Starting from Information

Ruichen Wang

November 16, 2018

Abstract

Starting with the introducing the origins of information, extend to entropy and its families. and some introduction and explanation of entropy related algorithms, like log-likelihood, softmax classification. Then we will talk about one basic algorithm solving intractable problems, which is called variational bayesian inference, and it's closely related algorithms like latent dirichlet allocation(LDA), variational auto encoder(VAE), generative adversarial network(GAN),etc.

Contents

1 What is Information?

1.1 Defination of Information

How to measure the uncentainty of certain event in a mathematics?

Given x is some event, P(x) is probability which event x happens. Intuitively, the information should have inverse proportion to the probability, which is

$$I(x) = \frac{1}{P(x)}$$

We also want the information more stable,remove the division, so

$$I(x) = log\frac{1}{P(x)} = -logP(x)$$

1.2 Property of Information

As a result, the $-logP(x)$ has every properties we want:

- Lower probability, higher information
- Higher probability, lower information
- Multi-event happens, the probability is multiplied. the information is summed

Mathematics As we know, $P(x) \in [0,1]$, and larger P(x) should have smaller information.

$$P(x_1, x_2) = P(x_1) * P(x_2)$$

$$logP(x_1, x_2) = logP(x_1) + logP(x_2)$$

2 Entorpy (Expectation(sum) of Information)

2.1 Shannon's Information Theory

Claude Elwood Shannon(1916-2001).
1937 MIT Master degree.
1940 MIT Ph.D degree from MIT.
1948 Published a landmark paper 'A mathematical Theory of Communication'.
Entropy is defined as:

$$H(x) = E[I(x)] = \sum_{i=1}^{n} P(x_i)I(x_i) = -\sum_{i=1}^{n} P(x_i)logP(x_i)$$

2.2 Property of Entropy

The property of the entropy is quite simple

- Higher probability, the less information, the lower entropy
- Non-negative, every event has some information
- Cumulative, multile events happens, the information is the sum of them.

3 Families of Entropy

3.1 Cross-Entropy

Suppose we don't know P(x) yet, so we make an 'artificial' probability distribution Q(x). How can we measure the cost as we using Q(x) to approximate P(x)? We define corss entropy as:

$$H(P,Q) = -\sum_{x} P(x)logQ(x)$$

Estimation There are many situations where P(x) is unknown. Given a test set N observed, which comes from a Monte Carlo sampling of the true distribution P(x). Cross entropy is calculated using :

$$H(T,Q) = -\sum_{i=1}^{N} \frac{1}{N} logQ(x_i)$$

3.1.1 Relation to Log-likelihood

for the maximum likelihood estimation (MAE), we have:

$$\prod_i q_i^{N_{p_i}}$$

So log-likelihood,divided by N is :

$$\frac{1}{N} log \prod_i q_i^{N_{p_i}} = \sum_i p_i log q_i = -H(p, q)$$

So maximum the likelihood is the same as minimizing the cross entropy.

3.1.2 Cross-entropy Loss in Classification

In machine learning, cross-entropy loss is widely used now, it often defines as :

$$L = -ylog(y^{'}) = H(y, y^{'})$$

It describes the distance between the prediction and truth.

3.1.3 Simple and Elegant Relationship with Softmax

As the softmax probability and cross-entropy loss is so so common, and they often work together. But why? Because the simplicity of the derivative. Softmax function:

$$p_i = \frac{e^{a_i}}{\sum_{k=1}^{N} e^{a_k}}$$

Derivative of softmax $\frac{\partial p_i}{\partial \alpha_j}$:

$$\frac{\partial p_i}{\partial \alpha_j} = \begin{cases} p_i(1 - p_j) & i = j \\ -p_j * p_i & i \neq j \end{cases}$$

The cross entropy loss:

$$L = -\sum_i y_i log p_i$$

Derivative of cross entropy loss:

$$\frac{\partial L}{\partial o_i} = -\sum y_k \frac{1}{p_k} * \frac{\partial p_k}{\partial o_i}$$

From the dervative of softmax we derived earlier,

$$\frac{\partial L}{\partial o_i} = -y_i(1 - p_i) - \sum_{k \neq i} y_k \frac{1}{p_k}(-p_k * p_i)$$

$$= p_i(y_i + \sum_{k \neq i} y_k) - y_i = p_i - y_i$$

This is why we often use softmax and cross entropy together, The gradient is quite simple to calculate.

