Generative Modeling: Lecture Notes

---

1 Preliminary Information

Project: Critical analysis of a selected research article.
Course website: *Mathurin Massias page*

Warning

There will be preliminary tests at the beginning of the course as well as handwritten exercises to be completed at home.

2 Lecture 1: Introduction to Generative Modeling

2.1 Supervised Learning

In the supervised learning paradigm, one is provided with a dataset consisting of n independent and identically distributed pairs

$$(X_i, Y_i)_{i=1}^n,$$

where X_i denotes the input data and Y_i the corresponding label.

Goal

The primary objective is to estimate a predictive function that maps input data X to its associated output Y.

A canonical illustration is image classification: given a collection of labeled images of cats and dogs, the task is to construct a classifier capable of distinguishing between the two categories.

This framework is fundamentally distinct from generative modeling, where the focus is not prediction but rather the generation of new synthetic samples that resemble the observed data.

2.2 Generative Modeling (101)

Tip

In generative modeling, the dataset consists solely of unlabelled observations $\{X_i\}_{i=1}^n$. The aim is to construct a probabilistic model capable of generating new data points that are statistically indistinguishable from the original ones.

Goal

Learn an approximation of the underlying data distribution p_{data} and generate new samples from it.

Example

- Consider data sampled from a Gaussian distribution. Even if only a small set of observations is available, the task is to generate new samples consistent with the Gaussian law governing the data.
- This setting parallels large-scale language modeling: given a vast corpus of text, the objective is to generate coherent new sentences that resemble the distribution of natural language.

2.3 Generative Models (201): Class-Conditional Models

Suppose now that the dataset consists of labeled pairs $(X_i, Y_i)_{i=1}^n$.

Goal

Construct a generative model that, conditioned on a specified class label, can produce new data points belonging to that class (e.g., images of cats or images of dogs).

Formally, samples are assumed to follow $X_i \sim p_{\text{data}}(\cdot \mid Y_i)$. A natural strategy is to partition the dataset according to labels and learn a distinct generative model for each class.

2.4 Generative Models (301): Text-Conditioned Models

A more advanced scenario arises when each data point is paired with textual information, such as captions for images. A prominent example is the LAION-5B dataset, comprising over 5.6 billion image–text pairs.

Goal

Learn a conditional generative model $x^{\text{new}} \sim p_{\text{data}}(\cdot \mid \text{caption})$ that synthesizes new images consistent with arbitrary textual prompts.

This setting is considerably more challenging, as the conditioning variable (the prompt) can be arbitrary. Such models underpin text-to-image systems including *DALL-E*, *Imagen*, and *Stable Diffusion*. In the domain of text generation, one analogously considers $p_{\text{data}}(\text{next word} \mid \text{text so far})$.

2.5 Overview of Generative Models

The general methodology is to approximate p_{data} with a model distribution p_θ, from which new samples can be drawn.

Recent advances—particularly the advent of diffusion models in 2021 with the introduction of *Stable Diffusion*—have revolutionized the field, enabling applications such as text-to-image generation, video synthesis (e.g., VEO), and diffusion-based large language models.

Beyond creative media, generative modeling is increasingly relevant for scientific domains including molecular design, protein folding (*RF-Diffusion*), materials science, and inverse problems in imaging and data assimilation.

3 Maximum Likelihood Estimation

3.1 General Principle

Let $x_1, \dots, x_n \sim p_{\text{data}}$ denote training samples and $x_1^{\text{new}}, \dots, x_m^{\text{new}} \sim p_\theta$ denote samples generated by a parametric model p_θ.

Goal

Estimate parameters θ such that p_θ is as close as possible to p_{data}, according to a suitable statistical discrepancy $D(\cdot, \cdot)$.

This leads to the optimization problem

$$\hat{\theta} = \arg\min_\theta D(p_{\text{data}}, p_\theta).$$

A common choice is the maximum likelihood principle: given the independence assumption, the likelihood of the training data under the model is

$$L(\theta) = p(x_1, \dots, x_n \mid \theta) = \prod_{i=1}^n p(x_i \mid \theta).$$

The log-likelihood is then

$$\ell(\theta) = \sum_{i=1}^n \log p(x_i \mid \theta).$$

Tip

Maximum likelihood estimation tends to overfit: the degenerate solution is a Dirac distribution concentrated on the training data. Practical models aim at obtaining smoother distributions that generalize to unseen samples.

