KAN-Mixer: Kolmogorov-Arnold Networks for Gene Expression Prediction in Plant Species

Jin Gao[1], Juntu Zhao[1⋆], Keyu Li[1⋆], and Dequan Wang[1,2†]

[1] Shanghai Jiao Tong University
[2] Shanghai Artificial Intelligence Laboratory

Abstract. Understanding the intricate relationships between cis-regulatory elements and gene expression is crucial for decoding genetic regulation in ecological systems. In this study, we introduce a novel application of Kolmogorov-Arnold Networks (KANs) for predicting gene expression across diverse plant species, including Arabidopsis thaliana, Solanum lycopersicum, Sorghum bicolor, and Zea mays. Our model, named KAN-Mixer, utilizes k-mers as inductive biases to capture biologically relevant patterns in nucleotide sequences. By employing token embeddings and mixer architectures, KAN-Mixer enhances both the interpretability and usability of KANs. Our results indicate that KAN-Mixer achieves comparable accuracy to ConvNet-based approaches while offering superior interpretability, making it a robust tool for ecological data analysis in a variety of environments.

Keywords: KAN · Genetic Regulation · Ecology

1 Introduction

Gene expression regulation is orchestrated by complex interactions between proteins and nucleic acids, particularly DNA and RNA. A key element in this regulation is cis-regulatory elements (CREs), short DNA sequences near a gene's genomic region recognized by transcription factors. At the transcript level, RNA processing is regulated by nucleotide sequences binding protein factors, forming a sophisticated gene regulatory network. Understanding these regulatory mechanisms is essential for comprehending ecological dynamics across diverse environments.

Current experimental molecular biology techniques often provide a limited understanding of these nucleotide codes due to their reductionistic nature. Consequently, a holistic approach, such as deep learning, can offer substantial insights into the plant gene regulatory code. With the increasing availability of genomic data, deep learning methods can more effectively annotate and functionally characterize CREs. Systematic investigations of sequence-to-regulation relationships across various plant species and regulatory domains are essential, as genome-scale identification and annotation of cis-regulatory sequence features remain largely unexplored, especially considering the diverse ecological systems ranging from tropical forests to Arctic tundras, and urban to rural settings.

⋆ Equal contribution. † Corresponding author.

Deep learning applications in exploring gene regulatory networks, particularly for annotating and functionally characterizing CREs, require a nuanced understanding of neural network architectures, their inductive biases, and interpretability. Multi-layer perceptrons (MLPs) [12], known for their universal approximator capability, excel at learning complex representations but struggle with tasks involving spatial or temporal dependencies, such as identifying CREs in genomic sequences. Their minimal inductive bias requires significant data to uncover relevant features, often leading to overfitting and making the models challenging to interpret.

Inspired by the Kolmogorov-Arnold representation theorem [14] [3], Kolmogorov-Arnold Networks (KANs) [19] replace the fixed activation functions of MLPs with learnable activation functions on edges, parameterized as splines. This change allows KANs to outperform MLPs in both accuracy and interpretability. Both theoretically and empirically, KANs exhibit faster neural scaling laws compared to MLPs, suggesting a more efficient learning process. For interpretability, KANs offer intuitive visualization and easy interaction, which is particularly beneficial in scientific research. However, to ensure interpretability, the overly long gene sequences must be flattened and input into the KANs, which introduces a significant parameter burden. This has become a major obstacle hindering the widespread application of KANs in the processing of genetic data.

In this paper, we demonstrate the interpretable use of KANs for gene expression prediction in four plant species *Arabidopsis thaliana*, *Solanum lycopersicum*, *Sorghum bicolor*, and *Zea mays*. We present KAN-Mixer, a straightforward yet interpretable model for predicting gene sequence-expression relationships. Using k-mers as an inductive bias for gene sequence analysis is analogous to applying convolutional filters in convolutional neural networks (ConvNets) for visual recognition. K-mers, which are sequences of fixed length k, encapsulate significant biological signals by assuming that closely positioned nucleotides and recurring patterns hold important biological relevance. Specifically, we propose token embeddings for gene sequences and constrain the modeling of the KANs at the token level, enhancing the usability of KANs while preserving interpretability.

Inspired by the MLP-Mixer architecture [24], we utilize mixer architectures to facilitate communication between different channels and tokens. KAN-Mixer emphasizes specific sequence motifs and their hierarchical organization, offering more insightful and explainable results compared to MLPs and ConvNets. This aligns with the ecological objectives of genomics research, where understanding the underlying mechanisms and functional implications of CREs necessitates models that provide clear and biologically interpretable insights. Our experiments demonstrate that KAN-Mixer is at least as effective as existing ConvNet-based baselines [21], with the added benefit of enhanced interpretability.

By providing a robust and interpretable model for gene expression prediction, KAN-Mixer contributes to the broader goal of advancing ecological research through the integration of cutting-edge computer vision techniques. Understanding gene regulatory mechanisms across diverse plant species not only enhances our knowledge of fundamental biological processes but also informs conservation strategies, agricultural practices, and ecological management. This work highlights the potential for deep learning models to bridge the gap between complex genetic data and practical ecological applications, fostering a deeper connection between the fields of computer vision and ecology.

