1. Dropout on Linear Regression

Recall that linear regression optimizes:

$$\mathcal{L}(w) = \|y - Xw\|_2^2$$  \hspace{1cm} (1)

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

$$\mathcal{L}(\tilde{w}) = \mathbb{E}_{R \sim \text{Bernoulli}(p)} \left[ \|y - (R \odot X)\tilde{w}\|_2^2 \right]$$  \hspace{1cm} (2)

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

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

$$\mathcal{L}(\tilde{w}) = \|y - pX\tilde{w}\|_2^2 + p(1-p)\|\tilde{\Gamma}\tilde{w}\|_2^2$$  \hspace{1cm} (3)

with \(\tilde{\Gamma}\) 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:

$$\|y - Pw\|_2^2 = y^T y - 2w^T P^T y + w^T P^T Pw$$  \hspace{1cm} (4)

That is:

$$\mathbb{E}_{R \sim \text{Bernoulli}(p)}[\|y - R \odot Xw\|_2^2] = \mathbb{E}_R[y^T y - 2w^T P^T y + w^T P^T Pw]$$  \hspace{1cm} (5)

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

$$\mathbb{E}_R[P]_{ij} = \mathbb{E}_R[(R \odot X)_{ij}] = X_{ij} \mathbb{E}_R[R_{ij}] = pX_{ij}$$  \hspace{1cm} (6)

It follows that:

$$\mathbb{E}_R[2w^T P^T y] = 2pw^T X^T y$$  \hspace{1cm} (7)

and:

$$\mathbb{E}_R[(P^T P)]_{ij} = \sum_{k=1}^{N} \mathbb{E}_R[R_{ki}R_{kj}X_{ki}X_{kj}]$$  \hspace{1cm} (8)
where:
\[
\mathbb{E}_R[(P^T P)]_{ij} = \begin{cases} 
\sum_{k=1}^{N} \mathbb{E}_R[R_{ki}R_{kj}X_{ki}X_{kj}] = \sum_{k=1}^{N} \mathbb{E}_R[R_{ki}^2]X_{ki}X_{kj} & \text{if } i \neq j \\
\sum_{k=1}^{N} \mathbb{E}_R[R_{ki}X_{ki}X_{kj}] = \sum_{k=1}^{N} \mathbb{E}_R[R_{ki}^2]X_{ki}X_{kj} & \text{if } i = j 
\end{cases}
\]

Finally, we note that:
\[
(\mathbb{E}_R[(P^T P)])_{ij} - p^2 (X^T X)_{ij} = \begin{cases} 
0 & \text{if } i \neq j \\
(p - p^2) (X^T X)_{ij} & \text{if } i = j 
\end{cases}
\]

we now can put everything together as follow:

\[
\mathcal{L}(w) = \mathbb{E}_R[||y - R \odot Xw||_2^2] 
= \mathbb{E}_R[y^T y - 2w^T P^T y + w^T P^T P w] 
= y^T y - 2pw^T X^T y + p^2 w^T X^T X w - p^2 w^T X^T X w + w^T \mathbb{E}_R[P^T P] w 
= ||y - pXw||_2^2 + (w^T \mathbb{E}_R[P^T P] w - p^2 w^T X^T X w) 
= ||y - pXw||_2^2 + (p - p^2) w^T (\text{diag}(X^T X)) w 
= ||y - pXw||_2^2 + p(1 - p) ||\hat{\Gamma} w||_2^2 
\]

where \(\text{diag}(X^T X)\) refers to the matrix where the non-diagonal elements of \(X^T X\) are set to 0, and \(\hat{\Gamma} = (\text{diag}(X^T X))^{1/2}\), which exists as \(X^T X\) 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 $2 \times 2$ 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.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Number of Parameters</th>
<th>Dimension of Activations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>0</td>
<td>$28 \times 28 \times 1$</td>
</tr>
<tr>
<td>CONV3-10</td>
<td>Solution: $3 \times 3 \times 1 \times 10 + 10$</td>
<td>$28 \times 28 \times 10$</td>
</tr>
<tr>
<td>POOL2</td>
<td>0</td>
<td>$14 \times 14 \times 10$</td>
</tr>
<tr>
<td>CONV3-10</td>
<td>Solution: $3 \times 3 \times 10 \times 10 + 10$</td>
<td>Solution: $14 \times 14 \times 10$</td>
</tr>
<tr>
<td>POOL2</td>
<td>Solution: 0</td>
<td>Solution: $7 \times 7 \times 10$</td>
</tr>
<tr>
<td>FLATTEN</td>
<td>0</td>
<td>490</td>
</tr>
<tr>
<td>FC-3</td>
<td>Solution: $490 \times 3 + 3$</td>
<td>3</td>
</tr>
</tbody>
</table>