Visualizing Neural Nets

By Daniel Fishbein and Jivitesh Debeta

ARITHMOPHOBIA WARNING

The rest of the this presentation may not be appropriate for some audiences.

Viewer discretion is advised.

We wanted to do a reproducibility paper.

That was not what happened….

Summary:

1. What was caution?
2. Explainable AI Background
3. “Viewing” the NN terrain
4. ”No Seriously, Viewing” the NN terrain
5. Make the simplest toy network
6. Can the toy network learn to choose between black white and grey?
7. It behaves like an accurate AI. Which has “limits”
8. What are the limits or boundaries ?
9. Where we are at now
10. Future work
11. What we learned through this process
12. Q&A

Our story starts with “Caution”.

Criticisms of “Caution”.

- “Cool idea but ‘Caution’ is not clearly defined”
- “What are you guys trying to do?”
- “Do you have any metrics for “caution”

“Could you be less vague on what you want to do??”

What we decided to do:

- Focused on when AI messes up
- Can we predict when it will fail?

Can we predict when it will fail?

- Explainable AI - “**Explainable AI** (**XAI**), also known as **Interpretable AI**, or **Explainable Machine Learning** (**XML**),[1] is artificial intelligence (AI) in which humans can understand the reasoning behind decisions or predictions made by the AI.” - Wikipedia
  - Prediction accuracy
    - Accuracy is a key component of how successful the use of AI is in everyday operation. By running simulations and comparing XAI output to the results in the training data set, the prediction accuracy can be determined. The most popular technique used for this is Local Interpretable Model-Agnostic Explanations (LIME), which explains the prediction of classifiers by the ML algorithm.
  - Traceability
    - Traceability is another key technique for accomplishing XAI. This is achieved, for example, by limiting the way decisions can be made and setting up a narrower scope for ML rules and features. An example of a traceability XAI technique is DeepLIFT (Deep Learning Important FeaTures), which compares the activation of each neuron to its reference neuron and shows a traceable link between each activated neuron and even shows dependencies between them.
  - Decision understanding
    - This is the human factor. Many people have a distrust in AI, yet to work with it efficiently, they need to learn to trust it. This is accomplished by educating the team working with the AI so they can understand how and why the AI makes decisions.
      - \- Pulled from IBM

The moment of clarity

Machine learning, specify Neural Nets, is taught as: “The weights are randomly initialized and optimized during the training to minimize a loss function.” - [1]. “The Network is trying to find the ‘local minima’ where it gets the best results” - Every Intoduction to AI ever. ~CSCI 635

What if we didn’t care about finding a better minima and wanted to view the minima we are at?

View the minima

Project Goal

1.) Visualize what this landscape could look like.
2.) Visualise this landscape in an actual NN

Previous work:

neural network output visualized

About 322,000 results (**0.11** sec)

[BOOK] Opening the black box-data driven **visualization** of **neural networks** — [PDF] escholarship.org

FY Tzeng, KL Ma - 2005 - ieeexplore.ieee.org

… To highlight the information in a **neural network**, we color the input and **output** nodes based on a selected voxel's value, where low value maps to blue, middle value maps to yellow, and …

Save Cite Cited by 197 Related articles All 11 versions

Evaluating the **visualization** of what a deep **neural network** has learned — [PDF] ieee.org, more! @ RIT

W Samek, A Binder, G Montavon… - … on **neural networks** …, 2016 - ieeexplore.ieee.org

… This principle ensures that the **network output** activity is fully redistributed through the layers of a DNN onto the input variables, ie, neither positive nor negative evidence is lost. In the …

Save Cite Cited by 1010 Related articles All 8 versions Web of Science: 365

Full-gradient representation for **neural network visualization** — [PDF] neurips.cc

S Srinivas, F Fleuret - Advances in **neural** information …, 2019 - proceedings.neurips.cc

… We measure the **neural network** function **output** for the most confident class, before and after perturbation, and plot the absolute value of the fractional difference. We use our pixel …

Save Cite Cited by 164 Related articles All 9 versions

[HTML] How convolutional **neural networks** diagnose plant disease — [HTML] science.org

Y Toda, F Okura - Plant Phenomics, 2019 - spj.science.org

… variety of **neuron**-wise and layer-wise **visualization** methods … We showed that **neural networks**

Previous work:

neural network output visualized

About 322,000 results (**0.11** sec)

[BOOK] Opening the black box-data driven **visualization** of **neural networks** — [PDF] escholarship.org

FY Tzeng, KL Ma - 2005 - ieeexplore.ieee.org

… To highlight the information in a **neural network**, we color the input and **output** nodes based on a selected voxel's value, where low value maps to blue, middle value maps to yellow, and …

Save Cite Cited by 197 Related articles All 11 versions

Evaluating the **visualization** of what a deep **neural network** has learned — [PDF] ieee.org more! @ RIT

W Samek, A Binder, G Montavon… - … on **neural networks** …, 2016 - ieeexplore.ieee.org

… This principle ensures that the **network output** activity is fully redistributed through the layers of a DNN onto the input variables, ie, neither positive nor negative evidence is lost. In the …

Save Cite Cited by 1010 Related articles All 8 versions Web of Science: 365

Full-gradient representation for **neural network visualization** — [PDF] neurips.cc

S Srinivas, F Fleuret - Advances in **neural** information …, 2019 - proceedings.neurips.cc

… We measure the **neural network** function **output** for the most confident class, before and after perturbation, and plot the absolute value of the fractional difference. We use our pixel …

Save Cite Cited by 164 Related articles All 9 versions

[HTML] How convolutional **neural networks** diagnose plant disease — [HTML] science.org

Y Toda, F Okura - Plant Phenomics, 2019 - spj.science.org

… variety of **neuron**-wise and layer-wise **visualization** methods … We showed that **neural networks**

Previous work:

neural network output visualized

About 322,000 results (**0.11** sec)

[BOOK] Opening the black box-data driven **visualization** of **neural networks** — [PDF] escholarship.org
FY Tzeng, KL Ma - 2005 - ieeexplore.ieee.org
… To highlight the information in a **neural network**, we color the input and **output** nodes based on a selected voxel's value, where low value maps to blue, middle value maps to yellow, and …
Save Cite Cited by 197 Related articles All 11 versions