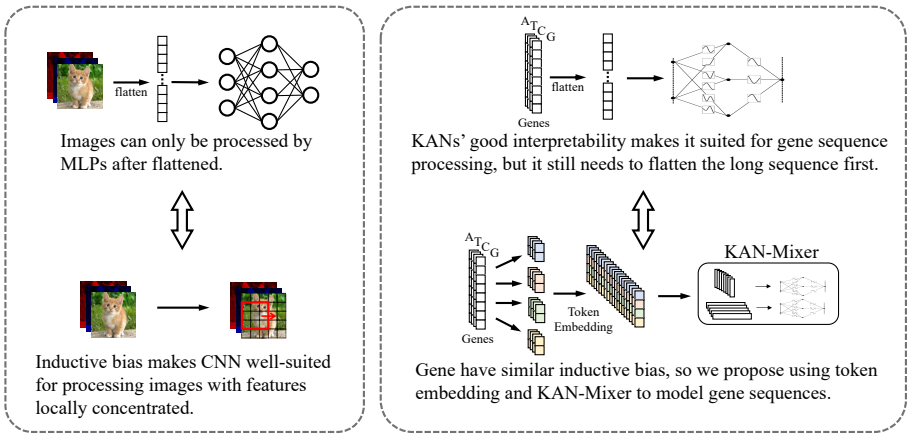


Fig. 1: KANs' notable interpretability makes it suitable for gene sequence processing, yet it struggles with extremely long gene sequences due to significant parameter burdens and performance declines. Inspired by convolutional inductive biases, we introduce KAN-Mixer with Token Embedding to enhance KANs usability in gene sequence analysis without compromising interpretability.

2 Related Work

MLPs & KANs Multi-layer perceptrons (MLPs) [7, 11, 12, 25] are widely recognized as fundamental components of contemporary deep learning models. Despite their prevalent use, MLPs have significant limitations, particularly in terms of interpretability [6]. Recently, Kolmogorov-Arnold Networks (KANs) [18, 19] have been introduced, addressing this issue by employing learnable activation functions directly on the network's edges. These functions are formulated as splines based on the Kolmogorov-Arnold representation theorem [3, 14, 15, 23]. This structural innovation eliminates the need for linear weights and enhances interpretability, allowing scientists to directly identify critical components within the input data. This feature makes KANs particularly valuable for aiding new scientific discoveries. However, in the field of biology, the application of KANs presents unique challenges due to the extremely high dimensionality of data, such as gene sequences. The substantial length of these data imposes a significant parameter burden on KANs. Although recent efforts have focused on improving the usability and computational efficiency of KANs, significant challenges remain in their application within biological domains. In this paper, we aim to improve the usability of KANs in modeling gene sequences without compromising their interpretability and performance.

Gene Sequence-Expression Prediction Although deep learning models often yield accurate results, they frequently lack interpretability, a critical shortcoming in many scientific applications. For instance, in the task of gene sequence-expression prediction, biologists require not only precise predictions but also an understanding of which specific gene segments play key roles, providing insights that lead to new scientific discoveries. In this paper, we specifically focus on the relationship between non-coding

regulatory element sequences and gene expression, a fundamental task for understanding gene regulation. Previous work [21] employed ConvNet-based approach to predict gene expression in various plant species. Our approach seeks to enhance predictive effectiveness with deeper biological insights.

3 KAN-Mixer

In this section, we detail our method, KAN-Mixer, to deal with the challenges posed by KANs in processing long gene sequences. As shown in Figure 3, we introduce a token embedding layer at the beginning of the model to extract features, with interpretability ensured by the k-mers characteristics of gene sequences. Subsequently, we innovatively incorporate KAN-Mixer to simultaneously focus on both channel and token information within the tensor.

3.1 Premilinary

KAN is predicated on the Kolmogorov-Arnold Representation Theorem (Equation 1) with learnable activation functions on edges and summation operations on nodes. This theorem postulates that any multivariate continuous function on a bounded domain can be expressed as a finite composition of continuous functions of a single variable and the binary operation of addition.

$$f(\mathbf{x}) = \sum_{q=1}^{2n+1} \Phi_q \left(\sum_{p=1}^{n} \phi_{q,p}\left(x_p\right) \right) \tag{1}$$

for a smooth $f : [0,1]^n \to \mathbb{R}$, $\phi_{q,p} : [0,1] \to \mathbb{R}$ and $\Phi_q : \mathbb{R} \to \mathbb{R}$.

KAN uses B-spline basis functions (Equation 2) to approximate $\phi_{q,p}$ and Φ_q. Although KAN exhibits competitive accuracy and superior interpretability compared to MLP, its computational speed is notably slower. To mitigate this deficiency, we replace the B-spline basis functions (Equation 2) with *Gaussian Radial Basis Functions* (RBF) (Equation 4) [2, 4, 10], following the same rationale as in [18]. These RBFs accurately approximate the B-spline basis (in Figure 2) and are computationally efficient.

$$B_{i,0}(x) = \begin{cases} 1 & \text{if } x_i \le x < x_{i+1}, \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$B_{i,p}(x) = \frac{x - x_i}{x_{i+p} - x_i} B_{i,p-1}(x) + \frac{x_{i+p+1} - x}{x_{i+p+1} - x_{i+1}} B_{i+1,p-1}(x) \tag{3}$$

$$R_i(x) = \exp\left(- \left(\frac{x - x_i}{h} \right)^2 \right) \tag{4}$$

Except using RBF instead of B-spline basis function, *LayerNorm* (Equation 5) [1] is also applied compared to the original KAN. LayerNorm is placed before RBF transform after the input in order to scale inputs to the range of spline grids.

