Positional frequency chaos game representation for machine learning-based classification of crop lncRNAs

Athanasios Papastathopoulos-Katsaros[1,✉] **and Zhandong Liu**[1,2]

[1]Department of Pediatrics, Baylor College of Medicine, 1 Baylor Plaza, Houston, TX, 77030, United States of America
[2]Data Science Center, Jan and Dan Duncan Neurological Research Insitute, 1250 Moursund Street, Houston, TX, 77030, United States of America

Motivation: Alignment-based methods are fundamental for sequence comparison but are often computationally prohibitive for large-scale genomic analyses. This limitation has spurred the development of quicker, alignment-free alternatives, such as k-mer analysis, which are crucial for studying long non-coding ribonucleic acids (lncRNAs) in plants. These lncRNAs play critical roles in regulating gene expression at both the epigenetic and transcriptomic levels. However, existing alignment-free approaches typically lose positional information, which can be vital for achieving accurate classification.

Results: We propose positional frequency chaos game representation (PFCGR), a novel encoding that improves the traditional frequency chaos game representation (FCGR) by incorporating four statistical moments of k-mer positions: mean, standard deviation, skewness, and kurtosis. This creates a multi-channel image representation of genomic sequences, enabling machine learning models such as Logistic Regression, Random Forests, and Convolutional Neural Networks to classify plant lncRNAs directly from raw genomic sequences. Tested on seven major crop species, our PFCGR-based classifiers achieve classification accuracies comparable to or exceeding those of the computationally intensive DNABERT-based model (Danilevicz et al. (1)), while requiring significantly less computational resources. These results demonstrate PFCGR's potential as an efficient and accurate tool for plant lncRNA identification, as well as its ability to facilitate large-scale computational studies in genomics.

Correspondence: ***u253726@bcm.edu***

DRAFT

Introduction

Alignment-based methods are essential for analyzing biological sequences but require significant computational resources. This need has accelerated the development of faster, alignment-free options, like k-mer analysis, which are vital for the study of different biological organisms. In plant biology, alignment-free methods show great promise for identifying long non-coding RNAs (lncRNAs) in crop species, offering significant advantages over traditional alignment-based approaches.

LncRNAs, defined as RNA molecules exceeding 200 nucleotides in length, are a critical component of the plant transcriptome with no coding potential but significant regulatory functions (Jin et al. (2), Wang et al. (3), Zhang et al. (4)). They are transcribed by RNA Polymerase II and are often polyadenylated, though significant exceptions exist. Plant genomes, unlike those of yeast and mammals, also utilize the plant-specific RNA polymerases Pol IV and Pol V to produce distinct lncRNAs, some of which are non-polyadenylated. These include upstream non-coding transcripts (UNTs) in *Arabidopsis* that are analogous to mammalian promoter upstream transcripts (PROMPTs), suggesting a nuanced regulatory landscape (Chekanova et al. (5)).

LncRNAs are involved in a myriad of regulatory roles across various stages of plant development and are differentially expressed across tissues, reflecting their specialized functions. Notably, lncRNAs influence key developmental processes, such as the timing of flowering (Lee et al. (6), Waheed and Zeng (7)), and are crucial for plants' resilience to environmental stresses (Thiebaut et al. (8, 9), Salvador-Guirao et al. (10), Liu et al. (11), Hu et al. (12)).

In terms of functionality, lncRNAs modulate gene expression through a variety of mechanisms, including epigenetic modifications (Heo and Sung (13), Guil and Esteller (14), Urquiaga et al. (15)). For instance, in rice, the lncRNA LAIR notably increases grain yield by upregulating the LRK gene cluster, which leads to larger and more numerous panicles (Wang et al. (16)). Another lncRNA, Ef-cd, sourced from the antisense strand of the OsSOC1 locus, significantly shortens the maturity period for late-maturing rice varieties without reducing yield, showcasing lncRNAs' ability to fine-tune plant growth under cultivation constraints (Fang et al. (17)). These advancements in lncRNA research are paving the way for significant contributions to agricultural biotechnology, including the development of genetically engineered crops with increased yields, improved resistance to pests and diseases, and greater environmental stress tolerance. Thus, understanding and predicting lncRNAs remain crucial in modern plant science and biotechnology. For additional implications of lncRNA's in physiological processes, we refer the reader to Wang and Chekanova (18).

Bioinformatics pipelines for lncRNA prediction often combine custom scripts and tools like CPC2 (Kang et al. (19)), LncMachine (Cagirici et al. (20)), CPAT (Wang et al. (21)), PlncPRO (Singh et al. (22)), and PLEK (Li et al. (23)) to filter and estimate the coding potential of transcripts from transcriptome datasets. These methods typically involve manually defining features from transcripts, which are then used in machine-learning models to classify transcripts as mRNA or

lncRNA (Pian et al. (24), Fan and Zhang (25)). However, the reliance on handcrafted features can introduce bias and affect classification accuracy (Eraslan et al. (26), Shrestha and Mahmood (27)), while the training of models depends heavily on transcriptome data from sources like CANTATAdb (Szcześniak et al. (28)), PLncDB (Jin et al. (2)), GreeNC (Paytuví Gallart et al. (29), Di Marsico et al. (30)) and AlnC (Singh et al. (31)). For this reason, a shift towards models that utilize genomic references could improve the detection of lncRNAs, particularly given their tissue- or stage-specific low expression rates.

One such work (Danilevicz et al. (1)) is based on DNABERT (Ji et al. (32)), a specialized adaptation of the language BERT model (Devlin et al. (33)) tailored for DNA sequences. It addresses the binary classification task of distinguishing true lncRNA from synthetic lncRNA sequences, as explained in more detail in subsequent sections. DNABERT is capable of performing sequence classification by encoding DNA sequences into k-mer tokens, which allows it to capture intricate biological features. This is achieved through the transformer architecture and the attention mechanism (Vaswani et al. (34)), which unfortunately comes with a significant computational cost and the black-box nature of the model. For a more extensive discussion on related work, refer to Li et al. (35) and Danilevicz et al. (1).