Evaluating the **visualization** of what a deep **neural network** has learned — [PDF] ieee.org
more! @ RIT
W Samek, A Binder, G Montavon… - … on **neural networks** …, 2016 - ieeexplore.ieee.org
… This principle ensures that the **network output** activity is fully redistributed through the layers of a DNN onto the input variables, ie, neither positive nor negative evidence is lost. In the …
Save Cite Cited by 1010 Related articles All 8 versions Web of Science: 365

Full-gradient representation for **neural network visualization** — [PDF] neurips.cc
S Srinivas, F Fleuret - Advances in **neural** information …, 2019 - proceedings.neurips.cc
… We measure the **neural network** function **output** for the most confident class, before and after perturbation, and plot the absolute value of the fractional difference. We use our pixel …
Save Cite Cited by 164 Related articles All 9 versions

~~[HTML] How convolutional **neural networks** diagnose plant disease~~ — ~~[HTML] science.org~~
~~Y Toda, F Okura - Plant Phenomics, 2019 - spj.science.org~~
~~… variety of **neuron**-wise and layer-wise **visualization** methods … We showed that **neural networks**~~

Previous work:

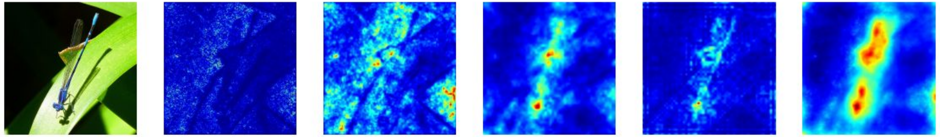


Figure 1: Visualization of bias-gradients at different layers of a VGG-16 pre-trained neural network. While none of the intermediate layer bias-gradients themselves demarcate the object satisfactorily, the full-gradient map achieves this by aggregating information from the input-gradient and all intermediate bias-gradients. (see Equation 2).

Full-Gradient Representation for Neural Network Visualization

Suraj Srinivas
Idiap Research Institute & EPFL
suraj.srinivas@idiap.ch

François Fleuret
Idiap Research Institute & EPFL
francois.fleuret@idiap.ch

Abstract

We introduce a new tool for interpreting neural net responses, namely full-gradients, which decomposes the neural net response into input sensitivity and per-neuron sensitivity components. This is the first proposed representation which satisfies two key properties: *completeness* and *weak dependence*, which provably cannot be satisfied by any saliency map-based interpretability method. For convolutional nets, we also propose an approximate saliency map representation, called *FullGrad*, obtained by aggregating the full-gradient components.

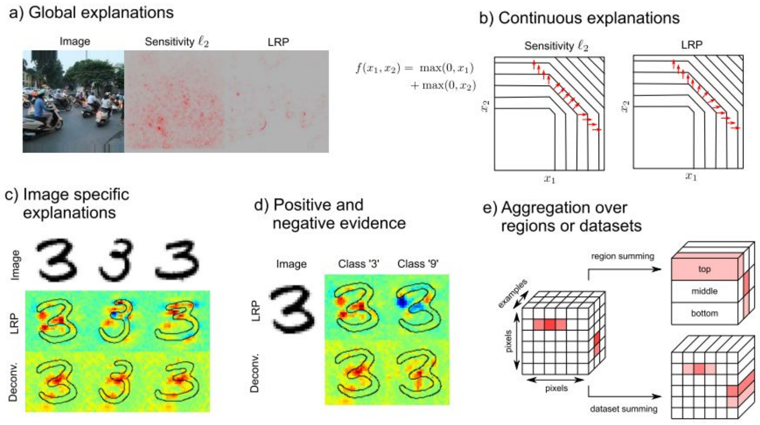
Previous work:

Evaluating the Visualization of What a Deep Neural Network Has Learned

Wojciech Samek, *Member, IEEE*, Alexander Binder, *Member, IEEE*, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller, *Member, IEEE*

***Abstract*— Deep neural networks (DNNs) have demonstrated impressive performance in complex machine learning tasks such as image classification or speech recognition. However, due to their multilayer nonlinear structure, they are not transparent, i.e., it is hard to grasp *what* makes them arrive at a particular classification or recognition decision, given a new unseen data sample. Recently, several approaches have been proposed enabling one to understand and interpret the reasoning embodied in a DNN for a single test image. These methods quantify the "importance" of individual pixels with respect to the classification decision and allow a visualization in terms of a heatmap in pixel/input space. While the usefulness of heatmaps can be judged subjectively by a human, an objective quality measure is missing. In this paper, we present a general methodology based on region perturbation for evaluating ordered collections of pixels such as heatmaps. We compare heatmaps computed by three different methods on the SUN397, ILSVRC2012, and MIT Places data sets.**

mated image classification [1]–[4], natural language processing [5], [6], human action recognition [7], [8], or physics [9] (see also [10]). Since DNN training methodologies (unsupervised pretraining, dropout, parallelization, GPUs, etc.) have been improved [11], DNNs are recently able to harvest extremely large amounts of training data and can thus achieve record performances in many research fields. At the same time, DNNs are generally conceived as black box methods, and users might consider this lack of transparency a drawback in practice. Namely, it is difficult to intuitively and quantitatively understand the result of DNN inference, i.e., for an *individual* novel input data point, *what* made the trained DNN model arrive at a particular response. Note that this aspect differs from feature selection [12], where the question is: which fea-



Previous work:

Opening the Black Box — Data Driven Visualization of Neural Networks

Fan-Yin Tzeng* Kwan-Liu Ma*

Department of Computer Science
University of California at Davis

ABSTRACT

Arti cial neural networks are computer software or hardware models inspired by the structure and behavior of neurons in the human nervous system. As a powerful learning tool, increasingly neural networks have been adopted by many large-scale information processing applications but there is no a set of well de ned criteria for choosing a neural network. The user mostly treats a neural network as a black box and cannot explain how learning from input data was done nor how performance can be consistently ensured. We have experimented with several information visualization designs aiming to open the black box to possibly uncover underlying dependencies between the input data and the output data of a neural network. In this paper, we present our designs and show that the visualizations not only help us design more ef cient neural networks, but also assist us in the process of using neural networks for problem solving such as performing a classi cation task.

error bound, learning rate, training algorithm, hidden layer size, and the data vector used, are often chosen in a trial-and-error process.

