Advanced Topics in Machine Learning

Introduction to Probabilistic Models & Generative Models: Sampling and Graphical Models

Group 3

Priyanshu Kumar (190651)
Harshit Itondia(190369)

1 Introduction to Probabilistic Models

1.1 Introduction

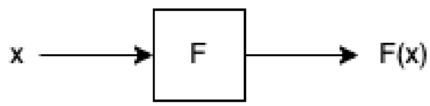

Figure 1

In a typical machine learning task, given a model F, we want to make its prediction as close to Y_d ($Y_{desired}$) as possible where Y_d may be any specific class in classification problem or any continuous value in regression problem. For example, suppose we have trained our model on a classification problem where classes are car, boy, girl and plane and inputs are their images. Here F(x) is P(y|x) basically and we have to maximise its probability value with respect to car if the input itself is of car and the categorical distribution desired is something as shown below:

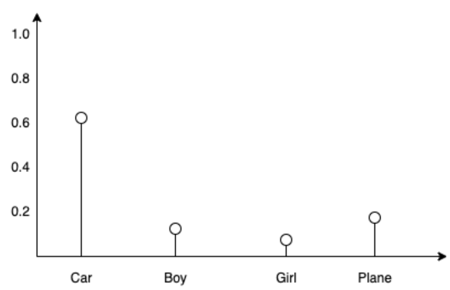

Figure 2: Categorical Distribution

1.2 Motivation behind Sampling:

1.2.1 Imbalanced Dataset

Given, Imbalanced Dataset (which is basically data set with skewed class proportions), with the help of sampling, we can generate more samples by modeling the probability distribution with respect to various classes. For example, given (x, plane) data, we can model its probability distribution and ultimately generate more sample p(x|plane).
We have two major problems at hand:

1. How to sample efficiently from a given distribution p(x)?
2. Given samples x_s, how to estimate p(x) and sample more x from it?

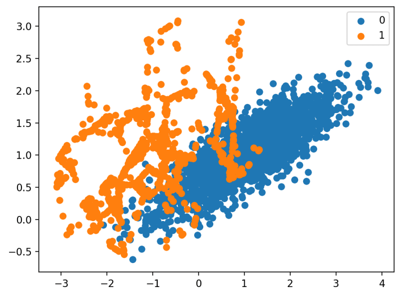

Figure 3: Imbalanced Data

1.2.2 Other Uses of sampling:

1. To generate new pictures with different emotions.
2. To generate artistic images by combining different concepts.
3. For security purpose, to learn the actual distribution of data, so that we can know what is real and what is fake.
4. To fill in missing values like suppose a human face is given to us, and one part let's say nose is missing, then with the help of probability distribution, we can actually generate nose for the face.
5. Given p(x), we have to find E[x], E[x^2], E[f(x)]

$$E[f(x)] = \int p(x)f(x)dx \tag{1}$$

   If p(x) is complicated, it cannot be solved analytically. Therefore, we have to sample from those distributions and can calculate expectation values easily using Monte Carlo Aprroximation:
   Let's say we have x_1, x_2, x_3,..., $x_n \sim$ p(x), then we can calculate expectation values as shown below:

$$E[x] = \int xp(x)\,dx = \frac{1}{N}\sum_{s=1}^{N} x_s \tag{2}$$

$$E[f(x)] = \int p(x)f(x)\,dx = \frac{1}{N}\sum_{s=1}^{N} f(x_s) \tag{3}$$

Note: Sometimes it is difficult to sample from p(x) and sometime only samples are given, we have to estimate p(x) and then sample from it. So approximate p(x) with some q(x), which should be close to p(x) and easy to sample.

1.3 How to check closeness of 2 distributions?

1. KL divergence and Cross Entropy can be used to compare two distributions.
2. Using MLE, we can match q(x) and samples from p(x).

