RESEARCH

MAVE-NN: learning genotype-phenotype maps from multiplex assays of variant effect

Ammar Tareen[1], Anna Posfai[1], William T. Ireland[2], David M. McCandlish[1] and Justin B. Kinney[1*]

Abstract

Multiplex assays of variant effect (MAVEs), which include massively parallel reporter assays (MPRAs) and deep mutational scanning (DMS) experiments, are being rapidly adopted in many areas of biology. However, inferring quantitative models of genotype-phenotype (G-P) maps from MAVE data remains challenging, and different inference approaches have been advocated in different MAVE contexts. Here we introduce a conceptually unified approach to the problem of learning G-P maps from MAVE data. Our strategy is grounded in concepts from information theory, and is based on the view of G-P maps as a form of information compression. We also introduce MAVE-NN, a Python package that implements this approach using a neural network backend. The capabilities and advantages of MAVE-NN are then demonstrated on three diverse DMS and MPRA datasets. MAVE-NN thus fills a major need in the computational analysis of MAVE data. Installation instructions, tutorials, and documentation are provided at `https://mavenn.readthedocs.io`.

Keywords: multiplex assay of variant effect; neural networks; deep mutational scanning; massively parallel reporter assay; global epistasis; mutual information

Background

Over the last decade, the ability to quantitatively study genotype-phenotype (G-P) maps has been revolutionized by the development of multiplex assays of variant effect (MAVEs), which can measure molecular phenotypes for thousands to millions of genotypic variants in parallel [1]. MAVE is an umbrella term that describes a diverse set of experimental methods [2, 3], three examples of which are illustrated in **Fig. 1**. Deep mutational scanning (DMS) experiments are one large class of MAVE [4]. These work by linking proteins [5, 6, 7] or structural RNAs [8, 9, 10, 11] to their coding sequences, either directly or indirectly, then using deep sequencing to assay which variants survive a process of activity-dependent selection (**Fig. 1a**). Massively parallel reporter assays (MPRAs) are another major class of MAVE [12, 13, 14, 15], and are commonly used to study DNA or RNA sequences that regulate gene expression at a variety of steps, including transcription [16, 17, 18, 19, 20, 21], splicing [22, 23, 24, 25, 26], polyadenylation [27], and mRNA degradation [28, 29, 30, 31, 32]. Most MPRAs read out the expression of a reporter gene in one of two ways [1]: by quantifying RNA abundance via the sequencing of RNA barcodes that are linked to known variants (RNA-seq MPRAs; **Fig. 1c**), or by quantifying protein abundance using fluorescence-activated cell sorting (FACS) and then sequencing the sorted variants (sort-seq MPRAs; **Fig. 1e**).

MAVE data can enable rich quantitative modeling that goes far beyond the simple cataloguing of observed effects for individual variants. This key point was recognized in some of the earliest work on MAVEs [17, 18] and has persisted as a major theme in MAVE studies [33, 34, 35, 25, 31, 36]. But in contrast to MAVE experimental techniques, which continue to advance rapidly, there remain key gaps in the methodologies available for quantitatively modeling G-P maps from MAVE data.

Most computational methods for analyzing MAVE data have focused on accurately quantifying the activities of individual assayed sequences [37, 38, 39, 40, 41, 42, 43]. However, MAVE measurements for individual sequences often cannot be interpreted as providing direct quantification of the underlying G-P map that one is interested in. First, MAVE measurements are usually distorted by strong nonlinearities and noise, and distinguishing interesting properties of G-P maps from these confounding factors is not straight-forward. Second, MAVE data is often incomplete. Missing data is common, but a more fundamental issue is that researchers often want to understand G-P maps over

*Correspondence: jkinney@cshl.edu
[1]Simons Center for Quantitative Biology, Cold Spring Harbor Laboratory, 11375, Cold Spring Harbor, NY
Full list of author information is available at the end of the article

vastly larger regions of sequence space than can be exhaustively assayed.

Quantitative modeling can address both the incompleteness and indirectness of MAVE measurements [1]. The goal here is to determine a mathematical function that, given any sequence as input, will return a quantitative value for that sequence's molecular phenotype. Quantitative models thus fill in the gaps in G-P maps and, if appropriate inference methods are used, can further remove confounding effects of nonlinearities and noise. The simplest quantitative modeling strategy is linear regression (e.g. [18, 44]). However, linear regression yields valid results only when one's measurements are linear readouts of phenotype and exhibit uniform Gaussian noise. Such assumptions are often violated in dramatic fashion by MAVEs, and failure to account for this reality can give rise to major artifacts, such as spurious epistatic interactions [45].

Multiple MAVE analysis approaches that can separate the effects of nonlinearities and noise from underlying G-P maps have been reported, but a conceptually unified strategy is still needed. Work in the theoretical evolution literature has focused on a phenomenon called global epistasis (GE), in which measurements reflect a nonlinear function of an underlying "latent phenotype" [46, 47, 48, 49, 50, 51, 52, 53, 45]. In particular, Otwinowski et al. [46] described a regression approach in which one parametrically models G-P maps while nonparametrically modeling nonlinearities in the MAVE measurement process.

Parallel work in the biophysics literature has focused on developing ways to infer G-P maps from high-throughput data in a manner that is fully agnostic to the quantitative form of the measurement process used to read out latent phenotypes [54, 55, 56, 17]. This approach, which focuses on the use of mutual information as an objective function, arose from techniques in sensory neuroscience [57, 58] that were elaborated and adapted for the analysis of microarray data [56, 59], then later applied to MPRAs [34, 33, 17, 18] and other MAVE experiments [60]. However such analyses of MAVE data have relied on Metropolis Monte Carlo, which (in our experience) is too slow to support widespread adoption.

A third thread in the literature has arisen from efforts to apply techniques from deep learning to modeling G-P maps [61], including in the context of MAVEs [25, 28, 62, 27, 31]. Here the emphasis has been on using the highly expressive nature of deep neural networks to directly model experimental output from input sequences. Yet it has remained unclear how such neural networks might separate out the intrinsic features of G-P maps from effects of the MAVE measurement processes. This is a manifestation of the neural network interpretability problem, one that that is not addressed by established post-hoc attribution methods [63, 64].

Here we describe a unified conceptual framework for the quantitative modeling of MAVE data, one that unites the three strains of thought described above. As illustrated in **Fig. 2**, we assume that each sequence has a well-defined latent phenotype, of which the MAVE experiment provides a noisy indirect readout. To remove potentially confounding effects due to the quantitative form of this readout, we explicitly model both the G-P map and the MAVE "measurement process". As discussed in previous theoretical work [54, 55] and elaborated below, this separates the task of compressing sequence-encoded information (the job of the G-P map) from the task of mapping this information to a realistic experimental output (the job of the measurement process). Both the G-P map and the measurement process are formulated as neural networks, and their parameters are then chosen to maximize likelihood. Importantly, this strategy is equivalent to a variational approach to mutual information maximization in which one seeks a G-P map that compresses experimentally relevant information as losslessly as possible. Unlike previous implementations of mutual information maximization for MAVE analysis, however, this variational approach is compatible with backpropagation and is consequently much faster.

We also introduce MAVE-NN, a software package that makes this inference approach available to the broader MAVE community. MAVE-NN supports two distinct implementations of our unified conceptual approach: GE regression and measurement process agnostic (MPA) regression. GE regression is modeled after the approach of [46], while MPA regression resembles previously reported mutual information maximization analyses [34, 33, 17, 60]. In the following sections, we demonstrate the utility of MAVE-NN on previously published DMS and MPRA datasets. Along the way we note the substantial advantages that MAVE-NN provides over other state-of-the-art methods for modeling MAVE data.

Results

Modeling strategy

MAVE-NN supports the analysis of DNA, RNA, and protein sequences. All sequences must be the same length and, for the resulting models to be interpretable, must satisfy a natural notion of alignment. The measurement y corresponding to this sequence can be either continuous or discrete. Given a dataset comprising a set of N sequence-measurement pairs $\{(x_n, y_n)\}_{n=1}^{N}$, MAVE-NN aims to infer a probabilistic mapping $p(y|x)$. The primary enabling assumption

is that this mapping occurs in two stages (**Fig. 2a**). First, each sequence x is mapped to a latent phenotype ϕ via a deterministic G-P map $f(x)$. This latent phenotype is then probabilistically mapped to y via a the measurement process, i.e., a conditional distribution $p(y|\phi)$. During training, the G-P map $f(x)$ and measurement process $p(y|\phi)$ are simultaneously learned by maximizing a regularized form of likelihood. The present implementation of MAVE-NN supports only scalar values for ϕ, but multidimensional ϕ are also compatible with this conceptual framework.

Four different types of G-P maps are currently supported by MAVE-NN: "additive", "neighbor", "pairwise", and "blackbox". Additive G-P maps assume that each character at each position within a sequence contributes independently and additively to ϕ. Neighbor G-P maps incorporate interactions between nearest-neighbor characters, while pairwise G-P maps include interactions between all pairs of characters regardless of separation distance (see Methods). Blackbox G-P maps have the form of a densely connected multilayer perceptron (MLP), the specific architecture of which can be controlled by the user.

Two different strategies for modeling measurement processes are used. In MPA regression, MAVE-NN uses an overparameterized neural network to directly model $p(y|\phi)$. At present, MPA regression is only supported in cases where y is discrete. **Fig. 2b** illustrates one such measurement process, inferred from the sort-seq MPRA data of [17] (**Fig. 1**f). Global epistasis (GE) regression, by contrast, assumes additional structure in the measurement process (**Fig. 2c**). First, ϕ is deterministically mapped to a quantity $\hat{y}$ that represents the most probable measurement value. We call $\hat{y}$ the "prediction," and the function $g(\phi)$ the "nonlinearity." The prediction is then probabilistically mapped to measurements y through a conditional distribution $p(y|\hat{y})$ called the "noise model." MAVE-NN supports both homoskedastic and heteroskedastic noise models based on three different classes of distribution: Gaussian, Cauchy, and skewed-t. Notably, the skewed-t noise model [65] reduces to the Gaussian and Cauchy noise models in certain limits, while also accommodating highly asymmetric noise distributions. **Fig. 2d** illustrates a GE measurement process inferred from the DMS data of [66] (**Fig. 1**b).

G-P maps as information compression

It is useful to think of the MAVE-NN modeling approach in terms of information compression. In information theory, a quantity called "mutual information" quantifies the amount of information, measured in units of "bits", that one variable communicates about another [67, 68, 69]. Exactly computing the mutual information $I[a;b]$ between two variables a and b requires known their joint probability distribution $p(a,b)$. In data analysis contexts one typically does not have direct access to such joint distributions, and must instead estimate mutual information (and related quantities like entropy) from finite samples. MAVE-NN does this using a variety of approaches (see Methods).

In what follows we make use of three conceptually distinct information quantities: intrinsic information (I_{int}), predictive information (I_{pre}), and variational information (I_{var}). Intrinsic information, which we define as $I_{\text{int}} = I[x;y]$, is simply the mutual information between sequences and measurements. This quantity is intrinsic to each MAVE dataset and provides a benchmark against which to compare the performance of inferred G-P maps. Estimating I_{int} can be tricky because MAVE datasets usually provide only a sparse sampling of the joint distribution $p(x,y)$. There are, however, multiple strategies for estimating upper and lower bounds on this quantity. The specific techniques we use in this paper are described in Methods.

Each G-P map can be thought of as compressing the experimentally relevant information encoded in each (high-dimensional) assayed sequence x into a (low-dimensional) latent phenotype ϕ. When learning G-P maps from MAVE data, our goal to infer a quantitative model $f(x)$ for this G-P map that is as lossless as possible. In this vein we define the predictive information of a model, $I_{\text{pre}} = I[\phi;y]$, as the mutual information between MAVE measurements y and the predicted latent phenotype values ϕ.[1] When evaluated on test data, $I_{\text{pre}} \leq I_{\text{int}}$ with equality only when ϕ encodes the sequence-dependent information that actually affects the measurements. Importantly, I_{pre} characterizes the quality of the G-P map alone, and is not influenced by the inferred measurement process.

