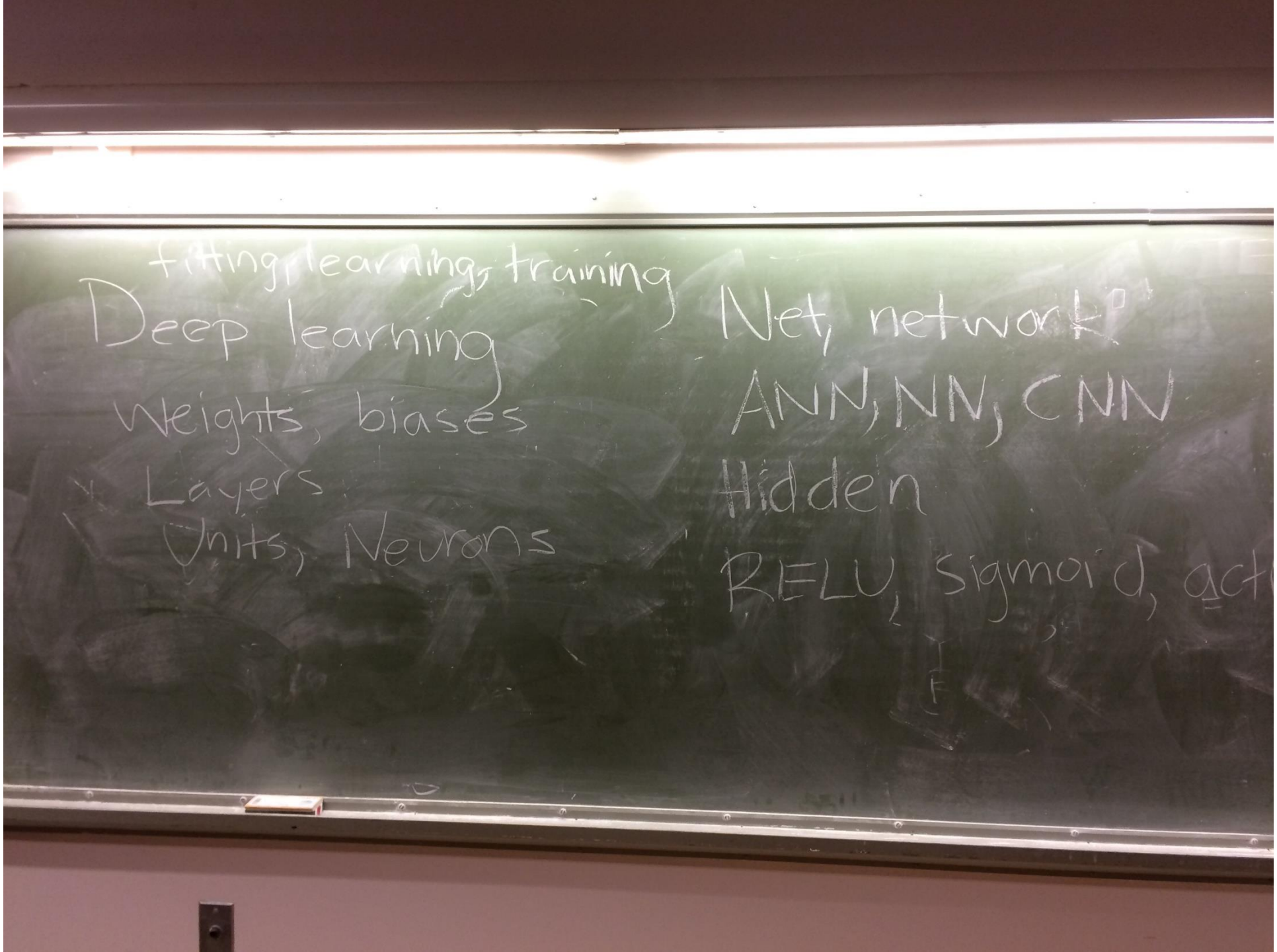
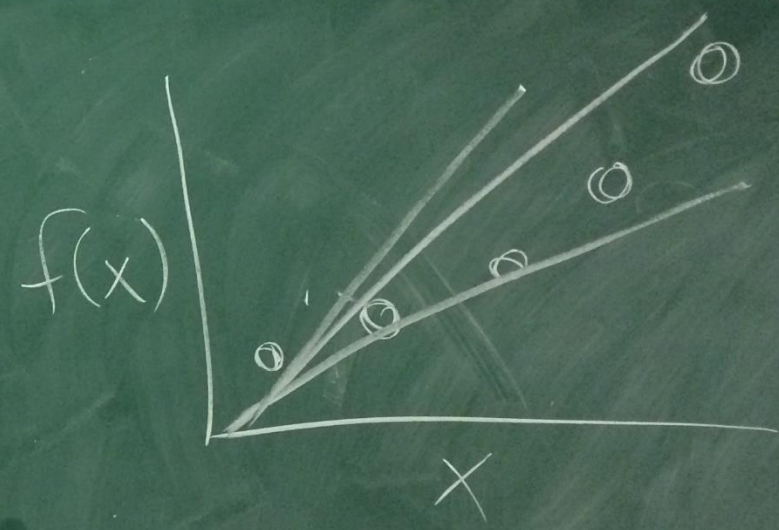

