TTIC 31230 Fundamentals of Deep Learning

Regularization and Generalization Problems

Problem 1. The Stationary Points for L_2 Regularization. Consider the regularized objective

$$\Phi^* = \operatorname*{argmin}_\Phi \; E_{(x,y)\sim\mathrm{Train}} \; \left(\mathcal{L}(\Phi,x,y) + \frac{1}{2N_{\mathrm{train}}\sigma^2}||\Phi||^2\right)$$

By setting the gradient of the objective to zero, solve for Φ as a function of the average gradient g defined by

$$g = E_{\langle x,y\rangle\sim\mathrm{Train}} \nabla\Phi\mathcal{L}(\Phi,x,y).$$

Solution:

$$\begin{array}{rcl} & & \nabla_\Phi E_{(x,y)\sim\mathrm{Train}} \; \mathcal{L}(\Phi,x,y) + \frac{1}{2N_{\mathrm{train}}\sigma^2}||\Phi||^2 \\ & = & \left(E_{(x,y)\sim\mathrm{Train}} \; \nabla_\Phi\mathcal{L}(\Phi,x,y)\right) + \frac{1}{N_{\mathrm{train}}\sigma^2}\Phi \\ & = & g + \frac{1}{N_{\mathrm{train}}\sigma^2}\Phi \quad = 0 \\ \Phi & = & N_{\mathrm{train}}\sigma^2 g \end{array}$$

Note that a larger sample size justifies having a larger norm for the parameter vector.

PAC-Bayes Background for the problems 2 through 5. Consider any probability distribution $P(h)$ over a discrete class $\mathcal{H}$. Assume $0 \leq \mathcal{L}(h,x,y) \leq L_{\max}$. Define

$$\begin{array}{rcl} \mathcal{L}(h) & = & E_{(x,y)\sim\mathrm{Pop}} \; \mathcal{L}(h,x,y) \\ \hat{\mathcal{L}}(h) & = & E_{(x,y)\sim\mathrm{Train}} \; \mathcal{L}(h,x,y) \end{array}$$

We now have the theorem that with probability at least $1-\delta$ over the draw of training data the following holds simultaneously for all h.

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

This motivates

$$h^* = \operatorname*{argmin}_h \; \hat{\mathcal{L}}(h) + \frac{5L_{\max}}{N_{\mathrm{train}}}\ln\frac{1}{P(h)} \qquad (2)$$

The Bayesian maximum a-posteriori (MAP) rule is

$$h^* = \operatorname*{argmax}_h \; P(h) \prod_{(x,y)\in\mathrm{Train}} P(y|x,h) \qquad (3)$$

Problem 2. The Meaning of a PAC-Bayes Prior. Consider an optimal hypothysis for the population distribution.

$$h^* = \operatorname*{argmin}_h \; E_{\langle x,\,y\rangle \sim \mathrm{Pop}}\; \mathcal{L}(h,x,y)$$

Equation (1) holds for any prior P. Consider two priors P_{lucky} and P_{unlucky} where we have

$$P_{\mathrm{lucky}}(h^*) >> P_{\mathrm{unlucky}}(h^*)$$

Explain how equation (1) can hold for both of these priors.

Solution: The prior P should be interpreted as saying which hyopthesis will be measured accurately first as N_{train} increases. We can interpret P as a "guess" as to where we think the good hypotheses are. The prior P is not stating any actual propability of where the optimal hypothesis is. We get accurate measurements first for the hypotheses h for which $P(h)$ is large. For P_{unlucky} we get an accurate measureent of $\mathcal{L}(h^*)$ only much later than we do under P_{lucky}.

Problem 3. Code Length as Probability. Assume that a model h is represented by a (compressed) file $|h|$ bits long. Files have a specific length and no file is a proper prefix of any other file. We say that the set of file bit strings is **prefix free**.

(a) Show that for any prefix-free representation of files as bit strings we have the following Kraft inequality where the sum is over all possible files (of unbounded size).

$$\sum_h 2^{-|h|} \leq 1$$

Solution: Consider a probabilistic process which flips an unbiased coin to determine a next bit until it generates a legal file bit string at which point it outputs that file. This process generates file h with probability $2^{-|h|}$ and, by the prefix-free property, all files can be generated by this process. However, it is possible that this process never terminates The Kraft inequality then follows from $\sum_h P(h) \leq 1$ where we also have $P(\mathrm{divergence}) + \sum_h \; P(h) = 1$.

(b) rewrite (1) in terms of $|h|$ where we take $P(h) = 2^{-|h|}$.

Solution:

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

Problem 4. Comparing Bayesian MAP to PAC-Bayes For $\mathcal{L}(h,x,y) = -\ln P(y|x,h)$ (cross entropy loss) rewrite (2) so as to be as similar to (3) as possible. Note that (1) holds independent of any "truth" of the "prior" P.

Solution:

$$\begin{aligned}
& \operatorname*{argmin}_h \left(\frac{1}{N}\sum_{(x,y)\sim\mathrm{Train}} -\ln P(y|x,h)\right) + \frac{5L_{\max}}{N}\ln\frac{1}{P(h)} \\
= \;& \operatorname*{argmax}_h \left(\frac{1}{N}\sum_{(x,y)\sim\mathrm{Train}} \ln P(y|x,h)\right) + \frac{5L_{\max}}{N}\ln P(h) \\
= \;& \operatorname*{argmax}_h \left(\sum_{(x,y)\sim\mathrm{Train}} \ln P(y|x,h)\right) + 5L_{\max}\ln P(h) \\
= \;& \operatorname*{argmax}_h \ln\left(P(h)^{5L_{\max}} \prod_{(x,y)\sim\mathrm{Train}} P(y|x,h)\right) \\
= \;& \operatorname*{argmax}_h \; P(h)^{5L_{\max}} \prod_{(x,y)\sim\mathrm{Train}} P(y|x,h)
\end{aligned}$$

Problem 5. Finite Precision Parameters.

