Systems for Diagnosis and Treatment of Breast Cancer

Connor Amorin, Gabriel P. Andrade, Sandra Castro-Pearson, Abdel Kader Geraldo, Brandon Iles, Dean Katsaros,Terry Mullen, Sam Nguyen, Oliver Spiro, Melissa Sych

University of Massachusetts Amherst

February 26, 2018

Outline

Motivation

- 39.6 percent of men and women in the US will be diagnosed with cancer at some point during their lifetimes
- One in eight women in the US will be diagnosed with breast cancer
- Early detection is essential in treatment
- Computer Automated Detection and Diagnosis is currently used as "second reader" to the radiologist to make sure no detection is missed

Our Goals

- Detect abnormal regions in a mammogram
- Classify those regions as malignant or benign
- Understand and implement tumor development models that account for competing cell populations and chemotherapy

Detection System Goal

- The goal of our group was to research and further develop algorithms to identify the presence of a mass in a mammogram
- Can we identify the same masses that radiologist do? Can we do better?

The Data Used

- Public data base from the Cancer Imaging Archive called the Curated Breast Imaging Subset of DDSM (CBIS-DDSM)
- For each patient we had full mammogram and mass mask images as well information on the type of mass identified

Breast Cancer Diagnosis and Treatment

Applied Mathematics Masters Program

One Case

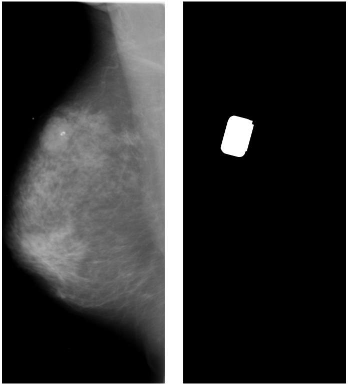


Figure: Patient 100, Benign Mass

Challenges and Solutions

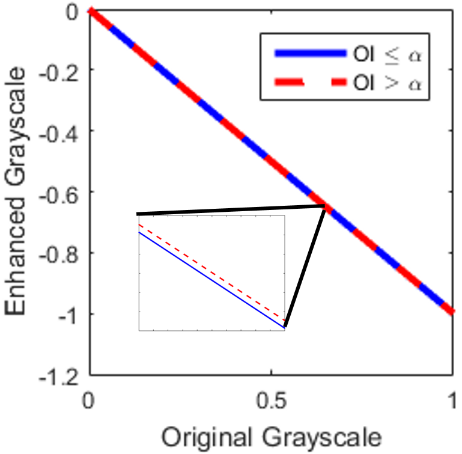
- Mass characteristics (shape, size, density) differ for each patient
- The tissue in the background has similar characteristics to masses
- Mammogram images are from 1990s and not digital so they have poor contrast/quality
- In order to improve image quality and identify masses we will use a three step process:
  1. Apply a linear transformation enhancement filter
  2. Segment mass regions
  3. Use adaptive thresholding for mass identification

Image Enhancement Filter

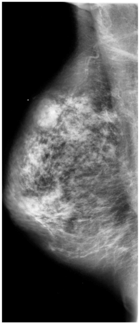
$$\mathrm{EI}_{ij} = \begin{cases} a \log(1 + b\mathrm{OI}_{ij} & \mathrm{OI}_{ij} < \alpha \\ \dfrac{\exp\left(\dfrac{\mathrm{OI}_{ij}}{a} - 1\right)}{b} & \mathrm{OI}_{ij} > \alpha \end{cases}$$

- OI = original image
- EI = enhanced image
- m is the maximum value of the gray level in the image
- a and α are parameters to be chosen empirically
- $b = \dfrac{1 - \exp\left(\dfrac{m}{a}\right)}{m}$

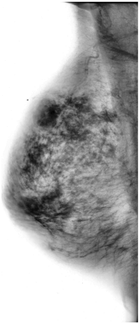
Why does it work?



Result



(a) Orginal mammogram



(b) Enhanced mammogram

Segmentation of Mass Regions

Segment the regions of interest by:

$$SI = OI + EI$$

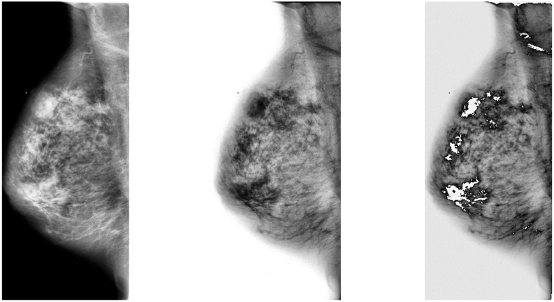


Figure: The full (left), enhanced (middle), and segmented (right) mammograms.

