Machine learning - Statistics

Yonghoon Dong

March 8, 2024

Model fitting: Estimating parameters

The process of estimating θ from $\mathcal{D}$ is called **model fitting**, or **training** which is at the heart of machine learning. This can be done by

$$\hat{\theta} = \underset{\theta}{\arg\min}\, \mathcal{L}(\theta)$$

where $\mathcal{L}(\theta)$ is some kind of loss function or objective function.

Maximum likelihood estimation (MLE)

Definition (Maximum likelihood estimation)

$$\hat{\theta}_{\text{mle}} = \arg\max_{\theta} p(\mathcal{D}|\theta)$$

Normally, we assume the training examples are independently sampled from the same distribution, so

$$p(\mathcal{D}|\theta) = \prod_{n=1}^{N} p(y_n|x_n, \theta)$$

For convenience, we normally use **log likelihood**

$$l(\theta) = \log p(\mathcal{D}|\theta) = \sum_{n=1}^{N} \log p(y_n|x_n, \theta)$$

Maximum likelihood estimation (MLE)

Since most optimization algorithms are designed to minimize cost functions, we can redefine the **objective function** to be the **negative log likelihood**

$$\mathrm{NLL}(\theta) = -\sum_{n=1}^{N} \log p(y_n | x_n, \theta)$$

In this case,

$$\mathcal{L}(\theta) = \mathrm{NLL}(\theta)$$

Maximum likelihood estimation (MLE)

If the model is **unsupervised**, it becomes

$$\hat{\theta}_{\text{mle}} = \arg\min_{\theta} - \sum_{n=1}^{N} \log p(y_n|\theta)$$

since we have output y_n but no inputs x_n

Maximum likelihood estimation (MLE)

Alternatively we may want to **maximize the joint likelihood of inputs and outputs**. In this case it becomes

$$\hat{\theta}_{\text{mle}} = \arg\min_{\theta} - \sum_{n=1}^{N} \log p(y_n, x_n | \theta)$$

KL divergence and MLE

Suppose we want to find the distribution q that is as close as possible to p, as measured by KL divergence

$$q^* = \arg\min_q D_{\mathrm{KL}}(p\|q) = \arg\min_q \int p(x)\log p(x)dx - \int p(x)\log q(x)dx$$

Now suppose p is the empirical distribution, which puts a probability only on the observed training data and zero mass everywhere else

$$p_{\mathcal{D}}(x) = \frac{1}{N}\sum_{n=1}^{N}\delta(x - x_n)$$

KL divergence and MLE

$$
\begin{aligned}
D_{\mathrm{KL}}(p\|q) &= C - \int p_{\mathcal{D}}(x)\log q(x)dx \\
&= C - \int \frac{1}{N}\sum_{n=1}^{N}\delta(x-x_n)\log q(x)dx \\
&= C - \frac{1}{N}\sum_{n=1}^{N}\log q(x_n)
\end{aligned}
$$

where $C = \int p(x)\log p(x)dx$

KL divergence and MLE

Then,

$$
\begin{aligned}
q^* &= \arg\min_q D_{\mathrm{KL}}(p \| q) \\
&= \arg\min_q C - \frac{1}{N} \sum_{n=1}^{N} \log q(x_n) \\
&= \arg\min_q - \frac{1}{N} \sum_{n=1}^{N} \log q(x_n)
\end{aligned}
$$

Thus we can see that minimizing KL divergence to the empirical distribution is equivalent to maximizing likelihood.

KL divergence and MLE

This perspective points out the flaw with likelihood-based training because it puts too much weight on the training set. To alleviate this problem, we could use the following techniques:

1. We could smooth the empirical distribution using kernel density estimation
2. Alternatively, we can use data augmentation, which is a way of perturbing the observed data samples in way that we believe reflects plausible "natural variation"

Empirical risk minimization

We can generalize MLE by replacing log loss term with any other loss function

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{n=1}^{N} \ell(y_n, \theta; x_n)$$

This is known as **empirical risk minimization**.

Regularization

A fundamental problem with MLE is that it will try to pick parameters that minimize loss on the training set, but this may **not** result in a model that has low loss on future data. This is called **overfitting**

Regularization

The main solution to overfitting is to use **regularization**. So we optimize an objective of the form

$$\mathcal{L}(\theta; \lambda) = \left[\frac{1}{N} \sum_{n=1}^{N} \ell(y_n, \theta; x_n) \right] + \lambda C(\theta)$$

Maximum a posterior estimation (MAP)

Commonly we use $C(\theta) = -\log p(\theta)$, where $p(\theta)$ is the **prior** for θ If ℓ is the log loss and $\lambda = 1$, the regularized objective becomes

$$\mathcal{L}(\theta; \lambda) = -\frac{1}{N} \sum_{n=1}^{N} \log p(y_n | x_n, \theta) - \log p(\theta) = -[\log p(\mathcal{D}|\theta) + \log p(\theta)]$$

Therefore, minimizing this is equivalent to maximizing the log posterior:

$$\begin{aligned} \hat{\theta} &= \arg\min_{\theta} \mathcal{L}(\theta; \lambda) \\ &= \arg\max_{\theta} [\log p(\mathcal{D}|\theta) + \log p(\theta)] \\ &= \arg\max_{\theta} \log p(\theta|\mathcal{D}) \text{ (Since } p(\mathcal{D}) \text{ is independent from } \theta) \end{aligned}$$

This is known as MAP estimation, which stands for **maximum a posterior estimation**.