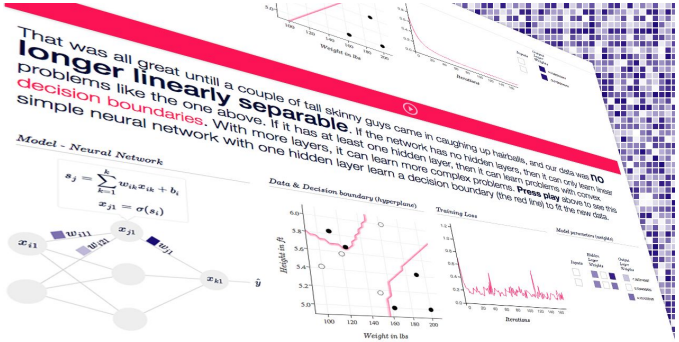


Neural Networks - A (very*) light introduction

W209 final project Fall 2016

Kyle Hamilton
Jun Luo
Walter Alfredo Erquinigo Pezo
Linfang Yang

What problem are we trying to solve?

Goal: The visualization is an introduction to Neural Networks with a view to facilitate the learning process for incoming Machine Learning students. With this intuition we expect that it will be easier to learn more complex concepts.

The animations and illustrations should provide intuition, but are not designed to be comprehensive. For this reason, we intentionally skipped many details.

Audience: The expected audience are students with some very rudimentary understanding of linear models, and a notion of what a Neural Network is.

A background in math is not required.

Idea evolution

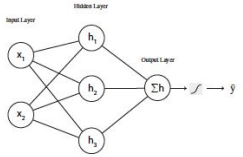
The following screens contain initial designs and concepts.

These illustrations show the complexity of networks and the challenge of simplifying these representations without losing important intuition.

It quickly became apparent that showing detailed math equations and trying to account for every operation was cumbersome and confusing.

We decided to focus on one particular aspect of NN mechanics that is not typically visualized - namely the model parameters. This focus makes our project unique in a saturated space.

Feedforward network with a 2 dimensional input X, and one hidden layer of dimesion 3, and a single output y



INPUT
2 dimensional vector
x: [x_1 x_2]

shape: `1 by 2`

WEIGHTS
initialize weight matrix with the appropriate shape:
depends on x dimension and hidden layer dimention

w: [
[w_{11} w_{12} w_{13}]
[w_{21} w_{22} w_{23}]
]

shape: `2 by 3`

OUTPUT
$\hat{y}$: y

shape: `1`

Hidden Layer

h = σ(xw)

xw => `1 by 2` · `2 by 3` => `1 by 3`

must be same

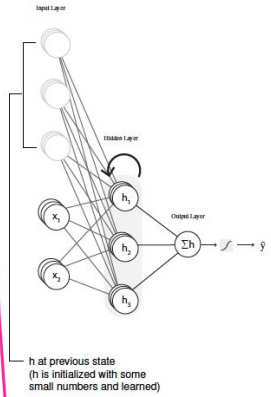
h = σ(xw) => `1 by 3`

Output Layer

Σh => `1`

Recurrent Neural Network

1



h at previous state
(h is initialized with some
small numbers and learned)

INPUT
3 x2 dimensional matrix - a sequense (sentence) of 3 words, where each word is a 2 dimensional embedding
Each row of X is a word embedding concatenated with a previous state.

x: [
[x_{11} x_{12} h_{11} h_{12} h_{13}]
[x_{21} x_{22} h_{21} h_{22} h_{23}]
[x_{31} x_{32} h_{31} h_{32} h_{33}]
]

shape: `3 by (2 + 3)`

WEIGHTS
initialize weight matrix with the appropriate shape:
depends on x dimension and hidden layer dimention

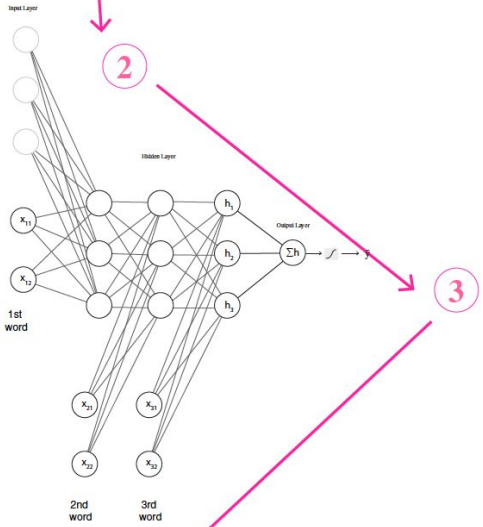
`RE-USE FOR EACH ROW OF X - So each word gets the same weights`

w: [
[w_{x11} w_{x12} w_{x13}]
[w_{x21} w_{x22} w_{x23}]
[w_{h11} w_{h12} w_{h13}]
[w_{h21} w_{h22} w_{h23}]
[w_{h31} w_{h32} w_{h33}]
]