Near the Mass

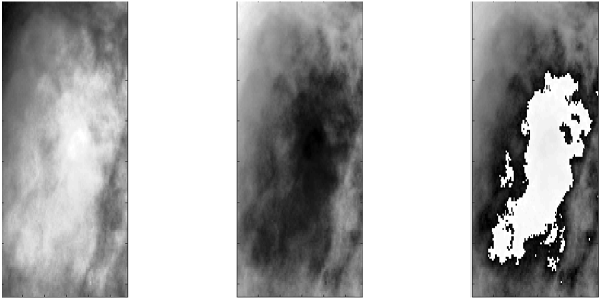
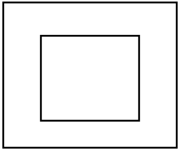


Figure: Original (left), enhanced (middle), and segmented (right) mammograms

Adaptive Local Thresholding



- Create an adaptive threshold

$$\mathrm{TH}_{ij} = M_{ij} + \gamma \mathrm{SI}_{diff\,ij},$$

  - $\mathrm{SI}_{diff\,ij} = \mathrm{SI}_{max\,ij} - \mathrm{SI}_{min\,ij}$ from large window
  - M_{ij}= mean intensity in small window
  - γ to be set empirically between 0 and 1
- If $\mathrm{SI}_{ij} \geq \mathrm{TH}_{ij}$ and $\mathrm{SI}_{ij} \geq M_{ij}$, then the pixel is suspicious

Why adaptive?

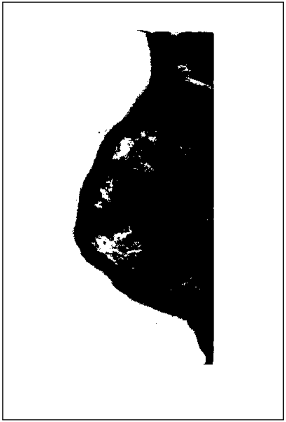


Figure: The resulting mask when we threshold every pixel with the same threshold

Results

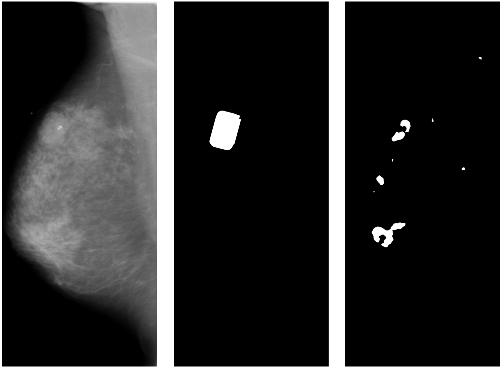


Figure: Original mammogram (left), the mask given by the data set (center), and the mask found by adaptive thresholding (right)

Classification

- Goal: Classify possible masses as benign or malignant
  1. Use hand crafted features to classify
  2. Use deep neural networks to learn features to classify
- Let $\mathbf{x}$ be a feature obtained from the mammogram and $y \in \{-1, 1\}$ be the label -1 if the mass is benign and 1 if the mass is malignant.
- A classifier is a function f that takes in $\mathbf{x}$ and outputs y

Histogram of Oriented Gradients

- Let I be the image
- At each pixel compute:
  1. Gradients: I_x and I_y
  2. Orientation: $\theta = \tan^{-1}(\frac{I_y}{I_x})$
  3. Magnitude: $\sqrt{I_x^2 + I_y^2}$
- Partition image into blocks
- For each block take weighted histogram of orientations weighted by the gradient magnitude

HOG Features

Figure: A visualization of the HOG features

Classifiers

Logistic Regression:

$$y^* = \arg\max_y P(y|\mathbf{w}, \boldsymbol{b}, \mathbf{x}) = (1 + e^{\boldsymbol{w}^T\boldsymbol{x}+b})^{-1}$$

Support Vector Machines:

$$y^* = \text{sign}(\mathbf{w}^T\boldsymbol{x} + b)$$

- y^*: label found by classifier
- $\boldsymbol{w}$: weight vector
- b bias:

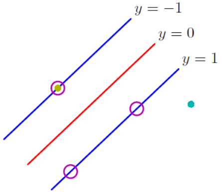


Figure: SVM maximized the margin and finds the optimal seperating hyperplane between classes [1].

