Bioinformatics, 35(13), 2019, 2177–2184
doi: 10.1093/bioinformatics/bty964
Advance Access Publication Date: 27 November 2018
Original Paper

Genome analysis

An equivariant Bayesian convolutional network predicts recombination hotspots and accurately resolves binding motifs

Richard C. Brown* and Gerton Lunter *

Wellcome Trust Centre for Human Genetics, University of Oxford, Roosevelt Dr, Oxford OX3 7BN, UK

*To whom correspondence should be addressed.

Associate Editor: John Hancock

Received on June 25, 2018; revised on November 13, 2018; editorial decision on November 19, 2018; accepted on November 26, 2018

Abstract

Motivation: Convolutional neural networks (CNNs) have been tremendously successful in many contexts, particularly where training data are abundant and signal-to-noise ratios are large. However, when predicting noisily observed phenotypes from DNA sequence, each training instance is only weakly informative, and the amount of training data is often fundamentally limited, emphasizing the need for methods that make optimal use of training data and any structure inherent in the process.

Results: Here we show how to combine equivariant networks, a general mathematical framework for handling exact symmetries in CNNs, with Bayesian dropout, a version of Monte Carlo dropout suggested by a reinterpretation of dropout as a variational Bayesian approximation, to develop a model that exhibits exact reverse-complement symmetry and is more resistant to overtraining. We find that this model combines improved prediction consistency with better predictive accuracy compared to standard CNN implementations and state-of-art motif finders. We use our network to predict recombination hotspots from sequence, and identify binding motifs for the recombination–initiation protein PRDM9 previously unobserved in this data, which were recently validated by high-resolution assays. The network achieves a predictive accuracy comparable to that attainable by a direct assay of the H3K4me3 histone mark, a proxy for PRDM9 binding.

Availability and implementation: https://github.com/luntergroup/EquivariantNetworks

Contact: richard.brown@well.ox.ac.uk or gerton.lunter@well.ox.ac.uk

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Deep Learning based models have been highly successful in many areas where traditional modelling approaches appeared to have reached their limits. This is true also for modelling biology from sequence, where Deep Learning sequence models have been shown to outperform previous state of the art techniques (Alipanahi *et al.*, 2015; Kelley *et al.*, 2016; Zhou and Troyanskaya, 2015). These models have several attractive characteristics, including their ability to learn without the need for manual feature curation or model seeding, and the ability to learn complex non-linear interactions. This is balanced by the need for large amounts of training data, and the tendency of these models to overtrain. In many situations this is not problematic, but for applications in biology training data are often fundamentally limited, either by the size of the genome, the limited genetic diversity of a population or the cost of assaying individuals or samples. In this context it is particularly important to exploit the known structure of the model as much as possible, and to try and avoid overtraining and improve generalizability.

As an example of a biologically motivated problem with limited training data, we consider the problem of predicting recombination hotspots from sequence. In humans, the rate of meiotic recombination varies greatly along the genome, with recombinations occurring primarily in short regions colloquially known as recombination hotspots. The mechanism for this localization has been shown to be the action of the zinc finger protein PRDM9 (Baudat *et al.*, 2013). After being expressed in meiotic prophase, PRDM9 binds DNA in a sequence-specific manner, and catalyzes H3K4 and H3K36 trimethylation and double-stranded breaks, some of which are resolved as recombinations.

The canonical PRDM9 binding motif, `CCTCCCTNNCCAC`, was identified by an enrichment analysis of sequences underlying hotspot versus those in regions not involved in recombination (Myers *et al.*, 2008). However, while significantly enriched, this motif is only weakly predictive of recombination. For example, in our data it appears in around 2% of hotspots and 0.3% of coldspots. This, coupled with the fact that there are only around 20 000 hotspots, many of which are ill resolved [median length ~2000 base pairs (bp)], makes prediction of recombination hotspots challenging.

To build a model that optimizes predictive power given these constraints, we combine two recently introduced ideas. One is that of equivariant convolutional networks (Cohen and Welling, 2016) which we use to build a network exhibiting the reverse-complement (RC) symmetry of double-stranded DNA. Equivariance is a richer concept than invariance; while a sequence and its RC are expected to exhibit the same pre-disposition for recombination, the binding of proteins to DNA is usually not RC-symmetric, and protein–protein interactions can be similarly directional. This is reflected by symmetries on higher levels in the convolutional neural network (CNN) that mirror the RC symmetry on the sequence level. A similar approach specific to RC symmetry was found independently by Shrikumar *et al.* (2017), who reported that enforcing RC symmetry by weight tying increased predictive accuracy and stability of the learned motifs.

The second insight is that dropout, a commonly used regularization technique for CNNs, can be interpreted as an approximation of a variational Bayesian inference (Gal and Ghahramani, 2016), here referred to as Bayesian dropout. This interpretation suggests particular modifications of standard applications of dropout. In particular, this interpretation suggests the use of Monte Carlo (MC) averaging of activations (MC dropout) rather than weight averaging at the prediction stage. The implementation suggested in the literature would break equivariance (Gal and Ghahramani, 2016). Here we show how to obtain an equivariant version of Bayesian dropout, and how to use this to obtain a model exhibiting exact RC symmetry while retaining the advantages of Bayesian dropout.

