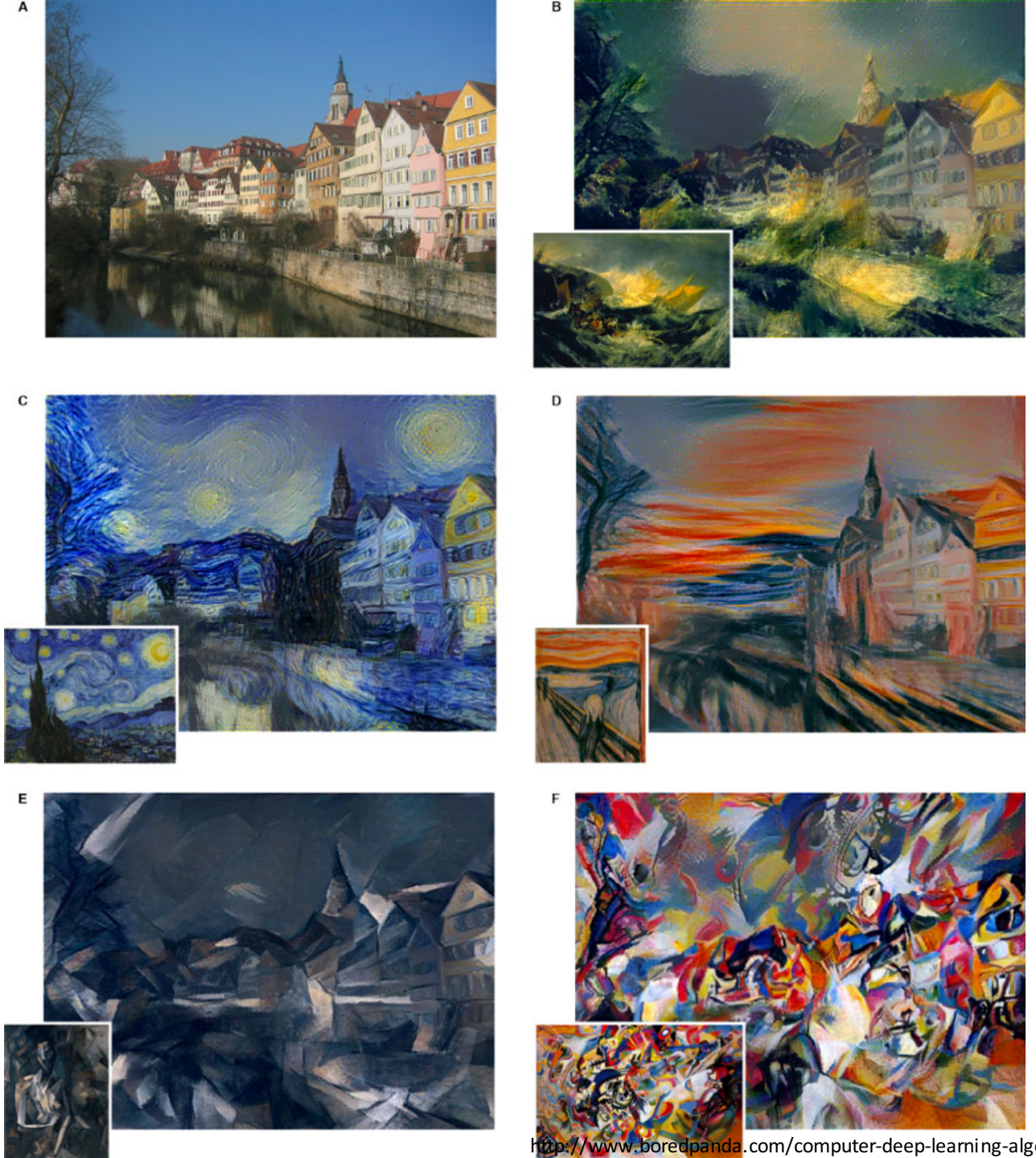
Deep Learning
Tutorial and Applications

Artit “Art” Wangperawong
artitw@gmail.com

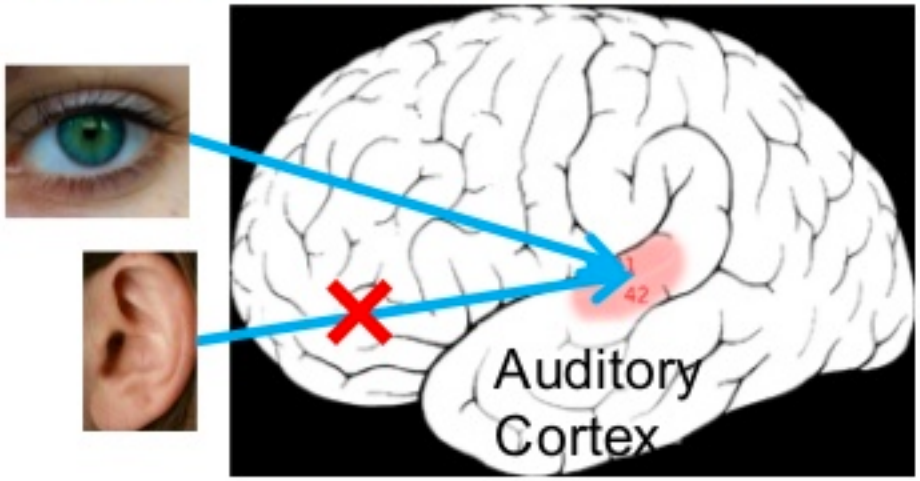
http://www.imore.com/siri

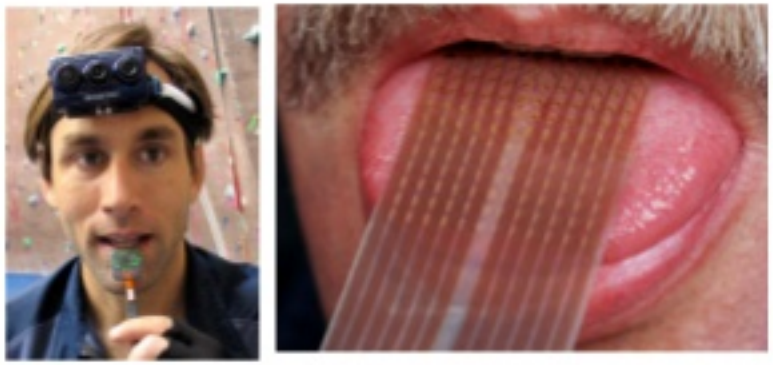


http://www.boredpanda.com/computer-deep-learning-algorithm-painting-masters/

http://www.techrepublic.com/article/can-googles-deepmind-beat-world-go-champ-watch-live-this-week/

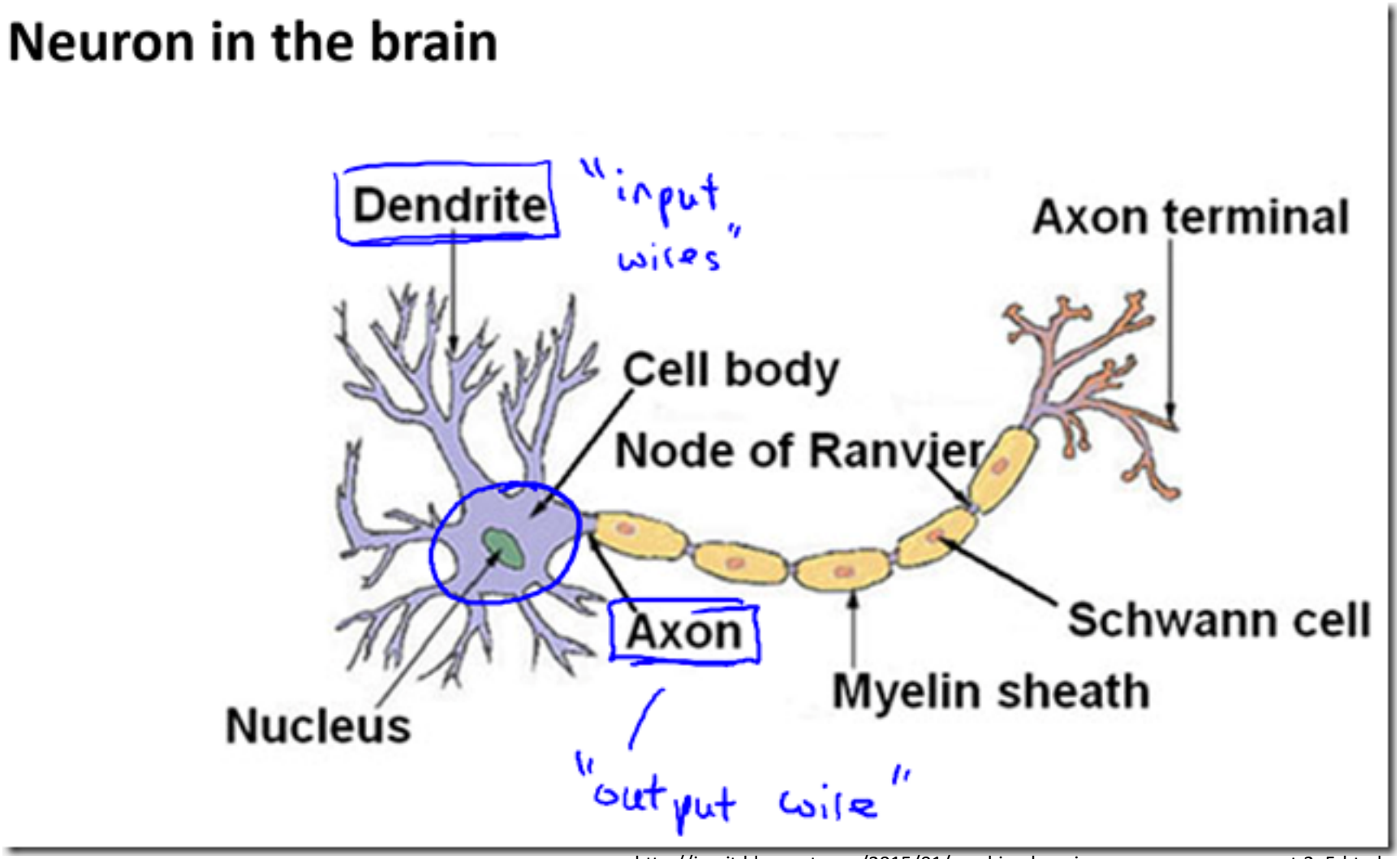
Sensor representation in the brain





[Roe et al., 1992; BrainPort; Welsh & Blasch, 1997]