We believe visualization, which proves to help illustrate and understand the behaviors of complex systems, can also help us understand ANNs and design better ANNs. Previous attempts in using visualization to gain understanding into ANNs, as discussed in Section 3, mainly studied the weights and connections of a neural network and analyzed neural networks in isolation; the data used by the neural network were mostly not looked at.

We therefore take a data-driven approach to the problem of visualizing ANN since gaining insights into a neural network requires the study of not only the network but also how it responds to the input data that it was designed to process. The methods we present enable the interactive exploration of both the input data and the neural network so as to gain more complete picture of how the neural network performs its task. The visualizations can also assist in the selection of network structure and other parameters for an assigned

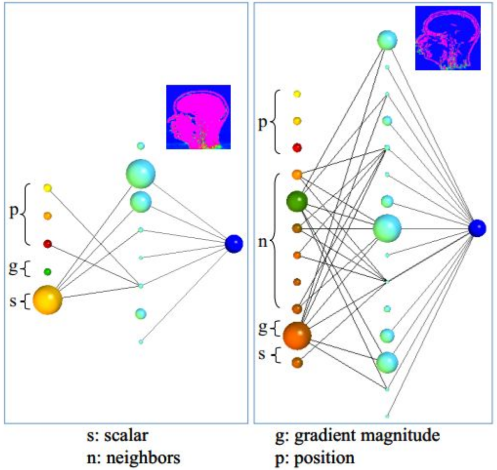
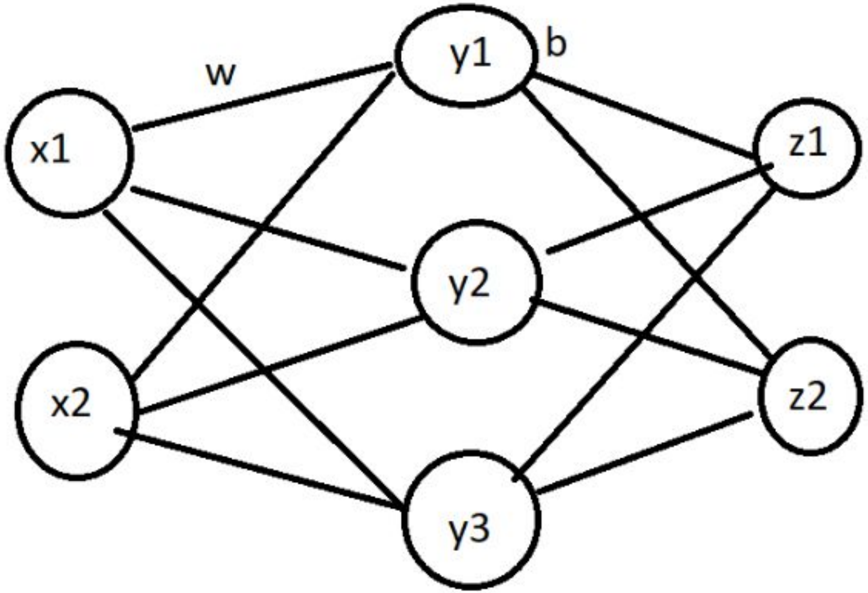


Figure 4: The left image shows a neural network which is trained for classifying the entire head from the data set. The scalar value is the main criterion considered in this classification. The right image is the result of classifying the boundaries. In this case, neighbors and gradient magnitude are shown to be more important. The classification result is shown at the upper right of each network.

Previous work: NONE (:

Our First Task: Visualize what this landscape could look like.



$$\sigma(x) = \frac{1}{1+e^{-x}}$$

$$y_i = \sigma(w_{x_1->y_i} * x_1 + w_{x_2->y_i} * x_2 + b_{y_i})$$

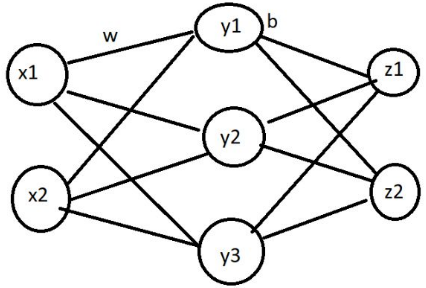
$$z_i = \sigma(w_{y_1->z_i} * y_1 + w_{y_2->z_i} * y_2 + w_{y_3->z_i} * y_3 + b_{z_i})$$

ARITHMOPHOBIA WARNING

The rest of the this presentation may not be appropriate for some audiences.

Viewer discretion is advised.

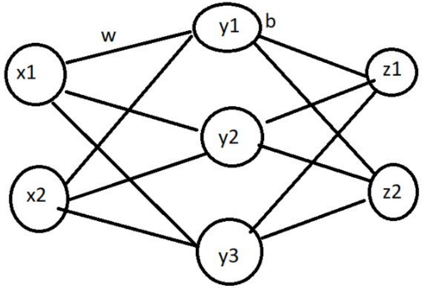
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our First Task: Visualize what this landscape could look like.

$$\Big(1 + \exp\Big[-\Big(w_{y_1->z_1} * \frac{1}{1+\exp[w_{x_1->y_1} * x_1 + w_{x_2->y_1} * x_2 + b_{y_1}]} + w_{y_2->z_1} * \frac{1}{1+\exp[w_{x_1->y_2} * x_1 + w_{x_2->y_2} * x_2 + b_{y_2}]} + w_{y_3->z_1} * \frac{1}{1+\exp[w_{x_1->y_3} * x_1 + w_{x_2->y_3} * x_2 + b_{y_3}]} + b_{z_1}\Big)\Big]\Big)^{-1}$$

