Final Project For ECE228 Track Number #1

Group Number #33: Puning, Zhu Zhaoling, Huang Xihe, Gu Juan, Castillejos

Abstract

Composite materials pose challenges for accurate modeling, often requiring computationally expensive simulations. To address this, a representative volume element (RVE) approach is commonly employed to condense material properties. This study investigates the efficacy of Convolutional Neural Networks (CNNs), Vision Transformers, and Kolmogorov-Arnold Networks (KANs) in predicting homogenized mechanical properties—specifically, Young's modulus and Poisson's ratios—from microscopic images of composite materials and material properties. While CNNs have been the traditional approach for such tasks, emerging architectures like Vision Transformers and KANs prompt a comparative analysis to uncover their respective strengths and weaknesses. We design and train CNNs, Convolutional KANs (CKANs), and a hybrid model, KKan, and Vision Transformers to predict homogenized properties from RVE images, with or without variable material properties, simulating diverse scenarios. Our findings indicate our Vision Transformer architecture exhibits inferior performance compared to CNNs and convolutional KANs, attributed to model complexity and overfitting issues, requiring larger datasets for optimal performance. CNNs and convolutional KANs excel with smaller datasets due to their effective management of model complexity, outperforming simple linear regression models. Additionally, CNNs capture structural nuances effectively through localized receptive fields, while KANs simplify high-dimensional functions into one-dimensional operations, making them well-suited for this data type. Overall, CKANs outperform ViTs and CNNs, but CNNs remain robust choices for capturing and analyzing structural complexities in composite materials.

I certify that I have filled out the evaluation.

1 Introduction

1.1 Composite Material

A composite material is often inconvenient to model since it consists of various components exhibiting multiple mechanical properties, making high-fidelity modeling computationally expensive. To save computational costs, a representative volume element (RVE) is used to condense these material properties. For the dataset used, a recurrent material piece in microscopic view is simplified to a material point with some estimated overall material properties in macroscopic view[4]. This homogenization technique leads to a fictitious homogeneous material that is handy to model.

The homogenized image is shown in figure 1. With the scale only for reference when comparing it to the original image. Our dataset contains 4000 RVE images, 4000 sets of material properties E_0, ν_0, E_1, ν_1, and the corresponding homogenized stiffness components. This project aims to explore and compare the effectiveness of Convolutional Neural Networks (CNNs), Transformers, and a new architecture called Kolmogorov- Arnold Networks (KANs) in predicting the homogenized Young's modulus and Poisson's ratios from images of a composite material.

Preprint. Under review.

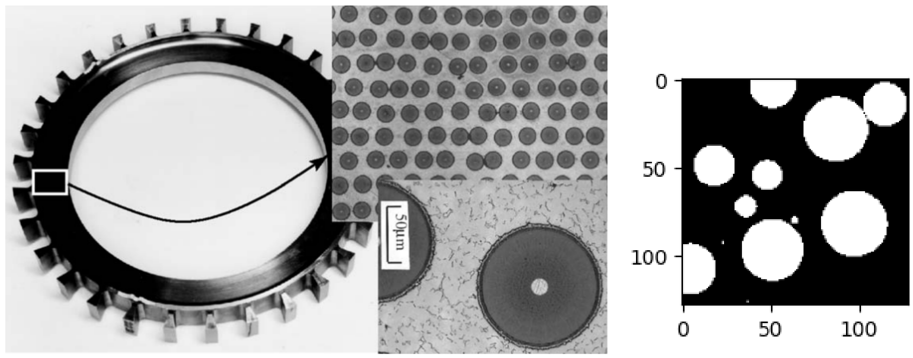


Figure 1: Left: An SiC-Ti ring, composed of two materials. On the lower-right corner is its micro-structure.[4] Right: Example of an RVE input image, sized 128*128.