The remainder of the paper is organized as follows. In Section 2, we introduce equivariant networks, and show how to build equivariance into standard CNN layers including convolutional layers, max-pooling and dropout. We show how to make the Bayesian dropout scheme equivariant, and we introduce a new max-pooling layer that acts over the action of the RC symmetry group. In Section 3, we show that RC-equivariant networks and Bayesian dropout each and in combination significantly increase predictive accuracy, both on simulated and real data, whereas classical dropout hurts performance. We further show that our network outperforms state-of-art motif finders, and is able to identify high-resolution binding motifs. We finish with discussion and conclusions in Section 4.

2 Materials and methods

2.1 Equivariant networks

We use feedforward neural networks to generate a learnable mapping directly from the input sequence to an output response variable, which in this case is a class assignment. The network is composed of a sequence of layers that define a directed acyclic computation graph, with each layer acting in turn on the output of its ancestor. Explicitly, for a two dimensional input tensor X_{ij}, the network can be viewed as a function composition

$$F(X) = (F_n \circ \cdots \circ F_1)(X) = F_n(F_{n-1}(\ldots F_1(X)\ldots)) \tag{1}$$

with component functions F_i representing the actions of layer i; here $\circ$ denotes function composition. Note that component functions are defined on their own tensor spaces, $F_i : T^{(i-1)} \to T^{(i)}$.

In our case, X_{ij} represents a length-N DNA sequence from an alphabet $\{A, C, G, T\}$ which is usually one-hot encoded (Alipanahi *et al.*, 2015; Lanchantin *et al.*, 2016; Zhou and Troyanskaya, 2015) so that $T^{(0)} = \mathbb{R}^{4\times N}$. The DNA represented by the input sequences X exists physically mostly in a double-stranded form, with one strand hydrogen bonded to its RC. This means that the sequence seen by the model could just as naturally be represented by its RC, and the network should arrive at identical outputs for these two sequences:

$$F(X) = F(RC(X)) \tag{2}$$

where $RC : T^{(0)} \to T^{(0)}$ maps the encoding of a sequence to the encoding of its RC. One way to achieve this symmetry is to require that $F_1(RC(X)) = F_1(X)$. However, this is highly restrictive; the first layer often represents protein binding motifs, which are often not RC-symmetric. Intuitively, one wants the output of a layer to exhibit the 'equivalent' symmetry appropriate for the encoding of the next layer. The mathematical translation of this is to require equivariance:

$$(F_i \circ RC_{i-1})(X) = (RC_i \circ F_i)(X) \tag{3}$$

for all i and all $X \in T^{(i-1)}$. A graphical representation of this relation is given in Figure 1. Note that the two operators RC in Equation (3) are different, as indicated by their index, as they act on different tensor spaces. In particular, in our case RC_n acts as the identity on $T^{(n)}$ to ensure that the full model is invariant under reverse-complementing, and on $T^{(0)}$ the operator RC_0 is determined by the chosen encoding. The modeller has freedom in choosing RC_i on intermediate layers, subject to constraint (3), which also imposes constraints on the F_i, the initialization, training procedure and any parameter ties used during training. Note that this setup is not restricted to RC equivariance, and is valid for any group with actions on tensor spaces, including mirror symmetries, rotations and translations. In this general case the operators RC_i in Figure 1 are replaced by group actions $J_g^{(i)}$ operating on $T^{(i)}$, where g is a group element. We will not pursue that direction here, but see Cohen and Welling (2016) for an exposition.

2.2 Choice of one-hot basis

We define the vectors

$$A = [1, 0, 0, 0]', C = [0, 1, 0, 0]', G = [0, 0, 1, 0]', T = [0, 0, 0, 1]'$$

where $'$ denotes transposition, and encode a genomic sequence by

$$\begin{tikzcd}
T^{(0)} \arrow[r, "F_1"] \arrow[d, "RC_0"'] & T^{(1)} \arrow[r, "F_2"] \arrow[d, "RC_1"] & \cdots \arrow[r, "F_n"] & T^{(n)} \arrow[d, "RC_n"] \\
T^{(0)} \arrow[r, "F_1"] & T^{(1)} \arrow[r, "F_2"] & \cdots \arrow[r, "F_n"] & T^{(n)}
\end{tikzcd}$$

Fig. 1. Commutative diagram for a RC equivariant network. Compositions of functions along any path in the network that respects the arrow directions depend only on the start and end point, and not on the path taken

the concatenation of corresponding column vectors. For a sequence encoded in this way,

$$RC(X)_{ij} = X_{-i,-j} \tag{4}$$

with negative indices denoting offsetting from the opposite end of each dimension by that amount. Note that we use 1-based indexing.

