$, its *modulus of continuity* is

$$\omega_h(\delta) = \sup_{\substack{x,y\in[0,1] \\ |x-y|\le\delta}} |h(x) - h(y)|.$$

Instead of only asking whether nearby inputs have nearby outputs, we ask how nearby the outputs must be. For example, a Lipschitz bound $|h(x) - h(y)| \le L|x-y|$ gives $\omega_h(\delta) \le L\delta$.

Proposition 4.C.3 (Quantitative uniform continuity)**.** *A function $h : [0,1] \to \mathbb{R}$ is uniformly continuous if and only if $\omega_h(\delta) \to 0$ as $\delta \downarrow 0$.*

Proof. Uniform continuity makes every difference in the defining supremum at most ε once δ is sufficiently small. Conversely, $\omega_h(\delta) < \varepsilon$ gives $|h(x) - h(y)| < \varepsilon$ whenever $|x-y| \le \delta$. □

In our comparison, the function is the expected-cost map $P \mapsto \mathbb{E}_P[c]$. We measure input discrepancy using KL and output discrepancy using d_c. Taking the supremum over P as well as Q gives the uniform comparison. If $\mathcal{P}$ contains all data laws, Pinsker's inequality gives $\omega^{\mathcal{P}}_{D\to c}(\kappa) \le C_{\max}\sqrt{\kappa/2}$ for $0 \le c \le C_{\max}$: small KL uniformly forces nearby expected costs.

The expected-cost map can send distinct laws to the same value. Accordingly, d_c is a *pseudometric*: it is symmetric, satisfies the triangle inequality, and vanishes on equal laws, but can also vanish on distinct laws. The converse modulus quantifies how far apart the input laws can be while their expected costs remain close. Propositions 4.3.2 and 4.3.3 show how much this depends on the cost and reference law.

From continuity to approximation rates. A quantitative continuity bound can also tell us how accurately a function can be approximated by polynomials. Jackson's theorem states that, for a universal constant C, every continuous $h : [0,1] \to \mathbb{R}$ satisfies

$$\inf_{\deg p \le n} \|h - p\|_\infty \le C\, \omega_h(1/n), \qquad n \ge 1.$$

Thus $\omega_h(\delta) \le L\delta^\alpha$, with $0 < \alpha \le 1$, guarantees a degree-n approximation with uniform error at most $CLn^{-\alpha}$. The modulus turns a qualitative continuity statement into a bound on the degree needed for a given approximation error.

4.D Likelihood fitting and conditional support

4.D.1 Proof of the finite-class likelihood guarantee

Theorem 4.4.4 (Approximate maximum likelihood controls cost)**.** *Let $\mathcal{Q}$ be a finite class containing P, and observe n independent sequences with law P. Let $0 \le c \le C_{\max}$. Suppose $\widehat{Q} \in \mathcal{Q}$ satisfies Eq. (4.36) for $\eta \ge 0$. For $0 < \delta < 1$, put*

$$b = \frac{2\log(|\mathcal{Q}|/\delta)}{n} + T\eta.$$

With probability at least $1 - \delta$,

$$d_c(P, \widehat{Q}) \le 2\sigma_P(c)\sqrt{b} + C_{\max} b. \tag{4.38}$$

In particular, if $\mathbb{E}_P[c] = 0$, then $\mathbb{E}_{\widehat{Q}}[c] \le C_{\max} b$.

Proof. We first prove $h(P, \widehat{Q})^2 \le b$ with probability at least $1 - \delta$, then apply Proposition 4.C.1. The probability tools below are proved in Lecture 2, Exercise 27; the inequality $1 - u \le e^{-u}$ is proved in Exercise 5(a) of that lecture.

