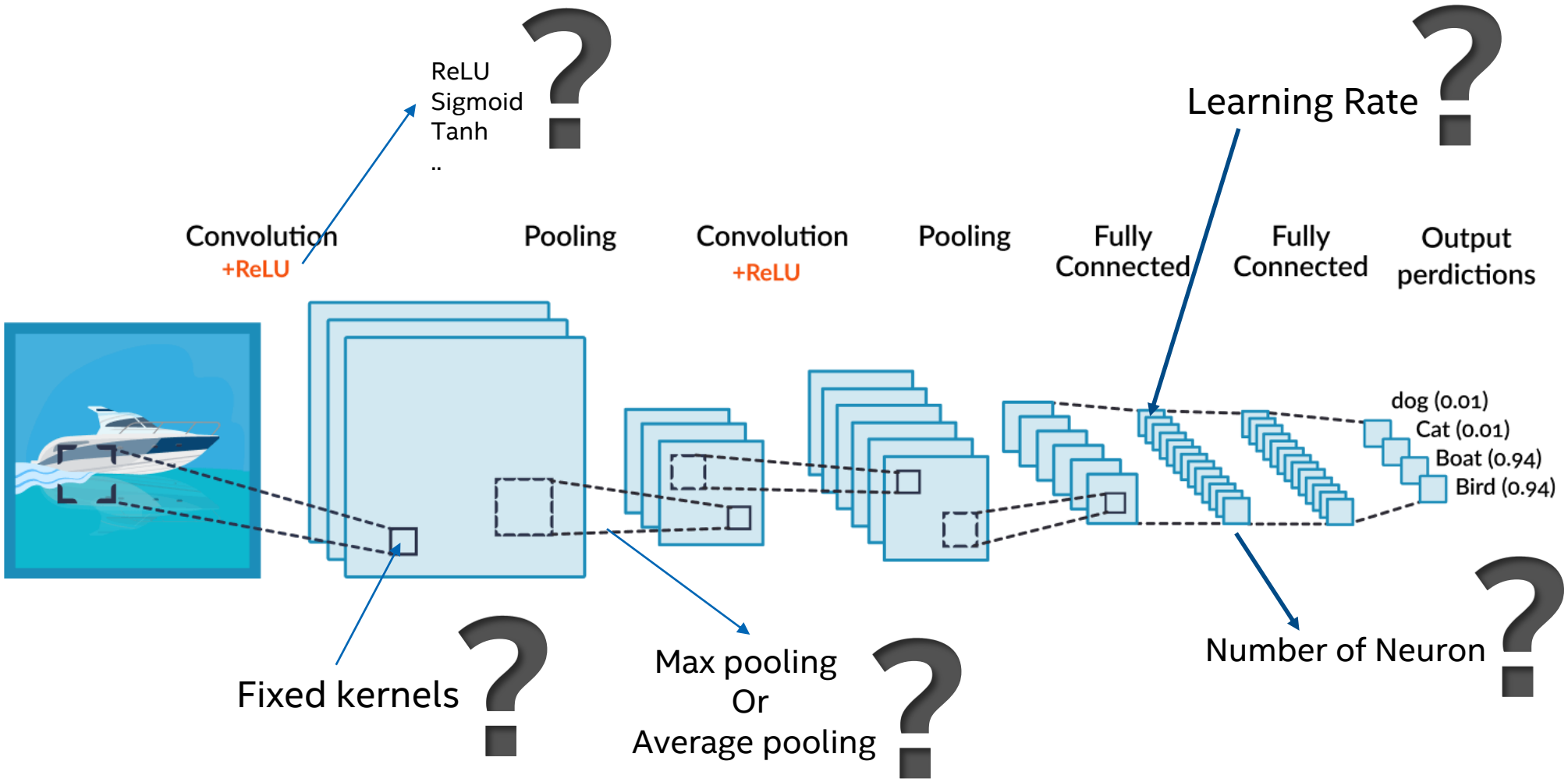
Overview

Fractional Adaptation of Activation Functions In Neural Networks

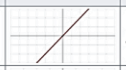
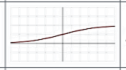
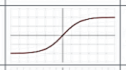
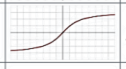
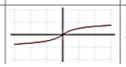
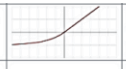
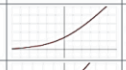
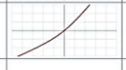
Presenter: Julio Zamora, PhD.

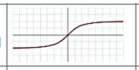
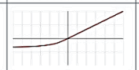
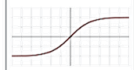
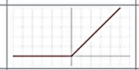
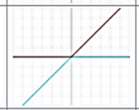
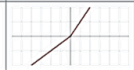
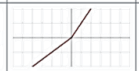
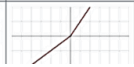
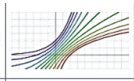
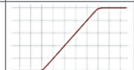
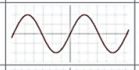
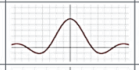
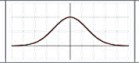
Authors:
Julio Zamora Esquivel, Jesus Adan Cruz Vargas, Paulo Lopez-Meyer, Hector Cordourier Maruri, Jose Rodrigo Camacho Perez, Omesh Tickoo

Hyper parameters of a NN model



Activation Function

Name	Plot	Equation	Derivative (with respect to x)
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a. Sigmoid or Soft step)		$f(x) = \sigma(x) = \frac{1}{1+e^{-x}}$[1]	$f'(x) = f(x)(1-f(x))$
TanH		$f(x) = \tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2+1}$
ArSinH		$f(x) = \sinh^{-1}(x) = \ln\left(x + \sqrt{x^2+1}\right)$	$f'(x) = \frac{1}{\sqrt{x^2+1}}$
ElliotSig[9][10][11] Softsign[12][13]		$f(x) = \frac{x}{1+\lvert x\rvert}$	$f'(x) = \frac{1}{(1+\lvert x\rvert)^2}$
Exponential linear unit (ELU)[20]		$f(\alpha, x) = \begin{cases} \alpha(e^x - 1) & \text{for } x \leq 0 \\ x & \text{for } x > 0 \end{cases}$	$f'(\alpha, x) = \begin{cases} f(\alpha, x) + \alpha & \text{for } x \leq 0 \\ 1 & \text{for } x > 0 \end{cases}$
Scaled exponential linear unit (SELU)[21]		$f(\alpha, x) = \lambda \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$ with $\lambda = 1.0507$ and $\alpha = 1.67326$	$f'(\alpha, x) = \lambda \begin{cases} \alpha(e^x) & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
S-shaped rectified linear activation unit (SReLU)[22]		$f_{t_l, a_l, t_r, a_r}(x) = \begin{cases} t_l + a_l(x - t_l) & \text{for } x \leq t_l \\ x & \text{for } t_l < x < t_r \\ t_r + a_r(x - t_r) & \text{for } x \geq t_r \end{cases}$ t_l, a_l, t_r, a_r are parameters.	$f'_{t_l, a_l, t_r, a_r}(x) = \begin{cases} a_l & \text{for } x \leq t_l \\ 1 & \text{for } t_l < x < t_r \\ a_r & \text{for } x \geq t_r \end{cases}$
Adaptive piecewise linear (APL)[23]		$f(x) = \max(0, x) + \sum_{s=1}^{S} a_i^s \max(0, -x + b_i^s)$	$f'(x) = H(x) - \sum_{s=1}^{S} a_i^s H(-x + b_i^s)$[4]
SoftPlus[24]		$f(x) = \ln(1 + e^x)$	$f'(x) = \frac{1}{1+e^{-x}}$
Bent identity		$f(x) = \frac{\sqrt{x^2+1} - 1}{2} + x$	$f'(x) = \frac{x}{2\sqrt{x^2+1}} + 1$
Sigmoid Linear Unit (SiLU)[25] (AKA SiL[26] and Swish-1[27])		$f(x) = x \cdot \sigma(x)$[5]	$f'(x) = f(x) + \sigma(x)(1 - f(x))$[6]