Deep Learning – Simulating the Brain

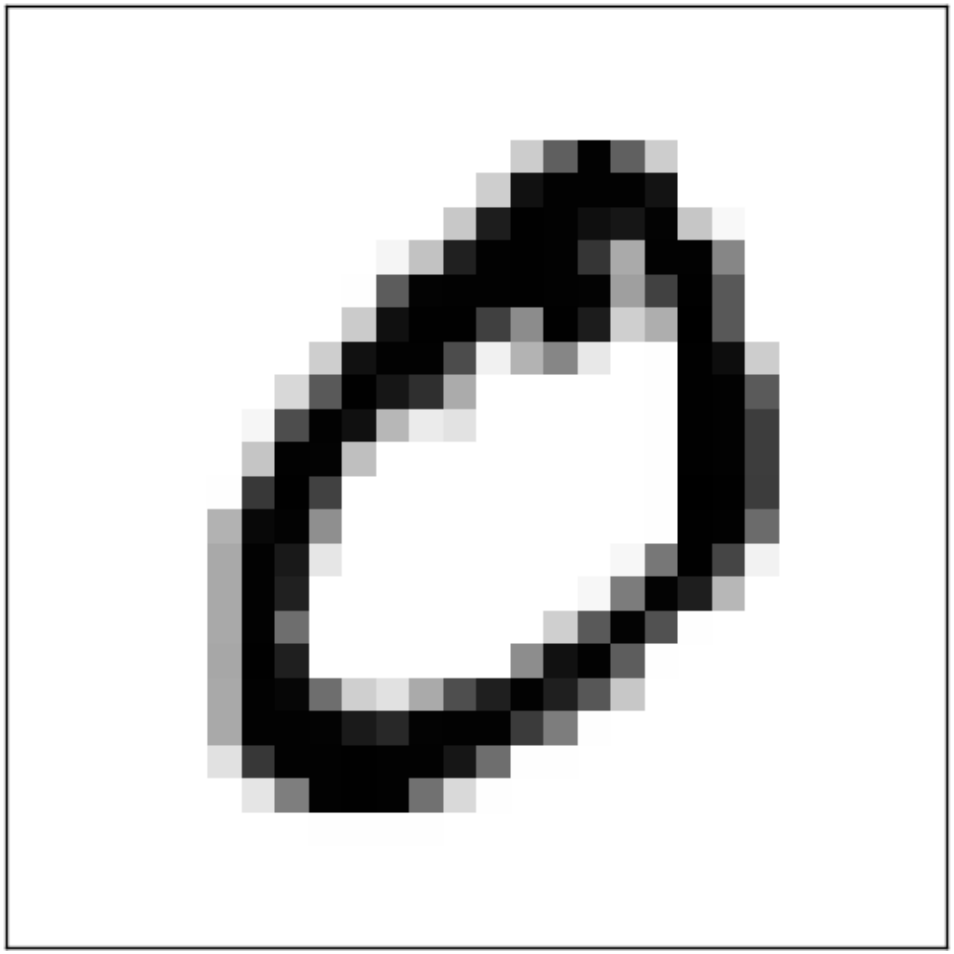


http://jomit.blogspot.com/2015/01/machine-learning-course-summary-part-2_5.html

Computer learns to read handwriting

neuralnetworksanddeeplearning.com

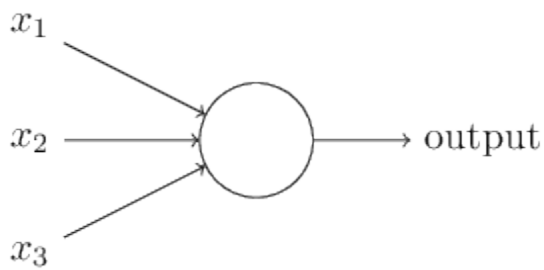
2D pixel intensity input data



28x28=784

neuralnetworksanddeeplearning.com

Input and output from a neuron



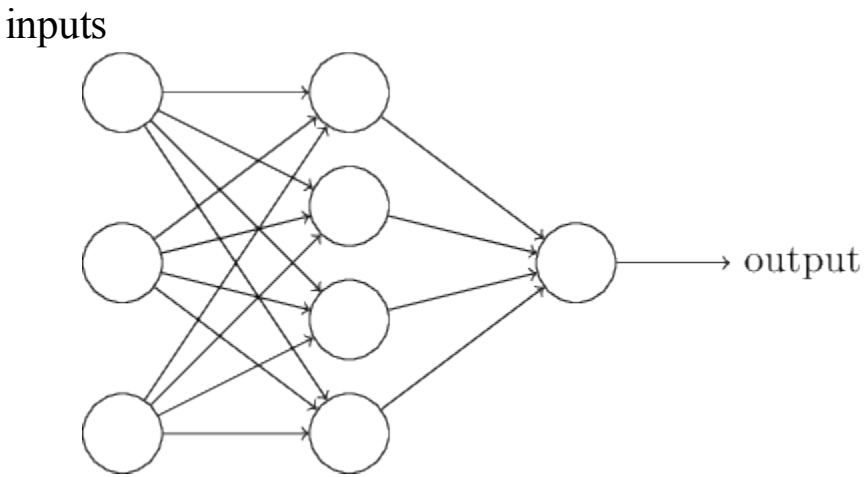
Learning algorithm #1:

$$\text{output} = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{ threshold} \\ 1 & \text{if } \sum_j w_j x_j > \text{ threshold} \end{cases} \tag{1}$$

$$\text{output} = \begin{cases} 0 & \text{if } w \cdot x + b \leq 0 \\ 1 & \text{if } w \cdot x + b > 0 \end{cases} \tag{2}$$

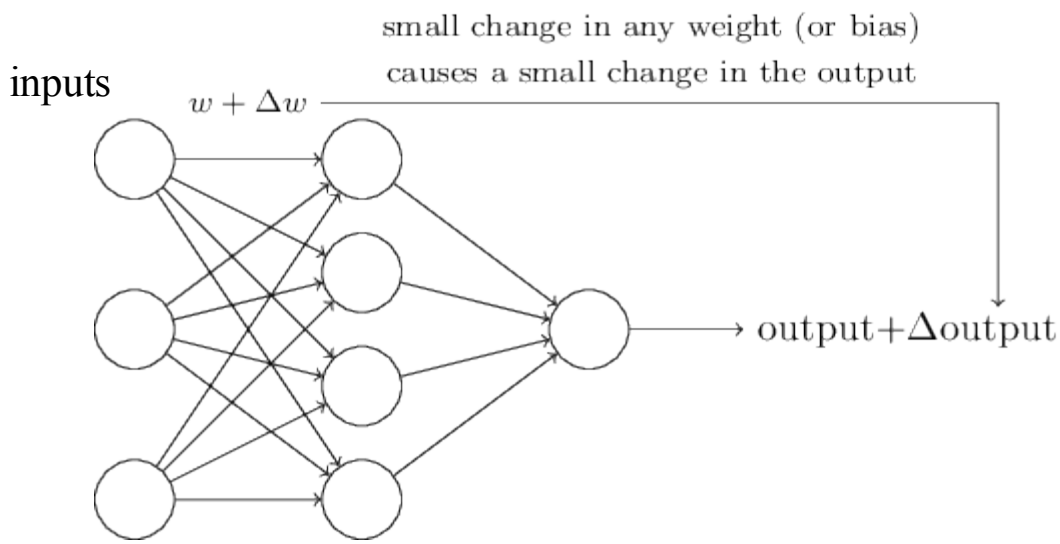
neuralnetworksanddeeplearning.com

Network of neurons = neural network



neuralnetworksanddeeplearning.com

Train a neural network – make it learn



neuralnetworksanddeeplearning.com

Learning algorithm #2:

But the output is not 0 or 1. Instead, it's $\sigma(w \cdot x + b)$, where σ is called the *sigmoid function**, and is defined by:

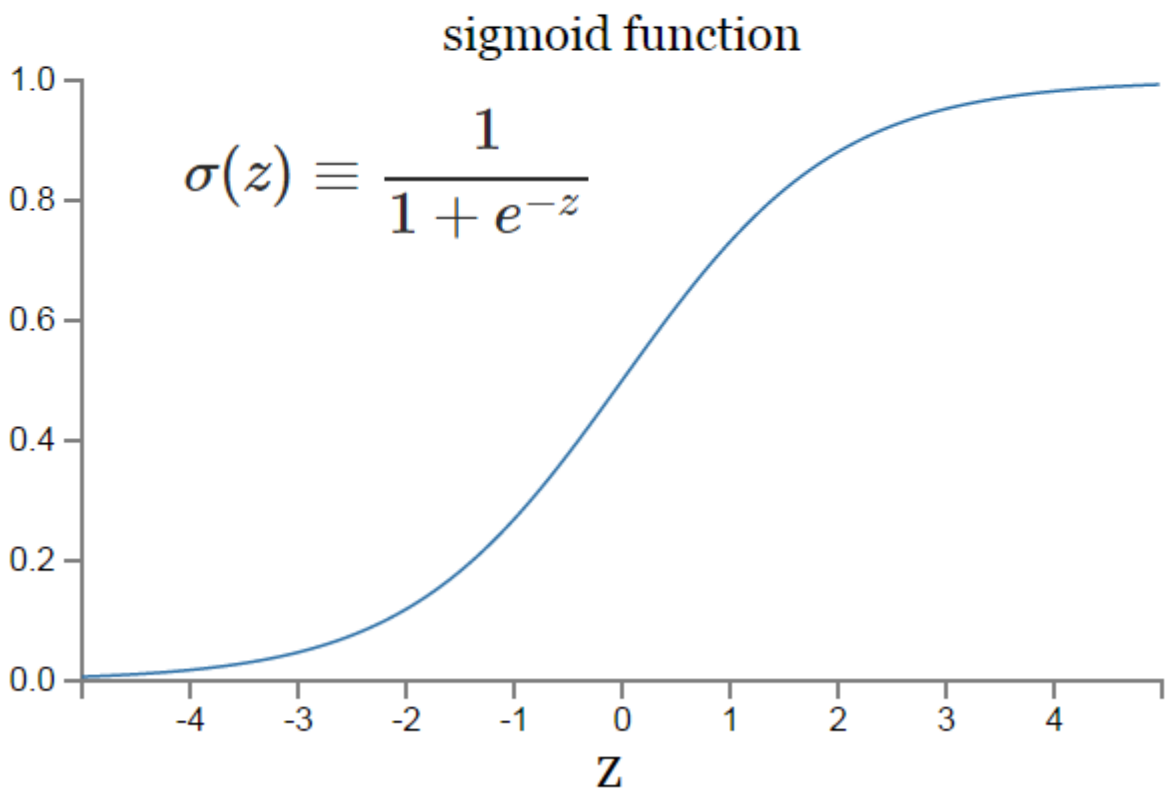
$$\sigma(z) \equiv \frac{1}{1+e^{-z}}. \tag{3}$$

To put it all a little more explicitly, the output of a sigmoid neuron with inputs $x_1, x_2, \ldots$, weights $w_1, w_2, \ldots$, and bias b is