It is also useful to define what we call the "variational information," I_{var}. This quantity is a linear transformation of log likelihood and thus depends on both the G-P map and the measurement process. It is not a mutual information quantity per se, but rather serves as a variational lower bound on I_{pre} [72]. Indeed, the difference $I_{\text{pre}} - I_{\text{var}}$ quantifies how accurately the inferred measurement process matches the observed distribution of y and ϕ values (see Methods). I_{var} also serves as a useful metric during training because it can be rapidly computed at each iteration of the optimization algorithm.

MAVE-NN infers model parameters by maximizing a (lightly) regularized form of likelihood. These compu-

[1] We note that the term "predictive information" carries a different meaning in sensory neuroscience; e.g. see [70, 71]

tations are performed using the standard backpropagation-based training algorithms provided within the TensorFlow 2 backend. With certain caveats noted (see Methods), this optimization procedure also maximizes I_{pre} [55, 54] and does so without requiring costly estimates of mutual information at each iteration.

Application: Deep mutational scanning

To demonstrate MAVE-NN, we now turn to the DMS dataset of [66] (**Fig. 1**a). This study focused on protein G [73], an immunoglobulin-binding protein expressed in streptococcal bacteria that has long served as a model system for studying epistasis, protein folding, and the effects of mutations on protein function. In [66], the authors made all single and double mutations to the 55-residue B1 domain of protein G (GB1), which binds to immunoglobulin G (IgG). To quantify the IgG binding affinity of GB1 variants, the authors used mRNA display, an assay in which variant GB1 domains were covalently linked to their mRNAs and enriched using IgG beads. Deep sequencing of these mRNAs was then used to measure the enrichment of GB1 variants, as quantified by the ratio of read counts for selected mRNAs versus input mRNAs (**Fig. 1**b).

We sought to quantitatively model $\log_2$ enrichment (y) as a function of GB1 protein sequence (x). To this end, we used MAVE-NN to infer a GE regression model comprising an additive G-P map $\phi = f(x)$, a monotonic nonlinearity $\hat{y} = g(\phi)$, and a heteroskedastic skewed-t noise model $p(y|\hat{y})$. The results of this analysis are shown in **Fig. 3**. Panel **a** illustrates the additive G-P map via the effect ($\Delta\phi$) of every possible single-residue mutation. From this heatmap we can identify critical residues, including at positions 25, 29, 39, 42, and 50, from the fact that nearly all mutations at these positions substantially reduce ϕ. And as expected from biochemical considerations, mutations to proline also tend to negatively impact ϕ.

Fig. 3b illustrates the inferred measurement process $p(y|\phi)$, revealing a sigmoidal relationship between y and ϕ. The solid line indicates the inferred GE nonlinearity $g(\phi)$, i.e., the deterministic mapping from latent phenotype ϕ to prediction $\hat{y}$. Dashed lines show a corresponding central interval (CI) within which the inferred model anticipates 95% of MAVE measurements will fall. **Fig. 3c** provides a direct comparison of y to $\hat{y}$, again with the corresponding 95% CI shown.

As is typically the case with DMS datasets, there is good *a priori* reason to expect nonlinearities in the readout of the underlying G-P map [66]. In the simplest case, we can imagine that the Gibbs free energy of a GB1 variant bound to IgG is an additive function of GB1 sequence, reflecting no energetic epistasis between positions. Then even if the enrichment of variant GB1 molecules is performed under equilibrium thermodynamic conditions, so that enrichment values reflect the equilibrium occupancy of each GB1 variant, we would still observe a nonlinear relationship between the additive latent phenotype (binding energy) and the experimental readout (log enrichment), due to strongly bound variants having saturated occupancy. In more complicated scenarios, such as enrichment far from equilibrium, the experimental readout is likely to be even more nonlinear. And if any nonspecific binding also occurs, this relationship would become sigmoidal. Indeed, such sigmoidal nonlinearities are typical of MAVE datasets.

Our results thus far largely mirror those of Otwinowski *et al.* [46], who also fit a GE regression model with an additive G-P map to this GB1 dataset. There are important differences in our approach, however. Otwinowski et al. modeled the GE nonlinearity using splines, whereas MAVE-NN uses a mixture of sigmoids. They enforced monotonicity in $g(\cdot)$ by specifically using I-splines [74], whereas MAVE-NN does this by constraining component sigmoids to have nonnegative slope. Otwinowski et al. also assumed a uniform Gaussian noise model, whereas our analysis finds that a heteroskedastic skewed-t noise model describes the distribution of residuals far more accurately, thus leading to a more accurate G-P map (see below). Also, from a practical standpoint, MAVE-NN is far easier to use. Whereas Otwinowski et al. performed their analysis using custom Julia scripts, MAVE-NN provides a fully documented and thoroughly tested Python API that makes such modeling broadly accessible to the MAVE community.

The biggest conceptual innovation of MAVE-NN, we argue, is the information-theoretic perspective that it brings to the modeling of MAVEs. In standard methods for modeling G-P maps, it is typical to report quantities such as R^2 that quantify the fraction of variance explained by the model. However, these values are not robust to experimental nonlinearities and non-Gaussian noise. By contrast, I_{pre} and I_{int} retain their meaning regardless of how nonlinear or non-Gaussian the true measurement process is, and indeed the comparison of these two quantities provides a universal way of assessing model completeness. Likewise, comparisons of I_{var} to I_{pre} provide a way of evaluating how accurate one's model of the measurement process is. And although aspects of this information-theoretic approach have been discussed in prior work [56, 17, 55, 54], MAVE-NN provides the first computational implementation of these methods suitable for general use.

Fig. 3d summarizes the relevant information quantities for the GB1 analysis. We find a predictive infor-

mation value of $I_{\text{pre}} = 2.220 \pm 0.008$ bits, and a variational information value of $I_{\text{var}} = 2.194 \pm 0.020$ bits. The similarity of these two values suggests that the heteroskedastic skewed-t noise model has nearly sufficient accuracy to describe the distribution of residuals. Computing bounds on intrinsic information (see Methods), we find that I_{int} falls between 2.680 ± 0.008 bits and 3.213 ± 0.033 bits. Our inferred G-P map thus accounts for 70%-84% of the sequence-dependent information in the data, revealing that there is substantial structure in the true G-P map that is missed. This is in line with the finding of [75] that a biophysical model accounting for both GB1 folding and GB1-IgG binding better explains the data of [66] than a simple additive model.

It is worth noting that, when we use a homoskedastic Gaussian noise model (as in [46]), we obtain $I_{\text{pre}} = 2.115 \pm 0.010$ bits and $I_{\text{var}} = 1.758 \pm 0.017$ bits. The disparity between these values indicates substantial mismatch between the inferred noise model and the observed distribution of residuals. Nevertheless, I_{pre} is remarkably close to the value achieved by the heteroskedastic skewed-t model, indicating that this inferred G-P map has roughly similar (though noticeably less) accuracy.

Application: Massively parallel splicing assay

Pre-mRNA splicing, a process in which introns are excised from mRNA transcripts and the remaining exons are ligated together, is a key step in the expression of human genes. The boundaries between introns and their upstream exons are defined by 5′ splice sites, which bind the U1 snRNP during the initial stages of spliceosome assembly. 5′ splice site sequences are approximately 9 nucleotides in length and largely adhere to the motif NNN/GYNNNN.[2] Even mRNA sequences that have this motif, however, can vary dramatically in their splice site activity. A quantitative understanding of this variation is important for elucidating the fundamental biology of splicing as well as the causes of many genetic diseases [76, 77].

To this end, Wong *et al.* [26] used an MPSA to measure the splicing activity of nearly all 32,768 possible 9-nucleotide 5′ splice sites (**Fig. 1c**). Their experimental strategy used three-exon minigenes in which the 5′ splice site of the central exon was varied. Minigene constructs were transfected into HeLa cells, bulk RNA was extracted, and the fraction of processed transcripts containing the central exon was assayed using RT-PCR coupled to high-throughput sequencing. Specifically, the authors calculated a percent-spliced-in (PSI) value for each splice site variant based on the amount of exon inclusion mRNA relative to total mRNA (**Fig. 1d**). Here we discuss an analysis of one of the BRCA2 exon 17 splicing datasets reported in that work (library 1, replicate 1).

We used MAVE-NN to infer four different types of G-P maps from these data: additive, neighbor, pairwise, and blackbox. As with GB1, these G-P maps were inferred as part of a GE regression model with a monotonic nonlinearity and a heteroskedastic skewed-t noise model. For comparison, we also inferred an additive G-P map using the "epistasis" package of Sailer and Harms [50].

Fig. 4a shows the performance of these models on held-out test data. There are a few notable differences from the GB1 modeling results in **Fig. 3**. First, the information values in panel **a** are substantially lower, ranging from about 0.2-0.5 bits. This is largely due to the imbalanced nature of the MPSA dataset: the majority of assayed sequences are non-functional, with only about 5% having PSI values above background. The GB1 variants assayed by Olson et al., by contrast, have a much more even distribution of measured activities. We also observe a larger disparity between I_{pre} and I_{var} values. This likely reflects the skewed-t noise model not accounting well for a small number of "false positive" sequences that have substantial measured PSI but predicted $\hat{y}$ values close to background; see the central and upper-left regions of panels **c** and **d** in **Fig. 3**.

Unsurprisingly, we find that G-P maps of increasing complexity are able to explain held-out test data with increasing accuracy. For example, the additive G-P map exhibits only $I_{\text{pre}} = 0.262 \pm 0.011$ bits of predictive information, whereas the pairwise G-P map attains $I_{\text{pre}} = 0.367 \pm 0.015$ bits. Visually, we can see that the pairwise G-P map produces a larger region of ϕ values that correspond to $\hat{y}$ above background ($\phi \gtrsim 1.5$), and that the corresponding noise model has a tighter density in this region. The additive effects ($\Delta\phi$) and pairwise effects ($\Delta\Delta\phi$) components of this pairwise model are illustrated in **Fig. 4e** and **Fig. 4f**.

Even though they have the same mathematical form, the additive G-P map inferred using the epistasis Python package of [50] exhibits less predictive information ($I_{\text{pre}} = 0.220 \pm 0.012$ bits) than the additive G-P map inferred using MAVE-NN ($p = 0.007$, two sample Z-test). This is because the epistasis package estimates G-P map parameters using standard linear regression, and only after these are fixed is the GE nonlinearity inferred. Also, while the epistasis package provides a variety of options for modeling the GE nonlinearity, none of these options appears to work as well as our mixture-of-sigmoids approach; **Fig. 4b** shows the fit obtained using a power law nonlinearity, the specific nonlinearity studied in [50].

[2] N indicates any nucleotide, Y indicates a pyrimidine (C or U), and "/" indicates the exon/intron boundary.

The blackbox G-P map, comprising 5 densely connected hidden layers of 10 nodes each, performs the best of all four G-P maps, achieving a predictive information of $I_{\text{pre}} = 0.489 \pm 0.012$ bits. This suggests that even the pairwise model is not flexible enough to fully account for the MPSA data. Moreover, the blackbox I_{pre} value falls at or above the intrinsic information lower bound of 0.461 ± 0.007 bits, which we computed using data from a replicate experiment (see Methods). We were unable to compute a convincing upper bound on I_{int}, however, so the completeness of the blackbox G-P map is still unclear. Overall, this analysis highlights the need for models that go beyond simple position weight matrices. It also underscores the need for a single API that is capable of both inferring a variety of different G-P maps through a uniform interface, as well as assessing the performance of those G-P maps in a unified manner.

Application: Sort-seq MPRA

The *lac* promoter of *Escherichia coli* has long served as a model system for studying transcriptional regulation [78]. In one of the first MAVE studies, Kinney *et al.* [17] used this system to demonstrate the sort-seq approach to MPRAs (Fig. 1e). The authors created a library of *lac* promoters mutagenized within a 75 bp region that binds two transcriptional regulators, CRP and σ^{70} RNA polymerase (RNAP). These variant promoters were then used to drive the expression of GFP. Cells containing expression constructs were sorted using FACS and the variant promoters within each bin were then sequenced. The resulting data consisted of a list of unique promoter variants along with the number of times each variant was observed in each FACS bin (Fig. 1f).