2.3 Convolutional layer

For sequence classification we use a 1D convolution layer, with n_f filters of length f_l stored in a weight matrix W_{ijk}. The output of such an operation on tensor X_{ij} (ignoring bias terms for simplicity) is

$$C(X)_{ij} = f\left[\sum_{m=1}^{4}\sum_{n=1}^{f_l} X_{m,j+n-1}\, W_{mni}\right] \tag{5}$$

where f is the activation function. Note that we used 'valid padding' so the sequence dimension is reduced by $f_l - 1$. Applying the RC operation to the input tensor of shape $[4, N]$ yields

$$C(RC_{l-1}(X))_{ij} = f\left[\sum_{m=1}^{4}\sum_{n=1}^{f_l} X_{-m,-(j+n-1)}\, W_{mni}\right] \tag{6}$$

$$= f\left[\sum_{m'=1}^{4}\sum_{n'=1}^{f_l} X_{m',[-j]+n'-1}\, W_{-m',-n',i}\right] \tag{7}$$

$$= f\left[\sum_{m'=1}^{4}\sum_{n'=1}^{f_l} X_{m',[-j]+n'-1}\, W_{m',n',-i}\right] \tag{8}$$

$$= C(X)_{-i,-j} := RC_l(C(X))_{ij} \tag{9}$$

where we used the substitutions $m = -m'$, $n = f_l + 1 - n'$, and $[-j]$ denotes the positive index $(N+1) - (f_l - 1) - j$. At (8) we assumed that W obeys the symmetry $W_{m,n,i} = W_{-m,-n,-i}$. Therefore, the convolutional layer satisfies (3) if this weight symmetry holds, and if we define RC on the output layer as in (9). A schematic of this is shown in Figure 2A. We note that this specific symmetry was used in a convolutional layer in Shrikumar *et al.* (2017) and was shown to improve inference.

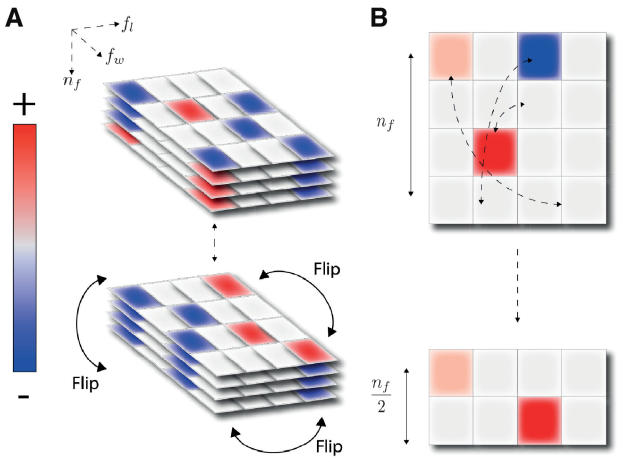

Fig. 2. (**A**) The three dimensional filter tensor has enforced symmetry by weight tying, flipping the second half of the filter axis as shown. (**B**) In contrast to a conventional max-pooling along the spatial dimension, this approach permits pooling along filters

2.4 Max-pooling

Spatial max-pooling along the position dimension of a tensor is used in order to increase the receptive field at the expense of resolution. We consider the special, commonly used, case where the stride length is the same as pool width. For this case we have, for a pool length p_l,

$$MP(X)_{ij} = \max_{k\in[1+(j-1)\frac{N}{p_l}, j\frac{N}{p_l}]}(X_{ik}). \tag{10}$$

As long as p_l divides N this defines an equivariant mapping, with RC defined on the next layer in the obvious way.

2.5 RC max-pooling

We frequently found it useful to pool along the 'filter axis' rather than along the spatial direction. More precisely, we take maxima along orbits under the group action, which in our case consist of two elements. After RC max-pooling we therefore have a new output

$$M(x)_{ij} = \max(X_{i,j}, X_{-i,-j}). \tag{11}$$

This process halves the size of the output tensor compared to the input. The resulting compression is depicted in Figure 2B. Equation (3) is again satisfied, now with RC acting as the identity on the new layer. A network that contains RC max-pooling is therefore automatically symmetric under reverse-complementing. In general, more complex symmetry groups may contain non-trivial subgroups, and any of those may be used to do partial symmetric max-pooling; see Cohen and Welling (2016).

2.6 Dropout

Dropout is a form of stochastic regularization for neural networks designed to prevent overfitting by adding noise to the output of a layer during network training (Srivastava *et al.*, 2014). This is commonly implemented by dropping out nodes according to a Bernoulli-distributed random variable, and can be implemented by taking the Hadamard product between two identically shaped tensors. Define

$$\epsilon_{ij} \overset{\text{i.i.d.}}{\sim} \text{Bern}(p), \tag{12}$$

then dropout applied to a tensor X with dropout rate p is

$$D^{\epsilon}(X)_{ij} = (\epsilon * X)_{ij} = \epsilon_{ij} X_{ij} \tag{13}$$

with the tensor ϵ sampled on a batch-wise basis. We denote the Hadamard product by $*$ rather than the more usual $\circ$ to avoid confusion with function composition. Applying RC gives the condition

$$D^{\epsilon}(RC_{l-1}(X))_{ij} = X_{-i,-j}\epsilon_{ij} = RC_l(D^{\epsilon}(X)) \tag{14}$$

which holds if

$$\epsilon_{ij} = \epsilon_{-i,-j} \tag{15}$$

and if $RC_l = RC_{l-1}$.