1.2 Objectives

The conventional method for addressing the homogenized stiffness prediction problem typically relies on CNN. The data set used was of interest because theoretical homogenization methods such as the Mori–Tanaka theory combined with the Finite Element Method [3] result in a problem that can be computationally expensive. The complexity of solving the resulting partial differential equations [7] is one of the main limitations of the traditional methods that don't use machine learning. Furthermore, traditional ML techniques rely largely on the feature engineering which is time-consuming and requires expert knowledge. Our hypothesis is that in addition to CNN,we can use Kolmogorov-Arnold Networks and a Vision Transformer to automatically find the most salient features to be learned and can produce relatively good predictions for the homogenized properties of each RVE. In our study, we seek to contrast the established CNN approach with two emerging methodologies: Vision Transformer and Kolmogorov-Arnold Networks (KANs), which have recently been introduced in the literature, and we aim to provide insights into their respective strengths and weaknesses for solving the homogenized stiffness prediction problem.

Our project aligns partially with track #1, as previous research has utilized CNN to address similar problems [7]. However, the absence of code in the referenced paper requires the development of our implementation. Furthermore, while we intend to leverage PyTorch tools for our implementation, there is a lack of available code demonstrating the application of transformers for predictive tasks. Regarding KANs [5], no native PyTorch functions are readily available for our specific task. Although the original authors have provided some code, KANs have not been previously applied to our problem domain, which makes our project more aligned with the exploratory nature of track #2.

We developed an interest in using transformers because we wanted to incorporate the use of self-attention mechanisms to see if the homogenized properties predictions would improve. In [2] the authors explain that attention is either applied in conjunction with CNNs, or used to replace certain components of CNNs while keeping their overall structure in place. However, in the paper they claim that a pure transformer applied directly to sequences of image patches can perform very well on image classification tasks. For that reason, we used the architecture developed in their paper for our experiment involving the use of a transformer.

2 Related work

2.1 BEM and CNN

Homogenization is a technique used to predict the effective properties of a heterogeneous material by averaging the properties of its constituent phases. A classic approach is based on the boundary element method(BEM), a numerical technique for solving partial differential equations. In a notable study by Okada et al.[6], BEM is adeptly employed to model the micro-structure of heterogeneous materials, laying a foundational framework for subsequent investigations.

Subsequently, another paper proposed a 3D-CNN approach to predict a heterogeneous material's effective properties based on its micro-structural features[7]. The training involves feeding the

3D-CNN with pairs of micro-structural samples and their corresponding homogenized properties, allowing the network to learn the underlying patterns and relationships. Once trained, the 3D-CNN can accurately predict the effective material properties of unseen micro-structures. This approach addresses the limitations inherent in conventional BEM methods, which often rely on computationally expensive finite element simulations or analytical models. The paper discusses the advantages of employing 3D-CNN over traditional finite-element-based homogenization methods, particularly regarding computational efficiency, uncertainty quantification, and the model's transferability.

2.2 KAN

The KAN, or Kolmogorov-Arnold Network, is a type of neural network based on the Kolmogorov-Arnold representation theorem. This theorem states that any multivariate continuous function can be represented as a series of continuous functions of a single variable, which are then added together. This has important implications for neural networks as it suggests a way to build networks that can approximate complex multivariate functions. KANs use this theorem to create network architectures that can effectively handle high-dimensional data with potentially more straightforward and more understandable structures than traditional neural networks.

KAN typically consists of multiple layers, each applying a series of univariate functions and summing up the results. This differs from traditional neural networks, which often use multivariate functions (e.g., matrix multiplications) in each layer. The Kolmogorov-Arnold theorem underpins KANs' versatility, theoretically enabling them to approximate any continuous multivariate function. This theoretical foundation translates into practical applications, making KANs a powerful tool in various scenarios. One key advantage of KANs is their interpretability. KANs can provide more transparent insights into their decision-making process by utilizing activation functions on the edge. This feature is essential in applications where understanding the model's decision-making process is paramount.[5]

2.3 ViT

The Vision Transformer (ViT) architecture is a deep learning model designed to handle image recognition tasks by leveraging the principles of transformers, which are typically used in natural language processing. Unlike CNNs, ViT divides an image into fixed-size patches and treats each patch as a "word" in a sequence, similar to the tokens in a sentence. These patches are then embedded into a high-dimensional space and processed through transformer layers that apply self-attention mechanisms to capture global context and relationships between patches.

