A new convolution model for effective bio-motif detection via rationally design the "black box"

Shen Jin[1,2,+], Jingyi Li[3,4,+], Yang Ding[3,4,*], and Ge Gao[3,4,*]

[1]Computational Biology Department,Carnegie Mellon University,5000 Forbes Ave, Pittsburgh, PA
[2]Department of Biological Sciences,Carnegie Mellon University,5000 Forbes Ave, Pittsburgh, PA
[3]Beijing Advanced Innovation Center for Genomics (ICG), Biomedical Pioneering Innovation Center (BIOPIC), Peking University, Beijing 100871, People's Republic of China.
[4]State Key Laboratory of Protein and Plant Gene Research, School of Life Sciences, Center for Bioinformatics, Peking University, Beijing 100871, People's Republic of China.
[+]Contribute equally
[*]Corresponding author

Abstract

Bio-motif detection is one of essential computational tasks for bioinformatics and genomics. Based on a theoretical framework for quantitatively modeling the relationship of convolution kernel shape and the motif detection effectiveness, we design and propose a novel convolution-based model, VCNN (Variable CNN), for effective bio-motif detection via the adaptive kernel length at runtime. Empirical evaluations based on both simulated and real-world genomics data demonstrate VCNN's superior performance to classical CNN in both detection power and hyper-parameter robustness. All source code and data are available at https://github.com/gao-lab/VCNN/ freely for academic usage.

1 Introduction

DNA (or RNA) motifs are generally defined as recurring sequence patterns from DNA (or RNA) sequences (Kulakovskiy and Makeev [2013], Achar and Sætrom [2015]). So far, such motifs are typically known in the form of "conserved" segments, i.e., fixed-length segments with some or all positions occupied by fixed nucleotides (Kulakovskiy and Makeev [2013]). A series of DNA (or RNA) motifs have been experimentally confirmed to exert or regulate important biological functions, including protein-binding (Stormo [2015]), transcription initiation (Kadonaga [2012]), alternative splicing (Blencowe [2000]), subcellular localization (Zhang et al. [2014]), translation efficiency (Zucchelli et al. [2015]), and serving as microRNA's target (Thomson and Dinger [2016]). In addition, many motifs have been shown to be closely related to a series of complex diseases, such as cancer (Wolfe et al. [2014]). Precisely profiling major DNA and RNA motifs is thus critical to the understanding of how they work and how they are related to various diseases.

Traditional motif discovery algorithms are typically completely heuristic. For example, the well-known MEME software (Bailey and Elkan [1994]) is dedicated to handle a set of arbitrary unlabeled DNA/RNA sequences to discover motifs that can be described in Position Weight Matrix (PWM), where the length of the expected motif is predefined, and the probability distribution of four nucleotides on each position of this motif is independent of all other positions; in this way it can also be regarded as a zero-order Markov chain. DREME (Bailey [2011]) goes more stringent in this direction by allowing fixed segments (i.e., the k-mers) only. CentriMo (Bailey and Machanick [2012]), which is designed specifically for finding motifs from ChIP-Seq peaks, assumes instead that all input sequences should be of equal length and centered at the peaks. On the other hand, HMMER

Preprint. Work in progress.

(Finn et al. [2011]) assumes that motifs can be better described by Hidden Markov Model (HMM). The RNA structural motif-finding software Infernal (Nawrocki and Eddy [2013]) complexified it even further by introducing a variant of HMM called Stochastic Context-Free Grammar (SCFG), allowing the motif to have varied length while still conserved in secondary structures. Nevertheless, in this paper we focus on PWM-form motifs only.

While these heuristic algorithms have been used extensively, they do not scale well with the explosion of high-throughput sequencing datasets these days. This is where supervised motif discovery algorithms, benefiting from fast-growing deep learning techniques, come into a play. Alipahani et al. developed the first Convolutional Neural Network (CNN) , DeepBind, to discover protein binding motifs from DNA and RNA sequences (Alipanahi et al. [2015b]). This CNN network treats each convolutional kernel as an individual motif scanner, extracts the major signal by global max- or average-pooling, and predicts whether the input sequence binds to the protein in question by a simple logistic regression or a two-layer neural network on such signals from all kernels. Most, if not all, subsequent CNN models for DNA/RNA sequences have followed DeepBind's kernel-as-motif-scanner strategy and successfully handled various DNA/RNA problems with huge volumes of input datasets (e.g., Zhou et al. [2018], Kelley et al. [2018], Angermueller et al. [2017], Wang et al. [2018], Zhang et al. [2018]).

