Problem Set 6

Claire Goeckner-Wald

November 7, 2016

Overfitting and Deterministic Noise

1. **[b]** In general, deterministic noise will increase

Assume that $\mathcal{H}' \subseteq \mathcal{H}$ is a subset such that there are fewer hypotheses in $\mathcal{H}'$ than in $\mathcal{H}$. Consider natural subsets, such as $\mathcal{H}_2 \subset \mathcal{H}_1 0$. Bias, or deterministic noise, is the average squared error over the $\mathcal{X}$-space of our average hypothesis $\overline{g}(x)$ from target function f. Because we are constricting our hypothesis set to fewer total hypotheses, we in general increase the deterministic noise, because the average hypothesis $\overline{g}(x)$ may move away from the target function.

Regularization with Weight Decay

2. **[a]** 0.03 and 0.08

See the attached code.

3. **[d]** 0.03 and 0.08

See the attached code.

4. **[e]** 0.04 and 0.04

See the attached code.

5. **[e]** -1

See the attached code.

6. **[b]** 0.06

Between k = -1000 and k = 100, the least value of the out-sample error occurs at k = -1, and is 0.056. See the attached code.

Regularization for Polynomials

7. **[c]** $\mathcal{H}(10,0,3) \cap \mathcal{H}(10,0,4) = \mathcal{H}_2$

$\mathcal{H}_n$ is the set of all polynomials of order n or less.

Then, some hypothesis set $\mathcal{H}(Q, C = 0, Q_0)$ would create a set of polynomials of order $\min(Q, Q_0 - 1)$ or less. Because $w_q = C = 0$ for $q \geq Q_0$ (versus $q > Q_0$), we use $Q_0 - 1$ as a potential upper bound of the orders of polynomials in the set. However, Q itself is a potential upper bound the set is created using $\mathcal{H}_Q$ and filtering out undesirables.

[a] $\mathcal{H}(10, 0, 3) \cup \mathcal{H}(10, 0, 4) = \mathcal{H}_4$

This set is the union of: 1) a set of all polynomials of order 2 or less, 2) a set of all polynomials of order 3 or less. Thus, because this is a union, this is equivalent to the set $\mathcal{H}_3$, not the set $\mathcal{H}_4$.

[b] $\mathcal{H}(10, 1, 3) \cup \mathcal{H}(10, 1, 4) = \mathcal{H}_3$

Since this is a union, we can simply consider $\mathcal{H}(10, 1, 4)$. Here, $w_q = 1$ for $4 \leq q \leq 10$. Thus, this set contains some polynomials of order greater than 3. Since $\mathcal{H}_3$ is limited to polynomials of order 3 and less, this is not $\mathcal{H}_3$.

[c] $\mathcal{H}(10, 0, 3) \cap \mathcal{H}(10, 0, 4) = \mathcal{H}_2$

This set is the intersection of: 1) a set of all polynomials of order 2 or less, 2) a set of all polynomials of order 3 or less. Thus, because this is an intersection, this is equivalent to the set $\mathcal{H}_2$.

[d] $\mathcal{H}(10, 1, 3) \cap \mathcal{H}(10, 1, 4) = \mathcal{H}_1$

The intersection of both sets contains $\mathcal{H}_2$ because w_q is unlimited for $q < \min(3, 4)$. Thus, this set contains some polynomials of order greater than 1. Since $\mathcal{H}_1$ is limited to polynomials of order 1 and less, this is not equivalent to $\mathcal{H}_1$.

Neural Networks

8. **[d]** 45

Consider how much each operation adds to the total. The number of weights in this situation is $6 \cdot (4 - 1) + 4 \cdot (1 - 0) = 22$. We will use $o(\text{operation})$ as a function that returns the number of operations occurred.

Forward computation: Compute all $x_j^{(l)}$, starting with $l = 1$. Thus, the number of operations necessary here is equivalent to the number of weights.

$$o(w_{ij}^{(l)} x_i^{(l-1)}) = 22$$

Backward computation: For each level, we must calculate the δ for each node. Consider that for $l = 0$ and $l = L$, the number of operations here are 0. This is because for $l = 0$, no node takes in a signal. For $l = L$, we do not need to do any operations as we already know the δ. For each $\delta_i^{(l-1)}$, we only need $d^{(l)}$ operations. Thus, for $l = 1$ to $l = L - 1$, we take $\sum_{i=0}^{d^{(l)}} d^{(l+1)}$. Here, we only need to calculate the sum for the middle layer, $l = 1$, for which $d^{(1)} = 3$.

