Machine Learning Week 7 Assignment

> Explain the concept of score matching and describe how it is used in score-based (diffusion) generative models.

動機

若給定一堆手寫數字的照片（尺寸為28x28），要如何辨別是哪個數字，並且再生成同樣數字的手寫照片呢？

我們可以將這個問題簡化成：

> 給定一組資料，其維度為 $\mathbb{R}^{784}$，我們想要估計一個將**手寫照片**打到其**對應的數字**的機率密度函數 $p(x;\theta):\mathbb{R}^{784}\to\mathbb{R}$。

簡單的想法是利用 energy-based 的方法構造出這個機率密度函數的形式：

$$p(x;\theta)=\frac{e^{q(x;\theta)}}{Z(\theta)} \tag{1}$$

其中 θ 是控制機率密度函數的參數、$Z(\theta)$ 是讓積分值為 1 的標準化常數。這樣假設可以保證 $\int_{\mathbb{R}^d}p(x;\theta)dx=1$ 而且 $p(x;\theta)\geq 0$

但很多時候，要計算 Z_θ 的話需要做高維度的積分，而這並不好做。

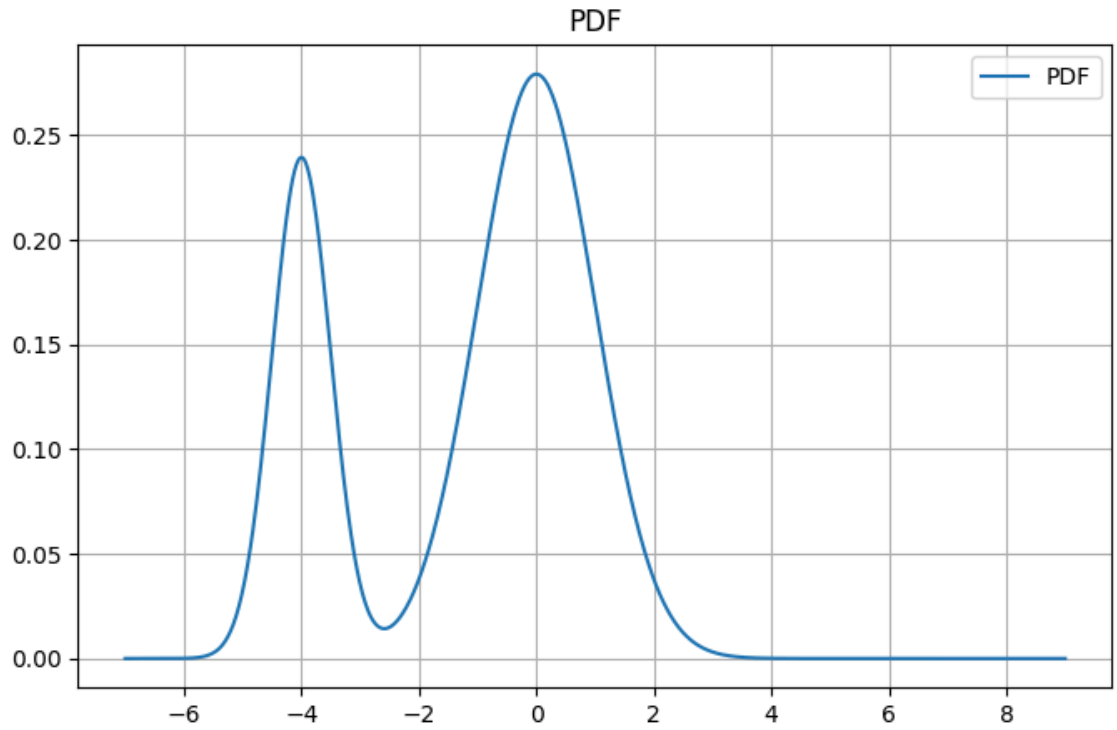
引入 Score function

若我們定義 score function：

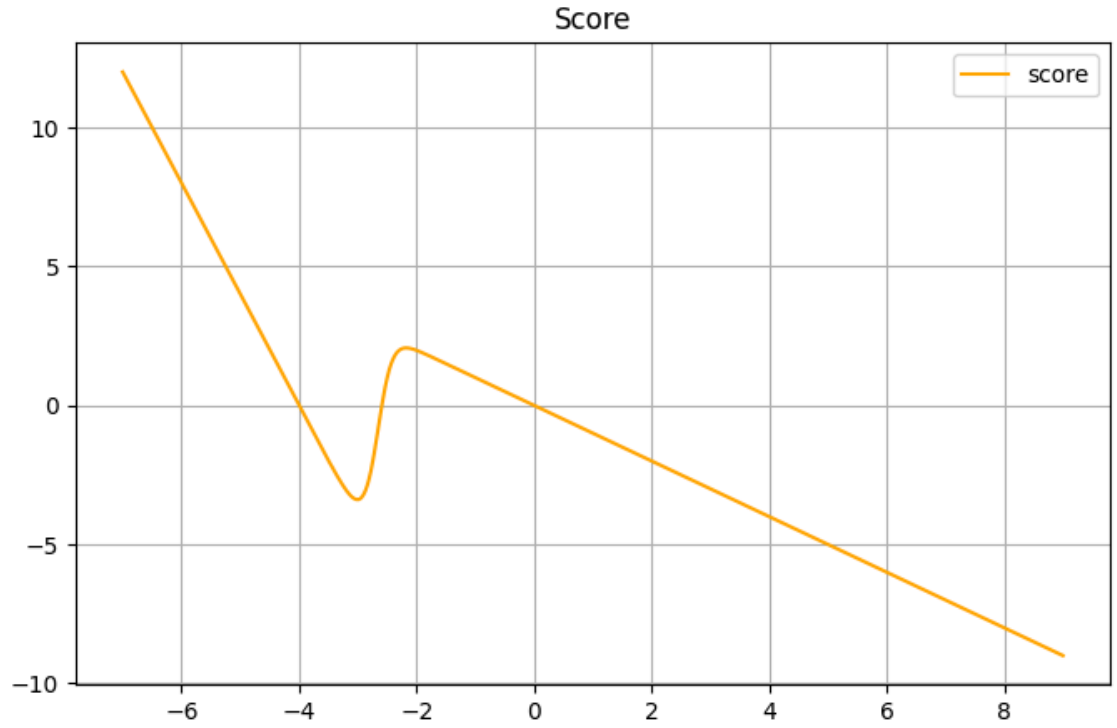
$$S(x;\theta)=\nabla_x\log p(x;\theta)=\nabla_x q(x;\theta) \tag{2}$$