To address the aforementioned challenges, we introduce a novel method for classifying plant lncRNAs based on genomic sequences, leveraging machine-learning techniques such as Logistic Regression (LR), Random Forest (RF), and Convolutional Neural Networks (CNN). Our approach uses Frequency Chaos Game Representation (FCGR) and our innovative improved version, "positional FCGR" (PFCGR), which incorporates the following statistical features of the positions of all the k-mers of the sequences: mean, standard deviation, skewness, and kurtosis. Therefore, the aims of this study are: (i) to address FCGR's limitation of losing positional information; (ii) to develop and evaluate PFCGR-based machine-learning models for the identification of plant lncRNAs directly from genomic sequences; (iii) to compare the accuracy and computational efficiency of PFCGR-based models against the very accurate and much more complex DNABERT-based method across seven crop species.

Materials and methods

FCGR encoding. The Chaos Game Representation (CGR), originally used to construct fractals from random inputs, was later adapted to encode DNA sequences (Jeffrey (36)). For readers unfamiliar with CGR, a simple illustration is provided in Figure 1. In this work, we use the Frequency CGR (FCGR), a discrete matrix equivalent of CGR that counts k-mer frequencies at each position. Unlike traditional encoding methods such as one-hot encoding, FCGR encodes nucleotide sequences into images/numerical data, standardizing sequences of varying lengths into images of the same size. For further reading, see Löchel and Heider (37).

In a nutshell, the idea behind FCGR is to map k-mers onto a unit square by recursively dividing the square into smaller regions according to the sequence of bases in each k-mer. Each base (A, C, G, T) determines which quadrant of the current region to select, and this subdivision continues for all positions in the k-mer, resulting in a unique location within a $2^k \times 2^k$ grid corresponding to that k-mer. The FCGR image is constructed by plotting each k-mer's frequency on that grid, which is normalized to compute pixel intensities. In this study, k ranges from 3 to 6. Figure 2 shows FCGR images from the *Zea Mays* dataset, highlighting key regions that could potentially aid in sequence classification.

FCGR has shown success in various applications, such as detecting COVID-19 (Hammad et al. (38), Avila Cartes et al. (39)), distinguishing human coding and non-coding regions (Nasr and Oueslati (40)), and predicting antimicrobial resistance (Ren et al. (41)). It has also been shown to outperform traditional one-hot encoding in bacterial 16S rRNA classification (Rizzo et al. (42)). Recently, a new FCGR-based representation was developed based on ranked frequencies instead of raw, which was integrated with Transformers and other advanced neural network architectures and showed improved performance, particularly in low-coverage genomic data (De Medeiros et al. (43)). Lastly, Alipour et al. (44) applied FCGR in unsupervised clustering of DNA sequences using CNN combined with twin contrastive learning, demonstrating high clustering accuracy across viral genomes and other genomic datasets.

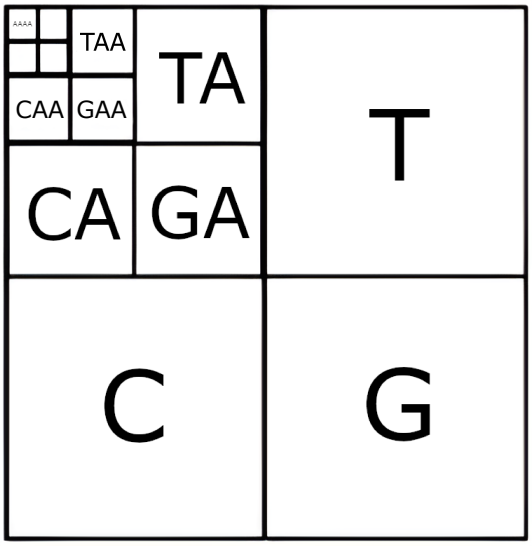

Fig. 1. Illustration of the CGR algorithm: The square is initially divided into four equal parts, each representing one of the four nucleotides found in DNA: Adenine (A), Thymine (T), Cytosine (C), and Guanine (G). Each quadrant is assigned to one nucleotide. Each of these initial quadrants is further subdivided into four smaller squares. This division process respects the nucleotide of the larger square from which it is derived. For example, the square for Adenine (A) is subdivided into AA, TA, CA, and GA. This method applies recursively to all quadrants. With each iteration, the squares become increasingly specific to particular sequences of nucleotides. For instance, the top left quadrant, after the second iteration, shows further subdivisions like AAA, TAA, CAA, and GAA, each representing a specific 3-mer, and the grid in the top-left corner of this figure represents the 4-mer AAAA. Finally, to map a DNA sequence onto this square, we begin at the center and move to each subsequent quadrant based on the next nucleotide in the sequence. The position within the square becomes more precise with each nucleotide added, tracing a path through increasingly smaller quadrants.

DRAFT

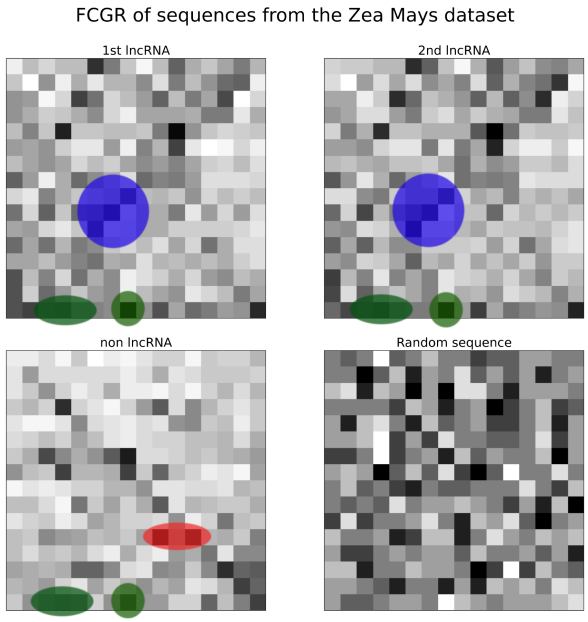

Fig. 2. FCGR images with k=4 are displayed for four sequences: two lncRNAs, one non-lncRNA, and one random sequence. Green circles highlight patterns common to both lncRNAs and the non-lncRNA, indicating shared genomic motifs. Blue circles indicate patterns unique to lncRNAs, suggesting motifs specific to lncRNA. Red circles highlight patterns exclusive to the non-lncRNA sequence. Not all patterns are marked to maintain visual clarity in the images.