2.7 Bayesian equivariant networks

Convolutional networks have been shown to work well on large datasets, but it is known that they overfit quickly when relatively little training data are available (Gal and Ghahramani, 2016). This is often the case in the biological domain, calling for a principled approach to deal with limited training datasets. Networks with dropout after every convolutional layer and trained via backpropagation can be seen as approximating Bayesian variational inference

(Gal and Ghahramani, 2016), promising good behaviour in data-poor settings. The interpretation also suggested to use Bayesian dropout rather than traditional weight averaging for making predictions, using dropout to approximate sampling from a posterior weight distribution. In this interpretation, the corresponding prediction is obtained by the average over a sample of instantiations of the network:

$$p(Y^*|X^*, \mathbf{X}, \mathbf{Y}) \approx \frac{1}{K}\sum_{k=1}^{K} F(Y^*|X^*, \epsilon_k) \tag{16}$$

for unseen X^* given previous training examples $\mathbf{X} = (X_1, \ldots, X_n)$ and $\mathbf{Y} = (Y_1, \ldots, Y_n)$ used to train F. We refer to the process of using dropout during both training and test time as *Bayesian dropout*, while we use *dropout* to refer to networks that use weight averaging at test time.

The set of random variables $\epsilon = (\epsilon_1, \ldots, \epsilon_K)$ in (16) implement a random sample of the weights defining the network F. To implement equivariance, it is sufficient that these weights obey the correct symmetry, e.g. (15), as then each term will be RC-symmetric and so will the sum. The resulting function will be stochastic, but as long as ϵ is sampled and fixed beforehand, it will nevertheless exhibit exact RC symmetry.

2.8 Choice of activation function

Rectified linear units (ReLUs), defined as $\mathrm{ReLU}(z) = \max(0, z)$, are activation functions applied at the terminus of every layer, providing the requisite non-linearity for stacked layers to have richer representational power than a simple linear model. These activation functions have become ubiquitous in deep learning, though usually for networks which are composed of more layers than is typical in genomic problems, as they reduce or resolve the vanishing gradient problem. ReLU elements have the property that for a large number of units, the output will be identically zero. Although sparsity can be advantageous, e.g. by making interpretation easier, we found that it hampers convergence in all problems we have tried. We found that using exponential linear units (ELUs) (Clevert *et al.*, 2015) result in substantially better convergence behaviour on our dataset (see Supplementary Fig. S1). We observed qualitatively similar results using shifted ReLUs [$\mathrm{SReLU}(z) = \max(z, -1)$] (data not shown).

2.9 Initialization of the output layer

We found that using a custom initialization of the output layer, providing the classification scores before a final softmax transformation, substantially improved convergence for the networks we considered (Supplementary Fig. S2). We initialized the weight matrix of the final layer with 1, and the two bias parameters (corresponding to the two nodes representing class probability) with $\{1, -1\}$. A motivation for this choice was the observation that on a number of problems the output layers weights were always close to these values, so it appears that this initialization is closer to the global optimum than traditional approaches such as those proposed by Glorot and Bengio (2010). We also tried Batch normalization (Ioffe and Szegedy, 2015) to address this problem, but found that it did not improve convergence by as much as custom initialization did (see Supplementary Fig. S5).

2.10 Network architecture

To find optimal non-Bayesian non-equivariant networks we performed hyperparameter searches for both datasets (see Section 3.1) independently (see Supplementary Sections A.1 and I for the networks and search space, respectively). Next, for each dataset we built three extensions as follows:

1. An equivariant network, by adding an RC pooling layer and enforcing weight equivariance in the internal layers;
2. A Bayesian network, by adding one or two MC dropout layers;
3. A Bayesian equivariant network, by doing both, and changing MC dropout into equivariant MC dropout.

To build these networks we performed additional restricted hyperparameter searches on the position of the RC pooling and MC dropout layers, the type of RC pooling (max, sum or average pooling), and the L2 regularization parameter (because both equivariance and dropout are themselves inherently regularizing), while keeping all other network parameters fixed (see Supplementary Section A.2).

3 Results

3.1 Datasets

Simulated data were generated as follows. As a model for a regulatory network involving two binding proteins, we sampled two position weight matrices (PWMs) (ATAF4 and ERF1) from JASPAR (Sandelin *et al.*, 2004). We first randomly sampled 40 000 times a random {0, 1}-response with 50% probability for each, representing a measured phenotype of interest. For each response variable with value 1 we sampled a specific motif from each of the two PWMs (reverse-complementing them at random) and injected it into a random background sequence with 40% probability. Otherwise we injected the motifs with probability 20%. This procedure resulted in 40 000 sequences of length 1000, with 20 110 in category 1 and 19 890 in category 0. Note that there is a substantial amount of noise in this dataset, making for a challenging classification problem.