fitting, learning, training
Deep learning
Weights, biases
Layers
Units, Neurons
Net, network
ANN, NN, CNN
Hidden
RELU, sigmoid, act

$\mathbb{R}^N \to \mathbb{R}^M$

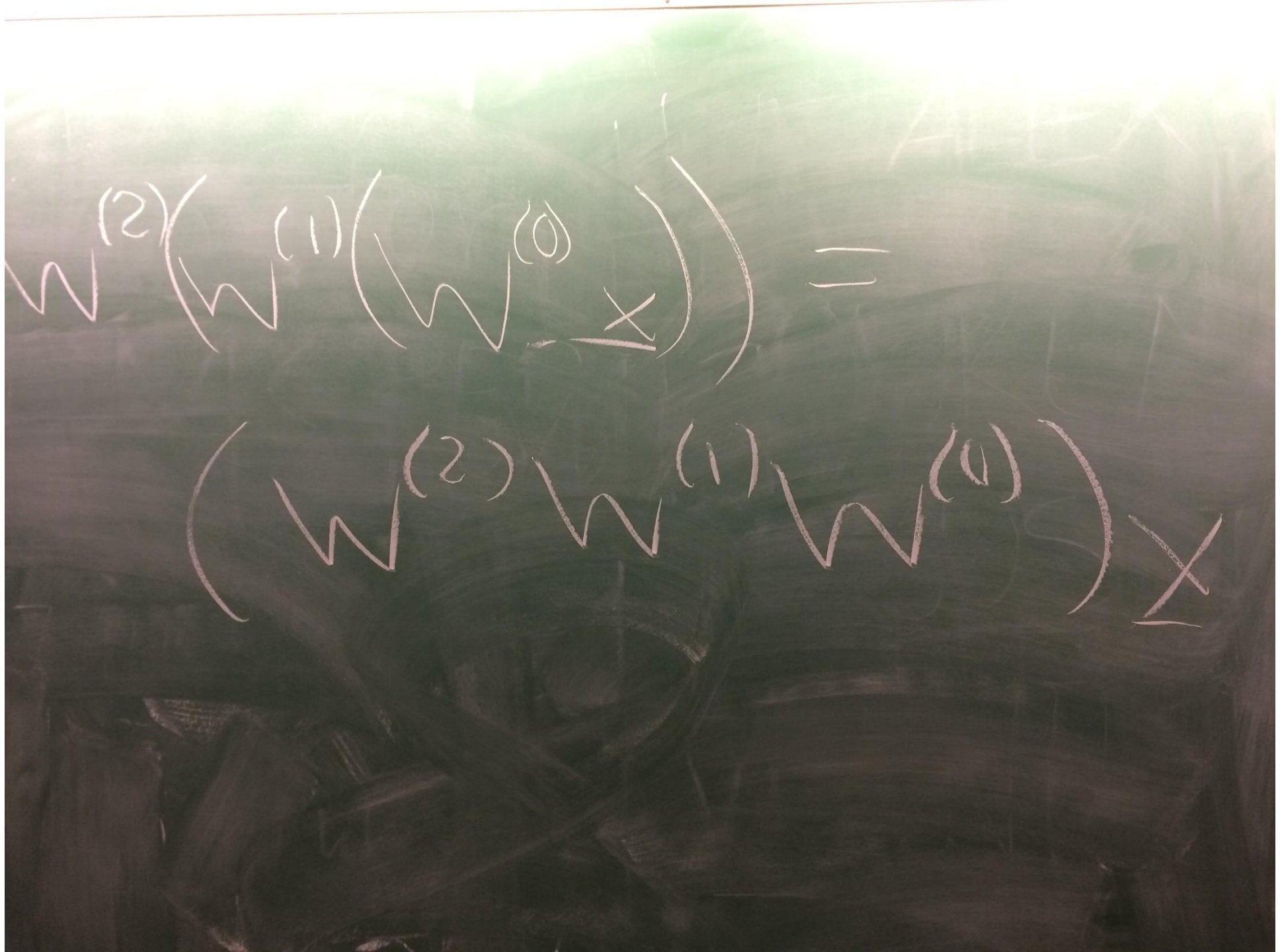
$y = ax + b$

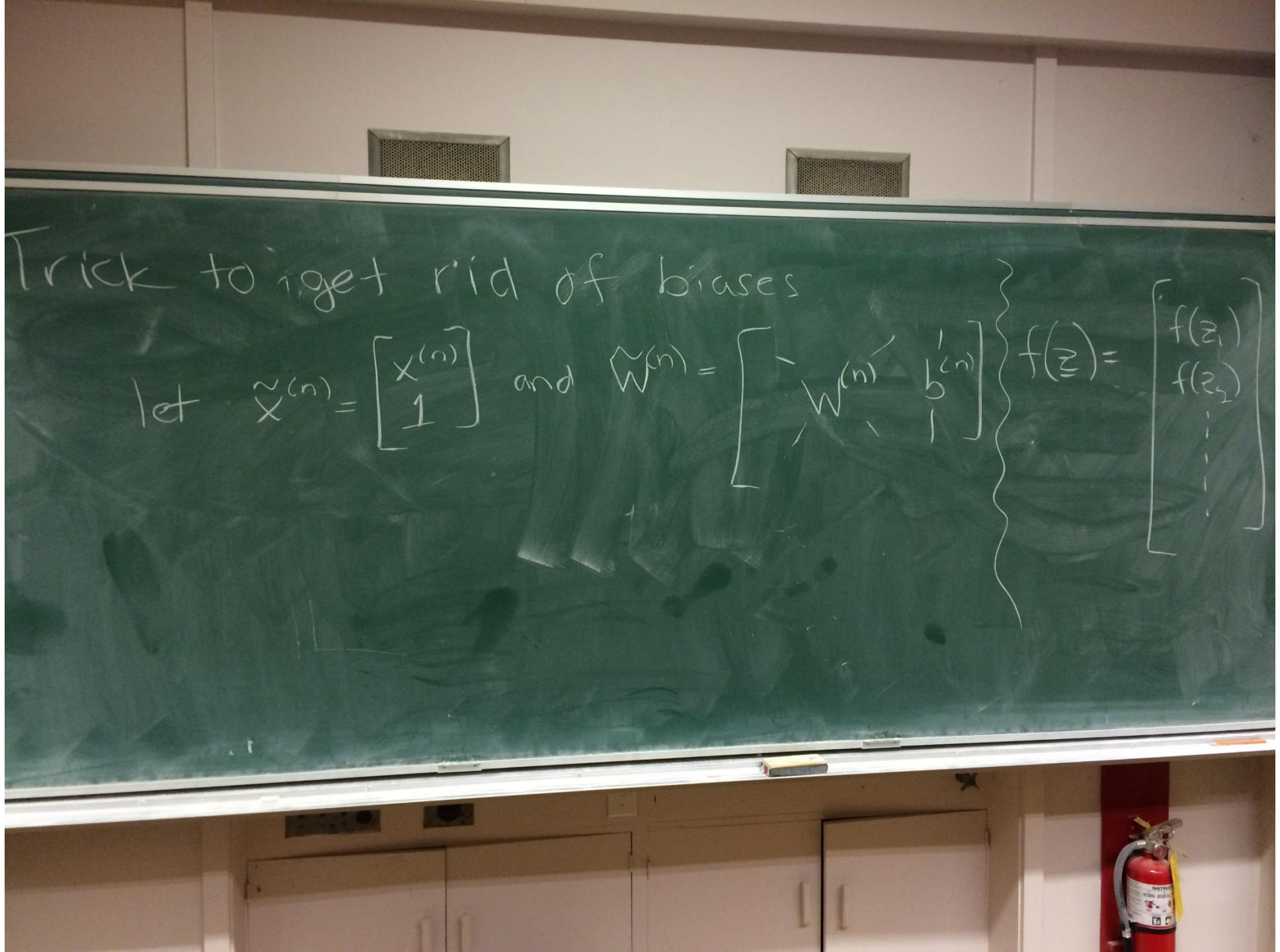
$\mathbb{R} \to \mathbb{R}$

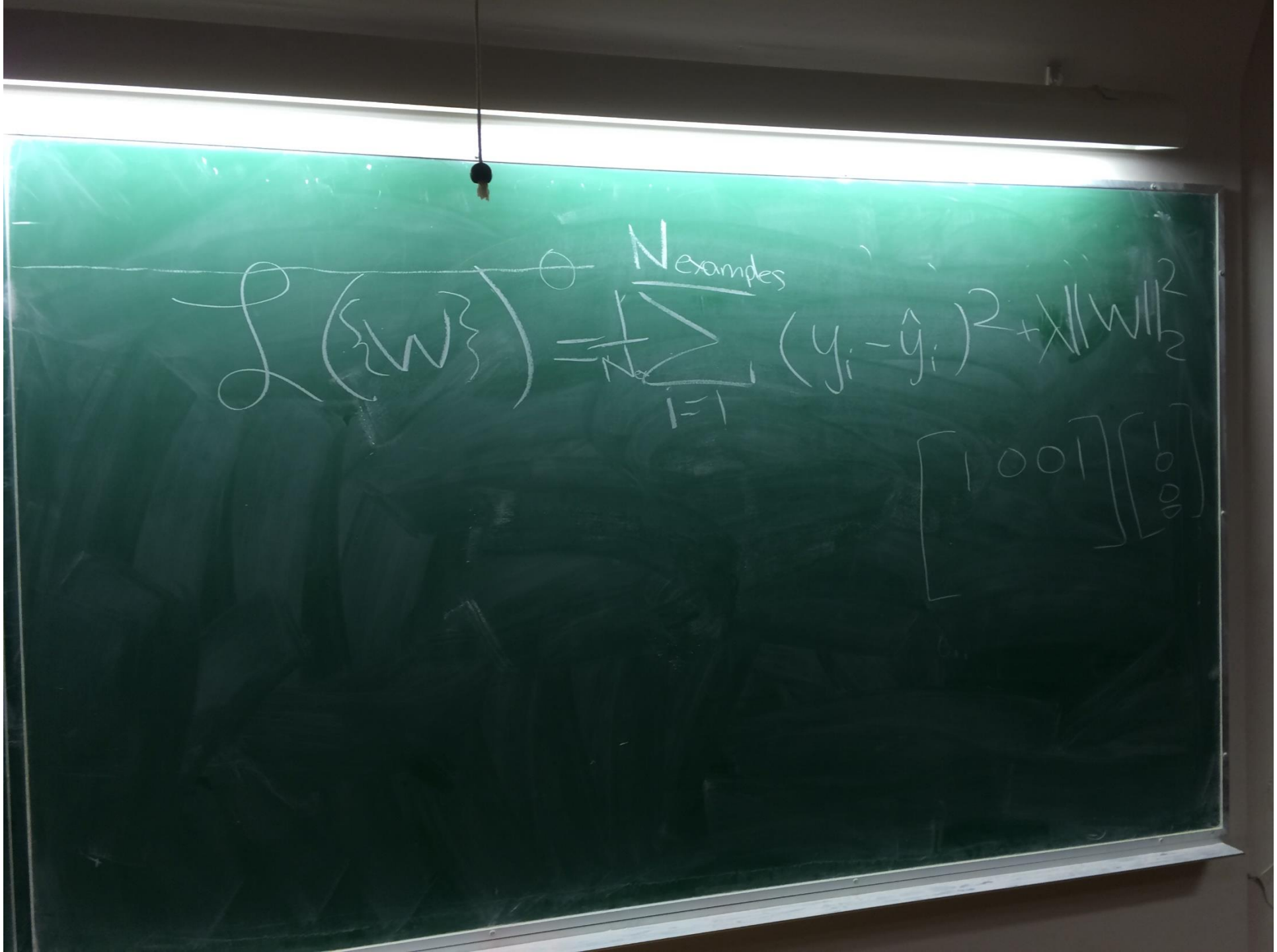
$ax^2 + bx + c$



