Probability Review
Optimization Problem-Solving

From ground truth to observed data, and back to convex objectives

CSCI 567 — Machine Learning • Discussion 03 • Sep 11, 2026

Outline

1. Probability review: probability spaces, random variables, conditioning, independence, expectation/variance, Gaussians
2. Optimization problem-solving: convexity checks, and deriving Lecture 03's ridge gradient/closed form step by step

Groundwork for Lecture 03's MLE/MAP derivations (Gaussian noise, Bayes' rule as the engine behind MAP) and hands-on practice with its regularized objectives.

1. Probability review

From ground truth to observed data

Probability vs. statistics: from the data-generating process to data, and back

Probability studies the behavior of **data-generating processes**. First, assume complete knowledge of how data is generated. Then, what can we say about:

- probabilities of interesting events?
- the effect of "conditioning" on an event?
- sequences of events?
- dependence / independence?

TLDR: probability takes you from "ground truth" to observed data. Statistics takes you the *other* way!

Probability spaces and random variables: two definitions not to confuse

Probability space

set Ω of events that can take place, and probabilities for each event $A \subseteq \Omega$ — random events "in the wild" (a coin flip) and nature's rules for their chances

Random variable

function $\Omega \to \mathbb{R}$, assigning a number to each $\omega \in \Omega$ — our choice of "value" for each outcome ("Heads" $\to$ 1), which lets us use math

Formally, a probability space is a pair (Ω, P): Ω a non-empty set, $P : 2^{\Omega} \to \mathbb{R}$, such that

1. $P(\Omega) = 1$ 2. if $\{A_i\}_{i \in \mathbb{N}}$ are disjoint, then $P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i)$

Exercise: probability identities

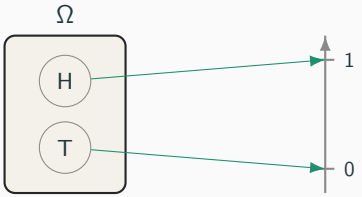
For events A, B and C, which of the following identities are correct?

(a) $P(A) - P(A \cap B) = P(A \cup B) - P(B)$
(b) $P(A \cup B) \leq P(A) + P(B) - P(A)P(B)$
(c) $P(A) = P(A \cap C) + P(A \cap \bar{C})$, where $\bar{C}$ is the complement of C

(a) and (c) are correct. (a): $P(A \cup B) = P(A) + P(B) - P(A \cap B)$, rearrange. (c): law of total probability, partitioning A by $C/\bar{C}$. (b) is **false** *in general: take a fair die, $A = \{1\}$, $B = \{2\}$ (disjoint) — $P(A \cup B) = 1/3 \approx 0.333$, but $P(A) + P(B) - P(A)P(B) = 11/36 \approx 0.306 < 1/3$.*

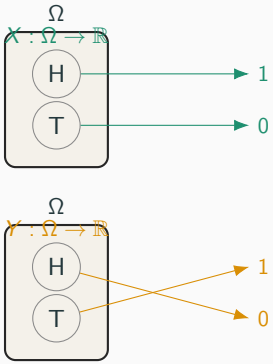
A random variable is a function $\Omega \to \mathbb{R}$; its distribution is the pushforward of P

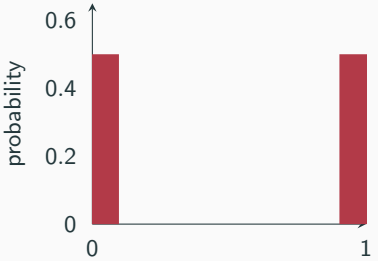
A random variable (RV) on a probability space (Ω, P) is a function $X : \Omega \to \mathbb{R}$. It produces probabilities over $\mathbb{R}$ via $P_X(A) = P(X \in A) = P(X^{-1}(A))$, the **distribution** of X.



Beware: very different RVs can have the same distribution! The RV contains more information than just the distribution...

Two different random variables, one distribution





$\Omega = \{H, T\}$, a fair coin. X and Y are literally opposite functions ($X = 1 - Y$) — yet both have the *identical* distribution, $P(\cdot = 0) = P(\cdot = 1) = 0.5$.

