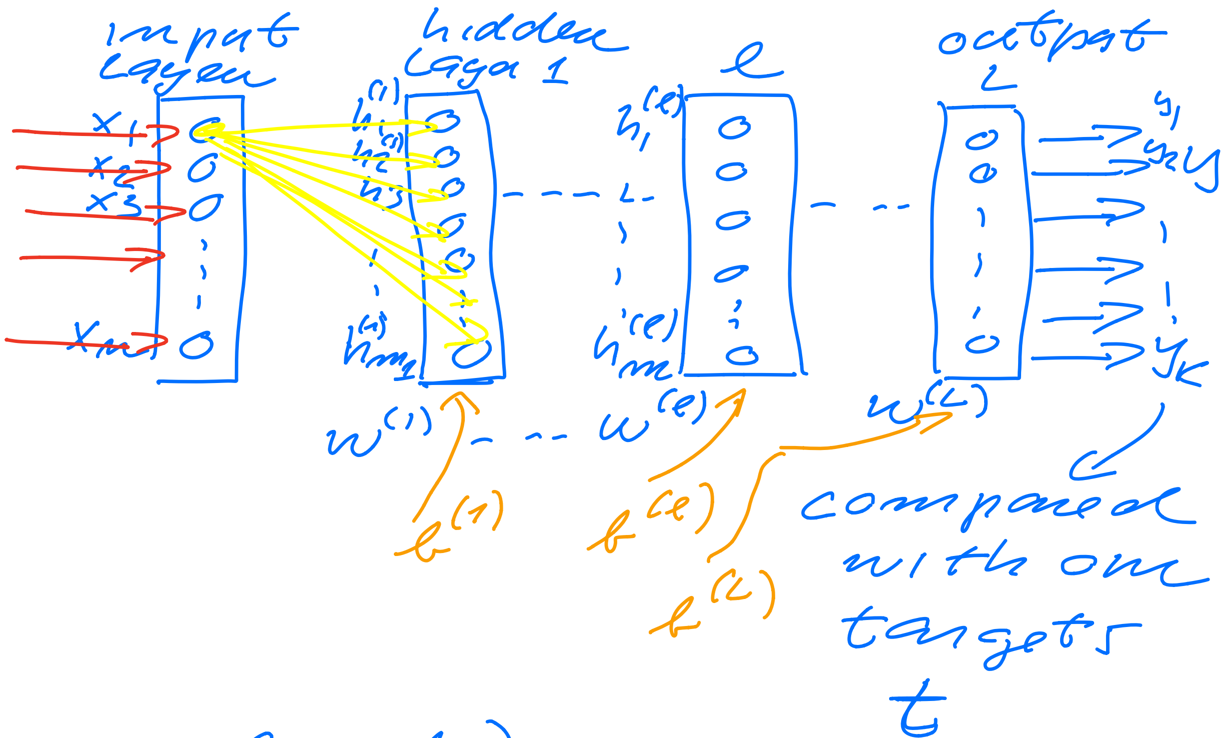
Lecture January 26

Neural network – Feed Forward (FFNN/MLP)

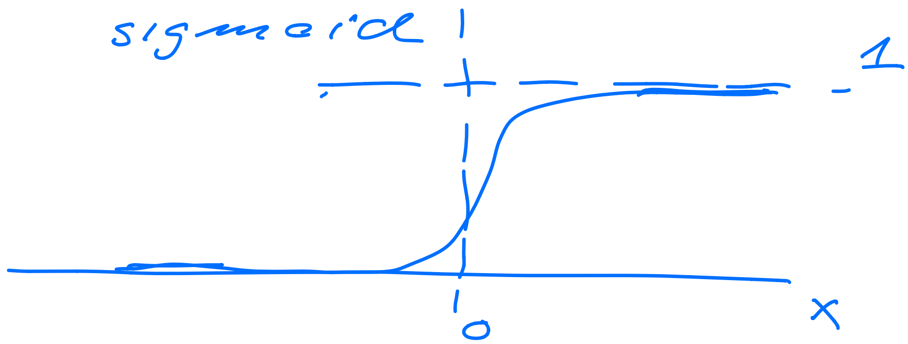


$C(W, b)$

- Backpropagation algo to define $\frac{\partial C}{\partial W}$ and $\frac{\partial C}{\partial b}$
- Use gradient methods to compute new W and new b.
- Feed Forward

- <u>Parameters</u>
  - activation functions

    sigmoid

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

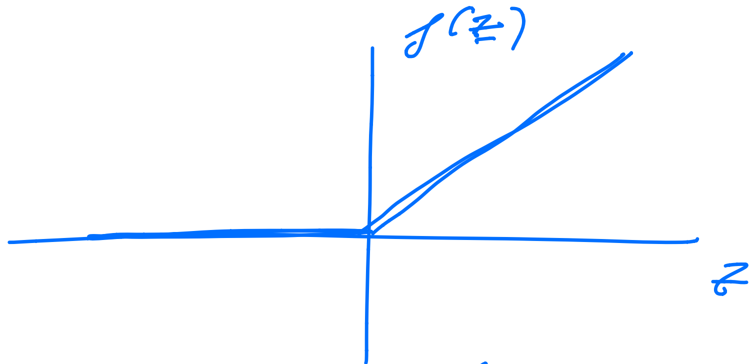
  - # hidden layers
  - # nodes/neurons
  - various algorithms for gradient descent
    - SGD with momentum
    - ADAM
    - AdaGrad
    - RMSprop

