

[AIMLBD] MACHINE LEARNING, BIG DATA, ARTIFICIAL INTELLIGENCE per medicina e chirurgia high tech

L03: Maximum Likelihood Estimation

Dott. Giorgio De Magistris

demagistris@diag.uniroma1.it

CORSO DI LAUREA IN MEDICINA E CHIRURGIA HIGH TECH



SAPIENZA
UNIVERSITÀ DI ROMA

I3S

FACOLTÀ DI INGEGNERIA DELL'INFORMAZIONE, INFORMATICA E STATISTICA

DIAG

DIPARTIMENTO DI INGEGNERIA INFORMATICA, AUTOMATICA E GESTIONALE

TUTTI I DIRITTI RELATIVI AL PRESENTE MATERIALE DIDATTICO ED AL SUO CONTENUTO SONO RISERVATI A SAPIENZA E AI SUOI AUTORI (O DOCENTI CHE LO HANNO PRODOTTO). È CONSENTITO L'USO PERSONALE DELLO STESSO DA PARTE DELLO STUDENTE A FINI DI STUDIO. NE È VIETATA NEL MODO PIÙ ASSOLUTO LA DIFFUSIONE, DUPLICAZIONE, CESSIONE, TRASMISSIONE, DISTRIBUZIONE A TERZI O AL PUBBLICO PENA LE SANZIONI APPLICABILI PER LEGGE

Loss Functions

- Previously we saw that in general a ML model is trained to minimize an error function (also called **loss** function)
- But how do these loss functions came from? How can we derive the right loss function for the specific problem?
- We can answer by giving a probabilistic interpretation to the question

Likelihood

- Given the training data $X = \{x_1, \dots, x_m\}$ sampled from the data probability distribution $x_i \sim p_{\text{data}}(x)$
- Given a parametric model $p_{\text{model}}(x; \theta)$
- What are the values of the parameters that maximize p_{model} when the x values come from the distribution p_{data} ?

$$\Theta_{ML} = \underset{\Theta}{\operatorname{argmax}} p_{\text{model}}(X; \Theta)$$

- That, tanks to the i.i.d. assumption (the samples in the training set are independent) can be reformulated as

$$\Theta_{ML} = \underset{\Theta}{\operatorname{argmax}} \prod_{i=1}^m p_{\text{model}}(x_i; \Theta)$$

Maximum Likelihood Estimation

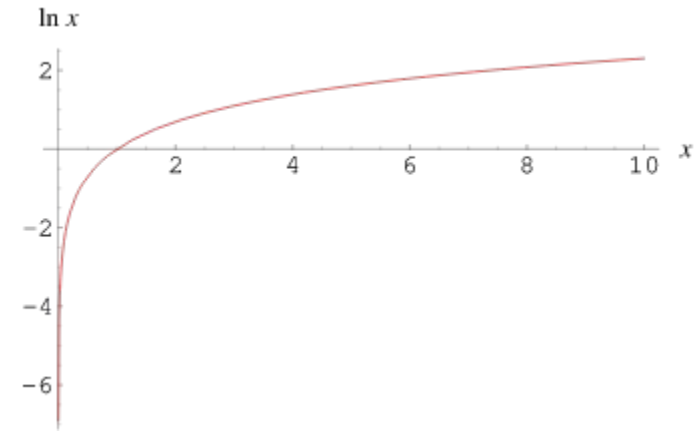
- The values of the parameters that maximize the likelihood Θ_{ML} are called Maximum Likelihood Estimation

$$\Theta_{ML} = \underset{\Theta}{\operatorname{argmax}} \prod_{i=1}^m p_{\text{model}}(x_i; \Theta)$$

- Sometimes we use to maximize the log (logarithm with base e) likelihood function instead of the likelihood

$$\Theta_{ML} = \underset{\Theta}{\operatorname{argmax}} \sum_{i=1}^m \log(p_{\text{model}}(x_i; \Theta))$$

- The logarithm is a monotonically increasing function so the argmax does not change (even though the max changes)



Remember

$\log(ab) = \log(a) + \log(b)$
 $\log(a/b) = \log(a) - \log(b)$
 $\log(a^c) = c \log(a)$

MLE for Supervised Learning

- We saw how to estimate the maximum likelihood parameters when a generic parametric probability distribution p_{model} tries to approximate the real data distribution p_{data}
- We can specialize MLE for different probability distributions, for example in supervised learning we often want to predict the targets Y given the inputs X
- In this case the MLE becomes

$$\Theta_{ML} = \underset{\Theta}{\operatorname{argmax}} p_{\text{model}}(Y|X; \Theta)$$