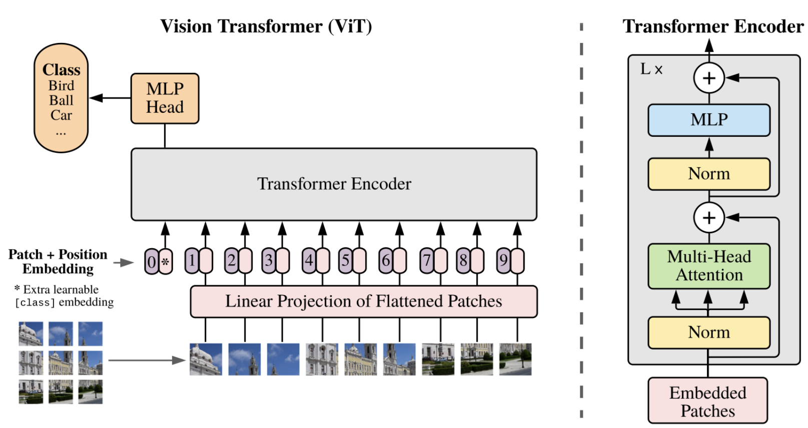


Figure 2: Architecture of ViT as presented in An Image is Worth 16X16 Words: Transformer for Image Recognition at Scale [2]

The original ViT architecture was designed for image classification. In our architecture, we use the MLP head for regression instead of classification. Our goal was that when using images and numerical material properties as input, the model could be trained to learn correlations between the visual features extracted from the images and the material properties. By feeding the images into the ViT to extract feature representations and combining those representations with the numerical properties through additional fully connected layers, we get a prediction that needs little feature engineering.

3 Methodology

We divide our experiments into two parts. In Part 1, we meticulously design and train a CNN, a Convolutional Kolmogorov-Arnold Network (CKAN), and a hybrid model, KKAN. These models are trained to predict homogenized properties using RVE images and corresponding material properties. Additionally, we explore scenarios where we train new CKAN and KKAN models using RVE images, with or without averaged material properties to simulate non-ideal conditions.

In Part 2, our focus shifts to transformers. Here, we design and train a new Vision Transformer (ViT) model with various configurations for predicting homogenized properties using RVE images and variable material properties. Initially, we utilize the structure described in [1] to code the ViT transformer, training it from scratch with our dataset.

This comprehensive approach allows us to thoroughly compare the efficacy of neural networks and transformers in predicting material properties for our dataset.

3.1 Data Pipeline

A standardized data pipeline is employed across most our models. Figure 3 is a flowchart depicting how our models are designed to predict homogenized material properties from 128x128 pixel RVE images. The model processes the image through multiple convolution layers or transformer blocks to extract features crucial for understanding material micro-structure. Additionally, it integrates varied material properties by concatenating these numerical inputs with the image-derived features. This enriched data set then passes through linear layers that map the features to the predicted homogenized properties. For models not utilizing material properties as input, the output of the CNN/C-KAN/ViT layers is directly fed into the MLP architecture to obtain predictions for the homogenized properties. This architecture exemplifies the integration of CNNs, KANs, and Visual Transformer and direct data inputs to predict material properties effectively with little feature engineering, which is crucial in materials science.

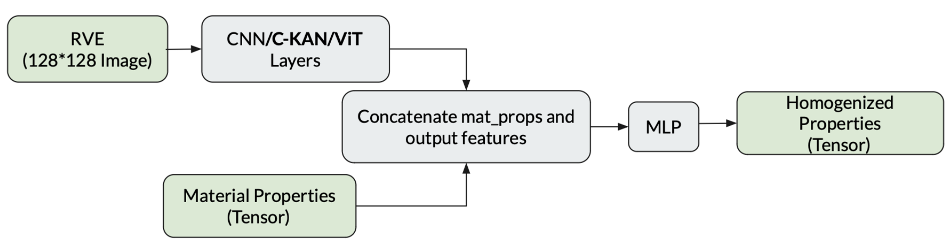


Figure 3: Data pipeline for models with RVE images and material properties as input.

3.2 Neural Network Architectures

The architectures of different neural networks under comparison are given in figure 4.

CNN (Convolutional Neural Network): This baseline model uses standard convolutional layers followed by linear layers to predict the target properties.