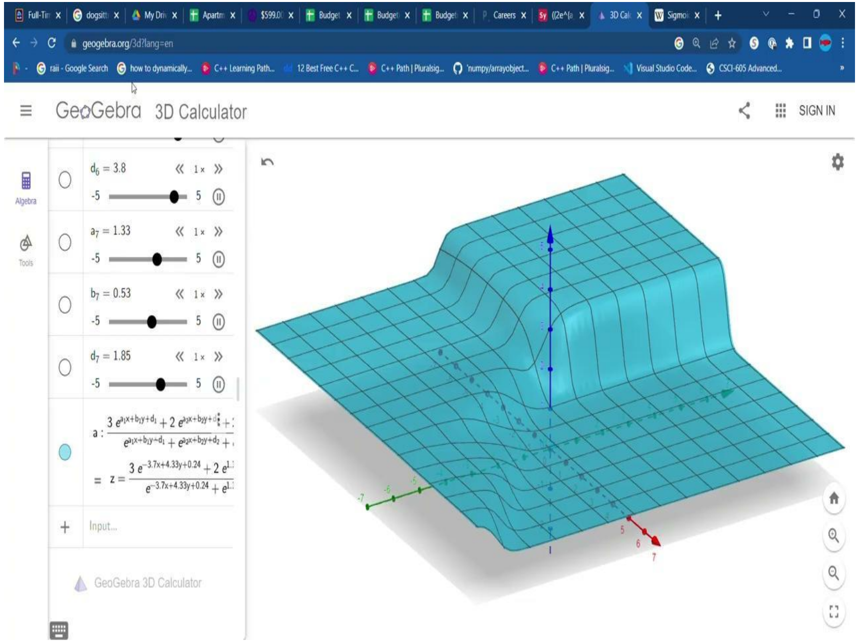
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our First Task: Visualize what this landscape could look like.

$$\frac{3e^{a_1x+b_1y+d_1} + 2e^{a_2x+b_2y+d_2} + 2e^{a_3x+b_3y+d_3} + 2e^{a_4x+b_4y+d_4} + e^{a_5x+b_5y+d_5} + e^{a_6x+b_6y+d_6} + e^{a_7x+b_7y+d_7}}{e^{a_1x+b_1y+d_1} + e^{a_2x+b_2y+d_2} + e^{a_3x+b_3y+d_3} + e^{a_4x+b_4y+d_4} + e^{a_5x+b_5y+d_5} + e^{a_6x+b_6y+d_6} + e^{a_7x+b_7y+d_7}}$$

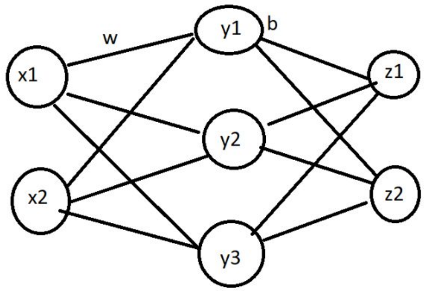
Our First Results: Visualize what this landscape could look like.



Project Goal

1.) ~~Visualize what this landscape could look like.~~
2.) Visualise this landscape in an actual NN

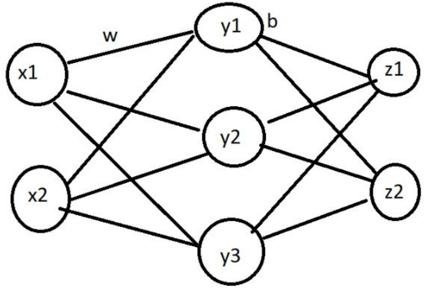
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our Second Task: Visualise this landscape in an actual NN

1.) Make this network in Tensorflow or Pytorch
2.) Give it a task to optimize
3.) Visualize the output

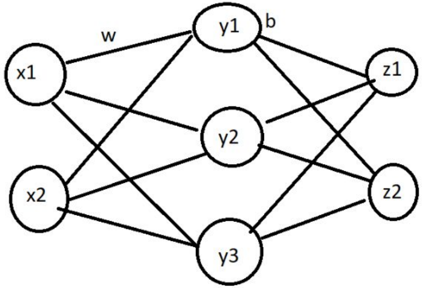
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our Second Task: Visualise this landscape in an actual NN

1.) ~~Make this network in Tensorflow or Pytorch~~
2.) Give it a task to optimize
3.) Visualize the output

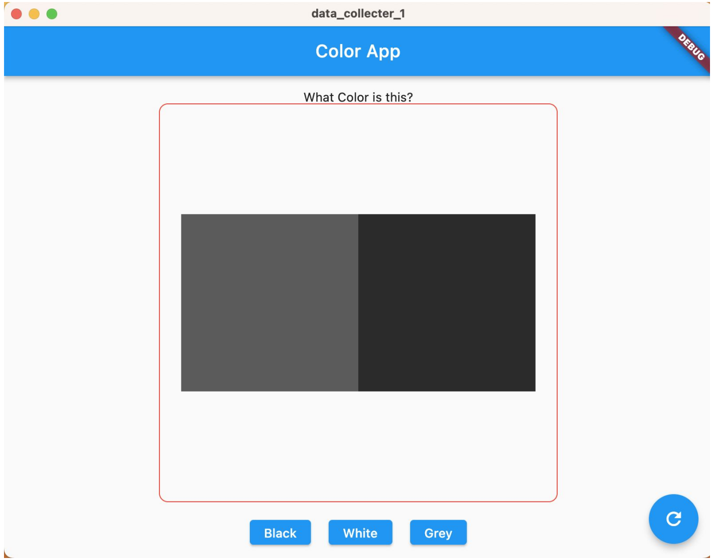
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



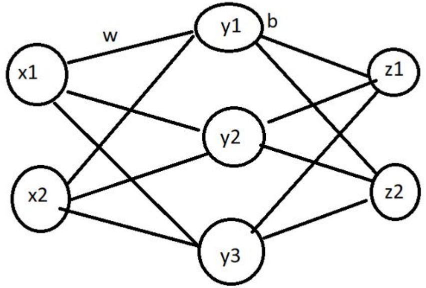
Our Experiment: Black, White, Grey Experiment

Task: Given 2 pixels, are `they` Black, White, or Grey?

a.) Mathematically defined dataset
b.) Human labeled dataset



$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our Experiment: Black, White, Grey Experiment

Task: Given 2 pixels, are they Black, White, or Grey?