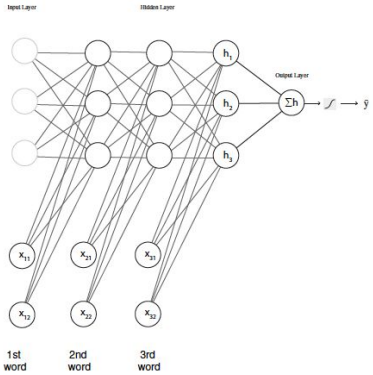
shape: `(2 + 3) by 3`

Recurrent Neural Network unravelled

2

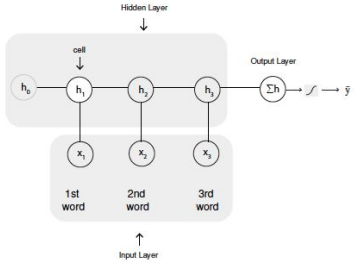


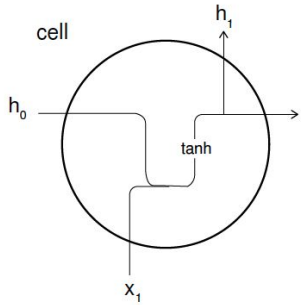
3



Recurrent Neural Network unravelled (with dimensions collapsed)

4

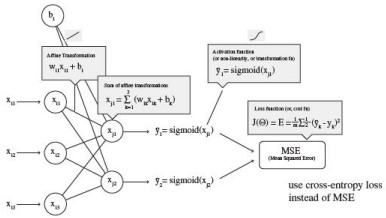




A Single Layer Nueral Network is just **Logistic Regression**

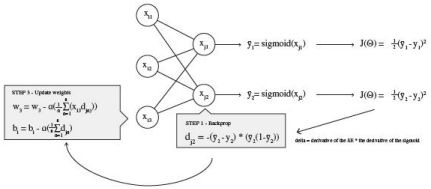
X: $[x_1, x_2, x_3]$ -> 3 dimensional input vector
W: $[w_1, w_2, w_3]$ -> 3 dimensional weight vector
b: $[b_1, b_2, b_3]$ -> 3 dimensional bias vector
Ŷ: $[\hat{y}_1, \hat{y}_2]$ -> 2 dimensional output vector "Y hat" - (predictions for 2 class logistic regression)
Y: $[y_1, y_2] \in \{0,1\}$ -> 2 dimensional target vector (training labels)

FORWARD PASS

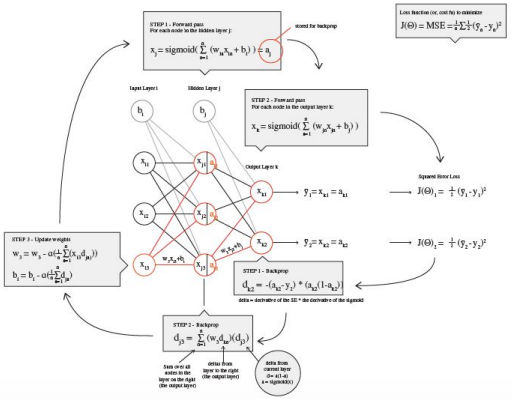


use cross-entropy loss instead of MSE

BACK PROPAGATION
Gradient Descent to minimize the loss (ERROR)



Neural Network with 1 hidden layer (for simplicity, without regularization)
Based on http://ufldl.stanford.edu/wiki/index.php/Backpropagation_Algorithm



Reference **http://ufldl.stanford.edu/wiki/index.php/Backpropagation_Algorithm**

$$\frac{\partial}{\partial W_{ij}^{(l)}} J(W,b) = \left[\frac{1}{m}\sum_{i=1}^{m}\frac{\partial}{\partial W_{ij}^{(l)}} J(W,b,x^{(i)},y^{(i)})\right] + \lambda W_{ij}^{(l)}$$
$$\frac{\partial}{\partial b_i^{(l)}} J(W,b) = \frac{1}{m}\sum_{i=1}^{m}\frac{\partial}{\partial b_i^{(l)}} J(W,b,x^{(i)},y^{(i)})$$

For an output node, we can directly measure the difference between the network's activation and the true target value, and use that to define \delta_i^{(n_l)} (where layer n_l is the output layer).
For each output unit i in layer n_l (the output layer), set

$$\delta_i^{(n_l)} = \frac{\partial}{\partial z_i^{(n_l)}} \frac{1}{2}\|y - h_{W,b}(x)\|^2 = -(y_i - a_i^{(n_l)}) \cdot f'(z_i^{(n_l)})$$

For l = n_l-1, n_l-2, n_l-3, \ldots, 2
For each node i in layer l, set

$$\delta_i^{(l)} = \left(\sum_{j=1}^{s_{l+1}} W_{ji}^{(l)} \delta_j^{(l+1)}\right) f'(z_i^{(l)})$$

