SUPPORTING INFORMATION FOR:

Reconfiguring primase DNA-recognition sequences by using a data-driven approach

Adam Soffer[1,2,3], Sarah A. Eisdorfer[1], Morya Ifrach[1], Stefan Ilic[1], Ariel Afek[1], Hallel Schussheim[1], Dan Vilenchik[2,3], Barak Akabayov[1,2*]

[1] Department of Chemistry, [2] Data Science Research Center, and [3] School of Computer and Electrical Engineering, Ben-Gurion University of the Negev, Beer-Sheva, Israel.

* To whom correspondence should be addressed. Tel: +972-8-6472716; Email: akabayov@bgu.ac.il

ORCID: https://orcid.org/0000-0002-3882-2742

METHODS

Materials. All chemical reagents were of molecular biology grade and were obtained from Sigma. ATP and CTP were purchased from Roche Molecular Biochemicals.

Protein overexpression and purification. Full length gene 4 protein (gp4, residues 1-566, 63 kDa) was overexpressed and purified as previously described (1). The T7 primase domain (residues 1–271, 27 kDa) was overexpressed and purified as previously described (2).

Design of the DNA library. The analysis was based on previously collected data (3,4), specifically, on 25,220 DNA sequences that include the T7 DNA-primase recognition sequence (5'-GTC-3'). The general pattern of each sequence was 5'-$(N)_{17}$-GTC-$(N)_{16}$-GTCTTGATTCGCTTGACGCTGCTG-3', where $(N)_{17}$ and $(N)_{16}$ represent the variable regions flanking the GTC recognition site. The above data Ωset contained accurate binding scores for T7 primase to each DNA sequence, obtained by PBMs as described previously (4). Data acquisition was performed using a GenePix 4400A scanner (Molecular Devices), and data was analyzed using custom scripts to obtain fluorescence intensities for all sequences represented on the array.

Data preprocessing. Each PBM consisted of 5,076 unique sequences and 25,220 samples, 6 repetitions per sequence, and overall 151,320 samples (instances). All scripts were written in Python (Python Software Foundation, version 3.7, http://www.python.org), Scikit learn (5), and the software PyCharm (community edition, https://www.jetbrains.com/pycharm/). The source code for the machine learning algorithms is available in the Github repository (https://github.com/csbarak/T7pdrs). This git repository also contains the data used for the analysis.

By extracting the coefficient of variation (6) for scores associated with each sequence (6 repetitions), we observed that the stronger the score, the more stable the coefficient of variation (Supplementary Figure S1). Finally, each sequence's score was determined as its median score. For the stability evaluation, it was necessary to account for the different binding score ranges; thus, to eliminate the different scales of the standard deviation, we evaluated the binding score stability of each sequence by using the coefficient of determination (COD, Eq. 1):

$$COD(x) = \frac{\sigma(x)}{\mu(x)} \quad (1)$$

where x is a set of binding score repeats for a specific sequence; σ is the standard deviation of that sequence; and μ is the mean value of x.

Method for sequence-based feature extraction. We tried out linear, quadratic and root weighting of the K-mers according to their distance from the GTC; e.g., while the 3-mer 'ACA' appears twice in the sequence 'ACATGTCACAT', the weighted linear count of 'ACA' would multiply its distance plus 1 from the kernel (GTC). However, this approach did not improve the performance of the model. While proper usage of the mer's location might lead to different results, using advanced algorithms to produce a more complex connection between features would limit our work's explainability and further exploration of the mer's effect. We therefore used simple K-mer counts and normalized by the length of the sequence to increase the generalization and prevent bias.

Principal component analysis. PCA is commonly used to reduce dimensions of datasets by de-correlating the features and extracting the linear combinations that hold the greatest variance. Thus, non-informative features are dropped, and the remaining features consist of highly variant linear combinations (principal components) of the original features. We used PCA on overlapping K-mer count instances so as to visualize the projected distribution of binding scores upon the three most significant principal components. Features were obtained by counting every combination of dimers, trimers, and tetramers in the DNA sequence (K-mer, Supplementary Figure S2). Different K values were used for the K-mer feature extraction, and all experiments resulted in a clear 5-cluster construct for the entire dataset. To compare data between clusters, we applied MinMax normalization (Eq. 2) and colored each instance according to its relative strength.

$$y_i' = \frac{y_i - \min(y)}{\max(y) - \min(y)} \quad (2)$$

where y = binding scores of the entire dataset, y_i = the ith binding score.

Conversion of the categorical DNA variables. The DNA data was converted to an array of integers by OHE, a process in which each nucleotide is represented by the following scheme: (A=[1000], C=[0100], G=[0010], T=[0001]). The N × 4 matrix represents every DNA oligonucleotide, and is used as input for both the Kmeans model and the WD-based hierarchical clustering model (7).

