Representation learning of genomic sequence motifs via information maximization

Nicholas Keone Lee [1] **Peter K. Koo** [1]

Abstract

Convolutional neural networks (CNNs) trained to predict regulatory functions from genomic sequence often learn partial or distributed representations of sequence motifs across many first-layer filters, making it challenging to interpret the biological relevance of these models' learned features. Here we present Genomic Representations with Information Maximization (GRIM), an unsupervised learning method based on the Infomax principle that enables more comprehensive identification of whole sequence motifs learned by CNNs. By performing systematic experiments, we empirically demonstrate that GRIM is able to discover motifs in genomic sequences in situations where supervised learning struggles.

1. Introduction

Convolutional neural networks (CNNs) have shown strong successes in taking DNA sequences as input and predicting regulatory functions, such as transcription factor binding or mRNA abundance (Kelley et al., 2018; Tasaki et al., 2020; Avsec et al., 2021). To understand the features learned by a CNN, one can visualize first-layer convolutional filters (Alipanahi et al., 2015; Kelley et al., 2016), which has been shown to correspond to biologically relevant motifs, or use attribution methods (Simonyan et al., 2013; Shrikumar et al., 2018). Despite the state-of-the-art performance of deep learning, there is no guarantee that resulting visualizations or attribution scores will reveal biologically meaningful features (Koo & Eddy, 2019; Koo & Ploenzke, 2021).

Indeed, CNNs trained with standard supervised learning tend to learn fragile representations that may, on average, be correlated with training labels but that do not reflect the underlying data generating processes (Ilyas et al., 2019). While architecture design principles can encourage CNNs to learn more robust features (Koo & Eddy, 2019; Koo & Ploenzke, 2021), supervised deep learning suffers from two major shortcomings: first, it learns discriminative features that are predictive of class labels but may fail to capture all relevant features; and second, it may learn only a subset of strongly discriminative features and miss weakly correlated features that could still be biologically important.

Here we present Genomic Representations with Information Maximization (GRIM), an unsupervised learning method based on the Infomax principle that enables more comprehensive learning and identification of whole sequence motif representations by CNNs. We demonstrate that GRIM is able to discover motifs in genomic sequences in situations where supervised learning methods struggle.

2. Background

In contrast with supervised learning, unsupervised learning methods are not given access to labels, and thus can avoid some of the shortcuts and pitfalls that plague supervised learning. While there are many types of unsupervised and semi-supervised learning methods (Anand & Huang, 2018; Sinai et al., 2017; Lu et al., 2020), we have chosen to employ a method based on the information maximization (InfoMax) principle (Linsker, 1988). Under the Infomax principle, the goal of learning is to find an encoding g (constrained by a function class $\mathcal{G}$) of input X such that the Shannon mutual information (MI) between the pair X and $g(X)$ is maximized; specifically, this is

$$\max_{g \in \mathcal{G}} I\left(X; g\left(X\right)\right).$$

However, MI is a notoriously difficult quantity to estimate from data. Thus, in practice, most methods rely on maximizing a lower bound, which is easier to compute (Becker & Hinton, 1992), usually given according to:

$$\max_{g_1 \in \mathcal{G}_1, g_2 \in \mathcal{G}_2} I_{\text{est}}\left(g_1\left(X^{(1)}\right); g_2\left(X^{(2)}\right)\right)$$

where $I_{\text{est}}(X;Y)$ is a sample-based estimator, g_1 and g_2 are different encoders that take as input $X^{(1)}$ and $X^{(2)}$, two different views of the data X (Tschannen et al., 2019). For example, in computer vision, $X^{(1)}$ and $X^{(2)}$ could be the top and bottom halves of an image.

[1]Simons Center for Quantitative Biology, Cold Spring Harbor Laboratory; Cold Spring Harbor, NY. Correspondence to: NK Lee <nlee@cshl.edu> and PK Koo <koo@cshl.edu>.

The 2021 ICML Workshop on Computational Biology.

