Probabilistic Generative models

Week 1 – Foundations

— Setup —

A PGM, or simply a GM, is a methodology for generating data. In general, the GM is constructed and adjusted using observations with the aim to generate data with the same statistical properties of the observations. The 'probabilistic' nature of the PGMs considered follow from the fact that the data generated will be considered to be realisations of a Random variable; e.g. X.

In this sense, the probability distribution of X, denoted $P_X(x)$, as well as its density $p_X(x)$ will be central to the study of PGM. In particular, targeting the pdf is one way of constructing PGMs, in which case the whole PGM paradigm becomes equivalent to the classical statistical model approach. However, as we will see in the course, enforcing the sought-after PGM to have an explicit parametric pdf can be rather restrictive.

Throughout the course, we will consider a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ with 3 RVs given by the measurable maps

$$X : \Omega \to \mathcal{X} \; ; \; Y : \Omega \to \mathcal{Y} \; ; \; Z : \Omega \to \mathcal{Z}$$

(observed input) (observed output) (latent variable)

Remark: Not all three variables will be present in all our models. For instance, in classification there's no justification for Z, while in clustering there's no need for X. However, we build the general set up here for formality.

We will also consider the σ-algebra $\mathcal{F}$ to be the product Borel σ-algebra. $\mathcal{F} = \mathcal{B}(X) \otimes \mathcal{B}(Y) \otimes \mathcal{B}(Z)$.

Furthermore, we will assume the joint probability of X, Y and Z has a density, that is, $\forall A \in \mathcal{B}(X)$, $B \in \mathcal{B}(Y)$, $C \in \mathcal{B}(Z)$, we have

$$\mathbb{P}(X \in A, Y \in B, Z \in C) = \int_{A \times B \times C} p(x,y,z)\, dx\, dy\, dz$$

We will also assume that all marginals and conditionals have a density. This includes $p(x,y)$, $p(y|x)$, $p(z|x,y)$ etc.

→ Discriminative V/s Generative. The generative approach aims to characterise the complete generative distribution $p(x,y,z)$, whereas in some application specific cases only the discriminative model $p(y|x)$ is needed. Let's see the following example.

Example: generative V/s discriminative binary classification.

Consider the bi classification problem, where given an observation $X = x$, one needs to estimate its label Y. A discriminative model would directly construct $p(y|x=x)$. Since this is a binary case, without loss of generality we can assume $Y \in \{0,1\}$, and model $\mathbb{P}(Y=1|X=x)$, since $\mathbb{P}(Y=0|X=x) = 1 - \%$. A model for this will only need to map $x \in \mathbb{R}^n \to \mathbb{P}(Y=1|X=x) \in [0,1]$. For instance

$$\mathbb{P}(Y=1|X=x) = \frac{1}{1+e^{-\theta x}}$$

which is known as the logistic regression.

Conversely, in a generative approach, we aim to model the joint probability $p(Y=1, X=x)$.

Modelling this distribution directly is not easy, however, we can observe that it can be factorised as

$$p(Y=1, X=x) = p(X=x \mid Y=1)\, p(Y=1)$$

which yields two much more intuitive distributions:

- the class prob $p(Y=1) \rightarrow \pi$, $(1-\pi)$
- the class-conditional prob $p(X=x \mid Y=1) \rightarrow$ choose... $f_{\theta_0}, f_{\theta_1}$

Therefore, the classifier is

$$p(Y=1 \mid X=x) = \frac{p(X=x \mid Y=1)\, p(Y=1)}{p(X=x)}$$

$$= \frac{1}{1 + \frac{p(X=x \mid Y=0)\, p(Y=0)}{p(X=x \mid Y=1)\, p(Y=1)}}$$

$$= \frac{1}{1 + e^{\log\left(\frac{1-\pi}{\pi} \cdot \frac{f_{\theta_0}(x)}{f_{\theta_1}(x)}\right)}} \quad (\star)$$

Exercise: evaluate $\star$ for $f_{\theta_1} = N(\mu_1, \Sigma)$, $f_{\theta_0} = N(\mu_2, \Sigma)$

→ Push forward measure

Despite the abundant collection of well studied statistical models, in some scenarios we can construct a more ad hoc model by applying an appropiate transformation.

Def: Consider a RV $X \in \mathcal{X}$ with measure P_X, and a measurable map $T: \mathcal{X} \to \mathcal{X}$. The measure of the transformed RV $Y = T(X)$ is known as the push forward measure of P_X through T and it is denoted by $P_{X\#T}$.