Kmeans. In the initial step of Kmeans, the distances of the sequence vectors in the training set from randomly located centroids are measured, with the number of centers (K) being considered as a hyperparameter. Then, the distance of every sequence from the centroid is computed using the Euclidean distance ($d(x) = \min_{j=0,1,\ldots,K} \| x - \mu_j \|_{l2}$). For the optimization step, each centroid's position (μ_j) is moved to its own cluster's geometric mean. This process is repeated until a stop condition is met, which is usually determined by an improvement in the loss function. The loss function of the i^{th} iteration is the sum of the distances between all instances and their matching centroids (Eq. 3):

$$L_i(X, \mu_i) = \sum_{x \in X} d(x) \quad (3)$$

where i is the iteration number, X denotes the entire data matrix, x represents an OHE vector, and μ_i represents the set of centroids at the beginning of the i^{th} iteration.
An optimized model is obtained when the difference in the value of the loss function between consecutive iterations is small enough (typically 10^{-4}) or the maximal number of iterations has been reached.

Hierarchical clustering. Ward's criterion is used to determine which clusters should be merged by creating new data partitions in such a way that the sum of cluster variances of the newly offered partitions is kept low; in our case, it amounts to the smallest number of nucleotide changes between same-cluster sequences. Since the sum of the squared errors is minimized when each "word" acts as its own cluster, the common way to choose the number of clusters K is to choose the K that maximizes the WD gap. Using this method, we can extract both K and the evolutionary stages of each cluster. WD calculates the similarity of two clusters (C_a, C_b) as the normalized distance of their corresponding cluster means (μ_a, μ_b, Eq. 4):

$$WD(C_a, C_b) = \frac{|\mu_a - \mu_b|_{l2}^2}{|C_a||C_b|} \quad (4)$$

The first step of the method initiates a cluster for each instance, and the second seeks the two most similar clusters in terms of WD. When found, these two clusters are united, and the second step is repeated until only one cluster remains.

Supervised learning – linear regression with L1 regularization (Lasso). The Lasso algorithm performs liner regression under L1 regularization. Its output is a closed form equation that is generated under the constraint of having the smallest number of variables as possible. The algorithm complies with this constraint by applying a penalty for each variable taken into account in the closed form equation. Simple linear regression uses a weighted combination of features to generate a prediction based on (Eq. 5):

$$Y = \sum_{i=1,2,\ldots,p} w_i x_i + b \quad (5)$$

where x_i is the i^{th} feature chosen from p features, while w_i and b are the learned weights (usually found by minimizing the mean square error over the training set) and the learned bias, respectively. While a simple linear model uses the entire set of features, Lasso applies a loss function on the number of features. Moreover, compared to L2 regularization, L1 regularization facilitates the zeroing out of features rather than minimizing their weights, leading to the selection of a smart subset of features. Using Lasso on our data required two preprocessing stages; the first was extracting K-mer counts for obtaining a simple and general solution, and the second was applying a square root on the binding scores to better match their values for linear regression. The MinMax-wise normalized scores yielded a cross-validated result with a mean absolute error (MAE) value of 0.10, calculated using (Eq. 6):

$$\text{MAE } E(X) = \sum p_{x_i} * x_i \quad (6)$$

where x_i is the MAE of bin i of the bins obtained by Kmeans, and p_{x_i} is the percentage of samples in that bin out of the entire data set.

We evaluated the results with MAE, and obtained the expected error in terms of a weighted MAE, where the weights refer to the percentage of clustered sequences (Eq. 7).

$$WMAE_{primo} = \sum_{i=0}^{4} \frac{|C_i|}{|dataset|} MAE_{C_i} \quad (7)$$

where C_i is the *i*th cluster, $|C_i|$ is the number of sequences belonging to the *i*th cluster, $|dataset|$ is the size of the entire dataset and MAE_{C_i} is the mean absolute error of the *i*th cluster.

Our main goal was to develop a predictive model with as small an error as possible, while maintaining model explainability and simplicity. Examining the results of different regression models (Supplementary Table S1), we see that the smallest error was achieved using XGBoost, yet the difference between the errors of XGBoost and those of Lasso is about 0.5% MAE. In contrast to the decision-tree-based XGBoost, Lasso generates a closed predictive equation (i.e., $score = \alpha_0 + \alpha_1 MER_1 + \alpha_2 MER_2$ ...), and combined with Lasso's L1 regularization, it constrains the number of features and the coefficients needed for the prediction. In addition, in contrast to support-vector-machine (SVM)-based models, Lasso enables limiting the coefficients to positive values, which could lead to a meaningful K-mer addition approach. Lastly, with Lasso the bias can be neutralized, meaning that the prediction is dependent solely on the K-mer count. Increasing the bias further enables a decrease in the variance and therefore a precise prediction.