Inverse square root unit (ISRU)[14]		$f(x) = \frac{x}{\sqrt{1+\alpha x^2}}$	$f'(x) = \left(\frac{1}{\sqrt{1+\alpha x^2}}\right)^3$
Inverse square root linear unit (ISRLU)[14]		$f(x) = \begin{cases} \frac{x}{\sqrt{1+\alpha x^2}} & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \left(\frac{1}{\sqrt{1+\alpha x^2}}\right)^3 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Square Nonlinearity (SQNL)[11]		$f(x) = \begin{cases} 1 & : x > 2.0 \\ x - \frac{x^2}{4} & : 0 \leq x \leq 2.0 \\ x + \frac{x^2}{4} & : -2.0 \leq x < 0 \\ -1 & : x < -2.0 \end{cases}$	$f'(x) = 1 \mp \frac{x}{2}$
Rectified linear unit (ReLU)[15]		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Bipolar rectified linear unit (BReLU)[16]		$f(x_i) = \begin{cases} ReLU(x_i) & \text{if } i \bmod 2 = 0 \\ -ReLU(-x_i) & \text{if } i \bmod 2 \neq 0 \end{cases}$	$f'(x_i) = \begin{cases} ReLU'(x_i) & \text{if } i \bmod 2 = 0 \\ -ReLU'(-x_i) & \text{if } i \bmod 2 \neq 0 \end{cases}$
Leaky rectified linear unit (Leaky ReLU)[17]		$f(x) = \begin{cases} 0.01x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0.01 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric rectified linear unit (PReLU)[18]		$f(\alpha, x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(\alpha, x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Randomized leaky rectified linear unit (RReLU)[19]		$f(\alpha, x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$[3]	$f'(\alpha, x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
SoftExponential[28]		$f(\alpha, x) = \begin{cases} -\frac{\ln(1-\alpha(x+\alpha))}{\alpha} & \text{for } \alpha < 0 \\ x & \text{for } \alpha = 0 \\ \frac{e^{\alpha x}-1}{\alpha} + \alpha & \text{for } \alpha > 0 \end{cases}$	$f'(\alpha, x) = \begin{cases} \frac{1}{1-\alpha(\alpha+x)} & \text{for } \alpha < 0 \\ e^{\alpha x} & \text{for } \alpha \geq 0 \end{cases}$
Soft Clipping[29]		$f(\alpha, x) = \frac{1}{\alpha} \log \frac{1 + e^{\alpha x}}{1 + e^{\alpha(x-1)}}$	$f'(\alpha, x) = \frac{1}{2} \sinh\left(\frac{p}{2}\right) \operatorname{sech}\left(\frac{px}{2}\right) \operatorname{sech}\left(\frac{p}{2}(1-x)\right)$
Sinusoid[30]		$f(x) = \sin(x)$	$f'(x) = \cos(x)$
Sinc		$f(x) = \begin{cases} 1 & \text{for } x = 0 \\ \frac{\sin(x)}{x} & \text{for } x \neq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x = 0 \\ \frac{\cos(x)}{x} - \frac{\sin(x)}{x^2} & \text{for } x \neq 0 \end{cases}$
Gaussian		$f(x) = e^{-x^2}$	$f'(x) = -2xe^{-x^2}$

Groups of activation function

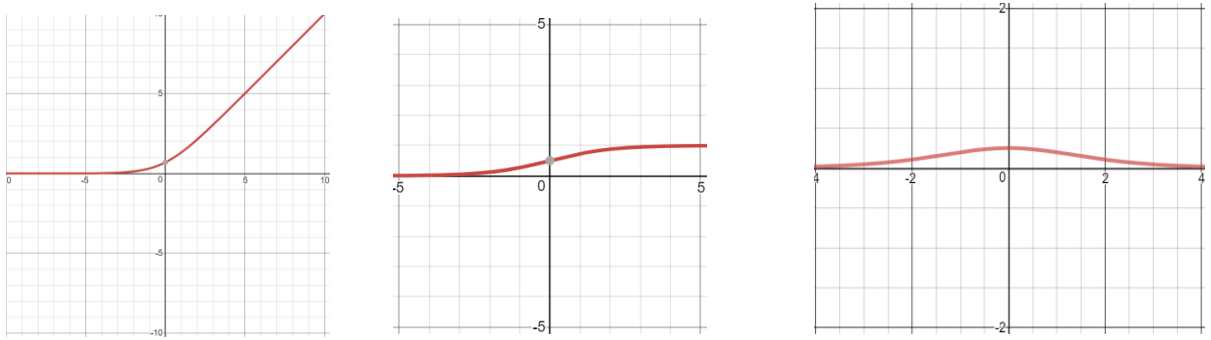
Group1

(ReLU) derivative Binary Step

Primitive

Group2

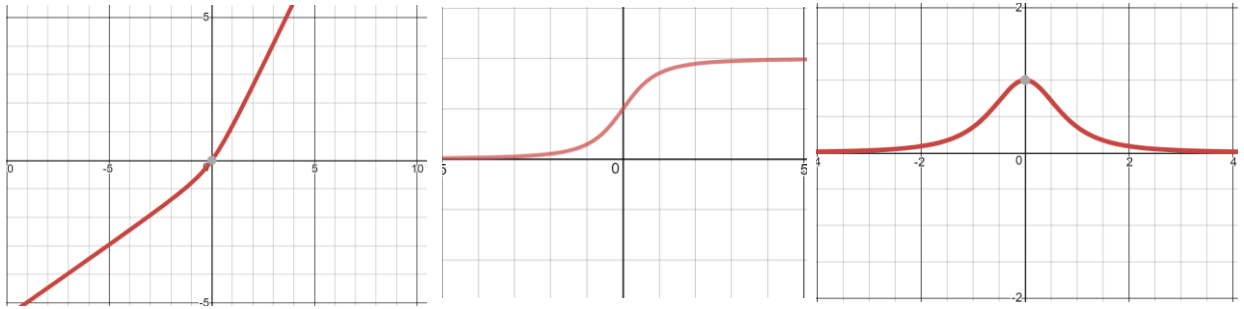
Softplus derivative Sigmoid derivative RBF like Gaussian



