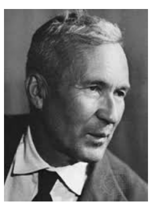
TTIC 31230, Fundamentals of Deep Learning

David McAllester, Winter 2020

PAC-Bayesian Learning Theory

Chomsky vs. Kolmogorov and Hinton

Noam Chomsky: Natural language grammar cannot be learned by a universal learning algorithm. This position is supported by the "no free lunch theorem".




Andrey Kolmogorov, Geoff Hinton: Universal learning algorithms exist. This position is supported by the "free lunch theorem".

The No Free Lunch Theorem

Without prior knowledge, such as universal grammar, it is impossible to make a prediction for an input you have not seen in the training data.

Proof: Select a predictor h uniformly at random from all functions from $\mathcal{X}$ to $\mathcal{Y}$ and then take the data distribution to draw pairs $(x, h(x))$ where x is drawn uniformly from $\mathcal{X}$. No learning algorithm can predict $h(x)$ where x does not occur in the training data.

The Free Lunch Theorem

Consider a classifier f written in C++ with an arbitrarily large standard library.

Let $|f|$ be the number of bits needed to represent f.

The Free Lunch Theorem

$$0 \leq \mathcal{L}(h, x, y) \leq L_{\mathrm{max}}$$

$$\mathcal{L}(h) = E_{(x,y)\sim\mathrm{Pop}}\, \mathcal{L}(h, x, y)$$

$$\hat{\mathcal{L}}(h) = E_{(x,y)\sim\mathrm{Train}}\, \mathcal{L}(h, x, y)$$

Theorem: With probability at least $1 - \delta$ over the draw of the training data the following holds simultaneously for all f.

$$E(f) \leq \frac{10}{9}\left(\hat{E}(f) + \frac{5L_{\mathrm{max}}}{N_{\mathrm{Train}}}\left((\ln 2)|f| + \ln\frac{1}{\delta}\right)\right)$$

Free Lunch Theorem (Probability Form)

Code length is inter-convertable with with probability.

$$P(h) = 2^{-|h|} \quad \text{or} \quad |h| = -\log_2 P(h)$$

Instead of fixing the language (e.g., C++ with a large library) we fix a prior $P(h)$.

Theorem: With probability at least $1 - \delta$ over the draw of training data the following holds simultaneously for all h.

$$\mathcal{L}(h) \leq \frac{10}{9}\left(\hat{\mathcal{L}}(h) + \frac{5L_{\max}}{N_{\text{Train}}}\left(\ln \frac{1}{P(h)} + \ln \frac{1}{\delta}\right)\right)$$

Proof

Define

$$\epsilon(h) = \sqrt{\frac{2\mathcal{L}(h)\left(\ln \frac{1}{P(h)} + \ln \frac{1}{\delta}\right)}{L_{\max} N_{\mathrm{Train}}}}.$$

By the relative Chernov bound we have

$$P_{\mathrm{Train} \sim \mathrm{Pop}}\left(\frac{\hat{\mathcal{L}}(h)}{L_{\max}} \leq \frac{\mathcal{L}(h)}{L_{\max}} - \epsilon(h)\right) \leq e^{-N_{\mathrm{Train}} \frac{\epsilon(h)^2 L_{\max}}{2\mathcal{L}(h)}} = \delta P(h).$$

Proof

$$P_{\text{Train}\sim\text{Pop}}\left(\hat{\mathcal{L}}(h) \leq \mathcal{L}(h) - L_{\max}\epsilon(h)\right) \leq \delta P(h).$$

$$P_{\text{Train}\sim\text{Pop}}\left(\exists h\ \hat{\mathcal{L}}(h) \leq \mathcal{L}(h) - L_{\max}\epsilon(h)\right) \leq \sum_h \delta P(h) = \delta$$

$$P_{\text{Train}\sim\text{Pop}}\left(\forall h\ \mathcal{L}(h) \leq \hat{\mathcal{L}}(h) + L_{\max}\epsilon(h)\right) \geq 1 - \delta$$

Proof

$$\mathcal{L}(h) \leq \widehat{\mathcal{L}}(h) + L_{\max} \sqrt{\frac{\mathcal{L}(h)}{L_{\max}} \left(\frac{2\left(\ln \frac{1}{P(h)} + \ln \frac{1}{\delta}\right)}{N_{\mathrm{Train}}} \right)}$$

using

$$\sqrt{ab} = \inf_{\lambda > 0} \frac{a}{2\lambda} + \frac{\lambda b}{2}$$

we get

$$\mathcal{L}(h) \leq \widehat{\mathcal{L}}(h) + \frac{\mathcal{L}(h)}{2\lambda} + \frac{\lambda L_{\max}\left(\ln \frac{1}{P(h)} + \ln \frac{1}{\delta}\right)}{N_{\mathrm{Train}}}$$