Compute the desired partial derivatives, which are given as:

$$\frac{\partial}{\partial W_{ij}^{(l)}} J(W,b;x,y) = a_j^{(l)} \delta_i^{(l+1)}$$
$$\frac{\partial}{\partial b_i^{(l)}} J(W,b;x,y) = \delta_i^{(l+1)}.$$

f(z) is the sigmoid = a

f'(z) is the derivative of the sigmoid

$$f'(z_i^{(l)}) = a_i^{(l)}(1 - a_i^{(l)})$$

Update weigths

$$W_{ij}^{(l)} = W_{ij}^{(l)} - \alpha \frac{\partial}{\partial W_{ij}^{(l)}} J(W,b)$$
$$b_i^{(l)} = b_i^{(l)} - \alpha \frac{\partial}{\partial b_i^{(l)}} J(W,b)$$

$$\nabla_{W^{(l)}} J(W,b;x,y) = \delta^{(l+1)}(a^{(l)})^T,$$
$$\nabla_{b^{(l)}} J(W,b;x,y) = \delta^{(l+1)}.$$

James Kunz 2: https://github.com/datasci-w266/main/blob/master/assignment

James Kunz 2: argmax_theta L(D, theta)

James Kunz 2: sigmoid(x) = s(x)(1-s(x))

James Kunz 2: 1-tanh^2

Data

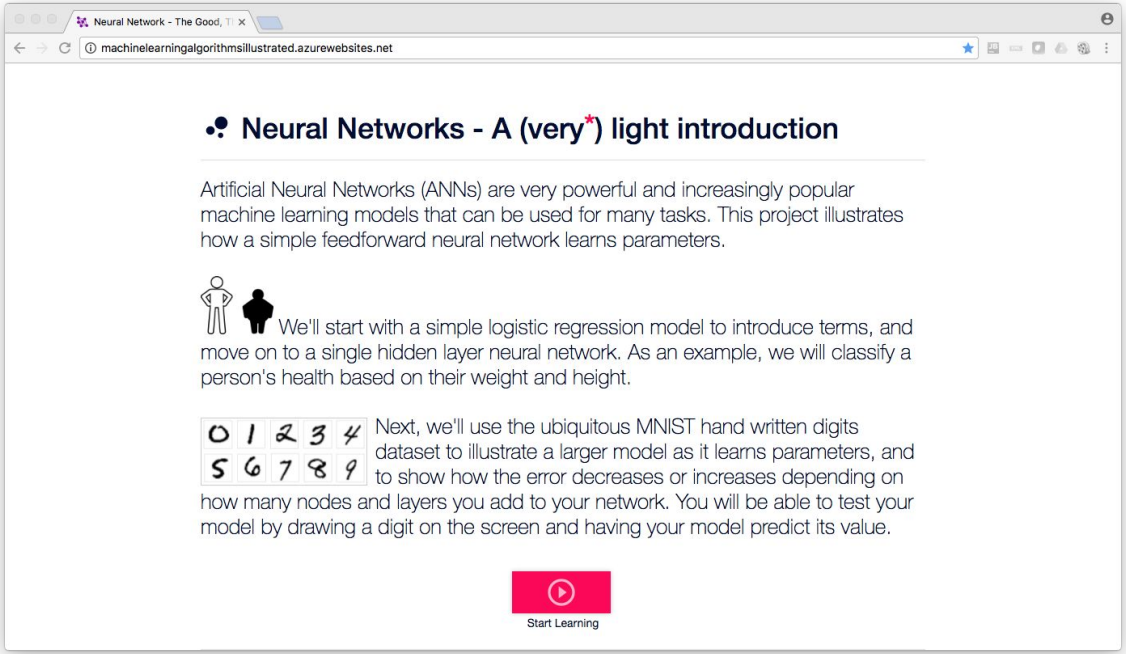
Learn page

We introduce concepts with a mere **6-8 data points** arbitrarily selected to illustrate a problem. We then use those data points to **pre-train** two simple models and capture intermediate data such as weights and losses, for the chart animations. This data lives in the project github repository.

Play page

We generate our dataset by training a neural network model **on the fly** using the **MNIST dataset**. To be clear, the data we are visualizing are the intermediate results (weights and losses) of the network, and not the MNIST dataset itself.