Primitive

Group3

Bent Identity derivative ISRU derivative RBF like Gaussian

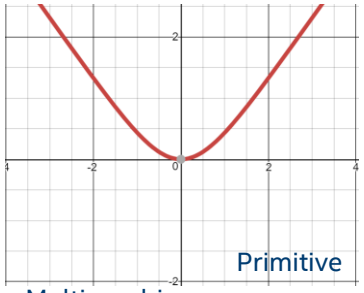


~LeakyReLU

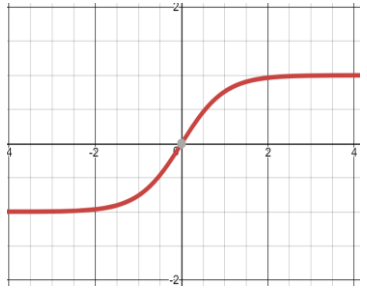
Groups of Activation function

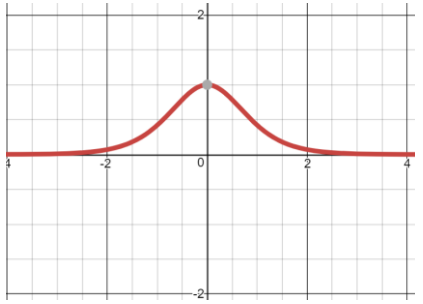
Group4

Ln(cosh(x)) —derivative→ Tanh —derivative→ RBF like Gaussian $\operatorname{sech}^2(x)$



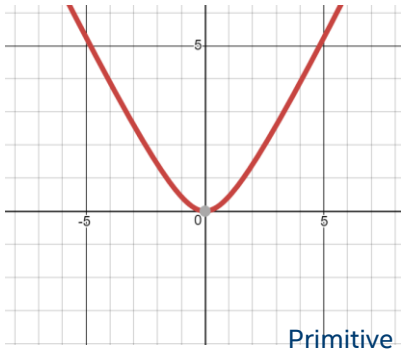
~Multiquadric



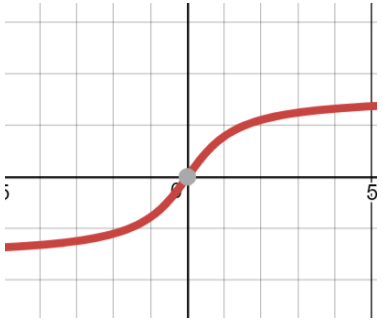


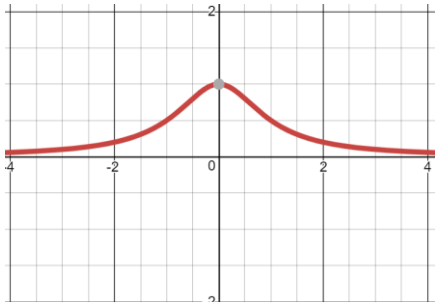
Group5

$x\tan^{-1}(x)-\frac{1}{2}\ln(1+x^2)$ —derivative→ Atan(x) —derivative→ RBF Like Gausian $\frac{1}{(x^2+1)}$



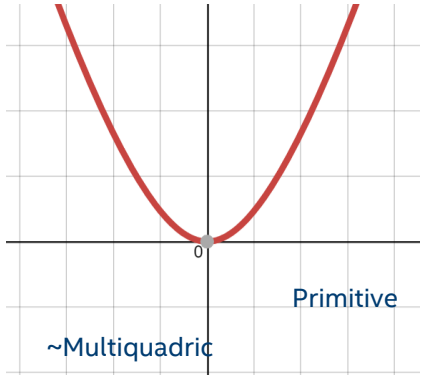
~Multiquadric

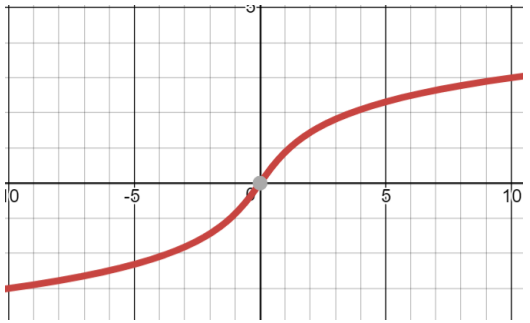


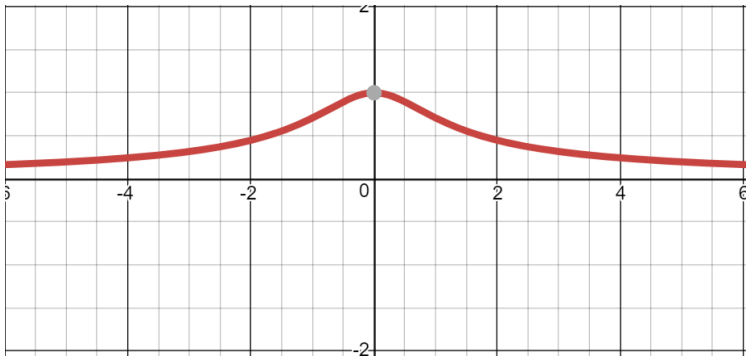


Grouping the activation function

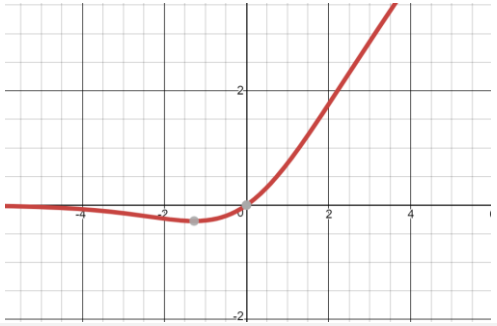
$x \sinh^{-1}(x) - \sqrt{(1 + x \cdot x)} + 1$ → derivative → $\sinh^{-1}(x)$ → derivative → RBF Like Gausian $\frac{1}{\sqrt{(x^2+1)}}$