Conditioning

For events A, B with $P(B) \neq 0$, we define the **conditional probability**

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}.$$

Clearly, $P(A \mid B) \cdot P(B) = P(A \cap B)$, by multiplying out $P(B)$.

this immediately yields **Bayes' rule**:

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

simple to prove, but very valuable!

Independence

Two events A, B are **independent** if

$$P(A \cap B) = P(A) \cdot P(B).$$

In this case, $P(A \mid B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A) \cdot P(B)}{P(B)} = P(A)$.

similarly, two random variables X and Y are independent if $P(X{=}x \wedge Y{=}y) = P(X{=}x) \cdot P(Y{=}y)$, equivalently $P(X{=}x \mid Y{=}y) = P(X{=}x)$

Exercise: a bag of balls

A bag contains 2 red balls and 3 blue balls. First, Alice draws a ball from the bag randomly (and removes it). Then, Bob draws a ball randomly too.

1) What is the probability that Alice gets a red ball *and* Bob gets a blue ball?

2) What is the probability that Alice gets a blue ball *given that* Bob gets a blue ball?

1) $P(\textit{Alice R, Bob B}) = \frac{2}{5} \cdot \frac{3}{4} = \frac{3}{10}$. *2)* $P(\textit{Alice B} \mid \textit{Bob B}) = \dfrac{P(\textit{both B})}{P(\textit{Bob B})} = \dfrac{\frac{3}{5} \cdot \frac{2}{4}}{\frac{2}{5} \cdot \frac{3}{4} + \frac{3}{5} \cdot \frac{2}{4}} = \dfrac{3/10}{3/5} = \frac{1}{2}$.

Exercise: more identities

For events A, B, C and $Z_1, \ldots, Z_T$, which of the following identities are correct?

(a) $P(A \mid B) = \dfrac{P(B \mid A)P(A)}{P(B)}$

(b) $\dfrac{P(A \mid B, C)}{P(A \mid C)} = \dfrac{P(B \mid A, C)}{P(B \mid C)}$

(c) $P\big(\bigcap_{t=1}^{T} Z_t\big) = \prod_{t=1}^{T} P(Z_t)$

(d) $P\big(\bigcap_{t=1}^{T} Z_t\big) = \prod_{t=1}^{T} P(Z_t \mid Z_1, \ldots, Z_{t-1})$

(a), (b), (d) are correct: (a) is Bayes' rule; (b) is Bayes' rule applied conditionally throughout on C; (d) is the general **chain rule of probability**, *always true. (c) is* **false** *in general — it requires the Z_t's to be mutually independent, which is not assumed.*

Exercise: why not evaluate on the training set?

Q: When training a machine learning model, why isn't it a good idea to evaluate a model's performance on the training set? After all, for any given predictor f, its error on a dataset S is an unbiased estimate of its true error?

A: it stops being unbiased once we *condition* on the fact that f was *trained on* S!

E.g., imagine 10,000 people flip a fair coin 10 times. On average, each person's performance is reflective of their true (50%) performance — but the $\sim$10 people who flipped all heads are unlikely to repeat this.

TLDR: you must condition on all available information! Not allowed to cherry-pick and ignore (un)favorable information.

Understanding train/test split

How should we evaluate the performance of a trained model?

key idea: use a separate, "fresh" **test** dataset!

Workflow:

1. Sample all data, S
2. Uniformly at random, split S into S_{train} and S_{test} (say, 80%:20%)
3. Train by minimizing loss on S_{train}:
   $\hat{f} \approx \arg\min_{f \in F} \hat{R}_{S_{\text{train}}}(f) = \arg\min_{f \in F} \frac{1}{n} \sum_{i \leq n} \ell(f(x_i), y_i)$
4. Evaluate via performance on S_{test}: $\hat{R}_{S_{\text{test}}}(f) = \sum_{j \leq m} \ell(f(x_j), y_j)$

Describing a random variable: PMF or PDF, then expectation and variance

A **discrete** RV puts all its mass on a countable set, and is described by its **probability mass function** $p_X(x) = P(X = x)$; a **continuous** RV by a **probability density function** f_X, with $P(X \in [a, b]) = \int_a^b f_X(x)\,dx$.

