Derivation of loglikelihood inside, spoiler alert

discussion posted 5 days ago by Cool7 (Community TA)

As title. This is easier than last one. Just put it here in case somebody interested. I'm practicing my latex writing, lol.

$$\log(\blacktriangleleft\prod\blacktriangleright)$$

$$=\sum_{t=1}^{n}(\log(\mathcal{N}(y_t|\theta x_t,\sigma^2))+\log(\mathcal{N}(\theta|0,\lambda^{-1})))$$

$$=n\log(\frac{1}{\sigma\sqrt{2\pi}})+\sum_{t=1}^{n}log(e^{-\frac{(y_t-\theta x_t)^2}{2\sigma^2}})+n\log(\sqrt{\frac{\lambda}{2\pi}})+\sum_{t=1}^{n}log(e^{-\frac{\lambda\|\theta\|^2}{2}})$$

$$=\sum_{t=1}^{n}(-\frac{1}{2\sigma^2}(y_t-\theta x_t)^2-\frac{\lambda}{2}\|\theta\|^2)+n\log(\frac{1}{\sigma\sqrt{2\pi}})+n\log(\sqrt{\frac{\lambda}{2\pi}})$$

$$=\sum_{t=1}^{n}-\frac{1}{2\sigma^2}(y_t-\theta x_t)^2-\frac{1}{2}\lambda\|\theta\|^2+\text{constant}$$

My understanding is

- First term is related to posterior distribution, it represents the accuracy of the estimation/training loss/bias.
- Second term is related to prior distribution, it represents the regularization(recall we imposed it on) / variance.

Thus λ as hyper parameter is to adjust the weights between bias and variance, inline with the error decomposition discussed a few pages before.

Related to: Unit 2 Nonlinear Classification, Linear regression, Collaborative Filtering (2 weeks):Lecture 5. Linear Regression / 9. Closing Comment
This post is visible to everyone.

Add a Response

1 response

Alexander_Konstantinidis
4 days ago

Another way to view this, is to consider λ as expressing the degree of our certainty (prior belief) that there is no real explanatory value in the model or stated differently very few if any of the predictors truly matter. (This is because λ is the inverse of the variance of probabilistic theta). The higher the λ the more evidence will be required to arrive to a complex model and vice versa.

Indeed, this is a very interesting interpretation. In the extreme case, where lambda is infinity, it means your prior belief is so strong that no matter what data is presented, the hard coded parameters do not change. On the other extreme, when lambda is 0, variance is infinity and thus you don't have a prior belief. Data takes control of everything, even if there're a lot of noise. So a moderate lambda lets the model to learn from data, but regularizes the parameters so that do not deviate too much from the prior belief.

posted 4 days ago by FutureStar

Just extending to multidimensional data, when we have datapoints $(X_i, Y_i)_{i=1...n}$

For ridge regression, we have, $Y = X\beta + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ and $Y \sim \mathcal{N}(X\beta, \sigma^2 I)$, with prior on $\beta \sim \mathcal{N}(0, \lambda^{-1})$, s.t., $p(Y) \propto \frac{1}{\sigma} e^{-(Y-X\beta)^T(Y-X\beta)/\sigma^2}$ and $p(\beta) \propto e^{-\lambda\beta^T\beta}$,

$\hat{\beta}_{MAP} = \underset{\beta}{\operatorname{argmax}}\, log\left(\prod_{i=1}^n p(X_i, Y_i|\beta, \sigma^2) \,.\, p(\beta)\right) = \underset{\beta}{\operatorname{argmax}} \sum_{i=1}^n \left(-(Y_i - X_i\beta)^2 - \lambda\beta^T\beta\right) = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^n (Y_i - X_i\beta)^2 + \lambda\|\beta\|^2$

posted less than a minute ago by sandipan_dey

Add a comment

Showing all responses

Add a response:

Preview

Submit