Improved prediction of bacterial CRISPRi guide efficiency through data integration and automated machine learning

Yanying Yu[1], Sandra Gawlitt[1], Lisa Barros de Andrade e Sousa[2], Erinc Merdivan[2], Marie Piraud[2], Chase Beisel[1,3], Lars Barquist[1,3*]

[1]Helmholtz Institute for RNA-based Infection Research (HIRI) / Helmholtz Centre for Infection Research (HZI), 97080 Würzburg, Germany.

[2]Helmholtz AI, Helmholtz Zentrum München, 85764 Neuherberg, Germany.

[3]Medical Faculty, University of Würzburg, 97080 Würzburg, Germany.

*To whom correspondence should be addressed: lars.barquist@helmholtz-hiri.de (to L.B.)

Abstract

CRISPR interference (CRISPRi), the targeting of a catalytically dead Cas protein to block transcription, is the leading technique to silence gene expression in bacteria. Genome-scale CRISPRi essentiality screens provide one data source from which rules for guide design can be extracted. However, depletion confounds guide efficiency with effects from the targeted gene. Using automated machine learning, we show that depletion can be predicted by a combination of guide and gene features, with expression of the target gene having an outsized influence. Further, integrating data across independent CRISPRi screens improves performance. We develop a mixed-effect random forest regression model that learns from multiple datasets and isolates effects manipulable in guide design, and apply methods from explainable AI to infer interpretable design rules. Our method outperforms the state-of-the-art in predicting depletion in an independent saturating screen targeting purine biosynthesis genes in *Escherichia coli*. Our approach provides a blueprint for the development of predictive models for CRISPR technologies in bacteria.

Introduction

CRISPR interference (CRISPRi), in which a catalytically dead Cas protein incapable of DNA cleavage (dCas) is targeted to interfere with transcription of a gene of choice (Bikard et al., 2013; Qi et al., 2013), is the most widely used CRISPR technology in bacteria. In contrast to eukaryotes, many bacteria lack the necessary repair pathways to survive genome editing by the double-stranded break induced by CRISPR-Cas9. Applications of CRISPR-Cas9 as a sequence-specific antibiotic notwithstanding (Bikard et al., 2014; Citorik et al., 2014; Gomaa et al., 2014), the main impact of CRISPR-Cas in engineering bacteria has come from using it as a platform on which to develop new technologies that can be guided to a specific locus in a programmable fashion. CRISPRi is the simplest example of this, where the dCas protein itself serves as an effector to silence gene expression by physically blocking the procession of the RNA polymerase.

The development of CRISPRi has opened up a range of biological applications, from down-regulating individual genes for genetic studies to performing genome-wide fitness screens or engineering genetic circuits (Luo et al., 2016; Vigouroux and Bikard, 2020). As an alternative screening technology to transposon mutagenesis (Cain et al., 2020), CRISPRi has the advantage that particular genes of interest can be directly targeted, avoiding the need for large mutant libraries to achieve gene saturation. Another area of application is engineering synthetic regulatory circuits (Jusiak et al., 2016) or metabolic networks (Cho et al., 2018a; Mougiakos et al., 2018), where collections are gRNAs are used to coordinately downregulate and upregulate associated genes and pathways. However, all of these applications critically depend on the efficiency of silencing provided by selected guides. Genetic screens already routinely employ tens of thousands of guides simultaneously, and it is impractical to individually test each guide's efficiency. This problem will only be accentuated as the scale of applications increases through the use of CRISPR array technology that allows multiplexed expression of suites of guides simultaneously (Liao et al., 2019; Reis et al., 2019) to dissect and engineer increasingly complex phenotypes. Reliable prediction of guide efficiency will therefore become increasingly important as applications of CRISPRi become increasingly ambitious.

Given the impact of CRISPR-based genome engineering in eukaryotes, significant effort has been expended in developing methods for predicting editing efficiency. The first attempts used classical machine learning methods on relatively small datasets comprising efficiency measurements for thousands of gRNAs. The applied methods include logistic regression (Doench et al., 2014), support vector machines (Labun et al., 2016; Wong et al., 2015), linear regression (Moreno-Mateos et al., 2015), and gradient-boosted decision trees (Doench et al., 2016). As the amount of Cas9 editing data has increased, deep learning approaches have become increasingly popular. These include convolutional neural networks (Chuai et al., 2018), which apply a collection of adaptive filters to automatically extract local sequence features, and long short-term memory networks (LSTM) (Wang et al., 2019), which retain a memory that potentially allows for the detection of long-range interactions between sequence features. Newer methods have put substantial effort into engineering deep learning architectures to further boost performance (Kim et al., 2019). It is important to note that many of these deep learning methods have been trained on tens of thousands of measurements of guide efficiency, and fusing datasets has played an important role in further increasing performance (Xiang et al., 2021).

So far, relatively little attention has been paid to predicting guide efficiency for bacterial CRISPRi. The only study to date developed a LASSO regression model for predicting CRISPRi guide efficiency (Calvo-Villamañán et al., 2020) with a limited sequence feature set using data from a single genome-wide CRISPRi screen in *Escherichia coli* (Rousset et al., 2018). Given the trajectory of prediction methods for eukaryotic genome engineering applications towards larger datasets and more complex machine learning approaches, we asked if a similar approach could improve our ability to extract design rules for gRNAs for bacterial CRISPRi applications. Starting with an investigation of features driving guide depletion in CRISPRi screens using automated machine learning, we find that gene effects that are not modifiable in guide design dominate. Starting from this foundation, we develop a machine learning approach that separates gene and guide effects while learning from multiple independent CRISPRi screens, allowing us to arrive at a predictive model of guide efficiency that we show

improves on the state-of-the-art using a saturating depletion screen of purine biosynthesis genes during growth in minimal media.

Results

Automated machine learning and feature engineering identifies gene-specific effects in CRISPRi depletion screens

We set out to devise design rules for CRISPRi in bacteria by combining machine learning with large experimental datasets. The largest available datasets come from genome-wide depletion screens. However, it is currently unknown how well depletion in these screens can be predicted given known guide and genomic features. We thus began our investigation by applying automated machine learning (autoML) (**Figure 1A**). AutoML refers to a collection of techniques that attempt to automate the often labor-intensive process of model selection and optimization. Rather than sequentially fitting different types of models and individually optimizing their hyperparameters as typically done in applying ML to a new problem, autoML techniques turn model selection itself into an optimization problem. We used the Auto-Sklearn package (Feurer et al., 2015) that wraps classification and regression models implemented in the Python Scikit-learn package (Pedregosa et al., 2011) in a Bayesian optimization framework.

We first asked how well we could predict gRNA depletion $\log_2$ fold-changes (logFCs) for essential genes as defined by the Keio collection (Baba et al., 2006), and what features would be required for accurate prediction. As essential genes should have an infinite fitness cost upon complete silencing, we assumed differences in depletion would mainly depend on gRNA silencing efficiency. We leveraged a published *E. coli* CRISPRi essentiality screen using dCas9 performed in rich media (Rousset et al., 2018), which included 1,951 guides targeting 293 essential genes. To predict depletion in this data set, we engineered a series of feature sets of increasing complexity (**Figure S1**; **Table S1**) starting with the one-hot-encoded gRNA and PAM sequence as well as the one-hot encoded dinucleotide sequence including four bases upstream of the gRNA sequence and three bases following the NGG PAM. This resulted in a poorly performing model with a median Spearman's ρ of ~0.20 in 10-fold cross-validation (**Figure 1B; Table S2**). We therefore iteratively added a set of additional features while monitoring changes in model performance. As targeting

efficiency has been suggested to depend on distance to the transcriptional start site (Qi et al., 2013; Wang et al., 2018), the set included absolute and relative distance to the start codon. We also included a suite of thermodynamic features describing gRNA:target interactions predicted using the ViennaRNA package (Lorenz et al., 2011): minimum free energy of the folded gRNA, hybridization of two gRNAs, and hybridization of the targeted DNA and gRNA (Lorenz et al., 2012). These additional feature sets resulted in only moderate improvement in Spearman correlation ($\rho \sim 0.24$) for our predictions.

Given that features describing the guide sequences themselves were inadequate to predict guide depletion, we developed a series of features associated with each targeted gene that we reasoned may have some explanatory power (**Figure 1B**). First we used gene ID alone as a predictor, reasoning that incomplete silencing of essential genes may lead to different rates of depletion. While doing so improved the accuracy of predictions, the Spearman correlation between predicted and measured log-change remained below 0.4. We reasoned that additional information about each gene might improve our capacity to predict depletion, and so we constructed eight additional features describing each gene. We collected publicly available RNA expression data over growth in minimal media (Conway et al., 2014) and computed minimum and maximum expression values. We collected transcription unit (TU) information from RegulonDB (Santos-Zavaleta et al., 2019) and calculated the distance from the target site to the start of the TU, the number of downstream genes in each TU, and the presence of other essential genes in the TU. Finally we also included gene GC content. Incorporating these additional gene features led to a major improvement in prediction accuracy, with cross-validation performance jumping to a Spearman's ρ of ~0.68

To understand the contribution of these features to the prediction of gRNA depletion, we used SHapley Additive exPlanation values (SHAP values) computed with TreeExplainer (Lundberg et al., 2020) on the best performing random forest regression model produced by Auto-Sklearn (**Figure 1C**; **Table S3**). SHAP values are a game-theoretic approach to feature importance that capture the marginal contribution of a given feature to a prediction. Looking at average absolute SHAP values provides a measure of feature importance, while plotting individual SHAP values shows how each

feature affects each individual prediction. Of all considered features, maximal RNA expression had the single largest average effect on depletion, making an average of a ~1.7 fold difference to the predictions. Unexpectedly, high target gene expression tended to be associated with higher depletion. There was also clear evidence for polar effects from CRISPRi, as the number of downstream essential genes was highly predictive of increased depletion. The most predictive effects that could actually be modified by guide design were associated with guide distance to the transcriptional start site, but on average these had fairly small effects compared to features associated with the target gene. In summary, we found that autoML can rapidly produce predictive models of CRISPRi guide depletion, and the predictions made by these models are dominated by the effects of gene features that can not be modified in guide design. These findings outline key features that need to be accounted for to accurately infer guide efficiency from genome-wide depletion screens.