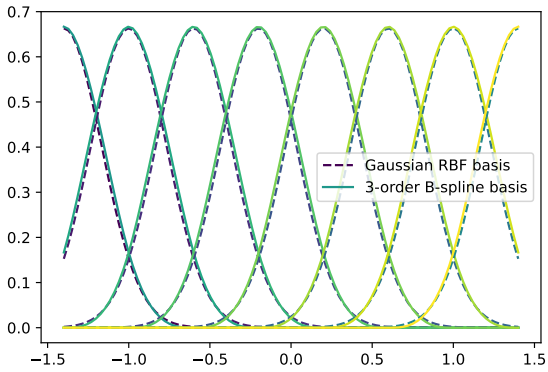


Fig. 2: Gaussian Radial Basis Functions can well approximate 3-order B-Spline basis.

$$\mathrm{L}(x) = \frac{\mathrm{x} - \mathrm{E(x)}}{\sqrt{\mathrm{Var(x)} + \epsilon}} * \gamma + \beta \tag{5}$$

$$\phi_{l,i,j}(\cdot) = R_{l,i,j}(\mathrm{L}(\mathbf{x}_l), g\_min, g\_max, num\_g) \tag{6}$$

where g_min, g_max, num_g determines the grid by changing the range and the smoothness of the RBF basis.

A KAN with layers $[n_0, n_1, \cdots, n_L]$ can be expressed as $\mathrm{KAN}(\mathbf{x}) = (\boldsymbol{\Phi}_L \circ \cdots \boldsymbol{\Phi}_2 \circ \boldsymbol{\Phi}_1)(\mathbf{x})$ with Φ_l shape $[n_{l+1}, n_l]$. The operation from layer n_l to layer n_{l+1} can be represented as

$$\mathbf{x}_{l+1} = \underbrace{\begin{pmatrix} \phi_{l,1,1}(\cdot) & \phi_{l,1,2}(\cdot) & \cdots & \phi_{l,1,n_l}(\cdot) \\ \phi_{l,2,1}(\cdot) & \phi_{l,2,2}(\cdot) & \cdots & \phi_{l,2,n_l}(\cdot) \\ \vdots & \vdots & & \vdots \\ \phi_{l,n_{l+1},1}(\cdot) & \phi_{l,n_{l+1},2}(\cdot) & \cdots & \phi_{l,n_{l+1},n_l}(\cdot) \end{pmatrix}}_{\boldsymbol{\Phi}_l} \mathbf{x}_l, \tag{7}$$

where $\boldsymbol{\Phi}_l$ is the function matrix corresponding to the l^{th} KAN layer, $\mathbf{x}_l$ is the input from the l^{th} KAN layer with shape $[n_l, 1]$ and $\mathbf{x}_{l+1}$ is the output from the $(l+1)^{\mathrm{th}}$ KAN layer with shape $[n_{l+1}, 1]$. The input layer data undergoes layer normalization to transform it into the range of RBF spline grids. Subsequently, the RBF basis is computed based on this normalized data, serving as an operator. These RBF basis-derived operators are then learned through a MLP layer with solely trainable weight parameters. This process constitutes the learning of the $\boldsymbol{\Phi}_l$ component.

3.2 Token Embedding

The utilization of k-mers [13, 22] enables the extraction of fixed-length features from gene sequences, analogous to employing fixed-size filters in image processing. This method aligns with token-level analysis, reflecting the principles of k-mers in identifying critical parts of gene sequences.

Convolutional neural networks (ConvNets) [17] possess inherent inductive biases that enhance learning efficiency [5, 8, 26], particularly through locality and translation invariance. These biases allow ConvNets to exploit spatial hierarchies in data, making them highly effective for image recognition tasks. The architecture of ConvNets enables the learning of local features through convolutional layers, making them particularly suitable for tasks involving spatial hierarchies, such as sequence analysis in genomics. In this context, ConvNets can effectively capture local dependencies within nucleotide sequences, identifying regulatory motifs that are crucial for gene expression regulation. The weight-sharing mechanism in ConvNets reduces the number of parameters, enhancing both generalization and interpretability, as the learned filters often correspond to biologically meaningful motifs.

Our model incorporates a Patch Embedding layer at the outset to ensure input interpretability. This layer transforms one-hot encoded gene sequence data into a format that enhances the model's ability to extract and interpret meaningful features from complex biological sequences. We employ a one-dimensional convolutional layer with both the kernel size and stride set to 8. This configuration allows us to analyze the importance of each input sequence segment in increments of 8 units. By balancing granularity and comprehensibility, we can precisely identify segments that significantly impact the model's predictions. This choice maintains the clarity of the model's decision-making process and aligns with our goal of enhancing the interpretability of deep learning models in genomic research.

3.3 Mixer Architecture

To extract richer features, our token embedding layer transforms the original 4-channel one-hot representation of the gene sequence into 128 channels. This transformation requires subsequent processing to calculate feature relationships not only between different tokens but also across different channels. Inspired by the MLP-Mixer [24], we leverage KANs to both token mixing and channel mixing processes, introducing our method, KAN-Mixer. Each block in KAN-Mixer contains two distinct KANs, dedicated to token mixing and channel mixing, respectively.

Token Mixing with KANs We replace the original fully connected layers with KANs to leverage RBF approximation. This block initially expands the dimensionality of the token representation to a higher-dimensional feature space, providing the necessary operational space for KANs to learn and integrate features across a broader functional landscape.

Channel Mixing with KANs Similarly, we utilize KANs to process the embedding dimensions across different input channels. This block maintains the original dimensionality but focuses on optimizing inter-channel feature integration directly within the embedding space.