Two numbers summarize most of what we need:

expectation (mean) of X: $\mathbb{E}[X] = \sum_x x \cdot p_X(x)$, or $\mathbb{E}[X] = \int_x x \cdot f_X(x)\,dx$ — the average value, a measure of "center"

variance of X with mean μ: $\mathrm{Var}[X] = \mathbb{E}[(X - \mu)^2] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$ — the "dispersion," how greatly X varies

Properties of expectation and variance

Linearity of expectation

1. $\mathbb{E}[X + Y] = \mathbb{E}[X] + \mathbb{E}[Y]$ — holds for *any* X, Y, regardless of dependence!

2. $\mathbb{E}[cX] = c\,\mathbb{E}[X]$ — can always pull out constants

Variance properties

1. $\mathrm{Var}[X + Y] = \mathrm{Var}[X] + \mathrm{Var}[Y]$, if X, Y *independent* — can fail without independence!

2. $\mathrm{Var}[cX] = c^2\mathrm{Var}[X]$ — variance uses "squared units," square root to recover std. deviation

Gaussians, one- and multi-dimensional: affine functions of standard normals

One-dimensional: $X \sim \mathcal{N}(\mu, \sigma^2)$, $f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$, equivalently $X = aZ + b$ with $Z \sim \mathcal{N}(0,1)$.

A random vector $X \in \mathbb{R}^d$ is **multivariate Gaussian** if

$$X = AZ + \mu, \qquad Z \sim \mathcal{N}(0, I_\ell),$$

$Z \in \mathbb{R}^\ell$ a vector of ℓ independent standard normals, $A \in \mathbb{R}^{d \times \ell}$, $\mu \in \mathbb{R}^d$.

key idea: Gaussian = affine function of a standard normal. Z is a spherical cloud, so every multivariate Gaussian is a **stretched, rotated, shifted sphere**

Joint Gaussians

Random vectors X and Y are **jointly Gaussian** if $\binom{X}{Y}$ is distributed as a multivariate Gaussian, i.e. $\binom{X}{Y} = AZ + \mu$.

key characterization: for random variables $X_1, \ldots, X_n$, the vector $(X_1, \ldots, X_n)$ is multivariate Gaussian if and only if $\sum_{i=1}^{n} a_i X_i$ is Gaussian for *all* $a \in \mathbb{R}^n$

this is **stronger** than X and Y each being distributed Gaussian!

Exercise: jointly Gaussian, true or false?

Which of the following statements are true?

(a) Suppose X, Y are jointly Gaussian. Then $Z = X - 2Y$ is also Gaussian.

(b) Suppose X, Y are jointly Gaussian. Then the marginal distribution of X is also Gaussian.

(c) Suppose X, Y are jointly Gaussian. Then the conditional distribution of X given Y is also Gaussian.

All three are true. *(a): the key characterization directly — $a = (1, -2)$. (b): set $a = (1, 0)$, so X itself is a valid linear combination. (c): a further standard fact about jointly Gaussian vectors — conditionals are Gaussian too, with a linear mean and constant covariance. All three are genuinely stronger claims than "X and Y are each marginally Gaussian," the point of the warning on the previous slide.*

Exercise: the spam classifier

Suppose your spam classifier gives the guarantee that (1) if an email is spam, it marks it as spam with probability 90%, (2) if an email is *not* spam, it marks it as spam with probability only 10%. Suppose 1% of all emails are spam. If the classifier marks a certain email as spam, what is the probability it is actually spam?

$$P(\text{spam} \mid \text{marked}) = \frac{(0.9)(0.01)}{(0.9)(0.01) + (0.1)(0.99)} = \frac{0.009}{0.108} = \frac{1}{12} \approx \mathbf{8.33\%}$$

Same base-rate lesson as the bag-of-balls and Discussion 01's disease classifier — a 90%/90%-accurate-sounding test, on a rare event, is barely better than a coin flip once you actually apply Bayes' rule.