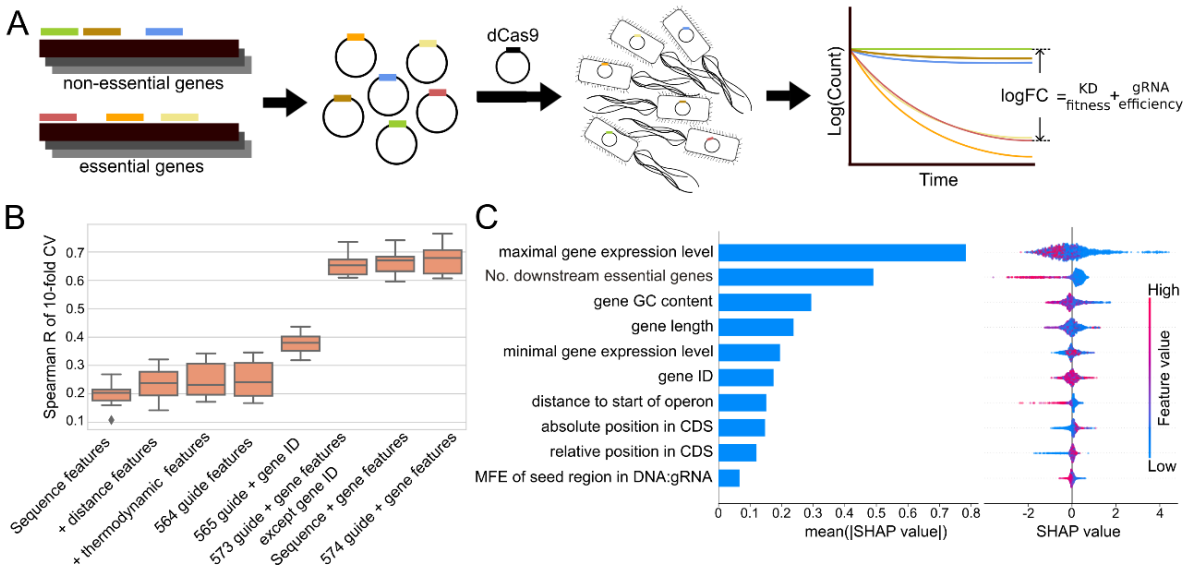

Figure 1: Automated machine learning predicts depletion in CRISPRi essentiality screens. (**A**) An overview of CRISPRi essentiality screens. gRNAs are designed targeting every gene in the genome and cloned into an appropriate plasmid for expression. This plasmid collection is then transformed into the target bacteria, and depletion is measured as the change in guide frequency over growth determined by sequencing relative to a set of non-targeting gRNAs. The measured depletion (logFC) is then a mixture of the fitness effect of gene knockdown with the efficiency of silencing itself. **(B)** Comparison of Spearman

correlation between actual and predicted guide depletion in 10-fold cross-validation (CV) of the best model trained with Auto-Sklearn with different feature combinations, using data from (Rousset et al., 2018). **(C)** The ten most predictive features determined using TreeExplainer on the optimal random forest model trained with Auto-Sklearn and 574 guide and gene features. Mean absolute SHAP value (left) provides a global measure of feature importances, while the beeswarm plot (right) shows the effect of each feature on each individual gRNA prediction.

Data fusion improves prediction performance

As we had exhausted new features to explore, we next asked whether the size of our dataset was limiting the accuracy of our predictions. To this end, we collected data from two additional CRISPRi screens of *E. coli* in rich media. First, we included data from an additional screen using the same gRNA library but with dCas9 expressed from a stronger promoter, which we refer to here as E18 Cui (Cui et al., 2018). Second, we included data from a completely independent screen using a higher density library containing twice as many guides targeting essential genes (4197; 528 are identical to gRNAs contained in Cui/Rousset), which we refer to as Wang (Wang et al., 2018). We refer to the original data set as E75 Rousset. It is also worth noting that while the E18 Cui and E75 Rousset libraries were grown repeatedly to stationary phase, the Wang screen was collected in log phase. The level of depletion in each dataset exhibited qualitative differences, with Wang showing a clearer bimodal separation between depleted and non-depleted guides (**Figure 2A**). There was a reasonable correlation of depletion between datasets, with E18 Cui and E75 Rousset exhibiting a Spearman's ρ of ~0.9. The correlation between Wang and the other two datasets was lower (ρ~0.75-0.79), but this seemed mostly attributable to a saturation effect in Wang, possibly due to the shorter growth period (**Figure S2**).

To investigate the impact of fusing these datasets on model performance, we trained a series of models using Auto-Sklearn with each dataset individually or in combination including a dataset indicator as a potential predictor, and we tested them on sets of guides held out from each dataset as well as a mixed test set (**Figure 2B**; **Table S4**). Unsurprisingly, models trained on single datasets tended to perform best on their cognate test set. Similarly, models trained on E18 Cui and E75 Rousset appeared to generalize better to each other than to the Wang dataset and vice versa. Combined

training datasets, particularly those mixing at least one of the Cui/Rousset sets with Wang, generalized better across datasets without degrading performance relative to models trained on individual datasets. In some cases, particularly with the Cui dataset, fused training sets actually improved performance on a test set drawn from a single dataset. In each case, the best performing model chosen by Auto-Sklearn was either a random forest regression or a gradient-boosted decision tree model.

To illustrate that the performance increases we saw when combining datasets was not an artifact of our autoML procedure, we tested data fusion with both an alternative autoML package, H2O (LeDell and Poirier, 2020) (**Table S4**) as well as a suite of individual model types (**Figure S3; Table S5**). Different model types responded differently to the fused data, with linear regression-based models showing little improvement (e.g. linear regression, lasso linear regression, elastic net linear regression; **Figure S3 A-C**), while tree-based methods (e.g. random forest regression, histogram-based gradient boosted trees; **Figure S3 E,F**) showed clear improvement. Importantly, none of the tested models appeared to degrade in performance when trained with fused data. These findings suggest that both increased generalizability and accuracy can be achieved by integrating multiple data sources for training tree-based models for CRISPRi depletion.

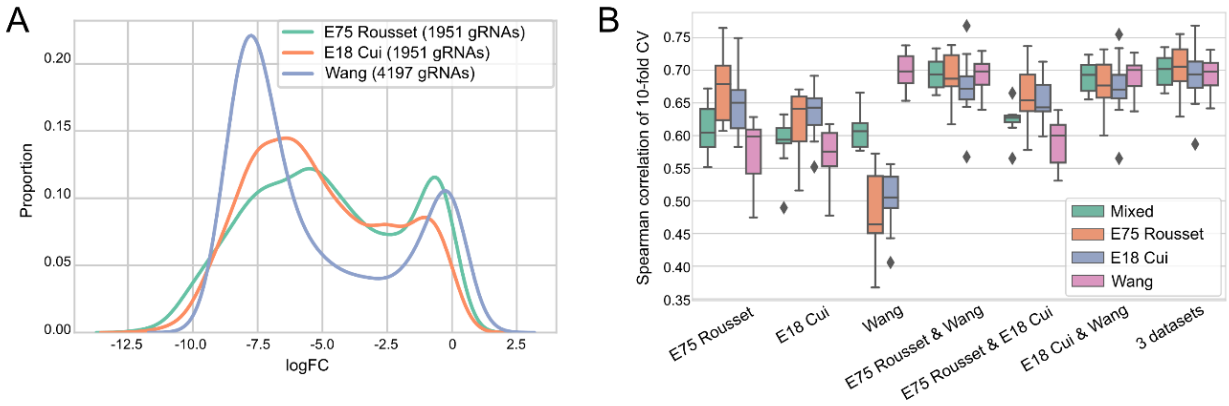

Figure 2: Data fusion improves prediction of depletion in genome-wide CRISPRi screens. (A) Distribution of logFCs of gRNAs targeting essential genes from three CRISPRi genome-wide essentiality screens in *E. coli.* **(B)** Comparison of Spearman correlation from 10-fold CV of the best Auto-Sklearn trained model on one dataset or the indicated combination of datasets.

Segregating guide and gene effects produces a predictive model for CRISPRi guide efficiency

Our exploration of the features most predictive of gRNA depletion in competitive screens highlighted that features describing the targeted gene often made much larger contributions to the prediction than features describing the guide sequence. This is problematic for predicting guide efficiency from depletion screens, as this large gene-to-gene variation in depletion must be removed to properly extract the contribution of guide efficiency.

We took two distinct approaches to separating guide and gene effects. The first was to explicitly model both effects jointly using Mixed-Effect Random Forest (MERF) regression (Hajjem et al., 2014). The MERF model handles data with an underlying cluster structure by defining two separate models: a linear model that captures random effects associated with the cluster, and a random forest (or other complex model) that captures fixed effects associated with each individual measurement. These models are then jointly optimized in an iterative process using the expectation-maximization algorithm. In our case, random effects correspond to features associated with each gene (e.g. gene ID, expression level) as well as dataset, while fixed effects correspond to features that could be manipulated in gRNA design (e.g. PAM and guide sequence, thermodynamic properties).

For the second approach, which we refer to as median subtracting (MS), we subtract the gene-wise median logFC from each gRNA depletion value to calculate relative “activity scores” following previous work (Calvo-Villamañán et al., 2020). However, this leads to problems in integrating multiple datasets, as the range of depletion values varies across datasets (**Figure 2A**). We adapted a previously described approach used for fusing CRISPR gene deletion datasets (Xiang et al., 2021). First, we averaged the logFCs between E75 Rousset and E18 Cui which share all guides in common. We then calculated a linear scale factor for guides shared between Wang and the averaged Rousset/Cui data set to make logFCs for the unshared guides in Wang comparable to logFCs derived from Rousset/Cui (**Figure S4A-C**). For cross-validation, scaling was performed within each test fold to avoid possible leakage of information between test and training sets.

Both the fixed effect model from the MERF and activity scores in the MS method remove gene-specific effects to estimate guide efficiency, making guide-wise cross-validation difficult as the true guide efficiency is unknown. As an alternative to guide-wise cross-validation, we developed a gene-wise cross-validation scheme. We trained new models using 10-fold cross validation, this time holding out all guides targeting a set of held-out genes, evaluating the Spearman correlation between predictions and measured depletion within each gene under the assumption that rank order should reflect guide efficiency within a gene.

We trained and tested three models. Since tree-based models performed best on predicting gRNA depletion, we trained both a MERF model and a random forest trained on MS data. These were compared to both a published LASSO model based on the MS approach (Calvo-Villamañán et al., 2020) (hereafter referred to as “Pasteur”) and a LASSO model we trained to evaluate the effect of our expanded feature and data sets on prediction accuracy. It should be noted that the Pasteur model was trained on the E75 Rousset data, so our benchmark results are not independent of its training data and will tend to overestimate the performance of the Pasteur model.

As we had previously observed in our evaluation of predictions of guide-wise depletion, data fusion between multiple CRISPRi screens consistently improved performance across models. In aggregate, the random forest models performed slightly better than the LASSO-based models (median ρ=0.375 (MERF) and 0.378 (MS) vs. 0.357 (Pasteur)). When we broke this down into performance on held-out genes in individual datasets (**Table 1**), the MERF and Pasteur performed roughly similarly on the E75 Rousset data on which the Pasteur model was trained (ρ=0.418 vs. 0.429) and the E18 Cui data from the same lab (ρ=0.418 vs. 0.411). Both random forest models performed better than the Pasteur model on the independent Wang dataset (0.354 and 0.344 vs. 0.298, respectively). A similar trend was seen in comparison with the MS LASSO model, where the Pasteur model performed better on the E75 Rousset and E18 Cui data, and worse on Wang. The MERF and MS random forest models performed generally similarly to one another, likely due to the high correlation observed between median gene-wise logFC and the MERF-predicted random effects across our datasets (**Figure S4D**). In sum, we find that random forests trained on multiple datasets

outperform simpler regression models on predicting guide efficiency for held-out genes, and that the MERF approach provides a straight-forward means of integrating datasets while isolating effects important for guide efficiency.

Table 1: Evaluating predictions of guide efficiency after removing gene effects. Spearman correlations between predictions and measured logFC for held-out genes. Genes were held out in 10-fold cross-validation, and the reported median Spearman correlation was calculated across all held-out genes.