For a fixed $Q \in \mathcal{Q}$, define the likelihood ratio

$$\Lambda_Q = \prod_{i=1}^{n} \frac{Q(X_{1:T}^{(i)})}{P(X_{1:T}^{(i)})}.$$

The denominators are positive almost surely because each observed sequence has law P. Independence of the sequences gives

$$\begin{aligned}\mathbb{E}[\sqrt{\Lambda_Q}] &= \prod_{i=1}^{n} \mathbb{E}\left[\sqrt{\frac{Q(X_{1:T}^{(i)})}{P(X_{1:T}^{(i)})}}\right] \\ &= \left(\sum_{x : P(x) > 0} P(x)\sqrt{\frac{Q(x)}{P(x)}}\right)^n = \left(\sum_x \sqrt{P(x)Q(x)}\right)^n.\end{aligned}$$

Using Eq. (4.76) and $1 - u \le e^{-u}$,

$$\mathbb{E}[\sqrt{\Lambda_Q}] = \left(1 - \frac{h(P,Q)^2}{2}\right)^n \le e^{-nh(P,Q)^2/2}.$$

Since $P \in \mathcal{Q}$, Eq. (4.36) gives $\widehat{L}(\widehat{Q}) \le \widehat{L}(P) + \eta$. Hence

$$\log \Lambda_{\widehat{Q}} = nT\big(\widehat{L}(P) - \widehat{L}(\widehat{Q})\big) \ge -nT\eta.$$

For a fixed candidate Q, apply Markov's inequality to $\sqrt{\Lambda_Q} \ge 0$:

$$\begin{aligned}\mathbb{P}(\Lambda_Q \ge e^{-nT\eta}) &= \mathbb{P}(\sqrt{\Lambda_Q} \ge e^{-nT\eta/2}) \\ &\le e^{nT\eta/2}\mathbb{E}[\sqrt{\Lambda_Q}] \\ &\le \exp\left(\frac{nT\eta}{2} - \frac{nh(P,Q)^2}{2}\right).\end{aligned}$$

For every Q in the set $\mathcal{B} = \{Q \in \mathcal{Q} : h(P,Q)^2 > b\}$, this probability is at most $\delta/|\mathcal{Q}|$. The selected model can belong to $\mathcal{B}$ only if one of these likelihood-ratio events occurs:

$$\{h(P,\widehat{Q})^2 > b\} \subseteq \bigcup_{Q\in\mathcal{B}} \{\Lambda_Q \ge e^{-nT\eta}\}.$$

The union bound therefore yields

$$\mathbb{P}(h(P,\widehat{Q})^2 > b) \le \sum_{Q\in\mathcal{B}} \mathbb{P}(\Lambda_Q \ge e^{-nT\eta}) \le |\mathcal{B}|\frac{\delta}{|\mathcal{Q}|} \le \delta.$$

On the complementary event, Proposition 4.C.1 gives $d_c(P,\widehat{Q}) \le 2\sigma_P(c)\sqrt{b} + C_{\max}b$. If $\mathbb{E}_P[c] = 0$, nonnegativity of c implies that $c = 0$ P-almost surely, so $\sigma_P(c) = 0$ and $d_c(P,\widehat{Q}) = \mathbb{E}_{\widehat{Q}}[c]$. □

The independent observations here are complete sequences. Dependence between tokens is part of their joint law P; it never enters this likelihood-ratio calculation as an independence assumption. Lecture 6 studies how much information dependent observations supply.

4.D.2 Proofs of the parameter-sharing results

Two-sided risk and the leading constant.

Proposition 4.4.1 (Sequence KL under parameter sharing)**.** *In the grouped independent-bit model, fit each probability by the pooled smoothed frequency from n independent length-T training sequences. Fix $0 < a < 1/2$ and suppose $p_j \in [a, 1-a]$. Then*

$$\kappa_n(T) \asymp_a \frac{d(T)}{n}, \qquad n, T \ge 1. \tag{4.31}$$

Here $\asymp_a$ means upper and lower bounds by positive constant multiples, with constants depending only on a. If $n\min_{1\le j\le d(T)} m_j \to \infty$, then more precisely

$$\kappa_n(T) \sim \frac{d(T)}{2n}. \tag{4.32}$$

Proof of Proposition 4.4.1. First consider $B \sim \text{Binomial}(L,p)$, with $L \ge 1$ and $0 < p < 1$. The smoothed estimator is $\widehat{p} = (B+1)/(L+2)$. Using $1/(B+1) = \int_0^1 u^B\,du$,

$$\mathbb{E}\left[\frac{1}{B+1}\right] = \int_0^1 (1-p+pu)^L\,du = \frac{1-(1-p)^{L+1}}{(L+1)p}.$$

Consequently,

$$p\,\mathbb{E}[1/\widehat{p}] \le \frac{L+2}{L+1}, \qquad (1-p)\,\mathbb{E}[1/(1-\widehat{p})] \le \frac{L+2}{L+1}.$$

