CAN WE APPLY KOLMOGOROV-ARNOLD NETWORKS TO IMAGES? PROBLEM DESCRIPTION

Alberto Cereser

May 8, 2024

ABSTRACT

In this report, we explore how Kolmogorov-Arnold Networks could used for processing images. The focus is on the theoretical aspects.

1 Introduction

Kolmogorov-Arnold Networks (KANs) [1] are an alternative to Multi-Layer Perceptrons (MLPs), powered by the Kolmogorov-Arnold representation theorem. Both MLPs and KANs can be described as a network of nodes connected by edges. With MLPs, nodes have a given activation function, and edges have learnable weights. With KANs, the nodes perform sum operations, and edges have learnable activation functions. Compared with MLPs, KANs have the advantage of being more accurate and interpretable.

By design, KANs take as input multivariate continuous functions defined over a bounded domain. Below, we will explore if digital images can be converted into this type of input.

2 Digital images as continuous functions over a bounded domain?

2.1 Preliminaries - 1D case

Let's consider a one-dimensional array consisting of a finite number of natural values. Something like $[4, 7, 1, 2, 7, 1, 6]$. The values (see Fig. 1) can be represented on the Cartesian plane as a set of points, whose x_i is a natural number (position in the array) and y_i is the i-th value of the array. Following Riemann's theory of integration, we can imagine each i-value as the top-middle point of a rectangle of width Δx, with the sum of the area of the rectangles approximating the area subtended by the curve. The intuition behind this is that, given a set of $\{x_i, y_i\}$ values, there is a continuous function f connecting them. The considered vector has a finite length, which makes the function bounded.

In the above section, we considered a monovariate function. The Kolmogorov-Arnold theorem applies to multivariate functions, which we will consider below.

2.2 2D case

A digital image is a 2D array of pixels, for which each discrete set of $\{x, y\}$ coordinates corresponds to a unique color, which can be described for example using the RGB color system.

Considering an image with 256×256 pixels, in formulas we have

$$\forall\{x, y\} : x, y \in \{n \in \mathbb{N} \mid 0 \leq n \leq 256\}, \exists! f : f(x, y) = C_{x,y} \tag{1}$$

This gives us a discrete function, defined over a bounded interval. Extending the reasoning explained above, we can consider each color value as being sampled from a continuous color function. By interpolating the color values we get a continuous, multivariate function defined over a bounded interval, for which the Kolmogorov-Arnold representation

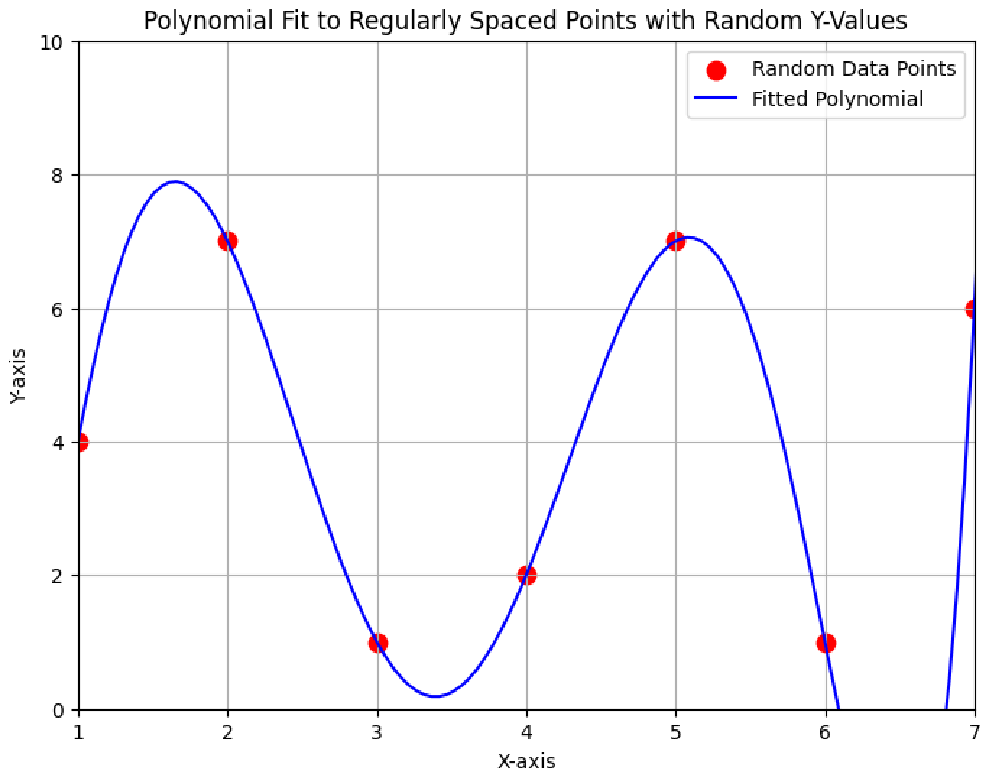

Figure 1: Example of a polynomial curve fitting a set of points, regularly spaced over X.

theorem is valid. As a consequence, the color function can, at least in theory, be processed by a Kolmogorov-Arnold Network. This means that, if the reasoning is correct, the Kolmogorov-Arnold Network can be used to process images.

2.3 2D case - alternative formulation

Alternatively, we can reshape an image as a 1D tensor, in which the value of each tensor element (the color) is a function of the corresponding x and y in the image. By fitting the values as in the 1D case, we will have a multivariate continuous function defined over a bounded interval.

3 Open questions

Points that should be approached and solved, both theoretically and with experiments:

1. What is the best algorithm for fitting a large set of points, placed at regular distance?
2. Is there any issue coming up when the number of points is large?
3. Does the proposed approach have any advantage, compared with what is normally used?

References

[1] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y Hou, and Max Tegmark. Kan: Kolmogorov-arnold networks. *arXiv preprint arXiv:2404.19756*, 2024.