$$\frac{1}{1+\exp(-\sum_j w_j x_j - b)}. \tag{4}$$

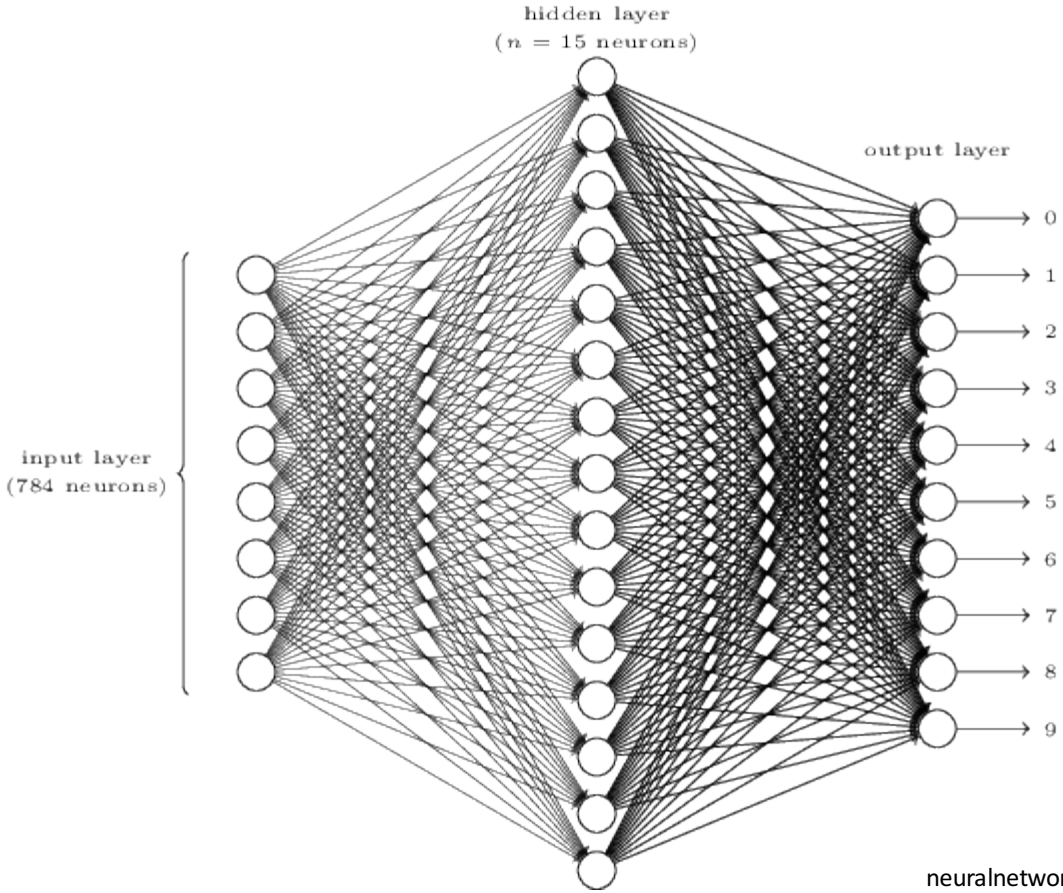
neuralnetworksanddeeplearning.com

Sigmoid Neurons



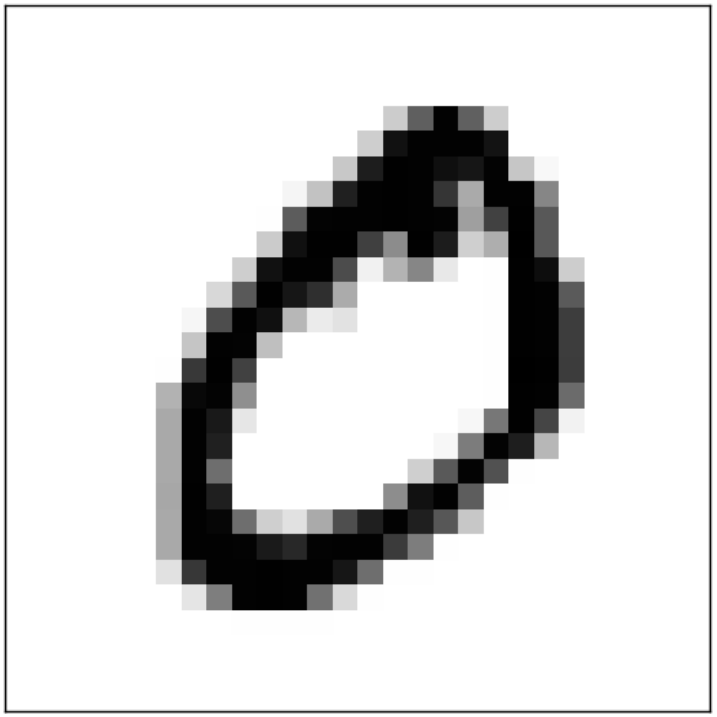
neuralnetworksanddeeplearning.com

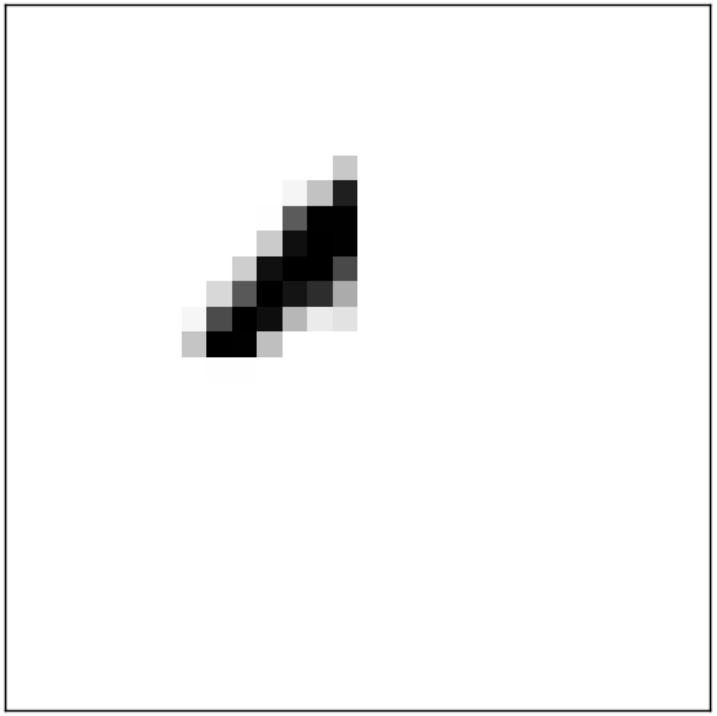
Simple neural network for OCR

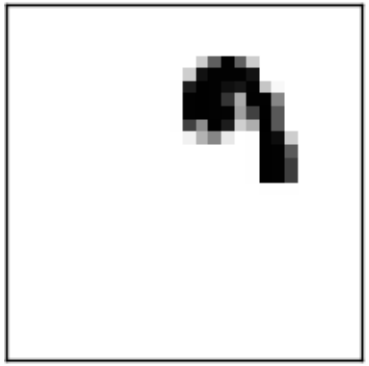


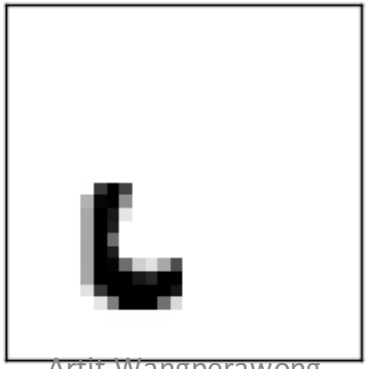
neuralnetworksanddeeplearning.com

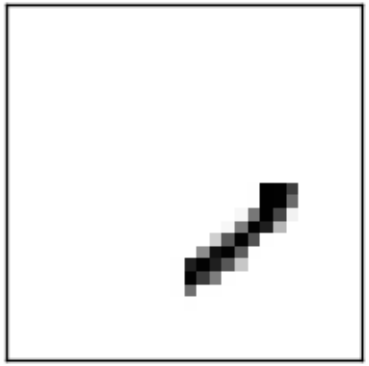
How does it work?