How today's probability feeds Lecture 03: Gaussian noise, Bayes' rule, test error

the noise model $\varepsilon \sim \mathcal{N}(0, \mathrm{Var}[\varepsilon])$ behind Lecture 03's bias-variance decomposition, and the Gaussian likelihood in the MLE view of least squares and the MAP view of Ridge, are exactly today's 1-D Gaussian

Bayes' rule is the mechanism behind MAP: $P(\mathbf{w} \mid X, \mathbf{y}) \propto P(\mathbf{y} \mid X, \mathbf{w})\, P(\mathbf{w})$ — likelihood times prior, exactly today's rule applied to parameters instead of events

the train/test split exercise is exactly why this course always reports test error, never training error, as the estimate of true risk

Notation heads-up: Lecture 03 writes noise variance as $\mathrm{Var}[\varepsilon]$, not σ^2 — σ already means the *sigmoid function* $\sigma(z)$ (Lecture 02).

References

- Content this section: Julian Asilis, CSCI 567 Discussion 3 – Probability Review (Spring 2026), taught in Vatsal Sharan's course, used with permission.
- The opening "Probability vs. Statistics" framing cites: Drgoňa, Ján (2017). *Model Predictive Control with Applications in Building Thermal Comfort Control.*

2. Optimization problem-solving

Convexity checks, then Ridge from scratch

Why convexity matters, recapped

Lecture 02's payoff: if g is convex and $\nabla g(w^\star) = 0$, then $w^\star$ is a **global** minimum — the reason gradient descent's stopping point can be trusted

Second-order test: g is convex $\iff$ its Hessian $\nabla^2 g(w)$ is **PSD** everywhere — exactly Discussion 02's PSD test, applied to a Hessian.

Convexity check, worked

$g(w_1, w_2) = 3w_1^2 + 2w_2^2 - 2w_1 w_2$. Hessian:

$$\nabla^2 g = \begin{pmatrix} 6 & -2 \\ -2 & 4 \end{pmatrix}$$

Characteristic equation: $(6-\lambda)(4-\lambda) - 4 = 0 \Rightarrow \lambda^2 - 10\lambda + 20 = 0 \Rightarrow \lambda = 5 \pm \sqrt{5}$.

eigenvalues $\approx \{2.76, 7.24\}$,
both positive $\Rightarrow$ PSD $\Rightarrow$ **convex**

Convexity trap, worked

$$g(w) = w^3. \qquad g''(w) = 6w.$$

at $w = -1$: $g''(-1) = -6 < 0$ — **not** PSD everywhere, so g is **not convex** (even though g restricted to $w \geq 0$ is convex)

lesson: convexity is a *global* property — checking the Hessian's sign on part of the domain is not enough, it must hold *everywhere*

Convexity is preserved under (non-negative) addition

If $f, g : \mathbb{R}^n \to \mathbb{R}$ are convex and $a, b \geq 0$, then $af + bg$ is convex.

Proof (Hessian form): $\nabla^2(af + bg) = a\nabla^2 f + b\nabla^2 g$. For any x: $x^\top(a\nabla^2 f + b\nabla^2 g)x = a\underbrace{x^\top \nabla^2 f\, x}_{\geq 0} + b\underbrace{x^\top \nabla^2 g\, x}_{\geq 0} \geq 0$ — the same PSD argument as Discussion 02's eigenvalue test, applied to a sum; the non-negativity of a, b is what makes it work (with $a = 1$, $b = -1$ it fails)

this is exactly why Ridge is convex: MSE (convex, Lecture 02) + $\frac{\lambda}{2}\|\mathbf{w}\|_2^2$ (convex, Hessian $\lambda I \succeq 0$) — Lecture 03 uses precisely this rule

Deriving Ridge's gradient, step by step

Start from Lecture 03's objective, coordinate by coordinate:

$$\hat{\mathcal{R}}_\lambda(\mathbf{w}) = \frac{1}{2n}\sum_{i=1}^{n}(\mathbf{w}^\top \mathbf{x}_i - \mathbf{y}_i)^2 + \frac{\lambda}{2}\sum_j w_j^2$$

Differentiate w.r.t. w_k, term by term:

$$\frac{\partial}{\partial w_k}\left[\frac{1}{2n}\sum_i(\mathbf{w}^\top \mathbf{x}_i - \mathbf{y}_i)^2\right] = \frac{1}{n}\sum_i(\mathbf{w}^\top \mathbf{x}_i - \mathbf{y}_i)\, x_{i,k}, \qquad \frac{\partial}{\partial w_k}\left[\frac{\lambda}{2}w_k^2\right] = \lambda w_k$$

Sum, vectorized

$$\nabla\hat{\mathcal{R}}_\lambda(\mathbf{w}) = \frac{1}{n}X^\top(X\mathbf{w} - \mathbf{y}) + \lambda\mathbf{w}$$

First term: identical to Lecture 02's plain-regression gradient. Second term: new, pulls *toward the origin* proportionally to $\mathbf{w}$ itself.