3.2 Kullback-Leibler Divergence

KL divergence is also called relative entropy. It is a measure of how one probability distribution is different from a second.
For discrete probability distirbutions P and Q defined on the same probability space, the KL divergence from Q to P (Q with respect to P) is defined as :

$$D_{KL}(P \parallel Q) = -\sum_i P(i)log(\frac{Q(i)}{P(i)})$$

This can also be written as the cross entropy minius original entropy:

$$D_{KL}(P \parallel Q) = H(P, Q) - H(P)$$

Which means the more entropy using Q generates with respect to original distirbution P.

3.2.1 Interpretations

In machine learning, $D_{KL}(P \parallel Q)$ is often called the information gain achieved if Q is used instead of P.

Expressed in the language of Bayesian inference, $D_{KL}(P \parallel Q)$ is a measure of the information gained when one revises one's beliefs from the prior probability distribution Q to the posterior probability distribution P.

In applications, P typically represents the true distribution of data. Q represents the model. Minimize $D_{KL}(P \parallel Q)$ is to find a Q that closest to P.

3.2.2 Property of KL

- Non-negative
  As a result known as Gibbs's inequality, with $D_{KL}(P \parallel Q)$ zero if and only if P = Q.
- Asymmetric
  $$D_{KL}(P \parallel Q) \neq D_{KL}(Q \parallel P)$$
  we can define symmetrised divergence as:
  $$\frac{D_{KL}(P \parallel Q) + D_{KL}(Q \parallel P)}{2}$$

3.2.3 Applications

Generative models like VAE, we may need a new section to go through this.I will go into details later. Here I just put a bayesian equation here :).

$$posterior = \frac{likelihood * prior}{evidence}$$

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)}$$

4 Variational Bayesian Inference

4.1 Variational Inference

The idea behind variational inference is to first posit a family of densities and then to find the member of that family which is close to the target. Closeness is measured by Kullback-Leibler divergence.

Variational inference is widely used to approximate posterior densities for Bayesian models,an alternative strategy to Markov chain Monte Carlo (MCMC) sampling. Compared to MCMC, MCMC algorithms sample a Markov chain, variational algorithms solve an optimization problem. variational inference tends to be faster and easier to scale to large data.

Variational inference has a close relationship with EM algorithm. You can view VAE, GAN as a certain form of variational inference. Variational inference doesn't have the global optimal point, that why VAE and GAN are very hard to train or converge.

***Tips** Compare MCMC with Metropolis-Hasting (MH). MH larger the acceptance ratio α. When extend to high dimensions, It is called Gibbs sampling. Actually, they are based on the same idea - bayesian stationary distribution.

Question description Suppose we have observations x, and hidden variables z, and some fixed parameters α.What we want is the posterior distribution.

$$p(z|x,\alpha) = \frac{p(z,x|\alpha)}{\int_z p(z,x|\alpha)dz}$$

In many cases, the $\int_z p(z,x|\alpha)dz$ is intractable. we don't know how to compute it especially in high dimensions.

Solution The main idea behind variational methods is to pick a family of distributions over the latent variables with its own **variational parameters**.

$$q(z_{1:m}|v)$$

Then find v to make q close to the posterior.