However, although being fast and decently accurate, all these supervised models still suffer from a subtle problem common in the deep learning field, but not in the traditional motif discovery algorithms: they are inherently unable to learn the motif length by themselves, because the length of kernels – the motif scanners – is predefined. Here we first confirmed that this subtle problem is, in fact, severe to the performance of CNN model; the CNN model will not only become inaccurate when kernels and motifs are mismatched in length, but also fail to discover mixed motifs of different lengths from the same input sequences. We then presented our solution, VCNN (Variable CNN), where the kernel is designed to learn from its own Information Content (IC) the real motif length autonomously during training. We showed empirically that VCNN is robust to the initial kernel length and capable of discovering mixed motifs effectively from the input sequences. All source code and data are available at https://github.com/gao-lab/VCNN/ freely for academic usage.

2 Theoretical Analysis

2.1 Overview

In this section, we introduce our theoretical framework. In brief, we derive a scoring function to quantify the "goodness" of a kernel length to detect a specific bio-motif. The main conclusion is that classical CNN with fixed kernel length cannot detect bio-motifs effectively. This intrinsic limitation motivated us to develop a new convolutional model, called VCNN (Variable CNN). This section is divided into three parts. First, we present the mathematical definitions in bio-motif detection task. Then we introduce the scoring function which measures the "goodness of kernel length". Finally, we describe the algorithm for implementing VCNN, a novel convolution model with adaptable kernel length.

2.2 Mathematical definition in bio-motif detection

In bio-motif detection, we start with bio-sequences and their matched biology function. Our ultimate goal is to discover motifs among the sequences, which are statistically important for a sequence to play its function. These motifs are potential biology functional elements and this discovery is critical to understand the underlying biology processes. Currently, we have following prior knowledge about Bio-motifs: they are short, recurring patterns which are presumed to be associated with particular biological functions (D'haeseleer [2006]).

The representation of Bio-sequences and their functions Bio-sequences consist of various alphabets, such as the A, C, G, T for DNA and A, C, G, U for RNA. In our case, we use $x^i \in R^{L_s \times 4}$ to represent a sequence, where L_s is the length of sequence and each column x^i_j is one-hot representation of the jth nucleotides in x_i. Each sequence has an integral label y^i referring to its function.

The representation of bio-motif We are going to use PWM (position weight matrix), the most common representation used by biologists (Sinha [2006], Xia [2012]), to represent bio-motif. Formally, each motif is a 2D matrix: $\mathcal{M}$, the jth column $\mathcal{M}_j$ represents the probability of each nucleotide at the jth position, where $\sum_{k=0}^{3} \mathcal{M}_j[k] = 1$ and $\mathcal{M}_j^i[k] \not< 0$.

2.3 Quantify the "goodness" of kernel length

We proposed our scoring function in the 1(b),the detailed formulas are shown in the Supplementary2.1.

2.3.1 The statistic model behind the scoring function

In this part, we introduce a novel statistic model of CNN. Then we further explain how to derive the scoring function using the statistic model. The overall logical flow refers to figure 1

The statistic model of CNN Here we focus on a specific architecture of CNN: a convolutional layer followed by a global max-pooling layer. We note that the only input nodes that contribute to kernel's update during backpropagation is those whose convolutional value is maximal across all convolutional values, and thus passed to the global max-pooling layer. Therefore, we calculate a statistic about the ratio of "useful" information that can be transmitted to the following neural layer. First, we assume that sequences with signal motif $\mathcal{M}$ are drawn from the following distribution:

$$\mathcal{S} = [\alpha_1, \alpha_2, ..., \alpha_j, \mathcal{M}, \gamma_{j+l}, ..., \gamma_n]$$

Each column in α_i and γ_k belongs to the background distribution, $\mathcal{B}$, which is a multinomial distribution. Without prior knowledge, we assume that there is no bias in type distribution of nucleotides and set $\mathcal{B} = (0.25, 0.25, 0.25, 0.25)$. Then we model the relationship between kernel $\mathcal{K}$ and $\mathcal{S}$. For convenience, we write L_k-length subsequence of x_i as: $\S_i[j] = x_i[j_i : j_i + L_k]$. $\mathcal{K}$ is initialized randomly and updated according to the gradient. Therefore, the update of $\mathcal{K}$ with length L_k can be modeled as: Specifically,

$$\mathcal{K}_{t+1} = \mathcal{K}_t + \sum_{x_i \in batch} w_i * \S_i[j_i], \quad where \quad j_i = \underset{j=1,...,L_k}{\arg\max} \{\mathcal{K}_t \star x_i\} \qquad (*1)$$

Where w_i is the weight when calculating the gradient, $\star$ is cross-correlation operator and * is the multiplication operator. Note that the change of kernel value is largely determined by the subsequences $\S_i[j]$ that have the largest convolution score. The higher percentage of $\S_i[j]$ that "falls" within the territory of $\mathcal{M}$ is, the better the kernel $\mathcal{K}$ will be learned to detect the motif $\mathcal{M}$. Therefore, we construct $\mathcal{K}_t$ using a parameter P_{ideal}, which means the percentage of $\S_i[j]$ that "falls" within the territory of $\mathcal{M}$ in previous state as following:

$$\mathcal{K} = P_{ideal} * \tilde{\mathcal{M}}^i + (1 - P_{ideal}) * \mathcal{R} \qquad (2)$$

Where $\tilde{\mathcal{M}}^i$ is a cut- or padded-version of $\mathcal{M}^i$ (so that it matches the size of $\mathcal{K}$; details are available from the Supplementary document, section 2). And $\mathcal{R}$ is a 2D matrix of length L_k. Each column is drawn from a distribution derived from $\mathcal{B}$. Then we calculated statistic about P_{real}, which refers to the percentage of "right" subsequence obtained by $\mathcal{K}_t$.