2 Generative models: Sampling and Graphical Models

2.1 Ancestral Sampling

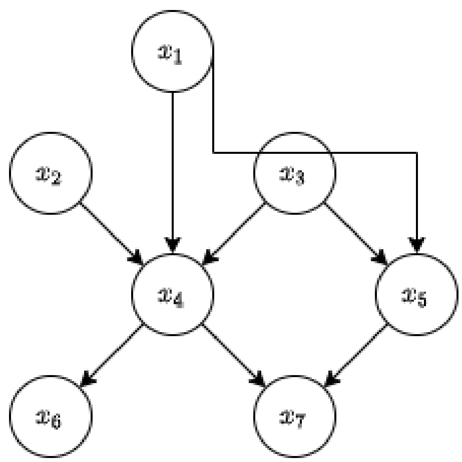

Figure 4: Example

Consider a joint distribution $P_{x1,x2,x3,x4,x5,x6,x7}(x_1,x_2,x_3,x_4,x_5,x_6,x_7)$ over 7 variables that factorizes according to Figure 4 which is a directed acyclic graph, we have supposed that the variables have been ordered such that no nodes are children of any lower numbered node. Our goal is drawing samples from such joint distribution, we can draw the samples in following way: $x_1 \sim p_{x1}(x_1)$, $x_2 \sim p_{x2}(x_2)$, $x_3 \sim p_{x3}(x_3)$, $x_4 \sim p_{x4}(x_4|x_1, x_2, x_3)$, $x_5 \sim p_{x5}(x_5|x_1, x_3)$,$x_6 \sim p_{x6}(x_6|x_4)$ and $x_7 \sim p_{x7}(x_7|x_4, x_5)$.

2.1.1 Steps required

1. Start from root.
2. Progressively sample the children.

2.2 Graphical Models Representation

Generally we have latent variables also in this type of representation.

$$p(x) = \sum_{w} p(x, w) \tag{4}$$

$$p(\mathbf{x}, w) = \prod_{i=1}^{n} p(x_i, w) p(w) \tag{5}$$

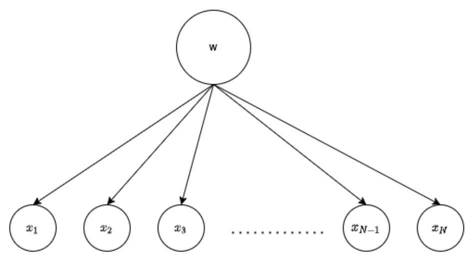

Figure 5: Representation for Equation 5

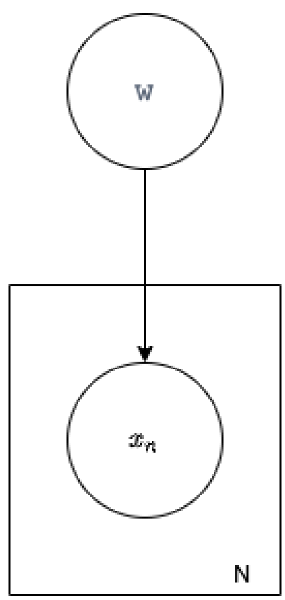

Figure 6: Simplified Representation

We can have deterministic parameters(μ, σ, α) or known observed values(t_n) and we want to estimate x_n.

$$p(x, w|t, \mu, \sigma, \alpha) = p(w|\alpha) \prod_{n=1}^{n} p(x_n|t_n, \mu, \sigma, w) \tag{6}$$

To train parameters μ, σ, α, training data has (x_n, t_n). E.g. for image recognition task, t_n can be image while x_n can be label correspoding to it

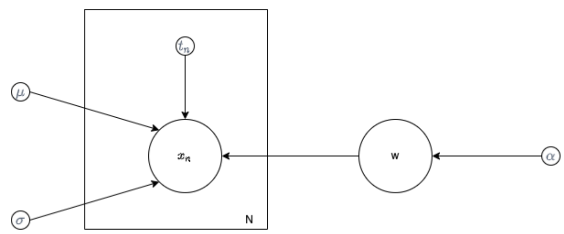

Figure 7: Representation Graph for Equation 6

2.2.1 Inference

To test for a new t_{n+1} or $\hat{t}$, we want to estimate p($\hat{x}$ | $\hat{t}$, μ, σ, w). The pictorial graph shown below illustrates how it can be estimated

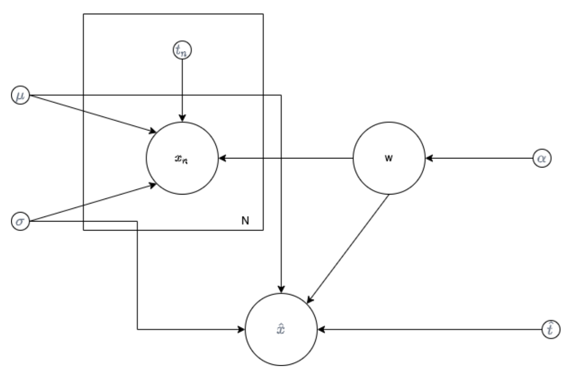

Figure 8: Representation Graph for Inference

For GMM or other discrete latent distribution, w $\sim$ Categorical distribution and such representation can very well fit those cases.