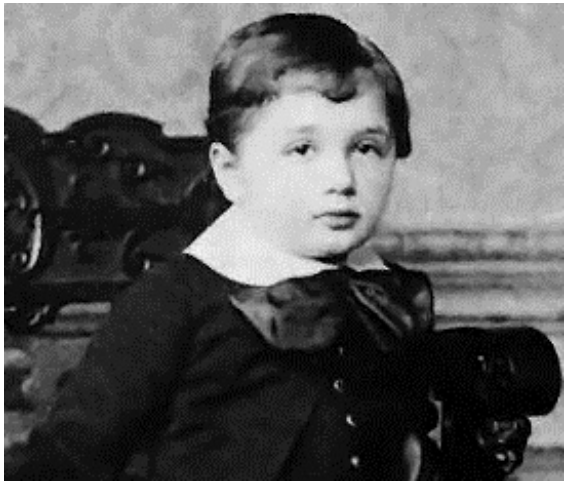


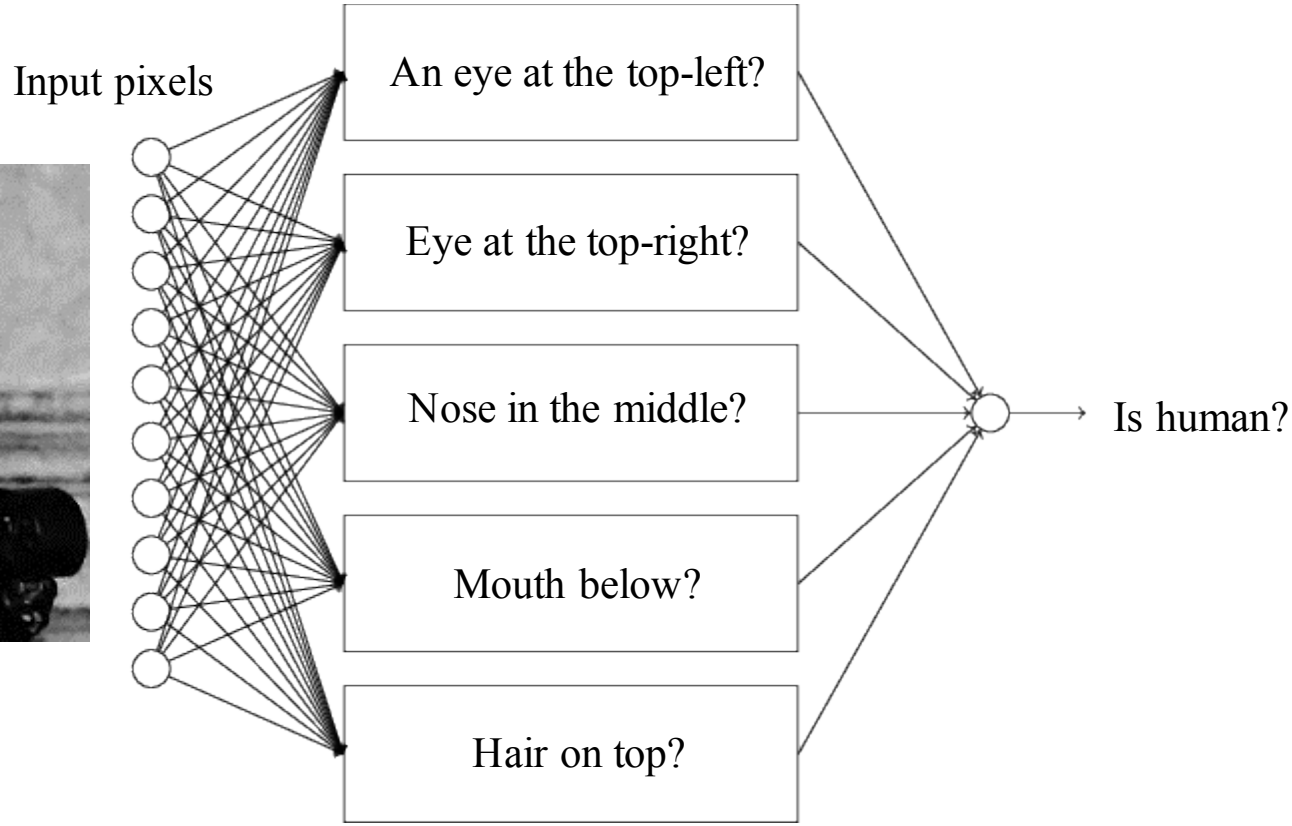




SO FAR NOT DEEP, STILL SHALLOW

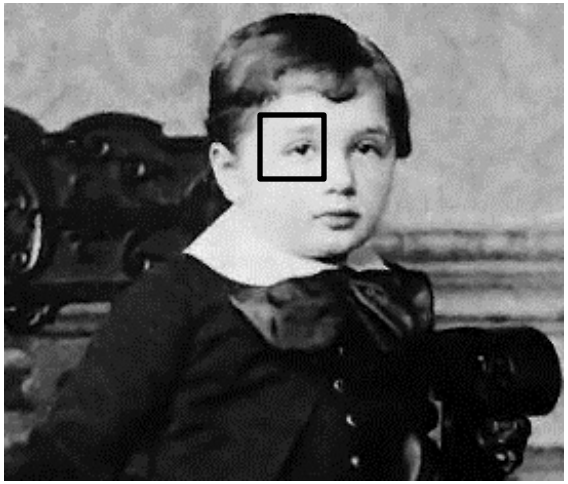
Hierarchical breakdown of features

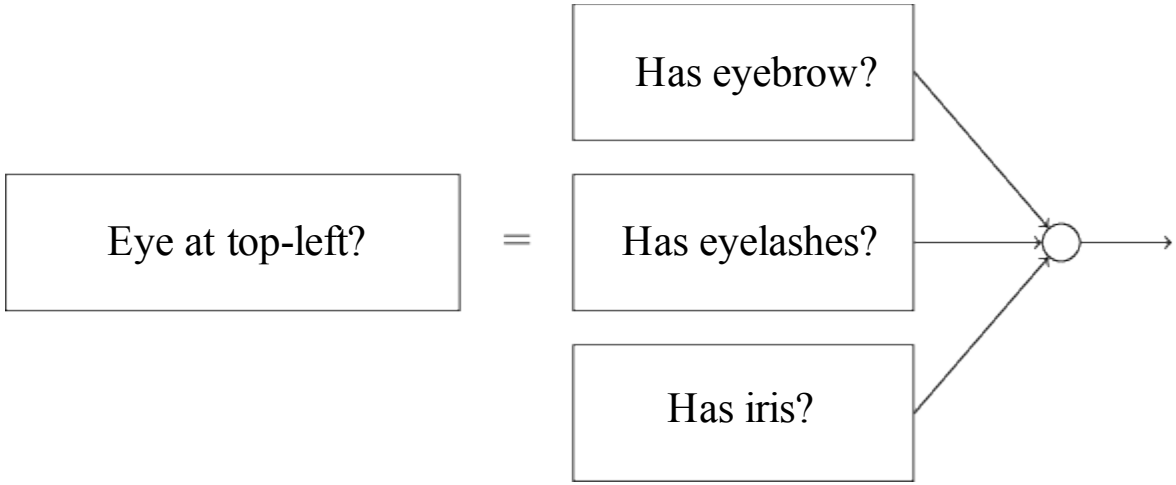




neuralnetworksanddeeplearning.com

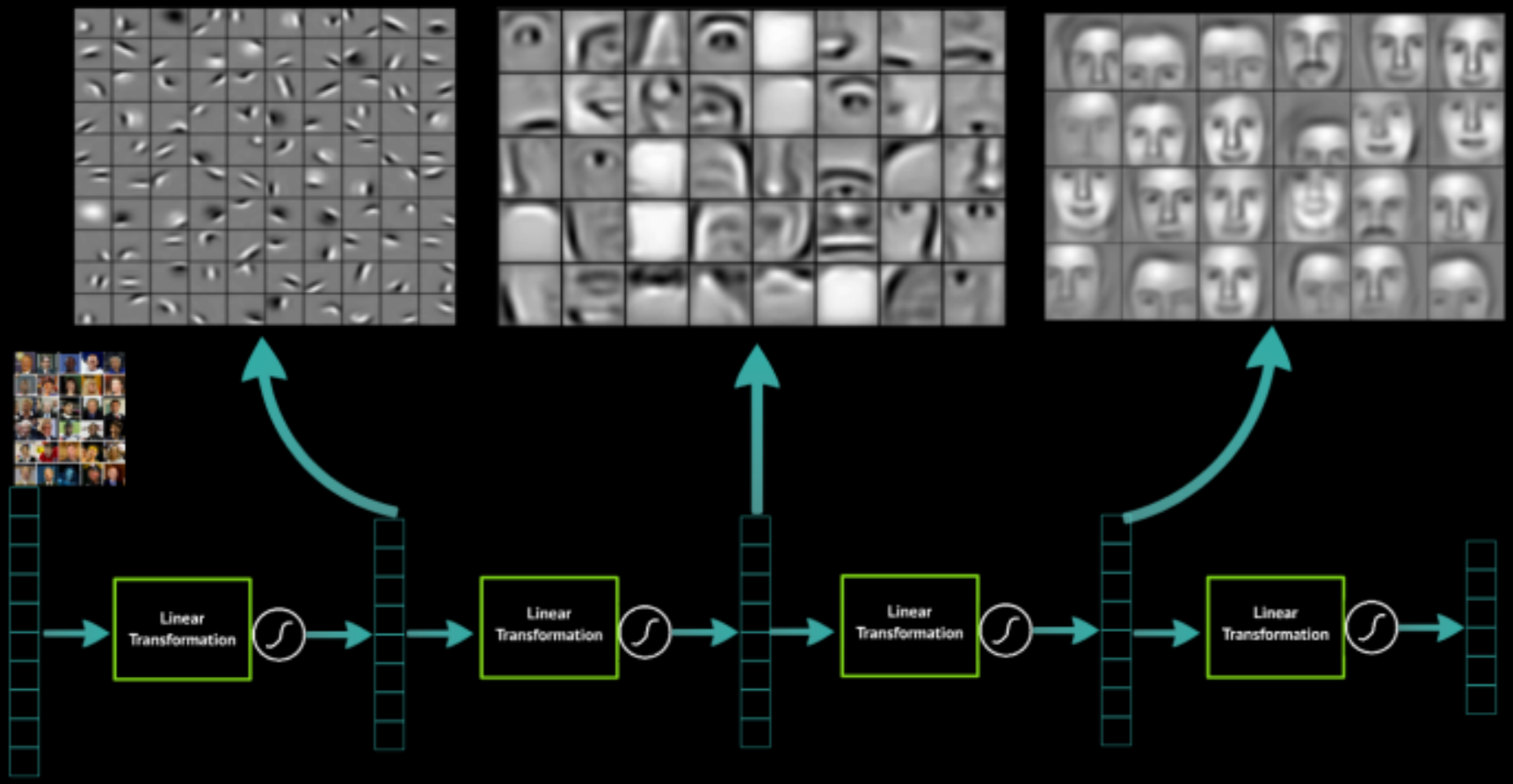
Hierarchical breakdown of features





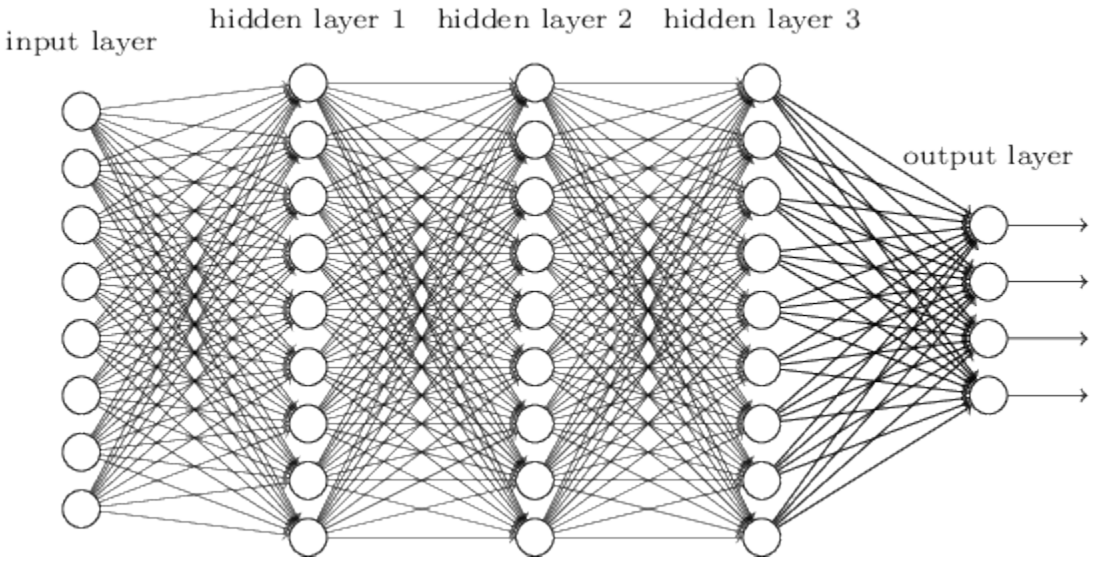
neuralnetworksanddeeplearning.com

Deep Learning learns layers of features



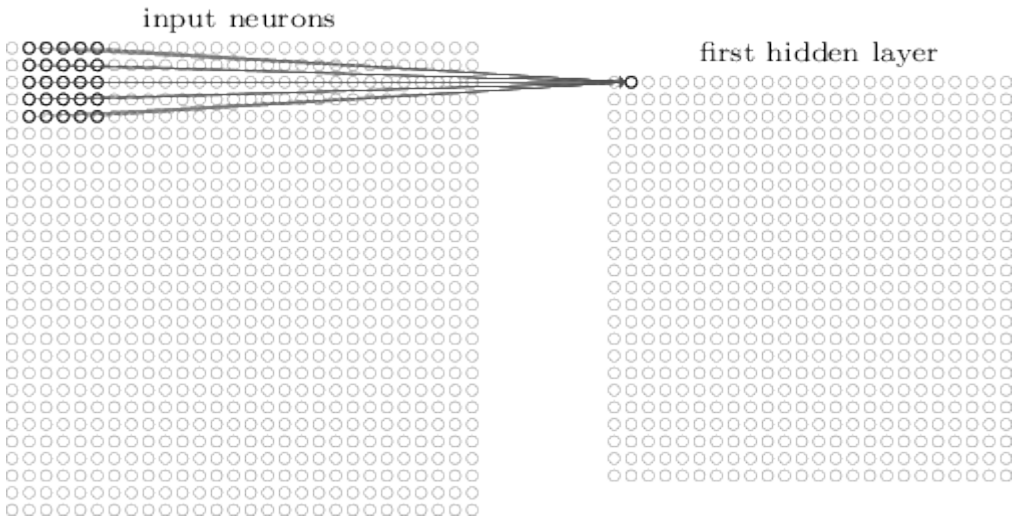
http://www.andrewng.org/

Deep neural network (DNN)



neuralnetworksanddeeplearning.com

Convolution – feature mapping



$1_{\times 1}$	$1_{\times 0}$	$1_{\times 1}$	0	0
$0_{\times 0}$	$1_{\times 1}$	$1_{\times 0}$	1	0
$0_{\times 1}$	$0_{\times 0}$	$1_{\times 1}$	1	1
0	0	1	1	0
0	1	1	0	0