Formal definition of the scoring function Formally, we define the scoring function as the expectation of P_{real}:

$$Score(L_k | P_{ideal}, \mathcal{M}) = Expectation_{\mathcal{K}}(P_{real})$$

Where P_{real} is defined as following:

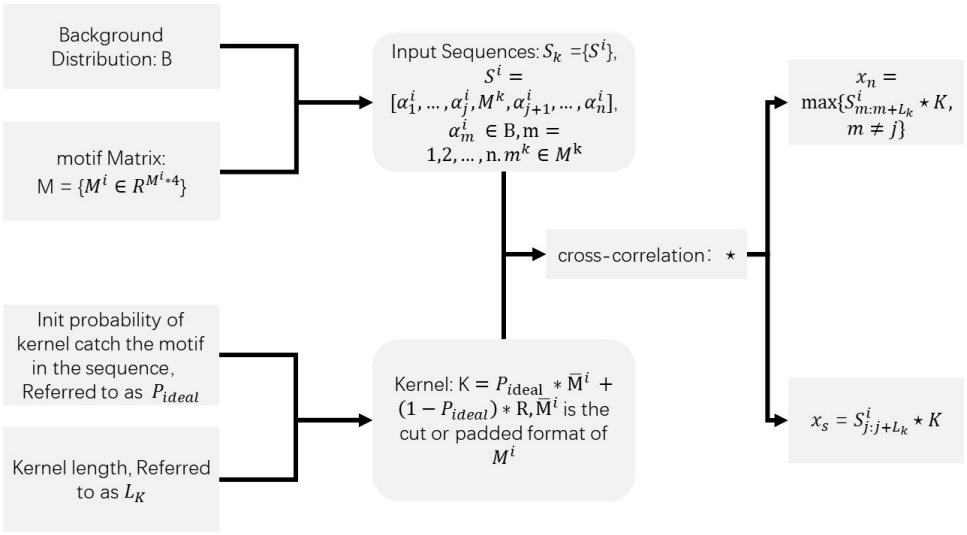
$$P_{real} = Prob(X_s > X_n | \mathcal{K}, \mathcal{M})$$

Because the value of kernel is a random variable, P_{real} can also be regarded as a random variable which is defined by the above function.
And X_s, X_n are two random variables, defined as following:

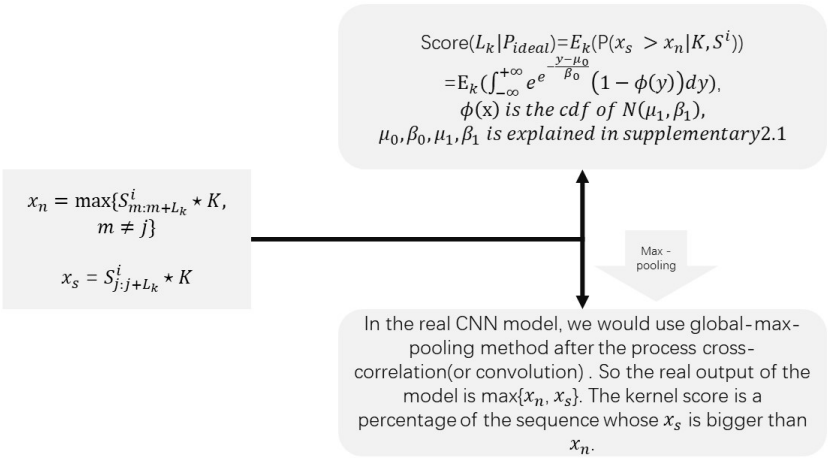
$$X_s = \mathcal{S}^i_{\bar{j}:\bar{j}+L_k} \star \mathcal{K}$$

$$X_n = \max\left\{\mathcal{S}^i_{j:j+L_k} \star \mathcal{K}, j \neq \bar{j}\right\}$$

And $\bar{j}$ is the starting position of the true motif in the sequence $\mathcal{S}^i$ predicted by $\mathcal{K}$. Since X_n can be regarded as the maximum of many i.i.d random variables, we used extreme value distribution to approximate X_n(details refer to the Supplementary document, section 2)



(a)



(b)

Figure 1: The logic flow of kernel scoring. There are 3 main steps to calculate the score for the kernel. There are 2 steps in figure (a), and the last one is in figure (b). First, we build the simulated sequences and the kernel which are used for scoring. Then we used cross-correlation product to obtain the x_n and x_s, which are highest volume number in background part and the signal part in sequences, respectively. Finally, we used extreme value distribution to define the score of the kernel for the specific signal motif. And in the figure (b), the parameters below are as defined. L_k is kernel length. $\mathcal{M}^i$ is a PWM of a given bio-motif. The function takes L_k,$\mathcal{M}^i$ and P_{ideal} as input and yields a score, called $P_{real} \in (0, 1)$, which measures the "goodness" of L_k to detect $\mathcal{M}^i$. More details are available from Section 2.1 of Supplementary document.