C-KAN (Kolmogorov-Arnold Network with convolution): Incorporates KAN convolution layers[8] , which are expected to better capture the complex non-linear interactions in material properties.

KKAN (Advanced KAN model): An extension of the C-KAN model with additional KAN layers[8][9] to enhance the learning capability through more sophisticated data interaction modeling.

(conv1): Conv2d(1, 16, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))	(conv1): **KAN_Convolution_Layer**(1, 16, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))	(conv1): **KAN_Convolution_Layer**(1, 16, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
(conv2): Conv2d(16, 32, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1))	(conv2):**KAN_Convolution_Layer**(16, 32, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1))	(conv2): **KAN_Convolution_Layer**(16, 32, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1))
(fc1): Linear(in_features=8196, out_features=256, bias=True)	(fc1): Linear(in_features=8196, out_features=256, bias=True)	(fc1): **KANLinear**(260, 256)
(fc2): Linear(in_features=256, out_features=3, bias=True)	(fc2): Linear(in_features=256, out_features=3, bias=True)	(fc2): **KANLinear**(256, 3)
(relu): LeakyReLU(negative_slope=0.01)	(relu): LeakyReLU(negative_slope=0.01)	(relu): LeakyReLU(negative_slope=0.01)
(softplus): Softplus(beta=1, threshold=20)	(softplus): Softplus(beta=1, threshold=20)	(softplus): Softplus(beta=1, threshold=20)
(maxpool): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)	(maxpool): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)	(maxpool): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
(dropout): Dropout(p=0.7, inplace=False)	(dropout): Dropout(p=0.7, inplace=False)	(dropout): Dropout(p=0.7, inplace=False)

Figure 4: Architectures of CNN, C-KAN and KKAN

4 Experiments

4.1 CNN and KAN

The main goal of this part was to assess the performance of three different neural network architectures in predicting homogenized stiffness from RVE images, focusing on their accuracy and computational efficiency.

The idea is using different models to process RVE images and extract features for predicting homogenized properties. The C-KAN and KKAN models dynamically incorporated material properties, improving accuracy and reducing model complexity. These models were trained and validated using performance metrics such as normalized mean square error(MSE) and accuracy.

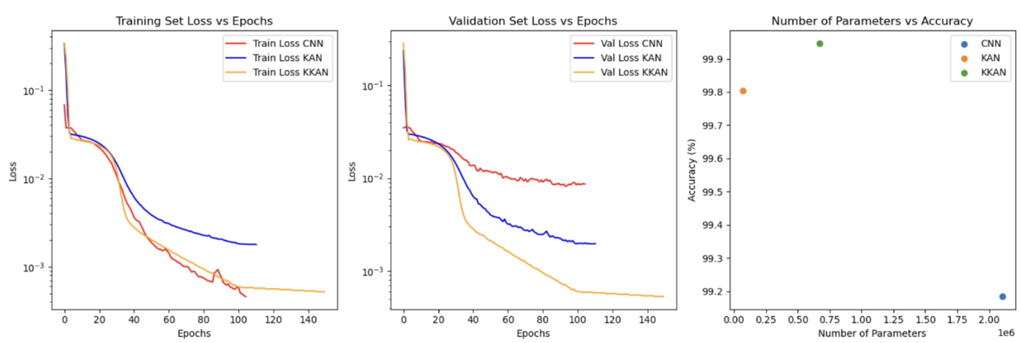


Figure 5: Training, validation loss and accuracy of CNN, C-KAN and KKAN

The results in figure 6 showed that the C-KAN model achieved an accuracy of 99.843% with significantly fewer parameters than the CNN model. In figure 5 is the loss and number of parameters comparison between different NN models. The KKAN model slightly outperformed the C-KAN model in accuracy but with more parameters. The training and validation loss graphs indicated that the KAN-based models converged faster with lower loss values than the CNN, suggesting better efficiency in learning from the training data.

The experiment highlighted that the C-KAN model offered a near-perfect balance between computational resource usage and predictive accuracy, making it a promising candidate for further development. On the other hand, the KKAN model achieved the highest accuracy but with increased model complexity, suggesting diminishing returns with added complexity. This experiment advances the understanding of neural network applications in predicting material properties from RVE images and opens avenues for deploying efficient ML tools in materials engineering and simulation fields.

