Text S1. Details of Neural Network Layers

# Description of layer types

We here give a brief description of the functions calculated by the artificial neural network (ANN) layers employed in Applications 1-3.

# Input layer:

The input layer represents the network input as a real-valued array. Applications of ANNs in genomics frequently employ a one-hot representation in which a length L genomic sequence is represented as an array. Each row corresponds to a base position and has a single 1 entry indicting the nucleotide content of the base, with other row entries equal to zero. Other representations of sequence inputs are possible. For example, the ANN in Application 3 uses an input layer representation constructed by first representing sequences as one-hot arrays and then subtracting from each column the genome wide frequency, *f*, of the corresponding nucleotide and finally dividing this difference by (Methods: Application 3).

# Fully connected layers:

The vector of unit activations of a fully connected layer indexed by *i*  is the result of composing an affine transformation of the preceding layer activations with a non-linear operation:

(S1.1.1)

,

wheredenotes multiplication of the matrix *W* of weights for layer *i* with the vector of activations of the preceding layer, is a bias term for layer *i*, and **( )** denotes element-wise application of a non-linear function (also called an activation function). Applications 1-3 used two non-linear functions for **:** Rectified-linear (ReLU) units use . Sigmoid units use .

# Convolutional layers:

Convolutional layers are defined by a collection of *N* convolutional filters where and are the weight matrices and biases for the filter. Each filter is composed of units with the same and , but receive different local patches of preceding layer activations as input. In our models, the preceding layer is the input array, so that is an matrix where *h* is the number of base pairs (bps) considered by the unit. The activation of a unit sensitive to a sequence interval starting at position *p* is

(S1.1.2)

where denotes entry of the row and column of the array representation of the input sequence, and where denotes either a ReLU or sigmoid function. A vector of filter activations (also called a feature map) has elements given by (S1.1.2) for values of *p* starting with and incrementing by a common integer called the stride of the layer. The *N* filter activation vectors are concatenated before being passed to the subsequent fully connected layer of our models.

# Output Layers:

Application 1 employs a single output unit with full connection to the penultimate layer and a signmoid non-linear function. Outputs greater than 0.5 indicate classifcation to class 1; outputs less than 0.5 indicate classification to class 0.

Applications 2 and 3 use a two unit output layer with a softmax activation function. The softmax function, for classification to one of K classes, yields a length K vector with component *j* given by

(S1.1.3)

where denotes the the *kth* entry resulting from affine transformation of the vector of penultimate layer activations. That is,

(S1.1.4)

where (following the notation of “Results: Constructing an input-specific contrained maximum entropy distribution”) is the weight of connection of the *ith* penultimate unit to output unit *k*, is the activation of the *ith* unit in the penultimate layer with *M* units, and is the bias of ouput unit *k*.

# Motivation for the distance metric weights in multiclass classification

ANNs performing multiclass classification among *K* classes typically encode the predicted class probabilities by using *K* output units, one for each class, with their activations calculated from softmax functions acting on affine-transformed penultimate units (S1.1.3). The softmax function ensures that class probabilities are normalized to 1. For network input , the log ratio of predicted class 0 to class *k* ( ) probabilities is

where is as defined in (S1.1.3). Substituting equation (S1.1.4) yields

(S1.2.1)

Thus, the magnitude of the difference is a natural choice for measuring the extent to which features encoded in the penultimate activations allow the network to identify as belonging to class 0 rather than class *k.*

(S1.2.2)