Here we demonstrate MPA regression on the MPRA data of [17] by inferring additive models for the sequence-dependent activity of the RNAP binding site. Each row in **Fig. 5** represents a different sort-seq experiment reported in [17]; these five experiments were performed using different promoter libraries, host strains, and growth conditions. In each row, the left-most panel shows a sequence logo representing the $\theta_{l:c}$ parameters of the additive G-P map inferred by MPA regression. The center panel illustrates the corresponding inferred measurement processes, $p(y|\phi)$. Although these measurement processes differ greatly from experiment to experiment, the G-P map parameters are remarkably consistent with each other and with the known bipartite structure of the RNAP binding motif [79].

The G-P map parameters determined in [17] were trained by directly maximizing I_{pre} using a parallel tempering Monte Carlo algorithm, a procedure we refer to here as information maximization (IM) regression. The right-most panels in Fig. 5 plot these parameters against those inferred by MAVE-NN. These plots reveal a high level of correspondence, indicating that the two approaches yielded similar results.

To address the question of which models perform better, we also report the predictive information I_{pre} of both G-P maps, the values of which are displayed within each scatter plot. For the rnap-wt and full-500 datasets, MPA regression gives detectably higher I_{pre} (two sample Z-test), and on none of these datasets does MPA regression perform worse than IM. This indicates that the price paid for using a variational approach rather than directly maximizing I_{pre} is minimal, and is likely offset by the improved optimization obtained using stochastic gradient descent. Indeed, simulations show that MAVE-NN accurately recovers ground-truth G-P map parameters for RNAP from simulated data (**Fig. S3**). Moreover, MPA regression dramatically reduces the inference time compared to IM regression. To infer each of the models shown in **Fig. 5**, MAVE-NN required approximately one minute on a standard laptop computer, whereas the original IM regression computations of [17] required several hours on a computer cluster.

Discussion

In this work we have presented a unified strategy for inferring quantitative models of G-P maps from diverse MAVE datasets. At the core of our approach is the conceptualization of G-P maps as a form of information compression. Specifically, we assume that, in a MAVE experiment, the G-P map of interest first compresses an input sequence into a latent phenotype, which is then read out indirectly by a noisy measurement process. By explicitly modeling this measurement process along with the G-P map, one can remove potentially confounding effects. Along the way, we have described information-theoretic metrics for assessing the quality of such models.

To make our approach available to the broader MAVE community, we have also introduced a software package called MAVE-NN. We demonstrated the capabilities and performance of MAVE-NN in the context of three diverse MAVE experiments: a DMS assay [66], an RNA-seq MPRA [26], and a sort-seq MPRA [17]. In these contexts, MAVE-NN exhibits superior performance relative to the epistatis package of [50] and the mutual information maximization strategy of [17]. MAVE-NN has an easy-to-use Python API, is thoroughly tested, and can be installed from PyPI by executing "`pip install mavenn`". Comprehensive documentation as well as examples and step-by-step tutorials are available at http://mavenn.readthedocs.io.

Conclusion

We have introduced a unified information-theoretic approach to the analysis of MAVE datasets. Our software package, MAVE-NN, makes this approach accessible to the broader MAVE community. This work thus fills a critical need in the quantitatively modeling of G-P maps, and greatly advances the ability of researchers to comprehensively analyze data from the ever-expanding universe of MAVE experiments.

Methods

Notation

We represent each MAVE dataset as a set of N observations, $\{(x_n, y_n)\}_{n=0}^{N-1}$, where each observation consists of a sequence x_n and a measurement y_n.[3] Here, y_n can be either a continuous real-valued number, or a nonnegative integer representing the bin in which the nth sequence was found. Note that, in this representation the same sequence x can be observed multiple times, potentially with different values for y due to experimental noise.

Latent phenotype models

We assume that all sequences have the same length L, and that at each of the L positions in each sequence there is one of C possible characters ($C = 4$ for DNA and RNA; $C = 20$ for protein). MAVE-NN represents sequences using a vector of one-hot features of the form

$$x_{l:c} = \begin{cases} 1 & \text{if character } c \text{ occurs at position } l \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

where $0 = 1, 2, \ldots, L-1$ indexes positions within the sequence, and c indexes the C distinct characters.

We assume that the latent phenotype is given by a linear function $\phi(x;\theta)$ that depends on a set θ of G-P map parameters. As mentioned in the main text, MAVE-NN supports four types of G-P map models, all of which can be inferred using either GE regression or MPA regression. The "additive" model is given by,

$$\phi_{\text{additive}}(x;\theta) = \theta_0 + \sum_{l=0}^{L-1} \sum_{c} \theta_{l:c} x_{l:c}. \quad (2)$$

Here, each position in x contributes independently to the latent phenotype. The "neighbor" model is given by,

$$\phi_{\text{neighbor}}(x;\theta) = \theta_0 + \sum_{l=0}^{L-1} \sum_{c} \theta_{l:c} x_{l:c} + \sum_{l=0}^{L-2} \sum_{c,c'} \theta_{l:c,l+1:c'} x_{l:c} x_{l+1:c'}, \quad (3)$$

and further accounts for potential epistatic interactions between neighboring positions. The "pairwise" model is given by,

$$\phi_{\text{pairwise}}(x;\theta) = \theta_0 + \sum_{l=0}^{L-1} \sum_{c} \theta_{l:c} x_{l:c} + \sum_{l=0}^{L-2} \sum_{l'=l+1}^{L-1} \sum_{c,c'} \theta_{l:c,l':c'} x_{l:c} x_{l':c'}. \quad (4)$$

and includes interactions between all pairs of positions. Note our convention of requiring $l' > l$ in the pairwise parameters $\theta_{l:c,l':c'}$. Unlike these three parametric models, the "blackbox" G-P map does not have a fixed functional form. Rather, it is given by an MLP that takes a vector of sequence features as input and which outputs from a single node having linear activation. Users are able to specify the number of hidden layers, the number of nodes in each hidden layer, and the activation function used by these nodes.

Gauge modes and diffeomorphic modes

G-P maps typically have non-identifiable degrees of freedom that must be "fixed", i.e. pinned down, before the values of individual parameters can be meaningfully interpreted or compared between models. These degrees of freedom come in two flavors: gauge modes and diffeomorphic modes. Gauge modes are changes to θ that do not alter the values of the latent phenotype ϕ. Diffeomorphic modes [55, 54] are changes to θ that do alter ϕ, but do so in ways that can be undone by transformations of the measurement process $p(y|\phi)$ along corresponding "dual modes". As shown in [55], the diffeomorphic modes of linear G-P maps like those considered here will in general correspond to affine transformations of ϕ (though there are exceptions at special values of θ).

MAVE-NN fixes both gauge modes and diffeomorphic modes when returning parameter values or otherwise providing access to model internals. The diffeomorphic modes of G-P maps are fit by transforming θ via

$$\theta_0 \to \theta_0 - a, \quad (5)$$

and then

$$\theta \to \theta / b, \quad (6)$$

where $a = \text{mean}(\{\phi_n\})$ and $b = \text{std}(\{\phi_n\})$ are the mean and standard deviation of ϕ values computed on the training data. This produces a corresponding change in latent phenotype values $\phi \to (\phi - a)/b$. To avoid altering likelihood values, MAVE-NN also

[3] In this section we index starting from 0, as is done in the Python implementation of these methods.

transforms the measurement process $p(y|\phi)$ along corresponding dual modes. In GE regression this is done by adjusting the GE nonlinearity via

$$g(\phi) \to g(a + b\phi), \tag{7}$$

while keeping the noise model $p(y|\hat{y})$ fixed, whereas in MPA regression MAVE-NN adjusts the full measurement process,

$$p(y|\phi) \to p(y|a + b\phi). \tag{8}$$

For the three parametric G-P maps, gauge modes are fixed using what we call the "hierarchical gauge." Here, the parameters θ are adjusted so that the lower-order terms in $\phi(x;\theta)$ account for the highest possible fraction of variance in ϕ. This requires that the user specifies a probability distribution on sequence space, with respect to which these variances are computed. MAVE-NN assumes that such distributions factorize by position, and can thus be represented by a probability matrix with elements $p_{l:c}$, denoting the probability of character c at position l. MAVE-NN provides three built-in choices for this distribution: "uniform", "empirical", or "wildtype". The corresponding values of $p_{l:c}$ are given by

$$p_{l:c} \quad = \quad \begin{cases} 1/C & \text{for uniform,} \\ n_{l:c}/N & \text{for empirical,} \\ x^{\text{wt}}_{l:c} & \text{for wildtype,} \end{cases} \tag{9}$$

where $n_{l:c}$ denotes the number of sequences (out of N total) that have c at position l, and $x^{\text{wt}}_{l:c}$ is the one-hot encoding of a user-specified wild-type sequence. After a sequence distribution is chosen, MAVE-NN fixes the gauge of the pairwise G-P map by transforming

$$\begin{aligned} \theta_0 \quad \to \quad & \theta_0 \\ & + \sum_{l} \sum_{c'} \theta_{l:c'}\, p_{l:c'} \\ & + \sum_{l} \sum_{l'>l} \sum_{c,c'} \theta_{l:c,l':c'}\, p_{l:c}\, p_{l':c'}, \end{aligned} \tag{10}$$

$$\begin{aligned} \theta_{l:c} \quad \to \quad & \theta_{l:c} \\ & - \sum_{c'} \theta_{l:c'}\, p_{l:c'} \\ & + \sum_{l'>l} \sum_{c'} \theta_{l:c,l':c'}\, p_{l':c'} \\ & + \sum_{l'<l} \sum_{c'} \theta_{l':c',l:c}\, p_{l':c'} \\ & - \sum_{l'>l} \sum_{c',c''} \theta_{l:c',l':c''}\, p_{l:c'}\, p_{l':c''}, \\ & - \sum_{l'<l} \sum_{c',c''} \theta_{l':c'',l:c'}\, p_{l:c'}\, p_{l':c''}, \end{aligned} \tag{11}$$

and

$$\begin{aligned} \theta_{l:c,l':c'} \quad \to \quad & \theta_{l:c,l':c'} \\ & - \sum_{c''} \theta_{l:c'',l':c'}\, p_{l:c''} \\ & - \sum_{c''} \theta_{l:c,l':c''}\, p_{l':c''} \\ & + \sum_{c'',c'''} \theta_{l:c'',l':c'''}\, p_{l:c''}\, p_{l':c'''}. \end{aligned} \tag{12}$$

This transformation is also used for the additive and neighbor G-P maps, but with $\theta_{l:c,l':c'} = 0$ for all l, l' (additive) or whenever $l' \neq l + 1$ (neighbor).

GE nonlinearity

GE models assume that each measurement y is a nonlinear function of the latent phenotype, $g(\phi)$, plus some noise. In MAVE-NN, this nonlinearity is represented as a sum of tanh sigmoids:

$$g(\phi;\alpha) = a + \sum_{k=0}^{K-1} b_k \tanh(c_k\phi + d_k). \tag{13}$$

Here, K specifies the number of "hidden nodes" contributing to the sum, and $\alpha = \{a, b_k, c_k, d_k\}$ are trainable parameters. We note that this mathematical form is an example of the bottleneck architecture previously used by [48] for modeling GE nonlinearities. By default, MAVE-NN constrains $g(\phi;\alpha)$ to be monotonic in ϕ by requiring all $b_k \geq 0$ and $c_k \geq 0$, but this constraint can be relaxed.