$$\begin{aligned} o(w_{ij}^{(l)}\delta_j^{(l)}) &= \sum_{i=1}^{d^{(1)}} d^{(1+1)} \\ &= \sum_{i=1}^{3} 1 \\ &= 1 + 1 + 1 = 3 \end{aligned}$$

Updating the weights: Each weight is updated once with the operation $w_{ij}^{(l)} = w_{ij}^{(l)} + \eta(x_i^{(l-1)}\delta_j^{(l)})$. Therefore, the number of operations for this calculation is equivalent to the number of weights.

$$o(x_i^{(l-1)}\delta_j^{(l)}) = 22$$

Totaling these up, we get $22 + 22 + 3 = 47$ operations. 47 is closer to 45 than to 50, thus the answer choice.

9. [**a**] 46

For all levels except for the last, two levels l and $l + 1$ with $x^{(l)}$ and $x^{(l+1)}$ units respectively, require $x^{(l)} \cdot (x^{(l+1)} - 1)$ weights to be fully connected. The last level connection requires $(x^{(l)} \cdot x^{(l+1)})$ weights because there is no bias term in the output. We can minimize the number of weights necessary by creating the hidden layers such that each layer l has a minimal number of units, $x^{(l)}$. Thus, we take $x^{(l)} = 2$ for each hidden layer, because we require a x_0, and at least one more term to be fully connected. Thus, the minimum number of weights o_{min} is as follows.

From the input to the first hidden layer:

$$o_{\text{min}} = 10(2 - 1) = 10$$

Between hidden layers, where terms $(x_0^{(l)}, x_1^{(l)})$ fully connect to $(x_0^{(l+1)}, x_1^{(l+1)})$, recalling that the term $x_0^{(l+1)}$ should not be connected to the previous layer:

$$o_{\text{min}} = 2(2 - 1) = 2$$

There are 36 hidden units. We desire that each layer is composed of (x_0, x_1). Thus, there must be $36/2 = 18$ hidden layers. Then, there are $18 - 1 = 17$ connection spaces between these 18 layers. Each connection space requires $o_{\text{min}} = 2$ weights for full connection to the previous layer. (Note that the first and last hidden layers are not necessarily the same.)

$$2 \cdot (17 \text{ layers}) = 34$$

From the last hidden layer to the output (since there is only one term in the output):

$$o_{\min} = 2(1) = 2$$

Then, our minimum number of weights is $34 + 10 + 2 = 46$. Using this same logic, if we instead had 9 hidden layers of 4 nodes each, the number of weights would be $10*(4-1)+8*(4*(4-1))+4*(1) = 130$. Thus, 46 is the likely minimum.

Note: the formula for number of weights o, given λ hidden layers of equal numbers of units each (assuming that $\frac{36}{\lambda} \in \mathbb{Z}$) is $o(\lambda) = 10(\frac{36}{\lambda} - 1) + (\lambda - 1)(\frac{36}{\lambda} \cdot (\frac{36}{\lambda} - 1)) + \frac{36}{\lambda} \cdot 1$. We desire to find the minimum and maximum of this function where $\lambda \in \mathbb{Z}$.

10. **[c]** 494

Follow from the intuition developed in the first problem. Consider a single hidden layer. Then, it must be composed of 36 units, or 36 nodes. Thus, we can calculate the maximum number of weights, $o_{\max}$.

From the input to the hidden layer of 36 nodes:

$$o_{\max} = 10(36-1) = 350$$

From the hidden layer of 36 nodes to the output:

$$o_{\max} = 36(1) = 36$$

Then, our maximum number of weights is $350 + 36 = 386$.

However, consider two hidden layers, each of 18 units, or 18 nodes.

From the input to the hidden layer of 36 nodes:

$$o_{\max} = 10(18-1) = 170$$

Between the two hidden layers:

$$o_{\max} = 18(18-1) = 306$$

From the last hidden layer of 18 nodes to the output:

$$o_{\max} = 18(1) = 18$$

Then, our maximum number of weights is $170 + 306 + 18 = 494$.

Consider three hidden layers, each of 12 units, or nodes.

From the input to the hidden layer of 36 nodes:

$$o_{\max} = 10(12-1) = 110$$

Between the three hidden layers:

$$o_{\max} = 12 * (12 - 1) \cdot (2 \text{ levels}) = 264$$

From the last hidden layer of 18 nodes to the output:

$$o_{\max} = 12(1) = 12$$

Then, our maximum number of weights is $170 + 306 + 18 = 386$. Thus, our likely maximum number of weights occurs at 494.