In summary, in this study, we chose to use Lasso, since it provides good performance and a closed predictive expression that is short and (intentionally) consists of non-negative coefficients. Other regression models also generated an expected error that was less than 10% MAE (Supplementary Table S1), meaning that the data collection and preprocessing techniques were highly informative regarding the researched binding score.

Oligoribonucleotide synthesis assay. Synthesis of oligoribonucleotides by DNA primase was performed as described previously (4). The reaction mixture contained 5 µM DNA sequences generated by our machine-learning prediction algorithms described above, 1 mM ATP, 1 mM [α-^{32}P]ATP, and T7 primase in a buffer containing 40 mM Tris-HCl, pH 7.5, 10 mM $MgCl_2$, 10 mM DTT, and 50 mM potassium glutamate. After incubation at room temperature for 10 min, the reaction was terminated by adding an equal volume of sequencing buffer containing 98% formamide, 0.1% bromophenol blue, and 20 mM EDTA. The samples were loaded onto 25% polyacrylamide sequencing gel containing 7 M urea and visualized using autoradiography.

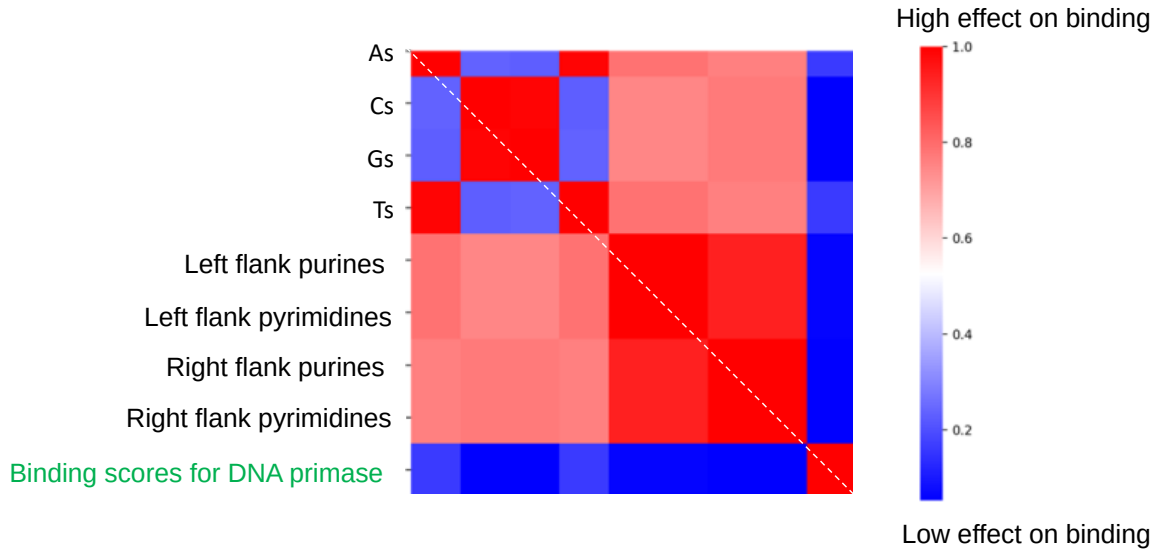

Figure S1. Diagonal correlation matrix of features representing oligonucleotide composition. This correlation presents the insignificant effect of hand-crafted features on the binding score of T7 primase. These features were calculated from the sequence of oligonucleotides in the DNA-protein microarray (PBM). Binding score of T7 primase was determined by PBM.

K-mers for GTCCC

k	*K-mers*
1	G, T, C
2	GT, TC, CC
3	GTC, TCC, CCC

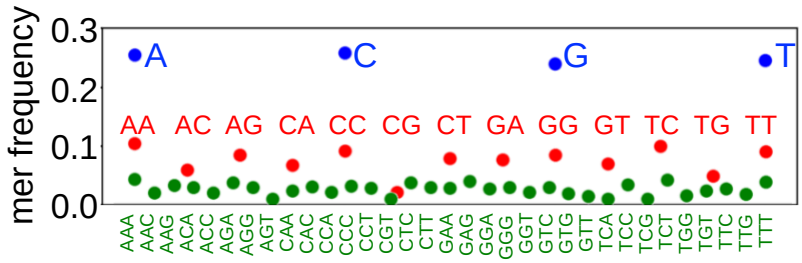

Figure S2. Illustration of K-mer features. Left: The term K-mer refers to all of a sequence's subsequences of length *k*, such that the sequence AGAT would have four monomers (A, G, A, and T), three 2-mers (AG, GA, AT), two 3-mers (AGA and GAT). Right: Higher *k* number is characterized with low frequency of occurrences on the genome.