    ⋮
- learning rate γ
- hyperparameter λ (shrinkage/regularization)

RELU = Rectified Linear Unit

$f(z) = \max(0, z)$

outputs zero for half of the domain

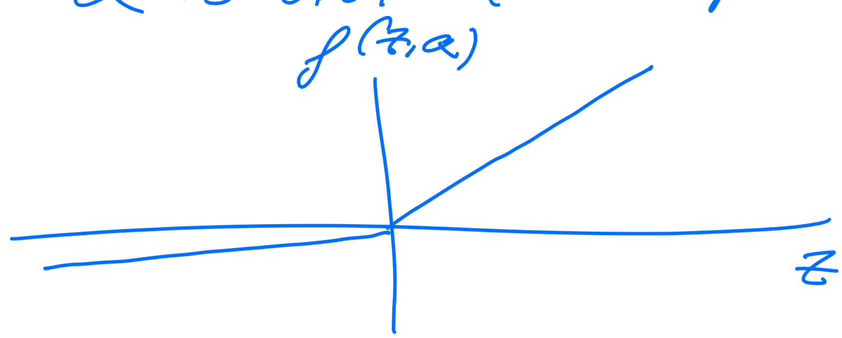


problem when $z \le 0$

Leaky ReLU:

$$f(z, \alpha) = \max(0, z) + \alpha \min(0, z)$$

$\alpha \approx 0.01$ (new parameter)



Convolutional NN (CNN)

- CNNs are NNs that use convolution instead of general matrix multiplication in at least one of the layers.

- convolution:

  location of spaceship as function, tracked with a laser sensor $\longrightarrow$ output $x(t)$, x, t are real-valued.
  assume that the output is noisy, Focus on recent measurements.
  a - parameter ~ age of measurements, recent measurements are weighted by a function $w(a)$

$$s(t) = \int x(a)\, w(t-a)\, da$$

= convolution.

↓ Feature

Kernel

$$= s(t) = (x * w)(t)$$

input

Discrete version:

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)\, w(t-a)$$

Two dim - image

$I(i,j)$ — input

$$S(i,j) = (I * K)(i,j)$$

Kernel

$$= \sum_{m} \sum_{n} \underbrace{I(m,n)}_{X} \underbrace{K(i-m, j-n)}_{W}$$

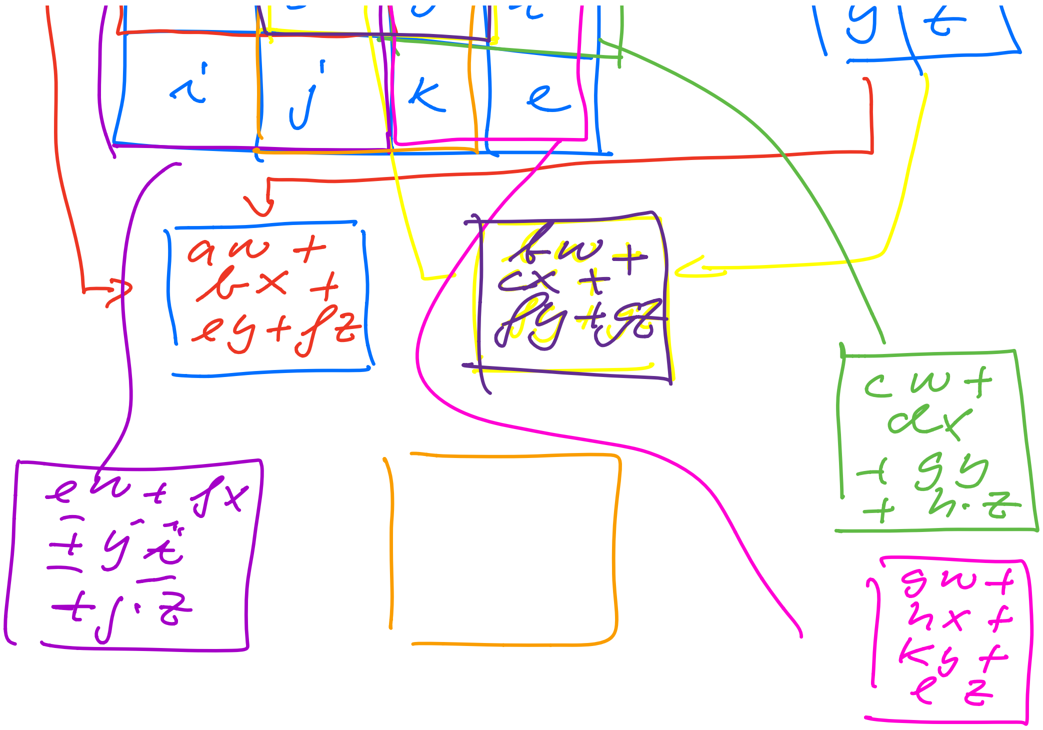
MNIST
1797 images
64 features

Input

a	b	c	d
e	f	g	h

Kernel

w	x
y	z



i
j
k
e
y
t
aw + bx + ey + fz
bw + cx + fy + gz
cw + dx + gy + h·z
ew + fx + y·t + j·z
gw + hx + ky + e z