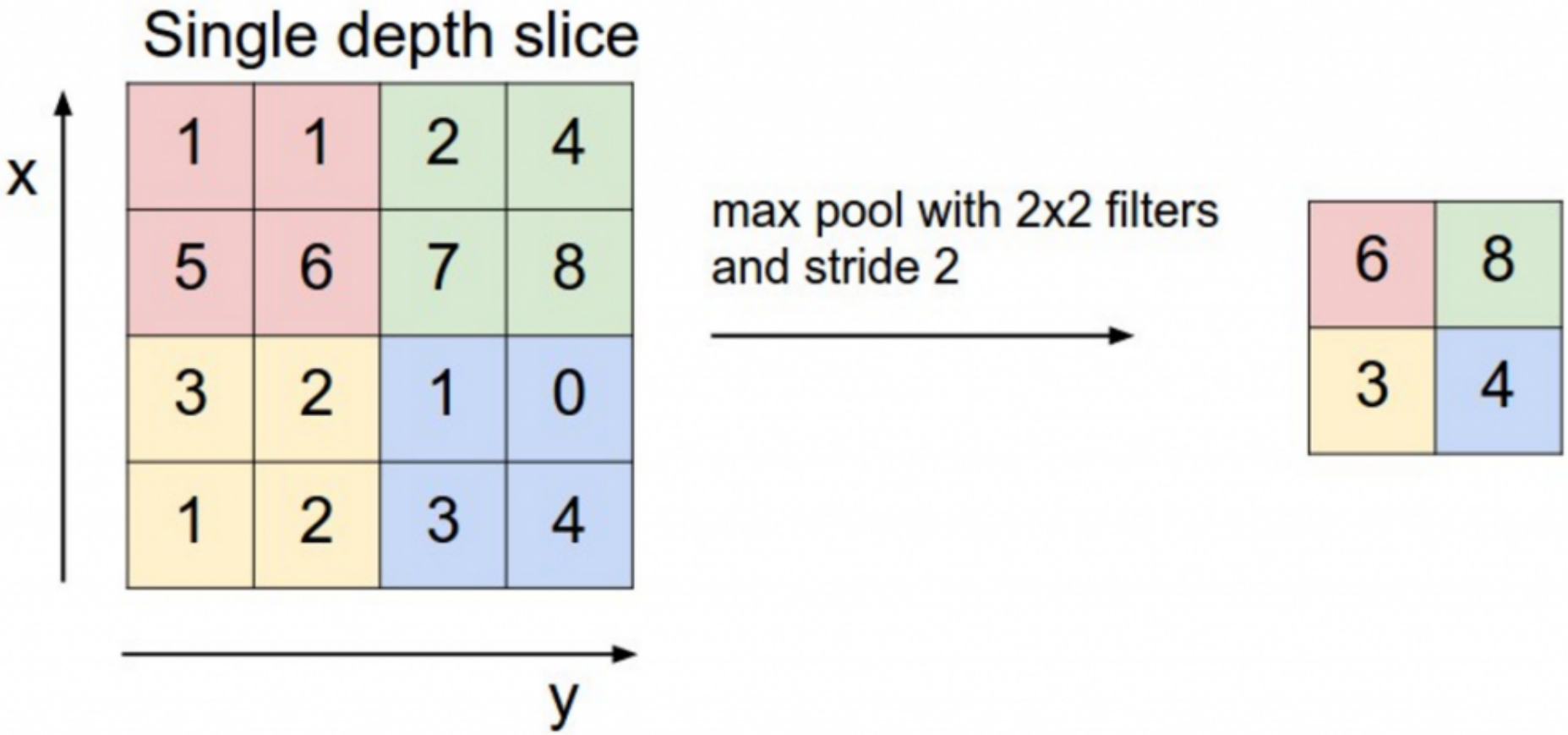
Image

4		

Convolved Feature

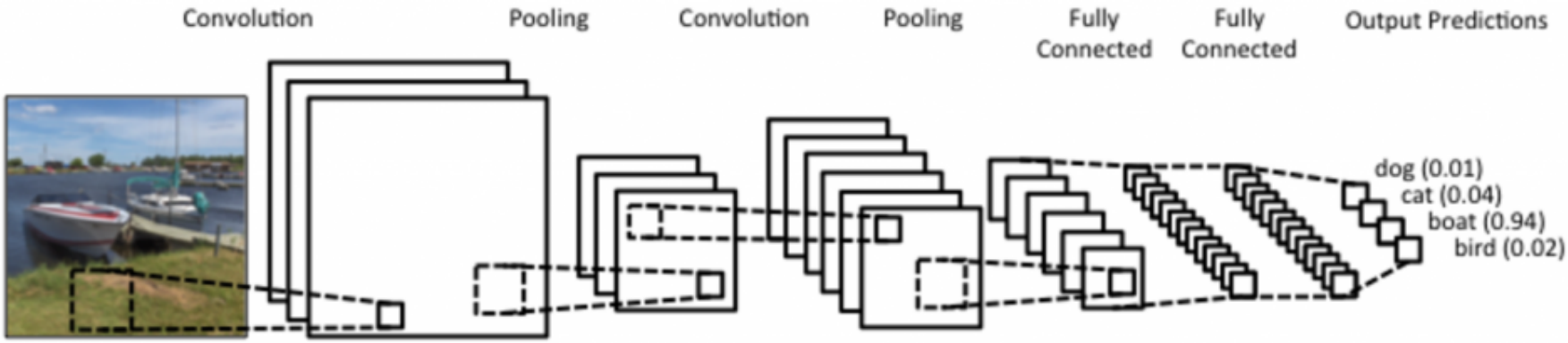
http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/

Pooling – aggregate and reduce



http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/

DNN + conv + pooling = Deep Learning



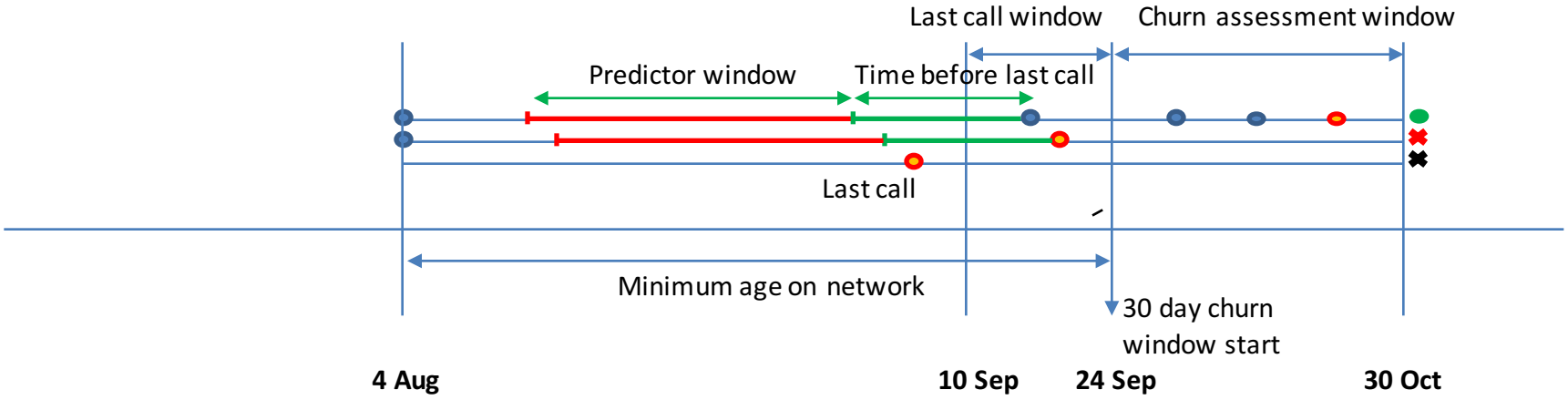
https://www.clarifai.com

How well does DL work?

MNIST	Simple NN	DNN
Accuracy	96.59	99.67

DL ON TELCO

Target Data Selection

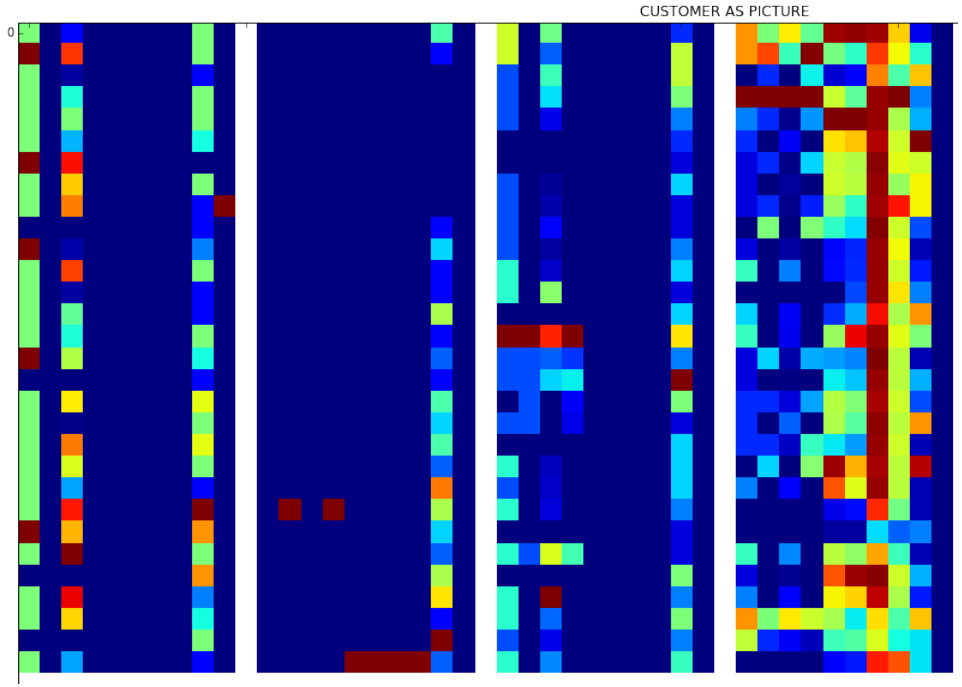


Parameter	Value	Implied date range
30 day churn window start	24 Sep 2015	24 Sep- 23 Oct
Last call window	14 days	10 Sep – 23 Sep
Time before last call	14 days	27 Aug – 23 Sep
Predictor window	30 days	28 Jul – 9 Sep
Minimum age on network	90 days	< 28 Jul