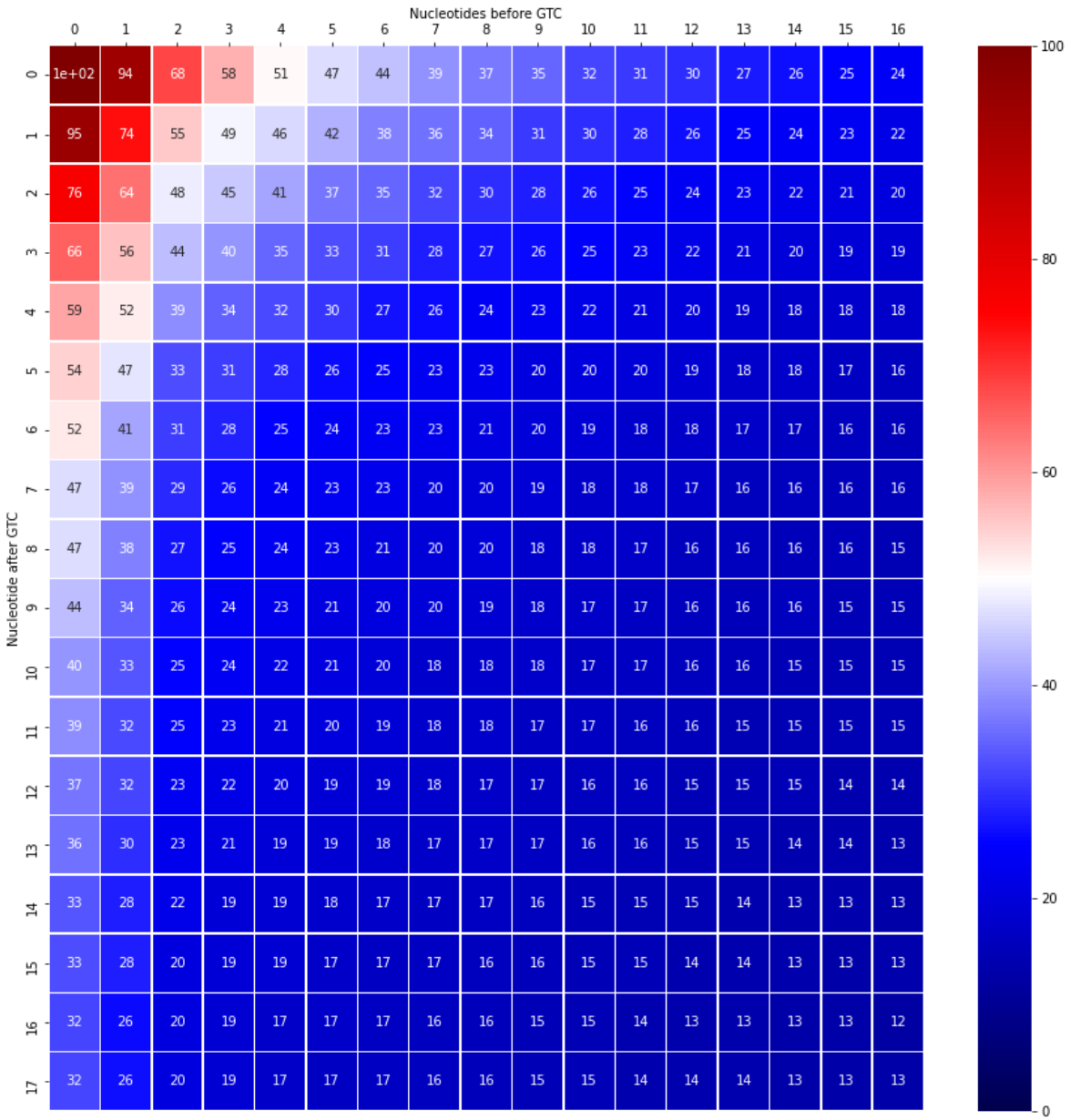

Figure S3. Effect of nucleotide position outside the ‘GTC’ on primase binding. A gradient boosting machine (GBM) was trained on a different sliding "window" of nucleotide positions before and after the labeled GTC-containing sequences. The results are represented by the measured errors (MAE score) between paired nucleotides on the test data.