The second inequality follows by applying the same calculation to $L - B$. Jensen's inequality gives

$$\begin{aligned}\mathbb{E}[\mathrm{kl}(p,\widehat{p})] &\le p\log\big(p\,\mathbb{E}[1/\widehat{p}]\big) + (1-p)\log\big((1-p)\,\mathbb{E}[1/(1-\widehat{p})]\big)\\ &\le \log\frac{L+2}{L+1} \le \frac{1}{L+1}.\end{aligned}$$

For $p = 0, 1$, the loss equals $\log((L+2)/(L+1))$ deterministically, so the bound also holds.

For group j, the pooled count is binomial with $L = nm_j$. Independence of the test tokens makes the sequence KL the sum of the Bernoulli KLs. Therefore

$$\kappa_n(T) = \sum_{j=1}^{d(T)} m_j\,\mathbb{E}[\mathrm{kl}(p_j,\widehat{p}_j)] \le \sum_{j=1}^{d(T)} \frac{m_j}{nm_j+1} \le \frac{d(T)}{n}.$$

For the lower bound, the binomial mean and variance give

$$\mathbb{E}[(\widehat{p}-p)^2] = \frac{Lp(1-p)+(1-2p)^2}{(L+2)^2}.$$

Pinsker's inequality implies

$$\mathbb{E}[\mathrm{kl}(p,\widehat{p})] \ge 2\mathbb{E}[(\widehat{p}-p)^2] \ge \frac{2Lp(1-p)}{(L+2)^2}.$$

With $L = nm_j$ and $p_j \in [a, 1-a]$,

$$m_j\,\mathbb{E}[\mathrm{kl}(p_j,\widehat{p}_j)] \ge \frac{2a(1-a)}{n}\left(\frac{nm_j}{nm_j+2}\right)^2 \ge \frac{2a(1-a)}{9n}.$$

Summing over groups proves Eq. (4.31), with explicit bounds

$$\frac{2a(1-a)}{9}\frac{d(T)}{n} \le \kappa_n(T) \le \frac{d(T)}{n}.$$

For the leading constant, the same mean-squared error formula gives, uniformly for $p \in [a, 1-a]$,

$$\mathbb{E}[(\widehat{p}-p)^2] = \frac{p(1-p)}{L} + O(L^{-2}).$$

In a fixed neighborhood of p, Taylor's theorem gives

$$\mathrm{kl}(p,q) = \frac{(q-p)^2}{2p(1-p)} + O(|q-p|^3).$$

Binomial moment bounds give $\mathbb{E}[|\widehat{p}-p|^3] = O(L^{-3/2})$. Outside that neighborhood the probability is exponentially small by a binomial tail bound, while $\mathrm{kl}(p,\widehat{p}) \le \log(L+2)$. Splitting the expectation between the two events yields

$$\mathbb{E}[\mathrm{kl}(p,\widehat{p})] = \frac{1}{2L} + O(L^{-3/2}).$$

The constants can be chosen uniformly for $p \in [a, 1-a]$, $0 < a < 1/2$.

Summing over groups gives

$$\kappa_n(T) = \frac{d(T)}{2n} + O_a\left(\frac{1}{n^{3/2}}\sum_{j=1}^{d(T)} m_j^{-1/2}\right).$$

The relative error is $O_a((n\min_j m_j)^{-1/2})$, which tends to zero under the stated condition. This proves Eq. (4.32). □

Learning rates and task-cost intervals.

Proposition 4.4.2 (Learning rates and task-cost intervals)**.** *For any d with $1 \le d(T) \le T$, there exist a data law, terminal and counting costs, and estimator families such that, for all $n \ge 1, T \ge 2$, the estimators have common expected sequence KL $\kappa_n(T) \asymp d(T)/n$ and their expected costs range over exactly*

$$\left[0, 1 - e^{-\kappa_n(T)/2}\right] \quad \text{and} \quad \left[1 - e^{-\kappa_n(T)/2}, T(1 - e^{-\kappa_n(T)/2})\right],$$

respectively.

