# Machine Learning and Computational Physics
# Fall 2020

**Assignment 4**

Due: October $19^{th}$ 2020, 11:59:59 PM PDT

## Backpropagation of second derivatives

In class notes we recognized that in order to solve higher-order PDEs using deep neural networks we need to evaluate the second derivative of the output layer with respect to the input layer. This can be done by developing a backpropagation algorithm for the second derivatives which relates the second derivative wrt. the output of layer $(l)$ to the second derivative wrt. to the output of layer $(l+1)$. That is we need a formula like:

$$\frac{\partial^2(\cdot)}{\partial x_k^{(l)}\partial x_i^{(l)}} = \sum_j A_{ikj}^{(l+1)} \frac{\partial(\cdot)}{\partial x_j^{(l+1)}} + \sum_{j,m} B_{ikjm}^{(l+1)} \frac{\partial^2(\cdot)}{\partial x_j^{(l+1)}\partial x_m^{(l+1)}}. \tag{1}$$

Using the notes in the class find an expression for $A_{ikj}^{(l+1)}$ and $B_{ikjm}^{(l+1)}$ in terms of the weights of the $(l+1)$ layer and the derivatives of the activation function.