GE noise models

MAVE-NN supports three types of GE noise models: Gaussian, Cauchy, and skew-t. The Gaussian noise model is given by

$$p_{\text{gauss}}(y|\hat{y};s) = \frac{1}{\sqrt{2\pi s^2}} \exp\left[-\frac{(y-\hat{y})^2}{2s^2}\right], \tag{14}$$

where s denotes the standard deviation. Importantly, MAVE-NN allows this noise model to be heteroskedastic by representing s as an exponentiated polynomial in $\hat{y}$, i.e.,

$$s(\hat{y}) = \exp\left[\sum_{k=0}^{K} a_k \hat{y}^k\right], \tag{15}$$

where K is the order of the polynomial and $\{a_k\}$ are trainable parameters. The user has the option to set K, and setting $K = 0$ renders this noise model homoskedastic. Quantiles are computed using $y_q = \hat{y} + s\sqrt{2}\,\mathrm{erf}^{-1}(2q-1)$ for user-specified values of $q \in [0,1]$.

Similarly, the Cauchy noise model is given by

$$p_{\mathrm{cauchy}}(y|\hat{y};s) = \left[\pi s\left(1 + \frac{(y-\hat{y})^2}{s^2}\right)\right]^{-1} \tag{16}$$

where the scale parameter s is an exponentiated K'th order polynomial in $\hat{y}$. Quantiles are computed using $y_q = \hat{y} + s\tan[\pi(q - \frac{1}{2})]$.

The skew-t noise model is of the form described by Jones and Faddy [65], and is given by

$$p_{\mathrm{skewt}}(y|\hat{y};s,a,b) = s^{-1} f(t;a,b), \tag{17}$$

where

$$t = t^* + \frac{y-\hat{y}}{s}. \quad t^* = \frac{(a-b)\sqrt{a+b}}{\sqrt{2a+1}\sqrt{2b+1}}, \tag{18}$$

and

$$\begin{aligned} f(t;a,b) &= \frac{2^{1-a-b}}{\sqrt{a+b}}\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\left[1 + \frac{t}{\sqrt{a+b+t^2}}\right]^{a+\frac{1}{2}} \\ &\quad \times \left[1 - \frac{t}{\sqrt{a+b+t^2}}\right]^{b+\frac{1}{2}}. \end{aligned} \tag{19}$$

Note that the t statistic here is an affine function of y chosen so that the distribution's mode (corresponding to t^*) is positioned at $\hat{y}$. The three parameters of this noise model, $\{s,a,b\}$, are each represented using K-th order exponentiated polynomials with trainable coefficients. Quantiles are computed using

$$y_q = \hat{y} + (t_q - t^*)s, \tag{20}$$

where

$$t_q = \frac{(2x_q - 1)\sqrt{a+b}}{\sqrt{1-(2x_q-1)^2}}, \quad x_q = I_q^{-1}(a,b), \tag{21}$$

and I^{-1} denotes the inverse of the regularized incomplete Beta function $I_x(a,b)$.

MPA measurement process

In MPA regression, MAVE-NN directly models the measurement process $p(y|\phi)$. At present, MAVE-NN only supports MPA regression for discrete values of y, which are indexed using nonnegative integers. MAVE-NN takes two alternative forms of input for MPA regression. One is a set of (non-unique) sequence-measurement pairs $\{(x_n, y_n)\}_{n=0}^{N-1}$, where N is the total number of independent measurements and each $y_n \in \{0, 1, \ldots, Y-1\}$, where Y is the total number of bins. The other is a set of (unique) sequence-count-vector pairs $\{(x_m, c_m)\}_{m=0}^{M-1}$, where M is the total number of unique sequences in the data set, and $c_m = (c_{m0}, c_{m1}, \ldots, c_{m(Y-1)})$ is a vector that lists the number of times, c_{my}, that the sequence x_m was observed in each bin y.

MPA measurement processes are represented as MLPs with one hidden layer (having tanh activations) and a softmax output layer. Specifically,

$$p(y|\phi) = \frac{w_y(\phi)}{\sum_{y'} w_{y'}(\phi)}, \tag{22}$$

$$w_y(\phi) = \exp\left[a_y + \sum_{k=0}^{K-1} b_{yk}\tanh(c_{yk}\phi + d_{yk})\right], \tag{23}$$

where K is the number of hidden nodes per value of y. The trainable parameters of this measurement process are $\eta = \{a_y, b_{yk}, c_{yk}, d_{yk}\}$.

Loss function

Let θ denote the G-P map parameters, and η denote the parameters of the measurement process. MAVE-NN optimizations these parameters using stochastic gradient descent (SGD) on a loss function given by

$$\mathcal{L} = \mathcal{L}_{\mathrm{like}} + \mathcal{L}_{\mathrm{reg}} \tag{24}$$

where $\mathcal{L}_{\mathrm{like}}$ is the negative log likelihood of the model, given by

$$\mathcal{L}_{\mathrm{like}}[\theta,\eta] = -\sum_{n=0}^{N-1}\log\left[p(y_n|\phi_n;\eta)\right], \ \phi_n = \phi(x_n;\theta), \tag{25}$$

and $\mathcal{L}_{\mathrm{reg}}$ provides for regularization of the model parameters.

In the context of GE regression, we can write $\eta = (\alpha, \beta)$ where α represents the parameters of the GE nonlinearity $g(\phi;\alpha)$, and β denotes the parameters of the noise model $p(y|\hat{y};\beta)$. The likelihood contribution from each observation n then becomes $p(y_n|\phi_n;\eta) = p(y_n|\hat{y}_n;\beta)$ where $\hat{y}_n = g(\phi_n;\alpha)$. In the

context of MPA regression with a dataset of the form $\{(x_m, c_m)\}_{m=1}^{M}$, the loss function simplifies to

$$\mathcal{L}_{\text{like}}[\theta, \eta] = \sum_{m=0}^{M-1} \sum_{y=0}^{Y-1} c_{my} \log[p(y|\phi_m; \eta)], \quad (26)$$

where $\phi_m = \phi(x_m; \theta)$. For the regularization term, MAVE-NN uses an L_2 penalty of the form

$$\mathcal{L}_{\text{reg}}[\theta, \eta] = \lambda_\theta |\theta|^2 + \lambda_\eta |\eta|^2, \quad (27)$$

where λ_θ and λ_η respectively control the strength of regularization for the G-P map and measurement process parameters. These parameters are user-adjustable, with a default value of 0.1 is used. We have not observed the specific values of these parameters to noticeably influence results.

Predictive information

In what follows, we use $p_{\text{model}}(y|\phi)$ to denote a measurement process inferred by MAVE-NN, whereas $p_{\text{true}}(y|\phi)$ denotes the empirical conditional distribution of y and ϕ values that would be observed in the limit of infinite test data.

The predictive information $I_{\text{pre}} = I[y; \phi]$, when computed on data not used for training (i.e., a held out test set or data from a different experiment), provides a measure of how strongly a G-P map predicts experimental measurements. Importantly, this quantity does not depend on the corresponding measurement process $p_{\text{model}}(y|\phi)$. To estimate I_{pre}, we use k'th nearest neighbor (kNN) estimators of entropy and mutual information adapted from the NPEET Python package [80]. Here, the user has the option of adjusting k, which controls a variance/bias tradeoff. When y is discrete (MPA regression), I_{pre} is computed using the classic kNN entropy estimator [81, 82] via the decomposition $I[y; \phi] = H[\phi] - \sum_y p(y) H_y[\phi]$, where $H_y[\phi]$ denotes the entropy of $p_{\text{true}}(\phi|y)$. When y is continuous (GE regression), $I[y; \phi]$ is estimated using the kNN-based Kraskov-Stögbauer-Grassberger (KSG) algorithm [82]. This approach optionally supports a local nonuniformity correction of [83], which is important when y and ϕ exhibit strong dependencies, but which also requires substantially more time to compute.

Variational information

We define the "variational information", I_{var}, as an affine transformation of $\mathcal{L}_{\text{like}}$,

$$I_{\text{var}} = H[y] - \frac{\log_2(e)}{N} \mathcal{L}_{\text{like}}. \quad (28)$$

Here, $H[y]$ is the entropy of the data $\{y_n\}$, which is estimated using the k'th nearest neighbor (kNN) estimator from the NPEET package [80]. Noting that this quantity can also be written as $I_{\text{var}} = H[y] - \text{mean}(\{Q_n\})$ where $Q_n = -\log_2 p(y_n|\phi_n)$, we estimate the associated uncertainty using

$$\delta I_{\text{var}}[y; \phi] = \sqrt{\delta H[y]^2 + \text{var}(\{Q_n\})/N}. \quad (29)$$

The inference strategy used by MAVE-NN is based on the fact that I_{var} provides a tight variational lower bound on I_{pre} [72, 55]. Indeed, in the large data limit,

$$I_{\text{pre}} = I_{\text{var}} + D_{\text{KL}}(p_{\text{true}} || p_{\text{model}}), \quad (30)$$

where $D_{\text{KL}}(\cdot) \geq 0$ is the Kullback-Leibler divergence between p_{true} and p_{model}, and thus quantifies the accuracy of the inferred measurement process. From Eq. 30 one can see that, with appropriate caveats, maximizing I_{var} (or equivalently, $\mathcal{L}_{\text{like}}$) will also maximize I_{pre} [55]. But unlike I_{pre}, I_{var} is compatible with backpropagation and stochastic gradient descent.[4] See Supplemental Information for a derivation of Eq. 30 and an expanded discussion of this key point.

Intrinsic information

Intrinsic information, $I_{\text{int}} = I[x; y]$, is the mutual information between the sequences x and measurements y in a dataset. This quantity is somewhat tricky to estimate, due to the high-dimensional nature of sequence space. We instead used three different methods to obtain the upper and lower bounds on I_{int} shown in Figs. 3d and 4c. More generally, we believe the development of both computational and experimental methods for estimating I_{int} is be an important avenue for future research.

To compute the upper bound on I_{int} for GB1 data (in Fig. 3d), we used the fact that

$$I[x; y] = H[y] - \langle H_x[y] \rangle_x \quad (31)$$

where $H[y]$ is the entropy of all measurements y, $H_x[y]$ is the entropy of $p(y|x)$ for a specific choice of sequence x, and $\langle \cdot \rangle_x$ indicates averaging over all sequences x. In this dataset, the measurement values were computed using

$$y = \log_2 \left[\frac{c_s + 1}{c_i + 1} \right] \quad (32)$$

[4] Sharpee et al. [57] cleverly showed that I_{pre} can, in fact, be optimized using stochastic gradient descent. Computing gradients of I_{pre}, however, requires a time-consuming density estimation step. Optimizing I_{var}, on the other hand, can be done using standard per-datum backpropagation.

where c_i is the input read count and c_s is the selected read count. $H[y]$ was estimated using the KNN estimator [80]. We then estimated the uncertainty in y by propagating errors expected due to Poisson fluctuations in read counts, which gives

$$\delta y = \log_2(e)\sqrt{\frac{1}{c_s+1}+\frac{1}{c_i+1}}, \tag{33}$$

and, assuming $p(y|x)$ to be approximately Gaussian, a corresponding conditional entropy

$$H_x[y] = \frac{1}{2}\log_2(2\pi e\,\delta y^2). \tag{34}$$

These $H[y]$ and $H_x[y]$ values were then used in Eq. 31 to estimate I_{int}. We expect this to provide an upper bound because the true uncertainty in y must be at least that expected under Poisson sampling. We note, however, that the use of linear error propagation and the assumption that $p(y|x)$ is approximately Gaussian complicate this conclusion. Also, when applied to MPSA data, this method yielded an upper bound of 0.96 bits. We believe this value is likely to be far higher than the true value of I_{int}, and that this mismatch probably resulted from read counts in the MPSA data being over-dispersed.

To compute the lower bound on I_{int} for GB1 data (Fig. 3d) we used the predictive information I_{pre} (on test data) of a GE regression model having a blackbox G-P map. This provides a lower bound because $I_{\text{int}} \geq I_{\text{pre}}$ for any model (when evaluated on test data) due to the Data Processing Inequality and the Markov Chain nature of the dependencies $y \leftarrow x \rightarrow \phi$ in **Fig. 2e**.

To compute a lower bound on I_{int} for MPSA data (**Fig. 4c**), we leveraged the availability of replicate data in [26]. Let y and y' represent the original and replicate measurements obtained for a sequence x. Because $y \leftarrow x \rightarrow y'$ forms a Markov chain, $I[x;y] \geq I[y;y']$. We therefore used an estimate of $I[y;y']$, computed using the method of [82], as the lower bound for I_{int}.