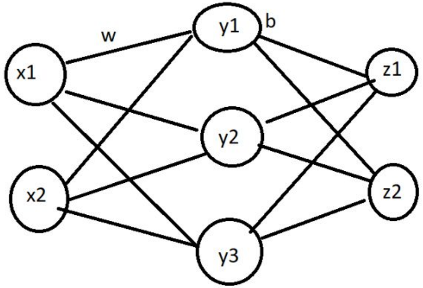
a. Mathematically defined dataset
   i. S1 DEF
      - P1 >P2 | P1<128
      - P1 <P2 | P2>128
      - P1 + P2 =128 | otherwise
   ii. S2 DEF :
      - P1 + P2 < 255 | P1>P2
      - P1 + P2 <255 | P2<P1
      - P1 + P2 =128 | otherwise
   iii. S3 DEF
      - P1 + P2 < 128 | P1<128 P2<128
      - P1 + P2 >128 | P1<128 P2<128
      - Every 3rd image
b. Human labeled dataset

```
if (x1 + x2) < (2*pixel_val_min + n):
    # WHITE
```

```
elif (x1 + x2) > (2*pixel_val_max - n):
    # BLACK
```

```
else:
    # GREY
```

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our Experiment: Black, White, Grey Experiment

Task: Given 2 pixels, are they Black, White, or Grey?

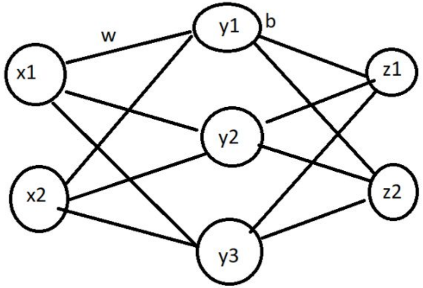
a.) Mathematically defined dataset
  i.) S1 DEF
    (1) P1 >P2 | P1<128
    (2) P1 <P2 | P2>128
    (3) P1 + P2 =128 | otherwise
  ii.) S2 DEF :
    (1) P1 + P2 < 255 | P1>P2
    (2) P1 + P2 <255 | P2<P1
    (3) P1 + P2 =128 | otherwise
  iii.) S3 DEF
    (1) P1 + P2 < 128 | P1<128 P2<128
    (2) P1 + P2 >128 | P1<128 P2<128
    (3) Every 3rd image
b.) Human labeled dataset
  i.) A human was given 2 squares and asked if the image is black white or grey.
  ii.) The dataset is called “human.zip” and is available at: https://drive.google.com/file/d/1E5riOt3Dg_wGxwqwxixbAvKhB1-GSg4o/view?usp=share_link

```
if (x1 + x2) < (2*pixel_val_min + n):
    # WHITE
```

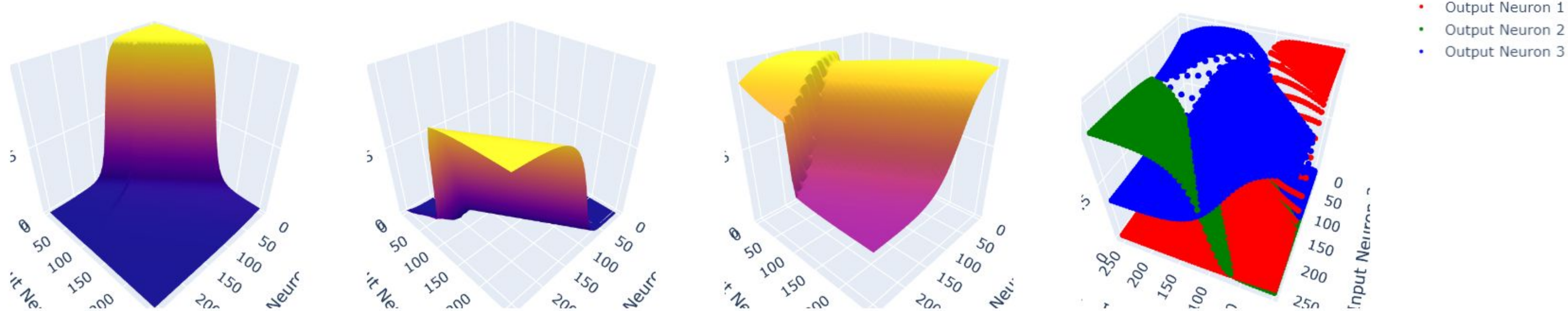
```
elif (x1 + x2) > (2*pixel_val_max - n):
    # BLACK
```

```
else:
    # GREY
```

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



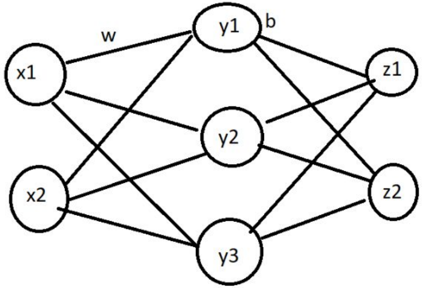
Our results to Black, White, Grey:



Project Goal

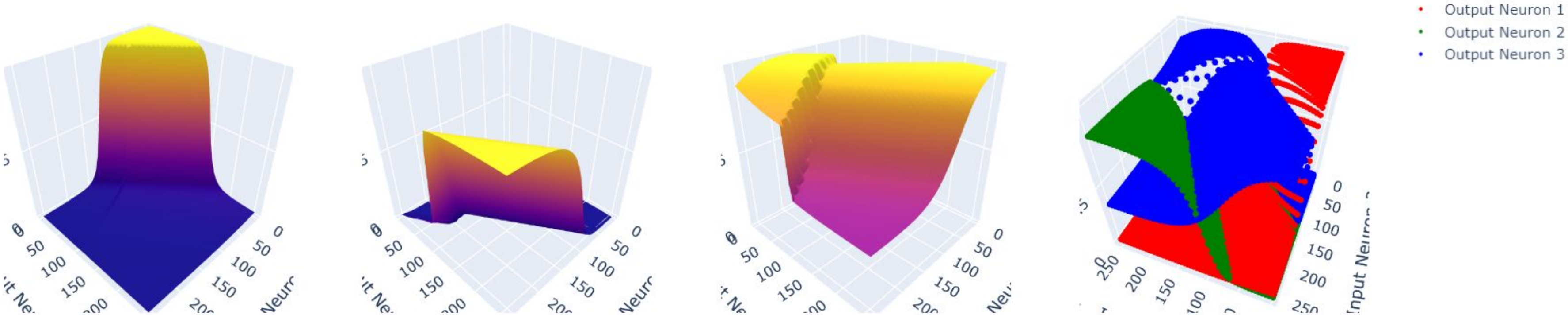
1.) ~~Visualize what this landscape could look like.~~
2.) ~~Visualise this landscape in an actual NN~~
3.) Find limits/boundaries between these plains
    a.) Highlight their intersections

