Fit probability density function without knowing the form of distribution

Motivation

After reading the paper of GAN[1], I wonder what will happen if the labels equal 1 or -1 instead of 1 or 0 during training the discriminative model, and how about 1 or 2, 31 or 11, and so on, so I may try a or b to get the best a and b by doing some mathematical things.

Method one

Denote two distributions by D_1 and D_2.

Denote the probability density functions of D_1 and D_2 by $p_1(x)$ and $p_2(x)$, respectively.

Denote the probabilities of D_1 and D_2 on the interval $[x, x+\Delta x]$ by $P_1([x, x+\Delta x])$ and $P_2([x, x+\Delta x])$, respectively.

Denote the sets of samples from D_1 and D_2 by S_1 and S_2, respectively.

Denote the numbers of elements in S_1 and S_2 by n_1 and n_2, respectively.

Denote the neural network to fit the probability density function $p_1(x)$ by $f(x)$.

Denote sample in $S_1 \cup S_2$ by $\vec{s} = (i, j, x)$, where x is the value, j differentiates the samples whose

values are the same, and $i = \begin{cases} 1 & \vec{s} \in S_1 \\ 2 & \vec{s} \in S_2 \end{cases}$

Let $label(\vec{s}) = label(i, j, x) = \begin{cases} a & i = 1 \\ b & i = 2 \end{cases}$

Denote the number of labels which equal a at the position x by $n_1(x)$.

Denote the number of labels which equal b at the position x by $n_2(x)$.

Let $loss = E[f(x) - label(i, j, x)]^2$

After training, $f(x) = E[label(i, j, x)]$

$$= \frac{a \cdot n_1(x) + b \cdot n_2(x)}{n_1(x) + n_2(x)}$$

$$= \lim_{\Delta x \to 0} \frac{a \cdot P_1([x, x+\Delta x]) \cdot n_1 + b \cdot P_2([x, x+\Delta x]) \cdot n_2}{P_1([x, x+\Delta x]) \cdot n_1 + P_2([x, x+\Delta x]) \cdot n_2}$$

$$= \lim_{\Delta x \to 0} \frac{a \cdot p_1(x) \cdot \Delta x \cdot n_1 + b \cdot p_2(x) \cdot \Delta x \cdot n_2}{p_1(x) \cdot \Delta x \cdot n_1 + p_2(x) \cdot \Delta x \cdot n_2}$$

$$= \frac{a \cdot p_1(x) \cdot n_1 + b \cdot p_2(x) \cdot n_2}{p_1(x) \cdot n_1 + p_2(x) \cdot n_2} \cdot \lim_{\Delta x \to 0} \frac{\Delta x}{\Delta x}$$

$$= \frac{a \cdot p_1(x) \cdot n_1 + b \cdot p_2(x) \cdot n_2}{p_1(x) \cdot n_1 + p_2(x) \cdot n_2}$$

When $b = 0$, $\quad f(x) = \dfrac{a \cdot p_1(x) \cdot n_1}{p_1(x) \cdot n_1 + p_2(x) \cdot n_2} = \dfrac{a \cdot p_1(x)}{p_1(x) + p_2(x) \cdot n_2 / n_1}$

When D_2 is a uniform distribution, let $p_2(x) = \begin{cases} p_2 & x \in [l, r] \\ 0 & x \notin [l, r] \end{cases}$, then

$$f(x) = \frac{a \cdot p_1(x)}{p_1(x) + p_2 \cdot n_2 / n_1} \qquad \forall x \in [l, r]$$

When $p_2 \cdot n_2 / n_1$ is large enough to ignore $p_1(x)$, $\quad f(x) \approx \dfrac{a \cdot p_1(x)}{p_2 \cdot n_2 / n_1} \quad \forall x \in [l, r]$.

Considering $p_2(x) = \begin{cases} p_2 & x \in [l, r] \\ 0 & x \notin [l, r] \end{cases}$, and divisor has to be non-zero, the interval $[l, r]$ has to cover

the interesting area of D_1.

When $a = p_2 \cdot n_2 / n_1$, $\quad f(x) \approx p_1(x) \quad \forall x \in [l, r]$.

Method two

Denote the same as the method one, then

$$f(x) = \frac{a \cdot p_1(x) \cdot n_1 + b \cdot p_2(x) \cdot n_2}{p_1(x) \cdot n_1 + p_2(x) \cdot n_2} .$$

When $a = 0$, $b = 1$, $n_1 = n_2$, $p_2(x) = \begin{cases} p_2 & x \in [l, r] \\ 0 & x \notin [l, r] \end{cases}$,

$$f(x) = \frac{p_2}{p_1(x) + p_2} \quad , \quad \forall x \in [l, r] \qquad \Rightarrow$$

$$p_1(x) = p_2 \left(\frac{1}{f(x)} - 1 \right) \quad , \quad \forall x \in [l, r]$$

Reference

[1] Goodfellow I , Pouget-Abadie J , Mirza M ,et al.Generative Adversarial Nets[C]//Neural Information Processing Systems.MIT Press, 2014.DOI:10.3156/JSOFT.29.5_177_2.