Proof. Let the U_t be independent fair bits and $V_t = 0$, giving one law P on infinite sequences. Use terminal cost V_T and counting cost $\sum_{t=1}^{T} V_t$. Since $1 \le \lceil d(T) \rceil \le T$, at each horizon we can partition the first T positions into $\lceil d(T) \rceil$ nonempty groups. The estimator may use a different partition at each horizon; the data law P remains fixed. Let $\widehat{P}_T^U$ be the smoothed pooled estimator and set $r_T = \mathbb{E}[D(P_T^U, \widehat{P}_T^U)]$. For every law R supported on $v_{1:T} \neq 0^T$, define

$$\widehat{Q}_R = e^{-r_T}(\widehat{P}_T^U \otimes \delta_{0^T}) + (1 - e^{-r_T})R.$$

Since all Bernoulli parameters are fixed at $1/2$, r_T is a deterministic function of the known sample sizes. Thus the prescribed mixture weight defines an estimator from the observed sample.

For this construction, $P_T = P_T^U \otimes \delta_{0^T}$. Since R assigns no mass to $v_{1:T} = 0^T$,

$$D(P_T, \widehat{Q}_R) = D(P_T^U, \widehat{P}_T^U) + r_T.$$

Hence every $\widehat{Q}_R$ has expected sequence KL $\kappa_n(T) = 2r_T \asymp \lceil d(T) \rceil / n \asymp d(T)/n$. Both costs vanish under the first mixture component, so

$$\mathbb{E}\left[\mathbb{E}_{\widehat{Q}_R}[c]\right] = (1 - e^{-r_T})\mathbb{E}_R[c].$$

On $v_{1:T} \neq 0^T$, terminal cost ranges from zero to one, while count cost ranges from one to T. Varying R over point masses and their mixtures gives the two intervals in Proposition 4.4.2, since $r_T = \kappa_n(T)/2$. If an infinite-sequence output law is required, extend each fitted length-T law by a fixed continuation. □

4.D.3 A sharp statistical obstruction to reverse-KL fitting

Suppose the learner must answer prompts drawn from a fixed law μ. Replaying a few training prompts would change the task: the distribution of prompts is supplied externally. This makes support recovery a conditional prediction problem.

Proposition 4.D.1 (Two candidate rules and rare distinguishing prompts)**.** *Let $\mathcal{U}$ be finite, with $\mu(u) > 0$ for every u. Two known functions $f_0, f_1 : \mathcal{U} \to \{0,1\}$ disagree on $\mathcal{A} \subseteq \mathcal{U}$, where $\mu(\mathcal{A}) = \alpha > 0$. Under candidate $i \in \{0,1\}$, a demonstration has law*

$$P_i(u, y) = \mu(u)\mathbb{I}\{y = f_i(u)\}.$$

After n independent demonstrations, a possibly randomized learner returns a conditional law $\widehat{q}$, inducing $\widehat{Q}(u, y) = \mu(u)\widehat{q}(y \mid u)$. Then

$$\inf_{\mathcal{L}} \max_{i \in \{0,1\}} \mathbb{P}_{P_i^{\otimes n}, \mathcal{L}}\big(D(\widehat{Q}, P_i) = +\infty\big) = \frac{1}{2}(1 - \alpha)^n. \tag{4.78}$$

Here $\mathcal{L}$ ranges over all such learning rules.

Proof. Finite reverse KL requires $\widehat{q}(f_i(u) \mid u) = 1$ for every u; hence it requires $\widehat{Q} = P_i$. With probability $(1-\alpha)^n$, none of the training prompts lies in $\mathcal{A}$. Conditional on this event the data have the same distribution under both candidates. No output can equal both P_0 and P_1. Averaging over the two equally likely candidates, the probability of failure on this event is therefore at least $1/2$. The averaging argument gives the lower bound on the worst-case failure probability.

For equality, use any observed prompt in $\mathcal{A}$ to identify i and return P_i. If none occurs, choose one of the two candidates with an independent fair coin. This randomized rule has the stated failure probability under each candidate. □

For $0 < \alpha < 1$ and $0 < \delta < 1/2$, success probability at least $1 - \delta$ requires

$$n \geq \frac{\log(1/(2\delta))}{-\log(1-\alpha)}.$$

To make this concrete, a prompt is a pair of m-bit integers (L, R), drawn uniformly and independently. The candidate rules are

$$f_0(L, R) = \mathbb{I}\{L \geq R\}, \qquad f_1(L, R) = \mathbb{I}\{L > R\}.$$

They differ exactly when $L = R$, an event of probability 2^{-m}. The sample requirement is consequently $\Omega(2^m \log(1/\delta))$ for $0 < \delta \leq 1/4$. The class has only two rules; random demonstrations rarely distinguish them.