	Model	Test Accuracy(%)	Num of Parameters
0	CNN	98.728	2104003
1	KAN	99.843	67767
2	KKAN	99.949	673460

Figure 6: Results of CNN, C-KAN and KKAN

Future studies could explore the scalability of these models to larger datasets and more varied material properties to validate their effectiveness and robustness in industrial applications.

4.2 ViT

For the experiments involving the visual transformer, the architecture follows the description in Figure 2. The transformer is trained from scartch. The data is reprocessed to normalize the inputs

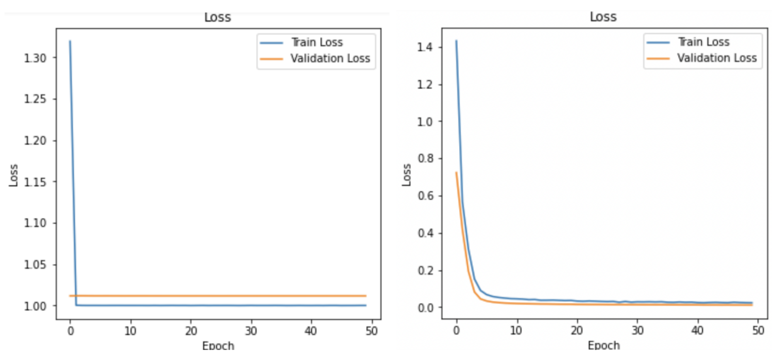


Figure 7: ViT

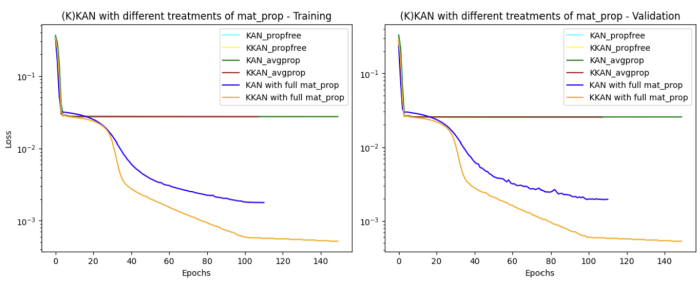
Figure 8: ViT prediction visualization for each homogenized property prediction vs actual value

As described in [2], when pre-trained on large amounts of data, a Vision Transformer (ViT) attains excellent results compared to state-of-the-art convolutional networks benchmarks. This, however, was a drawback for our project. Compared to the 14 million images in ImageNet which [2] calls a mid-sized dataset, our 4000 image dataset is not enough to train our transformer from scratch. Some attempts were made to use google's vit-base-patch16-224 a pre-trained transformer that has an architecture with patch resolution of 16x16 and fine-tuning resolution of 224x224. However, we could not get our implementation to produce any favorable results. We initially attributed this to the fact that the transformer's goal was image classification, but we later became convinced that our implementation was not completely correct as an epoch would take longer than expected.

4.3 Usage of Material Properties

Previous experiments about our KAN and ViT model have shown that they are both applicable to predicting homogenized properties based on given RVE image input and corresponding material properties. For the previous experiment representing ideal cases, Young's modulus and Poisson ratio are both avaliable for the KAN/ViT model as input, but in practise we only have one or neither of them. For instance, the worst case is that we only have images and homogenized properties as train-set, so a model based on sufficiently many RVE images will be needed. And in many industrial cases material properties can only be defined as estimated values instead of precisely matched to a specific RVE image.

In order to simulate these non-ideal conditions, experiments are down for KAN/KKAN/ViT models without material properties and with averaged properties. We simply take the average of the four parameters (E_0, ν_0, E_1, ν_1) from the 4000*4 whole data set. All the results in this section are from the same hyper-parameter set (e.g. step size for lr. reduction).