[1] Bishop,Christopher, "Pattern Recognition and Machine Learning"

Some Classification Results

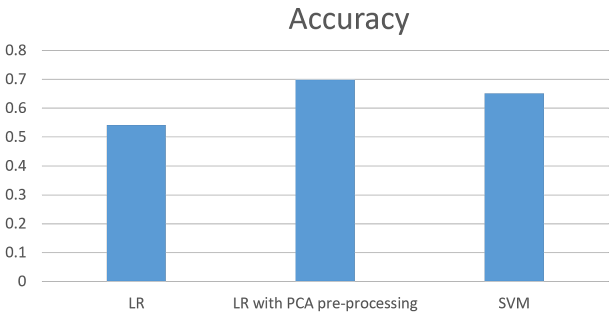


Figure: Classification accuracy for using the HOG features. The best performing HOG parameters were a block size of 32 and 11 angle bins (LR), block size of 32 and 14 angle bins(LR with PCA pre-processing), and block size of 32 and 15 angle bins (SVM).

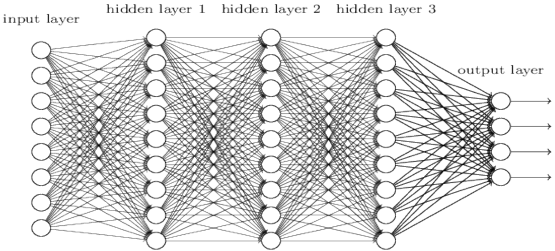
Neural Nets

The goal of our classification is to fit an approximation f to the true classifier function

$$f^* : \mathcal{M} \to \{-1, 1\}$$

Where $\mathcal{M} \subset [0,1]^{w \times h}$ is the space of mammogram images, and the labels $\{-1, 1\}$ represent the diagnosis

- Build this classifier from small parts
- The approximator $f(\boldsymbol{x})$ is the composition of multiple functions referred to as **layers**
- Each individual layer consists of simple functions known as **units**[2]



[2] Also called neurons.

Units

Each unit $g : \mathbb{R}^m \to \mathbb{R}$ is parametrized by a *weight vector* **w** and a scalar *bias* **b** Typically:

$$g(\boldsymbol{x}; \boldsymbol{w}, b) = h\left(\boldsymbol{w}^T \boldsymbol{x} + b\right)$$

- h is a fixed nonlinear function called an **activation**
  - Usually $h(z) = \max\{0, z\}$ or $h(z) = 1/(1 + e^{-z})$
- **w** and b are fit to the training data

Layers

A **Layer** $f^{(i)}$ is then a vector of a number of units stacked together

$$f^{(i)}(\boldsymbol{x}) = \left[g_1^{(i)}(\boldsymbol{x}), g_2^{(i)}(\boldsymbol{x}), ..., g_d^{(i)}(\boldsymbol{x})\right]^T$$

Finally, we combine these layers via nested composition. If $\boldsymbol{x}$ comes from the set of input images, we have

$$f^{(i)}(\boldsymbol{x}) = f^{(i)}(f^{(i-1)}(...(f^{(1)}(\boldsymbol{x}))))$$

This is our **Network**

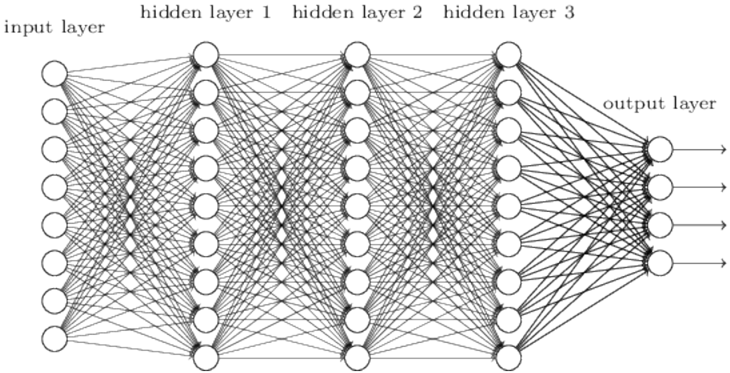
Network

Recall that our goal is to find an f that approximates the true classifier

$$f^* : \mathcal{M} \to \{-1, 1\}$$

- Normalize the output of the final layer to give a probability distribution
- f is the function that returns -1 or 1 based on which class has higher probability

Graphical Representation



Training/Fitting

Now we have the functional form of the function f used for prediction, but how do we find a good set of parameters?