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

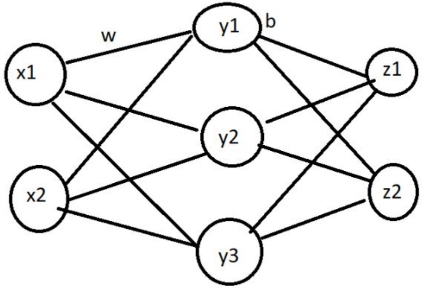


Our Third Task: Find limits/boundaries between these plains and highlight their intersections

- If each plane in this image is an output neuron, than they intersect when (Z_1 = Z_2), (Z_1 = Z_3), (Z_2 = Z_3)



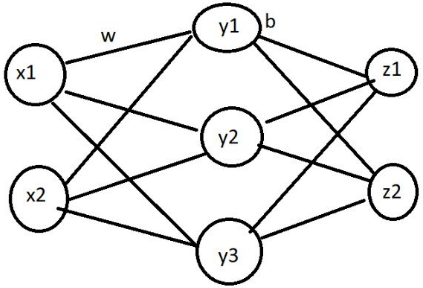
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our Third Task: Find limits/boundaries between these plains and highlight their intersections

$$\Big(1 + \exp\Big[-\Big(w_{y_1->z_1} * \frac{1}{1+\exp[w_{x_1->y_1} * x_1 + w_{x_2->y_1} * x_2 + b_{y_1}]} + w_{y_2->z_1} * \frac{1}{1+\exp[w_{x_1->y_2} * x_1 + w_{x_2->y_2} * x_2 + b_{y_2}]} + w_{y_3->z_1} * \frac{1}{1+\exp[w_{x_1->y_3} * x_1 + w_{x_2->y_3} * x_2 + b_{y_3}]} + b_{z_1}\Big)\Big]\Big)^{-1}$$

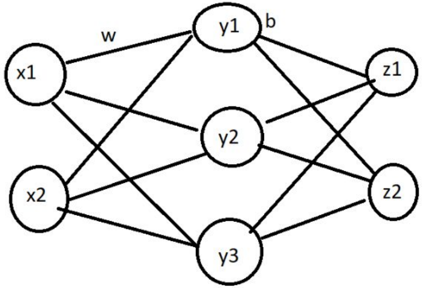
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



<u>Our Third Task:</u> Find limits/boundaries between these plains and highlight their intersections

$$\begin{aligned}
&(1+\exp[-(w_{y_1->z_1} * \frac{1}{1+\exp[w_{x_1->y_1} * x_1 + w_{x_2->y_1} * x_2 + b_{y_1}]} \\
&\quad + w_{y_2->z_1} * \frac{1}{1+\exp[w_{x_1->y_2} * x_1 + w_{x_2->y_2} * x_2 + b_{y_2}]} \\
&\quad + w_{y_3->z_1} * \frac{1}{1+\exp[w_{x_1->y_3} * x_1 + w_{x_2->y_3} * x_2 + b_{y_3}]} \\
&\quad + b_{z_1})])^{-1} \\
&= \\
&(1+\exp[-(w_{y_1->z_2} * \frac{1}{1+\exp[w_{x_1->y_1} * x_1 + w_{x_2->y_1} * x_2 + b_{y_1}]} \\
&\quad + w_{y_2->z_2} * \frac{1}{1+\exp[w_{x_1->y_2} * x_1 + w_{x_2->y_2} * x_2 + b_{y_2}]} \\
&\quad + w_{y_3->z_2} * \frac{1}{1+\exp[w_{x_1->y_3} * x_1 + w_{x_2->y_3} * x_2 + b_{y_3}]} \\
&\quad + b_{z_2})])^{-1}
\end{aligned}$$

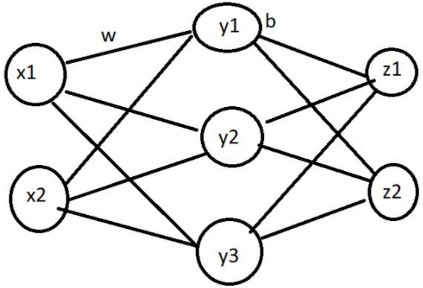
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



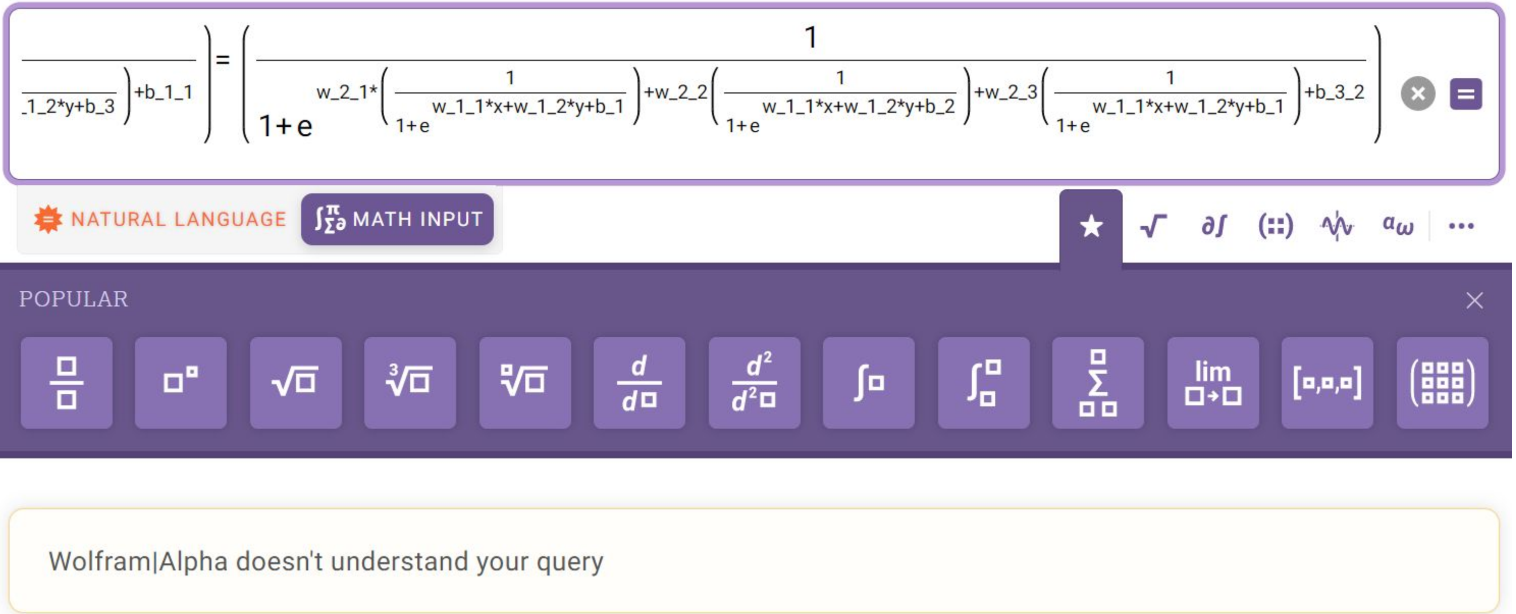
Our Third Task: Find limits/boundaries between these plains and highlight their intersections