Uncertainties in kNN estimates

MAVE-NN quantifies uncertainties in $H[y]$ and $I[y;\phi]$ using multiple random samples of half the data. Let $\mathcal{D}_{100\%}$ denote a full dataset, and let $\mathcal{D}_{50\%,r}$ denote a 50% subsample (indexed by r) of this dataset. Given an estimator $E(\cdot)$ of either entropy or mutual information, as well as the number of subsamples R to use, the uncertainty in $E(\mathcal{D}_{100\%})$ is estimated as

$$\delta E(\mathcal{D}_{100\%}) = \frac{1}{\sqrt{2}}\text{std}\left[\left\{E(\mathcal{D}_{50\%,r})\right\}_{r=0}^{R-1}\right]. \tag{35}$$

By default MAVE-NN uses $R = 25$. We note that computing such uncertainty estimates substantially increase computation time, as $E(\cdot)$ needs to be evaluated $R+1$ times instead of just once. We also note that bootstrap resampling [84, 85] is not advisable in this context, as it systematically underestimates $H[y]$ and overestimates $I[y;z]$ (data not shown).

Funding

This work was supported by NIH grant 1R35GM133777 (awarded to JBK), NIH Grant 1R35GM133613 (awarded to DMM), an Alfred P. Sloan Research Fellowship (awarded to DMM), a grant from the CSHL/Northwell Health partnership, and funding from the Simons Center for Quantitative Biology at Cold Spring Harbor Laboratory.

Availability of data and materials

- Project: mavenn
- Documentation: mavenn.readthedocs.io
- Programming language: Python
- Installation: `pip install mavenn`
- License: MIT
- Restrictions on use by non-academics: None

Competing interests

The authors declare that they have no competing interests.

Author's contributions

JBK, AT, and DMM conceived the project. AT and JBK wrote the software. AT tested the software and released it as a python package on PYPI. AT, DMM, and JBK wrote the manuscript. WTI wrote a preliminary version of the software. AP performed the gauge fixing analysis. All authors contributed to aspects of the analyses.

Acknowledgements

The authors thank Jesse Bloom and Peter Koo for providing valuable feedback on the manuscript.

Author details

[1]Simons Center for Quantitative Biology, Cold Spring Harbor Laboratory, 11375, Cold Spring Harbor, NY. [2]Department of Physics, California Institute of Technology, 91125, Pasadena, CA.

References

1. Kinney, J.B., McCandlish, D.M.: Massively Parallel Assays and Quantitative Sequence–Function Relationships. Annual Review of Genomics and Human Genetics **20**(1), 99–127 (2019). doi:10.1146/annurev-genom-083118-014845
2. Starita, L.M., Ahituv, N., Dunham, M.J., Kitzman, J.O., Roth, F.P., Seelig, G., Shendure, J., Fowler, D.M.: Variant Interpretation: Functional Assays to the Rescue. American journal of human genetics **101**(3), 315–325 (2017). doi:10.1016/j.ajhg.2017.07.014
3. Esposito, D., Weile, J., Shendure, J., Starita, L.M., Papenfuss, A.T., Roth, F.P., Fowler, D.M., Rubin, A.F.: MaveDB: an open-source platform to distribute and interpret data from multiplexed assays of variant effect. Genome Biology **20**(1), 223 (2019). doi:10.1186/s13059-019-1845-6
4. Fowler, D.M., Fields, S.: Deep mutational scanning: a new style of protein science. Nat Methods **11**(8), 801–807 (2014). doi:10.1038/nmeth.3027
5. Fowler, D.M., Araya, C.L., Fleishman, S.J., Kellogg, E.H., Stephany, J.J., Baker, D., Fields, S.: High-resolution mapping of protein sequence-function relationships. Nat Methods **7**(9), 741–746 (2010). doi:10.1038/nmeth.1492
6. Hietpas, R.T., Jensen, J.D., Bolon, D.N.A.: Experimental illumination of a fitness landscape. Proc Natl Acad Sci USA **108**(19), 7896–7901 (2011). doi:10.1073/pnas.1016024108
7. McLaughlin, R.N., Poelwijk, F.J., Raman, A., Gosal, W.S., Ranganathan, R.: The spatial architecture of protein function and adaptation. Nature **491**(7422), 138–142 (2012). doi:10.1038/nature11500

8. Pitt, J.N., Ferré-D'Amaré, A.R.: Rapid construction of empirical RNA fitness landscapes. Science **330**(6002), 376–379 (2010). doi:10.1126/science.1192001
9. Puchta, O., Cseke, B., Czaja, H., Tollervey, D., Sanguinetti, G., Kudla, G.: Network of epistatic interactions within a yeast snoRNA. Science **352**(6287), 840–844 (2016). doi:10.1126/science.aaf0965
10. Li, C., Qian, W., Maclean, C.J., Zhang, J.: The fitness landscape of a tRNA gene. Science **352**(6287), 837–840 (2016). doi:10.1126/science.aae0568
11. Domingo, J., Diss, G., Lehner, B.: Pairwise and higher-order genetic interactions during the evolution of a tRNA. Nature **558**(7708), 117–121 (2018). doi:10.1038/s41586-018-0170-7
12. Inoue, F., Ahituv, N.: Decoding enhancers using massively parallel reporter assays. Genomics **106**(3), 159–164 (2015). doi:10.1016/j.ygeno.2015.06.005
13. Levo, M., Segal, E.: In pursuit of design principles of regulatory sequences. Nature reviews Genetics **15**(7), 453–468 (2014). doi:10.1038/nrg3684
14. Peterman, N., Levine, E.: Sort-seq under the hood: implications of design choices on large-scale characterization of sequence-function relations. BMC Genomics **17**(1), 206 (2016). doi:10.1186/s12864-016-2533-5
15. White, M.A.: Understanding how cis-regulatory function is encoded in DNA sequence using massively parallel reporter assays and designed sequences. Genomics **106**(3), 165–170 (2015). doi:10.1016/j.ygeno.2015.06.003
16. Patwardhan, R.P., Lee, C., Litvin, O., Young, D.L., Pe'er, D., Shendure, J.: High-resolution analysis of DNA regulatory elements by synthetic saturation mutagenesis. Nat Biotechnol **27**(12), 1173–1175 (2009). doi:10.1038/nbt.1589
17. Kinney, J.B., Murugan, A., Callan, C.G., Cox, E.C.: Using deep sequencing to characterize the biophysical mechanism of a transcriptional regulatory sequence. Proceedings of the National Academy of Sciences **107**(20), 9158–9163 (2010). doi:10.1073/pnas.1004290107
18. Melnikov, A., Murugan, A., Zhang, X., Tesileanu, T., Wang, L., Rogov, P., Feizi, S., Gnirke, A., Callan, C.G., Kinney, J.B., Kellis, M., Lander, E.S., Mikkelsen, T.S.: Systematic dissection and optimization of inducible enhancers in human cells using a massively parallel reporter assay. Nature Biotechnology **30**(3), 271–277 (2012). doi:10.1038/nbt.2137
19. Patwardhan, R.P., Hiatt, J.B., Witten, D.M., Kim, M.J., Smith, R.P., May, D., Lee, C., Andrie, J.M., Lee, S.-I., Cooper, G.M., Ahituv, N., Pennacchio, L.A., Shendure, J.: Massively parallel functional dissection of mammalian enhancers in vivo. Nat Biotechnol **30**(3), 265–270 (2012). doi:10.1038/nbt.2136
20. Kwasnieski, J.C., Mogno, I., Myers, C.A., Corbo, J.C., Cohen, B.A.: Complex effects of nucleotide variants in a mammalian cis-regulatory element. Proc Natl Acad Sci USA **109**(47), 19498–19503 (2012). doi:10.1073/pnas.1210678109
21. Sharon, E., Kalma, Y., Sharp, A., Raveh-Sadka, T., Levo, M., Zeevi, D., Keren, L., Yakhini, Z., Weinberger, A., Segal, E.: Inferring gene regulatory logic from high-throughput measurements of thousands of systematically designed promoters. Nat Biotechnol **30**(6), 521–530 (2012). doi:10.1038/nbt.2205
22. Cheung, R., Insigne, K.D., Yao, D., Burghard, C.P., Wang, J., Hsiao, Y.-H.E., Jones, E.M., Goodman, D.B., Xiao, X., Kosuri, S.: A Multiplexed Assay for Exon Recognition Reveals that an Unappreciated Fraction of Rare Genetic Variants Cause Large-Effect Splicing Disruptions. Mol Cell **73**(1), 183–1948 (2019). doi:10.1016/j.molcel.2018.10.037
23. Ke, S., Chasin, L.A.: Context-dependent splicing regulation: exon definition, co-occurring motif pairs and tissue specificity. RNA biology **8**(3), 384–388 (2011). doi:10.4161/rna.8.3.14458
24. Ke, S., Anquetil, V., Zamalloa, J.R., Maity, A., Yang, A., Arias, M.A., Kalachikov, S., Russo, J.J., Ju, J., Chasin, L.A.: Saturation mutagenesis reveals manifold determinants of exon definition. Genome Res **28**(1), 11–24 (2018). doi:10.1101/gr.219683.116
25. Rosenberg, A.B., Patwardhan, R.P., Shendure, J., Seelig, G.: Learning the Sequence Determinants of Alternative Splicing from Millions of Random Sequences. Cell **163**(3), 698–711 (2015). doi:10.1016/j.cell.2015.09.054
26. Wong, M.S., Kinney, J.B., Krainer, A.R.: Quantitative Activity Profile and Context Dependence of All Human 5' Splice Sites. Molecular cell **71**(6), 1012–10263 (2018). doi:10.1016/j.molcel.2018.07.033
27. Bogard, N., Linder, J., Rosenberg, A.B., Seelig, G.: A Deep Neural Network for Predicting and Engineering Alternative Polyadenylation. Cell **178**(1), 91–10623 (2019). doi:10.1016/j.cell.2019.04.046
28. Cuperus, J.T., Groves, B., Kuchina, A., Rosenberg, A.B., Jojic, N., Fields, S., Seelig, G.: Deep learning of the regulatory grammar of yeast 5' untranslated regions from 500,000 random sequences. Genome Res **27**(12), 2015–2024 (2017). doi:10.1101/gr.224964.117
29. Dvir, S., Velten, L., Sharon, E., Zeevi, D., Carey, L.B., Weinberger, A., Segal, E.: Deciphering the rules by which 5'-UTR sequences affect protein expression in yeast. Proc Natl Acad Sci USA **110**(30), 2792–801 (2013). doi:10.1073/pnas.1222534110
30. Oikonomou, P., Goodarzi, H., Tavazoie, S.: Systematic identification of regulatory elements in conserved 3' UTRs of human transcripts. Cell reports **7**(1), 281–292 (2014). doi:10.1016/j.celrep.2014.03.001
31. Sample, P.J., Wang, B., Reid, D.W., Presnyak, V., McFadyen, I.J., Morris, D.R., Seelig, G.: Human 5' UTR design and variant effect prediction from a massively parallel translation assay. Nature Biotechnology **37**(7), 803–809 (2019). doi:10.1038/s41587-019-0164-5
32. Shalem, O., Sharon, E., Lubliner, S., Regev, I., Lotan-Pompan, M., Yakhini, Z., Segal, E.: Systematic dissection of the sequence determinants of gene 3' end mediated expression control. PLoS genetics **11**(4), 1005147 (2015). doi:10.1371/journal.pgen.1005147
33. Belliveau, N.M., Barnes, S.L., Ireland, W.T., Jones, D.L., Sweredoski, M.J., Moradian, A., Hess, S., Kinney, J.B., Phillips, R.: Systematic approach for dissecting the molecular mechanisms of transcriptional regulation in bacteria. Proceedings of the National Academy of Sciences **115**(21), 201722055 (2018). doi:10.1073/pnas.1722055115
34. Barnes, S.L., Belliveau, N.M., Ireland, W.T., Kinney, J.B., Phillips, R.: Mapping DNA sequence to transcription factor binding energy in vivo. PLOS Computational Biology **15**(2), 1006226 (2019). doi:10.1371/journal.pcbi.1006226
35. Boer, C.G.d., Vaishnav, E.D., Sadeh, R., Abeyta, E.L., Friedman, N., Regev, A.: Deciphering eukaryotic gene-regulatory logic with 100 million random promoters. Nature Biotechnology **38**(1), 56–65 (2019). doi:10.1038/s41587-019-0315-8
36. Kemble, H., Nghe, P., Tenaillon, O.: Recent insights into the genotype–phenotype relationship from massively parallel genetic assays. Evolutionary Applications **12**(9), 1721–1742 (2019). doi:10.1111/eva.12846
37. Fowler, D.M., Araya, C.L., Gerard, W., Fields, S.: Enrich: software for analysis of protein function by enrichment and depletion of variants. Bioinformatics **27**(24), 3430–3431 (2011). doi:10.1093/bioinformatics/btr577
38. Alam, K.K., Chang, J.L., Burke, D.H.: FASTAptamer: A Bioinformatic Toolkit for High-throughput Sequence Analysis of Combinatorial Selections. Mol Ther Nucleic Acids **4**(3), 230 (2015). doi:10.1038/mtna.2015.4
39. Bloom, J.D.: Software for the analysis and visualization of deep mutational scanning data. BMC Bioinformatics **16**(1), 168 (2015). doi:10.1186/s12859-015-0590-4
40. Rubin, A.F., Gelman, H., Lucas, N., Bajjalieh, S.M., Papenfuss, A.T., Speed, T.P., Fowler, D.M.: A statistical framework for analyzing deep mutational scanning data. Genome Biol **18**(1), 1–15 (2017). doi:10.1186/s13059-017-1272-5
41. Ashuach, T., Fischer, D.S., Kreimer, A., Ahituv, N., Theis, F.J., Yosef, N.: MPRAnalyze: statistical framework for massively parallel reporter assays. Genome Biology **20**(1), 183 (2019). doi:10.1186/s13059-019-1787-z
42. Niroula, A., Ajore, R., Nilsson, B.: MPRAscore: robust and non-parametric analysis of massively parallel reporter assays. Bioinformatics **35**(24), 5351–5353 (2019). doi:10.1093/bioinformatics/btz591
43. Faure, A.J., Schmiedel, J.M., Baeza-Centurion, P., Lehner, B.: DiMSum: an error model and pipeline for analyzing deep mutational scanning data and diagnosing common experimental pathologies. bioRxiv, 2020–0625171421 (2020). doi:10.1101/2020.06.25.171421