	Model	Accuracy(%)	Num of Parameters
1	KAN	99.804	67767
2	KKAN	99.947	673460
3	KAN_propfree	97.455	663220
4	KKAN_propfree	97.448	663220
5	KAN_avgprop	97.458	67767
6	KKAN_avgprop	97.456	673460

Accuracy evaluation parameter = 1-minimum loss

Figure 9: (K)KAN performance with different treatment of material properties

The result above shows that training of (K)KAN model without material properties will quickly get stuck in a common sub-optimal solution, with both train and test loss fluctuating slightly instead of continuing to decline. When the model have averaged estimation of material properties as a reference input(which is probably the most common case in industry), the parameters needed to achieve a sub-optimal solution can be reduced to about 1/10. With full data set considered, the convolutional KAN model successfully evolved to get lower loss in order of magnitude after getting stuck for a few epochs, with number of parameters still limited to five-digit levels. The final loss of the KKAN model is even lower but with much less efficiency.

We also tested the ViT model without material properties (left side in Figure 8). The iteration still get stuck after several epochs. But without material properties, both train and test loss of ViT can be more than 10 times larger than the loss of the original ViT model, indicating that missing parameters can have larger effect on the performance of transformer-based models than KAN models. We can tell that (K)KAN model performs much better than ViT in training with an incomplete data set.

5 Conclusion

In summary, in our context, Vision Transformers (ViT) have shown inferior performance compared to CNNs and convolutional Kolmogorov-Arnold Networks (KANs). This difference can mainly be attributed to model complexity and over-fitting issues. ViT generally requires a larger dataset to perform optimally. At the same time, CNNs and convolutional KANs excel with smaller datasets by effectively managing model complexity, as demonstrated by their improved performance over simple linear regression. To address this limitation for ViT, using pre-trained models could enhance its effectiveness by compensating for the smaller dataset. Also, regarding data characteristics, CNNs are better suited to capture the structural nuances of composite materials through their localized receptive fields. On the architectural front, KANs have a distinct advantage. They are designed to simplify complex, high-dimensional functions into simpler, one-dimensional operations, making them particularly suitable for physical applications. Conversely, ViT segments images into fixed-size patches for processing. While this approach enables attention mechanisms to identify inter-dependencies among patches, it proves less effective in scenarios with limited data availability. This limitation arises because ViT relies heavily on substantial datasets to learn meaningful patterns from the segmented patches, making it less suited for smaller datasets where such comprehensive learning is constrained.

6 Git

link: https://github.com/pzhu301/ECE-228-Final-Project.git

References

[1] S. J. Callis. Vision transformers, explained, 2024. URL `https://towardsdatascience.com/vision-transformers-explained-a9d07147e4c8`. Accessed: 2024-05-25.

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[3] Y. Hua and L. Gu. Prediction of the thermomechanical behavior of particle-reinforced metal matrix composites. *Composites Part B: Engineering*, 45(1):1464–1470, 2013. ISSN 1359-8368. doi: https://doi.org/10.1016/j.compositesb.2012.09.056. URL `https://www.sciencedirect.com/science/article/pii/S1359836812006063`.

[4] P. Kanouté, D. Boso, J.-L. Chaboche, and B. Schrefler. Multiscale methods for composites: a review. *Archives of Computational Methods in Engineering*, 16:31–75, 2009.

[5] Z. Liu, Y. Wang, S. Vaidya, F. Ruehle, J. Halverson, M. Soljačić, T. Y. Hou, and M. Tegmark. Kan: Kolmogorov-arnold networks. *arXiv preprint arXiv:2404.19756*, 2024.

[6] H. Okada, Y. Fukui, and N. Kumazawa. Homogenization method for heterogeneous material based on boundary element method. *Computers & Structures*, 79(20-21):1987–2007, 2001.

[7] C. Rao and Y. Liu. Three-dimensional convolutional neural network (3d-cnn) for heterogeneous material homogenization. *Computational Materials Science*, 184:109850, 2020. ISSN 0927-0256. doi: https://doi.org/10.1016/j.commatsci.2020.109850. URL `https://www.sciencedirect.com/science/article/pii/S0927025620303414`.

[8] A. Tepsich. Convolutional-kans, 2024. URL `https://github.com/AntonioTepsich/Convolutional-KANs`. Accessed: 2024-06-11.

[9] K. Xiaoming. pykan, 2024. URL `https://github.com/KindXiaoming/pykan.git`. Accessed: 2024-06-11.