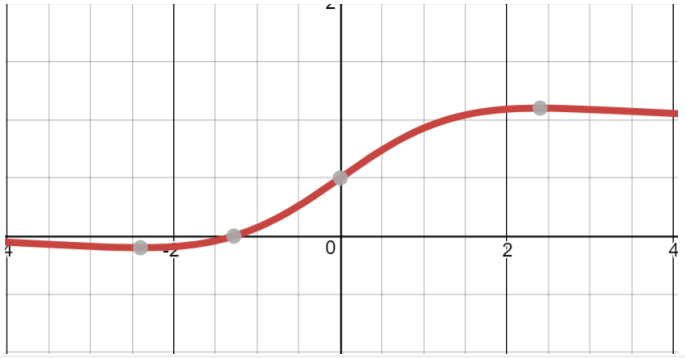


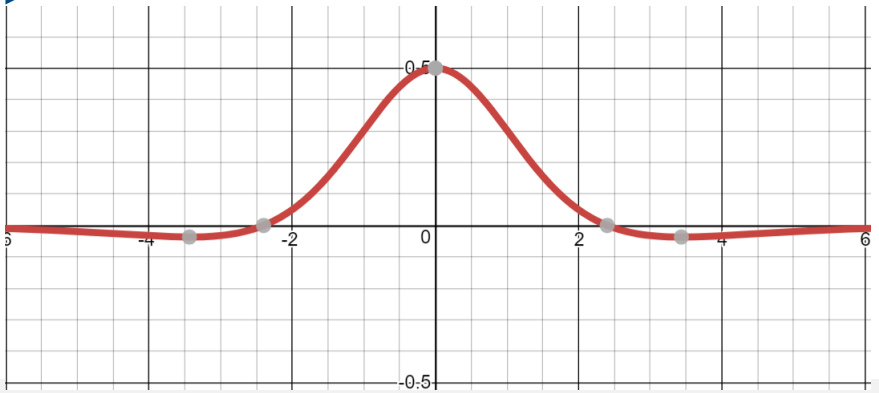




Swish/SiL → derivative → Sigmoid like → derivative → RBF Like Gausian







Fractional calculus

$$y' = \frac{\mathrm{d}y}{\mathrm{d}x}, y'' = \frac{\mathrm{d}^2y}{\mathrm{d}x}, y''' = \frac{\mathrm{d}^3y}{\mathrm{d}x}$$

Is it possible to generate the 1.5 Derivative?

Fractional derivative

$$f(x) = x^k$$

$$\begin{aligned}
\frac{\mathrm{d}f(x)}{\mathrm{d}x} &= kx^{k-1}, \\
\frac{\mathrm{d}^2 f(x)}{\mathrm{d}x^2} &= k(k-1)x^{k-2}, \\
\frac{\mathrm{d}^{\mathrm{a}} f(x)}{\mathrm{d}x^a} &= k(k-1)(k-2)\cdots(k-a+1)x^{k-a}
\end{aligned}$$

$$\frac{\mathrm{d}^{\mathrm{a}} f(x)}{\mathrm{d}x^a} = \frac{k!}{(k-a)!}x^{k-a}$$

Gamma Function

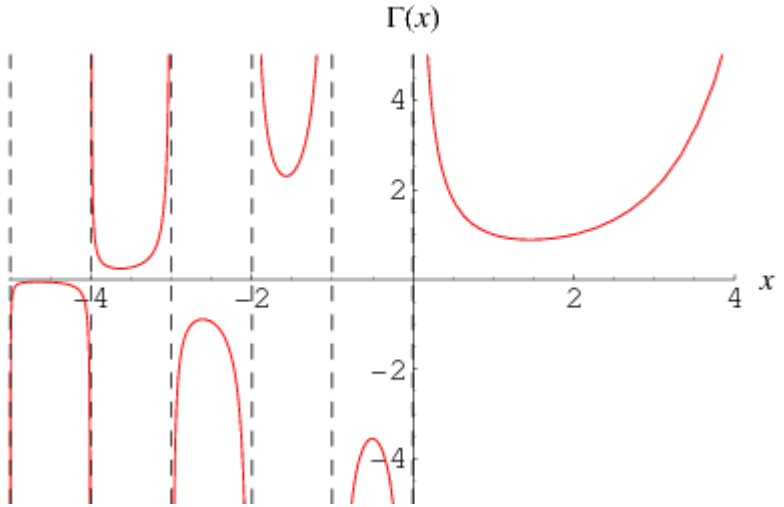
Gamma

$$\Gamma(z) = \int_0^\infty t^{(z-1)} e^{-t} dt,$$

$$\Gamma(n) = (n-1)!$$

$$\Gamma(z) = \frac{e^{-\gamma z}}{z} \prod_{k=1}^{\infty} \left(\left(1 + \frac{z}{k}\right)^{-1} e^{\frac{z}{k}} \right)$$

Plot of Gamma



Fractional Derivative

$$\frac{\mathrm{d}^{\mathrm{a}} f(x)}{\mathrm{d}x^a} = \frac{k!}{(k-a)!} x^{k-a}$$

Using Gamma function

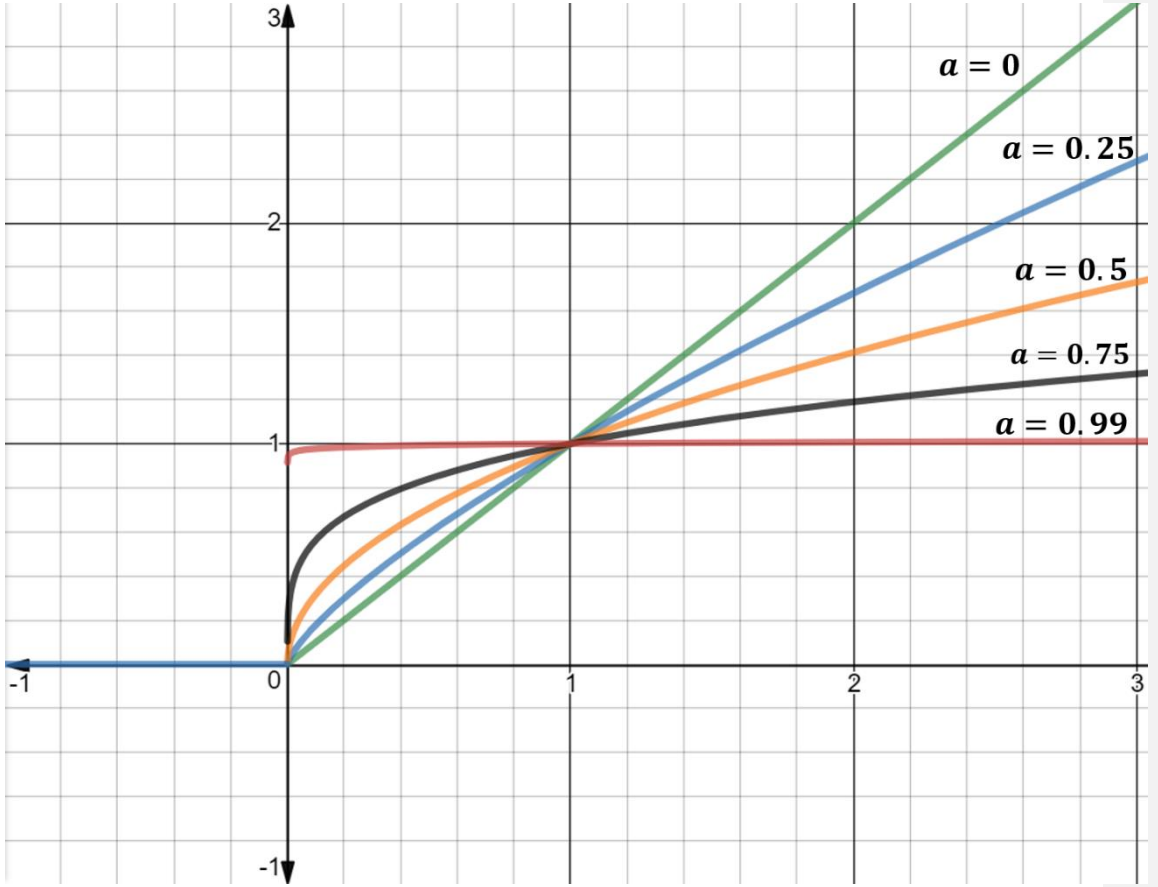
$$D^a f(x) = \frac{\mathrm{d}^{\mathrm{a}} f(x)}{\mathrm{d}x^a} = \frac{\Gamma(k+1)}{\Gamma(k+1-a)} x^{k-a}.$$