Positional FCGR encoding. One major drawback of FCGR is that it does not contain positional information. For this reason, we extend the FCGR method by adding positional information (PFCGR) through statistical measures of the k-mer positions, including mean, standard deviation, skewness, and kurtosis. These are computed using Welford's online algorithm (Welford (45)), reducing memory requirements and computational overhead. Each statistical measure generates a separate $2^k \times 2^k$ image or is kept as tabular data, as explained below. The intensity of each pixel corresponds to the statistical value of the respective k-mer and is normalized by min-max normalization. This added positional context improves FCGR by capturing more complex patterns in genomic sequences, where k-mer placement matters.

There are two ways to input these added statistical measures. On the one hand, we can organize the extracted features into tabular data. For FCGR, each column represents a k-mer (e.g., "AAT"), and each cell contains its frequency in each sequence of the dataset. For PFCGR, the columns are labeled with both the k-mer and the statistical descriptors (Frequency, Mean, Standard Deviation, Skewness, Kurtosis). The data in this format can then be inputted into LR and RF models. On the other hand, the data in image format (which can be thought of as either five images or one image with five channels) can be inputted into Convolutional Neural Network (CNN) models, which are effective in identifying patterns in the generated images, even across channels. For missing k-mers, we assign zero frequencies in the FCGR matrix. We experimented with different imputation strategies for handling the rest of the missing statistics in the PFCGR matrices, including filling with zeros, using -1 as a sentinel value, and adding an additional binary masking channel indicating presence or absence of a k-mer. Since all approaches yielded similar performance, we opted to use zeros for simplicity. This issue does not affect the LR and RF methods, as the tabular format does not contain those columns.

Lastly, we acknowledge that previous research has explored the decoding of positional information from FCGR images using the wavelet transformation (Frequency Chaos Game Signal, Messaoudi et al. (46)) and correlations among k-mers (Sievers et al. (47)). However, none have implemented a similar approach to ours to incorporate positional information.

Model architectures and training. In selecting our classifiers, we prioritized simplicity, opting for computationally efficient models. LR and RF were chosen for their low computational cost and ease of computing feature importance, while CNNs were included for their ability to capture complex patterns in image-based data. Our CNN architecture for FCGR includes, in this order, a convolutional layer, followed by a pooling layer, a fully-connected hidden layer, and a final sigmoid output neuron. All other layers use ReLU activation functions. The difference for PFCGR is that the five images are processed separately by one convolutional layer before merging their outputs to the rest of the neural network, which has the same architecture as the single-image FCGR setup. This is intended to capture unique features of each image before further processing.

Each species has a dedicated classifier, individually trained in species-specific datasets, to accurately distinguish true lncRNA from synthetic non-lncRNA sequences. For LR, we applied L2 regularization, which demonstrated more robust performance compared to LASSO. For RF, we optimized the number of trees and their maximum depths using grid search. Specifically, the number of trees was varied between 250 and 1250, while the maximum depth ranged from unrestricted to a limit ranging from 10 to 40, balancing model complexity and reducing overfitting. Lastly, for the CNN, the number of filters and filter sizes in the convolutional layers were optimized via grid search. The max pooling layer's window size and stride were also tuned, along with the number of neurons in the fully connected layer. Specifically, the number of filters was varied between 16 and 128, while kernel sizes ranged from 1×1 to 8×8. Pooling window sizes and strides were explored from 2×2 up to 6×6 to identify the optimal configuration. The model was trained using binary cross-entropy loss with the Adam optimizer (learning rate = 0.001), batch sizes ranging from 8 to 32, and for 7 to 10 epochs. To prevent overfitting and ensure reliable performance, stratified 5-fold cross-validation was applied. All CNN models were trained on an NVIDIA A30 GPU to reduce training time.

Computational complexity of PFCGR compared to DNABERT. To compare the computational complexity of PFCGR to DNABERT's, we consider both feature extraction and classifier complexity, using RF for illustration. For a sequence of length N and a fixed k-mer size, feature extraction involves iterating over the sequence once (sliding a window of length k), with each k-mer requiring $\mathcal{O}(1)$ time because

DRAFT

Species	Median	Max	Min
A. thaliana	1,514	34,398	1,200
B. napus	1,854	50,461	702
B. oleracea	1,832	40,951	203
B. rapa	2,070	54,816	205
G. max	2,143	102,476	1,200
O. sativa	1,943	169,841	1,200
Z. mays	1,978	648,853	1,200

Table 1. Sequence length statistics for each crop species dataset.

Species	lncRNA	Non-lncRNA
A. thaliana	3,467	3,464
B. napus	8,544	7,442
B. oleracea	5,350	4,997
B. rapa	7,075	6,882
G. max	2,320	2,288
O. sativa	2,335	2,334
Z. mays	8,889	8,747

Table 2. Label distribution for each crop species dataset.

of hashing and Welford's online algorithm ability to compute the statistical measures on-the-fly. Thus, the total complexity for generating the PFCGR feature vector is $\mathcal{O}(N)$. For M training sequences, each with a feature vector of length $\approx 5 \cdot 4^k$, the RF complexity with T trees and tree depth $\mathrm{d_{tree}}$ is $\mathcal{O}\big(M \times (5 \cdot 4^k) \times T \times \mathrm{d_{tree}}\big)$, with 4^k growing large but not being a concern for small k (e.g., $k \leq 6$).