The input first undergoes processing through the Token Mixing block, where features among different tokens (patches) are manipulated and integrated. The output from this block is combined with the input via a residual connection, enhancing information retention and gradient flow across the network. This combined output is then fed into

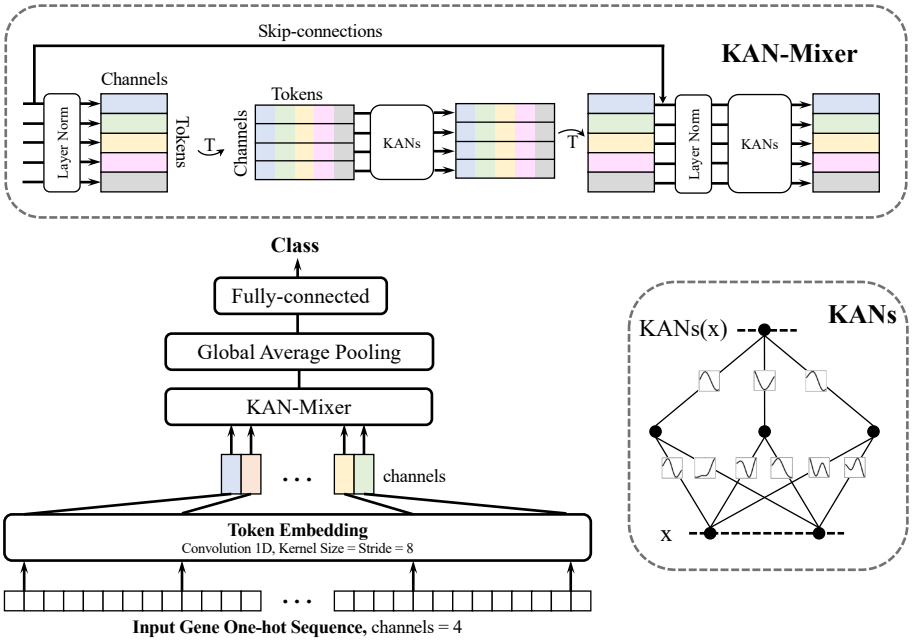


Fig. 3: Structure of our Method. We introduce a token embedding layer at the beginning of our model, setting the smallest modeling unit as a short genetic sequence. To enhance the learning of relationships between adjacent tokens, we innovatively employ the KAN for both token mixing and channel mixing, collectively forming the KAN Mixer.

the Channel Mixing block, where features are mixed at the channel level, enabling a comprehensive amalgamation of learned representations across different feature channels. In our experiments, for a vanilla KAN with layer dimensions [3020×4, 128, 2], KAN-Mixer reduces the parameter count from 13.94 million to 1.44 million, demonstrating its improved applicability and lighter parameter load.

3.4 Exploration on CIFAR

We first conducted a comparative study between KAN-Mixer and MLP-Mixer [24] on tasks simpler than gene sequence-expression prediction. Specifically, we performed image classification on the widely used CIFAR-10/100 datasets [16], which consist of 60,000 images divided into 10 and 100 classes, respectively. To ensure a fair comparison, we configured KAN-Mixer with a similar number of parameters as the tiny-sized MLP-Mixer. The experimental results are presented in Table 1.

In comparison to the state-of-the-art all-MLP architecture, MLP-Mixer, KAN-Mixer demonstrates comparable accuracy in classifying small images. The CIFAR-10/100 experiments were crucial for fine-tuning KAN-Mixer 's hyperparameters, *e.g.*, learning rate, normalization, weight decay, stride, layer number, and so on. As CIFAR-10/100 is a standard dataset in deep learning with extensive research, it provides an ideal benchmark

	CIFAR-10	CIFAR-100
MLP-Mixer [24]	89.8	70.1
KAN-Mixer (Ours)	91.2	70.2

Table 1: Our method, KAN-Mixer, achieves performance comparable to the MLP-Mixer on CIFAR-10/100 datasets. Accuracy, measured as a percentage, is used to evaluate performance, with higher values indicating better outcomes.

for identifying issues in KAN-Mixer. We train our model on the training set and select the suitable hyperparameters based on the test set. Moreover, since MLP-Mixer is a vision model, performing CIFAR-10/100 classification tasks allows for straightforward apples-to-apples comparisons.

During our experiments, we found that optimizing splines (using RBF) is a crucial factor influencing both the performance and stability of KAN-Mixer. Additionally, KAN exhibits a distinct advantage in interpretability, enabling the direct identification of key regions in the input images. Consequently, we extended our study to address the high-dimensional gene sequence expression challenge, thereby highlighting the interpretability of KAN-Mixer.

4 Experiments

We first introduce the experimental setting concerning gene data, models, and metrics for evaluation in Section 4.1. Besides, we report an overall comparison between KAN-Mixer and baselines in Section 4.2. Finally, we provide the visualization of KAN-Mixer and provide some interpretations in Section 4.3.

4.1 Settings

Dataset Following the same dataset setting[3] of the [21], we utilize the genome assemblies and annotations from the Ensembl Plants database (v52) [9]. These sequences span $500-3000$ nucleotides upstream and $100-700$ nucleotides downstream of the transcription start site (TSS), and $100-700$ nucleotides upstream and $500-3000$ nucleotides downstream of the transcription termination site (TTS). We encoded the gene flanking sequences using one-hot encoding, resulting in data shaped (3020, 4).