Model	Training data	Median Spearman Correlation across held-out genes			
		E75 Rousset	E18 Cui	Wang	Mixed
MERF	E75 Rousset	0.333	0.300	0.264	0.300
	E18 Cui	0.371	0.389	0.280	0.331
	Wang	0.321	0.382	0.350	0.351
	3 datasets	0.418	0.418	0.354	0.375
MS (RF)	E75 Rousset	0.365	0.333	0.281	0.318
	E18 Cui	0.373	0.404	0.297	0.341
	Wang	0.357	0.393	0.343	0.354
	3 datasets	0.400	0.401	0.344	0.378
MS (LASSO)	E75 Rousset	0.321	0.298	0.259	0.286
	E18 Cui	0.306	0.314	0.305	0.308
	Wang	0.314	0.318	0.327	0.320
	3 datasets	0.385	0.400	0.332	0.361
Pasteur	-	0.429	0.411	0.298	0.357

Model interpretation with explainable AI illustrates rational design rules for CRISPRi

To understand the features underlying model performance, we again examined SHAP values for our random forest models using TreeExplainer (Lundberg et al., 2020). We observed similar features with large impacts on predictions from both random forest models (**Figure 3A**; **S4E**; **Table S7**). In the MERF model, the strongest average effects were seen for a guanine at the +1 position following the PAM, followed by distances from the start codon. In particular, we found that targeting positions further from the start codon led to reduced guide efficiency, as has been inferred previously (Qi et al., 2013). Other top features involved the nucleotide at position 20 of the guide, directly adjacent to the PAM sequence (**Figure 3A & B**). Here guanine and particularly adenine at this position negatively impacted silencing efficiency, while cytosine and thymine increased

efficiency — almost the exact inverse of previous reports for Cas9 efficiency in eukaryotic genome editing applications (Doench et al., 2014). Within and following the PAM sequence, our SHAP values were qualitatively similar to previous observations in Cas9 genome editing. Cytosine was favored at the variable position of the NGG PAM, and a guanine residue immediately following the PAM had a negative impact on silencing, though we additionally observed a positive impact of cytosine at this position. These effects within and around the PAM sequence appeared to interact with each other, as we saw additional effects for dinucleotide sequences covering the end of the guide sequence and first residue of the PAM where cytosine-cytosine and thymine-cytosine residues improved performance while guanine-guanine residues had a strong negative effect.

To further investigate potential interactions between features, we estimated SHAP interaction values that quantify situations in which the presence of one feature changes the impact of another, so that the combined SHAP value for both features together is not the simple sum of each feature's SHAP value. To provide a visualization of these interactions, we calculated expected effects using the median SHAP value for each feature from guides containing only one of the interacting features, and compared the expected sum to the actual SHAP values for guides containing both features (**Figure 3C**; **Table S8**).

The majority of these interactions involved distance features or bases in the vicinity of the PAM. For instance, we saw a range of interactions between position 20 of the guide and the +1 position immediately downstream of the PAM. This can lead to guides with either reduced (**Figure 3C I**) or increased efficiency (**Figure 3C II&III**) compared to expectations based on single feature SHAP values. We also observed interactions between the variable position of the NGG PAM and surrounding bases, where for instance having an otherwise nearly neutral thymine at the variable PAM position (P1) can lead to a stronger reduction in efficiency when a guanine is present immediately downstream of the PAM (+1) (**Figure 3C IV**). The existence of such interactions between features in the guide sequence may provide one explanation for the superior performance of tree-based methods over linear regression, as tree regressors are particularly well suited to capture interaction effects.

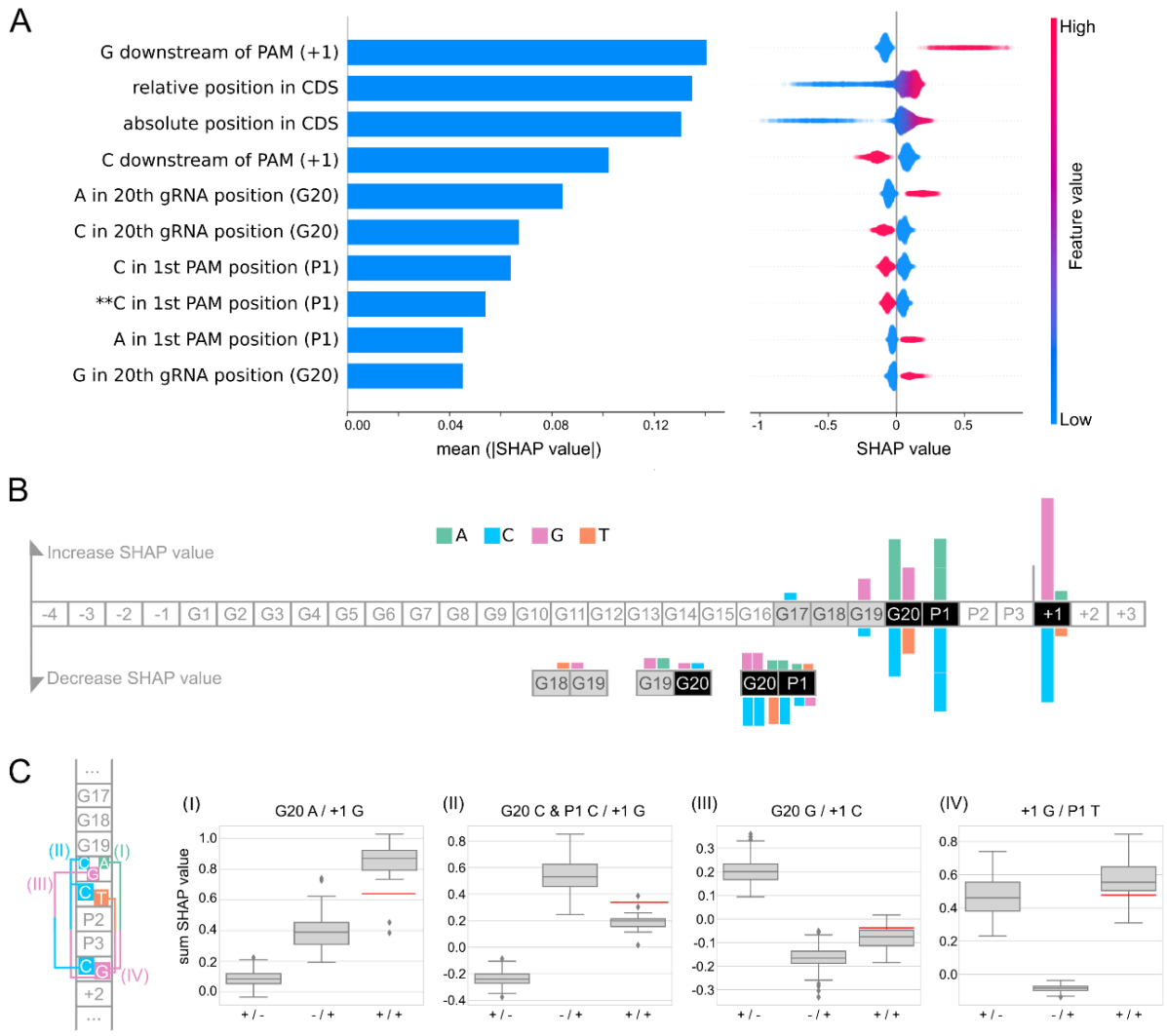

Figure 3: Important features for CRISPRi guide efficiency illustrate sequence preferences and interactions. (**A**) SHAP values for the top 10 features from MERF optimized random forest model. Global feature importance is given by the mean absolute SHAP value (left), while the beeswarm plot (right) illustrates feature importance for each guide prediction. (**: derived from dinucleotide features) (**B**) A summary of effects of sequence features. Increased SHAP values indicate features that lead to reduced guide efficacy, while decreased SHAP values indicate increased guide efficacy. The guide sequence is numbered G1 to G20 and the three positions of the PAM sequence are labeled P1, P2, P3. Negative and positive numbers refer to positions preceding the guide sequence and following the PAM, respectively. (**C**) An illustration of feature interactions. The schematic on the left illustrates the positions of three representative interacting positions in the vicinity of the PAM sequence. (**I-IV**) show SHAP values for features in guides containing one (+/-) or the other (-/+) feature, or both (+/+). The expected SHAP value (red line) is calculated as the sum of the median SHAP values observed for each feature when occurring

independently. 20 C/G/A: C/G/A in 20th gRNA position, +1 C/G: C/G downstream of PAM, P1 T/C: T/C in 1st PAM position.

Deep learning approaches do not improve prediction performance

Given that we saw better performance with tree-based methods over linear regression, we next asked if model complexity was a limiting factor in prediction. Deep learning approaches have been applied to predicting guide efficiency for a number of CRISPR technologies (Chuai et al., 2018; Kim et al., 2018, 2019; Wang et al., 2019; Xiang et al., 2021). Considering this, we asked if deep learning models would also improve performance in predicting gRNA efficiency for CRISPRi in bacteria. As a representative architecture, we implemented a one-dimensional convolutional neural network (CNN), which runs a series of kernel filters across the sequence to extract local features. In addition to a custom CNN architecture, we reimplemented and tested the state-of-the-art deep learning architecture used for predicting Cas9 gene editing efficiency by CRISPRon (Xiang et al., 2021), only trained using our CRISPRi data.

For our custom CNN architecture, we used the convolutional layers to extract sequence features before concatenating them to the rest of our guide feature set (**Figure S5A**). This concatenated feature set was then fed through a fully connected 4-layer multilayer perceptron (MLP) for regression using MS values for guide efficiency. Both the custom CNN and CRISPRon models exhibited lower Spearman correlations as compared to our previously trained random forest models when tested on held-out gene sets (**Figure S5B; Table S6;** CNN ρ=0.326, CRISPRon ρ=0.333, vs. MERF ρ=0.375). These results show that conventional machine learning approaches can outperform deep learning architectures and suggest that data may currently be limiting for more complex machine learning approaches.

A saturating screen of purine biosynthesis genes independently validates performance of tree-based models and data fusion

Our previous benchmarking indicated that tree-based methods trained on multiple datasets outperformed other methods in predicting guide efficiency. However, these results were based entirely on cross-validation within our training datasets. To produce

a truly independent test set, we first targeted a plasmid-encoded GFP construct with 19 gRNAs across a range of predicted guide efficiencies, and measured the reduction in cell fluorescence by flow cytometry (**Figure 4A**). Measuring performance by Spearman correlation, we found both random forest methods performed best (ρ=0.70 MS RF, 0.68 MERF) followed by our LASSO model (ρ=0.55), while the Pasteur model performed comparatively poorly (ρ=0.26). Replicating this study in *Salmonella* Typhimurium gave qualitatively similar results, though with lower Spearman correlations (**Figure S6A**). However, when we reanalyzed the data from a Miller assay (measuring β-Galactosidase activity) previously used to validate the Pasteur model (Calvo-Villamañán et al., 2020) (**Figure 4B**; **Table S10**), we found that the Pasteur model performed best (ρ=0.71), followed by the MERF and MS random forests (ρ=0.65, 0.59) and the LASSO model. We also tested three tools designed for predicting Cas9 guide efficiency for genome editing in eukaryotes (Doench et al., 2016; Kim et al., 2019; Wilson et al., 2018), and all performed universally poorly on both data sets.

While the exact reasons for the discrepancies in performance between our GFP measurements and the Miller assay are unclear, one plausible explanation is that these data sets simply have sample sizes too small to discriminate between prediction methods. To resolve this, we performed a high-throughput screen targeting nine genes from the purine biosynthesis pathway of *E. coli* known to be essential in minimal media, spread across seven independent transcriptional units (**Figure 4C**). To avoid any bias in guide selection, we saturated all potential target sites in each gene, ending with a total of 750 gRNAs, including between 35 and 223 guides per gene. Duplicate samples were then collected at three time points during growth in M9 minimal medium, and gRNA depletion was measured with reference to input samples, normalized using a set of 50 gRNAs designed not to target any *E. coli* sequence.

Comparing the experimentally determined depletion values to predictions from our tested models confirmed the results of our previous cross-validation (**Figure 4D**; **Table S13**): the MERF and MS random forest models performed best, with the MERF performing slightly better overall (median ρ~0.56 vs. ~0.55 across all time points). Comparing the MERF trained on the three fused data sets to the MERF trained on any single data set also showed improved performance. The choice of a tree-based model

also made a clear difference in performance, as a LASSO model trained on the same data and feature set showed worse performance (ρ~0.49—0.50) than either random forest. Both of our random forest models as well as our LASSO model performed better than the Pasteur model (ρ~0.43—0.48).