Generalized ReLU

$$f(x) = \begin{cases} x & x > 0 \\ 0 & x \leqslant 0 \end{cases}$$

$$g(x) = \begin{cases} D^a x & x > 0 \\ 0 & x \leqslant 0 \end{cases}$$

$$g(x) = \frac{x^{1-a}}{\Gamma(2-a)}$$



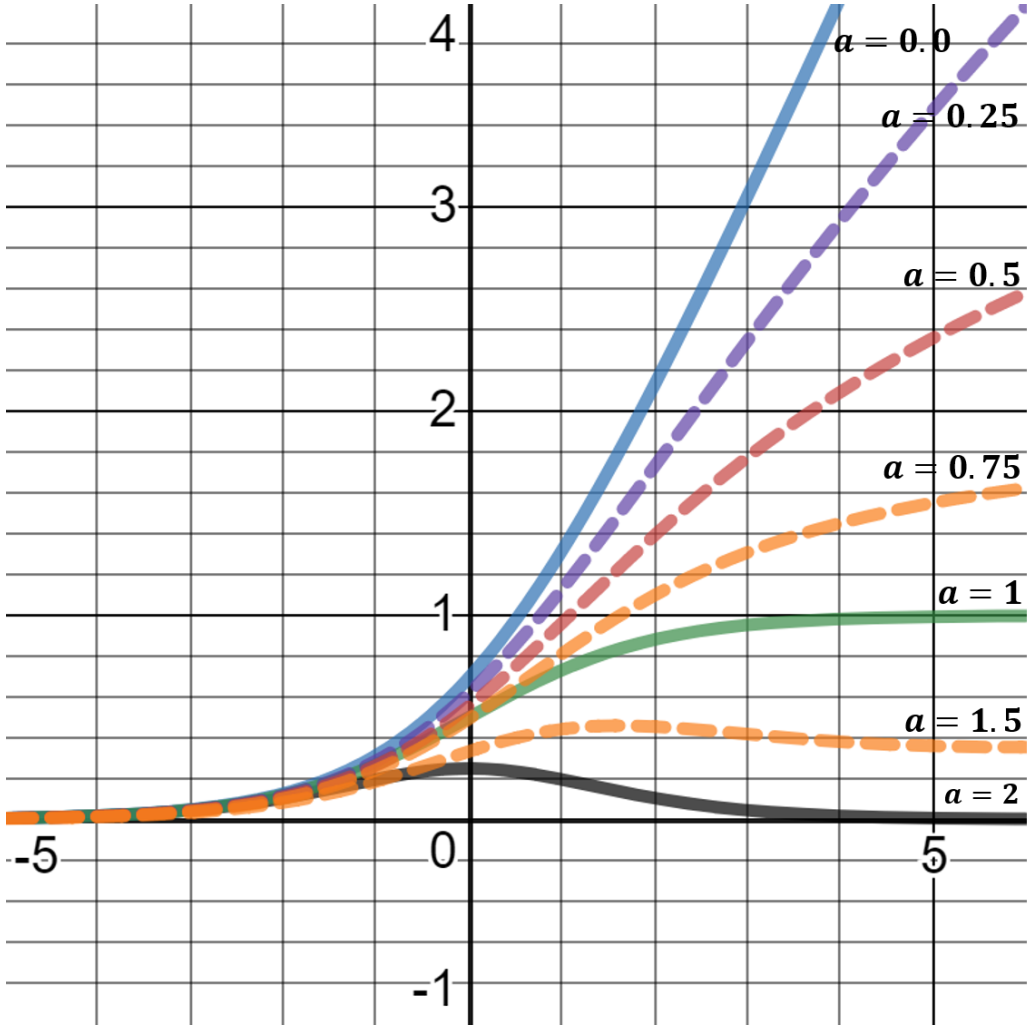
Generalized Sigmoid

Softplus

$g(x) = ln(1 + e^x)$

$$g(x) = D^a ln(1 + e^x)$$

$$g(x) = \lim_{h \to 0} \frac{1}{h^a} \sum_{n=0}^{\infty} (-1)^n \frac{\Gamma(a+1) \ln\left(1 + e^{(x-nh)}\right)}{\Gamma(n+1)\Gamma(1-n+a)}$$