$$\underline{x}^{(n+1)} = f\left(W^{(n)}\underline{x}^{(n)} + \underline{b}^{(n)}\right)$$


$W^{(2)}(W^{(1)}(W^{(0)}\underline{x})) =$
$(W^{(2)}W^{(1)}W^{(0)})\underline{x}$


Trick to get rid of biases
let $\tilde{x}^{(n)} = \begin{bmatrix} x^{(n)} \\ 1 \end{bmatrix}$ and $\tilde{W}^{(n)} = \begin{bmatrix} W^{(n)} & b^{(n)} \\ \cdots & 1 \end{bmatrix}$
$f(z) = \begin{bmatrix} f(z_1) \\ f(z_2) \\ \vdots \end{bmatrix}$


$\mathcal{L}(\{w\}) = \frac{1}{N_{ex}} \sum_{i=1}^{N_{examples}} (y_i - \hat{y}_i)^2 + \lambda \|w\|_2^2$
$\begin{bmatrix} 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$


threshold
f(x)
x
RELU
max(0,x)


|a|+|b|


$W^{(0)}$
$x^{(0)}$
$x^{(1)}$
N L_1 L_2 L_3 M
$NL_1 + L_1L_2 + L_2L_3 + L_3M$
Bayesian optimization
How to decide # of layers, sizes of layers...
Methodologies
famous architecture
LSTM