Telco churn DL – customer as image

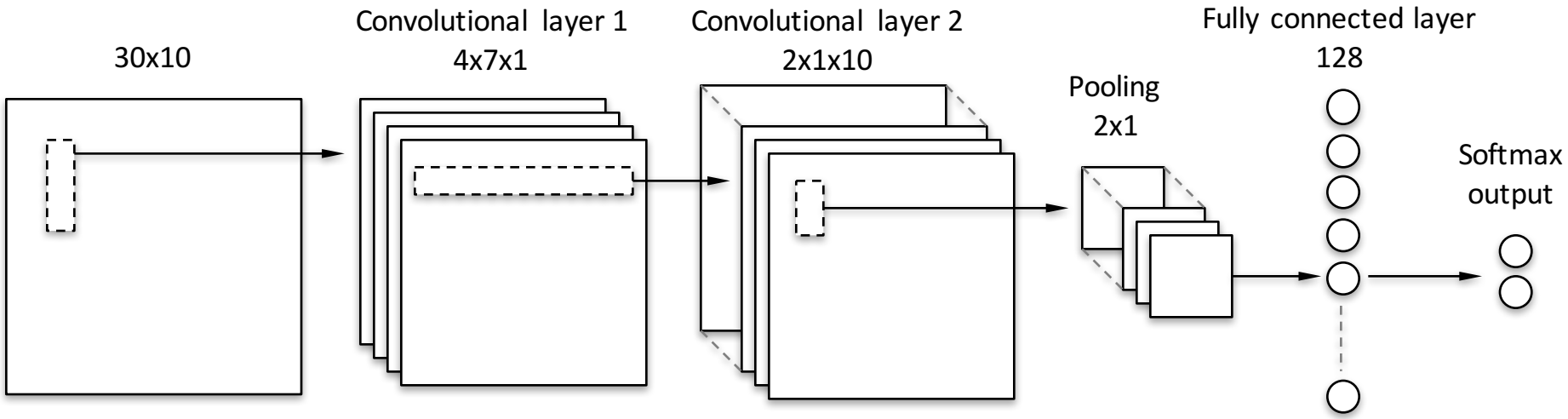
Label: 0 1 0 0

Row 1 = Day 1
Row 2 = Day 2
…
Row n = Day n



Data usage, SMS in, voice out, etc.