Beyond validating the performance of our models, our saturating screen of purine biosynthesis genes also revealed previously unobserved features of CRISPRi depletion screens. First, there were two genes, *purE* and *purK*, on which all methods performed poorly as measured by Spearman correlation. Upon inspection of the depletion values, it became clear that this was because there was surprisingly little variation in guide efficiency along these transcripts (**Figure 4E**; **S6C**). This meant that for these genes, differences in ranking reflected very small differences in depletion, likely within the error of our experimental measurements. We examined our initial training set to see if this might be a more widespread phenomenon, finding a substantial number of genes with low variation in their guide depletion values (**Figure S6D**). This may be a factor in the overall low average Spearman correlations we report in our cross-validation.

A second unexpected feature was the overall lack of a clear relationship between guide efficiency and distance to the transcriptional start site. Of the nine tested genes, only two, *purC* and *purM*, showed a clear linear dependency of depletion on position within the gene sequence. This was particularly surprising, as distance features were clearly important to our model predictions. We attempted to train a model excluding distance features, but this substantially degraded performance on predicting depletion in our high-throughput screen (**Figure S6B**). Whether this is an artifact of our training data, based on screens which used small collections of guides biased towards the 5′ end of genes, or if other guide features compensate for positional differences in guide efficiency remains unclear. In support of the latter, our analysis of feature interactions found many of the strongest effects came from interactions between distance features and sequence features in the vicinity of the PAM (**Table S8**), suggesting that sequence features have larger effects on efficacy as the distance from the transcription start site increases. In sum, our screen of guides targeting purine biosynthesis genes independently validated the better performance of our random forest models compared to the state-of-the-art, while also highlighting some unexpected features of CRISPRi.

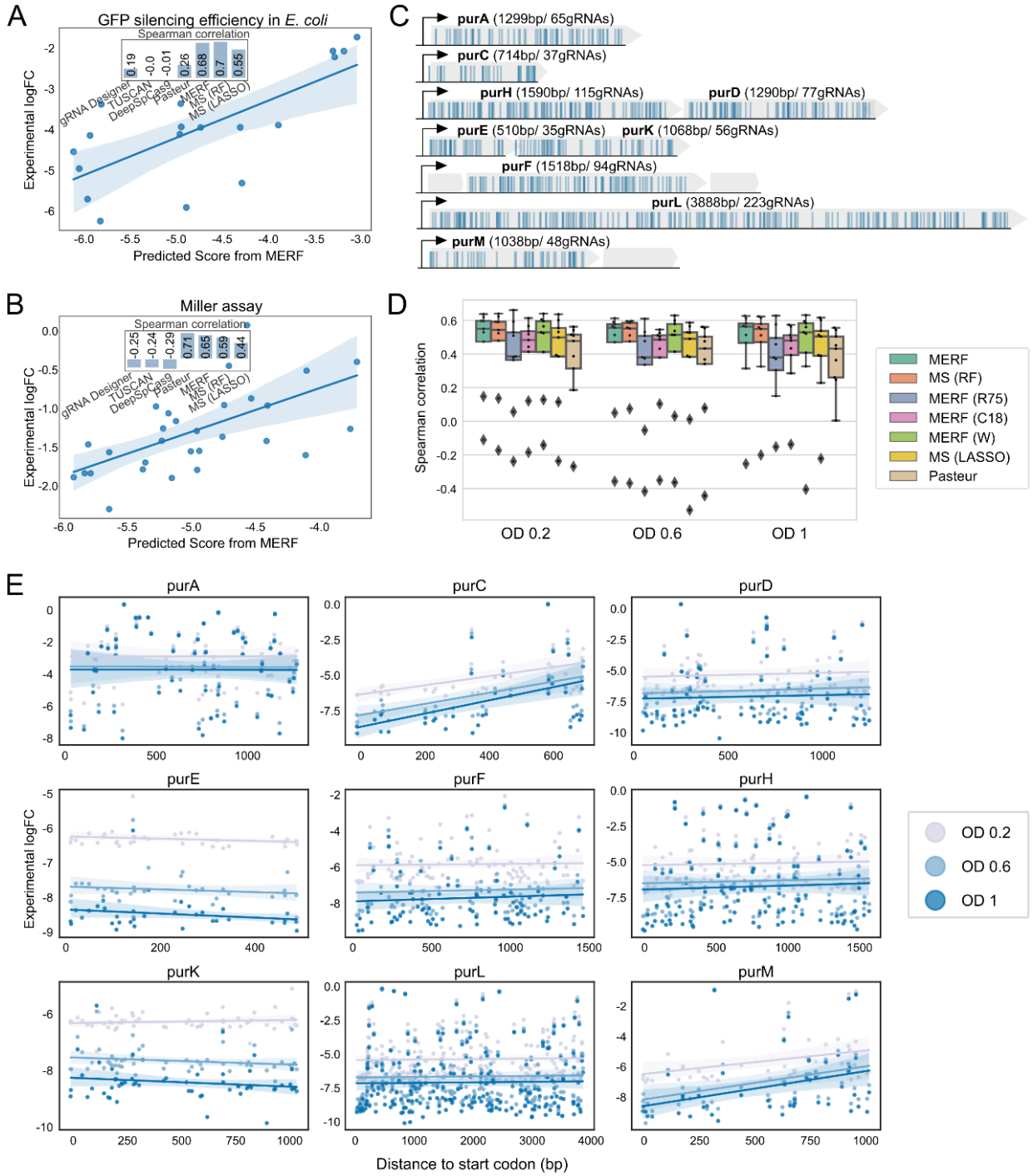

Figure 4: Independent validation of model performance using a saturating screen of purine biosynthesis genes. (**A**) The activity of 19 gRNAs targeting a plasmid-expressed deGFP gene was measured in *E. coli* using a flow cytometry-based assay. The measured activity compared to the control gRNA is plotted against the score predicted by the MERF model. The inset barplot illustrates Spearman

correlations for seven methods for predicting guide efficiency. (**B**) The activity of 30 gRNAs targeting *lacZ* measured with a Miller assay by Calvo-Villamañán et al., plotted as in A. **(C,D,E)** High-throughput screening of 750 gRNAs targeting 9 purine biosynthesis genes in *E. coli* K12 MG1655. **(C)** Transcriptional architecture of the targeted genes. All possible gRNAs were designed for each gene; each blue vertical line represents a gRNA. Grey boxes represent genes, black arrows transcriptional start sites. **(D)** Spearman correlations between the predicted scores and measured logFC across collected timepoints. **(E)** Measured logFCs for each guide as a function of distance to the start codon for each gene.

Discussion

In this study, we developed a predictive model for CRISPRi guide efficiency using integrated data from three gene essentiality screens in *E. coli*. We extensively explored the process of model development, evaluating how feature engineering, data integration, and model selection affect performance. We have shown that this model improves on the previous state-of-the-art using both gene-wise cross-validation on our training data as well as a fully independent screen of guides targeting purine biosynthesis genes essential in minimal media. These investigations provide a blueprint for developing similar predictive models, both for other CRISPR-Cas systems (Vialetto et al., 2021) and technologies, as well as for CRISPRi in different bacteria where design rules may vary. We have made a web server for predicting CRISPRi guide efficiency using our MERF publicly available at: https://ciao.helmholtz-hiri.de.

Prediction of guide efficiency will become increasingly important with more complex applications of CRISPR technologies. In particular, the potential for multiplexing CRISPRi presents could be transformative when compared to established technologies. One example of this would be in screening for fitness interactions between genes. The current state-of-the-art is based on arrayed mating of single gene deletion libraries (Butland et al., 2008; Typas et al., 2008), which is both labor intensive and technically challenging, and becomes increasingly so when querying higher-order interactions (Kuzmin et al., 2018). A similar example is in metabolic engineering where multiplexed CRISPRi can be used to modulate biosynthetic pathways to optimize

production of a particular metabolite for industrial applications (Lian et al., 2017). The development of CRISPR array technologies that can coexpress as many as 22 guides simultaneously (Liao et al., 2019; Reis et al., 2019) should accelerate the development of these approaches. However, large-scale, multiplex applications will require better tools for guide design to ensure robust results. Individually screening guides for activity quickly becomes prohibitive when one considers applications that require hundreds or thousands of guides. The machine learning approach presented here provides a straight-forward solution to this problem.

Applying machine learning to any problem presents a series of challenges. These include collecting data, engineering relevant features, model selection and optimization, and validation. Here, we have approached each of these challenges systematically. We have shown that incorporating data from multiple, independent CRISPRi screens improves model performance. This result suggests two things: that individual datasets may contain batch effects that affect generalizability, as training on combined datasets improved performance across datasets; and that available data may be a limiting factor in model performance, as training on combined datasets also improved performance within individual datasets, particularly those with fewer guides. We have also shown that a rich, biologically relevant feature set is important for predicting CRISPRi depletion. Strikingly, we found that gene identity alone was a poor predictor of depletion when compared to including measures of gene expression and potential for polar interactions within transcriptional units. In particular, gene expression was the single largest contributor to gene depletion as measured by SHAP values, and higher expression was counterintuitively associated with higher depletion. As the availability of transcriptomics data may be lacking for some organisms, we also tested the possibility of using the codon adaptive index (CAI) as a proxy, with promising results (**Figure S6B**). While these results do not necessarily imply a direct causal relationship between expression and depletion, they do suggest that caution should be taken when comparing guide depletion levels between genes in a screen as factors other than gene fitness may strongly influence the degree of depletion.

Model selection and tuning also had a large impact on prediction performance. The standard approach to developing a machine learning model for a particular

application generally involves a significant degree of trial and error. This is true both for selecting a type of model and tuning model hyperparameters, which is often critical to performance. To avoid this, we applied automated machine learning, which turns model selection and tuning into an optimization problem. Auto-Sklearn (Feurer et al., 2015) searches over a set of twelve regressors and their hyperparameters and was able to reliably select a final model comparable in performance to hand-tuning. For all of our various data and feature sets, Auto-Sklearn tended to select tree-based regressors. This is in contrast to previous work that suggested linear regression was adequate to capture guide efficiency (Calvo-Villamañán et al., 2020). Possibly this is due to the more complex feature set we constructed, containing a wide range of features beyond simple sequence features. We also applied several deep learning approaches, including the architecture successfully used by CRISPRon to predict Cas9 genome editing efficiency (Xiang et al., 2021), but these failed to achieve similar performance to tree ensemble regressors. This again suggests that the availability of data for training may be a limiting factor, as deep learning models often require large sample sizes to achieve high performance.

Our finding that gene features that cannot be modified during guide design are dominant in determining depletion in CRISPRi screens highlighted the importance of removing these before attempting to predict guide efficiency. Whereas predictive methods for CRISPR-Cas genome engineering tools targeted at eukaryotes are often trained on large datasets with direct measurements of guide efficiency (e.g. indel rates), for bacterial CRISPRi the largest data sets come from essentiality screens, which provide only indirect measurements of guide efficiency. We took two distinct approaches to extract efficiency from CRISPRi screens — directly modeling and removing gene effects using a mixed-effect random forest (MERF) (Hajjem et al., 2014), and heuristically subtracting median values for each gene from guide depletion values as described previously (Calvo-Villamañán et al., 2020). To our surprise, both approaches produced models with roughly similar performance and appeared to identify largely similar feature sets driving guide silencing efficiency. It is possible that the richer description of gene effects enabled by the MERF may become more important when incorporating data from additional screens in other conditions, or expanding beyond

using only essential genes for training. The MERF can also infer its own normalization between data sets as part of its random effect inference, which will greatly simplify training of more complex models.