In contrast, DNABERT, which is based on the Transformer architecture, has a computational complexity of $\mathcal{O}(L \times (N^2 \times d + N \times d^2))$ for a sequence of length N, where d is the embedding size and L is the number of feed-forward layers. The self-attention mechanism in DNABERT introduces a quadratic complexity on N, which makes it significantly more computationally demanding than PFCGR. Moreover, the sequences considered in this study range from a median length of 1500 to 2200 nucleotides (see Table 1), which is substantially longer than the typical input length range where DNABERT performs optimally. DNABERT's accuracy and efficiency are known to be more favorable when handling shorter sequences due to limitations in model architecture and resource demands. When applied to longer sequences, such as those in our dataset, the quadratic scaling of self-attention not only results in increased computational cost but may also lead to degradation in performance, as the model struggles to effectively capture long-range dependencies without prohibitive resource requirements. As a result, while PFCGR typically completes training in approximately thirty minutes to an hour, DNABERT may require several hours to days for fine-tuning.

As a last remark, we would like to add that comparing execution times for PFCGR and DNABERT is not straightforward due to several factors. The two methods run on different hardware, with DNABERT typically using GPUs, which accelerates training compared to PFCGR, typically run on CPUs. Additionally, each method has distinct optimizations: DNABERT may leverage techniques like mixed-precision training and optimized matrix operations, while PFCGR benefits from efficient hashing and parallel k-mer computation. Furthermore, their parallelization strategies differ, with DNABERT commonly utilizing multi-GPU setups and PFCGR using multi-threading on CPUs. These variations in hardware, optimizations, and parallelization make direct execution time comparisons unreliable. Hence, we only provide a theoretical analysis here.

Datasets and experimental setup. We used the same data set as described by Danilevicz et al. (1). In brief, the lncRNA datasets for all crop species (*Arabidopsis thaliana*, *Brassica napus*, *Brassica oleracea*, *Brassica rapa*, *Glycine max*, *Oryza sativa*, and *Zea mays*) were sourced from CantataDB 2.0 (Szcześniak et al. (28)) and PLncDB 2.0 (Jin et al. (2)). To enrich the dataset, flanking regions were added both upstream and downstream of each lncRNA sequence to prevent the classifier from merely identifying broad genomic features. A control dataset was constructed by extracting random genomic stretches, explicitly excluding the lncRNA sequences and their flanking regions. This approach increases the complexity of the classification task compared to the simpler mRNA versus lncRNA, encouraging the model to learn discriminative features specific to lncRNA regions within their genomic context. For a comprehensive explanation of the dataset construction, we direct the reader to Danilevicz et al. (1).

In our experiments, we performed binary classification between lncRNA and non-lncRNA sequences separately for each species. This choice was motivated by intrinsic differences in the datasets, such as variations in sequence length distributions, nucleotide composition, and class imbalances. Training separate classifiers per species allows them to better capture features unique to each genome. Table 1 summarizes the length statistics of the sequences for each species dataset used, including the median, maximum, and minimum sequence lengths, showing the scale of the datasets.

Table 2 summarizes the number of sequences labeled lncRNA and non-lncRNA for each dataset. The class distribution varies between species, reflecting differences in available annotated sequences and the size of the control sets. For this reason, to robustly evaluate model performance, we employed stratified 5-fold cross-validation. Each fold was used once as the test set, while the remaining four folds served as training data. This approach ensures that the evaluation is reliable and that each fold contains representative samples of both classes.

Metrics. As in Danilevicz et al. (1), model effectiveness was measured using accuracy, precision, recall, and F1 score. True positives, false negatives, and false positives represent correctly identified lncRNAs, misclassified lncRNAs, and falsely identified lncRNAs, respectively.

Results

We conducted two sets of experiments to evaluate our method. In the first set, we used LR, RF, and CNN for binary

DRAFT

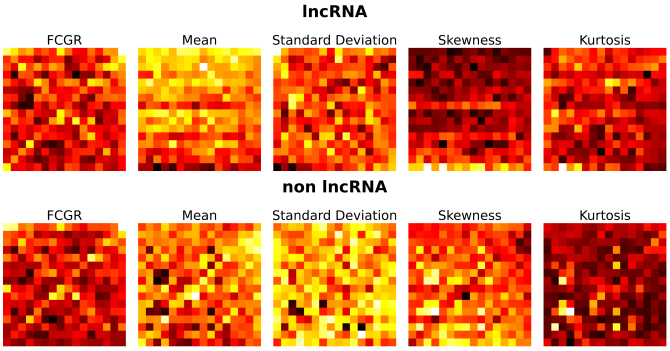

Fig. 3. One lncRNA and one non-lncRNA sequence are shown using the standard FCGR and the positional FCGR. The top row displays the lncRNA, while the bottom row depicts the non-lncRNA. Despite very similar FCGRs for both sequences, PFCGRs reveal differences, emphasizing their discriminative power. The color scale ranges from whiteish yellow (indicating low values) to dark red (indicating high values), with all values normalized across the statistics using min-max normalization to ensure consistent comparison.

classification between non-lncRNA and lncRNA sequences in the seven species. We applied FCGR and PFCGR and optimized the k-mer size for each species, ranging from 3-mers to 6-mers, to evaluate the impact of the additional statistical features and their importance. This set also included a comparison with the DNABERT model (Danilevicz et al. (1)). In the second set of experiments, we aimed to improve classification accuracy by applying RF, our most robust and accurate classifier, across all k-mer sizes combined (k=3 to k=6), to further improve the model's performance and provide a better comparison with DNABERT. This set of experiments served to support the findings of the first.

Performance evaluation using the optimal k-mers. To evaluate whether PFCGR boosts performance over the traditional FCGR method, we apply LR, RF, and CNN, using various k-mer sizes to determine the best accuracy. The range of k-mer lengths considered (k ranging from 3 to 6) reflects a common practice in k-mer-based approaches: smaller k values tend to be overly frequent and less informative, while k values greater than 6 become increasingly rare, resulting in sparse and potentially noisy features that can skew model performance. Thus, the selected range is more balanced.

Before presenting our results, we first want to address any skepticism regarding the utility of PFCGR over regular FCGR. To do that, we present an example from the *Zea Mays* dataset in Figure 3. As it can be seen, while lncRNA and non-lncRNA sequences exhibit similar FCGR patterns, incorporating additional statistical measures reveals significant differences. The four new metrics clearly distinguish between the two sequence types, suggesting that the PFCGR method offers more information, potentially improving lncRNA classification accuracy.

