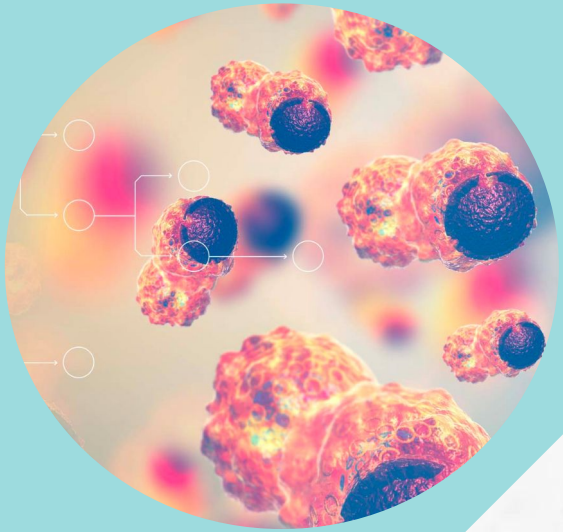


Project 2

Histopathologic Breast Cancer Detection

Abou Keita
William Simpson

DATA 690
Practical Deep Learning
Fall 2022

Background

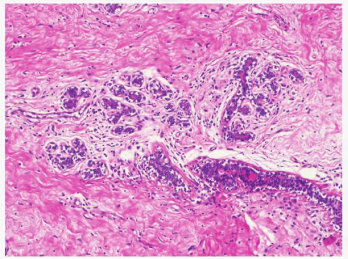
Breast Cancer

- A Disease where cells in the breast grow out of control
- Metastatic/Metastasized Breast cancer is when breast cancer spreads to other parts of the body

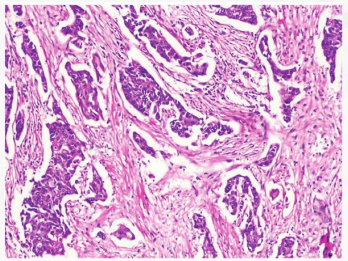
H&E (Hematoxylin and Eosin) Staining

- To identify the distribution and structure of cells by coloring different components
- Properties of Hematoxylin create a purplish-blue color stain within the cell nuclei
- Eosin interacts with cell cytoplasm to generate a pink color
- Used for primary diagnosis