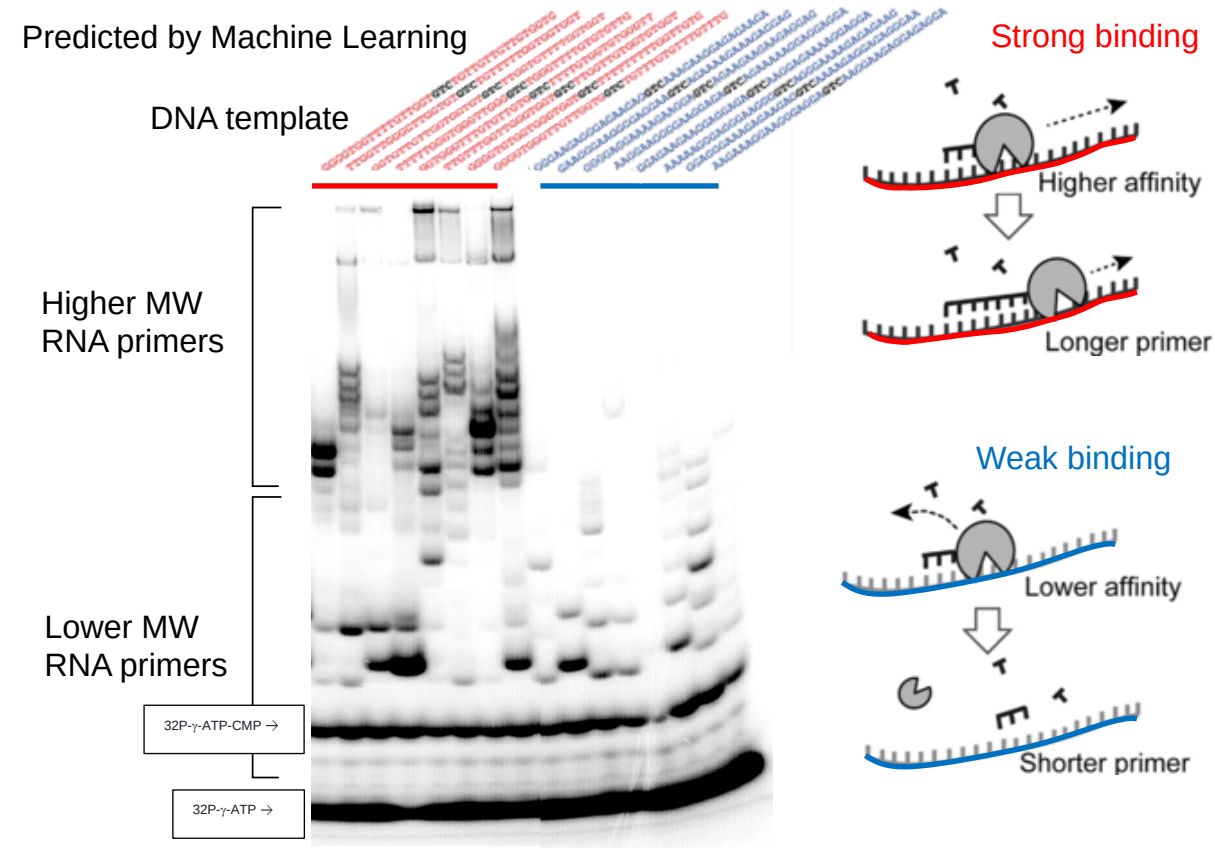

Figure S4. Template-directed RNA primer synthesis catalyzed by the T7 DNA primase. Oligonucleotide synthesis by the T7 DNA primase. The reactions contained oligonucleotides with the primase recognition sequence as indicated, and 32P-γ-ATP, ATP, CTP, UTP, and GTP in the standard reaction mixture. After incubation, the radioactive products were analyzed by electrophoresis through a 25% polyacrylamide gel containing 7 M urea, and visualized using autoradiography.