In Table 3, we summarize our results obtained with the optimal k-mer lengths for each species, where"optimal" refers to the k-mer size within the range 3–6 that yields the highest classification accuracy for that species. Specifically, a k-mer of length 3 works best for *A. thaliana*, *B. napus*, *B. oleracea*, *B. rapa*, and *O. sativa*, while lengths of 4 and 6 are optimal for *G. max* and *Z. mays* respectively. All results reported are the averages obtained from stratified 5-fold cross-validation to ensure robustness of the models. Our results

Species	Method	Acc.	Prec.	Rec.	F1
A. thaliana	LR - FCGR	0.56	0.56	0.55	0.56
	LR - PFCGR	0.59	0.59	0.60	0.60
	RF- FCGR	0.61	0.61	0.64	0.62
	RF - PFCGR	0.65	**0.67**	0.60	0.63
	CNN - FCGR	0.62	0.61	**0.68**	0.64
	CNN - PFCGR	**0.66**	0.66	0.65	**0.65**
	DNABERT	0.65	0.65	0.65	**0.65**
B. napus	LR - FCGR	0.55	0.54	1.00	0.70
	LR - PFCGR	0.64	0.66	0.68	0.67
	RF- FCGR	0.66	0.64	0.81	0.72
	RF - PFCGR	0.70	0.69	0.80	0.74
	CNN - FCGR	0.66	0.64	**0.81**	0.72
	CNN - PFCGR	0.70	0.69	0.80	**0.74**
	DNABERT	**0.75**	**0.75**	0.74	**0.74**
B. oleracea	LR - FCGR	0.59	0.56	0.94	0.70
	LR - PFCGR	0.63	0.63	0.66	0.65
	RF- FCGR	0.67	0.64	0.78	0.71
	RF - PFCGR	0.71	0.71	0.74	0.73
	CNN - FCGR	0.69	0.68	0.76	0.72
	CNN - PFCGR	0.68	0.65	**0.85**	**0.74**
	DNABERT	**0.74**	**0.75**	0.74	**0.74**
B. rapa	LR - FCGR	0.55	0.55	0.64	0.59
	LR - PFCGR	0.55	0.55	0.55	0.55
	RF- FCGR	0.58	0.58	0.65	0.61
	RF - PFCGR	**0.60**	**0.60**	0.67	0.63
	CNN - FCGR	0.59	0.59	0.66	0.62
	CNN - PFCGR	0.56	0.55	**0.80**	**0.65**
	DNABERT	0.58	0.58	0.58	0.58
G. max	LR - FCGR	0.50	0.50	1.00	0.67
	LR - PFCGR	0.62	0.63	0.63	0.63
	RF- FCGR	0.70	0.68	0.77	0.72
	RF - PFCGR	0.72	0.70	0.78	**0.74**
	CNN - FCGR	0.68	0.64	0.83	0.72
	CNN - PFCGR	0.70	0.66	**0.84**	**0.74**
	DNABERT	**0.73**	**0.73**	0.73	0.73
O. sativa	LR - FCGR	0.54	0.55	0.48	0.51
	LR - PFCGR	0.57	0.57	0.57	0.57
	RF- FCGR	0.61	0.61	0.61	0.61
	RF - PFCGR	**0.63**	**0.64**	0.60	0.62
	CNN - FCGR	0.62	0.62	0.65	**0.63**
	CNN - PFCGR	0.57	0.56	**0.67**	0.61
	DNABERT	0.62	0.62	0.62	0.62
Z. mays	LR - FCGR	0.72	0.69	0.80	0.74
	LR - PFCGR	0.72	0.72	0.74	0.73
	RF- FCGR	0.79	0.75	**0.88**	0.81
	RF - PFCGR	0.79	0.75	**0.88**	0.81
	CNN - FCGR	0.80	0.77	0.86	0.81
	CNN - PFCGR	0.81	0.78	**0.88**	**0.83**
	DNABERT	**0.83**	**0.84**	0.84	**0.83**

Table 3. Evaluation metrics of species-specific LR, RF, and CNN models using FCGR and PFCGR for the binary classification of lncRNAs across the seven crop species. A comparison with DNABERT is shown. Accuracy, precision, recall, and F1 score are reported for the optimal k-mer size across each method. Our results correspond to the mean performance obtained from stratified 5-fold cross-validation.

DRAFT

k	Metric	FCGR	PFCGR
3	Accuracy	0.61	0.65
4	Accuracy	0.61	0.64
5	Accuracy	0.61	0.64
6	Accuracy	0.59	0.65
3	F1 - score	0.62	0.63
4	F1 - score	0.62	0.63
5	F1 - score	0.60	0.62
6	F1 - score	0.57	0.63

Table 4. A comparison of the robustness of FCGR and PFCGR for different k-mer lengths based on accuracy and F1 score metrics. All results are computed using RF and the *A. thaliana* dataset and represent the mean performance obtained from stratified 5-fold cross-validation.

k	Statistic	FCGR	PFCGR
3	Mean abs cor.	0.20	0.06
3	St.d. of the cor.	0.15	0.08
4	Mean abs cor.	0.10	0.03
4	St.d. of the cor.	0.10	0.05
5	Mean abs cor.	0.05	0.04
5	St.d. of the cor.	0.06	0.05
6	Mean abs cor.	0.03	0.06
6	St.d. of the cor.	0.04	0.06

Table 5. Mean absolute correlations (Mean abs cor.) and their standard deviations (St.d. of the cor.) among all features for different k. All results computed using RF and the *A. thaliana* dataset.

demonstrate that PFCGR improves the classification performance for nearly all species compared to FCGR. For certain ones, RF and CNN models achieve results comparable to or better than DNABERT, whose results were fine-tuned separately for each species to optimize performance. In particular, the higher recall across the board suggests that our models are better at identifying lncRNAs, which is critical in most applications.

