EECS 182 Deep Neural Networks
Spring 2023 Anant Sahai

Review: Basics

1. Dropout on Linear Regression

Recall that linear regression optimizes:

$$\mathcal{L}(\mathbf{w}) = ||\mathbf{y} - X\mathbf{w}||_2^2 \tag{1}$$

One way of using *dropout* on the d-dimensional input features $\mathbf{x}_i$ involves keeping each feature at random with probability p (and zeroing it out if not kept). This makes our learning objective effectively become

$$\mathcal{L}(\check{\mathbf{w}}) = \mathbb{E}_{R\sim Bernoulli(p)}\left[||\mathbf{y} - (R \odot X)\check{\mathbf{w}}||_2^2\right] \tag{2}$$

where $\odot$ is the element-wise product, and the random binary matrix $R \in \{0,1\}^{n\times d}$ is such that $R_{i,j} \sim_{i.i.d} Bernoulli(p)$. We use $\check{\mathbf{w}}$ to remind you that this is learned by dropout.

Show that we can manipulate (2) to eliminate the expectations and get:

$$\mathcal{L}(\check{\mathbf{w}}) = ||\mathbf{y} - pX\check{\mathbf{w}}||_2^2 + p(1-p)||\check{\Gamma}\check{\mathbf{w}}||_2^2 \tag{3}$$

with $\check{\Gamma}$ being a diagonal matrix whose j-th diagonal entry is the norm of the j-th column of the training matrix X.

Solution: Let $P = R \odot X$ where $\odot$ is the element-wise multiplication. Therefore, we have:

$$||y - Pw||_2^2 = y^Ty - 2w^TP^Ty + w^TP^TPw \tag{4}$$

That is:

$$\mathbb{E}_{R\sim Bernoulli(p)}[||y - R \odot Xw||_2^2] = \mathbb{E}_R[y^Ty - 2w^TP^Ty + w^TP^TPw] \tag{5}$$

Since the expected value of a matrix is the matrix of the expected value of its elements, we have that

$$\mathbb{E}_R[P]_{ij} = \mathbb{E}_R[(R \odot X)_{ij}] = X_{ij}\mathbb{E}_R[R_{ij}] = pX_{ij} \tag{6}$$

It follows that:

$$\mathbb{E}_R[2w^TP^Ty] = 2pw^TX^Ty \tag{7}$$

and:

$$(\mathbb{E}_R[(P^TP)])_{ij} = \Sigma_{k=1}^N \mathbb{E}_R[R_{ki}R_{kj}X_{ki}X_{kj}] \tag{8}$$

where:

$$\mathbb{E}_R[(P^TP)]_{ij} = \begin{cases} \sum_{k=1}^N \mathbb{E}_R[R_{ki}R_{kj}X_{ki}X_{kj}] = \sum_{k=1}^N \mathbb{E}_R[R_{ki}]\mathbb{E}_R[R_{kj}]X_{ki}X_{kj} = p^2(X^TX)_{ij} & \text{if } i \neq j \\ \sum_{k=1}^N \mathbb{E}_R[R_{ki}^2 X_{ki}X_{kj}] = \sum_{k=1}^N \mathbb{E}_R[R_{ki}^2]X_{ki}X_{kj} = p(X^TX)_{ij} & \text{if } i = j \end{cases} \tag{9}$$

Finally, we note that :

$$(\mathbb{E}_R[(P^TP)])_{ij} - p^2(X^TX)_{ij} = \begin{cases} 0 & \text{if } i \neq j \\ (p-p^2)(X^TX)_{ij} & \text{if } i = j \end{cases} \tag{10}$$

we now can put everything together as follow:

$$\begin{aligned}
\mathcal{L}(w) &= \mathbb{E}_R[||y - R \odot Xw||_2^2] && (11)\\
&= \mathbb{E}_R[y^Ty - 2w^TP^Ty + w^TP^TPw] && (12)\\
&= y^Ty - 2pw^TX^Ty + p^2w^TX^TXw - p^2w^TX^TXw + w^T\mathbb{E}_R[P^TP]w && (13)\\
&= ||y - pXw||_2^2 + (w^T\mathbb{E}_R[P^TP]w - p^2w^TX^TXw) && (14)\\
&= ||y - pXw||_2^2 + w^T(\mathbb{E}_R[P^TP]w - p^2)w && (15)\\
&= ||y - pXw||_2^2 + (p^2 - p)w^T(\text{diag}(X^TX))w && (16)\\
&= ||y - pXw||_2^2 + p(1-p)w^T(\text{diag}(X^TX))w && (17)\\
&= ||y - pXw||_2^2 + p(1-p)||\check{\Gamma}w||_2^2 && (18)\\
& && (19)
\end{aligned}$$

where $\text{diag}(X^TX)$ refers to the matrix where the non-diagonal elements of X^TX are set to 0, and $\check{\Gamma} = (\text{diag}(X^TX))^{1/2}$, which exists as X^TX is PSD and therefore has non-negative diagonal elements.

2. Feature Dimensions in CNN

We are going to describe a convolutional neural net using the following pieces:

- CONV3-F denotes a convolutional layer with F different filters, each of size $3 \times 3 \times C$, where C is the depth (i.e. number of channels) of the activations from the previous layer. Padding is 1, and stride is 1.
- POOL2 denotes a 2×2 max-pooling layer with stride 2 (pad 0)
- FLATTEN just turns whatever shape input tensor into a one-dimensional array with the same values in it.
- FC-K denotes a fully-connected layer with K output neurons.

Note: All CONV3-F and FC-K layers have biases as well as weights. **Do not forget the biases when counting parameters.**

We are going to use this network to do inference on a single input. Fill in the missing entries in this table of the size of the activations at each layer, and the number of parameters at each layer. You can/should write your answer as a computation (e.g. $128 \times 128 \times 3$) in the style of the already filled-in entries of the table.

Layer	Number of Parameters	Dimension of Activations
Input	0	$28 \times 28 \times 1$
CONV3-10	**Solution:** $3 \times 3 \times 1 \times 10 + 10$	$28 \times 28 \times 10$
POOL2	0	$14 \times 14 \times 10$
CONV3-10	$3 \times 3 \times 10 \times 10 + 10$	**Solution:** $14 \times 14 \times 10$
POOL2	**Solution:** 0	**Solution:** $7 \times 7 \times 10$
FLATTEN	0	490
FC-3	**Solution:** $490 \times 3 + 3$	3