Labels are derived by estimating the log-transformed transcript per million values (logMaxTPM) and categorizing them as low, medium, or high, based on the lower and upper quartiles of the logMaxTPM distribution. We specifically target genes that fall below the lower quartile or above the upper quartile of this distribution. Consequently, the task is framed as a binary classification problem, aiming to predict genes as either low (below the lower quartile) or high (above the upper quartile).

[3] `https://github.com/NAMlab/DeepCRE`

Model The model is structured into four key components: a patch embedding layer, a KAN-Mixer layer, an average pooling layer, and a linear head for binary classification. For a fair comparison with the baseline [21], the patch embedding layer extracts an "8-mer" feature by employing a 1D convolution with both a patch size and a stride of 8. This results in $378-$dimensional tokens that are subsequently processed by the KAN-Mixer layer, which has a feature dimension of 128, matching that of the final linear head. Notably, the configuration of the KAN-Mixer layer here differs from that used in our CIFAR experiments. Detailed code will be released soon. For optimization, we employ an Adam optimizer with a learning rate of 5e-4 and a weight decay rate of 0.1 for the spline linear layer. Additional details are available in the appendix.

The baseline model, as detailed in the original paper by Peleke et al. [21], comprises three blocks and an MLP. Each block contains two convolutional layers, a max pooling layer, and a dropout layer. Further specifications are provided in the appendix. Notably, the original study used the test set to adjust the learning rate and implement early stopping, and considered the best training accuracy as the final result [4]. We address these methodological flaws by eliminating any test set exposure during training and limiting the training duration to 5 epochs, as both models demonstrate rapid convergence. Additionally, for comparison with KAN-Mixer, our baselines include both traditional KANs and MLPs with layer dimensions of $[3020 \times 4, 128, 2]$.

Metrics There are two settings in our experiment, *single species* (in-domain), and *cross species* (out-of-domain). We conduct experiments on four plant species, *Arabidopsis thaliana*, *Solanum lycopersicum*, *Sorghum bicolor*, and *Zea mays*.

In the **single species (in-domain)** task, each species had two tissue types examined, root and leaf. To testify to the validity of our models and baselines, we take chromosomal-level cross-validation, in which for each iteration, genes located on one of the chromosomes were used as a validation set and the rest for training. In other words, for a given chromosome within a specific tissue and species, all other chromosomes from that tissue and species constituted the training set, while that chromosome served as the validation set.

In the **cross species (out-of-domain)** task, for a particular chromosome within a specific tissue and species, all other species with the same tissue type formed the training set, and that chromosome functioned as the validation set.

The metrics evaluated are validation accuracy and the area under the receiver operating characteristic curve (AUC-ROC), which is a performance metric used to evaluate the ability of a binary classification model to distinguish between classes, representing the likelihood that a randomly chosen positive instance is ranked higher than a randomly chosen negative one.

4.2 Results

In-domain Experiments As evidenced by the results presented in Table 2, KAN-Mixer achieves highly competitive performance compared to the baseline method across various

[4] Details can be found in the official codebase.

Method	Zea		Sor		Ara		Sol		Average	
	Acc↑	AUC-ROC↑	Acc↑	AUC-ROC↑	Acc ↑	AUC-ROC ↑	Acc↑	AUC-ROC ↑	Acc ↑	AUC-ROC ↑
Vanilla-MLP	75.1	79.3	73.0	75.5	76.8	83.0	78.0	81.6	75.7	79.5
Vanilla-KAN	67.5	66.9	61.9	66.1	67.9	73.0	69.4	71.1	66.7	68.8
KAN-Mixer	76.4	82.5	75.5	81.5	78.9	87.2	79.5	85.5	77.5	83.8
ConvNet [21]	77.7	83.8	76.6	81.9	73.2	80.1	80.2	86.2	77.6	83.6

Table 2: Performance comparison on in-domain plant datasets. Our method KAN-Mixer achieves comparable accuracy (Acc) and AUC-ROC scores to the baseline across different plant datasets, with a significant improvement on the Arabidopsis thaliana dataset.

plant species. Notably, for the Zea mays dataset, KAN-Mixer attains superior accuracy and AUC-ROC score of 78.9% and 87.2%, respectively, outperforming the ConvNet's [21] 73.2% and 80.1%. This remarkable improvement highlights KAN-Mixer's capability to handle complex biological tasks in an internal environment. Moreover, the results demonstrate that KAN-Mixer's performance is on par with the baseline across the remaining plant species, with comparable or marginally lower scores, demonstrating the effectiveness and applicability of KAN-Mixer. A detailed experiment can be found in Figure 4. More results are shown in the Appendix.

Method	Zea		Sor		Ara		Sol		Average	
	Acc ↑	AUC-ROC ↑	Acc ↑	AUC-ROC ↑	Acc ↑	AUC-ROC ↑	Acc ↑	AUC-ROC ↑	Acc ↑	AUC-ROC ↑
KAN-Mixer	78.2	86.7	75.4	83.3	71.0	80.3	70.7	83.9	74.1	84.0
ConvNet [21]	76.8	83.4	74.0	81.9	75.0	83.2	74.0	81.9	75.8	83.4

Table 3: Out-domain performance comparison across plant datasets. While our method KAN-Mixer exhibits slightly lower accuracy and Auc-roc scores compared to the baseline for certain plant species like Arabidopsis thaliana, its overall performance remains highly competitive. Notably, KAN-Mixer outperforms the baseline in predicting traits for Zea mays, achieving higher accuracy of 0.782 and Auc-ROC of 0.867. This showcases KAN-Mixer's ability to effectively transfer knowledge from in-domain tasks to related out-domain contexts, demonstrating its strong generalization capabilities.