Another important observation is that despite logistic regression's tendency to predict an excessive number of lncRNA cases, which inflates recall, PFCGR moderates these predictions leading to more balanced performance metrics. However, one limitation observed with the CNN methodology is the notable disparity in accuracy compared to RF, particularly evident in the PFCGR scenario. We attribute this discrepancy to the large number of features presented in the PFCGR setup relative to the available sample size, which likely leads to overfitting, a frequent challenge encountered with neural networks. However, the results remain accurate, indicating that PFCGR improves the standard FCGR by providing a more robust classification method for lncRNAs without significantly increasing the computational cost.

Before we move on to the next section, we want to mention a few more observations. The difference in performance between different k-mer lengths is smaller for PFCGR, as shown in Table 4 for the example of *A. thaliana*. This suggests that PFCGR adds useful features that contribute to a more robust model. These features refer to the coordinates of FCGR/PFCGR in tabular format. Moreover, feature importance analysis revealed that, for both LR and RF models using PFCGR, the majority of the top 20 significant features were standard deviations and frequencies. This emphasizes the value of incorporating measures of k-mer dispersion along the sequence. To calculate feture importance, we used model-specific criteria: for the LR model, the absolute values of the regression coefficients served as a proxy, while for the RF model, we relied on the average decrease in Gini impurity to measure each individual contribution. Although a few skewness and kurtosis-related features were identified, they were fewer in number, aligning with expectations since they capture more specific distributional aspects and ought to be related to secondary effects. While feature importance analysis is also feasible for CNNs, it tends to be more elaborate and we did not see the need to pursue it here, as we expect the results to be similar to those obtained with LR and RF models.

Lastly, in general the features showed low correlations (Table 5), indicating that they provide complementary information. To compute feature correlations, we calculated the absolute mean correlation and its standard deviation, using Pearson's correlation coefficient as a measure. This low correlation among features is critical, as it reduces the likelihood of multicollinearity, which can skew the model performance.

Species	Best single k	RF with k=3-6	DNABERT
A. thaliana	0.66	**0.68**	0.65
B. napus	0.70	0.72	**0.75**
B. oleracea	0.71	0.72	**0.74**
B. rapa	0.60	**0.61**	0.58
G. max	0.70	**0.73**	**0.73**
O. sativa	0.63	**0.64**	0.62
Z. mays	0.81	0.80	**0.83**

Table 6. Comparison of classification accuracy between the best single k-mer models, multi-k RF using k-mers ranging from k=3 to k=6, and the best DNABERT model across the seven plant species. Our results correspond to the mean performance obtained from stratified 5-fold cross-validation.

Performance evaluation using all k-mers combined. Earlier we observed that the performance does not improve with increasing k-mer size, indicating a complex relationship between k and model accuracy. To address this, we previously tested multiple k values and selected the best one, as performed in the DNABERT study. In this section, we use a RF model with combined k-mer statistics, ranging from k=3 to k=6. We keep the same tabular format, but incorporate multi-scale k-mers in a single feature set. This multi-scale integration is non-trivial for LLM-based models, but straightforward in our approach. However, feature importance analysis reveals that when specific k-mers are significant, their related variants (e.g., subsets or extensions) may also appear important, indicating potential multicollinearity. While this could affect some classifiers, it is not a major issue for RF in practice.

Our results are summarized in Table 6, and as previously, represent averages from stratified 5-fold cross-validation. As shown, our multi-k RF model matches or surpasses DNABERT's performance, while being far less computationally demanding. This is very encouraging, as it demonstrates

DRAFT

that traditional machine learning methods, when combined with multi-scale k-mer features, can achieve competitive accuracy and reduce computational cost significantly, potentially lowering barriers for extensive genomic studies.

Conclusion and future work

This study presents a novel approach for extending FCGR with positional information (PFCGR) by incorporating statistical measures of k-mer positions, such as mean, standard deviation, skewness, and kurtosis. In this way, our method improves the standard FCGR by adding layers of positional information, providing a richer set of features for the classifiers. We combine this approach with machine-learning techniques, such as Logistic Regression (LR), Random Forest (RF) and Convolutional Neural Networks (CNN), to classify lncRNAs of seven crop species. Our findings reveal that the proposed method not only competes with, but in some instances surpasses the much more complex DNABERT model in both accuracy and computational efficiency.

In addition, our experiments across multiple plant species illustrate the robustness and adaptability of our approach. Notably, the RF and the multi-channel CNN have shown significant promise, leading to high accuracy and improved classification performance. However, challenges remain. The main limitation is related to the potential overfitting when using the PFCGR model with CNN, especially with a limited dataset size. However, applying our approach to other types of sequences and integrating more diverse datasets could help validate and potentially improve the generalizability of our findings.

Lastly, we want to underline that this study not only advances the computational tools available for genomic research, but also contributes to our understanding of lncRNA functions and their regulatory roles in plants. By providing a more efficient and accurate tool for lncRNA identification, we pave the way for discoveries in plant biology and improve the prospects for agricultural innovation. In future work, we can focus on exploring alternative neural network architectures or incorporating gapped k-mers, which could help improve accuracy and generalizability by accommodating sequencing errors or variations.

ACKNOWLEDGEMENTS

This work was supported by the National Institutes of Health (NIH) under grant number 5R01HG011795.