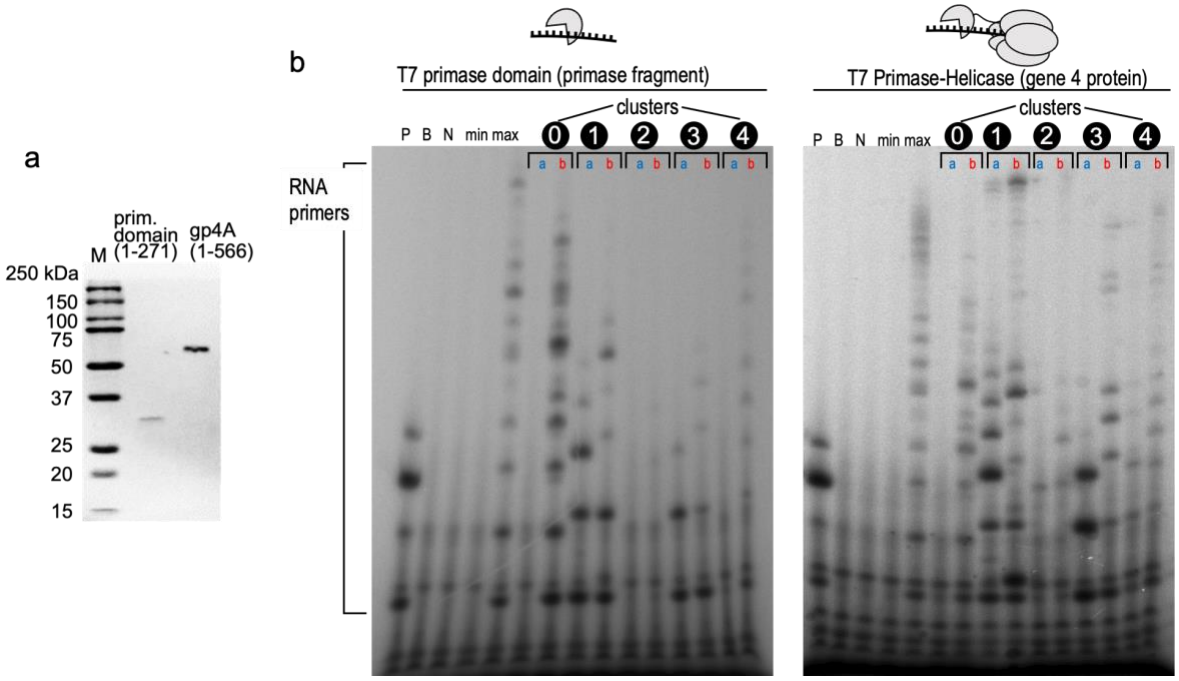

Figure S5. Template-directed RNA primer synthesis catalyzed by the T7 primase domain and full length T7 gp4. (**a**) SDS-PAGE analysis of pure T7 primase domain (27 kDa, residues 1-271) and full-length T7 gp4 helicase primase (63kDa, residues 1-566). (**b**) Oligonucleotide synthesis; comparison between T7 gp4 proteins: primase domain (left) and full length helicase-primase gp4 (right). The reactions contained oligonucleotides with the primase recognition sequence as indicated (Supplementary Table S3), and [^{32}P]γ-ATP, ATP, CTP, UTP, and GTP in the standard reaction mixture; protein was added to a final concentration of 300 nM. After incubation, the radioactive products were analyzed by electrophoresis through a 25% polyacrylamide gel containing 7 M urea, and visualized using autoradiography. Longer RNA primers were formed on DNA templates that were predicted to have higher binding affinity to the primase.

a

Sample			SEQUENCE
P			GG **GTC** AAAAAAAAAA
Min.			GACGAAGACGACGAAGA **GTC** CGAGGAAGCAGACGAA
Max			TGTGGTGGGTGTGTGTG **GTC** TTTTGTTTGTGGTGGT
	BINDING (%)	PREDICTION	SEQUENCE
0-10	10	0.588	TGGGTTGTGTGGATTTT **GTC** GGTGTGGGTGTTTGTG
0-50	50	0.643	TGGGTTTGTGGGGTTGT **GTC** GGGTTGTTGTTTGGTG
0-90	90	0.825	TTGTGGTGTGGTGTTTG **GTC** TGGTGTGTCTTTTGGG
4-10	10	0.107	AGAGAAGAGAGAAGAGA **GTC** GTAGAGAGAGAGAGAG
4-50	50	0.229	AGGAAAGAAGAGGAGAG **GTC** GGAGTAAAGGAAGAAG
4-90	90	0.368	GGGGAAAGAGGGAGGAG **GTC** AGGAAAAGAAAGGGAA