- Supervised learning: Give the network pairs $(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{y} \in \{-1, 1\}$ is the diagnosis
- compute a **Loss** function to quantify the "badness" of fit
- Similar to likelihood maximization used in linear regression, etc.
- Minimize loss using gradient descent and **Backpropagation** to fit weights and biases to the data

Convolutional Neural Networks

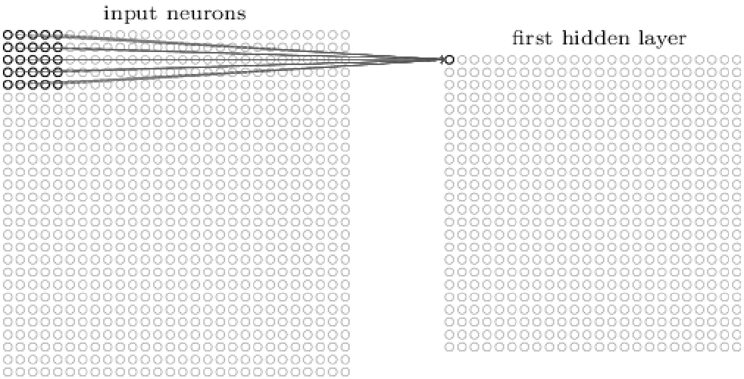
- As is common practice with image data, we actually used **Convolutional** neural networks
- The functional form differs subtly in these networks, using a (discrete) convolution instead of an inner product in the unit functions

  We now retain the 2D structure of each x, and then the convolution maps it to another 2D grid of units, with entries:

$$g_{mn}(\boldsymbol{x}; \boldsymbol{w}, b) = h\left((\boldsymbol{w} * \boldsymbol{x})_{mn} + b \right)$$

$$(\boldsymbol{w} * \boldsymbol{x})_{mn} = \sum_{k,l} w_{m+k,n+l} \cdot x_{kl}$$

Convolutional Neural Networks



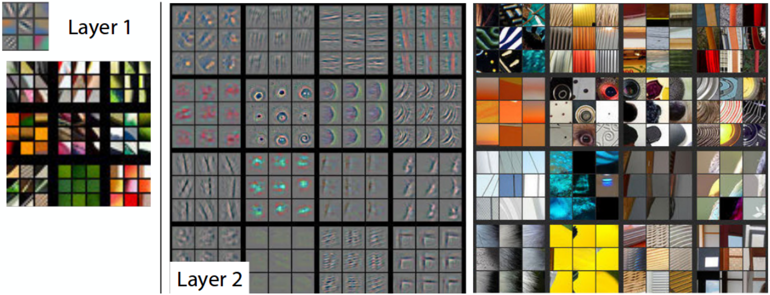
Transfer Learning

- Neural networks with numerous layers are referred to as "deep"
- One of the crippling drawbacks of such networks is the sheer volume of training data they need
- Results that make headlines with their near perfect accuracy can use upwards of a million training images

Transfer Learning

- We had $\approx$ 1200 mammography images
- In many applications gathering more data may not be feasible nor ethical (e.g medical data)
- We can take advantage of networks pretrained on tasks where data is abundant

Transfer Learning



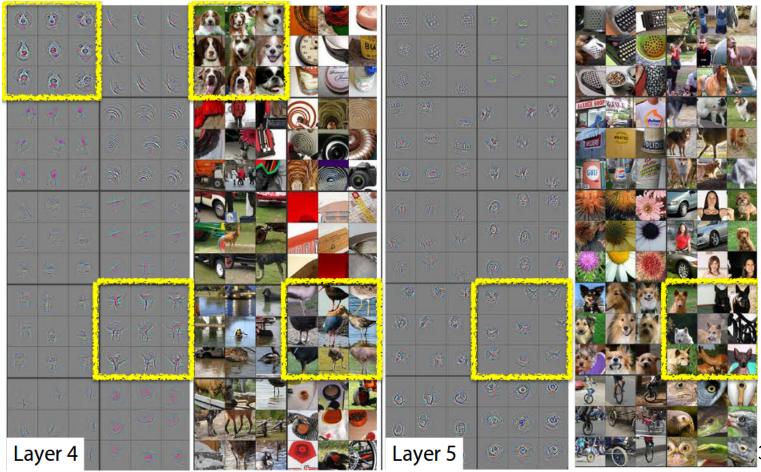
[3]

