CS109 Lecture 24 Worksheet: Diffusion

Learning to Denoise, Conditional Sampling, and Turning Text into Images

Autumn 2026

Lecture 24 Worksheet: Diffusion

Topics: images as random vectors, adding Gaussian noise, creating supervised training targets, estimates versus samples, and using text to guide image generation.

How to use this sheet. Work in small groups. A TA and the instructor will circulate to help. We will go over selected solutions on the board. You do not need to finish every problem. Focus on explaining why each step works, rather than memorizing a particular model architecture.

Notation (matching the slides). x_0 is a clean image written as a vector; x_t is its noisy version at noise level t. We construct $x_t = a_t x_0 + b_t \epsilon$, where the entries of ϵ are independent $N(0,1)$ draws. The **schedule** specifies the weights a_t, b_t at each noise level, with $a_t^2 + b_t^2 = 1$. The network predicts $\hat{\epsilon}_\theta(x_t, t, c)$, where c is the caption's numerical representation and θ denotes the learned weights.

Problem 1 (Review of Lecture 23): What Does the Network Learn?

Last lecture, a neural network predicted the probability of a binary label. Today, a network predicts the noise added to an image.

(a) For the binary classifier, why was a sigmoid appropriate at the output? Would requiring every predicted noise value to lie in $(0, 1)$ make sense? Explain.

(b) Which changes when we train the denoising network: its weights, its training inputs, or both? Which changes while we generate one new image with the trained network?

(c) The training loss is a sum of squared prediction errors. Do we use gradient ascent or gradient descent to minimize it? How does this relate to maximizing a log-likelihood?

Problem 2: Make a Noisy Image

(a) A 2×2 RGB image has three color-channel values per pixel. How many entries does its image vector have? Is x_t one pixel or the entire noisy image?

(b) For a tiny two-coordinate image, let

$$x_0 = [1, 0], \qquad \epsilon = [-1, 2], \qquad a_t = 0.8, \qquad b_t = 0.6.$$

Compute x_t. Check that the two weights satisfy the schedule's constraint.

(c) One entry in your noisy image lies outside $[0, 1]$, even though the clean entries were in that range. Is this a mistake? Explain using the support of Gaussian noise.

(d) Describe what the image looks like when a_t is near 1 and b_t is near 0, and when these roles reverse. Does "nearly pure noise" mean every image becomes the *same array of numbers*?

Problem 3: Where Do the Training Labels Come From?

Use the training example in Problem 2 and the caption "a tiny two-coordinate image."

(a) List the three inputs to the denoising network. What is the target output? Which quantity must be kept out of the network's inputs to avoid simply giving away the answer?

(b) Suppose the network predicts $\hat{\epsilon} = [-0.5, 1.5]$. Compute the loss

$$L = \sum_j (\epsilon_j - \hat{\epsilon}_j)^2.$$

(c) We train again using the same clean image but freshly sampled noise and a different noise level. Do we need a human to label this new exercise? Explain why one image can provide many training exercises.

(d) A classmate says, "During training, the network sees the clean image and learns to add noise." Correct this description of the denoising network's task.

Problem 4: An Estimate Is Not a Sample

Consider a toy world with only two possible clean images, each with two pixel values:

$$A = [0, 1], \qquad B = [1, 0].$$

Here 0 is black and 1 is white. After observing a very noisy image, suppose A and B each have conditional probability 1/2. In this problem, we predict the **clean image directly**. Error is the sum of squared errors across the two pixels.

(a) Describe A and B in words. What does their average $M = (A + B)/2$ look like? Is M a possible clean image in this toy world?

(b) If we always predict A, compute the error when the original was A and when it was B. What is the expected error?

(c) Repeat part (b) for the prediction M. Which prediction has smaller expected squared error?

(d) If we instead want a random sample from the conditional distribution of clean images, what should we return, and with what probabilities?

(e) Our diffusion network predicts noise, not a clean image directly. Explain why the network's estimate and the full image-generation procedure still have different roles.

Problem 5: One Model, Many Images

We fix a trained model, a sampling procedure, and the prompt "a frog walking on stilts across a sandy desert." Two runs produce different images, both matching the prompt.

(a) What randomness can account for the difference? Must the network's weights have changed?

(b) Explain $p_\theta(\text{image} \mid \text{text})$ in words. Which parts of the scene are specified by this prompt, and which details are left open? Give two examples of each.

(c) During generation, do we know the exact noise that should be subtracted from the current state? Explain how this differs from the training exercises we create ourselves.

(d) To investigate how the random seed changes the result, what should we hold fixed? What does a seed control?

Problem 6: How Does the Prompt Become One Scene?

The prompt has four useful pieces: **frog**, **walking**, **on stilts**, and **across a sandy desert**.

(a) For each piece, describe one requirement it places on the image. Why would a picture containing a frog, loose stilts, and sand fail to fully satisfy the prompt?

(b) CLIP learns from paired images and captions. What should be true of the similarity between a red baseball cap's image vector and the text vectors for "a red baseball cap" and "a blue bicycle"? Does this require a dedicated "red" coordinate in the vector?

(c) Explain attention in two sentences, using the frog prompt. Must each phrase be permanently assigned to exactly one image region?

(d) Is the caption used only once, to choose the initial noise, or does it influence repeated denoising predictions? Explain how the text encoder, denoising network, and sampler contribute different parts of the process.

Problem 7: New Combinations and Imperfect Results

Answer each in one or two sentences.

(a) Must training include the exact scene "a cicada eating a BLT on a catamaran in the desert" before the model can produce a plausible image of it? What might it still get wrong?

(b) Replace "catamaran" with the invented word "flibflooringdingdoo," without defining it. Why does this fail to specify a clear visual object, even if the model still generates an image?

(c) The whippet video looks convincing but gives one dog an extra leg to hold the tiller. What does this reveal about the difference between visual plausibility and physical consistency? What additional challenge does video introduce?

(d) In latent diffusion, distinguish the **image latent** from the **text representation**. What does the image decoder do at the end?

Challenge Problem (optional): Why Squared Error?

Let $u = (x_t, t, c)$ collect the network's inputs. As a prediction model for a d-coordinate training target, assume the coordinates are conditionally independent with

$$\epsilon_j \mid u \sim N\big(\hat{\epsilon}_{\theta,j}(u), \tau^2\big)\,,$$

where $\tau^2 > 0$ is fixed. This is a modeling assumption about prediction errors, not the forward process's noise variance.

(a) Write the log-likelihood of one observed target vector ϵ under this model. Show that maximizing it over θ is equivalent to minimizing the sum of squared prediction errors.

(b) For a scalar random target Z with finite variance and mean μ, use

$$E[(Z-m)^2] = \mathrm{Var}(Z) + (\mu - m)^2$$

to explain why the best fixed squared-error prediction is $m = \mu$.

(c) Does making that best prediction, by itself, draw a random sample from the distribution of Z? Connect your answer to Problem 4.