MLE for Regression

- Assuming that p_{model} is a parametric normal distribution:

$$p_{\text{model}}(y_i|x_i; \Theta) = N(\mu = f(x_i, \Theta), \sigma^2 = \text{const})$$

- We obtain the following Maximum Likelihood Estimation

$$\Theta_{ML} = \underset{\Theta}{\operatorname{argmax}} \sum_{i=1}^m \log\left(\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2} \frac{(y_i - f(x_i, \Theta))^2}{\sigma^2}}\right)$$

- And after further simplifications and by ignoring the terms that do not depend on θ , we obtain the well known **Mean Squared Error**

$$\Theta_{ML} = \underset{\Theta}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^m (y_i - f(x_i, \Theta))^2$$

MLE for binary classification

- For binary classification we assume that the distribution p_{model} is a parametric Bernoulli distribution

$$p_{\text{model}}(y_i|x_i; \Theta) = \begin{cases} p_i & \text{if } y_i = 1 \\ 1 - p_i & \text{if } y_i = 0 \end{cases} = p_i^{y_i} (1 - p_i)^{(1-y_i)} \quad p_i = \frac{1}{1 + e^{-f(x_i, \Theta)}} = \sigma(f(x_i, \Theta))$$

- We obtain the following the following MLE:

$$\Theta_{ML} = \underset{\Theta}{\operatorname{argmax}} \sum_{i=1}^m \log(\sigma(f(x_i, \Theta))^{y_i} (1 - \sigma(f(x_i, \Theta))^{(1-y_i)})$$

- That, after switching the sign and turning the argmax into an argmin, becomes the well known binary cross entropy loss

$$\Theta_{ML} = \underset{\Theta}{\operatorname{argmin}} - \frac{1}{m} \sum_{i=1}^m y_i \log(\sigma(f(x_i, \Theta))) + (1 - y_i) \log(1 - \sigma(f(x_i, \Theta)))$$

MLE for multi-class classification

- Similarly for a k classes classification problem we assume that the distribution p_{model} is a multinoulli distribution (also known as categorical distribution)

$$p_{\text{model}}(y_i|x_i; \Theta_i) = \begin{cases} p_1 & \text{if } y_i = 1 \\ \dots & \\ p_k & \text{if } y_i = k \end{cases} = \prod_{j=1}^k p_j^{\delta_j(y_i)} \quad p_j = \frac{e^{f(x_j; \Theta)}}{\sum_{i=1}^k e^{f(x_i; \Theta)}} = \hat{y}_j$$

$$\text{with } \delta_j(y_i) = \begin{cases} 1 & \text{if } y_i = j \\ 0 & \text{otherwise} \end{cases}$$

- And by maximizing the log likelihood we obtain the **categorical cross entropy loss**

$$\begin{aligned} \Theta_{ML} &= \underset{\Theta}{\operatorname{argmax}} \sum_{i=1}^m \log\left(\prod_{j=1}^k p_j^{\delta_j(y_i)}\right) = \underset{\Theta}{\operatorname{argmax}} \sum_{i=1}^m \sum_{j=1}^k \delta_j(y_i) \log(p_j) \\ &= \underset{\Theta}{\operatorname{argmin}} -\frac{1}{m} \sum_{i=1}^m \sum_{j=1}^k \delta_j(y_i) \log(\hat{y}_j) \end{aligned}$$

Slides distribuite con Licenza Creative Commons (CC BY-NC-ND 4.0) Attribuzione - Non commerciale - Non opere derivate 4.0 Internazionale

PUOI CONDIVIDERLE ALLE SEGUENTI CONDIZIONI

(riprodurre, distribuire, comunicare o esporre in pubblico, rappresentare, eseguire e recitare questo materiale con qualsiasi mezzo e formato)

Attribuzione*

Devi riconoscere una menzione di paternità adeguata, fornire un link alla licenza e indicare se sono state effettuate delle modifiche. Puoi fare ciò in qualsiasi maniera ragionevole possibile, ma non con modalità tali da suggerire che il licenziante avalli te o il tuo utilizzo del materiale.

Non Commerciale

Non puoi utilizzare il materiale per scopi commerciali.

Non opere derivate

Se remixi, trasformi il materiale o ti basi su di esso, non puoi distribuire il materiale così modificato.

Divieto di restrizioni aggiuntive

Non puoi applicare termini legali o misure tecnologiche che impongano ad altri soggetti dei vincoli giuridici a questa licenza