For comparison, let q give the common answer off $\mathcal{A}$ and a fair bit on $\mathcal{A}$. With $Q(u, y) = \mu(u)q(y \mid u)$ and task cost $c_i(u, y) = \mathbb{I}\{y \neq f_i(u)\}$, direct calculation gives

$$D(P_i, Q) = \alpha \log 2, \qquad \mathbb{E}_{P_i}[c_i] = 0, \qquad \mathbb{E}_Q[c_i] = \frac{\alpha}{2}, \qquad D(Q, P_i) = +\infty.$$

Finite reverse KL demands exact conditional support even on very rare prompts. Forward KL and expected error can already be small before those prompts have been observed. One expert answer to a *chosen* prompt with $L = R$ identifies the rule. This specifies additional information that resolves the difficulty. Lectures 19 and 24 return to such differences between random examples and targeted queries.

4.E Reverse-KL optimization

Direct reverse-KL optimization uses target scores on generated samples. We derive its gradient when these scores are supplied, then connect a cost-defined target to KL-regularized reinforcement learning.

4.E.1 A target known up to normalization

Suppose a target law is specified as

$$P(x) = \frac{w(x)}{\mathcal{Z}}, \qquad \mathcal{Z} = \sum_x w(x) > 0,$$

where w is evaluable but $\mathcal{Z}$ is expensive to compute. For $\widetilde{X} \sim Q_\theta$, with Q_θ supported where $w > 0$,

$$D(Q_\theta, P) = \mathbb{E}[\log Q_\theta(\widetilde{X}) - \log w(\widetilde{X})] + \log \mathcal{Z}. \tag{4.79}$$

The final term is independent of θ. We can sample from Q_θ, evaluate $\log Q_\theta - \log w$ on those samples, and estimate gradients without sampling from P.

For a differentiable model with fixed support contained in that of w, the gradient is

$$\mathbb{E}\Big[\big(\log Q_\theta(\widetilde{X}) - \log w(\widetilde{X})\big)\nabla_\theta \log Q_\theta(\widetilde{X})\Big], \qquad \widetilde{X} \sim Q_\theta.$$

Here $\widetilde{X}$ denotes the whole generated sequence. Differentiating the expectation gives this identity; the additional normalization derivative vanishes because $\sum_x \nabla Q_\theta(x) = 0$. Monte Carlo averages of these gradients give stochastic optimization updates. Eliminating the normalizer preserves the objective's minimizers; it does not certify that optimization has reached a small KL.

4.E.2 A cost-defined target connects reverse KL to RL

Suppose instead that a cost $c(x)$ can be evaluated. Given a positive reference law R and $\beta > 0$, define

$$P_\beta(x) = \frac{R(x)e^{-\beta c(x)}}{\mathcal{Z}_\beta}, \qquad \mathcal{Z}_\beta = \mathbb{E}[e^{-\beta c(X)}], \quad X \sim R.$$

For $Y \sim Q$, direct substitution gives

$$D(Q, P_\beta) = D(Q, R) + \beta\mathbb{E}[c(Y)] + \log \mathcal{Z}_\beta. \tag{4.80}$$

Thus minimizing reverse KL to P_β is equivalent to minimizing $\mathbb{E}[c(Y)] + \beta^{-1}D(Q, R)$. The reference penalizes departures from an existing model, while the cost evaluates generated behavior. The supplied cost defines a new target whose relative probabilities can be evaluated. This is a useful connection to KL-regularized reinforcement learning and post-training.

4.F Recovery for history-dependent generators

Final-answer accuracy can be controlled by bounding the influence of an earlier token on the final marginal. Markov contraction gives geometric decay of this influence. Here we allow the generator to use the entire prefix and quantify the influence directly.