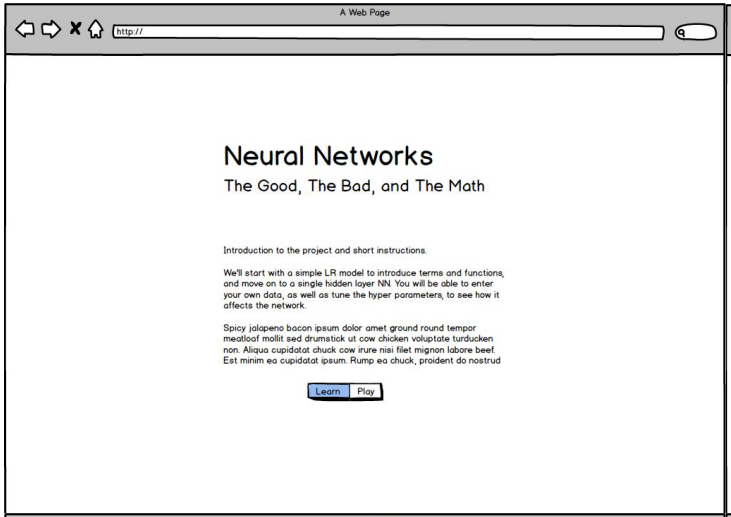
Demo

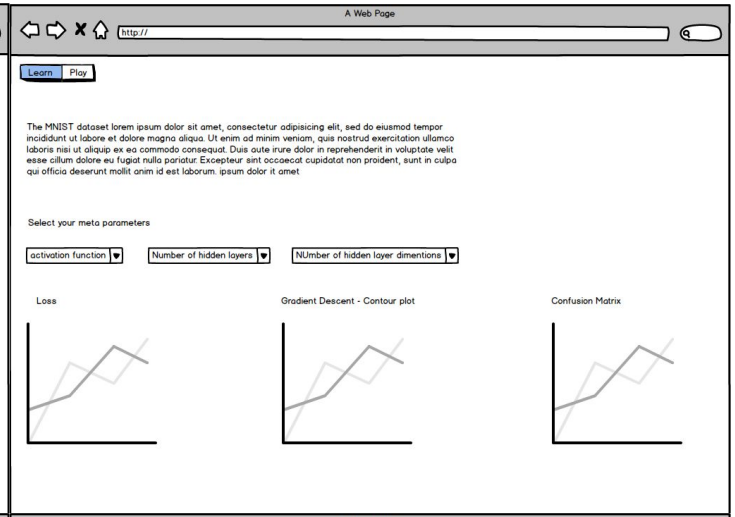


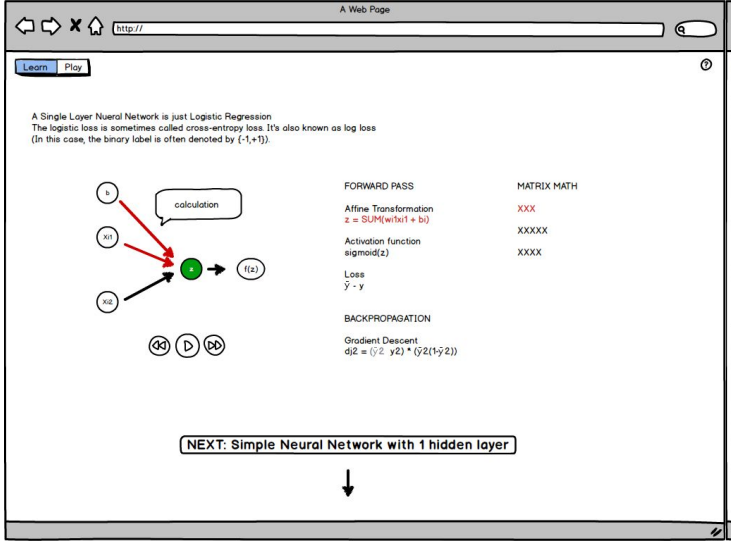
http://machinelearningalgorithmsillustrated.azurewebsites.net