We also prepared a dataset of human recombination hot and coldspots, similar to Myers *et al.* (2008). We first applied a simple hidden Markov model to segment the genome into regions classified as 'hot' and 'not hot'. For emission probabilities we used two exponential distributions for $p(\text{observed rate}|\text{hot})$ and $p(\text{observed rate}|\text{not hot})$, and used Viterbi training to set the parameters. The median recombination rate in regions classified as 'hot' was 10.5 cM/Mb. The length of hotspots, once annotated, had a median value of 2448 bp, but with a heavy tail up to around 20 kbp. For our purposes, we wanted to study localized recombination events, so we discarded all hotspots longer than 4 kb. After discarding sequences with unspecified bases, we then sampled a sequence of length 1 kb from the centre of the remaining hotspots, yielding a total of 17 552 truncated hotspots for classification.

To define a matched set of coldspot regions once hotspot regions were identified, we applied a greedy search within 300 kb of each hotspot region to identify a sequence within 10% GC content, and with a recombination rate below 0.5 cM/Mb. GC matching ensures that GC content, which tends to be higher in recombination hotspots due to GC-biased gene conversion, cannot be used as a proxy for recombination strength, forcing the network to focus on causal signals of DNA binding motifs. This pipeline yielded a total of 17 547 truncated coldspots, also of length 1000 bp.

3.2 ELU and SReLU activations improve convergence

Across a wide range of learning and topology hyperparameters we observed consistent difficulty with the ReLU activation function, with networks often not converging to optimal accuracies, and sometimes not converging at all, leaving no better than random guesses. We observed, upon experimentation, that ELU and SReLU activations gave much improved results (see Supplementary Fig. S1 for an illustration of this effect with ELU activations).

3.3 Equivariant Bayesian networks improve classification accuracy

Comparing the best-performing networks resulting from the optimization procedure outlined in Section 2.10 we found that for both datasets the equivariant Bayesian networks outperformed the best-in-class non-equivariant networks, achieving significantly better test accuracy over a sample of 50 runs (Fig. 3). We found that both equivariant non-Bayesian and Bayesian non-equivariant networks were significantly more accurate than the best-in-class convolutional network, and that the combination of Bayesian dropout and equivariance was again significantly more accurate than either.

We also investigated how well the equivariant Bayesian network performed for these two problems in comparison to classical data augmentation, whereby the RC of every sequence is added to the dataset, at the cost of doubling the training time in comparison to the unaugmented non-equivariant networks. As expected, data augmentation improved the accuracy compared to standard networks trained without data augmentation. Equivariant Bayesian networks remained significantly better than the best discovered network trained with data augmentation for the simulated dataset, while results were comparable for the recombination dataset. Here, we still note an empirical advantage in the gradient update speed in the equivariant network of 32% in comparison to two updates in the augmented data case, and 8% in comparison to the augmented case where we double the batch size for improved computational efficiency. The reason that this is not double the speed of the data augmented case is due to the additional gradient updates required to maintain weight tying in these equivariant networks.

Finally, we performed an experiment to test the efficacy of equivariant networks in the low-data regime. Here, we used the baseline networks optimized for the full dataset, and trained them on half and a quarter of the original training data comparing the asymmetric networks to the equivariant and equivariant Bayesian networks. For the recombination dataset we observed an increasing performance gap between the equivariant and baseline networks as the data size was reduced. For the simulated data, we found that the distribution of accuracies at convergence became increasingly broad and overlapping, as the networks often struggled to converge in this regime. This made interpretation more difficult, and all of the networks ended with around 0.54 area under the ROC curve (AUROC) (Supplementary Fig. S6).

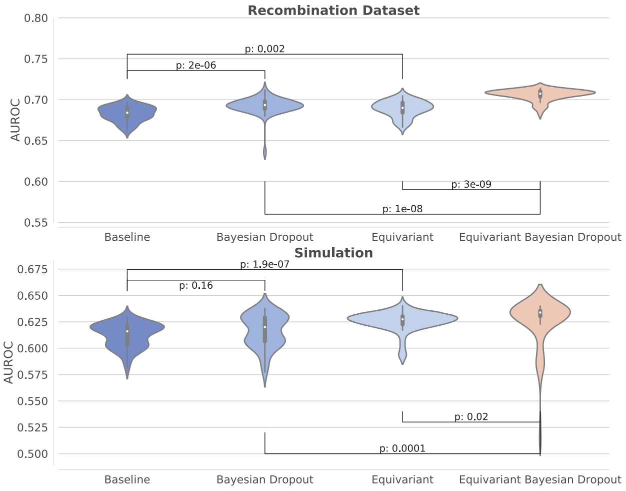

Fig. 3. For every network configuration, we trained to convergence a total of 50 times and plotted the distribution of the final AUROC statistic. For both the simulation (top) and the recombination data (bottom) we find a statistically significant improvement over the case where either equivariance or Bayesian dropout is not applied. *P*-values calculated using the Wilcoxon test

3.4 Conventional dropout considered harmful

To understand the contribution of Bayesian dropout on the performance of the network, we compared the final asymmetric topology for both datasets with no dropout, Bayesian dropout and conventional dropout; the latter uses dropout during training but weight scaling during test time. We recorded the mean accuracy over 50 trials. Compared to Bayesian dropout, an identical network that used the conventional dropout procedure yielded substantially inferior results (Table 1), and performs worse than the baseline without dropout. This behaviour was seen in both datasets. It appears that the approximation of weight averaging breaks down in this setting, apparently causing biases in the output leading to reduced accuracies.