Remark: the transformations considered in the course will be s.t. the push forward measure has a density, denoted $p_{X\#T}$

Example 1: discrete push fwd

Example 2: continuous push fw

Theorem: Change of variable. Consider $X, Y \in \mathbb{R}^d$ R.V.s such that $Y = f(X)$ with $f: \mathbb{R}^d \to \mathbb{R}^d$ diff invertible. If X, Y have densities p_X and p_Y, then

$$p_Y(y) = p_X(f^{-1}(y)) \left| \det f^{-1}(y) \right|$$

Remark: though the above result provides a closed-form expression for the push-forward measure only in the invertible case, a new law P_Y can be induced for any measurable map. Furthermore, if f is 'locally invertible' and 'not too wild' P_Y has a density.

this is because

$$\mathbb{P}(Y \in A) = \sum_{f(B_i)=A} \mathbb{P}(X \in B_i)$$

- Maximum likelihood.

ML is going to be the canonical methodology for training our PGMs, and as we will see, it will recover other forms of training criteria in particular cases.

Consider a PGM for the RV y, with density $p_\theta(y)$, where $\theta \in \Theta$ is the model parameter. Also, observing realisations $y_1 \dots y_N$.

Def: likelihood function. is the probability density of the data, given the model:

$$L(\theta) = p_\theta(y_1 \dots y_N)$$

NB. Here, we abused notation stating the joint likelihood

Def: the ML estimator is

$$\theta_{ML} = \arg\max L(\theta)$$

Remark. In general, we will be dealing with i.i.d samples from y, in such cases note that

$$L(\theta) = p(y_1 \dots y_N) = \prod_i p(y_i).$$

Furthermore, we consider the training objective

$$l(\theta) = \log L(\theta) = \sum \log p(y_i)$$

Example: Gaussian linear regression

Let us consider the PGM given by

$$Y|X \sim N(aX, \sigma^2),$$

this is equivalent to $Y = aX + \epsilon$, $\epsilon \sim N(0, \sigma^2)$. The parameters here are a and σ^2. Consider obs $(y_1 \dots y_n)$ at inputs $(x_1 \dots x_n)$.

$$L(\theta) = p(y_1 \dots y_n | x_1 \dots x_n) = \prod_i p(y_i|x_i)$$

where $p(y_i|x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-1}{2\sigma^2}(y_i - ax_i)^2\right)$. Thus,

$$l(\theta) = \sum_i -\frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}(y_i - ax_i)^2 \qquad (*)$$

$\rightarrow$ optimal a and σ^2

Remark: Recall the $(*)$: ML recovers least sq,

Example (binary classification). Consider observations $x_i \in \mathbb{R}^d$ and labels $y_i \in \{0, 1\}$. The classifier will be modelled by

$$p_\theta(y = 1|x) = \sigma(s(x))$$

where: $\sigma(s(x)) = \frac{1}{1 + e^{-s(x)}}$; and $s(\cdot)$ is a feature extractor (rf could be linear)

Assuming i.i.d. (Bernoulli)

$$L(\theta) = \prod_i p(y_i|x_i) = \prod_i \sigma(s(x_i))^{y_i} (1 - \sigma(s(x_i)))^{1-y_i}$$

and

$$l(\theta) = \sum_i y_i \log \sigma(s(x_i)) + (1 - y_i) \log(1 - \sigma(s(x_i)))$$

* Cross entropy

Example: clustering. Let us consider a set of observations $D = \{x_i\}_{i=1}^n$ and implement a clustering operation. We will assume that there are K clusters, each specified by a density p_k and that the prob of x coming from p_k is π_k; where $0 \le \pi_k \le 1 \; \forall k$ and $\sum_k \pi_k = 1$. The prob of x is

$$p(x) = \sum_k \pi_k p_k(x) \qquad (*)$$

and $\ell(\theta) = \sum_i \log \sum_k \pi_k p_k(x)$. [Say what parameters are]

Note that this has two probs.

i) this doesn't give the cluster assignments (though that can be solved by hand)

ii) optimising $(*)$ is ill posed: if $p_k = N(\mu_k, \Sigma_k)$ we can set $\mu_k = x_i$, $\Sigma = 0$ for $L = \infty$.