3.2 Example: Bernoulli Distribution

Consider binary data $x_i \in \{0, 1\}$, modeled as Bernoulli(θ) with probability mass function

$$p(x \mid \theta) = \theta^x (1-\theta)^{1-x}, \quad \theta \in [0, 1].$$

Exercise 3.1 (MLE for the Bernoulli Model)**.** Compute the MLE given $(x_1, \dots, x_n)$.

$$\ell(\theta) = \sum_{i=1}^n \log p(x_i \mid \theta) = \sum_{i=1}^n \left[x_i \log \theta + (1 - x_i) \log(1 - \theta)\right],$$
$$\hat{\theta} = \arg\max_\theta \ \ell(\theta).$$

Derivative (concave in θ):

$$\nabla \ell(\theta) = \sum_{i=1}^n \left(\frac{x_i}{\theta} - \frac{1 - x_i}{1 - \theta}\right) = 0 \iff \sum_{i=1}^n x_i - n\theta = 0 \iff \hat{\theta} = \frac{1}{n} \sum_{i=1}^n x_i.$$

Thus, the maximum likelihood estimator is simply the empirical mean. The generative process consists in sampling from Bernoulli($\hat{\theta}$).

3.3 Example: Gaussian Distribution

Consider real-valued data $x_i \in \mathbb{R}$ modeled as a Gaussian distribution

$$p(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

Exercise 3.2 (MLE for the Gaussian Model)**.** Show that the log-likelihood is

$$\ell(\mu, \sigma^2) = -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2.$$

Verify that the function is concave in (μ, σ^2) and compute the gradients to obtain

$$\nabla_\mu \ell = \sum_{i=1}^{n} \frac{(x_i - \mu)}{\sigma^2} = 0 \;\Rightarrow\; \hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i,$$
$$\nabla_{\sigma^2} \ell = -\frac{n}{2\sigma^2} + \sum_{i=1}^{n} \frac{(x_i - \mu)^2}{2\sigma^4} = 0 \;\Rightarrow\; \hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu})^2.$$

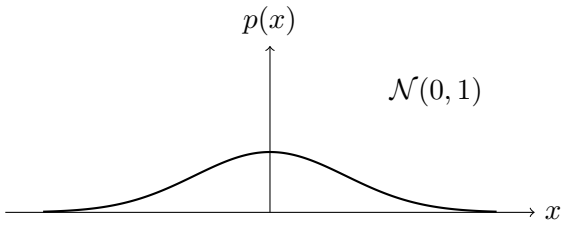

Figure 1: Illustration of a Gaussian probability density.

4 Beyond Simple Parametric Models

4.1 Limitations

The maximum likelihood framework is tractable when the parametric family of densities is well specified (e.g., Gaussian, Bernoulli). However, in many practical situations the true data distribution cannot be faithfully represented by a simple family.

4.2 Mixture Models

A natural extension is to consider *Gaussian Mixture Models* (GMMs), in which the density is expressed as a convex combination of Gaussian components:

$$p(x) = \sum_{i=1}^{K} \pi_i \mathcal{N}(x \mid \mu_i, \sigma_i^2).$$

Goal

Estimate mixture weights π_i and component parameters (μ_i, σ_i^2) that best fit the data distribution.

In this case, the likelihood function does not admit a closed-form maximizer. Numerical methods such as gradient-based optimization or quasi-Newton algorithms (e.g., BFGS) are typically employed.

4.3 Expectation–Maximization (EM)

The Expectation–Maximization algorithm provides a principled framework for maximum likelihood estimation in the presence of latent variables.

Let z denote an unobserved latent variable. Then

$$\ell(\theta) = \sum_{i=1}^{n} \log \sum_{z} p(x_i, z \mid \theta) = \sum_{i=1}^{n} \log \sum_{z} q(z \mid x_i) \frac{p(x_i, z \mid \theta)}{q(z \mid x_i)}$$

By introducing an auxiliary distribution $q(z \mid x_i)$. Then by applying Jensen's inequality we obtain the lower bound:

$$\ell(\theta) \geq \sum_{i=1}^{n} \sum_{z} q(z \mid x_i) \log \frac{p(x_i, z \mid \theta)}{q(z \mid x_i)} = \ell^{\text{surrogate}}(q, \theta).$$

Tip

The EM algorithm alternates between optimizing $\ell^{\text{surrogate}}$ with respect to q (E-step) and with respect to θ (M-step). The optimal q is given by the posterior distribution $q(z \mid x_i) = p(z \mid x_i, \theta)$.