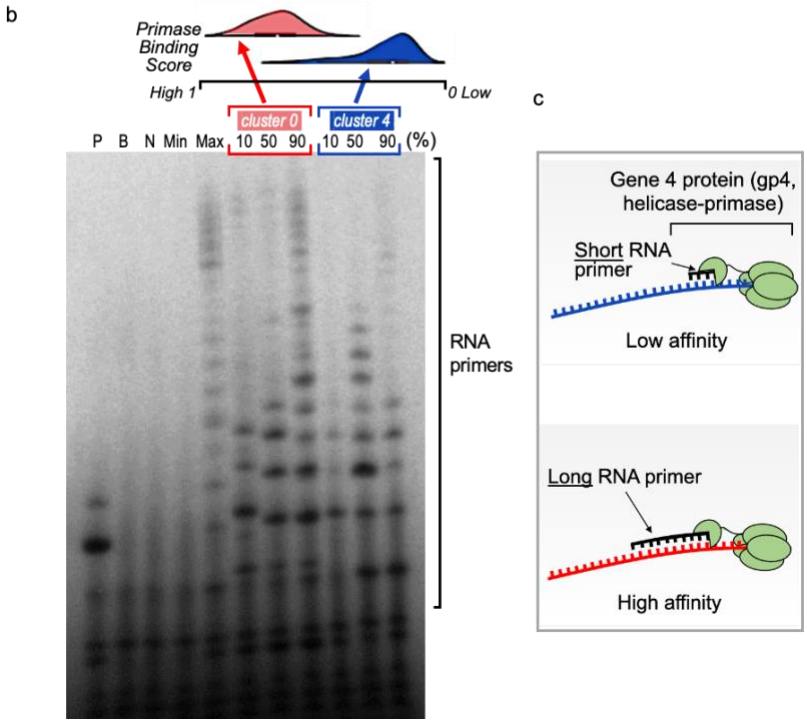

Figure S6. RNA primer synthesis by the full-length gene 4 helicase-primase on selected DNA templates from different clusters obtained by unsupervised learning. The experiment was performed as described in Figure 5 with one exception: the protein used for primase activity was the full length gp4 (helicase-primase) from bacteriophage T7. The reactions contained selected oligonucleotide sequences from clusters #0 and #4 obtained by unsupervised analysis (Figure 3). (**a**) Table summarizes the DNA template sequences used for the biochemical validation and their corresponded values. Three DNA sequences from each of the Kmeans clusters #0 and #4 that predicted the 10th, 50th, and 90th percentile binding scores were selected in each cluster. (**b**) Top: Distribution of binding values for the two clusters. Note that cluster #0 shows stronger primase binding values, on average, than cluster #4. Bottom: Oligoribonucleotide synthesis by T7 primase. The standard reaction mixture contained oligonucleotides with the primase recognition sequence, a control oligonucleotide 5'-GGGTCA10-3', and [γ-^{32}P]ATP, CTP, GTP, and UTP. After incubation, the radioactive products were analyzed by electrophoresis on a 25% polyacrylamide gel containing 7 M urea, and visualized using autoradiography. (**c**) illustration of the effect of primase-DNA binding affinity on the size of RNA primers. Note that the pattern of primase activity of the full length T7 gp4 remained identical to that obtained by T7 primase domain of bacteriophage T7 (Figure 5).

Table S1. Comparative analysis of different regressors, where K = 3, after clustering

Model	KNN	RBF-SVM	Linear-SVM	RF	XGBOOST	LASSO
MAE	0.096	0.093	0.091	0.094	0.088	0.093

Table S2. Test data

Sequence	Empirical results	Predicted binding
AAAAAGGGAGGGAAGGGGTCAGGGAAAAGAGAGAAG	**2016**	**0.358**
AAGAAAGGAAGGGAGGAGTCAAGGAAGAGGAGAGGA	**1956.5**	**0.232**
AAGGAAGGGGAAGGAGAGTCAGAAAAAGGAGGAGGA	**2570**	**0.295**
GAAGGGAAGGGGAGGAAGTCAGAAAAGAAAGAGGAG	**2365**	**0.293**
GGAGAAGAAGGAGGAGAGTCAAGGAGAAAAGGAGGA	**1650**	**0.244**
GGAGGGAAAGAGAAGAGGTCAAAAGAGGAGAGGGAA	**1904.5**	**0.317**
GGGAAGAGGGAGAAGAGGTCAAAGAAGGAGAGAAGA	**1859**	**0.292**
GGGGAGGAAAAGAAGGAGTCAGAAGAAGAAGAGGAG	**1895**	**0.283**
GGGGTGGGTTGTTGGTGGTCTGTTTGTGTTTGTTTG	**48919.5**	**0.785**
GGGGTGGTTTTGTTGGTGTCTGTTGTTGTTGTGGTG	**49672**	**0.807**
GGGGTGTGGTGGGTGGTGTCTTTTTTTTTGGTTGTG	**48331.5**	**0.815**
GGTGGGTTTGTGTTGTGGTCTTTTGTGGTGTGGGTT	**45403**	**0.797**
GGTGTTGTTGGTGGTGTGTCTTGGTGTTTTGGTGGT	**41314.5**	**0.822**
TTGGTTGGGGTTGGTGTGTCTGTTTTTGGTGGTGGT	**43744**	**0.746**
TTGTTTGGTTGGGTGGTGTCTTGGTTGTGGTGTGGT	**41522**	**0.791**
TTTTTGGGTGGGTTGGGGTCTGGGTTTTGTGTGTTG	**51845.5**	**0.659**

Table S3. DNA sequences from a generative algorithm, their binding prediction to primase, and their free energy prediction of folding.

	Cluster	Sequence	Binding prediction[1]	Secondary Structure prediction[2]	Oligo Evaluator™ [3]	Benchling ΔG[4] (kcal/mol)	NuPack ΔG[5] (kcal/mol)
P		GG**GTC**AAAAAAAAAA		unfolded	none	0.00	0.00
min		GACGAAGACGACGAAGA**GTC** CGAGGAAGCAGACGAA	0	**folded**	**weak**	**-3.32**	**-3.23**
max		TGTGGTGGGTGTGTGTG**GTC** TTTTGTTTGTGGTGGT	1	unfolded	none	0.00	0.00
0a	0	TTTTTTTTTTTGGGGGG**GTC** GGGTTTGGGGTGGGGT	0.670	unfolded	none	0.00	0.00
0b	0	TTGTGTGGGGTCTTGTG**GTC** TTTGTGTGTTGGGTGT	0.806	unfolded	none	0.00	0.00
1a	1	CCTCCCTTTTTTTTTTT**GTC** CTCTCCTCCTTTCCCC	0.316	unfolded	none	0.00	0.00