[3] From Zeiler and Fergus

Transfer Learning

Breast Cancer Diagnosis and Treatment

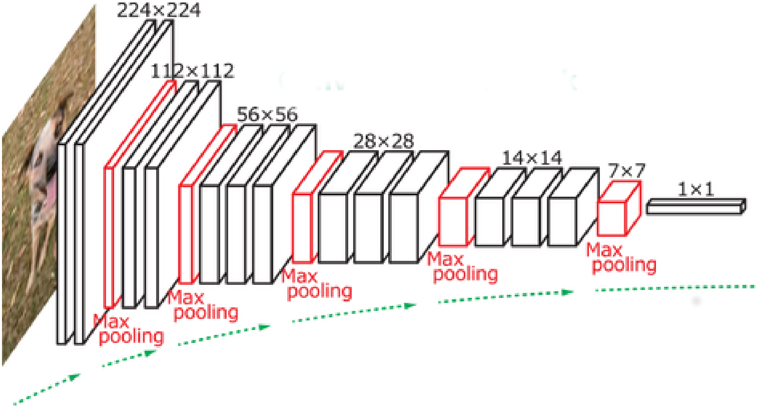
Applied Mathematics Masters Program



[3]

[3] From Zeiler and Fergus

Transfer Learning



Data Augmentation

- Artificially generates new data
- Addresses lack of data by increasing effective sample size
- Guards against overfitting to the training set
- Our augmentation included:
  - reflecting images horizontally
  - reflecting vertically
  - small-scale zooms

Architectures

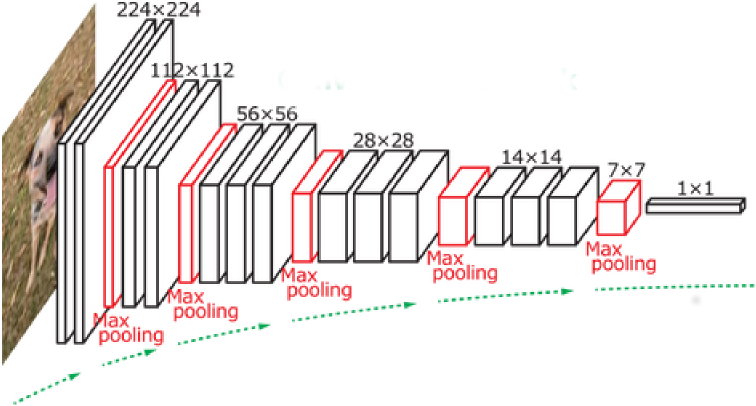
Baseline convolutional network:

- Trained only on our data
- 3 convolutional layers plus 2 fully connected layers
- Used as proof of concept

Architectures

VGG 16:

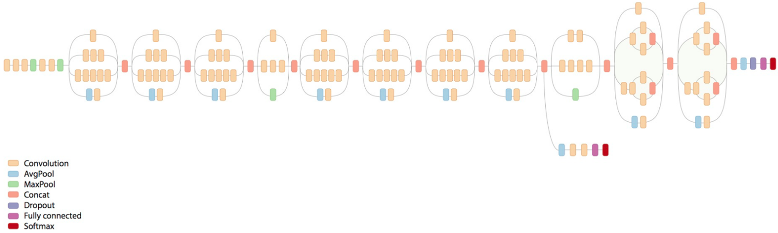
- 16 layers deep
- Performed exceptionally well for its simplicity in the 2014 ImageNet competition



Architectures

GoogLeNet (Inception):

- 22 layers deep
- Introduced inception module
- Won ImageNet competition in 2014
- We used inception V3

