## Introduction

The most common cancer in the U.S. is female breast cancer. All cancers can be broken down into smaller categories called *cancer subtypes*. The most common subtype of breast cancer is *Invasive Ductal Carcinoma* (IDC), which is when the cancer spreads from its origination in the breast ducts and invades other breast tissue.

The purpose of this project will be to classify breast cancer images as whether they are in IDC or not, which is a common task of pathologists.

Motivations for choosing this topic include:

- image classification lends itself well to using convolutional neural networks
- I am currently creating a Cancer Data Visualizer for the GW Cancer Center, and I wanted to apply a Deep Learning method to related data

This report will cover details related to the data that was used, what kinds of models were used, and the results that were obtained from the models.

## Data

The data that will be used include 277,524 images of breast cancer cells, each of which has a 50x50 resolution. 198,738 of them are classified as non-IDC and the other 78,786 are classified as IDC. They can be found on Kaggle: https://www.kaggle.com/paultimothymooney/breast-histopathology-images#10253_idx5_x1001_y1001_class0.png.

## Deep Learning Network

The base network used to classify these images is a convolutional neural network (CNN). Since these images have color, the network will have to allow for a 3-color channel. Small kernels will be run across the 50x50 cancer cell images to produce feature maps that will be able to categorize the image as IDC or non-IDC.

## Methods

For pre-processing, the data was split into 70% training, 15% testing, and 15% validation sets. The training data was used purely for creating the model, the testing set was used to produce accuracy scores, and the validation set was put aside and was not to be used until after the project. Since the data overall was 30% malignant and 70% benign, this ratio was maintained across the 3 data sets to ensure they were each well-balanced.

The final CNN architecture was built accordingly:

- take in the 50x50 image as a 3-color channel input
- use 2 convolution layers, each of which used a 2x2 kernel to stride across the image
- use a final layer that would output to 2 classes, benign or malignant
- the 1st convolution layer was max-pooled, while the 2nd convolution layer was average-pooled
- layer outputs were normalized between layers
- all activation functions were log softmax
- loss was measured by binary cross entropy

In addition, the following hyper-parameters were used:

- learning rate = 0.05
- 15 epochs
- batch size of 512
- dropout = 0.5

Due to the size of the training set, it was important to use batches and update the model's loss after each batch. After each epoch, the model's performance was evaluated by looking at the accuracy generated on the test set. 15 was chosen as the number of epochs, as that was where the test accuracy would level off.

## Results

Model 1:

- use just 10,000 images from the training set
- use a kernel of size 3x3
- test accuracy = 71%

Model 2:

- switched the kernel size to size 2x2
- test accuracy = 81%

Model 3:

- doubled training size to 20,000 images
- test accuracy = 82%

Model 4 (final):

- use all training images
- test accuracy 85%

## Summary

The biggest improvement in the model's performance came from decreasing the kernel size from 3x3 to 2x2, which is most likely due to the images themselves being quite small at 50x50. Increasing the amount of training data seemed to have very marginal returns on the accuracy, and the highest accuracy achieved was 85%.

There are 2 main strategies that could be used in the future to improve this model. First, image augmentation could be used to help the model understand variations of the images it was being presented with. This might help with the fact that the overall class ratio was biased toward benign images. Second, much more could be done with experimenting with the network features, such as learning rate, number of layers, etc.