Deep InfoMax (Hjelm et al., 2018) is a method that learns representations of image data by maximizing a lower bound to the MI; the lower bound used is of the form:

$$\widehat{\mathcal{I}}_\psi\left(X^{(i)}; E_\psi(X)\right) = \mathbb{E}_{\mathbb{P}}\left[-\mathrm{sp}\left(-D\left(C_\psi\left(x^{(i)}\right), E_\psi\left(x\right)\right)\right)\right] - \mathbb{E}_{\mathbb{P}\times\tilde{\mathbb{P}}}\left[\mathrm{sp}\left(D\left(C_\psi\left(x'^{(i)}\right), E_\psi\left(x\right)\right)\right)\right] \quad (1)$$

where i indexes different views of the data, C_ψ and E_ψ are encoders, D is a discriminator, $\mathbb{P}$ and $\tilde{\mathbb{P}}$ are distributions of the data $x \in X$ and negative examples $x' \in X'$ respectively, and $\mathrm{sp}(z) = \log(1 + e^z)$ is the softplus function. The encoder $C_\psi(x)$ gives a "local" encoding of the data X, while the encoder $E_\psi(x)$ gives a "global" encoding of the data; the discriminator D is a function that takes as input pairs of local and global encodings $(C_\psi(x), E_\psi(x))$ of the data and outputs a score representing a level of concordance between the two types of encodings. The Deep InfoMax discriminator projects each encoding $C_\psi(x)$ and $E_\psi(x)$ to a higher (and common) dimensionality space and then calculates an inner product; the intent of this procedure is to measure similarity. Deep InfoMax then uses the discriminator scores are then used to calculate a lower bound to the MI according to equation (1).

3. Genomic Representations with Information Maximization (GRIM)

GRIM adapts the Deep InfoMax method, which was designed for representation learning of images, for genomic sequences by using a redesigned encoder and discriminator architectures as well as a modified negative example selection process. Since we are often interested in learning motifs in regulatory genomics, we use each positional vector of the first convolutional layer feature map as the "local" encodings; further processing the local encodings with a deep CNN yields the "global" encodings. The quantity maximized by GRIM represents MI between these local views and the global view of the entire sequence. Intuitively, this should provide a way to learn informative local patterns, such as motifs, that inform properties—such as the regulatory activity—of the whole sequence overall.

Encoder. The encoder $E_\psi(x) = (F_\psi \circ C_\psi)(x)$ is a composite function, parameterized[1] by ψ, that takes as input a one-hot encoded genomic sequence x of length L and outputs a local encoding $c = C_\psi(x)$ and global encoding of the whole sequence $h = E_\psi(x)$. The local encoding for each position is given by different convolutional scans, and thus the i-th position is a vector of size d, where d is the number of first layer filters. The global encoding takes the local encoding as input and processes it with additional convolutional and fully-connected layers to generate $h = E_\psi(x) = F_\psi(C_\psi(x))$ ($h \in \mathbb{R}^p$, where p is the global encoding dimension). The intuition behind these encodings C_ψ and E_ψ is as follows: the local encoding c looks at smaller, overlapping windows (i) of the sequence x and matches the convolutional filters to each of these small windows, and the global encoding h summarizes the whole sequence x into a single vector.

[1]For convenience, we here notate the parameters of both the local encoding C and the further encoding F with ψ, as is done in the Deep InfoMax method; however, encoders C and F do not actually share any parameters or even the same form or architecture.

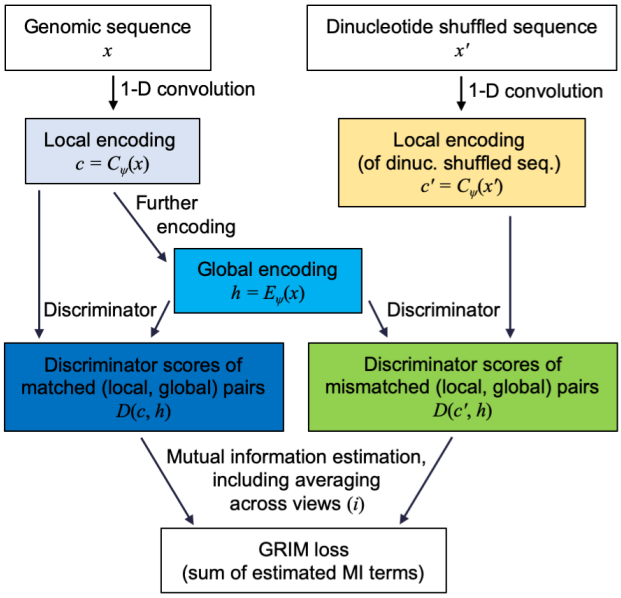

Figure 1. GRIM workflow. Note that box sizes do not necessarily correspond to dimensions of the quantities represented.