1b	1	TTCCACCACTCCATTCT**GTC** AACGTATTCTTCACCC	0.509	unfolded	none	0.00	0.00
2a	2	CTTCGAAGCAACCAAAG**GTC** GCAAGTTGAATAAGAC	0.468	**semi-folded**	**moderate**	**-2.72**	**-2.67**
2b	2	CGATGCTGTTCCGTTTG**GTC** AACTAAAGACCATGAT	0.502	**folded**	**strong**	**-5.10**	**-3.66**
3a	3	CCCAAAAAACCCCAAAA**GTC** TCCACCAACCCCAAAA	0.472	unfolded	none	0.00	0.00
3b	3	CCAAAACAAACCCAACA**GTC** ACCACCCCACCCTAAA	0.630	unfolded	none	0.00	0.00
4a	4	AAGGAAGGAGAAGAGAA**GTC** GAGGGAGAGCGAGGAA	0.137	**semi-folded**	**weak**	**-1.73**	**-0.73**
4b	4	GGAGAAGAGGAGGAGGT**GTC** AGAAGAAAAAGAAAGG	0.242	unfolded	none	0.00	-0.13
0-10	0	TGGGTTGTGTGGATTTT**GTC** GGTGTGGGTGTTTGTG	0.588	unfolded	none	-0.01	0.00
0-50	0	TGGGTTTGTGGGGTTGT**GTC** GGGTTGTTGTTTGGTG	0.643	unfolded	none	0.00	0.00
0-90	0	TTGTGGTGTGGTGTTTG**GTC** TGGTGTGTCTTTTGGG	0.825	unfolded	none	0.00	0.00
4-10	4	AGAGAAGAGAGAAGAGA**GTC** GTAGAGAGAGAGAGAG	0.107	unfolded	none	-0.01	-0.21
4-50	4	AGGAAAGAAGAGGAGAG**GTC** GGAGTAAAGGAAGAAG	0.229	unfolded	none	0.00	0.00
4-90	4	GGGGAAAGAGGGAGGAG**GTC** AGGAAAAGAAAGGGAA	0.368	unfolded	none	0.00	-0.32

[1]Normalized scores that predict the outcome of T7 primase-DNA binding for a given DNA sequence. Strong binding values are colored red, and weak binding are colored blue. Machine learning analysis allowed quantitative prediction scoring after several steps of algorithm development (including preparation of benchmark data, clustering DNA sequences, training a regression model, predicting scores of new sequences, and biochemical validation). [2]Among the web applications that exist to calculate minimum free energies of probable DNA secondary structures and to predict the possibility of their formation under different conditions, we used: [3]Sigma-Aldrich cloud-based informatics platform (OligoEvaluator™, http://www.oligoevaluator.com/LoginServlet). [4]A web-based oligonucleotide analysis tool that predicts secondary structure formation and minimum free energies (Benchling, https://benchling.com), and [5]a web application for the analysis and design of nucleic acid structures (NUPACK (8)), https://piercelab-caltech.github.io/nupack-docs/).

REFERENCES

1. Lee, S.J. and Richardson, C.C. (2001) Essential lysine residues in the RNA polymerase domain of the gene 4 primase-helicase of bacteriophage T7. *The Journal of biological chemistry*, **276**, 49419-49426.
2. Frick, D.N., Baradaran, K. and Richardson, C.C. (1998) An N-terminal fragment of the gene 4 helicase/primase of bacteriophage T7 retains primase activity in the absence of helicase activity. *Proceedings of the National Academy of Sciences of the United States of America*, **95**, 7957-7962.
3. Ilic, S., Cohen, S., Afek, A., Gordan, R., Lukatsky, D.B. and Akabayov, B. (2019) DNA Sequence Recognition by DNA Primase Using High-Throughput Primase Profiling. *J Vis Exp*.

4. Afek, A., Ilic, S., Horton, J., Lukatsky, D.B., Gordan, R. and Akabayov, B. (2018) DNA Sequence Context Controls the Binding and Processivity of the T7 DNA Primase. *iScience*, **2**, 141-147.
5. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, O., Dubourg, V. *et al.* (2011) Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, **12**, 2825-2830.
6. Brown, C.E. (1998) *Coefficient of Variation*. Springer, Berlin, Heidelberg.
7. Ward, J.H.J. (1963) Hierarchical Grouping to Optimize an Objective Function. *J Am Stat Assoc*, **58**, 236–244.
8. Zadeh, J.N., Steenberg, C.D., Bois, J.S., Wolfe, B.R., Pierce, M.B., Khan, A.R., Dirks, R.M. and Pierce, N.A. (2011) NUPACK: Analysis and design of nucleic acid systems. *J Comput Chem*, **32**, 170-173.