(a) Consider a model where the parameter vector Φ has d parameters each of which is represented by a 16 bit floating point number. Express the bound (1) in terms of the dimension d assuming all parameter vectors are equally likely.

Solution:

$$\mathcal{L}(h) \leq \frac{10}{9}\left(\hat{\mathcal{L}}(h) + \frac{5L_{\max}}{N}\left(16d\ln 2 + \ln\frac{1}{\delta}\right)\right)$$

(b) Assume a variable precision representation of numbers where $\Phi[i]$ is given with $|\Phi[i]|$ bits. Express the bound (1) as a function of Φ assuming that $P(\Phi)$ is defined so that each parameter is selected independently and that

$$P(\Phi[i]) = 2^{-|\Phi[i]|}$$

.

Solution:

$$\begin{aligned}
\mathcal{L}(h) &\leq \frac{10}{9}\left(\hat{\mathcal{L}}(h) + \frac{5L_{\max}}{N_{\mathrm{train}}}\left(\ln 2|\Phi| + \ln\frac{1}{\delta}\right)\right) \\
|\Phi| &= \sum_i |\Phi[i]|
\end{aligned}$$

(c) Repeat part (a) but for a model with d parameters represented by $\Phi_i = z[J[i]]$ where $J[i]$ is an integer index with $0 \leq J[i] < k$ and where $z[j]$ is a b bit floating point number and where all parameter vectors are equally likely.

Solution:

$$\mathcal{L}(h) \leq \frac{10}{9}\left(\hat{\mathcal{L}}(h) + \frac{5L_{\max}}{N}\left(kb\ln 2 + d\ln k + \ln\frac{1}{\delta}\right)\right)$$

Since d is large this is typically much tighter bound than using floating point or even integer representationfs of parameters. It is a much more compact representaiton of the parameters.

Problem 6. Implicit Bias for SGD on Least Squares Regression. Consider a hypothesis space $\mathcal{H}$ and a learning algorithm $\mathcal{A}$ that maps trainging data to a hyothesis in $\mathcal{H}$. Write $\mathcal{A}(\mathrm{Train})$ for the result of running algorithm $\mathcal{A}$ on training data Train. Also consider a given population distribution Pop where Train consists of N_{train} samples drawn independently from Pop. Let $P_{\mathcal{A},\mathrm{Pop}}(h)$ be the probability that $\mathcal{A}(\mathrm{Train}) = h$ when Train is drawn at random from Pop. The propbability distribution $P_{\mathcal{A},\mathrm{Pop}}$ is independent of any particular training sample and can be used as a PAC-Bayes prior on $\mathcal{H}$. A PAC-Bayes prior represents a learning bias. The distribution $P_{\mathcal{A},\mathrm{Pop}}$ is the **implicit bias** of algorithm $\mathcal{A}$ run on population Pop.

In this problem we consider the implicit bias of the SGD algorithm applied to least squares regression in the case where there are many more parameters than data points. Least squares regression is defined by

$$\Phi[J]^* = \operatorname*{argmin}_{\Phi}\; E_{\langle x,\,y\rangle\sim\mathrm{Train}}\left(\Phi[J]x[J] - y\right)^2$$

To solve this optimization problem we consider using SGD where Φ is initialized to the zero vector and we then apply the update

$$\begin{aligned}\Phi_{t+1} &= \Phi_t - \eta\nabla_\Phi\left(\Phi^\top x_y - y\right)^2 \\ &= \Phi_t - 2\eta(\Phi^\top x_t - y)x_t\end{aligned}$$

(a) In the case where $N_{\mathrm{train}} < d$, where d is the dimension of Φ and x, define a linear proper subspace of R^d such that we are guaranteed that Φ_t is in that space for al t.

Solution: Since every update is in the direction of some input vector x_t in the training data, SGD maintains the invariant that Φ_t is some linear combination of the training vectors $x_1, \ldots x_{N_{\mathrm{train}}}$. Since $N_{\mathrm{train}} < d$ the span of the training vectors must be a proper subspace of R^d.

(b) Assume that the training vectors $x_1, \ldots, x_{N_{\text{train}}}$ are linearly independent. In this case it can be shown that there exists a unique solution Φ^* in the space spanned by these vectors for which the square loss of the training data is zero (if these were not independent then we would have more training points than degrees of freedom in the space spanned by the input vectors). Let $b_1, \ldots, b_{N_{\text{train}}}$ be an orthonormal basis for the space spanned by the input vectors. For any $\Phi \in R^d$ define the projection of Φ into the subspace by

$$\begin{aligned}\Phi_\pi &= \sum_i (\Phi^\top b_i) b_i \\ \Phi_\perp &= \Phi - \Phi_\pi\end{aligned}$$

The convergence theorem for SGD now gives that SGD on least squares regression will converge in the limit to Φ^*. Show that SGD applied to least squared regression has a form of implicit bias similar to L_2 regression in that the result Φ^* is the least norm point in R^d for which the square loss of the training data is zero.

Solution: Consider any $\Phi \in R^d$ for which the training loss is zero. The projection Φ_π must also have zero training loss because each training vector can be written as a linear combination of basis vectors and Φ and Φ_π have the same inner product with each basis vector. Therefore $\Phi_\pi = \Phi^*$. Futhermore $\Phi = \Phi_\pi + \Phi_\perp$ and

$$||\Phi||^2 = ||\Phi_\pi||^2 + ||\Phi_\perp||^2 = ||\Phi^*||^2 + ||\Phi_\perp||^2$$

which gives that $||\Phi|| \geq ||\Phi^*||$ as desired.