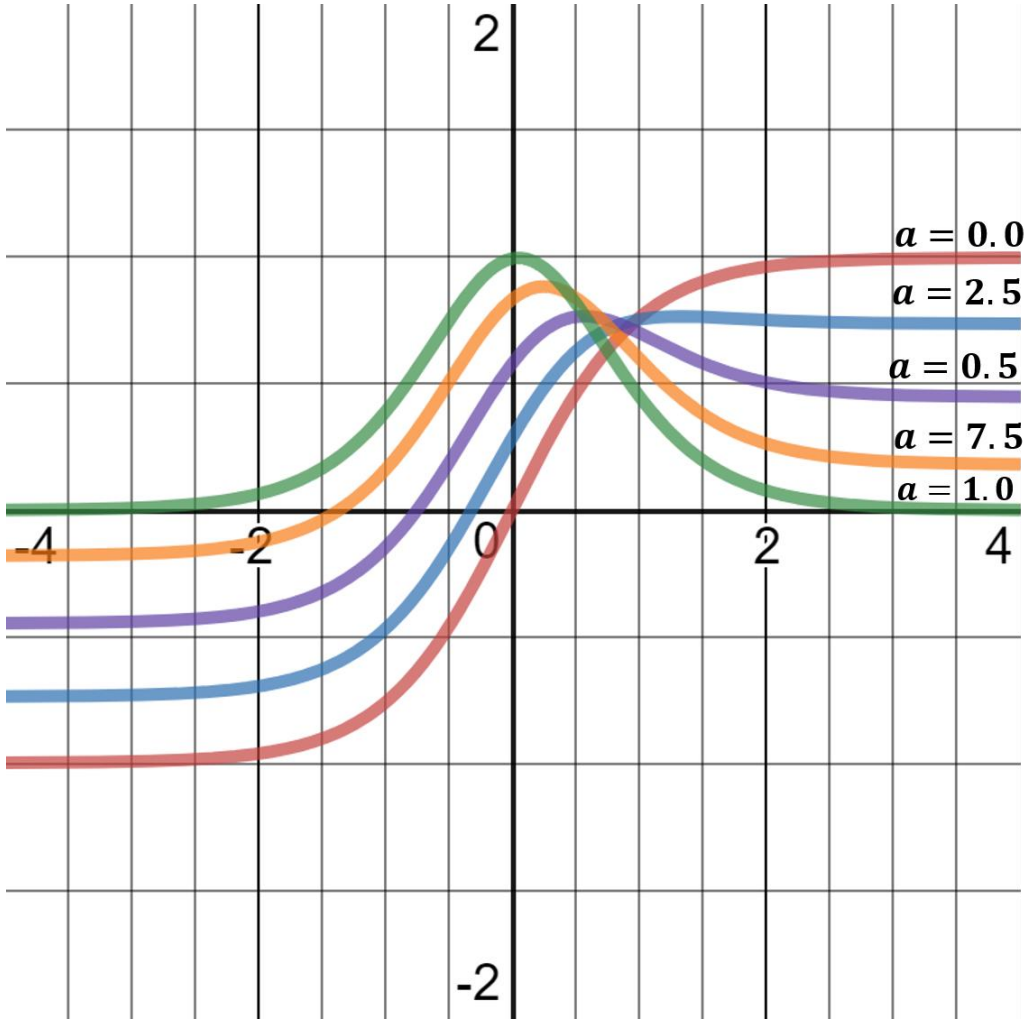


Generalized Hyperbolic Tangent

$$f(x) = Tanh(x)$$

$$g(x) = D^a Tanh(x)$$

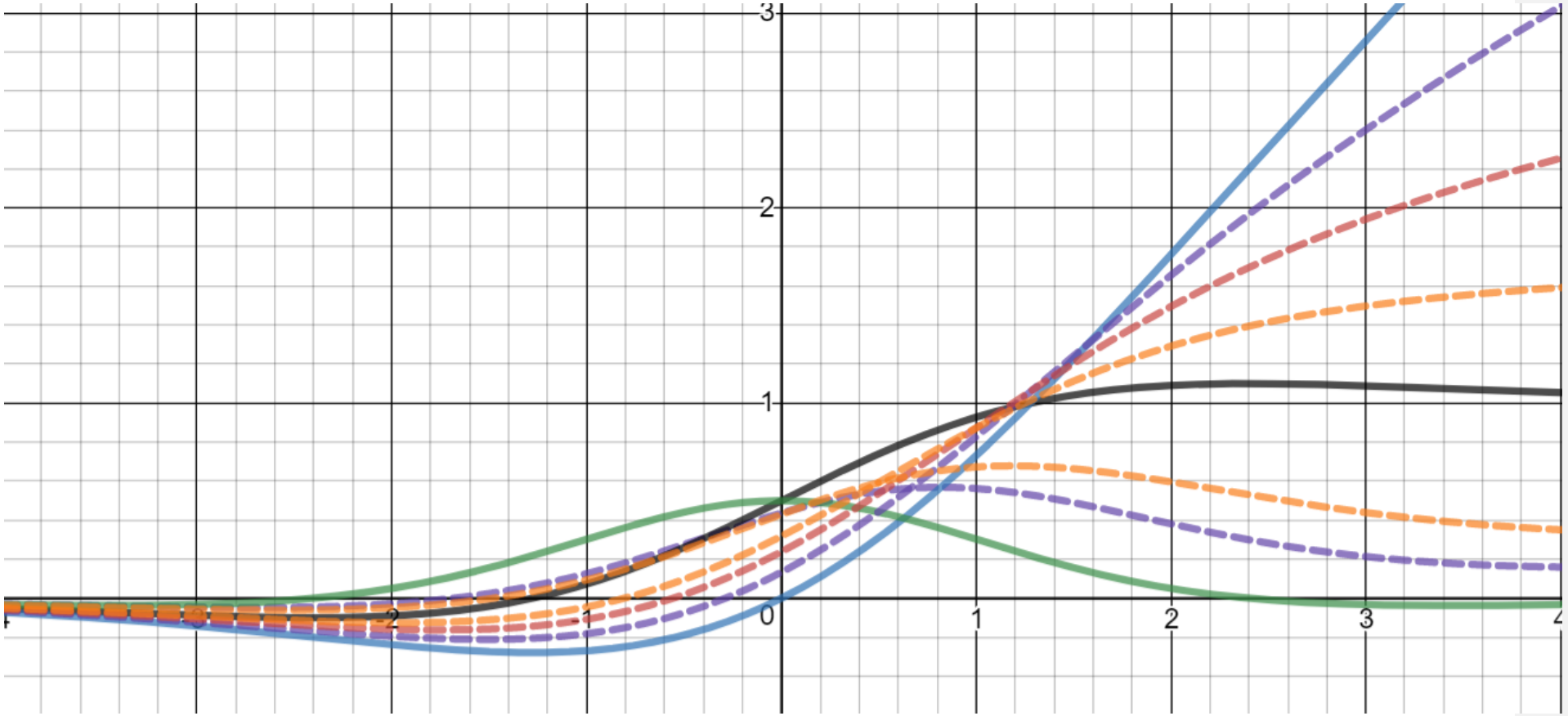
$$g(x) = \lim_{h \to 0} \frac{1}{h^a} \sum_{n=0}^{\infty} (-1)^n \frac{\Gamma(a+1) \tanh(x - n \cdot h)}{\Gamma(n+1)\Gamma(1-n+a)}$$