While we focused here on applications of CRISPRi with dCas9 in *E. coli*, the techniques we have developed are in principle generic and could be extended to CRISPRi with any catalytically-dead nuclease in any bacterium of interest, or even to entirely different CRISPR systems. For instance, we recently applied the same basic methodology to investigate the features underlying autoimmune activation of Cas13 targeting cellular RNA (Vialetto et al., 2021). It is becoming increasingly clear that the performance of CRISPRi depends on both genetic background and the specific Cas protein used. For instance, *Streptococcus pyogenes* dCas9 expression has low silencing efficiency in some bacteria and can even be toxic (Cho et al., 2018b), forcing the adoption of alternative Cas proteins (Rock et al., 2017). Alternative Cas proteins have large differences in their PAM preferences and the stringency of the PAM requirement (Collias and Beisel, 2021); presumably, alternative dCas proteins may also respond differently to the other gene and guide features described here. The approach outlined here, applying autoML and explainable AI to rapidly arrive at a description of the design rules underlying the efficiency of CRISPRi silencing, provides a means to rapidly characterize the behavior of new dCas proteins as genome-wide screening data becomes available.

Supplementary Figures

Figure S1

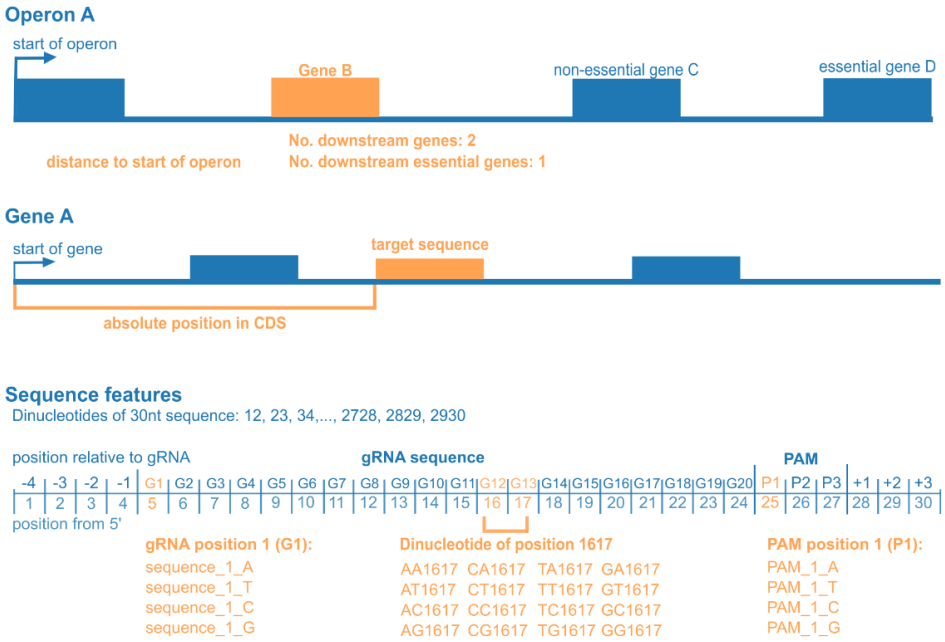

Figure S1: Illustration of the genomic and sequence features used, see also Table S1.

Figure S2

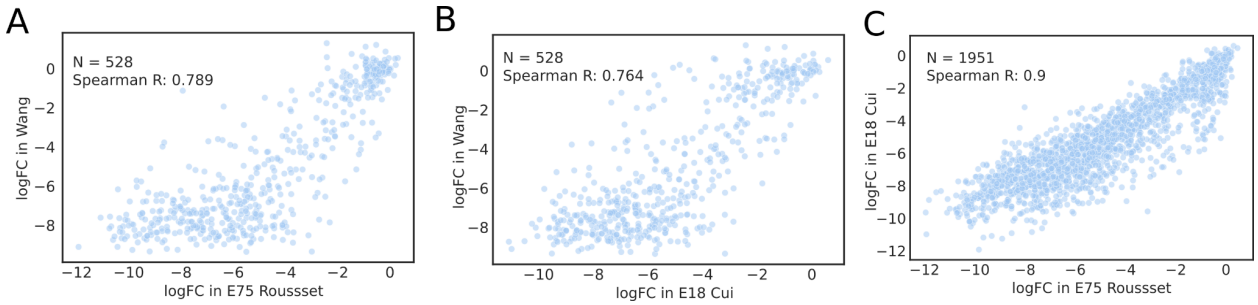

Figure S2: Comparison of guide depletion across datasets (**A**) The logFC of gRNAs in E75 Rousset plotted against that in Wang for shared gRNAs. (**B**) The logFC of gRNAs in E18 Cui was plotted against that in Wang for shared gRNAs. (**C**) The logFC of gRNAs in E18 Cui was plotted against that in E75 Rousset for overlapping gRNAs.

Figure S3

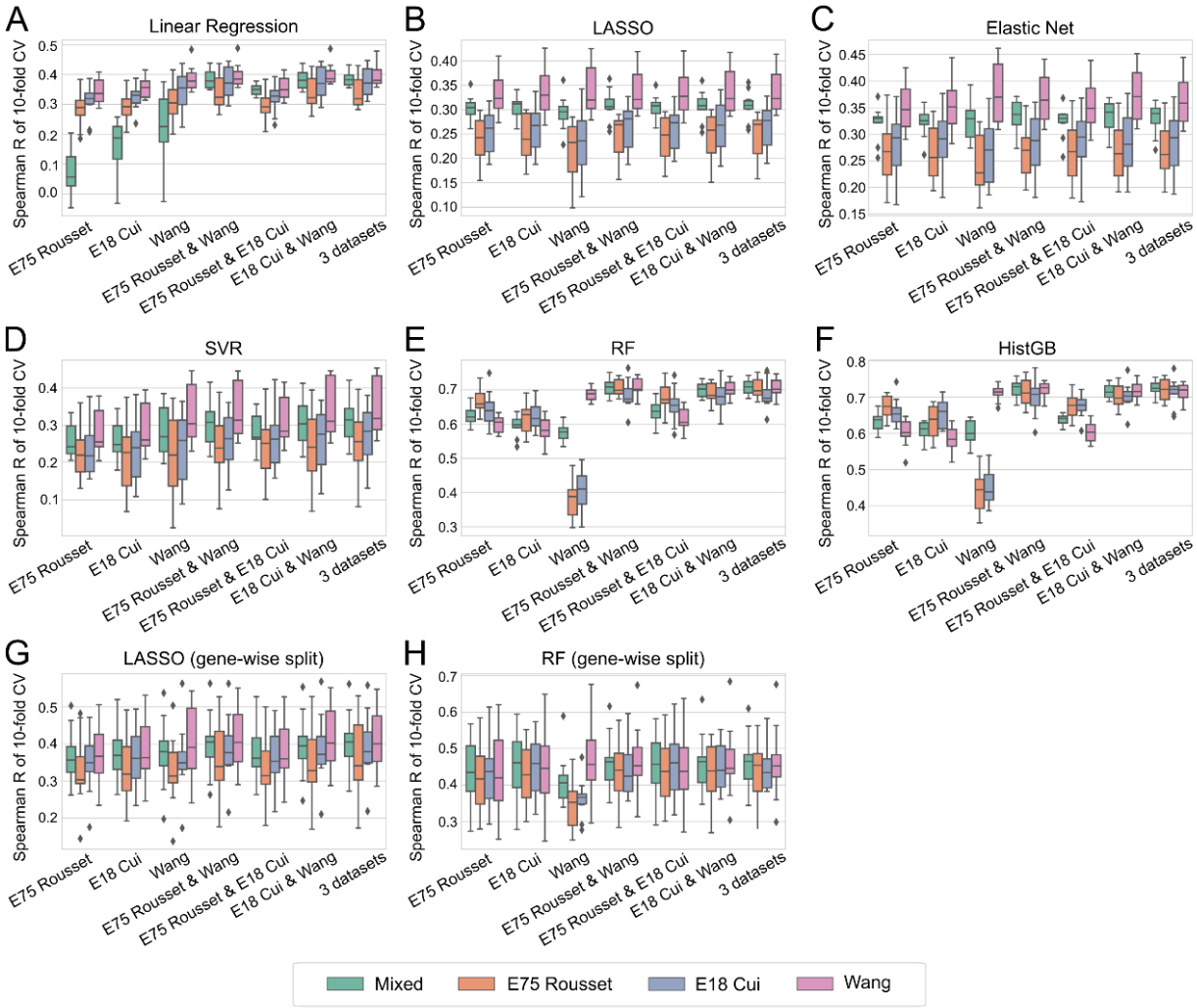

Figure S3: Spearman correlation of 10-fold cross-validation of models trained with one or mixed datasets. (A) linear regression, (B) LASSO, (C) Elastic net, (D) support vector regression (SVR), (E) Random forest (RF) regression, (F) Histogram-based gradient boosting regression. (G) LASSO (same hyperparameters as the MS LASSO model, gene-wise split). (H) Random forest (same hyperparameters as the MS random forest model, gene-wise split).

Figure S4

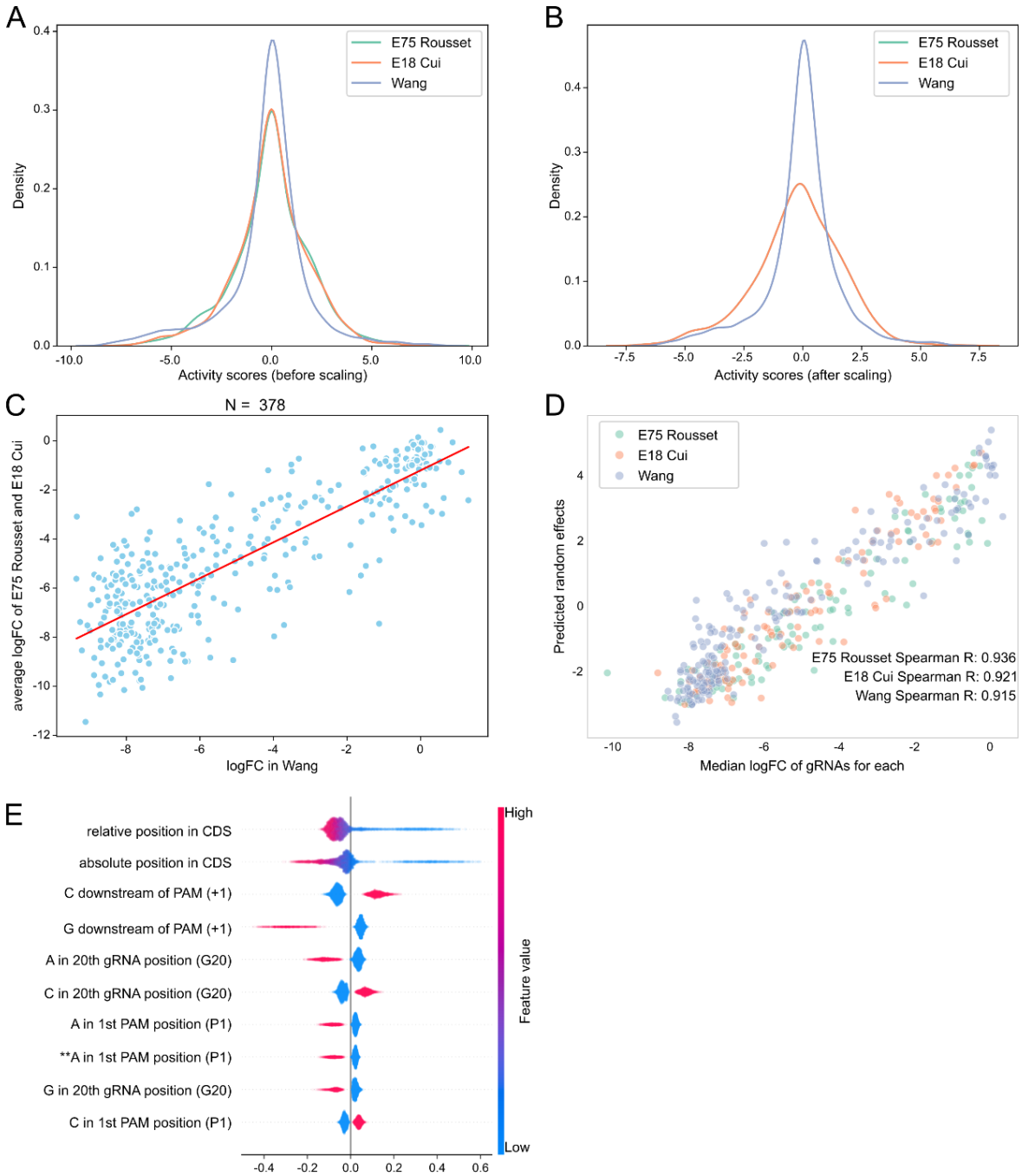

Figure S4: MS models for segregation of gene and guide effects. We subtract the gene-wise median logFC from each gRNA depletion value upon data fusion to obtain the activity scores of each gRNA. The distributions of activity scores (**A**) with and (**B**) without scaling are shown. (**C**) The logFC values in Wang were scaled based on the linear regression between the original logFC of Wang and the average logFC of E75 Rousset and E18 Cui for the 378 overlapping gRNAs. (**D**) Predicted scores of the random effect model from MERF (y-axis) compared to the median logFC across gRNAs (x-axis) for each gene in each dataset. (**E**) SHAP values for the top ten features in the MS random forest model.