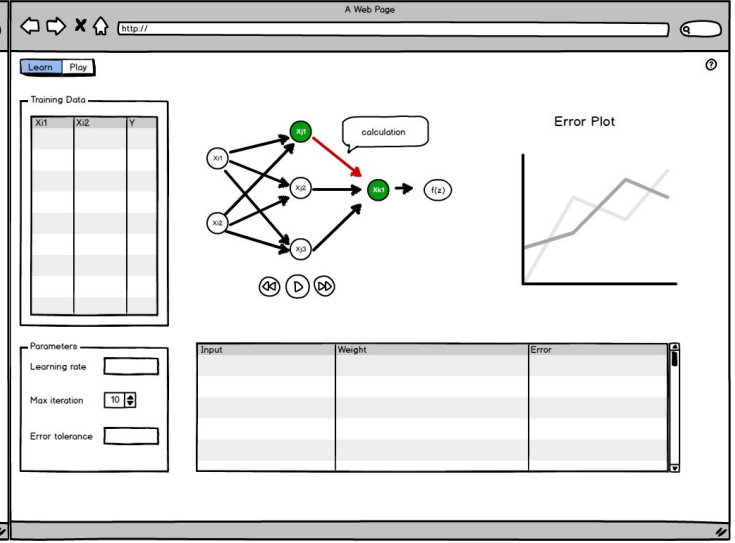
First steps to simplification

Original Balsamiq Mockups









Iterations of the Learn view

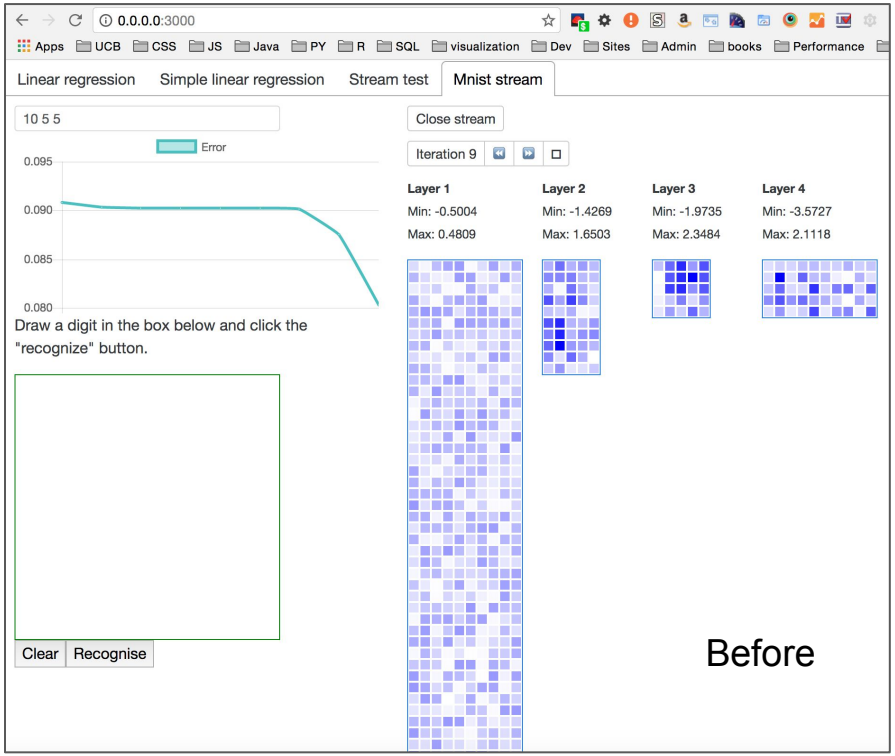
Play

Learn

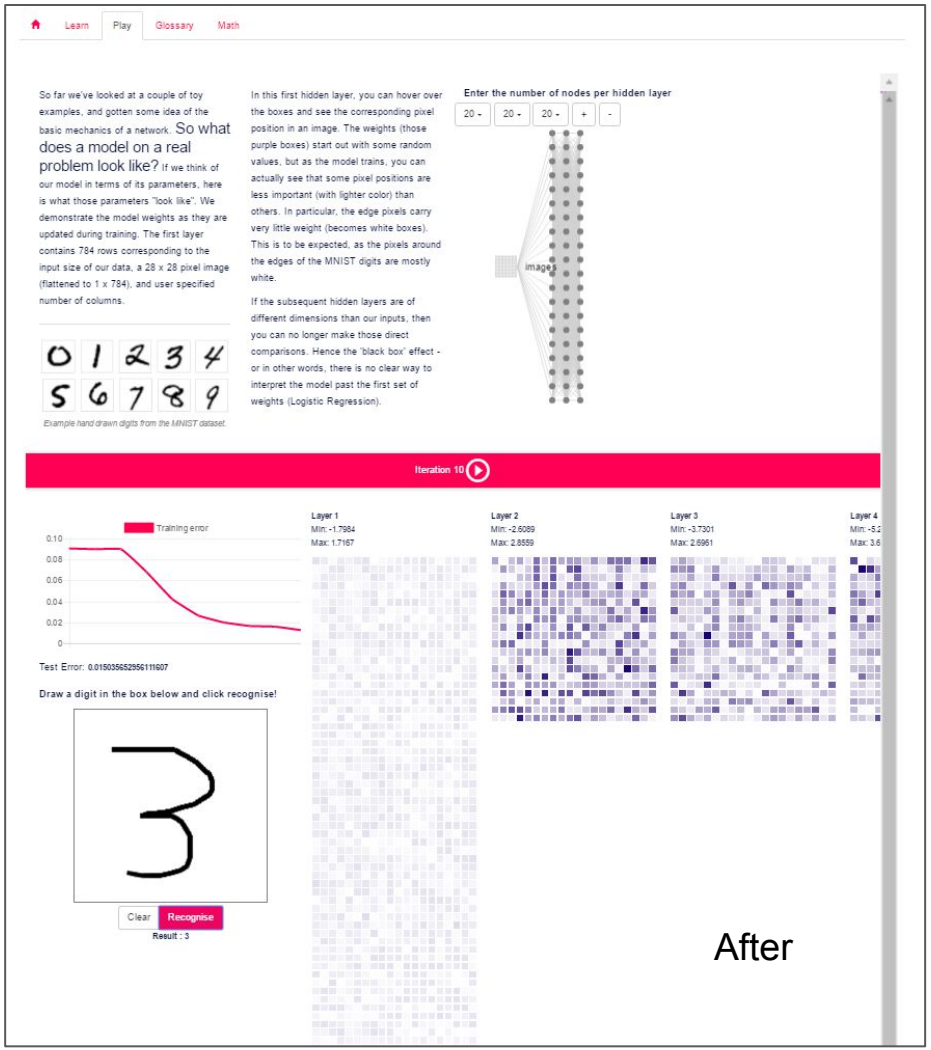
On the first iteration of the project it was difficult to make the segue from the Learn view to the Play view. An important detail that made sense of this transition, was the addition of the weights boxes animations on the Learn view which are reflected in the Play view, and the addition of the network graph illustration on the Play view which is reflected in the Learn view.