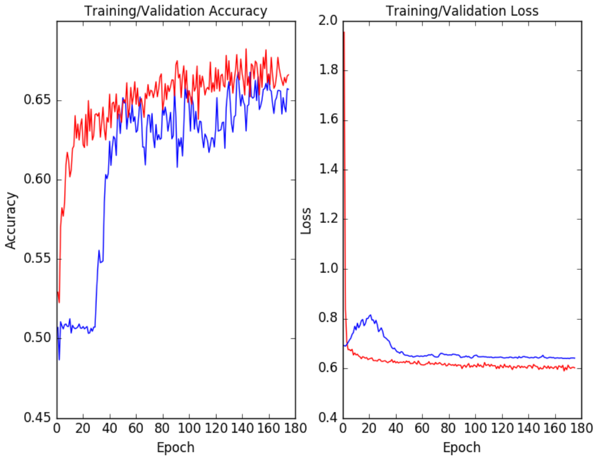


Training/Results

- All neural networks implemented in Keras with a Tensorflow backend
- All GPU intensive computations run on Amazon Web Services (Many thanks to Prof. Hajir and the department)
  - p2.xlarge GPU instances
  - Training time: 1-2 days

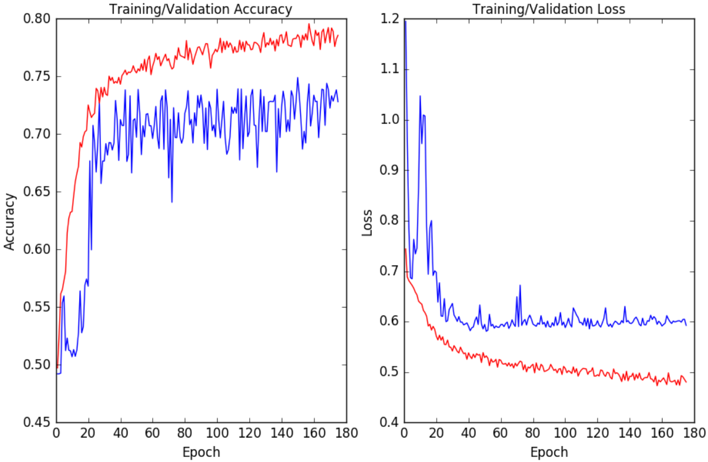
Baseline

Training (red) and Validation (blue) Accuracy / losses for Baseline Network



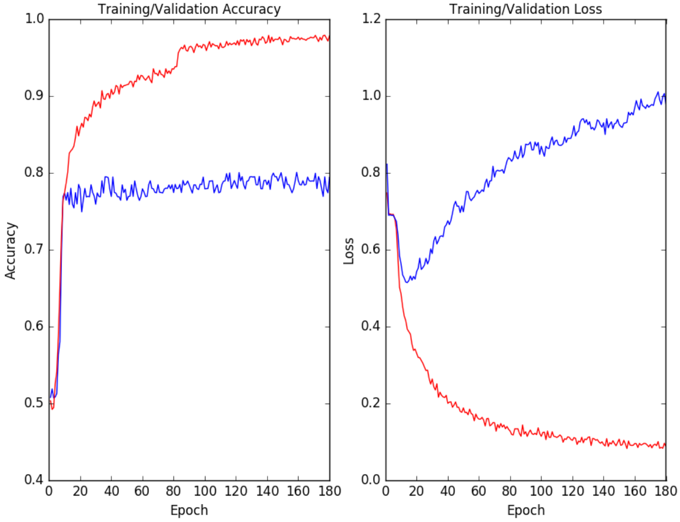
VGG 16

Training (red) and Validation (blue) Accuracy / losses for Transfer Learning on VGG16 Network



Inception V3

Training (red) and Validation (blue) Accuracy / losses for Transfer Learning on Inception V3



Final Results

After 180 epochs:

- Baseline: 65%
- VGG-16: 72%
- GoogLeNet (Inception V3): 78%
- Best in literature: 92%

Modeling Cell Populations

- Want to understand how tumors grow
- Over the years scientists and mathematicians have attempted to model tumor growth
- Our model describes the interaction between the host, effector, and tumor cells