4.2 KL Divergence Measure

As mentioned above, we can use KL for this variational inference:

$$D_{KL}(q(z) \parallel p(z|x)) = E_{q(z)}\left[log\frac{q(z)}{p(z|x)}\right]$$

Intuitively, According to this formula, there are three cases:

- If q is low, then we don't care (Because of the expectation)
- If q is high and p is high, good :)
- If q is high and p is low, bad :(

4.3 Evidence Lower Bound

Actually we can not minimize KL divergence. But we can minimize another function which is equal to this. This is evidence lower bound (ELBO).

4.3.1 Jensen's Inequality

Jensen's inequality are widely used in EM algorithm. In convex function, we have :

$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2)$$

In the context of probability theory, if X is a random variable, and φ is a convex function, then:

$$\varphi(E[x]) \leq E[\varphi(x)]$$

Back to the problem, we have observations $x^1, x^2, ..., x^n$, we want $p(x^i)$ get the max probability. Using MLE on it, which is the sum of the log-likelihood,

$$logp_\theta(x^1, x^2, ..., x^n) = \sum_{i=1}^{N} logp_\theta(x^i)$$

and

$$\begin{aligned} logp(x) &= log \int_z p(x,z) \\ &= log \int_z p(x,z) \frac{q(z)}{q(z)} \\ &= log \left(E_q \left[\frac{p(x,z)}{q(z)} \right] \right) \\ &\geq E_q \left[log \frac{p(x,z)}{q(z)} \right] \\ &\geq E_q[logp(x,z)] - E_q[logq(z)] \end{aligned}$$

Note the second term is the entropy
But what does this have to do with the KL divergence?

4.3.2 KL Transformation

As mentioned, we want $q(z)$ and $q(z|x)$ are close to each other:

$$\begin{aligned} KL(q(z)||p(z|x)) &= E_q \left[log \frac{q(z)}{p(z|x)} \right] \\ &= E_q[logq(z)] - E_q[logp(z|x)] \end{aligned}$$

As we know,

$$p(z|x) = \frac{p(z,x)}{p(x)}$$

so

$$\begin{aligned} KL(q(z)||p(z|x)) &= E_q[logq(z)] - E_q[logp(z,x)] + E_q[logp(x)] \\ &= -(E_q[logp(z,x)] - E_q[logq(z)]) + logp(x) \end{aligned}$$

The first term is ELBO we just met.
The formula can also be written as :

$$logp(x) = KL(q(z)||p(z|x)) + (E_q[logp(z,x)] - E_q[logq(z)])$$

As I mentioned before, For two different distributions, KL divergence is always non-negative. and $p(x)$ is the observation evidence, which is fixed. So minimizing the KL divergence is the same as maximizing the ELBO. This is also called as the variational lower bound.

4.3.3 Relationship with EM

EM algorithm is also known as a famous method to find the distirbutions of latent varibales. Unlike variational inference we are going to talk about, EM algorithm use the fact that ELBO is equal to the $p(x)$ when $q(z) = p(z|x)$. EM **alternates** between computing $p(z|x)$ (E step), and optimizing it with respect to the model parameters(M step). The biggest difference is EM assume $p(z|x)$ is computable and fix the parameter, use it, while variational inference use bayesian setting and apply to the models we can not compute.

* EM is out the scope of this article, I don't want to go into too detail about it. Actually the formula below explains pretty clear.
E-step:

$$q(z) := p(z|x;\theta)$$

M-step:

$$\theta := \arg\max_{\theta} -KL(q(z)||p(z|x;\theta))$$

4.4 Mean Field Theory

Mean field theory is also called **self-consistent field theory**. It studies the behavior of large and complex stochastic models by studying a simpler model.Such models consider a large number of small individual components that interact with each other.

The effect of all the other individuals on any given individual is approximated by a single averaged effect, thus reducing a many-body problem to a one-body problem.