則直觀上，score 可以告訴我們 往哪個方向才能走到更高的密度 。

舉例來說：給定一個 Mixtured Gaussian：



依照 (2) 可計算出對應的 score：



則可以看出在 $x = -2$ 時，因為 $S(x) > 0$，所以 x 往右移動可以帶我們來到更高的密度；而在 $x = 2$ 時，因為 $S(x) < 0$，所以 x 往左移動可以來到較高的密度。

因此，若我們想學習一個機率密度函數，相當於去學習一個 score function。

接下來，我們嘗試讓去找一個最好的 score function，這個動作稱為 score matching。

Explicit/Implicit Score Matching (ESM/ISM)

我們想要學習一個 score function $S(x;\theta)$ 使得以下誤差最小：

$$L_{ESM}(\theta) = \mathbb{E}_{x\sim p(x)}\|S(x;\theta) - \nabla_x \log p(x)\|^2 \tag{3}$$

上式稱為 **Explicit Score Matching (ESM)**。

不過實際上並沒辦法做，因為我們正是不知道原始資料的分布 $p(x)$。

所幸，在 Hyvärinen (2005) 中證明了下式：

$$L_{ESM}(\theta) = \mathbb{E}_{x\sim p(x)}\Big[\|S(x;\theta)\|^2 + 2\nabla_x \cdot S(x;\theta)\Big] + \mathbb{E}_{x\sim p(x)}\|\nabla_x \log p(x)\|^2 := L_{ISM}(\theta) + C$$

其中，前項稱為 **Implicit Score Matching (ISM)**，而 C_1 僅為和 θ 無關的常數。

簡單來說，透過以上的關係式，我們不需透過原始資料分布就能估計 score function。並且，因為 L_{ESM} 和 L_{ISM} 只相差一個常數，因此兩者的 minimizer 相同，意即：

$$\text{argmin}_\theta L_{ESM}(\theta) = \text{argmin}_\theta L_{ISM}(\theta) := \theta^* \tag{4}$$

所以，只要能找到能使 $L_{ISM}(\theta)$ 發生最小值的參數 θ 就可以了。

並且注意到，最好的狀況使得 $L_{ISM}(\theta^*) = 0$ 且 $L_{ISM}(\theta^*) < 0$。

於是，那當我們想要生成資料時，只要在空間中均勻地灑足夠多的點，那麼這些點就能依照 score function 告訴我們的方向去移動到密度最高的地方。

但是，這樣移動的話會讓所有的資料點都移動到密度高峰的地方。以上面的 Mixtured Gaussian 來說，這麼一來所有的點都移動到 -4 和 0 的地方了，這也不是我們想要的機率分布。

Denoise Score Matching (DSM)

因此，我們考慮在原始資料上再加入微小擾動，也就是

$$x = x_0 + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 I) \tag{5}$$

其中 x 為受擾動後的資料，x_0 為原始資料。並且令 $p_0(x)$ 是原始資料的分布、$p(x|x_0)$ 為受擾動資料的條件分布及 $p_\sigma(x)$ 為受擾動資料的分布。

若現在要學習的是受擾動後資料的分布，我們也考慮去學習它的 score function $S_\sigma(x;\theta) = \nabla_x \log p_\sigma(x)$，也就是希望下式最小：

$$L_{DSM} = \mathbb{E}_{x_0\sim p_0(x_0)}\mathbb{E}_{x|x_0\sim p(x|x_0)}\Big[\|S_\sigma(x;\theta) - \nabla_x \log p(x|x_0)\|^2\Big] \tag{6}$$

若我們將對 $S_\sigma(x;\theta)$ 的 loss function 用 Explicit Score Matching 的方式寫出：

$$\mathbb{E}_{x\sim p_\sigma(x)}(\|S_\sigma(x;\theta) - \nabla_x \log p_\sigma(x)\|^2) \tag{7}$$

依照期望值的定義展開，經過一些代數運算可以得到 (6) 和 (7) 也只差一個常數。因此只要能最小化 L_{DSM} 就好。

加入一點小擾動的用處在於，我們可以對不同大小的 σ 學習對應的 $S_\sigma(x;\theta)$。最後要生成資料時，只要反向依照遵從常態分布的機率密度函數推回即可生成有原始分布的資料。