3.5 Equivariant Bayesian networks yield more consistent predictions between runs

For the purposes of motif discovery and interpretation, it is advantageous for networks to result in consistent internal representations between training runs. It has been noted previously (Shrikumar *et al.*, 2017) that CNNs tend not to learn stable internal representations across different training runs, increasing the potential for different prediction results on the same test data. Enforcing equivariance reduces the potential for such representational instability, and we hypothesize that this would increase the consistency of predictions on hold-out data.

To test this hypothesis, we trained networks 50 times independently on identical training data, computed predictions on identical hold-out data, and computed 1225 pairwise correlations of these predictions. We performed this experiment for the previously determined optimal networks for the case of training with and without data augmentation, and for both cases we made the model equivariant without otherwise changing the topology. We found that in all four cases, the equivariant networks achieved significantly higher correlation between runs than their non-equivariant counterparts with the same number of filters (Fig. 4). We note that a hyperparameter search specifically for equivariant networks might further improve performance, but for this experiment we did not pursue this. The improved prediction consistency indicates that equivariance result in more training that converges more reliably to the same optima, which would be expected to improve the consistency of the harvested motifs.

3.6 Equivariant Bayesian networks improve upon existing motif finders

To assess our approach we compared our results with two state-of-art motif finders, DeepMotif (DeMo; Lanchantin *et al.*, 2016) and HOMER (Heinz *et al.*, 2010), using the recombination dataset.

Table 1. Mean test accuracies (± one standard error) for the symmetric networks with no dropout, Bayesian dropout and conventional dropout

Dataset	Baseline	Bayesian drop	Conventional drop
Simulation	58.2±0.3	59.2±0.2	56.1±0.2
Recombination	63.5±0.2	64.5±0.2	58.2±0.3

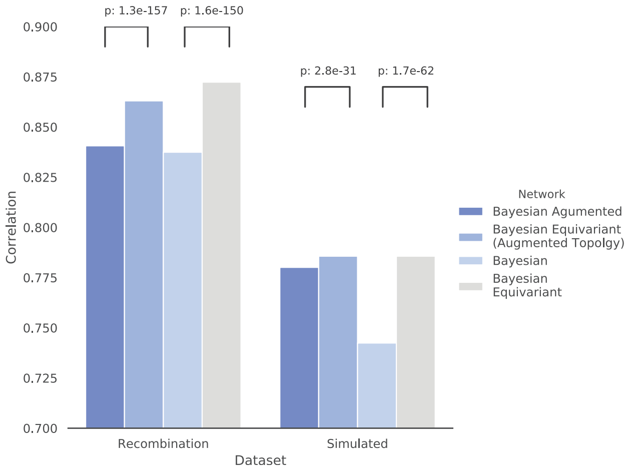

Fig. 4. Mean of correlation of predictions on unseen data for pairs of networks that were independently trained on identical data. Shown are results for Bayesian networks optimized on both augmented and unaugmented data, and for the same networks but constrained to be equivariant; the latter show significantly more consistent predictions than non-equivariant networks. Full architectural details given in Supplementary Material Section A

For the neural network-based algorithm DeMo we benchmarked our results against the three network configurations offered by the package, CNN, RNN and CNNRNN. All of these have substantially more complexity and weights than the model we used, but failed to perform well on this task. Indeed the CNN model (a CNN) often failed to converge at all, a problem that we also observed when training our vanilla convolutional networks on these datasets. The first training attempt that converged to better random predictions had AUROC 0.58. Both the RNN and CNNRNN topologies converged reliably, but achieved AUROCs of 0.64, substantially less than the median AUROC of 0.71 achieved by our equivariant network. These results are displayed in Figure 5.

We ran HOMER on the same dataset, using default parameters except for setting a target motif length of 22 in an attempt to find the sparse motif we discovered, but were unable to recover it. We did however identify the canonical PRDM9 motif, and a number of spuriously looking motifs (see Supplementary Fig. S7).

3.7 Discovery of novel PRDM9 binding motifs

Using the equivariant Bayesian network with the best classification accuracy, we recovered binding motifs by identifying a subset of input sequences responsible for the activation of particular nodes at the first layer, finding the subsequence driving the activation in each of those, and building a PWM from the aligned subsequences.

This process identified five motifs, three of which are versions of the classical 13mer that was identified via enrichment analysis and exhaustive search of motifs in Myers *et al.* (2008) (Fig. 5c–e). In addition we identified two substantially longer (22 nt) and sparse motifs that were not previously identified using this dataset (Fig. 6). Since our networks are only approximately Bayesian, to confirm that these motifs were not artefacts of overtraining, we confirmed their significance by frequentist tests of significance for enrichment on hold-out data (Table 2). The enrichment of the sparse motif is similar to that of the canonical 13mer originally associated with PRDM9 binding, but its complexity and under-constrained nature (only about 8 out of the 22 bases of the motif have substantial information content) mean that it was not originally discoverable using a traditional enrichment approach. We discuss this finding further in the next section.

