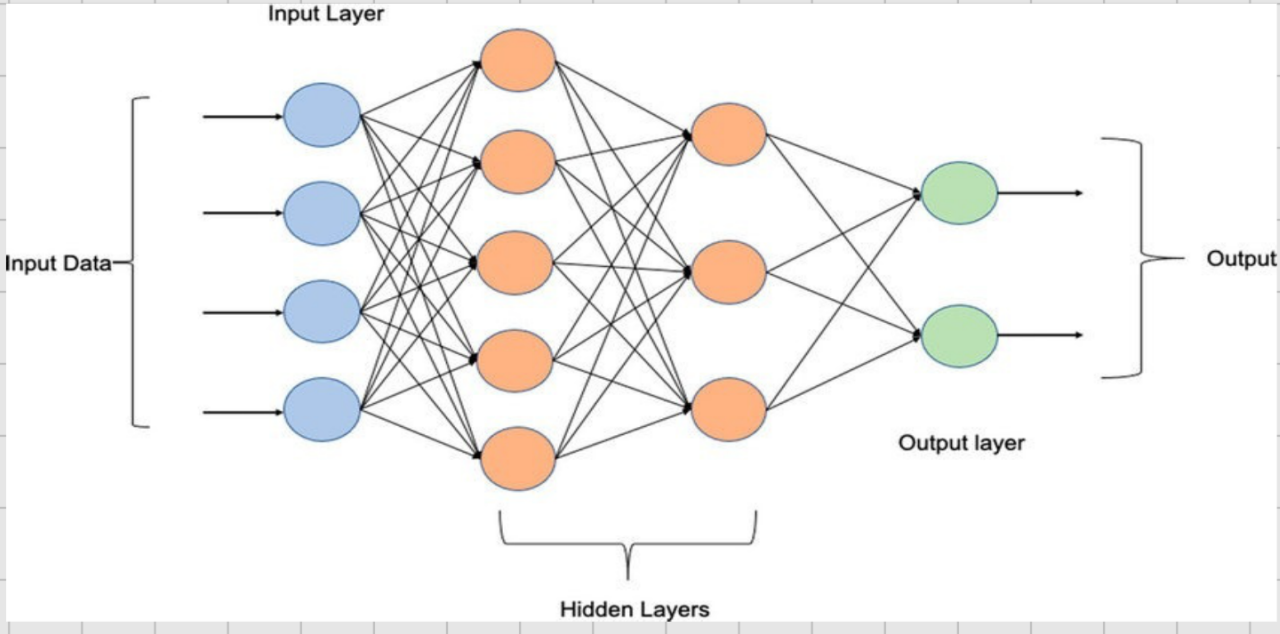
KAN: Kolmogorov Arnold Networks

The new type of network that is making waves in the ML world.

The Kolmogorov-Arnold Network (KAN) is a brand new class of neural network building block. It aims to be more expressive, less prone to overfitting and interpretable than the Multi-layer perceptron (MLP)

Multi-layer perceptron:



$$h_1 = w_1 x + b_1$$

$$f_1 = f(h_1)$$

$$h_2 = w_2 f_1 + b_2$$

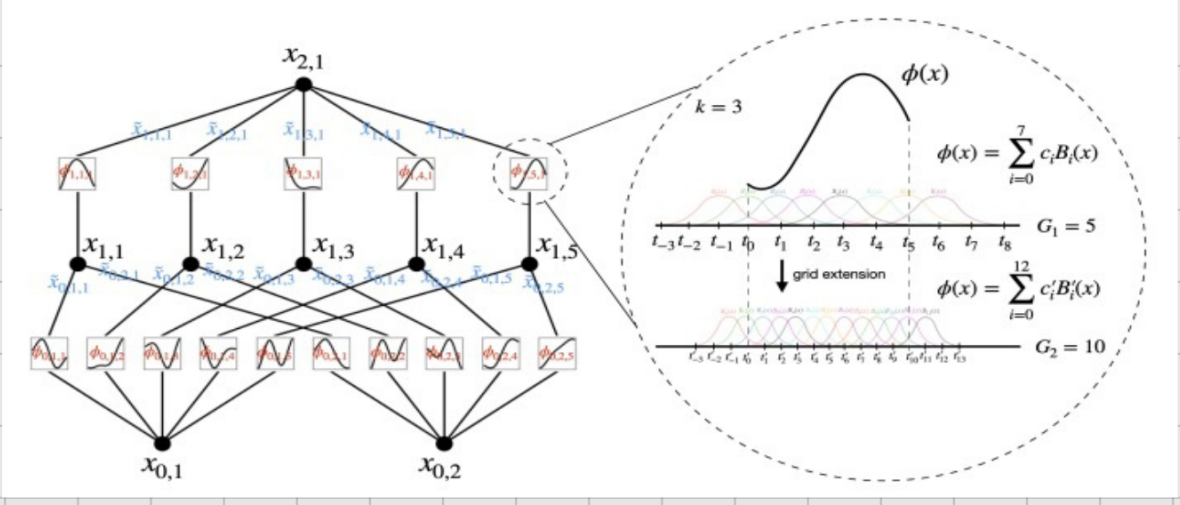
$$y = f(h_2)$$

Most AI model rely on a fundamental concept, (MLP) a mathematical way of mimicking the neurons of human brain. These perceptron have two main parts.

(i) weight ⟶ (which changes as we train the model)
(ii) Activation function ⟶ which never changes.

Instead of using a fixed activation function, KAN use a learnable-activation function.

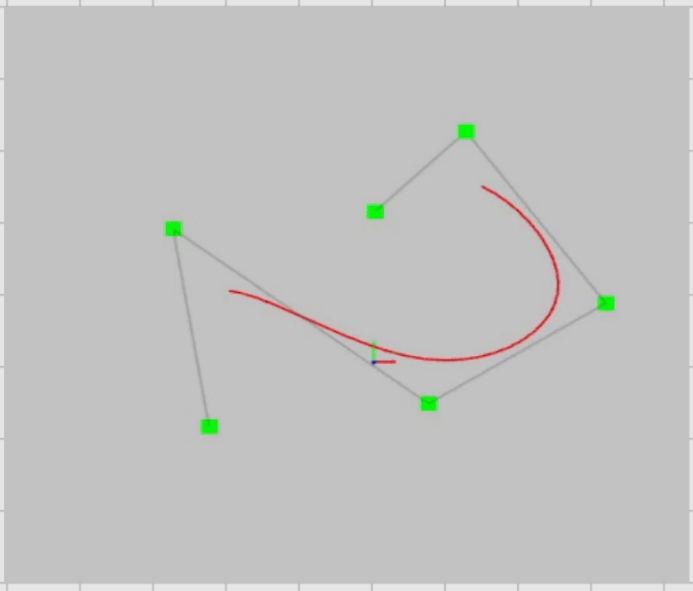
(function that changes we train the model)



Working

In order to make the activation function trainable they used B-Spline which is a piece-wise polynomial function which can be parameterized.

The B-spline parameters are like a series of points that act as magnets, which pulls the curve and changes its shape. This set of point is reffered to as a grid in the paper.



Visualization
changes continuously.
(B-spline)

Key points

* Do more with less parameter, KAN use x100 less parameters than standard MLP to have equal performance, but KAN is x10 slower to train.

* Scability, it potentially scales better than MLP with Kolmogorov-Arnold representation theorem.

* Interpretability, unlike MLP which is like a black box, you can perform symbolic regression and extract Interpretable equations for your Network thanks to the B-spline's properties.