Normal breast tissue



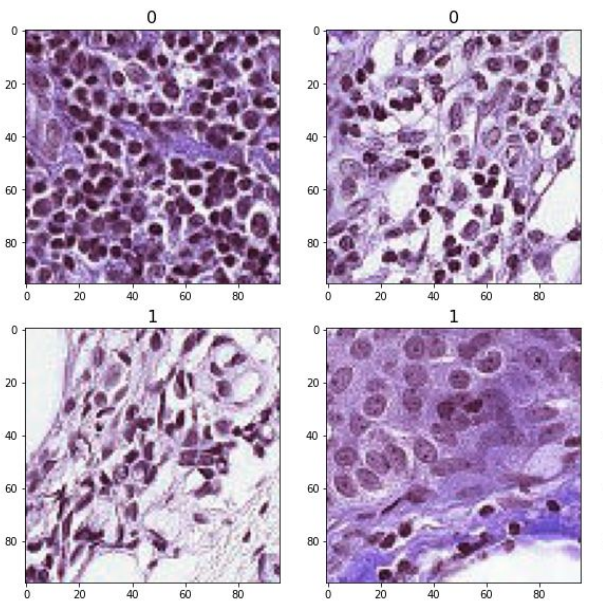
Breast Cancer



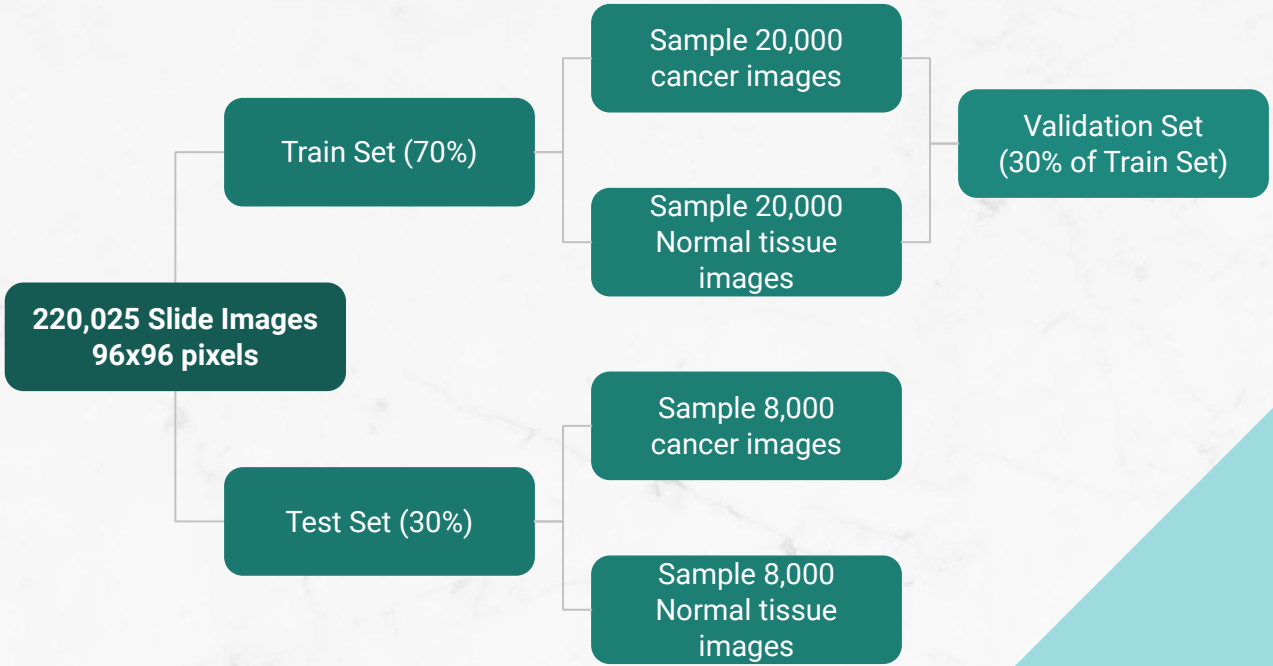
Sources: CDC, Visikol, and Naik et al., 2020

Data

Sample images and labels



Our Training Strategy



Conv2d: 3 layers
Kernel size: (2,2)
Stride: (1,1)

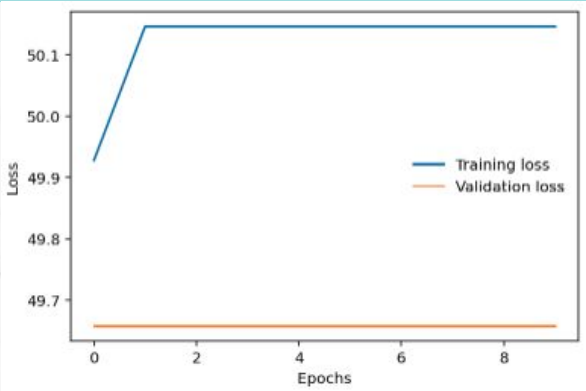
Batch Normalization
ReLU act. function
Max Pooling
Dropout in FC layers