Figure S5

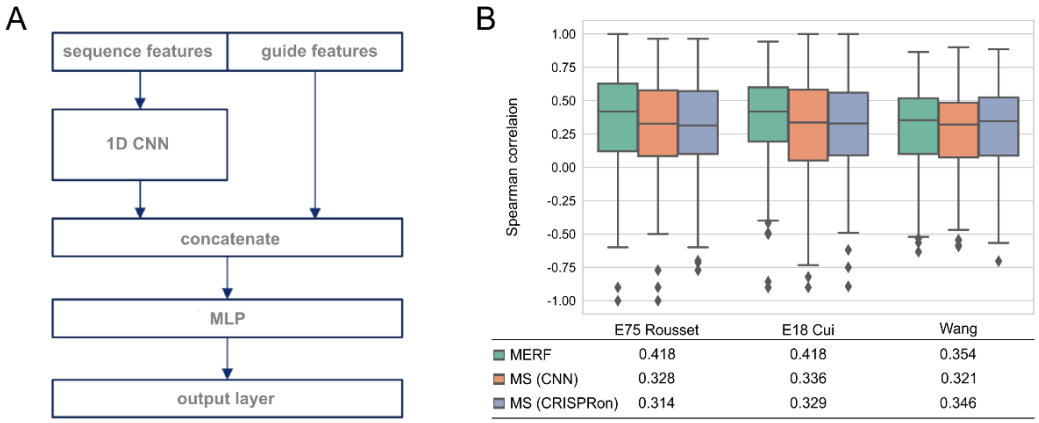

Figure S5: Deep learning approaches do not improve prediction performance. (**A**) Architectures of the applied deep learning models. Guide features refer to guide-specific features apart from sequence features. MLP: multilayer perceptron. (**B**) The Spearman correlation between predictions and measured logFC for each held-out gene.

Figure S6

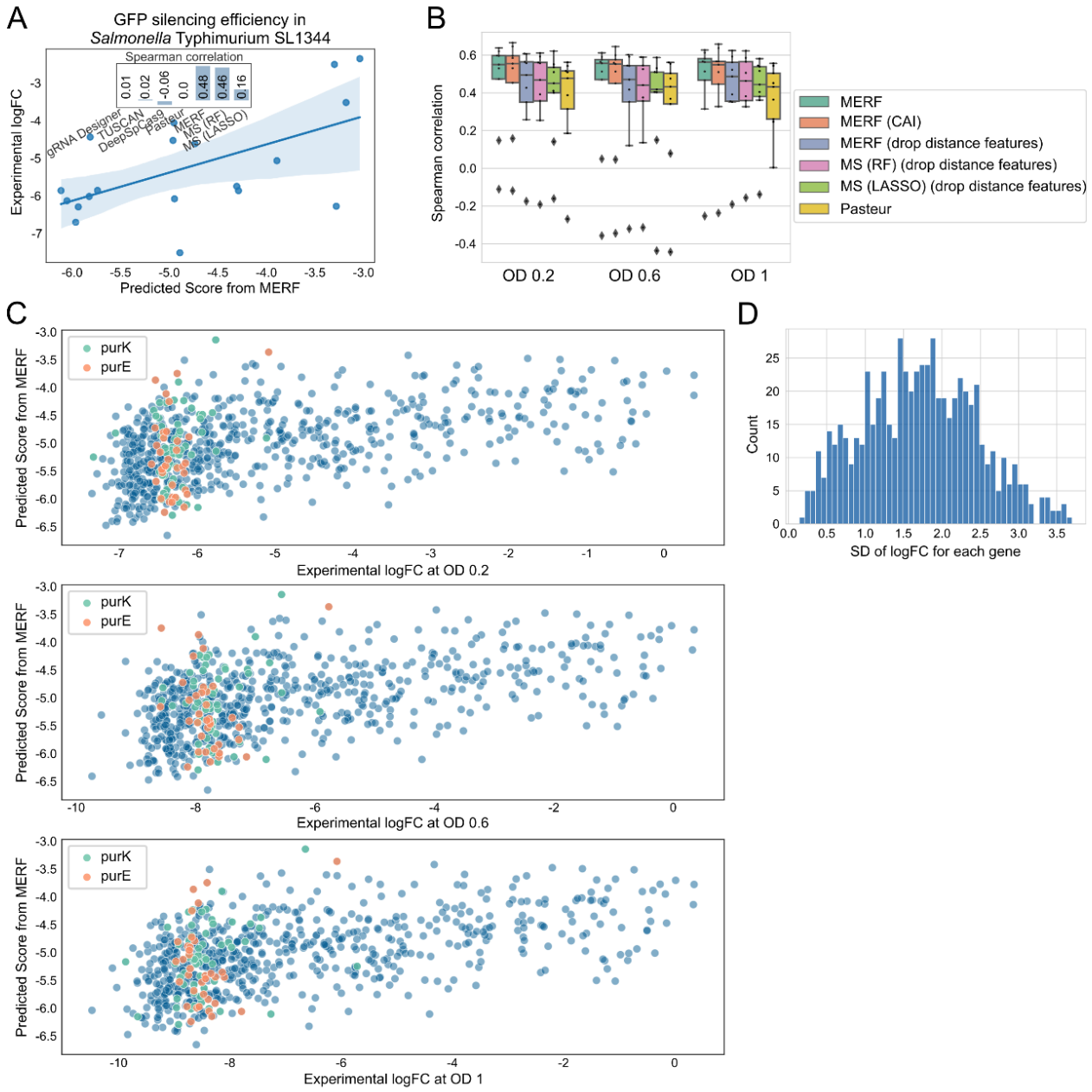

Figure S6: Additional figures related to model validation and a saturating screen of purine biosynthesis genes. (**A**) The activity of 19 gRNAs targeting deGFP gene was measured in *Salmonella* Typhimurium SL1344 using flow-cytometry-based assay. The main panel compares measured logFC to predictions from the MERF model, while the inset summarizes Spearman correlations of a similar comparison between 7 methods (**B**) Performance on the purine screen of MERF random forest models trained with individual or fused datasets, without distance features (drop distance features), and with CAI values. (**C**) The predicted scores from the MERF random forest model were plotted against experimental logFC in different time points. Guides targeting *purE* and *purK* were marked with orange and green

respectively. (**D**) The distribution of the standard deviation of the logFC for guides targeting each gene in the training data.

Methods

Training datasets

We collected the data from three previous CRISPRi genome-wide essentiality screens in *E.coli* K12 MG1655 (Cui et al., 2018; Rousset et al., 2018; Wang et al., 2018). The sequence, targeted gene, gene position, and fitness effect of each gRNA was retrieved from the supplementary information of each study. Gene sequences and positions were updated to be consistent with the latest reference genome version (NC_000913.3). We discarded gRNAs from the Wang data set previously removed as having insufficient read counts (Wang et al., 2018) or sequences from the Rousset and Cui datasets that differed from the reference sequence due to differences in the genome versions. 8099 gRNAs targeting the coding-strand within the coding regions of essential genes were extracted in total from all three datasets. Genes targeted by fewer than 5 gRNAs were removed.

Feature engineering

A Python script (feature_engineering.py) was used to compute 574 sequence, thermodynamic, genomic, and transcriptomic features. Sequence features including 556 single-nucleotide and dinucleotide features were one-hot encoded. Thermodynamic features including minimum free energy for different interactions were computed using the ViennaRNA Package (Lorenz et al., 2011): RNAduplex (version 2.4.12) for RNA:RNA hybrids; RNAduplex (version 2.1.9h) for DNA:RNA hybrids (Lorenz et al., 2012); RNAfold (version 2.4.12) for single RNA folding. Genomic features including gene and operon organizations were based on the reference genome, essential genes as determined in the Keio collection (Baba et al., 2006), and transcriptional unit definitions from RegulonDB (Tierrafría et al., 2022). Transcriptomic data including gene expression levels across growth at ten different ODs were obtained from a previous study (Conway et al., 2014). Minimal or maximal expression levels were calculated across the range of ODs until the growth phase when cells were collected in each

CRISPRi screen: OD 1.4 for the Wang dataset, and all ODs for the Rousset and Cui datasets. The codon adaptation index (CAI) for each gene was calculated using CAIcal (Puigbò et al., 2008).

Applying machine learning methods

The automated machine learning toolkits auto-sklearn (version 0.10.0) (Feurer et al., 2015) and H2O (version 3.30.1) (LeDell and Poirier, 2020) were used to develop optimized machine learning regression models. For auto-sklearn, all possible estimators were included. The following parameters were used: “time_left_for_this_task” = 3600, “per_run_time_limit” = 360, “resampling_strategy”= ‘cv’, and “resampling_strategy_arguments” = {“fold”: 10}, “metric” = autosklearn.metrics.mean_squared_error. The selected model parameters were saved and used with scikit-learn for downstream analysis. For H2O, the “StackedEnsemble” algorithm was excluded and parameters “max_runtime_secs = 0” and “seed = 1” were used. If not otherwise specified, parameters were left as default. Simple linear regression, LASSO, elastic net, SVR, random forest, and histogram-based gradient boosting models were trained using scikit-learn (version 0.22.2) (Pedregosa et al., 2011).

Deep learning models were trained using pytorch (version 1.8.1) (Paszke et al., 2019) and pytorch-lightning (version 1.5.10). For our custom 1D CNN model, sequence features were processed using 1D convolutional layers and later concatenated with other guide features. Concatenated features were further processed with fully connected layers. Three 1D convolutional layers were implemented sequentially with input channels 4, 64, and 64, output channel 64, 64, and 32, kernel size 5, 3, and 1, and stride 2, 2, and 1 respectively. For fully connected layers, output dimensions are 128, 64, 32 and 1 (which is predicted gRNA efficiency). The first three fully connected layers are accompanied by batch normalization (Ioffe and Szegedy, 2015), ReLU and dropout (Srivastava et al., 2014) (p=0.5). We trained the model using AdamW (Loshchilov and Hutter, 2017) optimiser with learning rate of 0.001 and batch size of 32. For CRISPROn, we followed the same architecture (CGx) with different numbers of non-sequential

features concatenated to processed sequential features. We trained the model using the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.001 and batch size of 32. Models were trained and evaluated with 10-fold cross-evaluation based on the gRNA sequences to predict gRNA depletion.