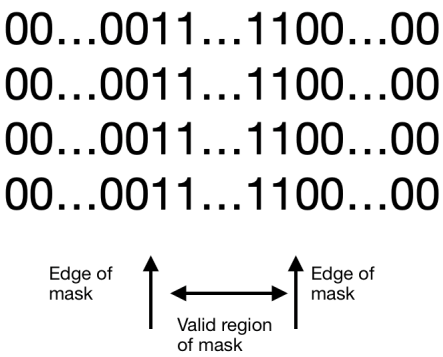


Figure 2: schematic diagram of a mask

2.4 VCNN Design and Implementation

The implementation of Variable CNN (VCNN) can be separated into two parts: first making the kernel length changeable at runtime, and then designing an algorithm to decide when and how to change. For the first part, we use a "mask" to cover the kernel.(A schematic diagram of a Mask refers to fig 2) "Mask" is a matrix consisting of ones and zeros only. The mask is applied to the kernel by Hadamard product; in this way, only those kernel elements multiplied by ones will contribute to the subsequent convolution. Therefore, we can change the valid kernel length by changing the position of ones in the "mask". For the second part, we design an algorithm based on Information Content (IC). IC measures the information encoded in the kernel. If a kernel's edge is "rich" in information in the sense that the IC-per-position is always high around this edge, we can judge that this part has captured some meaningful bio-motifs in the dataset. Moreover, it is likely that the edges did not delineate the whole motif. Therefore, we should widen the valid kernel length by moving the edges away from the kernel center. On the other side, if the information is not high enough, then the region around this edge did not overlap the underlying motif, and we should narrow the valid kernel length instead by shifting the edges towards the kernel center. Details about implementation are available from the Supplementary Document, section 3.

3 Experimental Results

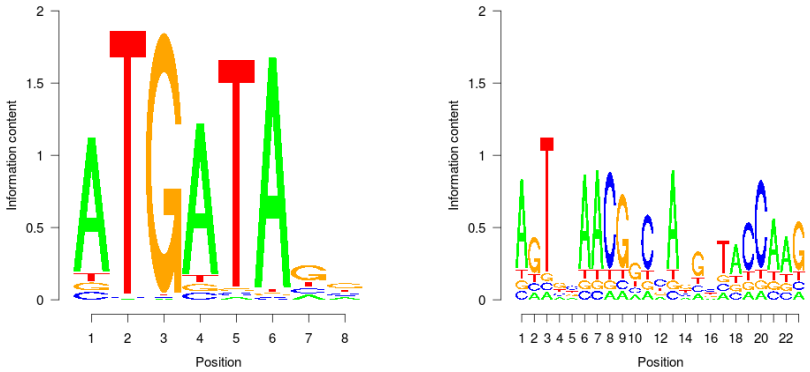
In this part, we first confirmed that the score we proposed mirrors the model performance on simulation datasets. Then we demonstrated the out-performance of our VCNN model over classical CNN on simulated datasets, w.r.t. model performance and motif recovery. Finally, we confirmed this out-performance w.r.t. model performance further on real-world dataset (i.e., the benchmark datasets from Alipanahi et al. [2015a]).

3.1 Settings for simulation datasets

We first constructed two motifs of different lengths and conservation (fig 3) where the first one, "Len-8_IC-10" (fig 3(a)) is shorter and more conserved than the second one, "Len-23_IC-12.png" (fig 3(b)). Then we simulated the following three sequences datasets: (1) "simulation dataset 01", where each sample is a random sequence inserted with a single fragment generated from the first motif; (2) "simulation dataset 02", which is similar to "simulation dataset 01" but with the fragment generated from the second motif; and (3) "simulation dataset 03", where each sample is a random sequence inserted with a fragment generated from either the first motif or the second, but not both.