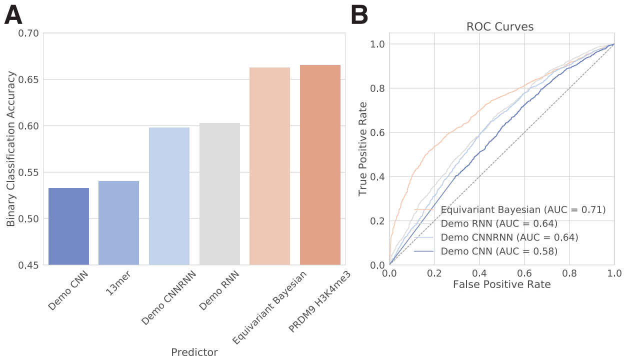

Fig. 5. Classification accuracies (**A**) and (area under) receiver operator curves (**B**) for the equivariant Bayesian network and the three models provided by the Deep Learning based motif finder DeMo (Lanchantin *et al.*, 2016). For comparison we also show the classification accuracy of the canonical PRDM9 13mer motif, as well as the classification accuracy obtainable by using the differential histone mark H3K4me3 under PRDM9 overexpression, a direct proxy for PRDM9 binding (Altemose *et al.*, 2017)

4 Discussion

Deep learning approaches have been shown to be very effective in building sequence models on large-scale data (Kelley *et al.*, 2016; Quang and Xie, 2016; Zhou and Troyanskaya, 2015). However, through simulated and biological data we show here that models designed using traditional building blocks for neural networks may struggle to converge consistently and produce reliable results in cases where the signal in the data is weak, and the amount of training data is limited. This problem was seen across a large number of network architectures, as well as in methods specifically designed to identify transcription factor binding sites from sequence data (Lanchantin *et al.*, 2016). To improve on this situation, we showed how to combine equivariant neural networks (here, neural networks that exhibit exact RC symmetry) with Bayesian dropout. While a naive combination of these ideas would result in networks that are only in expectation RC symmetric, we show that it is possible to achieve exact RC symmetry. In addition, we find that by modifying the activation functions, and the initialization of the output layer, we obtain a further significant improvement in accuracy.

Equivariant networks can be implemented in several ways. We chose to implement a standard (non-equivariant) network, enforcing equivariance by requiring certain identities on the parameters. Although this introduces some extra computation, this approach yields a number of advantages in comparison to data augmentation, where the training data are made symmetric and symmetry must be learned by the network. We find that in comparison to data augmentation, an equivariant network reduces the update time and results in equal or better test accuracies and improved consistency between training runs. Equivariant networks have the additional advantage of guaranteeing identical predictions on the forward and reverse strand, which may be desirable in applications.

We also found that Bayesian dropout resulted in a substantial performance improvement. This is striking, as dropout is normally thought of as a regularization technique that reduces the tendency of overtraining often found in large models. By contrast, in our regime the final models were small in comparison to the number of samples in the training set, and we did not see much evidence of overfitting

Table 2. Comparison of the novel motif with previously known PRDM9 binding site in both the training and test datasets

Motif	Total hot	Total cold	Test hot	Test cold	Odds ratio	Corrected *P*-value
CANNNNTNNTNNNNNNNNCCCC	2045	1186	83	38	2.16	2.03×10^{-8}
CANNNNTNNTNNNNNNNNCCCCC	652	251	29	8	3.63	1.4×10^{-5}
CANNNNTNNTNNNNNNNNCCNCC	3919	2933	158	109	1.45	4.7×10^{-6}
CCNCCNTNNCCNC	5695	3418	436	265	1.64	3.92×10^{-29}
CCTCCCTNNCCAC	400	77	57	14	4.07	5.34×10^{-12}
Sub motif CCNCC	22 370	22 273	1431	1496	0.96	0.31

Note: Total hot/cold- number of motifs in the full hotspot/coldspot dataset; test hot/cold- number of motifs among hold-out test data consisting of 1454 hotspots and 1530 coldspots; corrected *P*-value- Bonferroni-corrected *P*-values for association (Binomial test).