Out-domain Experiments Extending our evaluation to out-domain scenarios, the findings presented in Table 3 underscore KAN-Mixer's remarkable generalization capabilities. While KAN-Mixer's accuracy and AUC-ROC scores exhibit a slight decrease compared to the baseline for certain plant species, such as Arabidopsis thaliana and Solanum lycopersicum, its overall performance remains highly competitive and on par with the baseline method's average scores. Remarkably, KAN-Mixer outperforms the baseline in predicting traits for Zea mays, achieving superior accuracy and AUC-ROC scores of 78.2% and 86.7%, respectively. This outcome highlights KAN-Mixer's adeptness at transferring knowledge acquired from in-domain tasks to related yet distinct out-domain contexts, thereby enhancing its predictive prowess and generalization capabilities. The

Species	Tissue	Chromosome	Baseline		KAN-Mixer	
			acc ↑	aucroc↑	acc↑	aucroc↑
Arabidopsis thaliana	leaf	1	0.677	0.748	0.757	0.850
Arabidopsis thaliana	leaf	2	0.682	0.753	0.779	0.856
Arabidopsis thaliana	leaf	3	0.799	0.870	0.797	0.890
Arabidopsis thaliana	leaf	4	0.799	0.851	0.776	0.863
Arabidopsis thaliana	leaf	5	0.746	0.803	0.753	0.847
Arabidopsis thaliana	root	1	0.693	0.763	0.768	0.834
Arabidopsis thaliana	root	2	0.741	0.819	0.797	0.893
Arabidopsis thaliana	root	3	0.756	0.839	0.826	0.910
Arabidopsis thaliana	root	4	0.723	0.804	0.824	0.887
Arabidopsis thaliana	root	5	0.702	0.762	0.811	0.887
Solanum lycopersicum	leaf	1	0.748	0.843	0.755	0.845
Solanum lycopersicum	leaf	2	0.749	0.826	0.806	0.843
Solanum lycopersicum	leaf	3	0.754	0.820	0.757	0.796
Solanum lycopersicum	leaf	5	0.853	0.870	0.855	0.884
Solanum lycopersicum	leaf	6	0.790	0.867	0.788	0.860
Solanum lycopersicum	leaf	7	0.807	0.859	0.799	0.851
Solanum lycopersicum	leaf	8	0.800	0.872	0.815	0.863
Solanum lycopersicum	leaf	9	0.801	0.850	0.795	0.848
Solanum lycopersicum	leaf	10	0.828	0.864	0.772	0.852
Solanum lycopersicum	leaf	11	0.856	0.905	0.847	0.900
Solanum lycopersicum	leaf	12	0.793	0.811	0.793	0.823
Solanum lycopersicum	root	1	0.779	0.876	0.771	0.854
Solanum lycopersicum	root	2	0.802	0.854	0.792	0.851
Solanum lycopersicum	root	3	0.784	0.826	0.741	0.812
Solanum lycopersicum	root	4	0.820	0.872	0.811	0.879
Solanum lycopersicum	root	5	0.809	0.854	0.786	0.849
Solanum lycopersicum	root	6	0.849	0.906	0.842	0.897
Solanum lycopersicum	root	7	0.814	0.885	0.791	0.863
Solanum lycopersicum	root	8	0.823	0.896	0.755	0.841
Solanum lycopersicum	root	9	0.788	0.830	0.799	0.864
Solanum lycopersicum	root	10	0.836	0.878	0.788	0.861
Solanum lycopersicum	root	11	0.857	0.906	0.829	0.897
Solanum lycopersicum	root	12	0.759	0.823	0.781	0.799

Table 4: In-domain performance comparison across plant datasets.

consistent and robust performance of KAN-Mixer across both in-domain and out-domain experiments substantiates its efficacy and potential for widespread adoption in diverse biological applications.

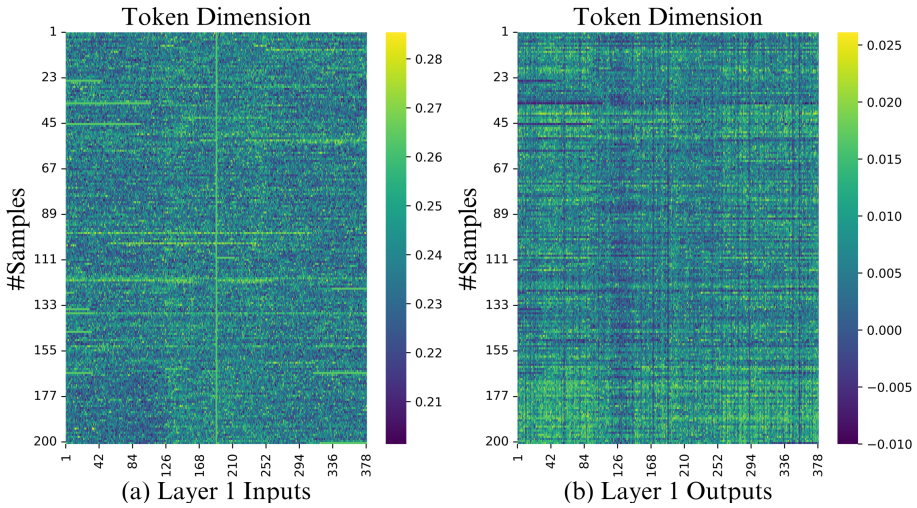
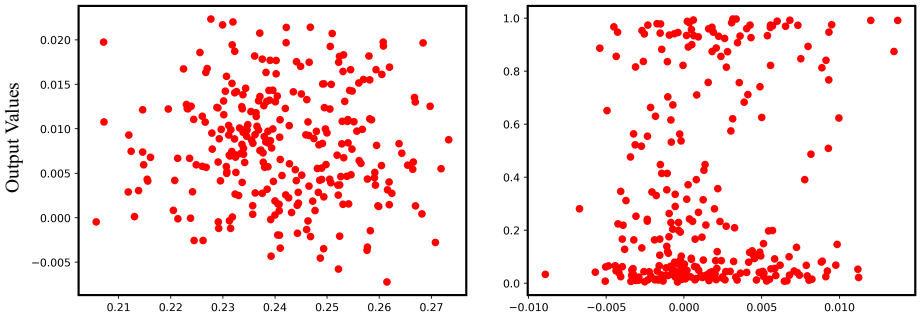