We can sort this out introducing a latent variable that represents the cluster assignment, namely

$$z_{nk} = 1 \text{ iff } x_n \in C_k; \quad r_{nk} = P(z_{nk} = 1)$$

With this $p(x_n | z_{n:}) = \prod_k p_k^{r_{nk}}$

we have the "complete-data likelihood"

$$\ell(\theta) = \log \prod_n \prod_k p_k^{z_{nk}}(x_n) = \sum_n \sum_k z_{nk} \log p_k(x_n)$$

Which is now optimisable but impractical, since we don't have the z's

Workaround: estimate z_{nk} by $r_{nk} = P(z_{nk} = 1 | x_n, \theta)$ which is equal to $E(r_{nk} | x_n, \theta)$ $\longrightarrow$ can't do ML here.

Some properties of ML

- Consistency
- Equivariance
- Asymptotic normality
- Asymptotic efficiency

Intro to information theory

Motivation. Let us consider a RV X with P_X. Observe that $-\log p_X(x)$ is represent a measure of information gained from obtain the value x as a sample of X. We consider a communication channel $\mathcal{X}^A \longrightarrow \mathcal{X}^B$ where A is transmitting samples from X to B

- B sees samples from X, so it's average 'surprise' is

$$H(X) = -\int_X p(x)\log(p(x))\,dx \quad \text{(later, this)}$$

$$H(x) = -\sum_X p(x)\log(p(x)) \geq 0$$

- Properties: (Jensen: $\mathbb{E}(\log(p(x))) \leq \log \mathbb{E}(p(x))$

① $H(X) = \sum p(x)\log\left(\frac{1}{p(x)}\right) \leq \log \sum_K 1 = \log K$

and $H(X_u) = \sum \frac{1}{K}(\log(K)) = \log K$

② if $\exists i$ s.t. $p(x_i) = 1 \;\; p(x_{\neq i}) = 0 \rightarrow H(x) = 1 \cdot \log(1) = 0$

Bernoulli example

$X \sim p(x) = \theta^x (1-\theta)^{1-x}$

$H(X) = -\theta \log \theta + (1-\theta) \log (1-\theta)$

→ show plot

- The entropy, in addition to the measure of disorder, can be understood as the cost of the optimal form of compression. Think of a compression strategy using symbols $s_1, \ldots,$ where with increasing size. E.g. $s_1 = 0$ $s_2 = 11$ $s_2 = 1$ ⋮ $s_3 = 10$ the strategy would be to assign s_i to the ith most frequent. Then $H(X) = \sum_i p(x) \log \left(\frac{1}{p(x)}\right)$. is the average message size of the code.

Now assume a communication channel $A \to B$ where B receives samples $X \sim p$ but mistakenly thinks that $X \sim q$. Its assessment of info would be

$H_{CE}(p, q) = -\sum \log(q(x_i))\, p(x_i)$. → this is also the cost of compressing $X \sim p$ via q

Discussion. Would $H(p,q)$ greater or smaller than $H(p,p)=H(p)$?

Let's see: $H(p) - H(p,q) = \sum p \log\left(\frac{q}{p}\right)$

$$\leq \log \sum p \cdot \frac{q}{p} = \log 1 = 0$$

$\Rightarrow H(p) \leq H(p,q)$ with e.g. attainable only for $p=q$

Therefore minimising Cross-entropy wrt. to one arg. is precisely an attempt to match $p=q$.

The notion of entropy/cross-entropy can be extended to continuous RVs with densities via

- $H(X) = \int p(x) \log(p(x)) dx$
- $H(p,q) = \int p(x) \log(q(x)) dx$

Remark: $H(X)$ can be positive, negative or zero.

$H(U_{[0,a]}) = \int_0^a \frac{1}{a} \log\left(\frac{1}{a}\right) dx = \log(a^{-1}) = -\log(a)$. However, $H(p,q) \geq H(p)$

Quantifying this discrepancy gives

$$KL(p||q) = H(p,q) - H(p) = \sum p \log\left(\frac{p}{q}\right) \geq 0$$

KL is a divergence measure, that is, a function that quantifies how far is p from q, that always positive and $D(p,q)=0$ iff $p=q$. (a.e.) - identity of the indiscernibles.

- However, note that
  - KL is not symetric
  - Does not have Δ ineq.
  - it's only defined when $p \gg q$

KL as a metric to compare p and q

In the continuous case, it is interesting to understand what type of convergen KL gives

- Let us consider other two divergences

L_1: $D_{L_1}(p,q) = \int |p-q| dx$

χ^2: $D_{\chi^2}(p,q) = \int \frac{|p-q|}{q} dx$

Scenario 1 $p = U_{[0,1]}$, $q = \begin{cases} e^{-n} & x \in [0, 1/n] \\ c_n & x \in (1/n, 1] \end{cases}$

$$D_{L_1}(p,q) = \int_0^{1/n} |e^{-n} - 1| dx + \int_{1/n}^1 |c_n - 1| dx = \frac{|e^{-n}-1|}{n} + \frac{|c_n - 1|}{\frac{n-1}{n}} \to 0 \quad n \to \infty$$

$$KL(p,q) = \int p(x) \log \frac{p(x)}{q(x)} = \int_0^{1/n} -\log e^{-n} + \int_{1/n}^{n} -\log c_n = n \cdot \frac{1}{n} + \frac{-n}{n-1}\log c_n$$

($\frac{1}{n} \to 0$, $\log c_n \to 0$)

$= KL$ not conv.