Iterations of the Play view

Consolidating colors and style



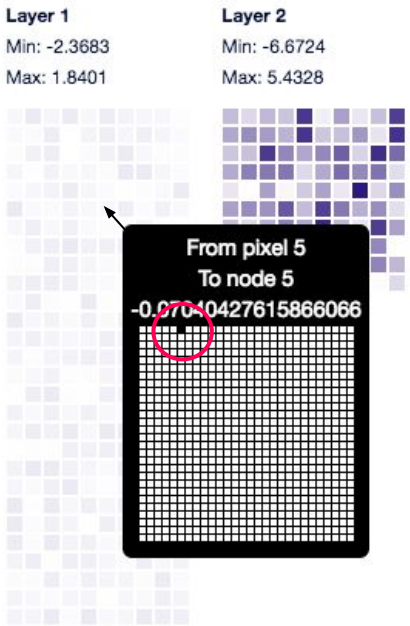
Before

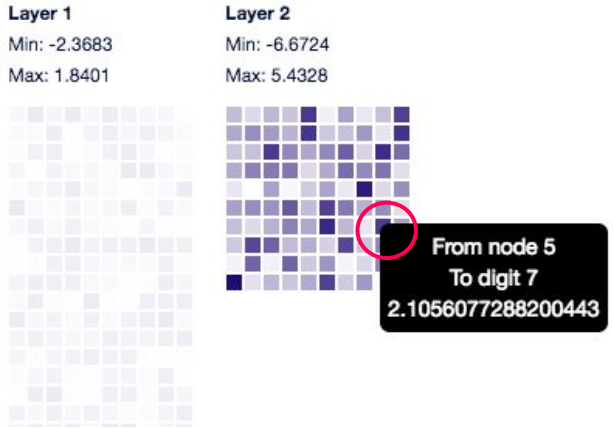


After

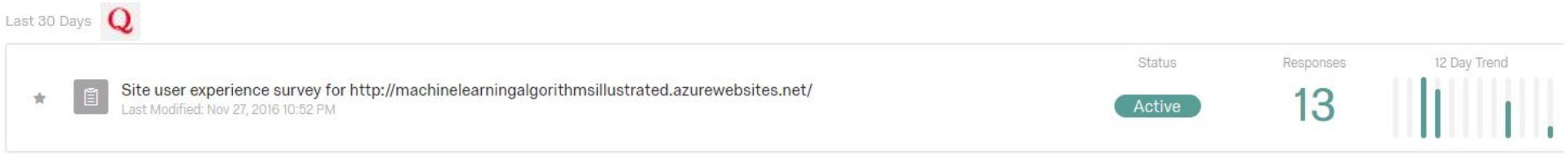
Insights from the Play view

The color of each box and its position in the data matrix allows the user to clearly see which pixels are more important than others. Important meaning weight absolute value is much greater than 0.





Major takeaways from usability study



Most users **skipped** reading the texts.

Users were **confused** about which buttons to click.

Users were confused about **math terms**, such as LR, NN.

Users didn't understand the significance of 'linear separability" and "decision boundaries"

User cannot make **connections** between charts, such as weight line and weight box are the same thing

Consolidated Usability Test Results

Must

- Add clear labels to charts (Done!)
- Use text format (font and color) to highlight key terms (Done!)
- Provide clear instructions or visual indicators to guide user through intended actions (Done!)
- Improve animation responsiveness (Done! can be further improved)

Should

- Improve description and instructions so that users can easily understand what they are reading (Done!)
- In "Play" page, provide example and brief introduction of MNIST dataset (Done!)

Consolidated Usability Test Results (continued)

Could

- Add another section to show how a linear model fails when applied to nonlinear separable data.
- Create animation of backpropagation in NN learning

Won't

- It is very difficult to explain all the mathematical details behind neural network learning. It does not seem to be helpful to add more math.

Q1 - Thanks for taking the time to answer this survey. It will take you about 15 minutes for this survey.
Section 1, prerequisites(Answer before visiting site) What's your math level in terms of machine learning?Site url: http://machinelearningalgorithmsillustrated.azurewebsites.net/

#	Answer	%	Count
1	Not much.	37.50%	3
2	Understand simple logistic regression, and gradient descent.	12.50%	1
3	Understand simple neural network and knows about layers and weights.	25.00%	2
4	Understand different activation function and loss function.	25.00%	2
	Total	100%	8

Q19 - Did this project help you get some insight into the mechanics of NNs?