Tree-based models were interpreted using TreeExplainer from the python shap package (version 0.39.0) (Lundberg et al., 2020). SHAP value plots were generated with the 'summary_plot' function in shap.

Segregation of guide and gene effects

We removed genes with less than 5 gRNAs in each dataset to stabilize estimates of median gRNA activity scores (see below), resulting in 7400 gRNAs in total. This included 1618 gRNAs targeting 171 genes in E75 Rousset/E18 Cui and 4164 targeting 300 genes in Wang.

MERF models were trained using package merf (version 1.0). Hyperparameters for the fixed-effect random forest model were taken from auto-sklearn. 564 guide-specific features were assigned as fixed effects, while 9 gene-specific features except gene ID were assigned as random effects. 301 unique gene IDs were used as cluster IDs. The trained fixed-effect model was used to predict gRNA efficiency. To train simplified models excluding transcriptomic measurements (**Figure S6B**), CAI value, gene length, gene GC content, and dataset were included for the random-effect model.

For median subtracting (MS) models, logFC values were scaled to integrate the datasets, as an adaptation of a previously applied data fusion method (Xiang et al., 2021). First, the mean of logFCs of E75 Rousset and E18 Cui were calculated and used as the scaled logFC (**Figure S4A&B**). Then linear regression was performed between the logFCs in Wang and scaled logFCs in E75 Rousset for 378 overlapping gRNAs. All of the logFCs from Wang were then scaled by the fitted slope and intercept (**Figure S4C**). The 378 overlapping gRNAs in Wang were excluded in the subsequent training

for MS models. Activity scores were calculated by subtracting the scaled logFC of each gRNA from median scaled logFC for each gene across all 3 datasets.

MS models were trained with guide-specific features to predict activity scores for each gRNA. The hyperparameters of the MS random forest model were the same as for MERF, while those of the LASSO model were optimized using hyperopt (version 0.2.5) (Bergstra et al., 2013) with a search space for alpha ranging from 0 to 0.1. The trained models were directly used to predict activity scores.

Training and test sets were split gene-wise based on gene identifier. 10-fold cross-validation was used to evaluate model performance.

The fixed-effect model from MERF and the MS (RF) model were interpreted using the shap package (Lundberg et al., 2020). SHAP interaction values were calculated using the shap_interaction_values function in TreeExplainer with 1000 guides. Absolute SHAP interaction values were averaged over 1000 samples. The rank of interaction was obtained based on the sorted mean absolute SHAP interaction values across all unique feature pairs. To compare interaction effects to expectations based on single-feature SHAP values, four feature combinations were considered: both absent (-/-), only the first feature present (+/-), only the second feature present (-/+), and both present (+/+). For the top 5,000 interacting feature pairs, the SHAP values for each feature in samples with each combination of features were extracted. For each feature pair (F1 and F2), the expected value for +/+ was calculated as the sum of the median F1 SHAP values for +/- samples with the median of F2 SHAP values for -/+ samples, while the expected value for -/- was calculated as the sum of the median F1 SHAP values for -/+ samples and the median of F2 SHAP values for +/- samples.

Strains and growth conditions

All strains, plasmids, and primers are listed in Supplementary **Table S14 and S15**. *E. coli* cells were grown in Lysogeny Broth (LB) (10 g/L NaCl, 5 g/L yeast extract, 10 g/L

tryptone) at 37 °C with shaking at 250 rpm. To maintain plasmids, the antibiotics ampicillin, chloramphenicol, and/or kanamycin were added at 50 µg/mL, 34 µg/mL, and 50 µg/mL, respectively as necessary. For screening experiments, *E. coli* MG1655 was grown in M9 minimal medium (1x M9 salts, 1 mM thiamine hydrochloride, 0.4% glucose, 0.2% casamino acids, 2 mM MgSO4, 0.1 mM CaCl2) supplemented with the appropriate antibiotics.

Validation of GFP silencing by flow cytometry in *E. coli* and *S.* Typhimurium

To investigate gene repression efficiency, 19 sgRNAs were selected to target the coding strand of a *degfp* reporter gene at different positions in *E. coli* BL21(DE3) (**Table S9**). Cells were initially transformed with three compatible plasmids encoding dCas9, a *degfp*-targeting sgRNA, and a deGFP reporter (**Table S16**). For normalization purposes, a positive control strain harboring a non-targeting sgRNA and a negative control strain lacking the *degfp* encoding reporter plasmid was included. Overnight cultures of cells harboring the above-mentioned plasmids were back-diluted to OD_{600} ~0.01 in LB medium with ampicillin, chloramphenicol and/or kanamycin and incubated with shaking at 250 rpm at 37 °C, until reaching an OD_{600} of 1. Cultures were then diluted 1:25 in 1x phosphate-buffered saline (PBS) and analyzed on an Accuri C6 flow cytometer with C6 sampler plate loader (Becton Dickinson) equipped with CFlow plate sampler, a 488-nm laser, and a 530+/− 15-nm bandpass filter. Forward scatter (cutoff of 11,500) and side scatter (cutoff of 600) were used to eliminate non-cellular events. The mean green fluorescence value (measured by the FL1-H channel) across 30,000 events within a gate set for *E. coli* was used for further analysis. The log fold repression of each gRNA was calculated as the ratio between the difference in fluorescence values between the gRNA and negative control, and the difference between the positive and the negative control, followed by log transformation. The mean log fold repression across three replicates was compared to predicted values from the machine learning models (**Table S11**).

For experiments with *S.* Typhimurium, the procedure was similar, but cells were grown until an OD_{600} of ~0.8 before analysis on an Accuri C6 flow cytometer. To eliminate non-cellular events, the forward scatter (cutoff of 10,000) was used and the mean green fluorescence value (FL1-H) across 30,000 events within a gate set for *S.* Typhimurium was used for data analysis as described above across four replicates.

Generation of the sgRNA library

For the sgRNA library targeting the purine biosynthesis pathway in *E. coli* MG1655, plasmid DC512 served as a backbone, following a previously established protocol (Liao et al., 2019). To generate a library with 800 sgRNAs (including 50 non-targeting sgRNAs; **Table S12**), 800 forward and reverse oligonucleotides each encoding one spacer and a 4-nt junction, were synthesized as an oPool (1600 oligos at 10pmol/oligo) by Integrated Device Technology (IDT). The same 5′ and 3′ assembly junction sequences were used for all spacer pairs leading to the same integration site within the backbone (5′ TAGT overhang at the 5′ end and a 3′ AAAC overhang at the 3′ end). Supplementary **Table S16** contains the specific oligonucleotides and assembly junctions used for the library generation. The oligos were phosphorylated and annealed to form dsDNA with a 5′ and 3′ overhang. The steps of phosphorylation and annealing were combined and conducted in one pot, by adding 8,000 fmol of the oPool and 1 µl T4 polynucleotide kinase (10 units) to 5 µl 10x T4 ligation buffer and then, adding water until reaching a final volume of 50 µl. After mixing briefly by pipetting the mix was incubated at 37°C for 30 minutes in a thermocycler and then incubated at 65°C for 20 minutes in a thermocycler to heat-inactivate the kinase. For the annealing of the forward and reverse oligo pairs, the following thermocycler steps were added: 95°C for 5 min, 94°C for 15 s, decrease by 1°C, and hold for 30 seconds for 79 cycles. For integrating the dsDNA inserts into DC512, 400 fmol of the dsDNA, 20 fmol of backbone plasmid, 0.5 µL of T4 ligase (1000 units), and 1.5 µL of BbsI (15 units) were added to 2 µL of 10x T4 ligation buffer, then water was added to reach a total volume of 20 µl. A thermocycler was used to perform 35 cycles of digestion and ligation (37 °C for 2 min, 16 °C for 5 min) followed by a final digestion step (60 °C for 10 min) and a heat inactivation step (80 °C

for 10 min). After NdeI digestion (37°C, 1h) of the ligation mix to remove any remaining original backbone plasmids and subsequent ethanol precipitation, 10 µl of the ligation mix was transformed into electrocompetent *E. coli* NEB10 beta (NEB, Ipswich, MA, USA), following the manufacturer's instructions. After transformation and recovery in 1 ml SOC for 1 h at 37 °C with shaking at 250 rpm, different dilutions of the recovered cells were plated on LB agar containing the appropriate antibiotic and incubated for 16 h to check the number and color of the resulting colonies (ensuring a ~58X coverage). The rest of the recovered culture was added to 100 mL LB media containing the appropriate antibiotic and incubated at 37 °C with shaking at 250 rpm to $OD_{600} \approx 1$. Cells were harvested by centrifugation and subjected to plasmid extraction. Sanger sequencing was used to validate the library plasmid DNA.

Screening experiment

E. coli strain MG1655 was initially transformed with a dCas9 encoding plasmid (2.0 kV, 200 Omega, and 25 µF). The resulting strain SG332 was then transformed with the sgRNA library by electroporation and recovered in 900 µl SOC for 1.5h at 37 °C with shaking at 250 rpm. Different dilutions of the recovered cells were plated on LB agar containing the appropriate antibiotics and incubated for 16 h to check the number of the resulting colonies (~56^5 colonies). The recovered culture was back-diluted to OD_{600} 0.01 in LB medium with appropriate antibiotics and incubated at 37 °C with shaking for 13h. Subsequently, 5 mL of the culture was sampled and the library was extracted by miniprep (Nucleospin Plasmid, Macherey-Nagel) to obtain the initial sgRNA distribution. The calculated amount of culture to reach OD_{600} 0.01 in 50 ml M9 minimal medium, was sampled and washed twice with M9 minimal medium to remove traces of the LB medium. The culture was incubated at 37°C with shaking until it reached OD_{600} 1, allowing ~6 replications. 5 ml of the culture was sampled at OD_{600} 0.2 and OD_{600} 0.6 and at OD_{600} 1 and the library was extracted by miniprep. The experiment was performed in duplicate starting from two independent transformations of MG1655 with the plasmid library.

Library sequencing

The sequencing library was generated using the KAPA HiFi HotStart Library Amplification Kit for Illumina® platforms (Roche) and the primers listed in Supplementary Table S16. The first PCR adds the first index. The second PCR adds the second index and flow cell-binding sequence. The amplicons of the first and second PCR reactions were purified using solid-phase reversible immobilization beads (AMPure XP, Beckman Coulter) following the manufacturer's instructions to remove excess primers and possible primer dimers. The sequencing library samples, with the required DNA concentrations ranging from 100 pg - 200 ng in a total volume of 10 µL, were submitted to the HZI NGS sequencing facility (Braunschweig, Germany) for paired-end 2 × 50 bp deep sequencing with 800,000 reads per sample on a NovaSeq 6000 sequencer.

Sequencing data processing

Paired-end reads were merged using BBMerge (version 38.69) with parameters "qtrim2=t, ecco, trimq=20, -Xmx1g". Merged reads with perfect matches were assigned to the gRNA library using a Python script. After filtering guides for at least 1 count per million in at least 4 samples, read counts of each gRNA were normalized by factors derived from non-targeting guides using the trimmed mean of m-values method in edgeR (version 3.28.0) (Robinson et al., 2009). An extra column was added to the design matrix to capture batch effects between the two replicate experiments. Differential abundance (logFC) of gRNAs between time points and the input library were estimated using edgeR, and a quasi-likelihood F test was used to test for significance after fitting in a generalized linear model.

Code and data availability

All code necessary to reproduce this results in the manuscript are available at: https://github.com/BarquistLab/CRISPRi_guide_efficiency_bacteria. Raw sequencing data for the CRISPRi purine screen has been deposited in GEO under accession GSE196911. A webserver implementation of the final MERF model is available at: http://ciao.helmholtz-hiri.de.