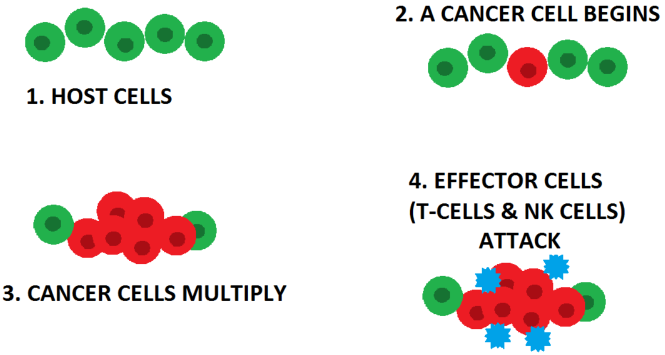


Figure: How Cells interact

System of ODE's

$$\dot{T} = r_1 T(1 - \frac{T}{K_1}) - a_{12}HT - D(E,T)T$$

$$\dot{H} = r_2 H(1 - \frac{H}{K_2}) - a_{21}HT$$

$$\dot{E} = \sigma - d_3 E + g\frac{D^2(E,T)T^2}{h + D^2(E,T)T^2}E - a_{31}TE$$

$$D(E,T) = d\frac{E^\lambda}{sT^\lambda + E^\lambda}$$

- T - tumor cells, H - host cells, E - effector cells
- a - competition terms
- r - individual growth constants
- K - carrying capacity

Nondimensionalized Equations

$$\dot{x} = x(1-x) - a_{12}yx - D(x,z)x$$

$$\dot{y} = r_2 y(1-y) - a_{21}xy$$

$$\dot{z} = 1 - d_3 z + g\frac{D^2(x,z)x^2}{h + D^2(x,z)x^2}z - a_{31}xz$$

$$D(x,z) = d\frac{f^\lambda z^\lambda}{sx^\lambda + f^\lambda z^\lambda}$$

- x - tumor cells
- y - host cells
- z - effector cells

Stable Fixed Points

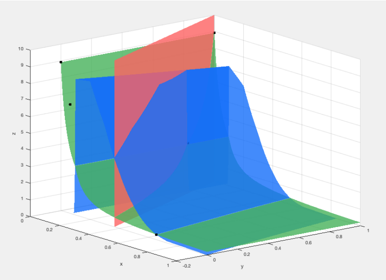


Figure: Graph of nullclines and stable fixed points

- Obtain fixed points by setting $\dot{x} = \dot{y} = \dot{z} = 0$
- $x_1^* = (0, 1, 8.93)$
- $x_2^* = (0.65, 0, 0.31)$
- $x_3^* = (0.06, 0, 6.55)$

Cell Growth

Breast Cancer Diagnosis and Treatment

Applied Mathematics Masters Program

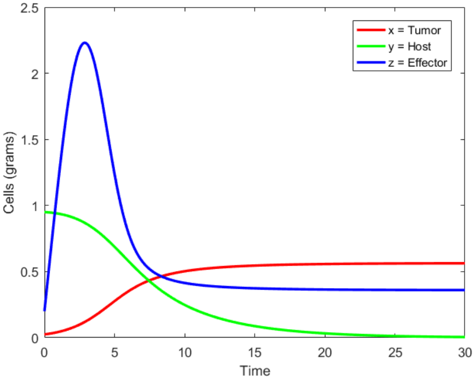


Figure: A function of each cell population over time

Experimental Data

- Need to test the model with other data
- Hiramoto and Ghanta (1974)
  - 36-Day Experiment
  - Day 0: Injected mice with tumor cells
  - Day 10: Cell populations start to change
  - Record cell populations at Days 10,18,21

Breast Cancer Diagnosis and Treatment

Applied Mathematics Masters Program

Process

- Given the data, fit coefficients to approximate the data
- Some coefficients found experimentally
- Use Least Squares to find the others
- We fit d, s, and g
- Solve for x, y, z using RK4

$$\dot{x} = x(1-x) - a_{12}yx - D(x,z)x$$

$$\dot{y} = r_2 y(1-y) - a_{21}xy$$

$$\dot{z} = 1 - d_3 z + g\frac{D^2(x,z)x^2}{h + D^2(x,z)x^2} z - a_{31}xz$$

$$D(x,z) = d\frac{f^\lambda z^\lambda}{s x^\lambda + f^\lambda z^\lambda}$$

Fit of the Data

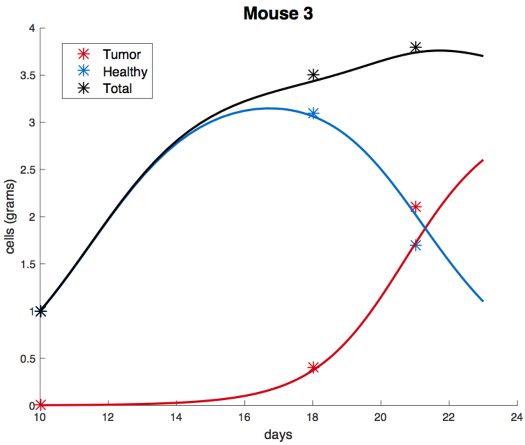


Figure: A function of tumor and healthy cells over time, for Days 10, 18, and 21

Fit of the Data [cont'd]