Bibliography

1. Monica F. Danilevicz, Mitchell Gill, Cassandria G. Tay Fernandez, Jakob Petereit, Shriprabha R. Upadhyaya, Jacqueline Batley, Mohammed Bennamoun, David Edwards, and Philipp E. Bayer. DNABERT-based explainable lncRNA identification in plant genome assemblies. *Comput. Struct. Biotechnol. J.*, 21, 2023.
2. Jingjing Jin, Peng Lu, Yalong Xu, Zefeng Li, Shizhou Yu, Jun Liu, Huan Wang, Nam-Hai Chua, and Peijian Cao. PLncDB V2.0: a comprehensive encyclopedia of plant long non-coding RNAs. *Nucleic Acids Res.*, 49(D1), 2021.
3. Huan Wang, Pil Joong Chung, Jun Liu, In-Cheol Jang, Michelle J. Kean, Jun Xu, and Nam-Hai Chua. Genome-wide identification of long noncoding natural antisense transcripts and their responses to light in *Arabidopsis*. *Genome Res.*, 24(3), 2014.
4. Yu-Chan Zhang, Jian-You Liao, Ze-Yuan Li, Yang Yu, Jin-Ping Zhang, Quan-Feng Li, Liang-Hu Qu, Wen-Sheng Shu, and Yue-Qin Chen. Genome-wide screening and functional analysis identify a large number of long noncoding RNAs involved in the sexual reproduction of rice. *Genome Biol*, 15(12), 2014.
5. Julia A. Chekanova, Brian D. Gregory, Sergei V. Reverdatto, Huaming Chen, Ravi Kumar, Tanya Hooker, Junshi Yazaki, Pinghua Li, Nikolai Skiba, Qian Peng, Jose Alonso, Vladimir Brukhin, Ueli Grossniklaus, Joseph R. Ecker, and Dmitry A. Belostotsky. Genome-Wide High-Resolution Mapping of Exosome Substrates Reveals Hidden Features in the Arabidopsis Transcriptome. *Cell*, 131(7), 2007.
6. Hanna Lee, Seong Jeon Yoo, Jeong Hwan Lee, Wanhui Kim, Seung Kwan Yoo, Heather Fitzgerald, James C. Carrington, and Ji Hoon Ahn. Genetic framework for flowering-time regulation by ambient temperature-responsive miRNAs in Arabidopsis. *Nucleic Acids Res.*, 38(9), 2010.
7. Saquib Waheed and Lihui Zeng. The Critical Role of miRNAs in Regulation of Flowering Time and Flower Development. *Genes*, 11(3), 2020.
8. Flávia Thiebaut, Cristian A Rojas, Clícia Grativol, Mariana Romeiro Motta, Tauan Vieira, Michael Regulski, Robert A Martienssen, Laurent Farinelli, Adriana S Hemerly, and Paulo Cg Ferreira. Genome-wide identification of microRNA and siRNA responsive to endophytic beneficial diazotrophic bacteria in maize. *BMC Genomics*, 15(1), 2014.
9. Flávia Thiebaut, Cristian A. Rojas, Karla L. Almeida, Clícia Grativol, Giselli C. Domiciano, Caren Regina C. Lamb, Janice De Almeida Engler, Adriana S. Hemerly, and Paulo C. G. Ferreira. Regulation of miR319 during cold stress in sugarcane. *Plant Cell & Environment*, 35(3), 2012.
10. Raquel Salvador-Guirao, Patricia Baldrich, Detlef Weigel, Ignacio Rubio-Somoza, and Blanca San Segundo. The MicroRNA miR773 Is Involved in the *Arabidopsis* Immune Response to Fungal Pathogens. *MPMI*, 31(2), 2018.
11. Qing Liu, Tifeng Yang, Ting Yu, Shaohong Zhang, Xingxue Mao, Junliang Zhao, Xiaofei Wang, Jingfang Dong, and Bin Liu. Integrating Small RNA Sequencing with QTL Mapping for Identification of miRNAs and Their Target Genes Associated with Heat Tolerance at the Flowering Stage in Rice. *Front. Plant Sci.*, 8, 2017.
12. Guang Hu, Mengyan Hao, Le Wang, Jianfen Liu, Zhennan Zhang, Ye Tang, Qingzhong Peng, Zuoren Yang, and Jiahe Wu. The Cotton miR477- *CBP60A* Module Participates in Plant Defense Against *Verticillium dahlia*. *MPMI*, 33(4), 2020.
13. Jae Bok Heo and Sibum Sung. Vernalization-Mediated Epigenetic Silencing by a Long Intronic Noncoding RNA. *Science*, 331(6013), 2011.
14. Sònia Guil and Manel Esteller. Cis-acting noncoding RNAs: friends and foes. *Nat Struct Mol Biol*, 19(11), 2012.
15. Maria Clara De Oliveira Urquiaga, Flávia Thiebaut, Adriana Silva Hemerly, and Paulo Cavalcanti Gomes Ferreira. From Trash to Luxury: The Potential Role of Plant LncRNA in DNA Methylation During Abiotic Stress. *Front. Plant Sci.*, 11, 2021.
16. Ying Wang, Xiaojin Luo, Fan Sun, Jianhua Hu, Xiaojun Zha, Wei Su, and Jinshui Yang. Overexpressing lncRNA LAIR increases grain yield and regulates neighbouring gene cluster expression in rice. *Nat Commun*, 9(1), 2018.
17. Jun Fang, Fantao Zhang, Hongru Wang, Wei Wang, Fei Zhao, Zijuan Li, Changhui Sun, Faming Chen, Fan Xu, Shuoqi Chang, Liang Wu, Qingyun Bu, Pingrong Wang, Jiankun Xie, Fan Chen, Xuehui Huang, Yijing Zhang, Xinguang Zhu, Bin Han, Xiaojian Deng, and Chengcai Chu. *Ef-cd* locus shortens rice maturity duration without yield penalty. *Proc. Natl. Acad. Sci. U.S.A.*, 116(37), 2019.
18. Hsiao-Lin V. Wang and Julia A. Chekanova. Long Noncoding RNAs in Plants. In M.R.S. Rao, editor, *Long Non Coding RNA Biology*, volume 1008, pages 133–154. Springer Singapore, Singapore, 2017. Series Title: Advances in Experimental Medicine and Biology.
19. Yu-Jian Kang, De-Chang Yang, Lei Kong, Mei Hou, Yu-Qi Meng, Liping Wei, and Ge Gao. CPC2: a fast and accurate coding potential calculator based on sequence intrinsic features. *Nucleic Acids Res.*, 45(W1), 2017.
20. H. Busra Cagirici, S. Galvez, Taner Z. Sen, and Hikmet Budak. LncMachine: a machine learning algorithm for long noncoding RNA annotation in plants. *Funct Integr Genomics*, 21 (2), 2021.
21. Liguo Wang, Hyun Jung Park, Surendra Dasari, Shengqin Wang, Jean-Pierre Kocher, and Wei Li. CPAT: Coding-Potential Assessment Tool using an alignment-free logistic regression model. *Nucleic Acids Res.*, 41(6), 2013.
22. Urminder Singh, Niraj Khemka, Mohan Singh Rajkumar, Rohini Garg, and Mukesh Jain. PLncPRO for prediction of long non-coding RNAs (lncRNAs) in plants and its application for discovery of abiotic stress-responsive lncRNAs in rice and chickpea. *Nucleic Acids Res.*, 45(22), 2017.
23. Aimin Li, Junying Zhang, and Zhongyin Zhou. PLEK: a tool for predicting long non-coding RNAs and messenger RNAs based on an improved k-mer scheme. *BMC Bioinformatics*, 15(1), 2014.
24. Cong Pian, Guangle Zhang, Zhi Chen, Yuanyuan Chen, Jin Zhang, Tao Yang, and Liangyun Zhang. LncRNApred: Classification of Long Non-Coding RNAs and Protein-Coding Transcripts by the Ensemble Algorithm with a New Hybrid Feature. *PLoS ONE*, 11(5), 2016.
25. Xiao-Nan Fan and Shao-Wu Zhang. lncRNA-MFDL: identification of human long non-coding RNAs by fusing multiple features and using deep learning. *Mol. BioSyst.*, 11(3), 2015.
26. Gokcen Eraslan, Ziga Avsec, Julien Gagneur, and Fabian J. Theis. Deep learning: new computational modelling techniques for genomics. *Nat Rev Genet*, 20(7), 2019.
27. Ajay Shrestha and Ausif Mahmood. Review of Deep Learning Algorithms and Architectures. *IEEE Access*, 7, 2019.
28. Michał Wojciech Szcześniak, Oleksii Bryzghalov, Joanna Ciomborowska-Basheer, and Izabela Makałowska. CANTATAdb 2.0: Expanding the Collection of Plant Long Noncoding RNAs. In *Plant Long Non-Coding RNAs*, volume 1933, pages 415–429. Springer New York, New York, NY, 2019.
29. Andreu Paytuví Gallart, Antonio Hermoso Pulido, Irantzu Anzar Martínez de Lagrán, Walter Sanseverino, and Riccardo Aiese Cigliano. GREENC: a Wiki-based database of plant lncRNAs. *Nucleic Acids Res*, 44(D1), 2016.
30. Marco Di Marsico, Andreu Paytuvi Gallart, Walter Sanseverino, and Riccardo Aiese Cigliano. GreeNC 2.0: a comprehensive database of plant long non-coding RNAs. *Nucleic Acids Res.*, 50(D1), 2022.
31. Ajeet Singh, A. T. Vivek, and Shailesh Kumar. AlnC: An extensive database of long non-coding RNAs in angiosperms. *PLoS ONE*, 16(4), 2021.
32. Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome.