44. Myint, L., Avramopoulos, D.G., Goff, L.A., Hansen, K.D.: Linear models enable powerful differential activity analysis in massively parallel reporter assays. BMC Genomics **20**(1), 209 (2019). doi:10.1186/s12864-019-5556-x
45. Baeza-Centurion, P., Miñana, B., Schmiedel, J.M., Valcárcel, J., Lehner, B.: Combinatorial Genetics Reveals a Scaling Law for the Effects of Mutations on Splicing. Cell **176**(3), 549–56323 (2019). doi:10.1016/j.cell.2018.12.010
46. Otwinowski, J., McCandlish, D.M., Plotkin, J.B.: Inferring the shape of global epistasis. Proc Natl Acad Sci USA **115**(32), 7550–7558 (2018). doi:10.1073/pnas.1804015115
47. Otwinowski, J., Nemenman, I.: Genotype to phenotype mapping and the fitness landscape of the E. coli lac promoter. PLoS ONE **8**(5), 61570 (2013). doi:10.1371/journal.pone.0061570. 1206.4209
48. Sarkisyan, K.S., Bolotin, D.A., Meer, M.V., Usmanova, D.R., Mishin, A.S., Sharonov, G.V., Ivankov, D.N., Bozhanova, N.G., Baranov, M.S., Soylemez, O., Bogatyreva, N.S., Vlasov, P.K., Egorov, E.S., Logacheva, M.D., Kondrashov, A.S., Chudakov, D.M., Putintseva, E.V., Mamedov, I.Z., Tawfik, D.S., Lukyanov, K.A., Kondrashov, F.A.: Local fitness landscape of the green fluorescent protein. Nature **533**(7603), 397–401 (2016). doi:10.1038/nature17995
49. Pokusaeva, V.O., Usmanova, D.R., Putintseva, E.V., Espinar, L., Sarkisyan, K.S., Mishin, A.S., Bogatyreva, N.S., Ivankov, D.N., Akopyan, A.V., Avvakumov, S.Y., Povolotskaya, I.S., Filion, G.J., Carey, L.B., Kondrashov, F.A.: An experimental assay of the interactions of amino acids from orthologous sequences shaping a complex fitness landscape. PLOS Genetics **15**(4), 1008079 (2019). doi:10.1371/journal.pgen.1008079
50. Sailer, Z.R., Harms, M.J.: Detecting High-Order Epistasis in Nonlinear Genotype-Phenotype Maps. Genetics **205**(3), 1079–1088 (2017). doi:10.1534/genetics.116.195214
51. Sailer, Z.R., Harms, M.J.: Uninterpretable interactions: epistasis as uncertainty. bioRxiv, 378489 (2018). doi:10.1101/378489
52. Fernandez-de-Cossio-Diaz, J., Uguzzoni, G., Pagnani, A.: Unsupervised inference of protein fitness landscape from deep mutational scan. bioRxiv, 2020–0318996595 (2020). doi:10.1101/2020.03.18.996595
53. Domingo, J., Baeza-Centurion, P., Lehner, B.: The Causes and Consequences of Genetic Interactions (Epistasis). Annual review of genomics and human genetics **20**, 433–460 (2019). doi:10.1146/annurev-genom-083118-014857
54. Atwal, G.S., Kinney, J.B.: Learning Quantitative Sequence–Function Relationships from Massively Parallel Experiments. Journal of Statistical Physics **162**(5), 1203–1243 (2016). doi:10.1007/s10955-015-1398-3. 1506.00054
55. Kinney, J.B., Atwal, G.S.: Parametric Inference in the Large Data Limit Using Maximally Informative Models. Neural computation **26**(4), 637–653 (2014). doi:10.1162/neco_a_00568
56. Kinney, J.B., Tkačik, G., Callan, C.G.: Precise physical models of protein–DNA interaction from high-throughput data. Proceedings of the National Academy of Sciences **104**(2), 501–506 (2007). doi:10.1073/pnas.0609908104
57. Sharpee, T., Rust, N.C., Bialek, W.: Analyzing Neural Responses to Natural Signals: Maximally Informative Dimensions. Neural Computation **16**(2), 223–250 (2004). doi:10.1162/089976604322742010
58. Sharpee, T., Sugihara, H., Kurgansky, A., Rebrik, S., Stryker, M., Miller, K.: Adaptive filtering enhances information transmission in visual cortex. Nature **439**(7079), 936–942 (2006). doi:10.1038/nature04519
59. Elemento, O., Slonim, N., Tavazoie, S.: A universal framework for regulatory element discovery across all genomes and data types. Mol Cell **28**(2), 337–350 (2007). doi:10.1016/j.molcel.2007.09.027
60. Hu, Y., Tareen, A., Sheu, Y.-J., Ireland, W.T., Speck, C., Li, H., Joshua-Tor, L., Kinney, J.B., Stillman, B.: Evolution of DNA replication origin specification and gene silencing mechanisms. Nature Communications **11**(1), 5175 (2020). doi:10.1038/s41467-020-18964-x
61. Eraslan, G., Avsec, Z., Gagneur, J., Theis, F.J.: Deep learning: new computational modelling techniques for genomics. Nature Reviews Genetics (2019). doi:10.1038/s41576-019-0122-6
62. Movva, R., Greenside, P., Marinov, G.K., Nair, S., Shrikumar, A., Kundaje, A.: Deciphering regulatory DNA sequences and noncoding genetic variants using neural network models of massively parallel reporter assays. PLoS ONE **14**(6), 0218073 (2019). doi:10.1371/journal.pone.0218073
63. Shrikumar, A., Greenside, P., Kundaje, A.: Learning Important Features Through Propagating Activation Differences. bioRxiv **cs.CV** (2017). 1704.02685
64. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. arXiv (2013). 1312.6034
65. Jones, M.C., Faddy, M.J.: A skew extension of the t-distribution, with applications. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **65**(1), 159–174 (2003). doi:10.1111/1467-9868.00378
66. Olson, C.A., Wu, N.C., Sun, R.: A comprehensive biophysical description of pairwise epistasis throughout an entire protein domain. Current biology : CB **24**(22), 2643–2651 (2014). doi:10.1016/j.cub.2014.09.072
67. Cover, T.M., Thomas, J.A.: Elements of Information Theory, 2nd edn. Wiley. Wiley, ??? (2006)
68. Kinney, J.B.: Mutual information: a universal measure of statistical dependence. Biomedical Computation Review **10**(2), 33 (2014)
69. Kinney, J.B., Atwal, G.S.: Equitability, mutual information, and the maximal information coefficient. Proceedings of the National Academy of Sciences **111**(9), 3354–3359 (2014). doi:10.1073/pnas.1309933111. Wrote. 1301.7745
70. Bialek, W., Nemenman, I., Tishby, N.: Predictability, Complexity, and Learning. Neural Computation **13**(11), 2409–2463 (2001). doi:10.1162/089976601753195969
71. Palmer, S.E., Marre, O., Berry, M.J., Bialek, W.: Predictive information in a sensory population. Proceedings of the National Academy of Sciences **112**(22), 6908–6913 (2015). doi:10.1073/pnas.1506855112
72. Barber, D., Agakov, F.: The IM Algorithm: A variational approach to Information Maximization. Advances in neural information processing systems. (2003)
73. Sjobring, U., Bjorck, L., Kastern, W.: Streptococcal protein G: gene structure and protein binding properties. The Journal of Biological Chemistry **266**(1), 399–405 (1991)
74. Ramsay, J.O.: Monotone Regression Splines in Action. Statistical Science **3**(4), 425–441 (1988). doi:10.1214/ss/1177012761. Read sometime in July 2020
75. Otwinowski, J.: Biophysical Inference of Epistasis and the Effects of Mutations on Protein Stability and Function. Mol Biol Evol **35**(10), 2345–2354 (2018). doi:10.1093/molbev/msy141. Read Preprint. 1802.08744
76. Krawczak, M., Reiss, J., Cooper, D.: The mutational spectrum of single base-pair substitutions in mRNA splice junctions of human genes: Causes and consequences. Human Genetics **90**(1-2), 41–54 (1992). doi:10.1007/bf00210743
77. Srebrow, A., Kornblihtt, A.R.: The connection between splicing and cancer. Journal of cell science **119**(Pt 13), 2635–2641 (2006). doi:10.1242/jcs.03053
78. Ptashne, M., Gann, A.: Genes and Signals. Cold Spring Harbor Laboratory Press. Cold Spring Harbor Laboratory Press, ??? (2002)
79. Lisser, S., Margalit, H.: Compilation of E. coli mRNA promoter sequences. Nucl Acids Res **21**(7), 1507–1516 (1993). doi:10.1093/nar/21.7.1507
80. Steeg, G.V.: Non-Parametric Entropy Estimation Toolbox (NPEET) (2014). `https://www.isi.edu/[INSERT_TILDE]gregv/npeet.html`
81. Vasicek, O.: A Test for Normality Based on Sample Entropy. Journal of the Royal Statistical Society. Series B **38**(1), 54–59 (1976)
82. Kraskov, A., Stögbauer, H., Grassberger, P.: Estimating mutual information. Phys Rev E **69**(6), 066138 (2004). doi:10.1103/physreve.69.066138
83. Gao, S., Steeg, G.V., Galstyan, A.: Efficient Estimation of Mutual Information for Strongly Dependent Variables. arXiv (2014). 1411.2003
84. Efron, B.: Bootstrap Methods: Another Look at the Jackknife. The Annals of Statistics **7**(1), 1–26 (1979). doi:10.1214/aos/1176344552
85. Efron, B., Tibshirani, R.: Bootstrap Methods for Standard Errors,