Generalized swish

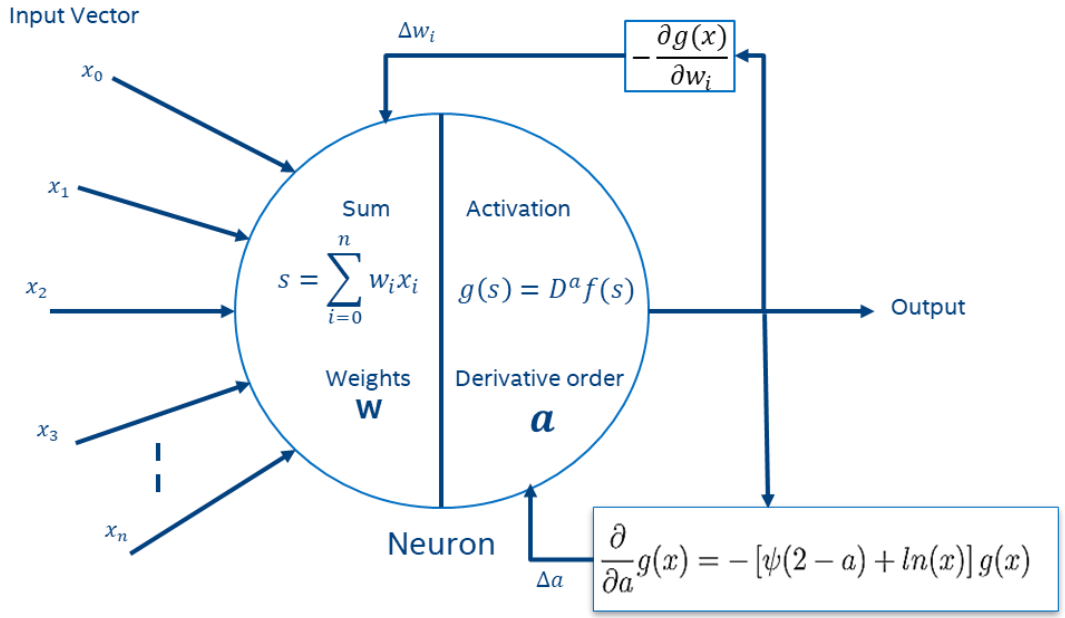
$$f(x) = \frac{x}{1 + e^{-x}}$$

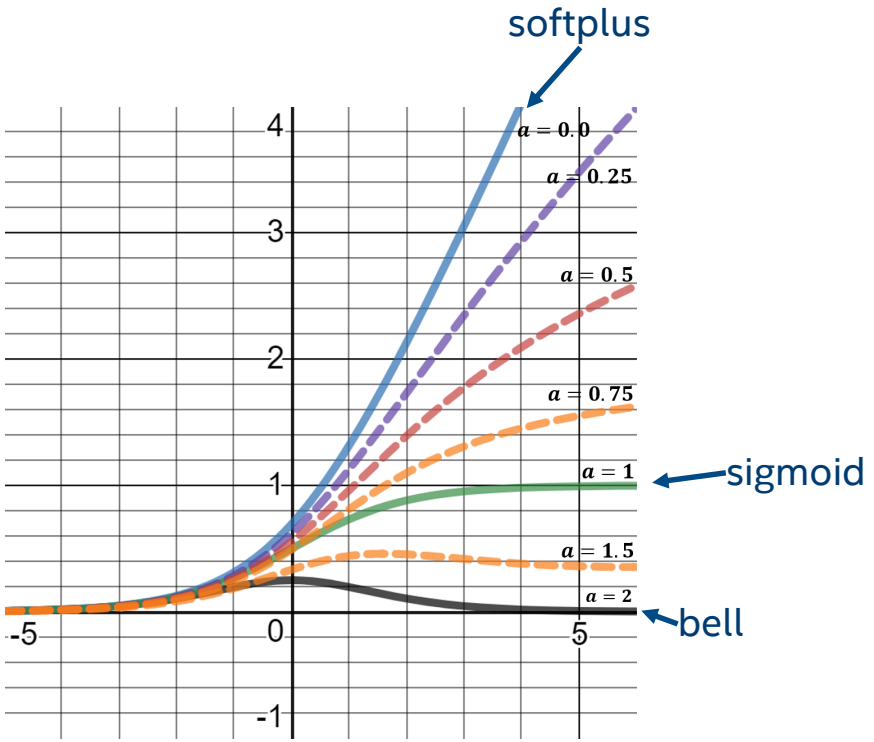
$$g(x) = D^{a}\left(\frac{x}{1+e^{-x}}\right)$$



Adaptive activation function

The neurons of the NN were modified to adjust not only their weights, but also their activation function. The parameter ***a*** represents the derivative order of the activation function, producing a generalization, it means an activation function that can morph from softplus to sigmoid to bell-like shape, for the best result.





Learning ReLU

$$g(x) = \frac{x^{1-a}}{\Gamma(2-a)}$$

$$\frac{\partial}{\partial a} g(x) = -\left[\frac{\Gamma'(2-a)}{\Gamma(2-a)^2} + \frac{ln(x)}{\Gamma(2-a)}\right] x^{1-a}$$

Using Digamma function

$$\psi(z+1) = \frac{\Gamma'(z+1)}{\Gamma(z+1)} = -\gamma + \sum_{n \geq 1}\left(\frac{z}{n(n+z)}\right) \quad \Rightarrow \quad \psi(2-a) = -\gamma + \sum_{n=1}^{\infty}\left(\frac{1-a}{n^2+n-an}\right)$$

$$\frac{\partial}{\partial a} g(x) = -\left[\psi(2-a) + ln(x)\right] g(x)$$

Adaptive activation function

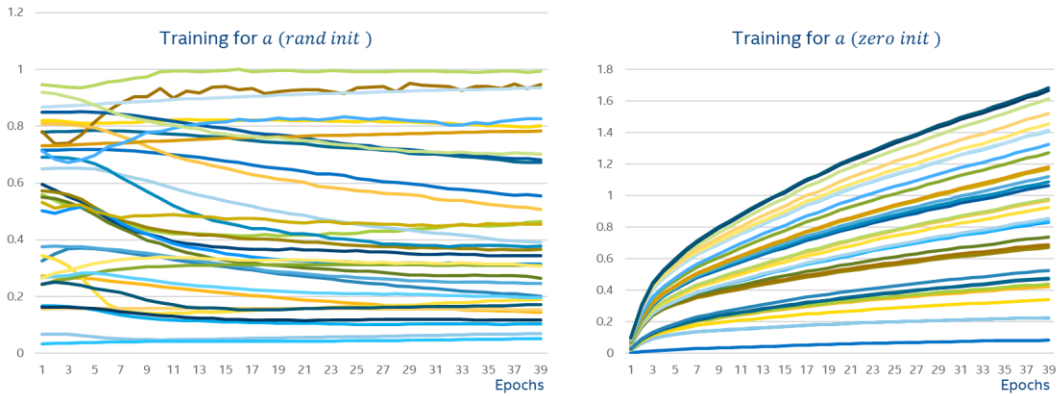
The results using this technology in comparison with academic state-of-the-art neural nets shows performance improvements for image recognition datasets, with no increase in parameter size.