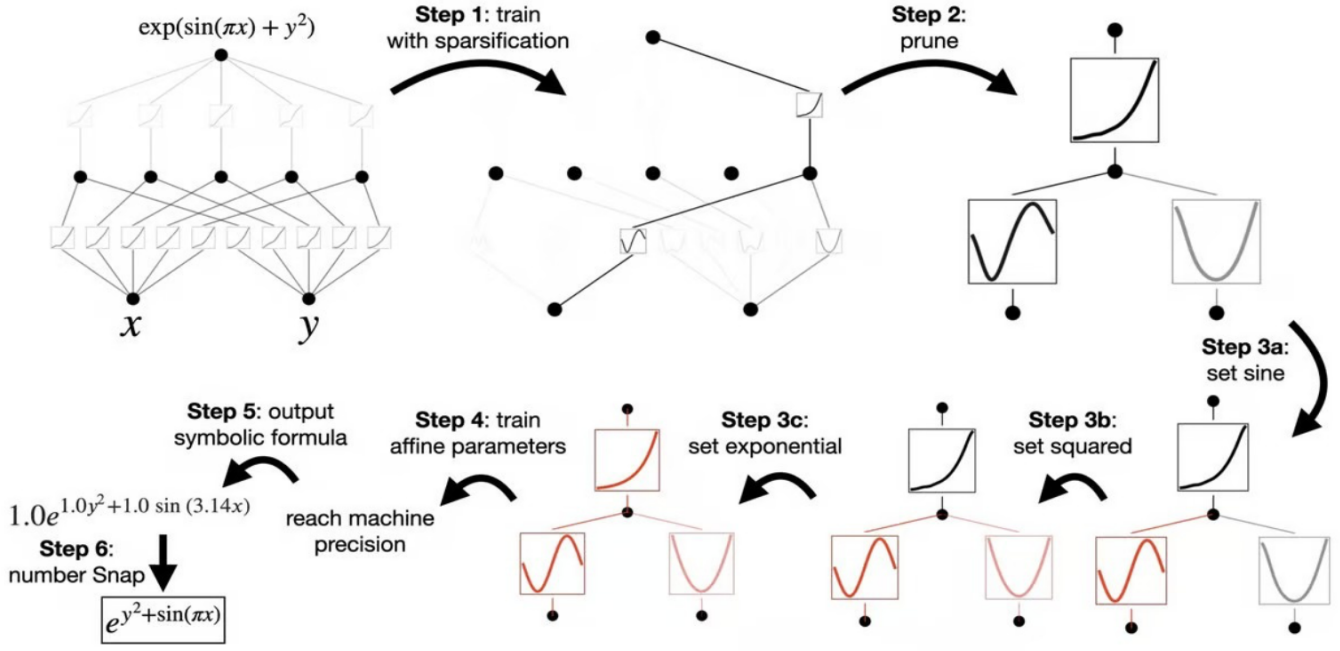
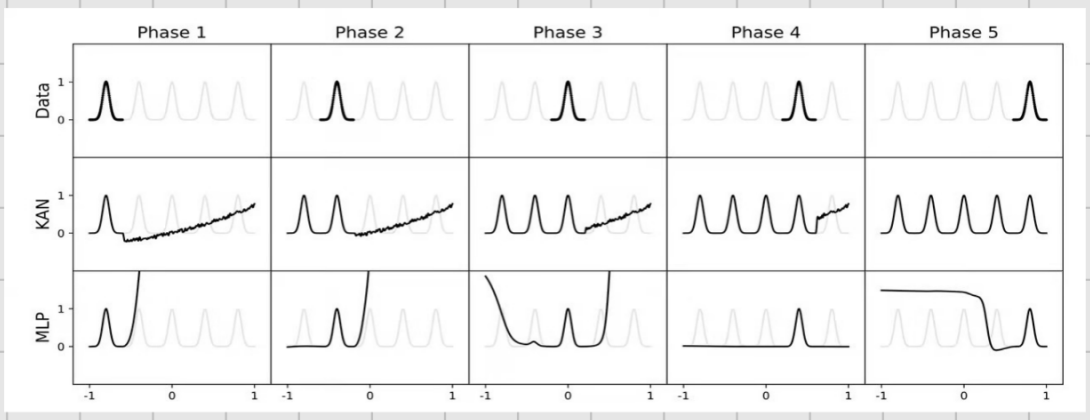


Figure 2.4: An example of how to do symbolic regression with KAN.

Outcomes:

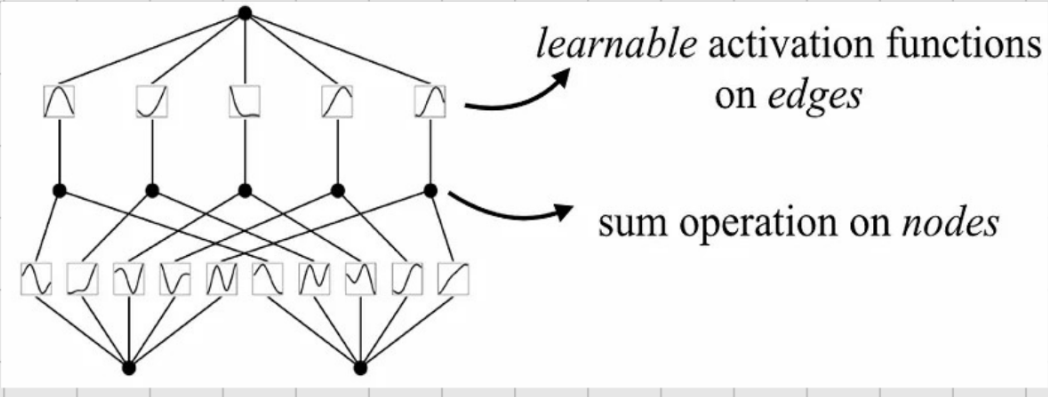
1. For solving partial differential equation (PDE), a 2-layer width-10 KAN is shown to be 100 times more accurate and 100 times more parameter efficient than a 4 layer width MLP.

2. KAN is able to avoid catastropic forgetting in a simple toy case setting



3. moreover, these network are not only applicable to the Machine learning tasks, but they can also rediscover complex relations

4. It seems very difficult to train and the researchers would need more time to figure out the optimal way to train them



KAN with finite grid size can approximate the function well with a residue rate independent of the dimension, hence beting curse of dimensionality.

Method	Architecture	Parameter Count	Accuracy
Deepmind's MLP	4 layer, width-300	3×10^5	78.0%
KANs	2 layer, $[17, 1, 14]$ $(G = 3, k = 3)$	2×10^2	81.6%

KAN converge faster, achieve lower losses, and have steeper scaling laws than MLPs.

10B KAN model = 100B MLP model.

So, 10 times less VRAM usage.

* KAN could easily overfit to training data, especially when there is noise (common for real world)

* KAN may be useful for 3D construction

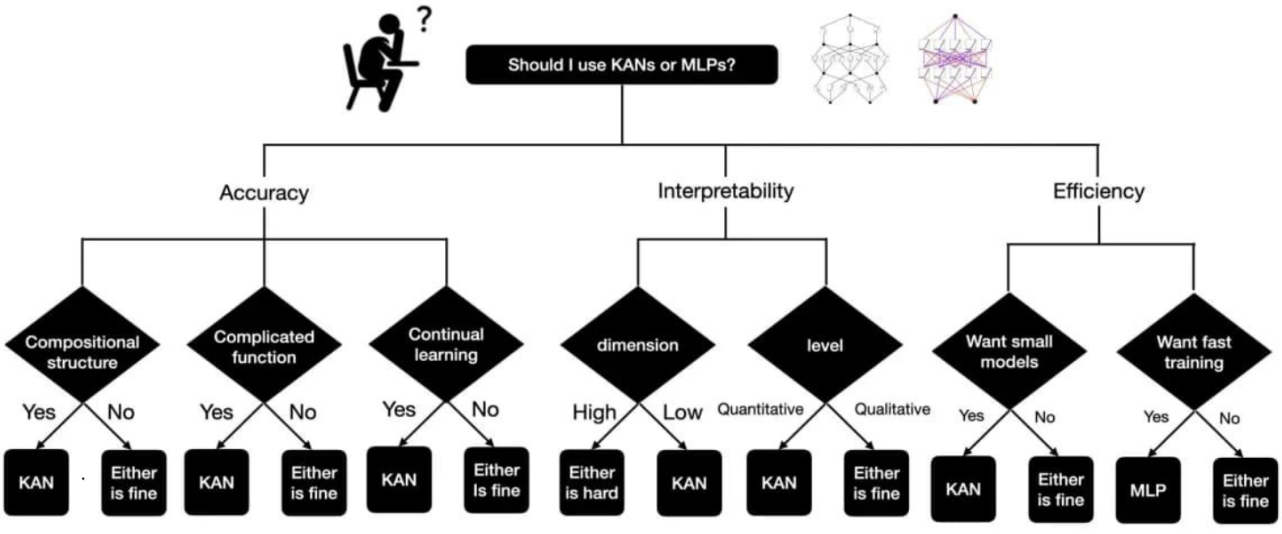


Figure 6.1: Should I use KANs or MLPs?

KAN - GPT-2