Homework 3

Deadline: Thursday, March 7, at 11:59pm.

Submission: You must submit your solutions as a PDF through MarkUs. You can produce the file however you like (e.g. LaTeX, Microsoft Word, scanner) as long as it is readable.

Late Submission: MarkUs will remain open until 3 days after the deadline, after which no late submissions will be accepted. The late penalty is 10% per day, rounded up.

Weekly homeworks are individual work. See the Course Information handout[1] for detailed policies.

1. **Dropout.** **[5pts]** For this question, you may wish to review the properties of expectation and variance: `https://metacademy.org/graphs/concepts/expectation_and_variance`

   Dropout has an interesting interpretation in the case of linear regression. Recall that the predictions are made stochastically as:

   $$y = \sum_j m_j w_j x_j,$$

   where the m_j's are all i.i.d. (independnet and identically distributed) Bernoulli random variables with expectation 1/2. (I.e., they are indepdendent for every input dimension and every data point.) We would like to minimize the cost

   $$\mathcal{J} = \frac{1}{2N} \sum_{i=1}^{N} \mathbb{E}[(y^{(i)} - t^{(i)})^2], \tag{1}$$

   where the expectation is with respect to the $m_j^{(i)}$'s.

   Now we show that this is equivalent to a regularized linear regression problem:

   (a) **[2pts]** Find expressions for $\mathbb{E}[y]$ and $\mathrm{Var}[y]$ for a given $\mathbf{x}$ and $\mathbf{w}$.

   (b) **[1pt]** Determine $\tilde{w}_j$ as a function of w_j such that

   $$\mathbb{E}[y] = \tilde{y} = \sum_j \tilde{w}_j x_j.$$

   Here, $\tilde{y}$ can be thought of as (deterministic) predictions made by a different model.

   (c) **[2pts]** Using the model from the previous section, show that the cost $\mathcal{J}$ (Eqn. 1) can be written as

   $$\mathcal{J} = \frac{1}{2N} \sum_{i=1}^{N} (\tilde{y}^{(i)} - t^{(i)})^2 \; + \; \mathcal{R}(\tilde{w}_1, \ldots, \tilde{w}_D),$$

   where $\mathcal{R}$ is a function of the $\tilde{w}_D$'s which does not involve an expectation. I.e., give an expression for $\mathcal{R}$. (Note that $\mathcal{R}$ will depend on the data, so we call it a "data-dependent regularizer.")

   Hint: write the cost in terms of the mean and variance formulas from part (a). For inspiration, you may wish to refer to the derivation of the bias/variance decomposition from the Lecture 12 course notes.

[1] `http://www.cs.toronto.edu/~rgrosse/courses/csc421_2019/syllabus.pdf`

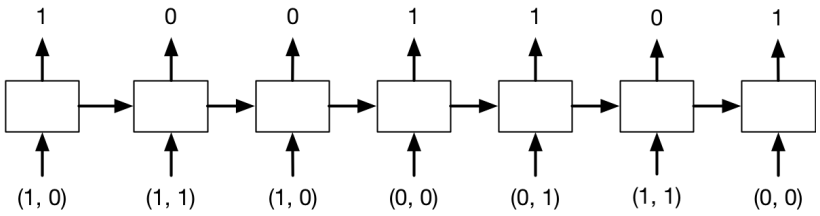
2. **Binary Addition [5pts]** In this problem, you will implement a recurrent neural network which implements binary addition. The inputs are given as binary sequences, starting with the *least* significant binary digit. (It is easier to start from the least significant bit, just like how you did addition in grade school.) The sequences will be padded with at least one zero on the end. For instance, the problem

$$100111 + 110010 = 1011001$$

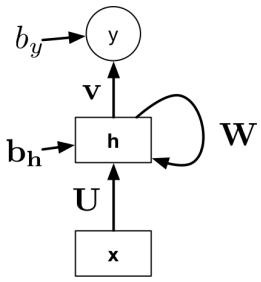
would be represented as:

- **Input 1:** 1, 1, 1, 0, 0, 1, 0
- **Input 2:** 0, 1, 0, 0, 1, 1, 0
- **Correct output:** 1, 0, 0, 1, 1, 0, 1

There are two input units corresponding to the two inputs, and one output unit. Therefore, the pattern of inputs and outputs for this example would be:



Design the weights and biases for an RNN which has two input units, three hidden units, and one output unit, which implements binary addition. All of the units use the hard threshold activation function. In particular, specify weight matrices $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{W}$, bias vector $\mathbf{b_h}$, and scalar bias b_y for the following architecture:



Hint: In the grade school algorithm, you add up the values in each column, including the carry. Have one of your hidden units activate if the sum is at least 1, the second one if it is at least 2, and the third one if it is 3.