Discriminator. In contrast to Deep InfoMax, our GRIM discriminator is designed using prior knowledge of biological sequence motifs and motif dependencies in genomic sequences. Ideally we want the encoder to learn patterns of motifs in the first convolutional layer (C_ψ). Since the deeper layers of F_ψ largely maintain spatial information of motif scans but across a much wider receptive field, for a given sequence x we can calculate a direct outer tensor product (i.e., Kronecker product) of the global encoding with the local encodings. This tensor product calculates a measure of similarity between the local features at each position i of the feature maps of c with the global features h. We then perform a global max pooling over the entire feature map (i.e., position-wise and depth-wise), according to:

$$D\left(c^{(i)}, h\right) = \max_{j,k}\left(c^{(i)} \otimes h\right)_{j,k} \quad (2)$$

where $(j.k)$ indexes the outer product tensor between local and global encodings.

The intuition behind the form of the GRIM discriminator is that the score of a correctly matched pair of encodings (c, h) from the same input datum x should be higher than the score of a mismatched pair of encodings (c', h) where the local representation c' comes from a negative example x' while the global representation h comes from the actual datum $x \in X$ if the local encoder C_ψ—which generates the local features c—learns robust motif representation in its convolutional filters. If the negative example sequences X' contained the same motifs as the actual sequences X, then the global representations would not be sufficient to discriminate matched and mismatched (local, global) pairs.

MI estimation. The Deep InfoMax MI estimator relies on discriminator scores $D\left(C_\psi\left(x^{(i)}\right), E_\psi\left(x\right)\right)$ and $D\left(C_\psi\left(x'^{(i)}\right), E_\psi\left(x\right)\right)$; the former is the discriminator score of a matched set of global representation $E_\psi\left(x\right)$ and local representation $C_\psi\left(x^{(i)}\right)$ from the same sequence x, while the latter is the discriminator score of a mismatched set of global representation $E_\psi\left(x\right)$ from the sequence x and local representation $C_\psi\left(x'^{(i)}\right)$ from a negative example sequence x'. For GRIM, we choose the negative example x' to be a dinucleotide shuffled version of the sequence x.

MI between the genomic sequences X and a local view $X^{(i)}$ of the same sequences is estimated using the form of the MI estimator from Deep InfoMax (Eq. (1)), where $x^{(i)}$ is a convolutional patch i of sequence x ($x \in X$), x' is a negative example, which we choose to be a dinucleotide shuffled version of x. Note also that by the Data Processing Inequality (Cover & Thomas, 2012), the MI $I\left(X^{(i)}; E_\psi(X)\right)$ is itself a lower bound on the MI $I\left(X^{(i)}; X\right)$.

To take into account the MI between the whole genomic sequences X and all local subsequence patches $X^{(i)}$, GRIM calculates the MI estimate $\widehat{\mathcal{I}}_\psi\left(X^{(i)}; E_\psi(X)\right)$ for each local subsequence location i and sums them together to form its loss function $\mathcal{L}_{\text{GRIM}}$. Formally, this is

$$\mathcal{L}_{\text{GRIM}} = \sum_{i=1}^{L} \widehat{\mathcal{I}}_\psi\left(X^{(i)}; E_\psi(X)\right). \tag{3}$$

This corresponds to the "local-only" objective used in the Deep InfoMax method. Maximizing $\mathcal{L}_{\text{GRIM}}$ leads to the optimal parametrization of the encoders, which can be examined to understand the representations learned by GRIM.

4. Experimental Overview

Task 1. This task represents a simplified version of a multi-task classification of TF binding sites. Briefly, 25,000 sequences ($L = 200$) were embedded with 3-5 known motifs (Mathelier et al., 2016), sampled randomly from a pool of 11 motifs—CEBPB, FOSL1, Gabpa, MAFK, MAX, MEF2A, NFYB, SP1, SRF, STAT1, and YY1. Positions were chosen such that each motif has a buffer of at least one nucleotide from other motifs and the ends of the sequence. A corresponding label vector of length 11, one for each unique TF, was generated for each sequence (with 1 for embedded and 0 otherwise). The sequences were then randomly split into a train, validation, and test set according to the fractions 0.7, 0.1, and 0.2, respectively.

Task 2. This task represents a single-task classification with a plausible biological scenario where a single, dominant motif is sufficient to discriminate a class label but other sequence motifs are also weakly correlated with the class label. Briefly, positive class sequences were embedded with the SRF motif and 1-3 other known motifs from a pool of 5 known motifs—namely CEBPB, FOSL1, MAFK, MAX, and SP1—sampled with replacement. The negative class sequences are dinucleotide shuffled versions of the positive class sequences. A total of $N = 25{,}000$ such synthetic sequences and their associated labels were generated; the sequences were then randomly split into a train (0.7), validation (0.1), and test set (0.2).