Ridge's closed form, worked numerically

Setting the gradient to zero (Lecture 03) gives $\mathbf{w}^\star_{\text{ridge}} = (X^\top X + n\lambda I)^{-1} X^\top \mathbf{y}$. Apply it to Lecture 02's own dataset: $x = (1, 2, 4)$, $y = (2, 4, 8)$ (no bias, scalar w, n=3), with $\lambda = 0.1$.

$$X^\top X = 1^2 + 2^2 + 4^2 = 21, \qquad X^\top y = 1(2) + 2(4) + 4(8) = 42$$

$$w^\star_{\text{plain}} = 42/21 = 2$$

$$w^\star_{\text{ridge}} = \frac{42}{21 + 3(0.1)} = \frac{42}{21.3} \approx \mathbf{1.972}$$

(This data lies exactly on $y = 2x$, so the plain fit is exact — $w^\star_{\text{plain}} = 2$.)

Reading it

Ridge pulls the perfect fit (w=2) down slightly toward 0 — exactly the shrinkage Lecture 03 promised, small here because λ is small relative to the data's scale.

GD on the ridge objective, worked

Same data, $w_0 = 0$, $\eta = 0.05$, $\lambda = 0.1$. Compare to Lecture 02's *unregularized* trace step by step.

Step 1

$$\nabla\hat{\mathcal{R}}_\lambda(0) = \underbrace{-14}_{\text{plain (Lecture 02)}} + \underbrace{0.1(0)}_{=0} = -14 \;\Rightarrow\; w_1 = 0 - 0.05(-14) = 0.7 \quad \text{(identical!)}$$

Step 2

$$\nabla\hat{\mathcal{R}}_\lambda(0.7) = \underbrace{-9.1}_{\text{plain (Lecture 02)}} + \underbrace{0.1(0.7)}_{=0.07} = -9.03 \;\Rightarrow\; w_2 = 0.7 - 0.05(-9.03) = 1.1515$$

Lecture 02's unregularized $w_2 = 1.155$ vs. here, 1.1515 — identical at step 1 (penalty gradient is $\lambda w = 0$ when $w = 0$), diverging from step 2 on as $\mathbf{w}$ moves away from the origin and the penalty starts pulling back.

SGD on the ridge objective, worked

Same data/hyperparameters, step 1 from $w_0 = 0$. SGD draws one example i uniformly and updates $w \leftarrow w - \eta\big[(wx_i - y_i)x_i + \lambda w\big]$ — unbiased for the full gradient, since $\mathbb{E}_i[(wx_i - y_i)x_i]$ is exactly the data-term average.

draw i	(x_i, y_i)	SGD gradient	w_1
1	$(1, 2)$	-2	0.1
2	$(2, 4)$	-8	0.4
3	$(4, 8)$	-32	1.6

average of the 3 possible SGD gradients: $(-2 - 8 - 32)/3 = \mathbf{-14}$, exactly the full-batch gradient; average of the 3 possible w_1's: $(0.1 + 0.4 + 1.6)/3 = \mathbf{0.7}$, exactly the full-batch GD step. **Unbiased in expectation, noisy on any single draw** — Lecture 02's SGD claim, checked on a concrete example

Wrap-up

- Probability review (probability spaces, conditioning, Bayes' rule, independence, expectation/variance, Gaussians) is exactly the machinery Lecture 03's MLE/MAP section runs on
- Convexity checks: verify PSD Hessian *everywhere*, not on part of the domain; sums of convex functions (non-negatively weighted) stay convex — why Ridge is convex
- Ridge's gradient and closed form, derived slowly and checked against concrete numeric examples built on Lecture 02's own dataset; SGD verified unbiased on the same example

Coming up

HW1 (Lectures 02–03) is due Mon 9/14, 11:59pm. Lecture 04 (Kernel Methods, Double Descent, Implicit Regularization) is next.