4.F.1 From the influence of a token to task cost

Let Q be a generator with next-token laws specified at every prefix. For a prefix $h = x_{<t}$ followed by a supplied token x, write $Q(X_s \in \cdot \mid h, x)$ for the law at position $s \geq t$ obtained by continuing this generator from (h, x). At $s = t$ this law is δ_x. Suppose that changing the supplied token changes later marginals by at most

$$\sup_{h,x,x'} \|Q(X_s \in \cdot \mid h, x) - Q(X_s \in \cdot \mid h, x')\|_{\mathrm{TV}} \leq b_{s-t}, \qquad s \geq t, \qquad b_0 = 1. \tag{4.81}$$

The coefficient $b_j \geq 0$ bounds the influence j positions later. For a Markov generator satisfying Eq. (4.49), we can take $b_j = \rho^j$.

For an arbitrary data law P, let

$$\varepsilon_t = \mathbb{E}_P[\|P(\cdot \mid X_{<t}) - Q(\cdot \mid X_{<t})\|_{\mathrm{TV}}], \qquad 1 \leq t \leq T, \tag{4.82}$$

where the prefix at $t = 1$ is empty.

Proposition 4.F.1 (Recovery with history-dependent conditionals)**.** *Under Eq. (4.81), for $c, c_t : [N] \to [0, 1]$,*

$$|\mathbb{E}_P[c(X_T)] - \mathbb{E}_Q[c(X_T)]| \leq \sum_{t=1}^{T} b_{T-t}\varepsilon_t, \tag{4.83}$$

$$\left|\mathbb{E}_P\left[\sum_{s=1}^{T} c_s(X_s)\right] - \mathbb{E}_Q\left[\sum_{s=1}^{T} c_s(X_s)\right]\right| \leq \sum_{t=1}^{T} \varepsilon_t \sum_{j=0}^{T-t} b_j. \tag{4.84}$$

In particular, if $\sum_{j\geq 0} b_j \leq B$, the cumulative-cost discrepancy is at most $B \sum_{t=1}^{T} \varepsilon_t$.

Proof. Replace the generator's conditionals by the data conditionals one at a time. Let $R^{(t)}$ use P for the first t tokens and Q thereafter:

$$R^{(t)}(x_{1:T}) = P(x_{\leq t}) \prod_{j=t+1}^{T} Q(x_j \mid x_{<j}), \qquad R^{(0)} = Q, \quad R^{(T)} = P.$$

The laws $R^{(t)}$ and $R^{(t-1)}$ have the same prefix distribution $P(X_{<t} \in \cdot)$. At a common prefix h, their next-token laws are $P(\cdot \mid h)$ and $Q(\cdot \mid h)$. A *coupling* puts two random variables with prescribed laws on one probability space. On a finite set we can couple these tokens to disagree with probability exactly their TV distance: assign probability $\min\{P(x \mid h), Q(x \mid h)\}$ to matching token x, then couple the remaining masses, whose supports are disjoint.

If the tokens agree, their Q-continuations have the same law. If they differ, Eq. (4.81) and the TV cost bound limit the difference in expected cost at position $s \geq t$ to b_{s-t}. Averaging over the coupled tokens and the data prefix gives

$$|\mathbb{E}_{R^{(t)}}[c_s(X_s)] - \mathbb{E}_{R^{(t-1)}}[c_s(X_s)]| \leq b_{s-t}\varepsilon_t.$$

For $t > s$, this difference is zero. Telescoping over t with $s = T$ proves Eq. (4.83). Summing the same bounds over s and reversing the two finite sums proves Eq. (4.84). □

An initial discrepancy contributes $b_{T-1}\varepsilon_1$ to the final-answer bound. Thus $b_j \to 0$ expresses recovery from an initial error. Summability gives a bounded total contribution to cumulative cost. With fresh errors $\varepsilon_t \leq \varepsilon$, the final-answer bound is at most $B\varepsilon$.

4.F.2 A sufficient condition on next-token laws

We can obtain the influence bounds by limiting sensitivity to each past position. Let $a_j \geq 0$ bound the effect of a token j positions back, and suppose that for every t and pair of prefixes,

$$\|Q(\cdot \mid x_{<t}) - Q(\cdot \mid y_{<t})\|_{\mathrm{TV}} \leq \sum_{j=1}^{t-1} a_j \mathbf{1}\{x_{t-j} \neq y_{t-j}\}, \qquad \sum_{j\geq 1} a_j = \rho < 1. \tag{4.85}$$

There can be nonzero coefficients at arbitrarily large lags.