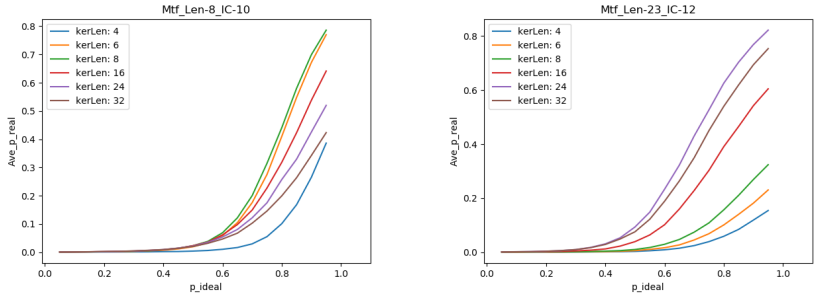
3.2 The score mirrors the model performance on simulated datasets

The score is parameterized by $P_{ideal} \in [0, 1]$ and the kernel length L_k. Therefore, we can plot the score's working curves directly, as shown in fig 4. Should the score mirror the model performance, we would expect the following observations in simulation:



(a) Len-8_IC-10 (b) Len-23_IC-12

Figure 3: Two motifs used for simulation datasets. (a) is shorter and more conserved than (b).



(a) Theoretical scoring for Len-8_IC-10 (b) Theoretical scoring for Len-23_IC-12

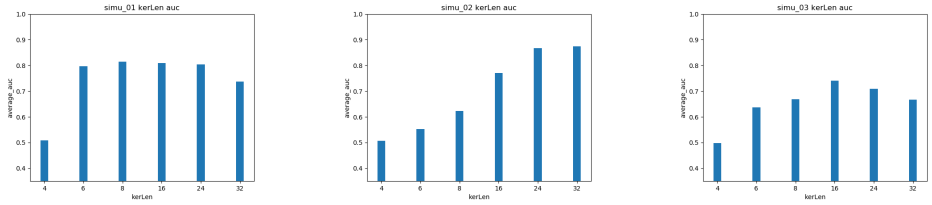
Figure 4: Theoretical prediction scores for different kernel length w.r.t different motifs. Working curves for the score w.r.t. P_{ideal} and the kernel length L_k (here denoted by "kerLen"). Higher the curve, better the model performance predicted in theory. (a) is for the motif Len-8_IC-10, and (b) for the motif Len-23_IC-12.

Expected observation 1: The best-performing kernel length for learning "Len-8_IC-10", the shorter and more conserved motif, is 8, because it has the highest curve in fig 4(a).

Expected observation 2: The best-performing kernel length for learning "Len-23_IC-12", the longer and less conserved motif, is 24, because it has the highest curve in fig 4(b).

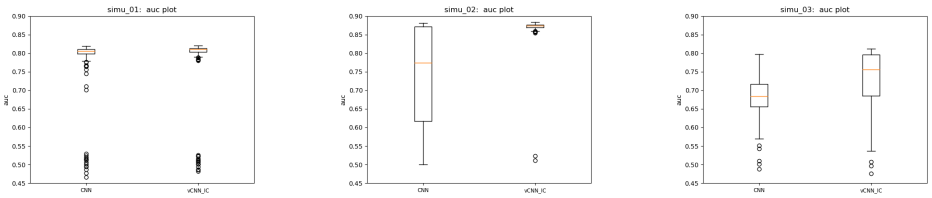
Expected observation 3: Learning the motif "Len-23_IC-12" is more sensitive to changes in kernel length than learning "Len-8_IC-10", because the former's curves (fig 4(b)) are more separated from each other (i.e., its performance drops more when the kernel length deviates from the best-performing one).

We then examined them by comparing the performance, defined as the average AUC value here, between different kernel lengths on each simulation dataset (see Supplementary Document for details of model structure and training). The first observation was confirmed by fig 5(a). The second one, however, was not well supported by fig 5(b); the 32-length kernel slightly outperformed the 24-length one. It reveals some intrinsic limitation of the scoring function, which will be further discussed in the Supplementary Document. And the last observation is confirmed by comparing the between fig 5(a) and fig 5(b), which indicates that the dataset with motif "Len-23_IC-12" is more sensitive to different kernel lengths. More detailed analyses on different kernel numbers refer to Supplementary Document, section 5.1.



(a) Simulation dataset 01 (b) Simulation dataset 02 (c) Simulation dataset 03

Figure 5: Empirical results on simulation dataset matches the theoretical prediction. Model empirical performance is defined by the average AUC. Figure (a), (b), and (c) are for simulation datasets 01, 02, and 03 respectively, which contains the motifs mentioned before. Kernel lengths are selected from 4,6,8,16,24,32, corresponding to the kernel lengths evaluated theoretically before.