Models. The models used in this study have similar architectures as (Koo & Eddy, 2019), namely CNN-S, where S is the first max pooling size and d first layer filters:

1. input (one-hot encoded DNA sequence x, length L)
2. convolution (d filters, size 19)
3. max pool (size S)
4. convolution (128 filters, size 5)
5. max pool (size $\lfloor L/(2S) \rfloor$)
6. fully-connected layer (512 units)
7. fully-connected output layer (number of labels)

Batch normalization (Ioffe & Szegedy, 2015) was applied to each hidden layer prior to ReLU activations, with the exception of GRIM models. Dropout (Srivastava et al., 2014) was applied after each max pooling layer with rate 0.1 and after the fully-connected hidden layer with 0.5. In the context of the GRIM encoder (see Sec. 3), layers 2-3 constitute the first local encoding (C_ψ) and layers 4-7 constitute the further encoding (F_ψ). All models were trained to minimize their respective loss functions (binary cross-entropy for supervised models; loss function in Eq. (3) for GRIM models) for 250 epochs using Adam with default settings (Kingma & Ba, 2014).

Filter evaluation. We employed activation-based alignments to visualize first-layer convolutional filters (Koo & Eddy, 2019). Tomtom, a motif comparison search tool (Gupta et al., 2007), was used to determine statistically significant matches between the filters and the JASPAR database (Mathelier et al., 2016).

5. Results

GRIM captures known motif representations. To test the representation learning capabilities of GRIM, we compared both GRIM using CNN-4 as an encoder and a supervised CNN-4 on the Task 1 dataset (see Section 4). Both models were trained with $d = 128$ first-layer filters to provide the models with ample opportunity to capture relevant motifs. We then compared the first-layer filter representations for each model. Since the ground truth motifs are available for our synthetic datasets, we can then test whether the respective models have indeed captured relevant motifs.

Evidently, GRIM outperforms the supervised CNN-4 model both in terms of the number of ground truth motifs it is able to learn as well as the number of first-layer convolutional filters that actually represent—i.e., statistically significant match with—a relevant known motif (Table 1). Despite using similar architectures, the overall representations learned by GRIM are more robust and identifiable compared to the same CNN-4 model trained with standard supervised learning, even without access to the class labels.

Table 1. Motif analysis on Task 1. True positive rate of filter matches to the 11 ground truth motifs (TPR) and false positive rate of filter matches to other motifs in JASPAR (FPR). Values shown are mean $\pm$ standard deviation for 10 independent trials.

MODEL	TPR	FPR
GRIM	0.929 ± 0.028	0.021 ± 0.016
CNN-4	0.263 ± 0.029	0.274 ± 0.036

GRIM learns sequence representations not identifiable using supervised deep learning. To illustrate another issue with supervised learning—that supervised models can learn only a subset of patterns when the task involves strong discriminative features coupled with weakly correlated features—we trained both GRIM using a CNN-25 architecture as an encoder and a supervised CNN-25 model on Task 2 data (see Section 4), which consists of one discriminative motif (SRF) and five other motifs that have a weaker connection to positive labels. Unlike CNN-4, CNN-25 is specifically designed to learn full motif representations (Koo & Eddy, 2019): when a supervised CNN-25 is trained on the Task 1 dataset, the model learns all relevant motifs and nearly all of the first-layer convolutional filters significantly match relevant motif representations.

By comparing the representations learned in first-layer filters of GRIM and a supervised CNN-25 both trained on Task 2 data (Table 2), each with $d = 64$ first-layer convolutional filters, we find supervised CNN-25 is unable to learn all of the relevant known motifs. Despite CNN-25's strong inductive bias to learn whole motif representations, it fails to capture all of the weakly correlated motifs, focusing solely on the stronger, non-variable SRF pattern. This phenomenon is inherent to the paradigm of supervised learning: once the supervised model learns to discriminate sequences (positive class vs. negative class) by learning the discriminative SRF motif, then it has achieved its objective—that is, minimized its loss. Applying such a model to real biological data, such as the prediction of chromatin accessibility sites, where many TF binding sites reside, this shortcut could then lead to an incomplete picture of the underlying mechanisms of the biology. Strikingly, GRIM is able to reliably detect all of the embedded motifs, including the discriminative SRF motif as well as the five other weakly correlated motifs.

Table 2. Motif analysis on Task 2. True positive rate of filter matches to the 6 ground truth motifs (TPR) and false positive rate of filter matches to other motifs in JASPAR (FPR). Values shown are mean $\pm$ standard deviation for 10 independent trials.