One architecture: DL-1



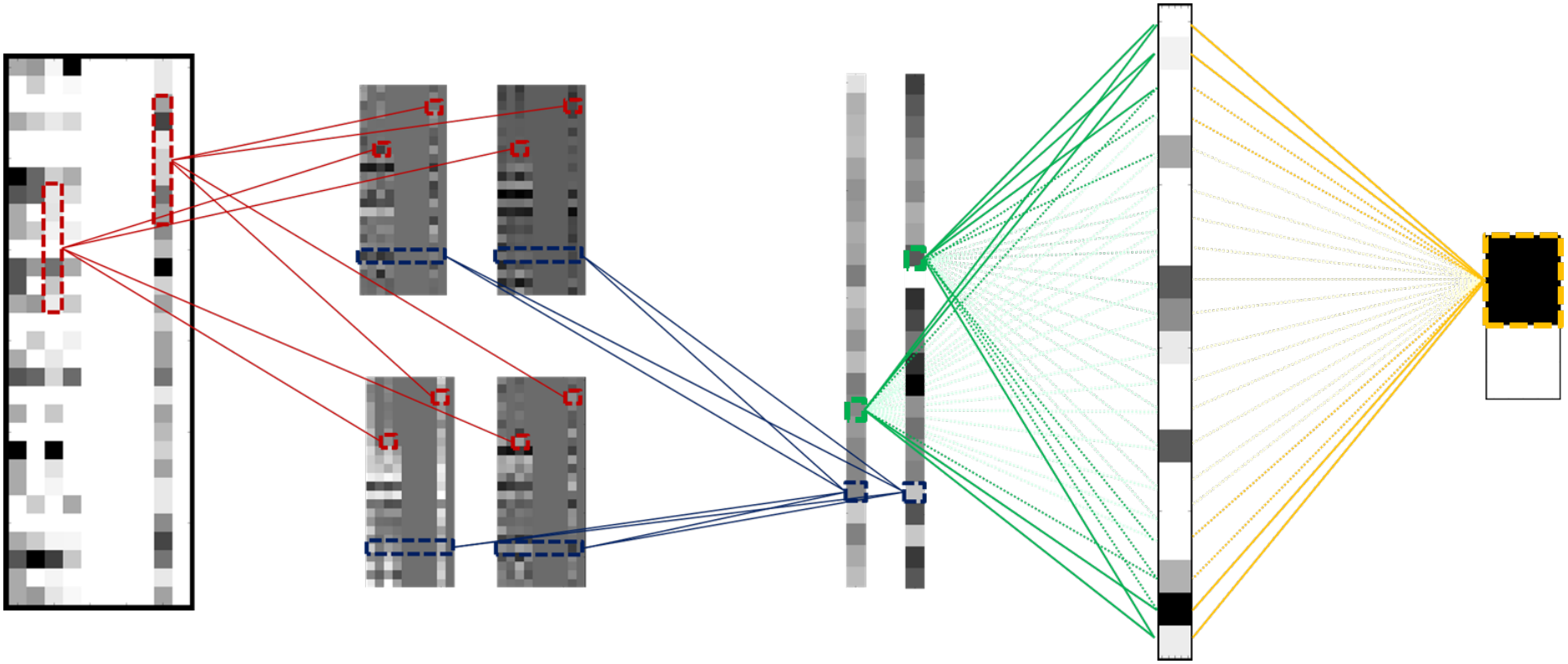
One architecture: DL-1

Convolution - 4 filters
7x1

Convolution - 2 filters
1x10

Fully
Connected

Fully
Connected



ReLU

ReLU

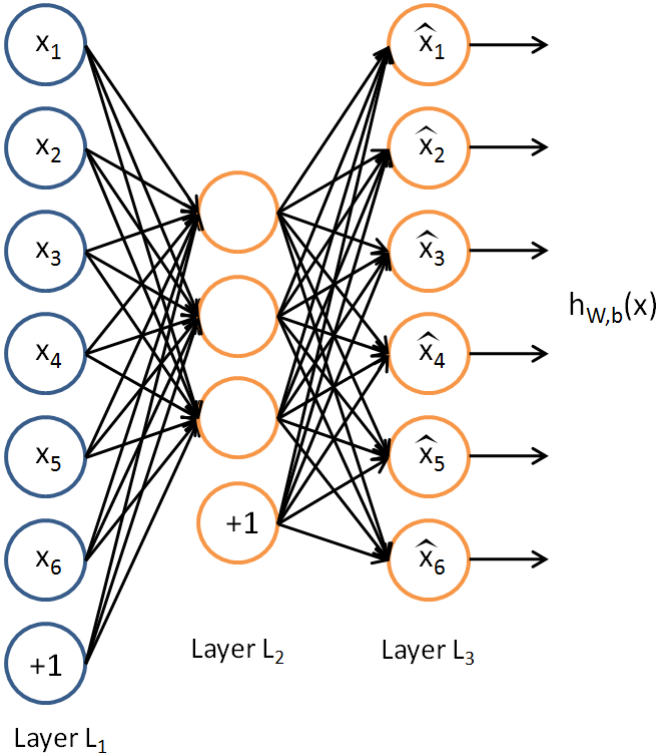
ReLU

SOFTMAX

DL versus SPSS

AUC	Training	Test
SPSS CHAID	0.699	0.665
DL-1	0.751	0.706
DL-2	0.748	0.743

Autoencoder – unsupervised learning

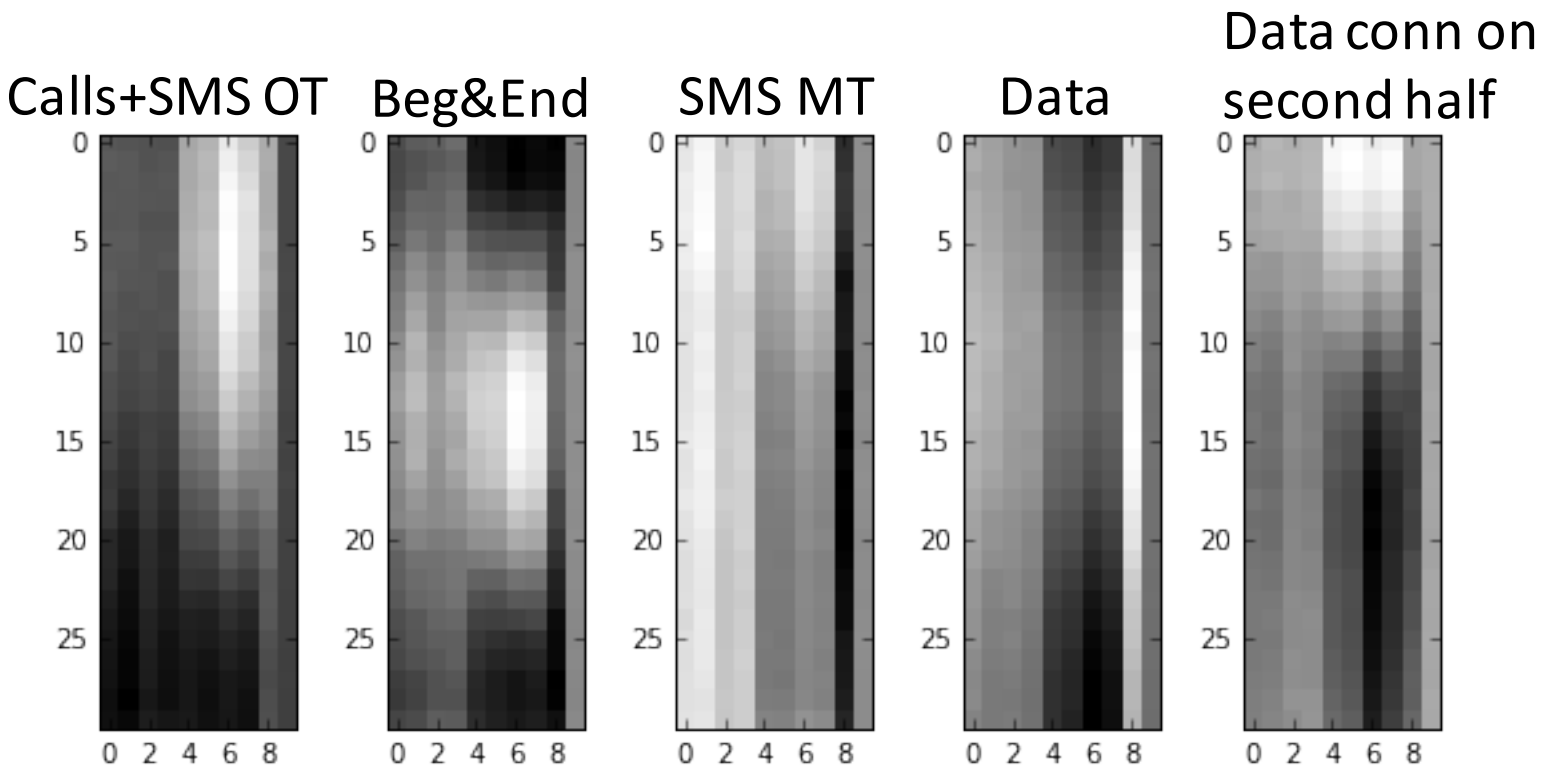


$$a_i^{(2)} = f\left(\sum_{j=1}^{100} W_{ij}^{(1)} x_j + b_i^{(1)}\right).$$

$$x_j = \frac{W_{ij}^{(1)}}{\sqrt{\sum_{j=1}^{100} (W_{ij}^{(1)})^2}}.$$

http://deeplearning.stanford.edu/wiki/index.php/Visualizing_a_Trained_Autoencoder

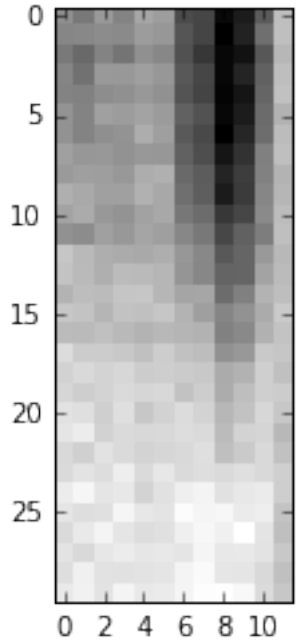
Images that maximally activate hidden units of autoencoder for all customers

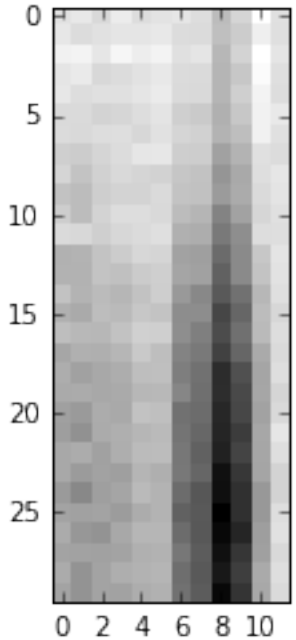


Images that maximally activate the weights of the autoencoder for churn

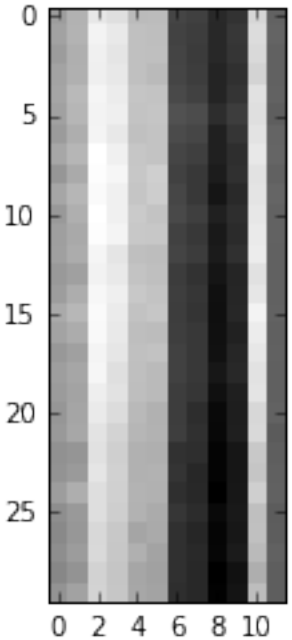
Strong data usage

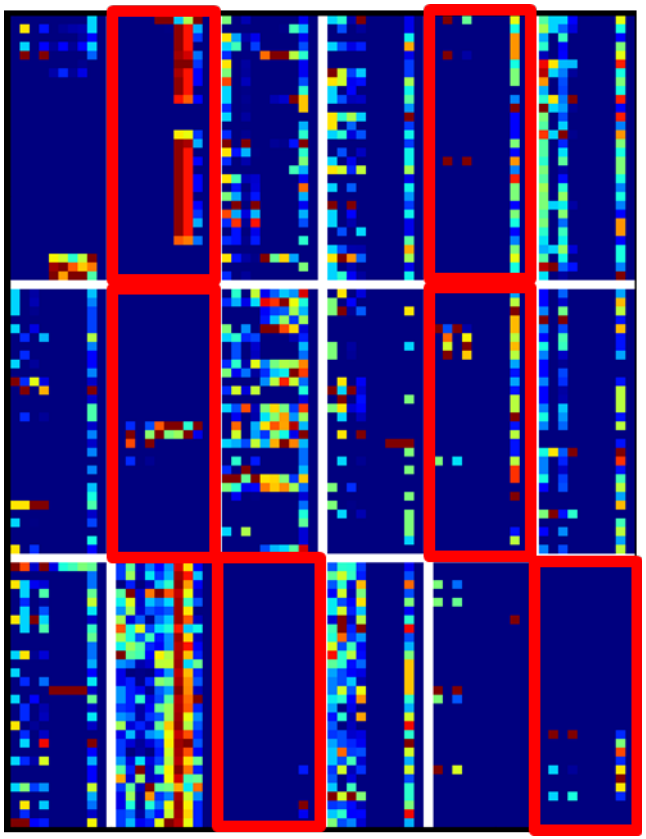
General usage dropoff

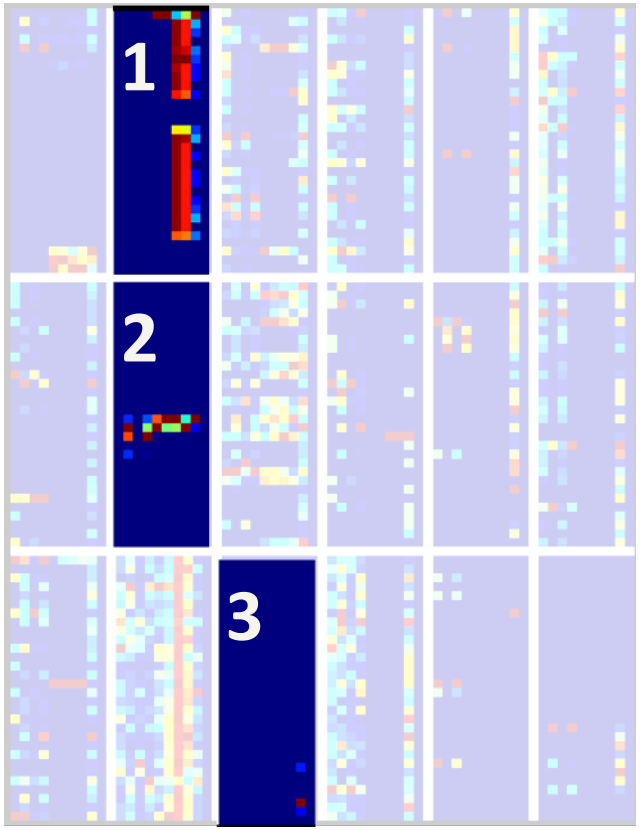




Pure data user







Customer who churned

1) Very high usage of data and only data
2) Usage of the phone only from 14-16th days
3) Almost no usage during the 30 days period

Churn analysis using deep convolutional neural networks and autoencoders

Artit Wangperawong[a,b,*]
Cyrille Brun[c]
Dr. Olav Laudy[c]
Rujikorn Pavasuthipaisit[a]

http://arxiv.org/abs/1604.05377

[a] True Corporation Public Company Limited, Bangkok 10310, Thailand
[b] Department of Electrical Engineering, Faculty of Engineering, King Mongkut's University of Technology Thonburi, Bangkok 10140, Thailand
[c] International Business Machines Corporation, Singapore 486048

[*] To whom correspondence should be addressed; Email: artitw@gmail.com

Customer temporal behavioral data was represented as images in order to perform churn prediction by leveraging deep learning architectures prominent in image classification. Supervised learning was performed on labeled data of over 6 million customers using deep convolutional neural networks, which achieved an AUC of 0.743 on the test dataset using no more than 12 temporal features for each customer. Unsupervised learning was conducted using autoencoders to better understand the reasons for customer churn. Images that maximally activate the hidden units of an autoencoder trained with churned customers reveal ample opportunities for action to be taken to prevent churn among strong data, no voice users.

Keywords: machine learning, deep learning, big data, churn prediction, telecommunications

Deep learning by convolutional neural networks (CNNs) has demonstrated superior performance in many image processing tasks [1,2,3]. In order to leverage such advances to predict churn and take pro-active measures to prevent it, we represent customers as images. Specifically, we construct a 2-dimensional array of normalized pixels where each row is for each day and each column is for each type of behavior tracked (Fig. 1). The type of behavior can include data usage, top up amount, top up frequency, voice calls, voice minutes, SMS messages, etc. In the training and testing data, each image is also accompanied by its label – 1 for churned and 0 for not churned. For this analysis, we examine prepaid customers in particular.