(a) Simulation dataset 01 (b) Simulation dataset 02 (c) Simulation dataset 03

Figure 6: Comparing with CNN, VCNN has a more robust performance in simulation dataset 02, a superior performance in simulation dataset 03 (which contains multiple motifs) and a similar performance in dataset 01. Each boxplot is a summary of AUC across models with different kernel initial length. This result indicates that VCNN

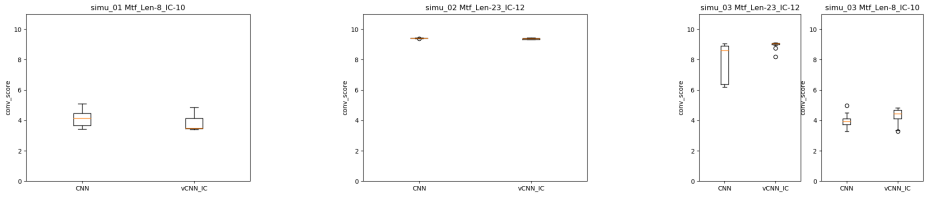
3.3 VCNN outperforms classical CNN

Having unveiled the problems of classical CNN on kernel length, we demonstrated now that our VCNN model can handle this more properly. Here the VCNN model is exactly the same in structure and training with the classical CNN model to compare with, except for its exclusive VCNN kernels. As shown in fig 6(a), 6(b) and 6(c), VCNN obtained an AUC similar or better than that of classical CNN across different kernel length setups. Specifically, we note that the out-performance is considerably larger for simulation dataset 03 than others, suggesting that VCNN, but not the kernel length-fixed classical CNN, can handle those cases with mixed-length motifs properly.

Besides models‘ prediction power, their fidelity to recover motifs is also important. By discovering motifs faithfully, models can further help biologists to get insights of the underlying biological processes. In order to investigate different models' ability to recover motifs, we compared the similarity between the real motif and the recovered one. For each model, the similarity score was calculated with respect to different motifs. In each dataset, only those models whose AUC is among the top 5% (i.e., likely to be selected by grid search in practice) had their motifs compared. The results were shown in fig 7. Note that in simulation datasets with single motif (01 and 02) (fig 7(a)7(b)), CNN and VCNN have similar similarity scores. However, in simulation dataset 03, where two different motifs are mixed, VCNN outperformed CNN (fig 7(c)). This result indicated that VCNN can recover the motifs more faithfully when different motifs are mixed altogether.

3.4 Result on real-world dataset

We next assessed the performance on real-world dataset. In Alipanahi et al. [2015a]'s work, the positive dataset was set as the 690 ENCODE ChIP-Seq datasets for transcription factors and other DNA-binding proteins, and the negative dataset was created by shuffling the positive dataset. We downloaded all these data from "http://cnn.csail.mit.edu". We then compared the motif detection effectiveness of our proposed VCNN with the reported benchmark results Alipanahi et al. [2015a]'s.



(a) Simulation dataset 01 (b) Simulation dataset 02 (c) Simulation dataset 03

Figure 7: VCNN can recover the motifs more effectively than CNN, especially when dataset contains multiple motifs. Each boxplot describes the similarity score distribution of models whose AUC is among the five largest ones for the dataset in question.

fig 8(b) shows that VCNN has a overall better performance than the classical CNN. Figure 8(a) shows that for 70 of all transcription factor datasets (marked in red circle), VCNN has a significant improvement in AUC. Statistically, VCNN significantly improves classical CNN's AUC (single-tailed Wilcoxon Rank Sum test p-value=1.624×10^{-13}; the detailed list of AUC improvement for each dataset is enclosed as Supplementary Table 1: AUC comparison between VCNN and deepbind).

We analyzed 36 datasets with poor VCNN performance (when AUC difference is larger than 0.02) and found that these datasets have a common feature — their sample size is small, fig 8(c). And on the dataset where vCNN is considerably less effective than DeepBind (AUC is lower by 0.1), we found that VCNN's kernel has basically no IC-rich positions (see Supplementary Document, section 5.3 for more details). Therefore, we would not expect the model to discover any conservative motifs in these dataset.

4 Discussion

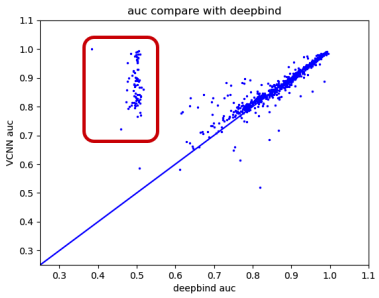
In this paper, we first proposed a scoring method to quantify the detection power of a given kernel length for a particular bio-motif. Then we designed and implemented a new convolution model, VCNN, which adaptively tunes kernel lengths at runtime. The experimental results based on both simulated and real-world dataset (Alipanahi et al. [2015a]) showed that VCNN outperforms the classical CNN in both motif detection power and accuracy.

Several efforts have been done to understand how the model's architecture influences its ability to detect patterns, with main focus on the potential of a model to fit a function (Hornik [1991]). However, our work pinpoints that the relationship between kernel length and performance strongly depends on the patterns in data. Therefore, it will be impossible to decide the kernel length beforehand. And our heuristic model VCNN is one of the first attempt to address this problem by adapting the kernel length at runtime.