Proof

$$\mathcal{L}(h) \leq \widehat{\mathcal{L}}(h) + \frac{\mathcal{L}(h)}{2\lambda} + \frac{\lambda L_{\max}\left(\ln \frac{1}{P(h)} + \ln \frac{1}{\delta}\right)}{N_{\mathrm{Train}}}$$

Solving for $\mathcal{L}(h)$ yields

$$\mathcal{L}(h) \leq \frac{1}{1 - \frac{1}{2\lambda}}\left(\hat{\mathcal{L}}(h) + \frac{\lambda L_{\max}}{N_{\mathrm{Train}}}\left(\ln \frac{1}{P(h)} + \ln \frac{1}{\delta}\right)\right)$$

Setting $\lambda = 5$ brings the leading factor to $10/9$ which seems sufficiently close to 1 that larger values of λ need not be considered.

A Model Compression Guarantee

Let $|\Phi|$ be the number of bits used to represent Φ under some fixed compression scheme.

Let $P(\Phi) = 2^{-|\Phi|}$

$$\mathcal{L}(\Phi) \leq \frac{10}{9}\left(\hat{\mathcal{L}}(\Phi) + \frac{5L_{\mathrm{max}}}{N_{\mathrm{Train}}}\left((\ln 2)|\Phi| + \ln\frac{1}{\delta}\right)\right)$$

A Bound for Continuous Densities

Let p be any "prior" and q be any "posterior" on any (possibly continuous) model space. Define

$$L(q) = E_{h \sim q}\, L(h)$$

$$\hat{L}(q) = E_{h \sim q}\, \hat{L}(h)$$

For any p and any $\lambda > \frac{1}{2}$, with probability at least $1-\delta$ over the draw of the training data, the following holds simultaneously for all q.

$$L(q) \leq \frac{1}{1 - \frac{1}{2\lambda}} \left(\hat{L}(q) + \frac{\lambda L_{\max}}{N_{\mathrm{Train}}} \left(KL(q, p) + \ln \frac{1}{\delta} \right) \right)$$

Adding Noise Simulates Limiting Precision

Assume $0 \leq \mathcal{L}(\Phi, x, y) \leq L_{\max}$.

Define:

$$\mathcal{L}(\Phi) \;=\; E_{\,(x,y)\sim \mathrm{Pop},\; \epsilon \sim \mathcal{N}(0,\sigma)^d}\; \mathcal{L}(\Phi + \epsilon, x, y)$$

$$\hat{\mathcal{L}}(\Phi) \;=\; E_{\,(x,y)\sim \mathrm{Train},\; \epsilon \sim \mathcal{N}(0,\sigma)^d}\; \mathcal{L}(\Phi + \epsilon, x, y)$$

Theorem: With probability at least $1 - \delta$ over the draw of training data the following holds **simultaneously** for all Φ.

$$\mathcal{L}(\Phi) \;\leq\; \frac{10}{9}\left(\hat{\mathcal{L}}(\Phi) + \frac{5L_{\max}}{N_{\mathrm{Train}}}\left(\frac{||\Phi - \Phi_{\mathrm{init}}||^2}{2\sigma^2} + \ln\frac{1}{\delta}\right)\right)$$

Implicit Regularization

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

The algorithm 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 finds 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 no larger norm solution.

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

An Implicit Regularization Generalization Guarantee

Let $\mathcal{H}$ be a discrete set of classifiers.

Let A be an algorithm mapping a training set to a classifier.

Let $P(h|A, \mathrm{Pop})$ be the probability over the draw of the training data that $A(\mathrm{Train}) = h$.

Theorem: With probability at least $1 - \delta$ over the draw of the training data we have

$$\mathrm{Err}(A(\mathrm{Train})) \leq \frac{10}{9}\left(\begin{array}{l} \hat{\mathrm{Err}}(A(\mathrm{Train})) \\ +\frac{5}{N_{\mathrm{Train}}}\left(-\ln P(A(\mathrm{Train})|A, \mathrm{Pop}) + \ln \frac{1}{\delta}\right) \end{array}\right)$$

Non-Vacuous Generalization Guarantees

Model compression has recently been used to achieve "non-vacuous" PAC-Bayes generalization guarantees for ImageNet classification — error rate guarantees less than 1.

Non-Vacuous PAC-Bayes Bounds at ImageNet Scale.

Wenda Zhou, Victor Veitch, Morgane Austern, Ryan P. Adams, Peter Orbanz

ICLR 2019

END