$$\frac{\qquad}{\text{\_1\_2*y+b\_3}}\Bigg)\text{+b\_1\_1}\Bigg) = \left(\frac{1}{1+e^{\text{w\_2\_1*}\left(\frac{1}{1+e^{\text{w\_1\_1*x+w\_1\_2*y+b\_1}}}\right)\text{+w\_2\_2}\left(\frac{1}{1+e^{\text{w\_1\_1*x+w\_1\_2*y+b\_2}}}\right)\text{+w\_2\_3}\left(\frac{1}{1+e^{\text{w\_1\_1*x+w\_1\_2*y+b\_1}}}\right)\text{+b\_3\_2}}}\right)$$

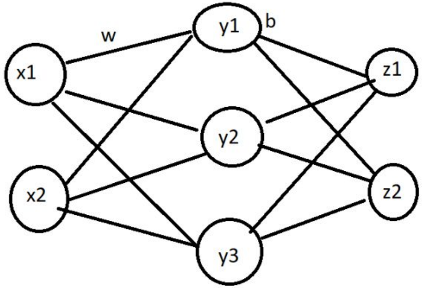
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our Third Task: Find limits/boundaries between these plains and highlight their intersections



$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our Third Task: Find limits/boundaries between these plains and highlight their intersections

$$C_0 + C_1 e^{B_1} + C_2 e^{B_2} + C_3 e^{B_3} + C_4 e^{B_2+B_3} + C_5 e^{B_1+B_3} + C_6 e^{B_1+B_2} + C_7 e^{B_1+B_2+B_3} = 0$$

$$\begin{gathered}
C_0 = w_{y_1->z_1} - w_{y_1->z_2} + w_{y_2->z_1} - w_{y_2->z_2} + w_{y_3->z_1} - w_{y_3->z_2} - b_{z_2} + b_{z_1} \\
C_3 = (w_{y_1->z_1} - w_{y_1->z_2} + w_{y_2->z_1} - w_{y_2->z_2} - b_{z_2} + b_{z_1}) \\
C_2 = (w_{y_1->z_1} - w_{y_1->z_2} + w_{y_3->z_1} - w_{y_3->z_2} - b_{z_2} + b_{z_1}) \\
C_1 = (w_{y_2->z_1} - w_{y_2->z_2} + w_{y_3->z_1} - w_{y_3->z_2} - b_{z_2} + b_{z_1}) \\
C_4 = (w_{y_1->z_1} - w_{y_1->z_2} - b_{z_2} + b_{z_1}) \\
C_5 = (w_{y_2->z_1} - w_{y_2->z_2} - b_{z_2} + b_{z_1}) \\
C_6 = (w_{y_3->z_1} - w_{y_3->z_2} - b_{z_2} + b_{z_1}) \\
C_7 = (-b_{z_2} + b_{z_1})
\end{gathered} \tag{69}$$

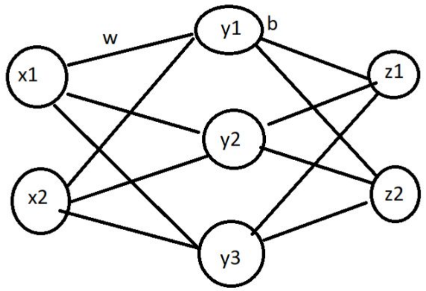
16

$$B_1 = -w_{x_1->y_1} * x_1 + w_{x_2->y_1} * x_2 + b_{y_1}$$

$$B_2 = -w_{x_1->y_2} * x_1 + w_{x_2->y_2} * x_2 + b_{y_2}$$

$$B_3 = -w_{x_1->y_3} * x_1 + w_{x_2->y_3} * x_2 + b_{y_3}$$

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Our Third Task: Find limits/boundaries between these plains and highlight their intersections

$$e^{3x-5y} + e^{2x+y} = 10$$

Solutions

Exact forms

$y \approx -i\,(6.28319\,n - i\log(\text{Root}[1\,\#1^6\,2.71828^{3x} - 10\,\#1 + 1 \times 2.71828^{2x}\,\&, 1]))$,
$n \in \mathbb{Z}$

$y \approx -i\,(6.28319\,n - i\log(\text{Root}[1\,\#1^6\,2.71828^{3x} - 10\,\#1 + 1 \times 2.71828^{2x}\,\&, 2]))$,
$n \in \mathbb{Z}$

$y \approx -i\,(6.28319\,n - i\log(\text{Root}[1\,\#1^6\,2.71828^{3x} - 10\,\#1 + 1 \times 2.71828^{2x}\,\&, 3]))$,
$n \in \mathbb{Z}$

$y \approx -i\,(6.28319\,n - i\log(\text{Root}[1\,\#1^6\,2.71828^{3x} - 10\,\#1 + 1 \times 2.71828^{2x}\,\&, 4]))$,
$n \in \mathbb{Z}$