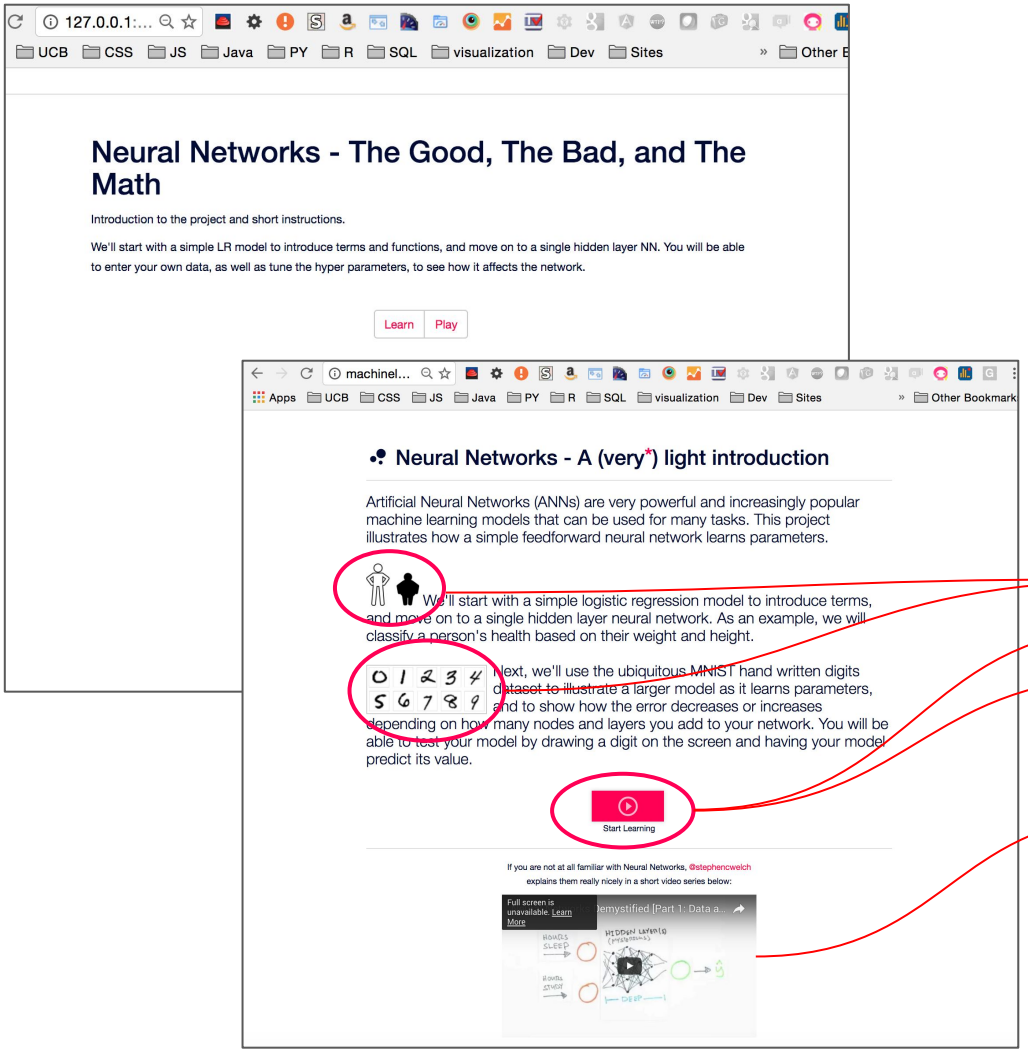
#	Answer	%	Count
1	Yes	50.00%	4
2	Maybe	37.50%	3
3	No	12.50%	1
	Total	100%	8

Q20 - Was there anything particularly frustrating or unclear?

#	Answer	%	Count
1	Yes, home page	12.50%	1
2	Yes, play page	25.00%	2
4	Yes, learn page	25.00%	2
5	Yes, TL;DR page	12.50%	1
3	No	25.00%	2
	Total	100%	8

Q21 - What did you like best?

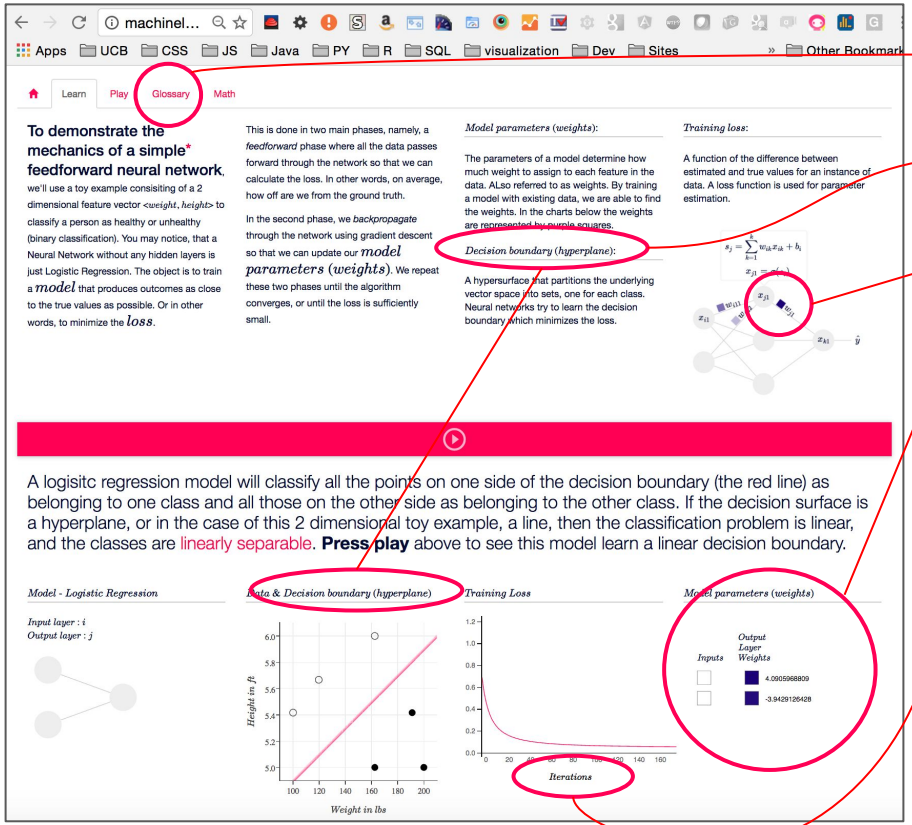
#	Answer	%	Count
1	Home	12.50%	1
2	Learn	37.50%	3
3	Play	37.50%	3
4	TL;DR	12.50%	1
	Total	100%	8