DRAFT

Bioinformatics, 37(15), 2021.
33. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv*, 2018.
34. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. *arXiv*, 2017.
35. Jing Li, Xuan Zhang, and Changning Liu. The computational approaches of lncRNA identification based on coding potential: Status quo and challenges. *Comput. Struct. Biotechnol. J.*, 18, 2020.
36. H.Joel Jeffrey. Chaos game representation of gene structure. *Nucleic Acids Res.*, 18(8), 1990.
37. Hannah Franziska Löchel and Dominik Heider. Chaos game representation and its applications in bioinformatics. *Comput. Struct. Biotechnol. J.*, 19, 2021.
38. Muhammed S. Hammad, Vidan F. Ghoneim, Mai S. Mabrouk, and Walid I. Al-atabany. A hybrid deep learning approach for COVID-19 detection based on genomic image processing techniques. *Sci Rep*, 13(1), 2023.
39. Jorge Avila Cartes, Santosh Anand, Simone Ciccolella, Paola Bonizzoni, and Gianluca Della Vedova. Accurate and fast clade assignment via deep learning and frequency chaos game representation. *GigaScience*, 12, 2022.
40. Feriel Ben Nasr and Afef Elloumi Oueslati. A new automatic method for human coding and non-coding zones characterization and classification based on FCGR coding and CNN classifier. In *2022 6th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, pages 1–6, Sfax, Tunisia, 2022. IEEE.
41. Yunxiao Ren, Trinad Chakraborty, Swapnil Doijad, Linda Falgenhauer, Jane Falgenhauer, Alexander Goesmann, Anne-Christin Hauschild, Oliver Schwengers, and Dominik Heider. Prediction of antimicrobial resistance based on whole-genome sequencing and machine learning. *Bioinformatics*, 38(2), 2022.
42. Riccardo Rizzo, Antonino Fiannaca, Massimo La Rosa, and Alfonso Urso. Classification Experiments of DNA Sequences by Using a Deep Neural Network and Chaos Game Representation. In *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016*, pages 222–228, Palermo Italy, 2016. ACM.
43. Bruno De Medeiros, Liming Cai, Peter Flynn, Yujing Yan, Xiaoshan Duan, Lucas Marinho, Christiane Anderson, and Charles Davis. A universal DNA barcode for the Tree of Life. *EcoEvoRxiv*, 2024.
44. Fatemeh Alipour, Kathleen A. Hill, and Lila Kari. CGRclust: Chaos Game Representation for twin contrastive clustering of unlabelled DNA sequences. *BMC Genomics*, 25(1), 2024.
45. B. P. Welford. Note on a Method for Calculating Corrected Sums of Squares and Products. *Technometrics*, 4(3), 1962.
46. Imen Messaoudi, Afef Elloumi Oueslati, and Zied Lachiri. Wavelet analysis of frequency chaos game signal: a time-frequency signature of the C. elegans DNA. *J Bioinform Sys Biology*, 2014(1), 2014.
47. Aaron Sievers, Katharina Bosiek, Marc Bisch, Chris Dreessen, Jascha Riedel, Patrick Froß, Michael Hausmann, and Georg Hildenbrand. K-mer Content, Correlation, and Position Analysis of Genome DNA Sequences for the Identification of Function and Evolutionary Features. *Genes*, 8(4), 2017.

DRAFT