Confidence Intervals, and Other Measures of Statistical Accuracy.
Statistical Science **1**(1), 54–75 (1986). doi:10.1214/ss/1177013815

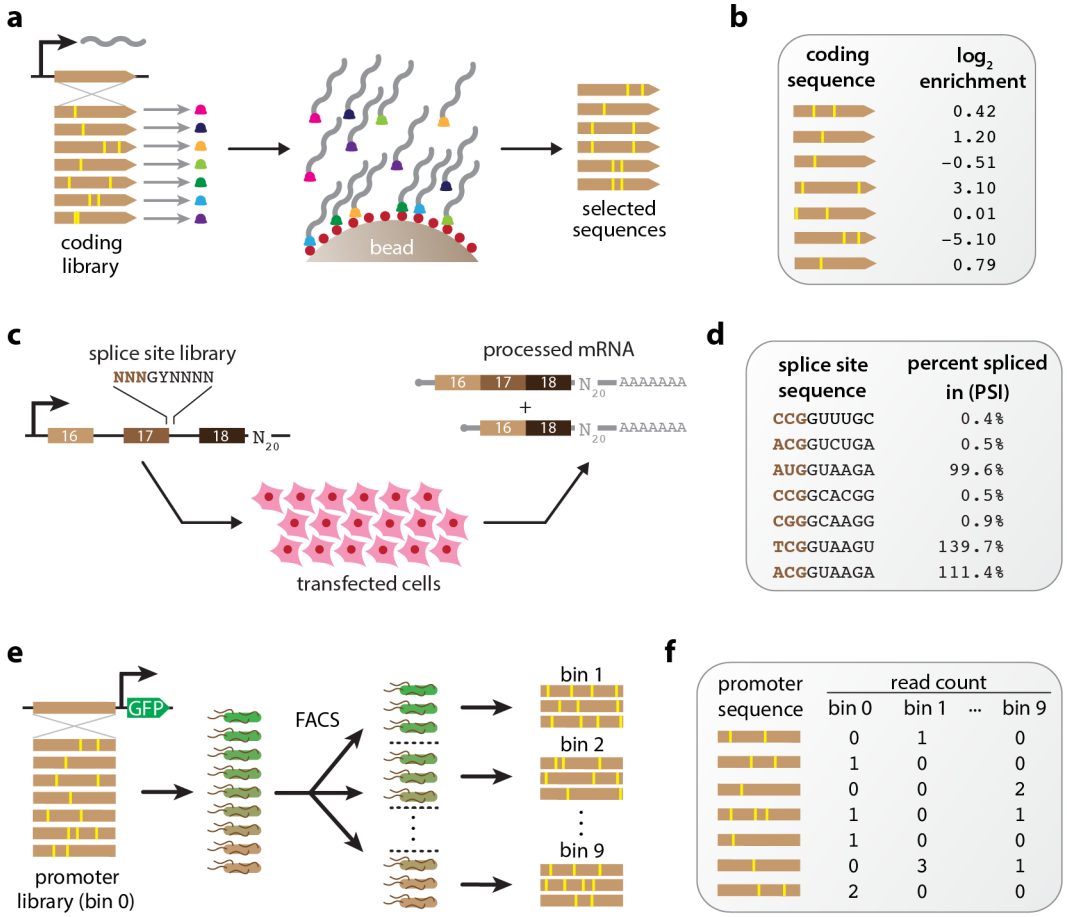

Figure 1 Three different multiplex assays of variant effect (MAVEs). (**a**) The deep mutational scanning (DMS) assay of [66]. A library of variant GB1 proteins (colored bells), comprising all single and pairwise mutations, were covalently linked to their coding mRNAs (gray wavy lines) using mRNA display. Functional GB1 proteins were then enriched using IgG beads, and deep sequencing was used to determine an enrichment ratio for each GB1 variant. (**b**) The resulting DMS dataset consists of variant protein sequences and their corresponding log enrichment values. (**c**) The massively parallel splicing assay (MPSA) of [26]. A library of 3-exon minigenes, containing exons 16, 17, and 18 of the human BRCA2 gene, was constructed. In each minigene variant, the $5'$ splice site of exon 17 was replaced by a randomized 9-nt sequence, and a random 20-nt barcode was inserted into the $3'$ UTR. This library was transfected into HeLa cells, followed by RNA extraction and reverse transcription. Deep sequencing was then used to count barcodes from (i) mRNA that included exon 17 and (ii) total mRNA. (**d**) The resulting MPSA dataset comprises variant $5'$ splice site sequences and corresponding percent-spliced-in (PSI) values computed from the ratio of these barcode counts. Note that some PSI values are greater than 100% due to experimental noise. (**e**) The sort-seq MPRA of [17]. A plasmid library was generated in which randomly mutagenized versions of the *Escherichia coli lac* promoter drove the expression of Green Fluorescent Protein (GFP). Cells carrying these plasmids were then sorted using fluorescence-activated cell sorting (FACS). The variant promoters in each bin of sorted cells were then sequenced. (**f**) The resulting data comprises a list of variant promoter sequences, as well as a matrix of counts for each sequence in each FACS bin.

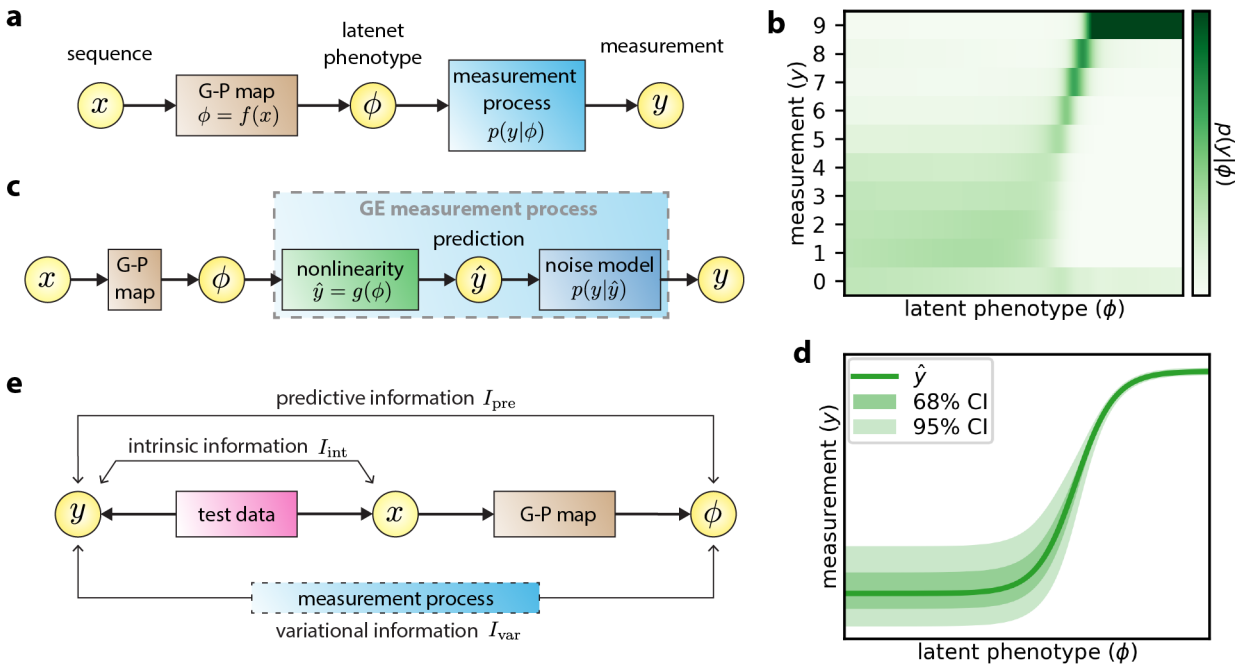

Figure 2 MAVE-NN's quantitative modeling strategy. (**a**) A deterministic genotype-phenotype (G-P) map $f(x)$ is assumed to map each input sequence x to a latent phenotype ϕ. This latent phenotype ϕ is assumed to govern the corresponding measurement y through a stochastic measurement process $p(y|\phi)$. During training, MAVE-NN infers quantitative models for both $f(x)$ and $p(y|\phi)$. (**b**) In measurement process agnostic (MPA) regression, MAVE-NN directly learns a model for $p(y|\phi)$. This panel illustrates one such measurement process, which was inferred from the sort-seq MPRA data of [17]. Note that, at present, MAVE-NN only supports MPA regression for datasets in which y is discrete. (**c**) In global epistasis (GE) regression, MAVE-NN assumes that each measurement y is a noisy readout of a prediction $\hat{y}$, the value of which is a deterministic function $g(\phi)$ of the latent phenotype ϕ. During training, MAVE-NN learns models for the nonlinear function $g(\phi)$ as well as the noise model $p(y|\hat{y})$. (**d**) Shown is a measurement process for the GB1 data of [66] inferred using GE regression. This is illustrated using the predictions $\hat{y}$ of the nonlinearity, as well as the 68% and 95% central intervals (CIs) of the noise model $p(y|\phi)$. (**e**) Three information-theoretic quantities are used to assess model performance in the MAVE-NN framework. Intrinsic information, I_{int}, represents the mutual information between sequences x and measurements y, and is inherent to a dataset. Predictive information, I_{pre}, quantifies the mutual information between measurements and the latent phenotype values ϕ assigned by a model. Variational information, I_{var}, is a linear transformation of log likelihood and thus depends on a model's measurement process (unlike I_{pre}). It does not quantify mutual information per se, but it does provide a lower bound on I_{pre} and serves as a useful metric during training. In fact, the inequality $I_{\text{int}} \geq I_{\text{pre}} \geq I_{\text{var}}$ always holds on test data (modulo finite data uncertainties), with $I_{\text{int}} = I_{\text{pre}}$ when the G-P map is correct, and $I_{\text{pre}} = I_{\text{var}}$ when the measurement process is also correct.