$y \approx -i\,(6.28319\,n - i\log(\text{Root}[1\,\#1^6\,2.71828^{3x} - 10\,\#1 + 1 \times 2.71828^{2x}\,\&, 5]))$,
$n \in \mathbb{Z}$

Our Third Task: Find limits/boundaries between these plains and highlight their intersections

Given:

$$\frac{dy}{dx} = ky$$

The General solution is:

$$y = Ce^{kx}$$

Our Third Task: Find limits/boundaries between these plains and highlight their intersections

Given:

$$\frac{dy}{dx} = ky$$

The General solution is:

$$y = Ce^{kx}$$

Our equation:

$$C_0 + C_1e^{B_1} + C_2e^{B_2} + C_3e^{B_3} +$$
$$C_4e^{B_2+B_3} + C_5e^{B_1+B_3} + C_6e^{B_1+B_2} +$$
$$C_7e^{B_1+B_2+B_3} = 0$$

And this is where we are at now:

Our equation(s):

$$C_0 + C_1 e^{B_1} + C_2 e^{B_2} + C_3 e^{B_3} + C_4 e^{B_2+B_3} + C_5 e^{B_1+B_3} + C_6 e^{B_1+B_2} + C_7 e^{B_1+B_2+B_3} = 0$$

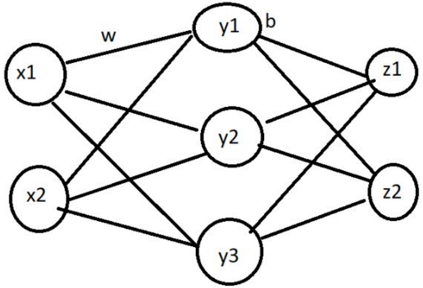
$$\theta(\begin{bmatrix} \theta(A_1) \\ \theta(A_2) \\ \theta(A_3) \end{bmatrix} \begin{bmatrix} w_{z11} & w_{z12} & w_{z13} \\ w_{z21} & w_{z22} & w_{z23} \\ w_{z31} & w_{z32} & w_{z33} \end{bmatrix} + \begin{bmatrix} \begin{bmatrix} b_{z1} \\ b_{z2} \\ b_{z3} \end{bmatrix} \end{bmatrix}) \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = 0$$

$$A_1 = (\sum_{i=1,j}^{len([x])} w_{yi,j=1} * x_i) + b_{yj=1}$$

$$A_2 = (\sum_{i=1,j}^{len([x])} w_{yi,j=2} * x_i) + b_{yj=2}$$

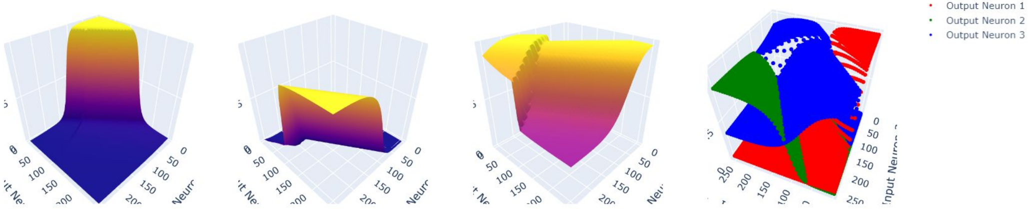
$$A_3 = (\sum_{i=1,j}^{len([x])} w_{yi,j=3} * x_i) + b_{yj=3}$$

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



And this is where we are at now:

Model: GaussianNB
Accuracy: 0.52




```
Modifying pixel:
Row:   0
Column:  15
Pixel_val: +1
prediction from:   9 to:   [2]
```

Original image:

Modified image:

Pixel that changed

Future work:

- Work back into Diff. Eq. and use numerical methods to approximate x1 and x2… Xn
    - This will be like nested integral That we do not want to actually solve but rather approximate
    - This work moves into the Boundary Value problem which has extensive research but needs more time to understand.
- More work into working with Non-linear functions
- Make an algorithm that given a feedforward NN architecture can output the points that are boundaries for the NN classifications.

Add recap slide of everything we did in timeline order

- Presented “Caution”
- Narrowed our question to “Can we predict when a NN will fail?”
- Decided to try and visualize a given NN
- Found no prior work
- Modeled what the surface of an NN equation for a single output looks like
- Designed and executed the “Black White Grey Experiment”
- Visualize the output
- Attempted to solve for the boundaries
- Left off looking into the Boundary value problem

What we learned through this process:

- Sometimes even obvious seaming questions have not been asked yet.
- It can be very hard to explain a new idea to someone without visuals
- Without clear direction it can be easy to lose focus and start spending time on something irrelevant
- Different people will have different strengths and weaknesses
    - This should be leveraged, not dismissed
- Communication is hard, both professionally and personally
- A difficult problem takes a long time to solve, having faith helps putting in time to solve the impossible ones.
- Perfection will always be something at infinity . However it can make you de-focus yourself from time constrained decision making problems

Sources

https://ai.plainenglish.io/neural-networks-for-beginners-b54e8e118c7

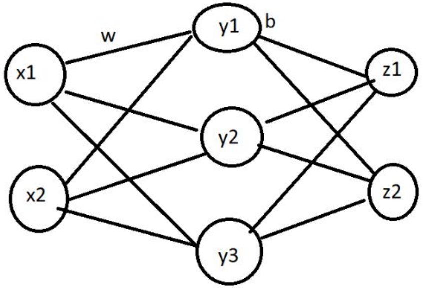
https://www.ibm.com/watson/explainable-ai
https://ieeexplore.ieee.org/abstract/document/1532820
https://ieeexplore.ieee.org/abstract/document/7552539
https://proceedings.neurips.cc/paper/2019/hash/80537a945c7aaa788ccfcdf1b99b5d8f-Abstract.html
https://en.wikipedia.org/wiki/Explainable_artificial_intelligence
https://www.wolframalpha.com/

https://drive.google.com/file/d/1E5riOt3Dg_wGxwqwxixbAvKhB1-GSg4o/view?usp=share_link

Q&A

Model: GaussianNB
Accuracy: 0.52

```
Modifying pixel:
Row:    0
Column: 15
Pixel_val: +1
prediction from:   9 to:   [2]
```



Original image:

Modified image: