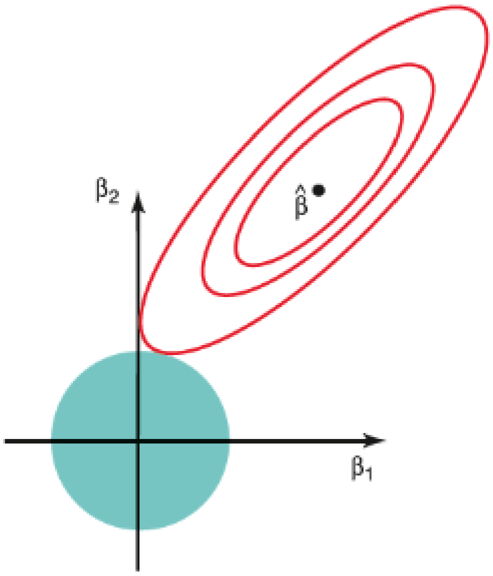
RAHUL JAIN

Follow 14 Followers About

L2 Regularisation: Maths

RAHUL JAIN Feb 8, 2020 · 3 min read

L2 regularisation, one of the most popular techniques in Machine Learning, is a method to reduce the variance of the model and increase the bias to make the model more generalisable.



Geometrical Representation of L2 Regularisation

L2 regularisation is a MAP estimation with Gaussian prior probabilities. Due to the imposed prior, the model can generalise the data well by scaling the weights as per their significance.

The L2 regularised objective function and its gradient is given by,

$$\nabla_{\boldsymbol{w}} \tilde{J}(\boldsymbol{w}; \boldsymbol{X}, \boldsymbol{y}) = \alpha \boldsymbol{w} + \nabla_{\boldsymbol{w}} J(\boldsymbol{w}; \boldsymbol{X}, \boldsymbol{y}).$$

From Taylor series expansion, we will make a quadratic approximation of the unregularized objective function around the point ***w****. ***w**** is the minimiser of the unregularized objective function.

$$\hat{J}(\boldsymbol{\theta}) = J(\boldsymbol{w}^*) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}^*)^\top \boldsymbol{H}(\boldsymbol{w} - \boldsymbol{w}^*),$$

$$\text{Since} \quad \boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} J(\boldsymbol{w})$$

$$\nabla_{\boldsymbol{w}} J(\boldsymbol{w}; \boldsymbol{X}, \boldsymbol{y}) \Big|_{\boldsymbol{w}^*} = 0$$

Thus, the regularised objective function and its gradient becomes,

$$\tilde{J}(\boldsymbol{w}) = \frac{\alpha}{2}\boldsymbol{w}^\top \boldsymbol{w} \; + \; J(\boldsymbol{w}^*) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}^*)^\top \boldsymbol{H}(\boldsymbol{w} - \boldsymbol{w}^*),$$

$$\nabla_{\boldsymbol{w}} \tilde{J}(\boldsymbol{w}) = \alpha \boldsymbol{w} + \boldsymbol{H}(\boldsymbol{w} - \boldsymbol{w}^*)$$

Let **w˜** represents the minimiser of the regularised objective function. The slope at this point is equal to zero.

$$\begin{aligned} \nabla_{\boldsymbol{w}} \tilde{J}(\boldsymbol{w}) &= 0 \\ \alpha \tilde{\boldsymbol{w}} + \boldsymbol{H}(\tilde{\boldsymbol{w}} - \boldsymbol{w}^*) &= 0 \\ (\boldsymbol{H} + \alpha \boldsymbol{I})\tilde{\boldsymbol{w}} &= \boldsymbol{H}\boldsymbol{w}^* \\ \tilde{\boldsymbol{w}} &= (\boldsymbol{H} + \alpha \boldsymbol{I})^{-1} \boldsymbol{H}\boldsymbol{w}^*. \end{aligned}$$

Since H is positive semi-definite, real and symmetric matrix, it can be decomposed into a diagonal matrix Λ and an orthonormal basis of eigenvectors, Q, such that,

$$\begin{aligned}\tilde{w} &= (Q\Lambda Q^\top + \alpha I)^{-1} Q\Lambda Q^\top w^* \\ &= \left[Q(\Lambda + \alpha I)Q^\top\right]^{-1} Q\Lambda Q^\top w^* \\ &= Q(\Lambda + \alpha I)^{-1}\Lambda Q^\top w^*.\end{aligned}$$

The effect of weight decay is to rescale w∗ along the axes defined by the eigenvectors of H. The component of w~ along the ith eigenvector of H is rescaled.

$$\tilde{w}_i = \frac{\lambda_i}{\lambda_i + \alpha} w_i^*$$

If λi >> α, the effect of regularisation is relatively small. However, components with λi << α will be shrunk to have nearly zero magnitudes. Only directions along which the parameters contribute significantly to reducing the objective function are preserved relatively intact. In other unimportant instructions, indicated by a small eigenvalue of the Hessian, weight vectors are decayed away through the use of the regularisation throughout the training.

The regularisation constant, α is a hyperparameter and is tuned to get the best results. As the value of α increases, weights are decayed more.

References:

Deep Learning (Adaptive Computation and Machine Learning series) — By Ian Goodfellow, Yoshua Bengio, Aaron Courville

L2 Regularization Machine Learning Maths Gradient Descent Proof