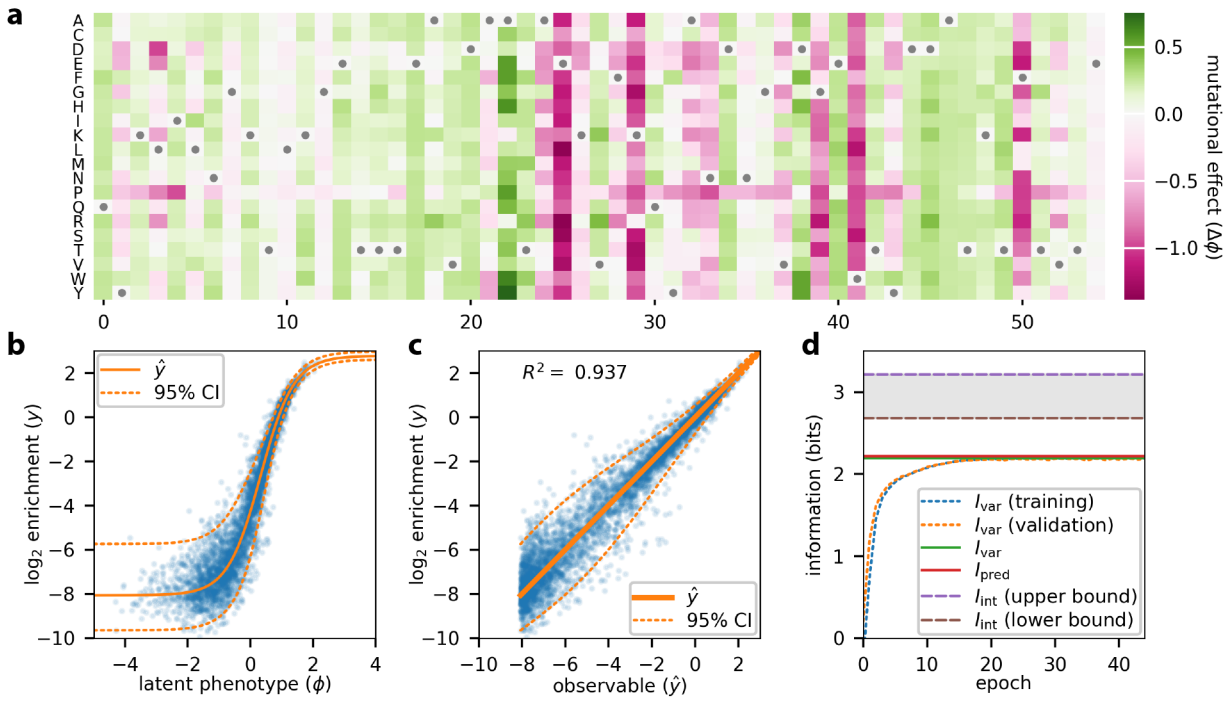

Figure 3 Analysis of DMS data using GE regression. MAVE-NN was used to infer a model comprising an additive G-P map, a monotonic GE nonlinearity, and a heteroskedastic skewed-t noise model. Parameters were inferred using the 530,737 pairwise variants reported in [66], with a 60:20:20 split into training, validation, and test sequences. (**a**) Heat map illustrating the inferred G-P map. Colors indicate the additive effects for each amino acid mutation at each of the 55 positions within protein GB1. The effects of all wild type residues (gray dots) are set to zero by our choice of gauge. (**b**) Measurements y verses latent phenotype values ϕ for 5,000 randomly selected test set sequences. Also shown is the GE nonlinearity $\hat{y} = g(\phi)$ (solid line) and the corresponding 95% central interval (CI) of the inferred noise model (dashed lines). (**c**) Measurements y versus the predictions $\hat{y}$ computed for these same sequences. (**d**) Information metrics. Dotted lines show the variational information (I_{var}) values on the training and validation sets computed during model inference. Solid lines show the predictive information (I_{pre}) and I_{var} values of the final inferred model on test data. Dashed lines provide upper and lower bounds for the data's intrinsic information (I_{int}; see Methods), the allowed values for which are shaded in gray. The uncertainties in these information estimates are roughly the width of the plotted lines and are not shown.

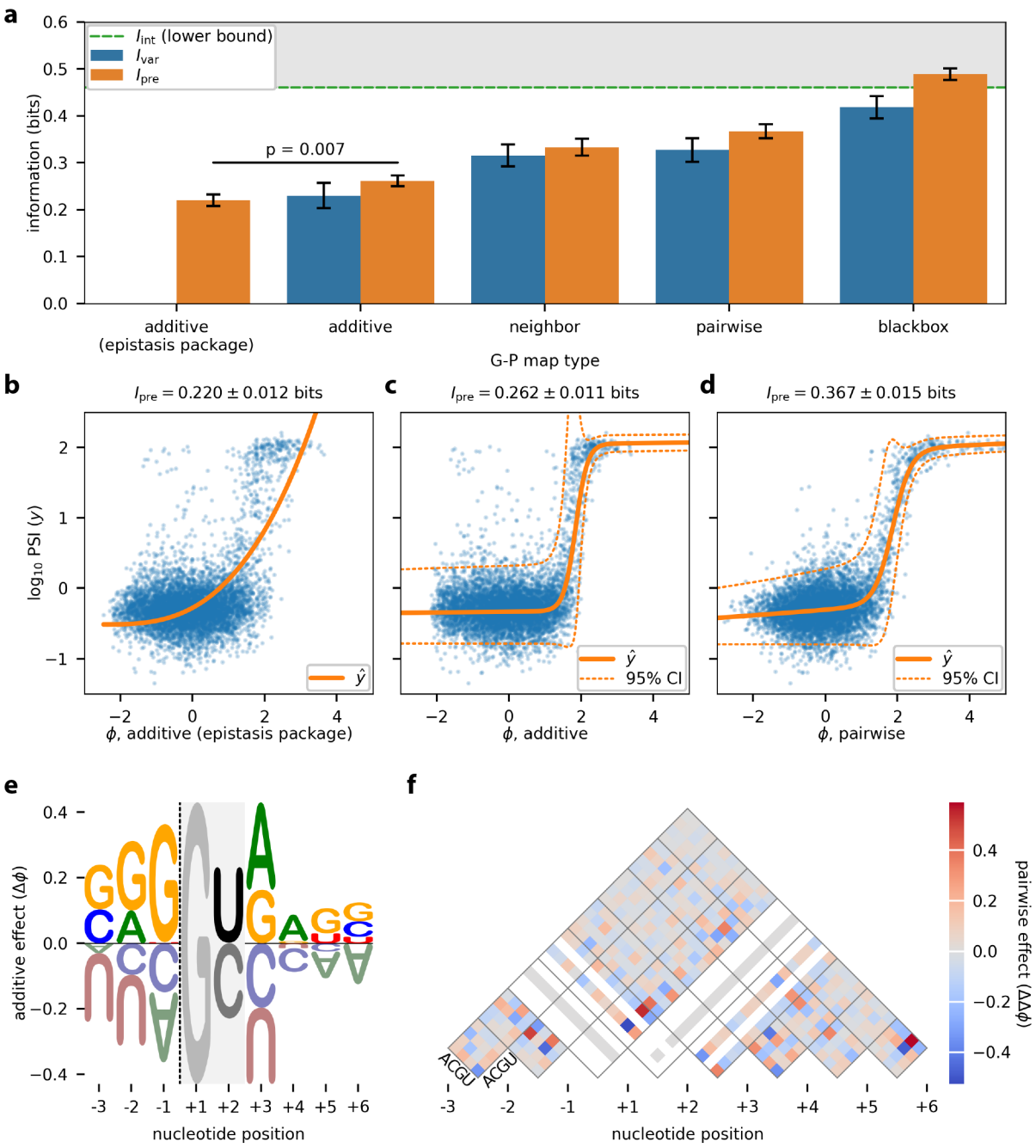

Figure 4 GE regression analysis of data from a massively parallel splicing assay (MPSA) [26]. MAVE-NN was used to infer GE regression models with four different types of G-P maps: additive, neighbor, pairwise, and blackbox. For comparison, we also trained an additive GE regression model using the epistasis package of Sailer and Harms [50]. To train and evaluate these models, we used $\log_{10}$ percent-spliced-in (PSI) values (y) measured for 30,483 variant $5'$ splice sites, with the data split 60:20:20 into training, validation, and test sets. (**a**) Performance of trained models as quantified by variational information (I_{var}) and predictive information (I_{pre}). Error bars indicate standard errors. The dashed green line indicates a lower bound on intrinsic information (I_{int}), the allowable values of which are indicated in gray. The p-value results from a two sample Z-test. Note that an I_{var} value for the additive (epistasis package) model was not computed because the epistasis package does not infer a corresponding measurement process. (**b-d**) GE plots of measurements y versus latent phenotypes ϕ for three selected models: the additive (epistasis package) model with power law nonlinearity (**b**), the additive model inferred by MAVE-NN (**c**), and the pairwise model inferred by MAVE-NN (**d**). (**e**) Sequence logo illustrating the additive effects component of the pairwise G-P map. The dashed line indicates the exon/intron boundary. The G at $+1$ serves as a placeholder because no other bases were observed at this position, and only values for U and C are shown at $+2$ because only these bases were observed. (**f**) The pairwise effects component of the pairwise G-P map. Diagonals corresponding to unobserved bases are colored in white. Note that all I_{pre} and I_{var} values shown here, as well as the scatter plots in panels **b-d**, reflect model performance on held-out test data.

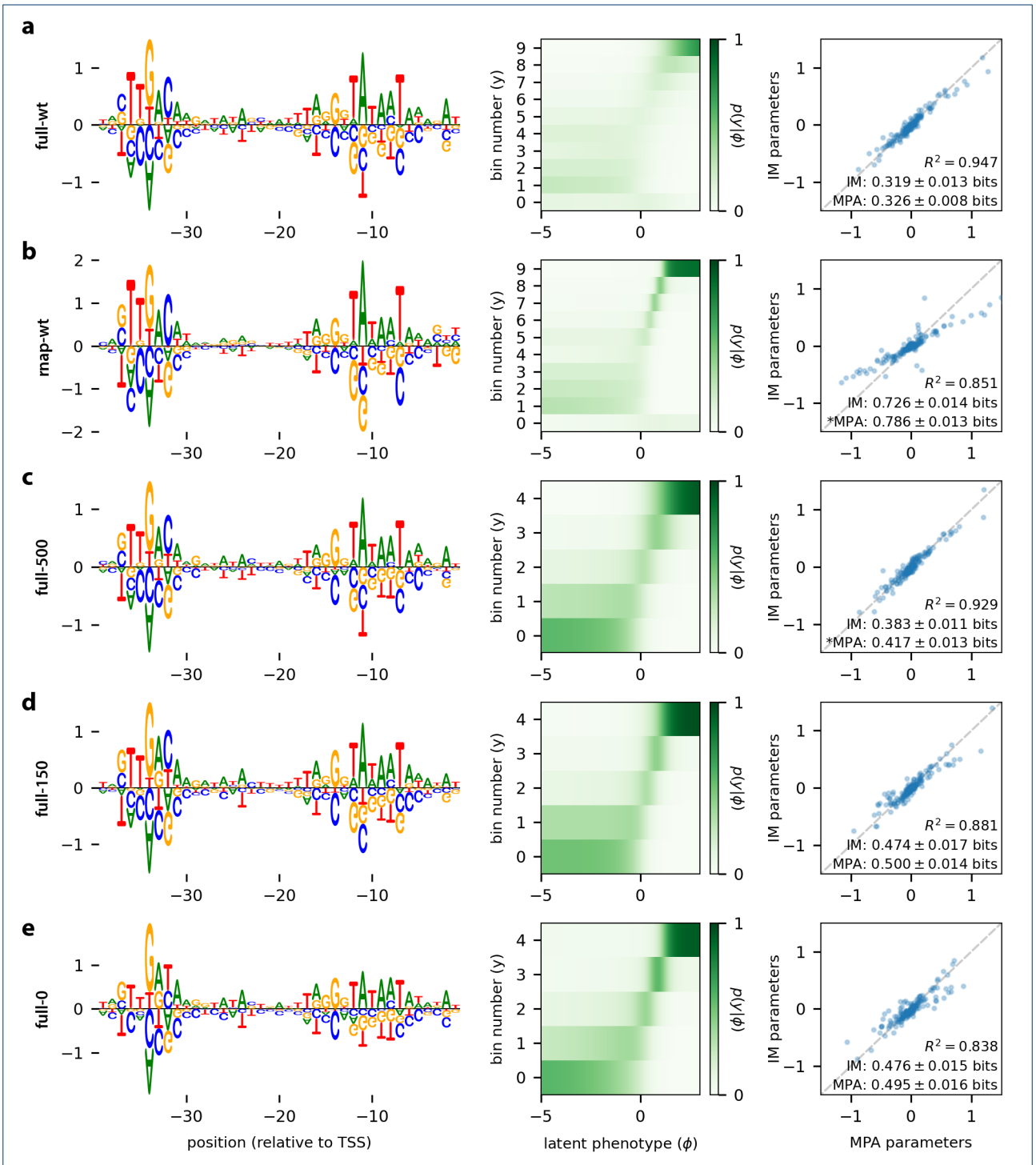

Figure 5 Analysis of sort-seq MPRA data using MPA regression. MAVE-NN was used to infer additive G-P maps representing the activity of *E. coli* σ^{70} RNA polymerase (RNAP). These were trained on five different sort-seq datasets reported in [17]: (**a**) full-wt, (**b**) rnap-wt, (**c**) full-500, (**d**) full-150, (**e**) full-0. In each row, the parameters θ of the inferred G-P map is shown as a sequence logo (left), while the corresponding measurement process $p(y|\phi)$ is illustrated as a heatmap (center). The left panel shows a scatter plot comparing θ for the G-P map inferred using MPA regression to that inferred by [17] using information maximization (IM). The squared correlation between parameter values are shown, as are the I_{pre} values of each model. Models with significantly higher information ($p < 0.05$; two sample Z-test) are indicated by an asterisk.