TTIC 31230, Fundamentals of Deep Learning

David McAllester, Winter 2020

Implicit Regularization

Implicit Regularization

Any stochastic learning algorithm, such as SGD, determines a stochastic mapping from training data to models.

The algorithm, especially with early stopping, can implicitly incorporate a preference or bias for models.

Implicit Regularization in Linear Regression

Linear regression with many more parameters than data points has many solutions.

But SGD converges to the minimum norm solution.

Implicit Regularization in Linear Regression

For linear regression SGD maintains the invariant that Φ is a linear combination of the (small number of) training vectors.

Any zero-loss (squared loss) solution can be projected on the span of training vectors to give a smaller (or no larger) norm solution.

It can be shown that when the training vectors are linearly independent any zero loss solution in the span of the training vectors is a least-norm solution.

A PAC-Bayes Analysis of Implicit Regularization

Let A be an algorithm that stochastically maps a training set to a model parameter vector.

In the case of SGD we take the stochasticity to include both the random initialization and the random sequence of training batches.

A PAC-Bayes Analysis of Implicit Regularization

Let $p(\Phi|A, \text{Train})$ be the probability desity on parameter vectors defined by the stochasticity of the algorithm A.

Define

$$p(\Phi|A) = E_{\left(\text{Train}\sim\text{Pop}^{N_{\text{Train}}}\right)} p(\Phi|A, \text{Train})$$

The density $p(\Phi|A)$ is independent of any choice of training data and can be used as the prior in a BAC-Bayesian bound.

A PAC-Bayes Analysis of Implicit Regularization

$$\mathcal{L}(\Phi, \text{Pop}) = E_{\langle x, y \rangle \sim \text{Pop}}\, \mathcal{L}(\Phi, x, y)$$

$$\mathcal{L}(\Phi, \text{Train}) = E_{\langle x, y \rangle \sim \text{Train}}\, \mathcal{L}(\Phi, x, y)$$

$$\mathcal{L}(A) = E_{\left(\text{Train} \sim \text{Pop}^{N_{\text{Train}}}\right)}\, E_{\Phi \sim p(\Phi \,|A,\, \text{Train})}\, \mathcal{L}(\Phi, \text{Pop})$$

$$\hat{\mathcal{L}}(A) = E_{\left(\text{Train} \sim \text{Pop}^{N_{\text{Train}}}\right)}\, E_{\Phi \sim p(\Phi \,|A,\, \text{Train})}\, \mathcal{L}(\Phi, \text{Train})$$

A PAC-Bayes Analysis of Implicit Regularization

$$\mathcal{L}(A) \leq \frac{10}{9}\left(\begin{array}{l} \hat{\mathcal{L}}(A) \\ \\ +\ \frac{5L_{\max}}{N_{\text{Train}}}\left(\begin{array}{l} E_{\text{Train}\sim\text{Pop}^{N_{\text{Train}}}} \\ \\ \qquad KL(p(\Phi|A,\text{Train}),p(\Phi|A)) \\ \\ +\ \ln\frac{1}{\delta} \end{array}\right) \end{array}\right)$$

There is no obvious way to calculate this guarantee.

However, it can be shown that $p(\Phi|A)$ is the optimal PAC-Bayeisan prior for algorithm A making this the best possible PAC-Bayesian bound for A.

END