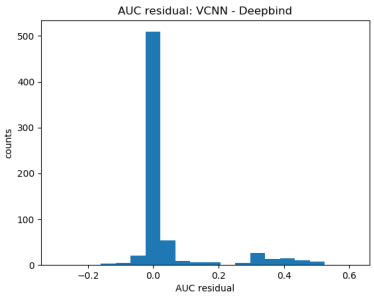
We noticed that our quantitative model provides an explanation for the observation that some datasets are more sensitive to the kernel length than others, as shown by Sun et al. [2016]. Additionally, we showed that our the scoring curve is strongly related to the model's performances. In the Supplementary Document, section 5.3, we discuss the relationship between the area under scoring curve and model's average AUC. The result indicated that the area is (closely) related to the AUC in a way independent of the dataset in question. Therefore, our scoring function is a good indicator that can judge the goodness of kernel length regardless of the differences in dataset. However, to calculate the scoring function, one has to know the motif within the dataset beforehand. This will not be possible in real-world data. This is a major limitation of the scoring function, which will need future efforts to overcome.

References

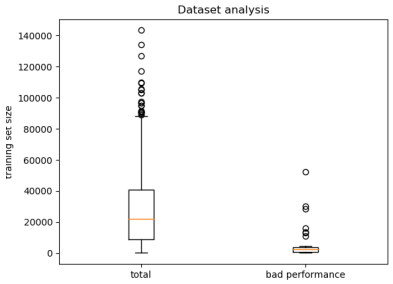
Avinash Achar and Pål Sætrom. RNA motif discovery: a computational overview. *Biology Direct*, 10(1), December 2015. ISSN 1745-6150. doi: 10.1186/s13062-015-0090-5. URL `http://www.biologydirect.com/content/10/1/61`.



(a) Result on Deepbind's dataset



(b) AUC improvement on Deepbind's dataset



(c) Dataset Size

Figure 8: VCNN out performance "Deepbind", the benchmark model, on real world datasets. (a) and (b) are the performance comparison between the DeepBind model and the VCNN model on the real-world dataset; Marked in red circle, 70 of all transcription factor datasets, VCNN has a significant improvement in AUC, comparing with CNN. (c) is the comparison of sample size between the datasets with insufficient performance of VCNN and all 690 datasets, when AUC difference is larger than 0.02

Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, 33(8):831, 2015a.

Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nature Biotechnology*, 33(8):831–838, August 2015b. ISSN 1087-0156, 1546-1696. doi: 10.1038/nbt.3300. URL `http://www.nature.com/articles/nbt.3300`.

Christof Angermueller, Heather J. Lee, Wolf Reik, and Oliver Stegle. DeepCpG: accurate prediction of single-cell DNA methylation states using deep learning. *Genome Biology*, 18(1), December 2017. ISSN 1474-760X. doi: 10.1186/s13059-017-1189-z. URL `http://genomebiology.biomedcentral.com/articles/10.1186/s13059-017-1189-z`.

T. L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proceedings. International Conference on Intelligent Systems for Molecular Biology*, 2:28–36, 1994. ISSN 1553-0833.

Timothy L. Bailey. DREME: motif discovery in transcription factor ChIP-seq data. *Bioinformatics*, 27(12):1653–1659, June 2011. ISSN 1460-2059, 1367-4803. doi: 10.1093/bioinformatics/btr261. URL `https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btr261`.

Timothy L. Bailey and Philip Machanick. Inferring direct DNA binding from ChIP-seq. *Nucleic Acids Research*, 40(17):e128–e128, September 2012. ISSN 1362-4962, 0305-1048. doi: 10.1093/nar/gks433. URL `https://academic.oup.com/nar/article/40/17/e128/2411117`.

Benjamin J Blencowe. Exonic splicing enhancers: mechanism of action, diversity and role in human genetic diseases. *Trends in Biochemical Sciences*, 25(3):106–110, March 2000. ISSN 09680004. doi: 10.1016/S0968-0004(00)01549-8. URL http://linkinghub.elsevier.com/retrieve/pii/S0968000400015498.

Patrik D'haeseleer. What are dna sequence motifs? *Nature biotechnology*, 24(4):423, 2006.

R. D. Finn, J. Clements, and S. R. Eddy. HMMER web server: interactive sequence similarity searching. *Nucleic Acids Research*, 39(suppl):W29–W37, July 2011. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkr367. URL https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkr367.

Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2): 251–257, 1991.

James T. Kadonaga. Perspectives on the RNA polymerase II core promoter: Perspectives on the RNA polymerase II core promoter. *Wiley Interdisciplinary Reviews: Developmental Biology*, 1(1): 40–51, January 2012. ISSN 17597684. doi: 10.1002/wdev.21. URL http://doi.wiley.com/10.1002/wdev.21.

David R. Kelley, Yakir A. Reshef, Maxwell Bileschi, David Belanger, Cory Y. McLean, and Jasper Snoek. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Research*, 28(5):739–750, May 2018. ISSN 1088-9051, 1549-5469. doi: 10.1101/gr.227819.117. URL http://genome.cshlp.org/lookup/doi/10.1101/gr.227819.117.