Model

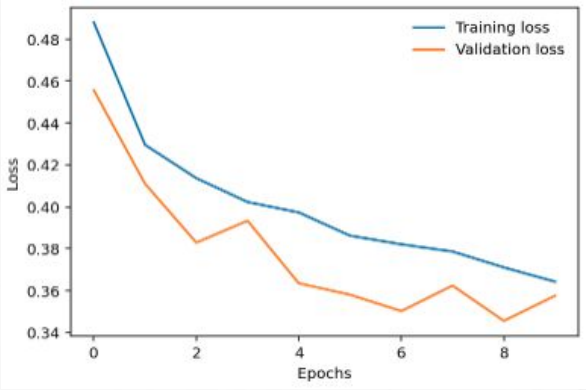
```
CNN(
  (conv1): Sequential(
    (0): Conv2d(3, 16, kernel_size=(2, 2), stride=(1, 1))
    (1): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace=True)
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  )
  (conv2): Sequential(
    (0): Conv2d(16, 32, kernel_size=(2, 2), stride=(1, 1), padding=(1, 1))
    (1): BatchNorm2d(32, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace=True)
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  )
  (conv3): Sequential(
    (0): Conv2d(32, 128, kernel_size=(2, 2), stride=(1, 1), padding=(1, 1))
    (1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace=True)
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  )
  (dropout2d): Dropout2d(p=0.5, inplace=False)
  (fc): Sequential(
    (0): Linear(in_features=18432, out_features=1024, bias=True)
    (1): ReLU(inplace=True)
    (2): Dropout(p=0.4, inplace=False)
    (3): Linear(in_features=1024, out_features=512, bias=True)
    (4): Dropout(p=0.4, inplace=False)
    (5): Linear(in_features=512, out_features=1, bias=True)
    (6): Sigmoid()
  )
)
```