Case 2 $p = (1-\epsilon, \epsilon)$, $q = (1-\epsilon^2, \epsilon^2)$ (Bernoullis)

$$KL = \sum p_i \log\left(\frac{p_i}{q_i}\right) = \underbrace{(1-\epsilon)\log\left(\frac{1-\epsilon}{1-\epsilon^2}\right)}_{\to 0} + \underbrace{\epsilon \log\left(\frac{\epsilon}{\epsilon^2}\right)}_{\to -\infty} \to 0 \quad (\text{l'Hôp})$$

$$\chi^2 : \frac{\|1-\epsilon-1+\epsilon^2\|^2}{1-\epsilon^2} + \frac{\|\epsilon-\epsilon^2\|^2}{\epsilon^2} = \frac{\|\epsilon^2-\epsilon\|^2}{1-\epsilon^2} + \|1-\epsilon\|^2$$

$\to 1$

This means that, even though KL provides a stronger sense of convergence than L_1, there are 'stronger' divergences.

- Direct V/s reverse KL.

Since $KL(p\|q)$ is not symmetric, it makes sense to study

$KL(q\|p)$, some observation.

Since $p \gg q$ is needed for direct $KL(p\|q)$, we have that $KL(p\|q)$, as a metric to penalise q as an approximation of p, will promote q covers all the support of p.

Example: $KL(N(\mu_0,\sigma_0)\,\|\,N(\mu_1,\sigma_1)) = \log\frac{\sigma_1}{\sigma_0} + \frac{\sigma_0^2}{2\sigma_1^2}(\mu_0-\mu_1)^2 - 1/2$

When $\mu_0 = \mu_1$; $KL = \log \frac{\sigma_1}{\sigma_0} + \frac{\sigma_0^2}{2\sigma_1^2} - \frac{1}{2}$ Recall that "σ_0^2" is the true param. and that $\sigma_1 = \sigma_0$ achieves $KL = 0$, However let's see what's worse: $\sigma_1 > \sigma_0$ or $\sigma_1 < \sigma_0$.

Plot $KL = \log \frac{\sigma_1}{\sigma_0} + \frac{\sigma_0^2}{2\sigma_1^2} - \frac{1}{2}$ for $\sigma_0^2 = 1$ as a fn of σ_1

(and vice versa)

Plot: Consider $p = GMM_4$, $q = GMM_2$; plot the optim. KL.

KL and Maximum Likelihood.

Let's now turn to our setting of a learning problem. Consider a PGM given by $p_\theta(x)$ and iid observations $x_1 \dots x_n \sim p(x)$. We could use the KL as a metric to adjust p_θ to p. that is

$$\theta = \arg\min KL(p \| p_\theta)$$

This is unfeasible in practice, since p is not known. However, we

$$\theta = \arg\min \int \log p \; p(x)dx - \int \log q_\theta \; p(x)dx$$

$$= \arg\max \mathbb{E}_p \log q_\theta \quad \Big\downarrow MC$$

$$= \arg\max \sum_{x_i} \log q_\theta(x_i)$$

$\Rightarrow$ minimizing KL is equivalent to ML (large data)

As we'll see in detail during the module, when designing a PGM, we have the following optio

- Closed-form likelihood based models. These are classical statistical models that we can choose, or design from first pple.
  - N, cat, expo, exp normal, log normal.

- likelihood based push forward models. This is a transformation of a simple base (or 'source') model via a learned map. the params of the map appear explicitly in the likelihood. For instance take a base measure $p(z)$ and model $x = T_\theta(z)$, with z = Gaussian.

this way $p_x = p_z(T^{-1}(x)) \left|\det \nabla T^{-1}(x)\right|$

this only works for invertible maps - otherwise likelihood is not available.

- Implicit generative models.

Take a base measure, e.g. $z \sim N(0, I)$ and map with e.g. a neural net. this does not admit ML, and models need to be learnt in some other way.

[plot these flow models]