Ivan V. Kulakovskiy and Vsevolod J. Makeev. DNA Sequence Motif. In *Advances in Protein Chemistry and Structural Biology*, volume 91, pages 135–171. Elsevier, 2013. ISBN 978-0-12-411637-5. doi: 10.1016/B978-0-12-411637-5.00005-6. URL http://linkinghub.elsevier.com/retrieve/pii/B9780124116375000056.

E. P. Nawrocki and S. R. Eddy. Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics*, 29(22):2933–2935, November 2013. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/btt509. URL https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btt509.

Saurabh Sinha. On counting position weight matrix matches in a sequence, with application to discriminative motif finding. *Bioinformatics*, 22(14):e454–e463, 2006.

Gary D. Stormo. DNA Motif Databases and Their Uses: DNA Motif Databases and Their Uses. In Alex Bateman, William R. Pearson, Lincoln D. Stein, Gary D. Stormo, and John R. Yates, editors, *Current Protocols in Bioinformatics*, pages 2.15.1–2.15.6. John Wiley & Sons, Inc., Hoboken, NJ, USA, September 2015. ISBN 978-0-471-25095-1. doi: 10.1002/0471250953.bi0215s51. URL http://doi.wiley.com/10.1002/0471250953.bi0215s51.

Zhun Sun, Mete Ozay, and Takayuki Okatani. Design of kernels in convolutional neural networks for image classification. In *European Conference on Computer Vision*, pages 51–66. Springer, 2016.

Daniel W. Thomson and Marcel E. Dinger. Endogenous microRNA sponges: evidence and controversy. *Nature Reviews Genetics*, 17(5):272–283, May 2016. ISSN 1471-0056, 1471-0064. doi: 10.1038/nrg.2016.20. URL http://www.nature.com/articles/nrg.2016.20.

Meng Wang, Cheng Tai, Weinan E, and Liping Wei. DeFine: deep convolutional neural networks accurately quantify intensities of transcription factor-DNA binding and facilitate evaluation of functional non-coding variants. *Nucleic Acids Research*, 46(11):e69–e69, June 2018. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gky215. URL https://academic.oup.com/nar/article/46/11/e69/4958204.

Andrew L. Wolfe, Kamini Singh, Yi Zhong, Philipp Drewe, Vinagolu K. Rajasekhar, Viraj R. Sanghvi, Konstantinos J. Mavrakis, Man Jiang, Justine E. Roderick, Joni Van der Meulen, Jonathan H. Schatz, Christina M. Rodrigo, Chunying Zhao, Pieter Rondou, Elisa de Stanchina, Julie Teruya-Feldstein, Michelle A. Kelliher, Frank Speleman, John A. Porco, Jerry Pelletier, Gunnar Rätsch, and Hans-Guido Wendel. RNA G-quadruplexes cause eIF4a-dependent oncogene translation in

cancer. *Nature*, 513(7516):65–70, September 2014. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature13485. URL http://www.nature.com/articles/nature13485.

Xuhua Xia. Position weight matrix, gibbs sampler, and the associated significance tests in motif characterization and prediction. *Scientifica*, 2012, 2012.

B. Zhang, L. Gunawardane, F. Niazi, F. Jahanbani, X. Chen, and S. Valadkhan. A Novel RNA Motif Mediates the Strict Nuclear Localization of a Long Noncoding RNA. *Molecular and Cellular Biology*, 34(12):2318–2329, June 2014. ISSN 0270-7306. doi: 10.1128/MCB.01673-13. URL http://mcb.asm.org/cgi/doi/10.1128/MCB.01673-13.

Juhua Zhang, Wenbo Peng, and Lei Wang. LeNup: learning nucleosome positioning from DNA sequences with improved convolutional neural networks. *Bioinformatics*, 34(10):1705–1712, May 2018. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/bty003. URL https://academic.oup.com/bioinformatics/article/34/10/1705/4796955.

Jian Zhou, Chandra L. Theesfeld, Kevin Yao, Kathleen M. Chen, Aaron K. Wong, and Olga G. Troyanskaya. Deep learning sequence-based ab initio prediction of variant effects on expression and disease risk. *Nature Genetics*, 50(8):1171–1179, August 2018. ISSN 1061-4036, 1546-1718. doi: 10.1038/s41588-018-0160-6. URL http://www.nature.com/articles/s41588-018-0160-6.

S Zucchelli, D Cotella, H Takahashi, C Carrieri, L Cimatti, F Fasolo, Mh Jones, D Sblattero, R Sanges, C Santoro, F Persichetti, P Carninci, and S Gustincich. SINEUPs: A new class of natural and synthetic antisense long non-coding RNAs that activate translation. *RNA Biology*, 12(8):771–779, August 2015. ISSN 1547-6286, 1555-8584. doi: 10.1080/15476286.2015.1060395. URL http://www.tandfonline.com/doi/full/10.1080/15476286.2015.1060395.