

# CS63 Spring 2018

## Lab 9: Convolutional Neural Networks

Cindy Li and Grace Zhang

### 1 Data Set

Our inputs are board states. Before creating our training and test sets, we made sure to transpose the boards corresponding to Player -1 turn to one that mimics the orientation of Player 1 boards. We did this so that we could predict the best moves of both players with the same neural net. We decided to predict the next move because we thought producing a two node output containing an x and y coordinate would be both straightforward and easy to implement.

### 2 Network

Our neural network is constructed of two layers. The first is a convolutional layer with 64 output channels, a window size of 3x3 making strides of 1x1 with a relu activation function. After flattening, we then add a dense layer, also with a relu activation function. We then compile using a mean-squared-error loss and the Adam optimizer. Our fit is based off of 12 epochs. We chose to start with 12 epochs and the Adam optimizer because we saw that it was successful in our last lab. The SGD optimizer did not do as well as Adam, nor did the categorical\_crossentropy loss do better than mean\_squared\_error. Increasing epochs past 12 also did not show a significant increase.

### 3 Evaluation

We originally had tried putting the dense layer first, but quickly realized that that was incorrect and would never work. Instead of one dense layer, we tried to do two with relu and it worsened our performance. Then we tried sigmoid and it performed similarly worse. Two convolutional layers, on the other hand, increased performance a little. Window sizes of 2x2 showed better performance than window sizes of 3x3. Surprisingly, a window size of 1x1 where there is zero overlap increased our performance slightly.

### 4 Discussion

Our network is good at extracting multiple features from the board states because we have multiple convolutional layers as opposed to our single dense layer. However, our network could be underfitting since we only have three layers in our network.