- The optimal solution is not a combination of the known activation functions:
  It is a fractional value
- Using AAF in 18 layers gives 95.08% accuracy :
  18(AAF)Layers = 1000 Layers in CIFAR10
- Two Percent Improvement with similar number of parameters

CIFAR10 Results

Table 1. Comparing ResNet18* (with adaptive activation function) Vs reported ResNet topologies

Neural Network	Depth	#Parameters	Accuracy%
ResNet18	18	11M	93.02
ResNet50	50	25.6M	93.62
ResNet100	100	44.5M	93.75
ResNet18*	18	11M	95.08



Activation function

Sigmoid

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

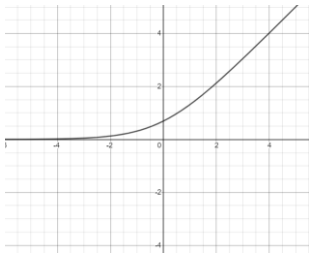
$$\ln(1 + e^{x})$$

Hyperbolic Tangent

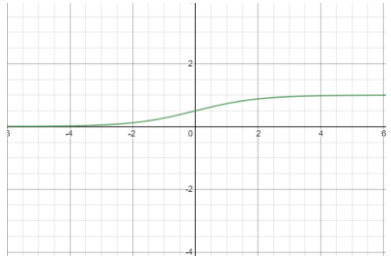
$$y = \tanh(x)$$

$$\ln(e^{-x} + e^{x})$$

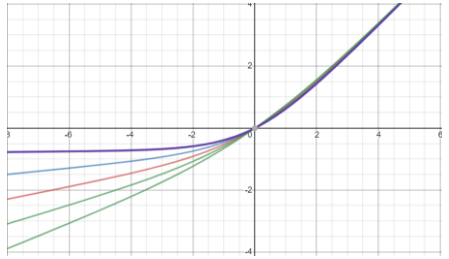
$$D^{a} \ln\left(e^{-bx} + e^{x}\right)$$



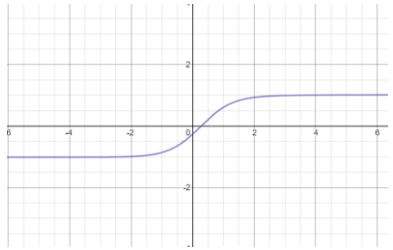
a=0,b=0



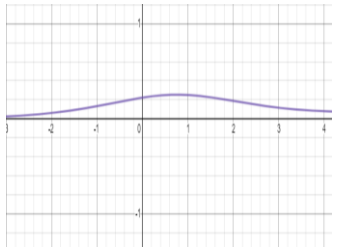
a=1,b=0



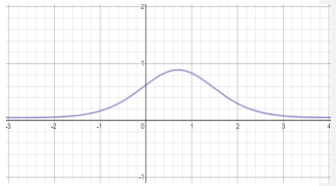
a=a0,b ∈(-0.1,-0.5)



a=1,b=1

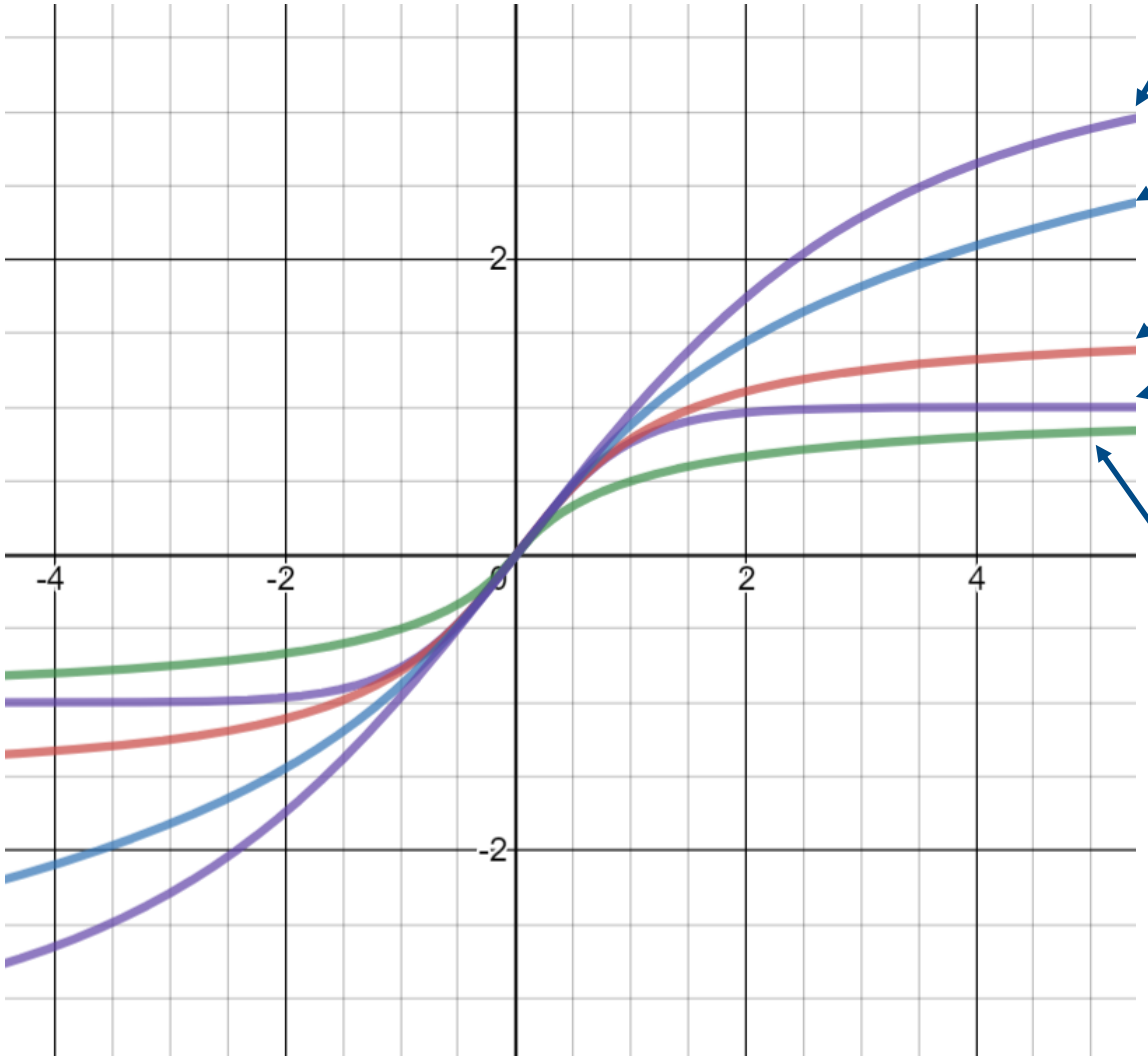


a=2,b=0



a=2,b=1

Family of groups



Approximates this family

$$\frac{x}{\sqrt{1+a_2x^2}}$$

$\sinh^{-1}(x)$ ~a=0.5,b=0.8

$\tan^{-1}(x)$ ~a=0.7,b=0.8

$\tanh(x)$

$$\frac{x}{1+|x|}$$ ~a=1.1,b=1

TABLE III
COMPARING SEVERAL RESNET TOPOLOGIES WITH AND WITHOUT ADAPTIVE ACTIVATION FUNCTION (AAF)

Neural Network	#Param	Original%	With AAF%
ResNet18	0.27M + 1376	91.25	92.74
ResNet34	0.46M + 2272	92.5	94.01
ResNet44	0.66M + 3168	92.83	94.39
ResNet56	0.85M + 4064	93.03	94.44
ResNet110	1.7M + 4960	93.57	94.52

18


intel