Proposition 4.F.2 (Summable sensitivity gives bounded error propagation)**.** *Under Eq. (4.85), Eq. (4.81) holds with*

$$b_0 = 1, \qquad b_k = \sum_{j=1}^{k} a_j b_{k-j}, \qquad \sum_{k\geq 0} b_k = \frac{1}{1-\rho}. \tag{4.86}$$

Proof. Start two Q-continuations from the same prefix with different tokens at position t. Couple each successive pair of next-token laws to minimize disagreement, as in the previous proof. Let d_k be the probability that the tokens at position $t+k$ differ. The supplied tokens give $d_0 = 1$. The earlier prefixes agree, so Eq. (4.85) gives

$$d_k \leq \sum_{j=1}^{k} a_j d_{k-j}.$$

Induction gives $d_k \leq b_k$. The TV distance of the two marginals is at most their disagreement probability, proving Eq. (4.81). For $B_m = \sum_{k=0}^{m} b_k$, the recurrence gives $B_m \leq 1 + \rho B_m$, hence $B_m \leq 1/(1-\rho)$. The full sum is therefore finite. Summing the recurrence and reordering nonnegative terms gives

$$\sum_{k\geq 0} b_k = 1 + \rho \sum_{k\geq 0} b_k = \frac{1}{1-\rho}.$$

□

The recurrence includes influence transmitted through intervening tokens: $b_1 = a_1$, $b_2 = a_2 + a_1^2$, and so on. For first-order Markov contraction, $a_1 = \rho$ and $a_j = 0$ for $j > 1$, giving $b_k = \rho^k$.

Return to the KL question. For $P = \delta_{0^T}$, the local errors in Eq. (4.82) satisfy

$$\varepsilon_t = 1 - Q(0 \mid 0^{t-1}) \leq -\log Q(0 \mid 0^{t-1}).$$

The KL chain rule, Eq. (4.5), therefore gives $\sum_t \varepsilon_t \leq D(P, Q)$. Under Eq. (4.85), we obtain

$$\mathbb{E}_Q[c_{\mathrm{fin}}] \leq \sum_{t=1}^{T} b_{T-t}[-\log Q(0 \mid 0^{t-1})], \qquad \mathbb{E}_Q[c_{\mathrm{cnt}}] \leq \frac{D(P,Q)}{1-\rho}. \tag{4.87}$$

For the final answer, recovery weights the local log losses by their remaining influence. For the error count, a fixed contraction margin removes the horizon factor in the unrestricted example $\mathbb{E}_{Q_+}[c_{\mathrm{cnt}}] = T(1 - e^{-\kappa})$.

4.F.3 Weak dependence at each step can preserve an old error

A small effect on each prediction can still accumulate if the same old error keeps affecting future predictions.

Example 4.F.3 (Repeated use of the first token)**.** Let $P = \delta_{0^T}$, $T \geq 2$, $0 < \rho < 1$, and

$$Q(X_1 = 1) = 1 - e^{-\kappa}, \qquad Q(X_t = 1 \mid X_{<t}) = \rho X_1, \quad t \geq 2.$$

At every time $t \geq 2$, any two next-token laws differ in TV by at most ρ. Nevertheless,

$$D(P, Q) = \kappa, \qquad \mathbb{E}_Q[c_{\text{fin}}] = \rho(1 - e^{-\kappa}), \qquad \mathbb{E}_Q[c_{\text{cnt}}] = (1 - e^{-\kappa})[1 + \rho(T - 1)].$$

Indeed, the all-zero path has probability $e^{-\kappa}$, and $Q(X_t = 1) = \rho Q(X_1 = 1)$ at every $t \geq 2$. Changing the first token keeps changing all later marginals by ρ, so Eq. (4.81) requires $b_j \geq \rho$ for every $j \geq 1$ when the model is continued indefinitely. The influences are not summable: the initial error keeps causing mistakes instead of being forgotten.

References

Léon Bottou and Olivier Bousquet. The Tradeoffs of Large Scale Learning. *Advances in Neural Information Processing Systems*, 20, 2007. https://leon.bottou.org/publications/pdf/nips-2007.pdf.