Fig. 4: KAN-Mixer efficiently exchanges input information across both token and feature dimensions. We visually represent the input (a) and output (b) values of a single layer of KAN-Mixer. The x-axis corresponds to the token dimensions derived from 8-mer gene sequences, ranging from 1 to 378 dimensions, while the y-axis represents the data dimensions. This visualization is based on 200 test samples post-training. Each plotted point (x, y) is color-coded based on the numerical value of the x-th dimension of the token for the y-th sample, where higher numerical values are depicted in shades of blue and lower ones in shades of yellow.

4.3 Interpretability and Visualization

KAN-Mixer effectively exchanges the input information in both token and feature dimensions. We plot the different values of the inputs and outputs (left and right) of one KAN-Mixer layer in Figure 4. The figure is drawn from the root of the Solanum lycopersicum, on the 6th chromosome. The x-axis delineates the token dimensions extracted from 8-mer gene sequences, ranging from 1 to 378 dimensions. The y-axis represents the data dimensions. After training, a total of two hundred test samples contributed to the generation of this graph. Each point (x, y) is colored according to the numerical value of the x-th dimension of the token for the y-th sample, where a greater numerical value corresponds to a bluer color, while a smaller one corresponds to a more yellow hue. Before our KAN-Mixer layer, the data feature has almost no distribution among the token dimensions. However, the distribution among the token

dimensions becomes vivid after KAN-Mixer layer, since there is a significant increase in the number of vertical lines in the output graph. Compared to the input, the output distribution exhibits a pattern of interweaving both horizontally and vertically.



(a) Input values of layer 1 (b) Input values of layer 2

Fig. 5: KAN-Mixer reveals an emergent intra-layer pattern in untranslated gene regions and gene expression. We employ the same pruning and visualization techniques as KAN [19]. Due to the high token dimension of 128, we rank and select the top dimension based on importance scores for visualization. Red points represent different data samples. The x-axis shows the input values of each sample at the top input dimension, while the y-axis shows the output values at the top output dimension. Layer 1 exhibits an approximate Gaussian distribution, whereas layer 2 shows a distinct Sigmoid distribution.

KAN-Mixer demonstrates a spontaneous intra-layer pattern in untranslated gene regions and gene expression. Figure 5 adopts the same pruning and visualization of KANs [19], where a scatter figure is plotted to record the pre-layer input and the post-layer output in diverse dimensions. As the token dimension of 128 is too high for visualization, we rank the same importance score and select the top dimension. The red points in the figure represent different data samples. The x-axis plots the value of input value of each sample at the top input dimension. The y-axis plots the value of output value of each sample at the top output dimension. Layer 1 has a rough Gaussian distribution while layer 2 represents a different Sigmoid distribution. Note that we never take both Gaussian and Sigmoid-like functions in our training losses. This spontaneous nature demonstrates the interpretability within different KAN-Mixer layers, which was neither feasible nor practical in previous deep learning models.

5 Discussion

KANs have recently demonstrated their potential in aiding scientists to identify crucial relationships within input dimensions, thereby facilitating new scientific discoveries due to their superior interpretability. In this paper, we explore the applicability of KANs in predicting biological gene sequence expression, specifically within an ecological context. To ensure interpretability, we encountered the challenge of inputting extremely long gene

sequences into KAN, resulting in a substantial parameter burden. Drawing inspiration from the concept of k-mer in bioinformatics, we propose introducing a locality-focused weight-sharing inductive bias to KAN, significantly reducing computational costs. Specifically, we employ convolution operators for token embedding to transform one-hot gene features into k-mer tokens. These tokens are then processed by KAN-augmented token mixing and channel mixing blocks, which efficiently exchange information across both token and feature dimensions. Our experiments on gene sequence expression prediction for non-coding regulatory elements indicate that our method, KAN-Mixer, achieves performance comparable to a state-of-the-art ConvNet-based approach. More importantly, this new approach retains KANs' excellent interpretability, accurately identifying key segments within input gene sequences without the need for additional tools. This novel modeling paradigm is poised to advance KANs' role in promoting ecological and biological scientific discoveries in the future. By providing a robust and interpretable framework for gene expression analysis, KAN-Mixer can significantly enhance our understanding of ecological dynamics and biodiversity. This approach has the potential to inform conservation strategies, improve agricultural practices, and contribute to ecological management across various environmental contexts.