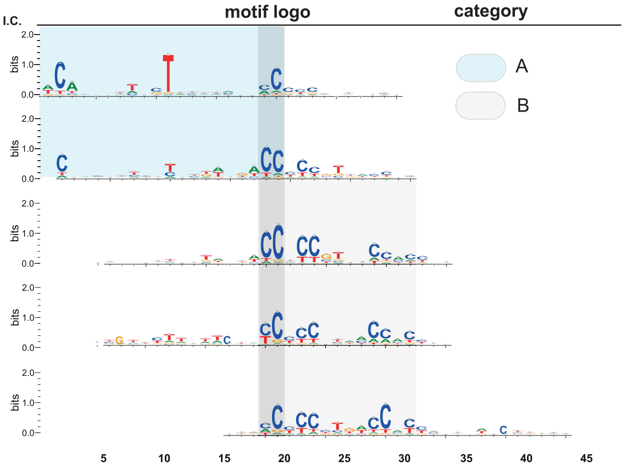

Fig. 6. We found the sequences corresponding to the maximum activation on each filter, and from these built sequence logos. We found a number of motifs (category A) corresponding to the classical PRDM9 binding motif that was discovered with an exhaustive search of an analogous dataset (Myers *et al.*, 2008). Additionally, we found two novel 22-base motifs (category B) that were previously undiscovered on this dataset. It is notable that they bear a striking resemblance to the recently discovered motif by Altemose *et al.* (2017)

either with or without using dropout, such as a substantial continued decrease of training loss after test loss stabilized, and we did not find that the model latched on to spurious motifs that were not statistically significant on test data. It appears that, in addition to addressing overtraining, Bayesian dropout leads to superior learning and feature extraction. It would be interesting to confirm this phenomenon in different settings.

We then applied a network incorporating both Bayesian dropout and equivariance to the problem of predicting meiotic recombination hotpots directly from sequence. It was straightforward to interpret the resulting model, and interrogation of the sequences that maximally activated the input layer revealed the previously characterized 13-base PRDM9 binding motif (Myers *et al.*, 2008). Additionally, we discovered a much sparser motif, with only about 8/22 bases showing substantial information content. Like the classical motif, these new motifs were statistically significant on hold-out data, indicating that they were truly predictive features and not artefacts of an overtrained model. These sparse motifs (category A in Fig. 6) bear strong resemblance to motifs recently shown to be associated with PRDM9 binding (Altemose *et al.*, 2017), lending further support to this conclusion. In fact, the observation that the model was able to achieve predictive accuracy on a par with hotspots predicted using differential H3K4 trimethylation as an input [also from Altemose *et al.* (2017)] is consistent with the hypothesis that the neural network model represents a near-optimal model for PRDM9 binding (Fig. 5). The alternative but less parsimonious explanation is that our model and the H3K4me3 assay are suboptimal to the same degree.

From a practical point of view, we noted that the motifs generated from this network with Bayesian dropout were qualitatively different and more numerous than the motifs identified with a classical convolutional approach, and we see a degree of degeneracy among the motifs learned by our network. It remains to be seen whether these different binding motifs correspond to relate but slightly different binding modalities, or whether these motifs are a result of dropout training and provide a way for the network to robustly identify motifs in the presence of weight noise. This would be an interesting direction for further research. Certainly, if indeed these motifs do correspond to different binding modalities, each with slightly different binding affinities, it would explain why these networks are able to achieve the superior classification accuracies compared to standard models.

Acknowledgement

We thank Simon Myers for helpful discussions.

Funding

This work was supported by Wellcome Trust grants [090532/Z/09/Z, 203141/Z/16/Z].

Conflict of Interest: none declared.

References

Alipanahi,B. *et al.* (2015) Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nat. Biotechnol.*, **33**, 831–838.

Altemose,N. *et al.* (2017) A map of human prdm9 binding provides evidence for novel behaviors of PRDM9 and other zinc-finger proteins in meiosis. *ELife*, **6**, e28383.

Baudat,F. *et al.* (2013) Meiotic recombination in mammals: localization and regulation. *Nat. Rev. Genet.*, **14**, 94–806.

Clevert,D. *et al.* (2015) Fast and accurate deep network learning by exponential linear units (ELUs). *CoRR*, **1511**, 07289.

Cohen,T.S. and Welling,M. (2016) Group equivariant convolutional networks. *CoRR*, **1602**, 07576.

Gal,Y. and Ghahramani,Z. (2016) Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19–24, 2016*, pp. 1050–1059.

Glorot,X. and Bengio,Y. (2010) Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13–15, 2010*, pp. 249–256.

Heinz,S. *et al.* (2010) Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and b cell identities. *Mol. Cell*, **38**, 576–589.

Ioffe,S. and Szegedy,C. (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning*, Vol. 37, ICML'15, JMLR.org, pp. 448–456.

Kelley,D.R. *et al.* (2016) Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Res.*, **26**, 990–999.

Lanchantin,J. *et al.* (2016) Deep motif: visualizing genomic sequence classifications. *CoRR*, **1605**, 01133.

Myers,S. *et al.* (2008) A common sequence motif associated with recombination hot spots and genome instability in humans. *Nat. Genet.*, **40**, 1124–1129.

Quang,D. and Xie,X. (2016) DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Res.*, **44**, e107.

Sandelin,A. *et al.* (2004) JASPAR: an open - access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Res.*, **32**, D91–D94.

Shrikumar,A. *et al.* (2017) Reverse-complement parameter sharing improves deep learning models for genomics. https://doi.org/10.1101/103663.

Srivastava,N. *et al.* (2014) Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, **15**, 1929–1958.

Zhou,J. and Troyanskaya,O.G. (2015) Predicting the effects of noncoding variants with deep learning-based sequence model. *Nat. Methods*, **322**, 931–934.