Léon Bottou, Frank E. Curtis, and Jorge Nocedal. Optimization Methods for Large-Scale Machine Learning. *SIAM Review*, 60(2):223–311, 2018. https://arxiv.org/abs/1606.04838.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. Section 9.1.2. https://web.stanford.edu/~boyd/cvxbook/.

Paul Doukhan and Olivier Wintenberger. Weakly dependent chains with infinite memory. arXiv:0712.3231, 2007. https://arxiv.org/abs/0712.3231.

Andrew Drucker and Ronald de Wolf. Uniform Approximation by (Quantum) Polynomials. *Quantum Information and Computation*, 11(3&4):215–225, 2011. https://arxiv.org/abs/1008.1599.

Dylan J. Foster, Adam Block, and Dipendra Misra. Is Behavior Cloning All You Need? Understanding Horizon in Imitation Learning. *Advances in Neural Information Processing Systems*, 2024. https://arxiv.org/abs/2407.15007.

Stéphane Gaubert and Zheng Qu. Dobrushin's ergodicity coefficient for Markov operators on cones. *Integral Equations and Operator Theory*, 81(1):127–150, 2015. https://arxiv.org/abs/1307.4649.

Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Rozière, David Lopez-Paz, and Gabriel Synnaeve. Better & Faster Large Language Models via Multi-token Prediction. *Proceedings of the 41st International Conference on Machine Learning*, PMLR 235:15706–15734, 2024. https://proceedings.mlr.press/v235/gloeckle24a.html.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge Distillation of Large Language Models. *International Conference on Learning Representations*, 2024. https://proceedings.iclr.cc/paper_files/paper/2024/hash/8ac015d409635f196f9e3e9dcfb9a94e-Abstract-Conference.html.

András György, Tor Lattimore, Nevena Lazić, and Csaba Szepesvári. Beyond Statistical Learning: Exact Learning Is Essential for General Intelligence. arXiv:2506.23908, 2025. https://arxiv.org/abs/2506.23908.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network. arXiv:1503.02531, 2015. https://arxiv.org/abs/1503.02531.

Jordan Hoffmann et al. Training Compute-Optimal Large Language Models. arXiv:2203.15556, 2022. https://arxiv.org/abs/2203.15556.

Jared Kaplan et al. Scaling Laws for Neural Language Models. arXiv:2001.08361, 2020. https://arxiv.org/abs/2001.08361.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization. *Advances in Neural Information Processing Systems*, 29, 2016. https://arxiv.org/abs/1606.00709.

Ian Osband. Planning to Learn. arXiv:2610.03667, 2026. https://arxiv.org/abs/2610.03667.

Maxim Raginsky and Igal Sason. *Concentration of Measure Inequalities in Information Theory, Communications and Coding*. Second edition, 2014. https://arxiv.org/abs/1212.4663.

Daniel Rudolf and Nikolaus Schweizer. Perturbation theory for Markov chains via Wasserstein distance. *Bernoulli*, 24(4A):2610–2639, 2018. https://doi.org/10.3150/17-BEJ938.

Eckhard Schlemm. The Kearns–Saul inequality for Bernoulli and Poisson-binomial distributions. arXiv:1405.4496, 2014. https://arxiv.org/abs/1405.4496.

Kaiwen Wang, Nathan Kallus, and Wen Sun. The Central Role of the Loss Function in Reinforcement Learning. arXiv:2409.12799, 2025. https://arxiv.org/abs/2409.12799.

Bin Yu. Assouad, Fano, and Le Cam. In David Pollard, Erik Torgersen, and Grace L. Yang, editors, *Festschrift for Lucien Le Cam: Research Papers in Probability and Statistics*, pages 423–435. Springer, 1997. https://doi.org/10.1007/978-1-4612-1880-7_29.

Hugging Face. Fixing gradient accumulation. October 16, 2024. https://huggingface.co/blog/gradient_accumulation.

Yongqiang Yao, Jingru Tan, Kaihuan Liang, Feizhao Zhang, Jiahao Hu, Shuo Wu, Yazhe Niu, Ruihao Gong, Dahua Lin, and Ningyi Xu. Hierarchical balance packing: Towards efficient supervised fine-tuning for long-context LLM. NeurIPS, 2025. arXiv:2503.07680v3. https://arxiv.org/abs/2503.07680v3.