Ethics Considerations In the application of KANs for gene expression prediction, it is essential to consider the ethical implications, particularly in the context of ecological research. Our work emphasizes the importance of transparency and interpretability in AI models, ensuring that the insights derived from these models can be understood and trusted by the broader scientific community. This transparency is crucial for the responsible application of AI in sensitive areas such as conservation and biodiversity, where decisions can have significant environmental impacts. Furthermore, the deployment of such models must respect the ethical guidelines surrounding genetic data usage, including privacy and consent, especially when dealing with indigenous and local communities' genetic resources. Adhering to ethical standards helps in maintaining public trust and fostering collaborations across different sectors. By integrating ethical considerations into our research, we aim to set a precedent for the responsible use of advanced computational techniques in ecological and biological studies, ensuring that scientific progress goes hand in hand with ethical responsibility.

Limitations & Future Work Our research targets the prediction of gene sequence expression in the biological domain, focusing on identifying crucial segments within well-expressed gene sequences. Despite the promising results, several critical questions remain unanswered, indicating directions for future research. Primarily, KAN-Mixer has concentrated on classifying gene sequence expression in various plant non-coding regulatory elements, offering a more intuitive and interactive framework. Leveraging KANs' interpretability, we aim to deepen our understanding of the underlying mechanisms of gene expression, thereby accelerating scientific discovery. Future research should explore whether a clearer link between input features and outputs can improve our comprehension of gene-phenotype relationships. Moreover, future work should investigate the integration of large-scale pre-trained features from DNA foundation models, such as Evo [20]. Utilizing these pre-trained features could significantly improve the efficiency and effectiveness of our models, leading to more accurate and insightful predictions.

References

1. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)
2. Bileschi, M.L., Belanger, D., Bryant, D.H., Sanderson, T., Carter, B., Sculley, D., Bateman, A., DePristo, M.A., Colwell, L.J.: Using deep learning to annotate the protein universe. Nature Biotechnology **40**(6), 932–937 (2022)
3. Braun, J., Griebel, M.: On a constructive proof of kolmogorov's superposition theorem. Constructive approximation **30**, 653–675 (2009)
4. Buhmann, M.D.: Radial basis functions. Acta numerica **9**, 1–38 (2000)
5. Cao, Y.H., Wu, J.: A random cnn sees objects: One inductive bias of cnn and its applications. In: Proceedings Of The AAAI Conference On Artificial Intelligence. vol. 36, pp. 194–202 (2022)
6. Cunningham, H., Ewart, A., Riggs, L., Huben, R., Sharkey, L.: Sparse autoencoders find highly interpretable features in language models. arXiv preprint arXiv:2309.08600 (2023)
7. Cybenko, G.: Approximation by superpositions of a sigmoidal function. Mathematics of control, signals and systems **2**(4), 303–314 (1989)
8. d'Ascoli, S., Touvron, H., Leavitt, M.L., Morcos, A.S., Biroli, G., Sagun, L.: Convit: Improving vision transformers with soft convolutional inductive biases. In: International conference on machine learning. pp. 2286–2296. PMLR (2021)
9. EMBL-EBI: Ensembl plants databas. `https://plants.ensembl.org/index.html` (2024)
10. Ghosh, J., Nag, A.: An overview of radial basis function networks. Radial basis function networks 2: new advances in design pp. 1–36 (2001)
11. Haykin, S.: Neural networks: a comprehensive foundation. Prentice Hall PTR (1998)
12. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural networks **2**(5), 359–366 (1989)
13. Idury, R.M., Waterman, M.S.: A new algorithm for dna sequence assembly. Journal of computational biology **2**(2), 291–306 (1995)
14. Kolmogorov, A.N.: On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables. American Mathematical Society (1961)
15. Köppen, M.: On the training of a kolmogorov network. In: Artificial Neural Networks—ICANN 2002: International Conference Madrid, Spain, August 28–30, 2002 Proceedings 12. pp. 474–479. Springer (2002)
16. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Computer Science (2009)
17. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
18. Li, Z.: Kolmogorov-arnold networks are radial basis function networks. arXiv preprint arXiv:2405.06721 (2024)
19. Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljačić, M., Hou, T.Y., Tegmark, M.: Kan: Kolmogorov-arnold networks. arXiv preprint arXiv:2404.19756 (2024)
20. Nguyen, E., Poli, M., Durrant, M.G., Thomas, A.W., Kang, B., Sullivan, J., Ng, M.Y., Lewis, A., Patel, A., Lou, A., et al.: Sequence modeling and design from molecular to genome scale with evo. bioRxiv pp. 2024–02 (2024)
21. Peleke, F.F., Zumkeller, S.M., Gültas, M., Schmitt, A., Szymański, J.: Deep learning the cis-regulatory code for gene expression in selected model plants. Nature Communications **15**(1), 3488 (2024)

22. Pevzner, P.A., Tang, H., Waterman, M.S.: An eulerian path approach to dna fragment assembly. Proceedings of the national academy of sciences **98**(17), 9748–9753 (2001)
23. Sprecher, D.A., Draghici, S.: Space-filling curves and kolmogorov superposition-based neural networks. Neural Networks **15**(1), 57–67 (2002)
24. Tolstikhin, I.O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., et al.: Mlp-mixer: An all-mlp architecture for vision. Advances in neural information processing systems **34**, 24261–24272 (2021)
25. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
26. Wang, Z., Wu, L.: Theoretical analysis of the inductive biases in deep convolutional networks. Advances in Neural Information Processing Systems **36** (2024)