MODEL	TPR	FPR
GRIM	0.913 ± 0.038	0.016 ± 0.013
CNN-25	0.383 ± 0.041	0.173 ± 0.034

6. Discussion

Deep learning models tend to learn distributed representations of sequence motifs that are not necessarily human interpretable. Moreover, training deep models with supervised learning may not be amenable to detecting the full range of biologically important motifs. To resolve these issues, we have introduced GRIM, an unsupervised learning method based on the InfoMax principle. We showed that GRIM is a powerful approach to learn interpretable representations of sequence motifs in easy to access first-layer filters. We have also shown that GRIM is able to learn representations that are difficult to capture within a supervised learning paradigm.

Although we limit this study to the quality of the local encodings c of GRIM, our preliminary work, in addition to others in unsupervised representation learning, have shown that downstream prediction tasks trained on the global encodings generally fall short of the performance of gold-standard supervised models. Thus, the local representations learned by GRIM may be of greater use to enhance the performance of supervised learning models via in transfer learning.

Another challenge that arises in learning genomic sequence features with GRIM—and, indeed, in representation learning of genomic features in general—is that first-layer convolutional filters often learn motifs well, but a large degree of redundancy in representations is often found. In practice, a large number of filters is required for good performance and is thus unavoidable with our current limited strategies for initialization. Thus, reducing the effective dimensionality of the learned first-layer representations could make GRIM more human interpretable.

References

Alipanahi, B., Delong, A., Weirauch, M. T., and Frey, B. J. Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nature Biotechnology*, 33(8):831–838, 2015.

Anand, N. and Huang, P.-S. Generative modeling for protein structures. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 7505–7516, 2018.

Avsec, Ž., Weilert, M., Shrikumar, A., Krueger, S., Alexandari, A., Dalal, K., Fropf, R., McAnany, C., Gagneur, J., Kundaje, A., et al. Base-resolution models of transcription-factor binding reveal soft motif syntax. *Nature Genetics*, 53(3):354–366, 2021.

Becker, S. and Hinton, G. E. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 355(6356):161–163, 1992.

Cover, T. M. and Thomas, J. A. *Elements of Information Theory*. John Wiley & Sons, 2012.

Gupta, S., Stamatoyannopoulos, J. A., Bailey, T. L., and Noble, W. S. Quantifying similarity between motifs. *Genome Biology*, 8(2):1–9, 2007.

Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., and Bengio, Y. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., and Madry, A. Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175*, 2019.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pp. 448–456. PMLR, 2015.

Kelley, D. R., Snoek, J., and Rinn, J. L. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Research*, 26(7): 990–999, 2016.

Kelley, D. R., Reshef, Y. A., Bileschi, M., Belanger, D., McLean, C. Y., and Snoek, J. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Research*, 28(5):739–750, 2018.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Koo, P. K. and Eddy, S. R. Representation learning of genomic sequence motifs with convolutional neural networks. *PLoS Computational Biology*, 15(12):e1007560, 2019.

Koo, P. K. and Ploenzke, M. Improving representations of genomic sequence motifs in convolutional networks with exponential activations. *Nature Machine Intelligence*, 3 (3):258–266, 2021.

Linsker, R. Self-organization in a perceptual network. *Computer*, 21(3):105–117, 1988.

Lu, A. X., Zhang, H., Ghassemi, M., and Moses, A. M. Self-supervised contrastive learning of protein representations by mutual information maximization. *bioRxiv*, 2020.

Mathelier, A., Fornes, O., Arenillas, D. J., Chen, C.-y., Denay, G., Lee, J., Shi, W., Shyr, C., Tan, G., Worsley-Hunt, R., et al. JASPAR 2016: a major expansion and update of the open-access database of transcription factor binding profiles. *Nucleic Acids Research*, 44(D1):D110–D115, 2016.

Shrikumar, A., Tian, K., Shcherbina, A., Avsec, Ž., Banerjee, A., Sharmin, M., Nair, S., and Kundaje, A. TF-MoDISco v0.4.4.2-alpha. *arXiv preprint arXiv:1811.00416*, 2018.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Sinai, S., Kelsic, E., Church, G. M., and Nowak, M. A. Variational auto-encoding of protein sequences. *arXiv preprint arXiv:1712.03346*, 2017.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Tasaki, S., Gaiteri, C., Mostafavi, S., and Wang, Y. Deep learning decodes the principles of differential gene expression. *Nature Machine Intelligence*, 2(7):376–386, 2020.

Tschannen, M., Djolonga, J., Rubenstein, P. K., Gelly, S., and Lucic, M. On mutual information maximization for representation learning. *arXiv preprint arXiv:1907.13625*, 2019.