Breast Cancer Diagnosis and Treatment
Applied Mathematics Masters Program
Introduction
Anomaly Detection
Classification
Classical ML
Neural Networks
Tumor Growth and Treatment
Conclusions
Bibliography

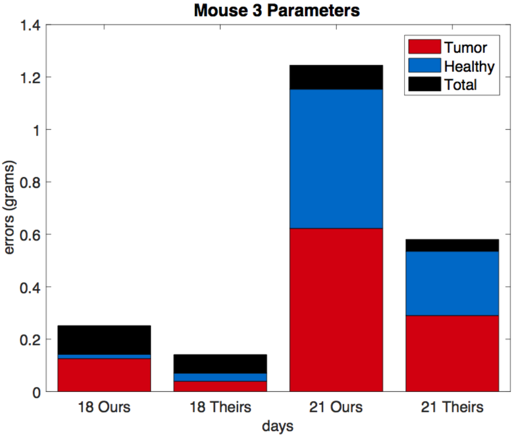


Figure: Residuals of our estimated parameters

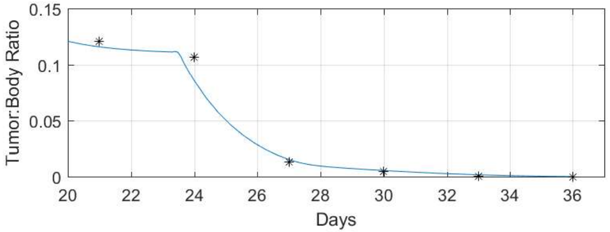
Chemotherapy

- After Day 21, inject mice with chemotherapy drug
  - Record populations at Days 24, 27, 30, 33, 36
- Implications for the Model
  - Chemotherapy targets cancer AND healthy cells
  - However, Hiramoto and Ghanta recorded tumor cell data
  - So for simplicity, only note the effects of chemotherapy on tumor cells
  - Body takes some time to realize what the drug is

$$\dot{x} = x(1-x) - a_{12}yx - D(x,z)x - (1 - e^{-pu(t-\tau)_+})x$$

After Chemotherapy

Figure: Effects of chemotherapy on tumor population.



Future Work

- Mass Detection and Classification
  1. Using newer digital images should improve performance.
  2. Need images from a patient over time to better mimic the true detection process
  3. Explore different neural network architectures
  4. Report metrics like precision and recall
  5. Increase interpretability using new techniques
- Tumor Growth and Treatment
  1. Repeat procedure with a newer data set
  2. Observe tumor cells in breast tissue instead of mice
  3. Analyze effects of chemotherapy on healthy cells

Thank You!

Thank you to Professor Whitaker, Dr. Joseph Polino, Professor Hajir, and everybody in the department.

Questions?

We will now take questions.

Bibliography

1. ''Deep Learning'' by Ian Goodfellow, Yoshua Bengio, Aaron Courville
2. ''Python Machine Learning'' by Sebastien Raschka
3. ''Breast Mass Classification from Mammograms using Deep Convolutional Neural Networks'' by Daniel Lévy, Arzav Jain
4. ''Introduction to Statistical Learning'' by James, Witten, Hastie, and Tibshirani.
5. ''Elements of Statistical Learning'' by Hastie, Tibshirani and Friedman.
6. ''A Validated Mathematical Model of Tumor Growth'' by Alvaro Lopez, Jesus Seoane, and Miguel Sanjuan
7. ''Chemotherapy and Rate of Kill of Tumor Cells in a Mouse Plasmacytoma'' by Raymond Hiramoto and Vithal Ghanta
8. "Automated detection of masses in mammograms by local adaptive thresholding", by Guillaume Kom, Alain Tiedeu, Martin Kom
9. "Pattern Recognition and Machine Learning", by Christopher Bishop
10. "A new feature extraction framework based on wavelets for breast cancer diagnosis", by Semih Ergin, OnurKilinc
11. VGG16 image, Pohang Univ. of Science and Technology Dept. of Computer Science Engineering Computer Vision Lab
12. ''The 9 Deep Learning Papers You Need To Know About'' Adit Deshpande's Blog