Hyperparameter Tuning

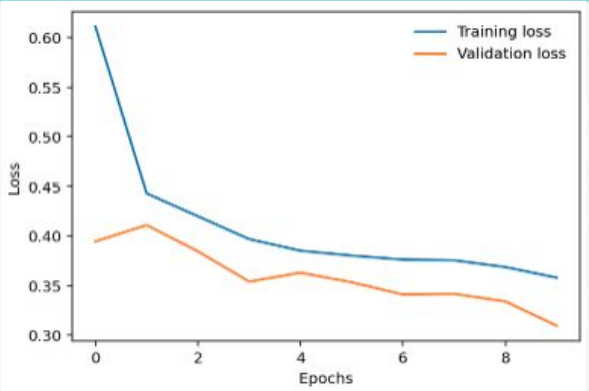
Model 1:
lr=0.001,
fc 1=1024



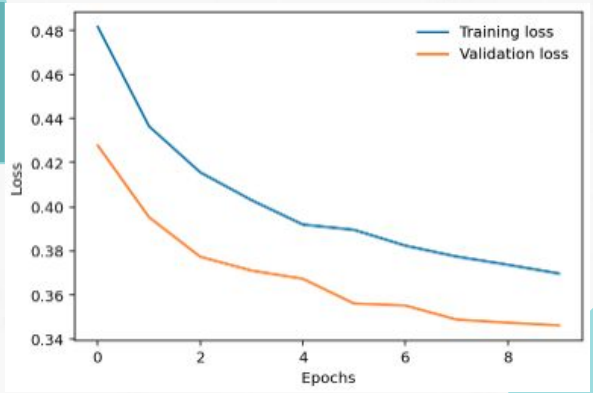
Model 2:
lr=0.0001,
fc 1=1024



Model 3:
lr=0.001,
fc 1=512

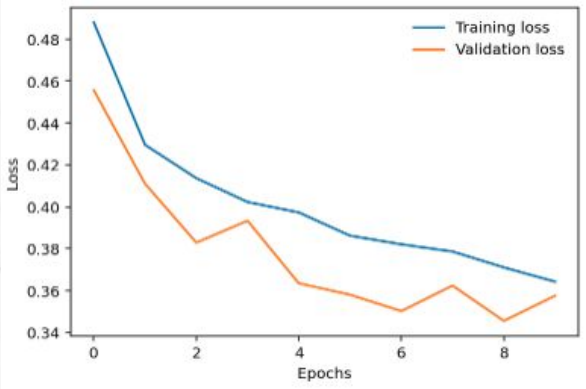


Model 4:
lr=0.0001,
fc 1=512

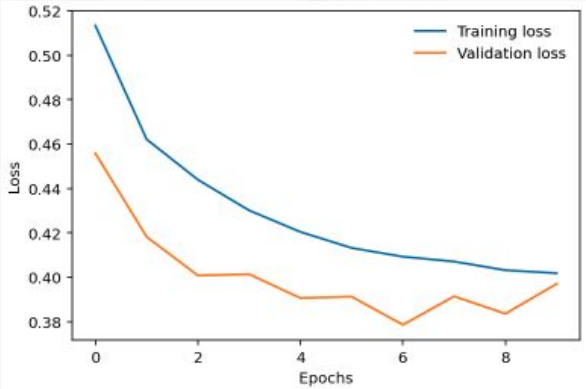


Hyperparameter Tuning

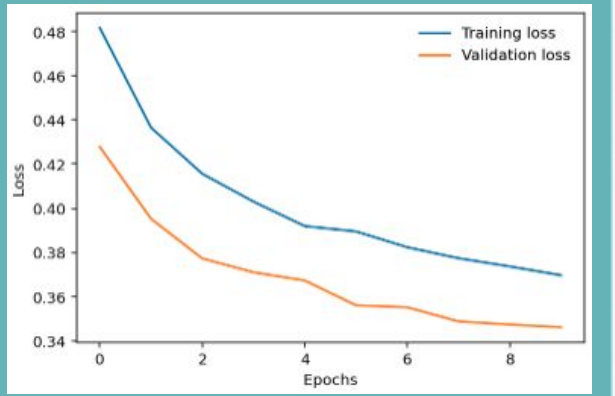
Model 2:
lr=0.0001,
fc 1=1024



Model 5:
lr=0.0001,
fc 1=1024
2 Conv layers



Model 4:
lr=0.0001,
fc 1=512



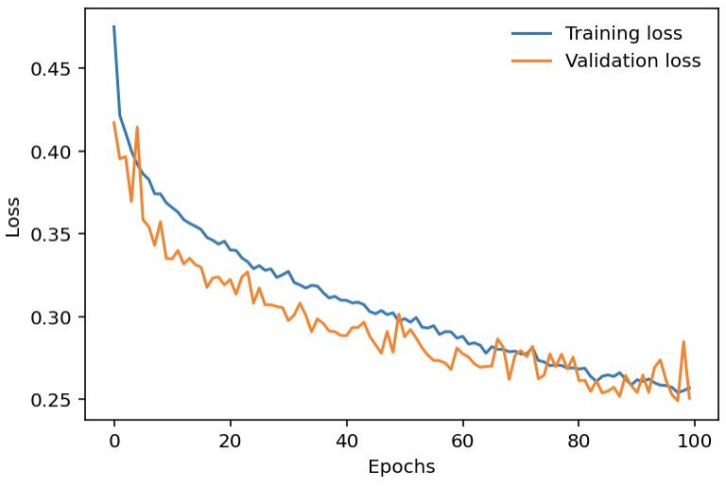
Model 4 = Best Model of the 5 tested here

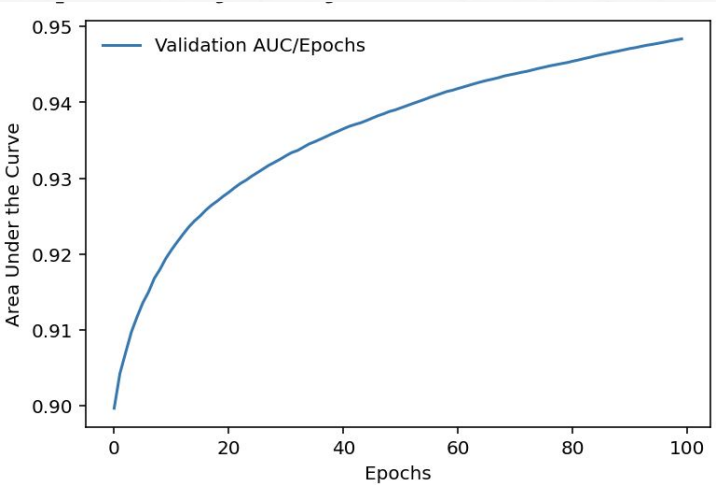
- Lowest training and validation loss (0.38 and 0.35)
- Learning is steady, which is shown by the smooth decrease of both loss curves

Results of Best Model

Model 4:
lr=0.0001,
fc 1=512,
3 Conv layers

100 epochs





Sample Predictions