Changes to the home page after usability testing

- Change the title to reflect revised project scope
- Add explanatory text. Short and sweet in big letters
- Add icons to break up the text.
- Remove choice of buttons on home page
- Make the button on the homepage the same as the play buttons in Learn and Play, so that the user can learn its meaning naturally.
- Move the video link to the home so it doesn't distract on the Learn view.

Changes to the Learn view after usability testing



- Add a Glossary tab
- Remove redundant back/forward buttons
- Provide definitions of each term as it relates to the charts. The terms mirror the chart labels.
- Provide illustration showing exactly where in the network the weights play a role. Further enforced with shape and color (purple boxes). Also included on the “Math” view.
- Add large text explanations of decision boundaries and linear separability
- Add scrolling to top of page upon Play button click to bring the relevant animation into the fore.
- Add appropriate graph labels

Tools & technology choices

- HTML/CSS/JS (jQuery)
- D3
- Plotly.js
- Node.js
- React
- Brain.js

- Balsamiq for storyboards
- Photoshop/Illustrator for graphics and comps

- Azure App Services as website hosting solution(continuous deployment)
- AWS VM for compute intensive tasks such as training NN models on the fly
- Github for source control
- IPython notebook for training models offline

- Berkeley survey service as user survey solution (https://berkeley.qualtrics.com)

*Tools explored as options that weren't used in the project: Flask, TensorFlow, Docker container cluster, Conrec.js

Questions?

Credits

MIDS W209 Fall 2016

Kyle Hamilton @kyleiwaniec

Walter Erquinigo @a20012251

Jun Luo @junluo-code

Linfang Yang @yanglinfang

References

- http://www.deeplearningbook.org/contents/mlp.html
- http://ufldl.stanford.edu/wiki/index.php/Neural_Networks
- https://web.stanford.edu/group/pdplab/pdphandbook/handbookch6.html
- https://www.quora.com/What-are-hyperparameters-in-machine-learning

Berkeley project site:
https://www.ischool.berkeley.edu/projects/2016/neural-networks-very-light-introduction

Machine learning algorithms illustrated site:
http://machinelearningalgorithmsillustrated.azurewebsites.net/

Project GitHub urls:
https://github.com/kyleiwaniec/ML_algorithms_illustrated
https://github.com/yanglinfang/machinelearningalgorithmsillustrated

User survey link:
https://berkeley.qualtrics.com/SE/?SID=SV_8odiD2hhiltQfvT

Housekeeping - 12 points - Is the final product of appropriate form? TODO: need to move site to your own berkeley site. **Done**
https://www.ischool.berkeley.edu/projects/2016/neural-networks-very-light-introduction

Intro slide -

12 points - Does it successfully solve a problem or inform the user? TODO: explain problem, explain how we solved it, dataset used **Done**

Usability test slide -

12 points - Is the explanatory text clear? TODO: talk about text in usability test **Done**

10 points - Can the visualization be understood without too much effort? TODO: talk about visual in usability test **Done**

10 points - Is it aesthetically pleasing? TODO: mention style change in usability test **Done**

10 points - Did you include results of usability testing? **Done**

Design iteration slides -

12 points - Did you show substantial iteration? TODO: show three iterations: initial design, then comp, then final product **Done**

Iteration 1 slide - plain math and simple comp

Iteration 2 slide - Balsamic comp

Iteration 3 slide - site with simple home page, learn page, play page and DL;DR

Iteration 4 (final) slide - current site

Novel ideas slide - 8 points - Did you present data in a novel way? TODO: mention about the purple box weight matrix **Done**

Tools slide - 8 points - Did you explain your choice of tools? TODO: include a slide about tools used **Done**

Teamwork slide - 6 points - Did everyone in your group participate? TODO: explain task breakdowns **Done**