4.4.1 Mean Field Approximation

We assume each variable is independent.Using this theory, we can write:

$$q(z_{1:m}) = \prod_{i=1}^{m} q(z_i)$$

$$E_q[logq(z_{1:m})] = \sum_{j=1}^{m} E_{q_j}[logq(z_j)]$$

Also we have the chain rule

$$p(z_{1:m}, x_{1:n}) = p(x_{1:n}) \prod_{j=1}^{m} p(z_j|z_{1:(j-1)}, x_{1:n})$$

4.4.2 Mean Field Method

Note that the order of j is irrelevant.Based on this theory, we can rewrite the lower bound as:

$$\mathcal{L} = \sum_{j=1}^{m} E_q[logp(z_j|z_{1:(j-1)}, x_{1:n})] - E_{q_j}[logq(z_j)]$$

Consider the variable z_j comes last:

$$\mathcal{L} = logp(x_{1:n}) + E_q[logp(z_j|z_{-j}, x)] - E_{q_j}[logq(z_j)]$$

And we can remove the first term because it's irrevalte to $q(z_j)$,the $\mathcal{L}$ can be written as

$$\begin{aligned} argmax_{q_j}\mathcal{L} &= E_q[logp(z_j|z_{-j}, x)] - E_{q_j}[logq(z_j)] \\ &= \int q(z_j)E_{-j}[logp(z_j|z_{-j}, x)]dz_j - \int q(z_j)logq(z_j)dz_j \end{aligned}$$

4.5 Coordinate Ascent Variational Inference

From here we can use Lagrange multipliers. Let's treat $q(z_j)$ as $f(x)$. For simplicity, I convert the formula into this:

$$\frac{d\mathcal{L}}{dq(z_j)} = \frac{d[\int Kf(x)dx - \int f(x)logf(x)dx]}{d[f(x)]} = 0$$

this is equal to:

$$\frac{d[\int Kf(x)dx - \int f(x)logf(x)dx]}{dx} \times \; dxd[f(x)] = 0$$

$$[Kf(x) - f(x)logf(x)] \times \frac{1}{f'(x)} = 0$$

which is :

$$Kf'(x) - [f'(x)logf(x) + f(x)\frac{1}{f(x)}f'(x)] = 0$$

and:

$$K - logf(x) - 1 = 0$$

which means the argmax of ELBO can be find at:

$$E_{-j}[logp(z_j|z_{-j},x)] - logq(z_j) - 1 = 0$$

$$log\frac{e^{E_{-j}[logp(z_j|z_{-j},x)]}}{q(z_j)} = log_e e$$

Or you can simply view it as $y - x - 1 = 0$. This lead to the conclusion:

$$q^*(z_j) \propto exp\{E_{-j}[logp(z_j|z_{-j},x)]\}$$

since $p(z_j|z_{-j},x) = \frac{p(z_j,z_{-j},x)}{p(z_{-j},x)}$, and $p(z_{-j})$ does not depend on z_j we can equivalenty write:

$$q^*(z_j) \propto exp\{E_{-j}[logp(z_j,z_{-j},x)]\}$$

Algorithm 1 Coordinate Ascent Variational Inference

```
Input: A model $p(x,z)$, a dataset x
Output: A variational density $q(z) = \prod_{j=1}^{m} q_j(z_j)$
  init: variational factors $q_j(z_j)$
  while ELBO has not converged do
    for all $j \in \{1,...,m\}$ do
      set $q_j(z_j) \propto exp\{E_{-j}[logp(z_j,z_{-j},x)]\}$
    end for
    compute $ELBO = E_q[logp(z,x)] - E_q[logq(z)]$
  end while
```

Note that there is generally no guarantee of convexity of ELBO, this coordinate ascent procedure converges to a local maximum.

Finally, We can find this method is closely related to Gibbs sampling. Actually Gibbs sampling is a very classical approximate inference method. In variational inference, we take the expected log and set each variables variational factor iteratively.

5 Latent Dirichlet Allocation

5.1 Introduction

LDA is a well known algorithm for topic modeling. You can view LDA as a bayesian version of PLSA (probabilistic latent semantic analysis).