References

Baba, T., Ara, T., Hasegawa, M., and Takai, Y. (2006). Construction of Escherichia coli K-12 in-frame, single-gene knockout mutants: the Keio collection. Mol. Syst. Biol.

Bergstra, J., Yamins, D., and Cox, D. (2013). Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. In Proceedings of the 30th International Conference on Machine Learning, S. Dasgupta, and D. McAllester, eds. (Atlanta, Georgia, USA: PMLR), pp. 115–123.

Bikard, D., Jiang, W., Samai, P., Hochschild, A., Zhang, F., and Marraffini, L.A. (2013). Programmable repression and activation of bacterial gene expression using an engineered CRISPR-Cas system. Nucleic Acids Res. *41*, 7429–7437. .

Bikard, D., Euler, C.W., Jiang, W., Nussenzweig, P.M., Goldberg, G.W., Duportet, X., Fischetti, V.A., and Marraffini, L.A. (2014). Exploiting CRISPR-Cas nucleases to produce sequence-specific antimicrobials. Nat. Biotechnol. *32*, 1146–1150. .

Butland, G., Babu, M., Díaz-Mejía, J.J., Bohdana, F., Phanse, S., Gold, B., Yang, W., Li, J., Gagarinova, A.G., Pogoutse, O., et al. (2008). eSGA: E. coli synthetic genetic array analysis. Nat. Methods *5*, 789–795. .

Cain, A.K., Barquist, L., Goodman, A.L., Paulsen, I.T., Parkhill, J., and van Opijnen, T. (2020). A decade of advances in transposon-insertion sequencing. Nat. Rev. Genet. https://doi.org/10.1038/s41576-020-0244-x.

Calvo-Villamañán, A., Ng, J.W., Planel, R., Ménager, H., Chen, A., Cui, L., and Bikard, D. (2020). On-target activity predictions enable improved CRISPR-dCas9 screens in bacteria. Nucleic Acids Res. https://doi.org/10.1093/nar/gkaa294.

Cho, S., Shin, J., and Cho, B.-K. (2018a). Applications of CRISPR/Cas System to Bacterial Metabolic Engineering. Int. J. Mol. Sci. *19*. https://doi.org/10.3390/ijms19041089.

Cho, S., Choe, D., Lee, E., Kim, S.C., Palsson, B.Ø., and Cho, B.-K. (2018b). High-level dCas9 expression induces abnormal cell morphology in Escherichia coli. ACS Synth. Biol. https://doi.org/10.1021/acssynbio.7b00462.

Chuai, G., Ma, H., Yan, J., Chen, M., Hong, N., Xue, D., Zhou, C., Zhu, C., Chen, K., Duan, B., et al. (2018). DeepCRISPR: optimized CRISPR guide RNA design by deep learning. Genome Biol. *19*, 80. .

Citorik, R.J., Mimee, M., and Lu, T.K. (2014). Sequence-specific antimicrobials using efficiently delivered RNA-guided nucleases. Nat. Biotechnol. *32*, 1141–1145. .

Collias, D., and Beisel, C.L. (2021). CRISPR technologies and the search for the PAM-free nuclease. Nat. Commun. *12*, 555. .

Conway, T., Creecy, J.P., Maddox, S.M., Grissom, J.E., Conkle, T.L., Shadid, T.M., Teramoto, J., San Miguel, P., Shimada, T., Ishihama, A., et al. (2014). Unprecedented high-resolution view of bacterial operon architecture revealed by RNA sequencing. MBio *5*, e01442–14. .

Cui, L., Vigouroux, A., Rousset, F., Varet, H., Khanna, V., and Bikard, D. (2018). A CRISPRi screen in E. coli reveals sequence-specific toxicity of dCas9. Nat. Commun. *9*, 1912. .

Doench, J.G., Hartenian, E., Graham, D.B., Tothova, Z., Hegde, M., Smith, I., Sullender, M., Ebert, B.L., Xavier, R.J., and Root, D.E. (2014). Rational design of highly active sgRNAs for CRISPR-Cas9-mediated gene inactivation. Nat. Biotechnol. *32*, 1262–1267. .

Doench, J.G., Fusi, N., Sullender, M., Hegde, M., Vaimberg, E.W., Donovan, K.F., Smith, I., Tothova, Z., Wilen, C., Orchard, R., et al. (2016). Optimized sgRNA design to maximize activity and minimize off-target effects of CRISPR-Cas9. Nat. Biotechnol. *34*, 184–191. .

Feurer, M., Klein, A., Eggensperger, K., Springenberg, J., Blum, M., and Hutter, F. (2015). Efficient and Robust Automated Machine Learning. In Advances in Neural Information Processing Systems 28, C. Cortes, N.D. Lawrence, D.D. Lee, M. Sugiyama, and R. Garnett, eds. (Curran Associates, Inc.), pp. 2962–2970.

Gomaa, A.A., Klumpe, H.E., Luo, M.L., Selle, K., Barrangou, R., and Beisel, C.L. (2014). Programmable removal of bacterial strains by use of genome-targeting CRISPR-Cas systems. MBio *5*, e00928–13. .

Hajjem, A., Bellavance, F., and Larocque, D. (2014). Mixed-effects random forest for clustered data. J. Stat. Comput. Simul. *84*, 1313–1328. .

Ioffe, S., and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift.

Jusiak, B., Cleto, S., Perez-Piñera, P., and Lu, T.K. (2016). Engineering Synthetic Gene Circuits in Living Cells with CRISPR Technology. Trends Biotechnol. *34*, 535–547. .

Kim, H.K., Min, S., Song, M., Jung, S., Choi, J.W., Kim, Y., Lee, S., Yoon, S., and Kim, H.H. (2018). Deep learning improves prediction of CRISPR-Cpf1 guide RNA activity. Nat. Biotechnol. *36*, 239–241. .

Kim, H.K., Kim, Y., Lee, S., Min, S., Bae, J.Y., Choi, J.W., Park, J., Jung, D., Yoon, S., and Kim, H.H. (2019). SpCas9 activity prediction by DeepSpCas9, a deep learning-based model with high generalization performance. Sci Adv *5*, eaax9249. .

Kingma, D.P., and Ba, J. (2014). Adam: A Method for Stochastic Optimization.

Kuzmin, E., VanderSluis, B., Wang, W., Tan, G., Deshpande, R., Chen, Y., Usaj, M., Balint, A., Mattiazzi Usaj, M., van Leeuwen, J., et al. (2018). Systematic analysis of complex genetic interactions. Science *360*. https://doi.org/10.1126/science.aao1729.

Labun, K., Montague, T.G., Gagnon, J.A., Thyme, S.B., and Valen, E. (2016). CHOPCHOP v2: a web tool for the next generation of CRISPR genome engineering. Nucleic Acids Res. *44*, W272–W276. .

LeDell, E., and Poirier, S. (2020). H2o automl: Scalable automatic machine learning. In Proceedings of the AutoML Workshop at ICML, (automl.org),.

Lian, J., HamediRad, M., Hu, S., and Zhao, H. (2017). Combinatorial metabolic engineering using an orthogonal tri-functional CRISPR system. Nat. Commun. *8*, 1688. .

Liao, C., Ttofali, F., Slotkowski, R.A., Denny, S.R., Cecil, T.D., Leenay, R.T., Keung, A.J., and Beisel, C.L. (2019). Modular one-pot assembly of CRISPR arrays enables library generation and reveals factors influencing crRNA biogenesis. Nat. Commun. *10*, 2948. .

Lorenz, R., Bernhart, S.H., Höner Zu Siederdissen, C., Tafer, H., Flamm, C., Stadler, P.F., and Hofacker, I.L. (2011). ViennaRNA Package 2.0. Algorithms Mol. Biol. *6*, 26. .

Lorenz, R., Hofacker, I.L., and Bernhart, S.H. (2012). Folding RNA/DNA hybrid duplexes. Bioinformatics *28*, 2530–2531. .

Loshchilov, I., and Hutter, F. (2017). Decoupled Weight Decay Regularization.

Lundberg, S.M., Erion, G., Chen, H., DeGrave, A., Prutkin, J.M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., and Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. Nature Machine Intelligence *2*, 56–67. .

Luo, M.L., Leenay, R.T., and Beisel, C.L. (2016). Current and future prospects for CRISPR-based tools in bacteria. Biotechnol. Bioeng. *113*, 930–943. .

Moreno-Mateos, M.A., Vejnar, C.E., Beaudoin, J.-D., Fernandez, J.P., Mis, E.K., Khokha, M.K., and Giraldez, A.J. (2015). CRISPRscan: designing highly efficient sgRNAs for CRISPR-Cas9 targeting in vivo. Nat. Methods *12*, 982–988. .

Mougiakos, I., Bosma, E.F., Ganguly, J., van der Oost, J., and van Kranenburg, R. (2018). Hijacking CRISPR-Cas for high-throughput bacterial metabolic engineering: advances and prospects. Curr. Opin. Biotechnol. *50*, 146–157. .

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res. *12*, 2825–2830. .

Puigbò, P., Bravo, I.G., and Garcia-Vallve, S. (2008). CAIcal: a combined set of tools to assess codon usage adaptation. Biol. Direct *3*, 38. .

Qi, L.S., Larson, M.H., Gilbert, L.A., Doudna, J.A., Weissman, J.S., Arkin, A.P., and Lim, W.A. (2013). Repurposing CRISPR as an RNA-guided platform for sequence-specific control of gene expression. Cell *152*, 1173–1183. .

Reis, A.C., Halper, S.M., Vezeau, G.E., Cetnar, D.P., Hossain, A., Clauer, P.R., and Salis, H.M. (2019). Simultaneous repression of multiple bacterial genes using nonrepetitive extra-long sgRNA arrays. Nat. Biotechnol. https://doi.org/10.1038/s41587-019-0286-9.

Robinson, M.D., McCarthy, D.J., and Smyth, G.K. (2009). edgeR: A Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics *26*, 139–140. .

Rock, J.M., Hopkins, F.F., Chavez, A., Diallo, M., Chase, M.R., Gerrick, E.R., Pritchard, J.R., Church, G.M., Rubin, E.J., Sassetti, C.M., et al. (2017). Programmable transcriptional repression in mycobacteria using an orthogonal CRISPR interference platform. Nat Microbiol *2*, 16274. .

Rousset, F., Cui, L., Siouve, E., Becavin, C., Depardieu, F., and Bikard, D. (2018). Genome-wide CRISPR-dCas9 screens in E. coli identify essential genes and phage host factors. PLoS Genet. *14*, e1007749. .

Santos-Zavaleta, A., Salgado, H., Gama-Castro, S., Sánchez-Pérez, M., Gómez-Romero, L., Ledezma-Tejeida, D., García-Sotelo, J.S., Alquicira-Hernández, K., Muñiz-Rascado, L.J., Peña-Loredo, P., et al. (2019). RegulonDB v 10.5: tackling challenges to unify classic and high throughput knowledge of gene regulation in E. coli K-12. Nucleic Acids Res. *47*, D212–D220. .

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. *15*, 1929–1958. .

Tierrafría, V.H., Rioualen, C., Salgado, H., Lara, P., Gama-Castro, S., Lally, P., Gómez-Romero, L., Peña-Loredo, P., López-Almazo, A.G., Alarcón-Carranza, G., et al. (2022). RegulonDB 11.0: Comprehensive high-throughput datasets on transcriptional regulation in Escherichia coli K-12. Microb Genom *8*. https://doi.org/10.1099/mgen.0.000833.

Typas, A., Nichols, R.J., Siegele, D.A., Shales, M., Collins, S.R., Lim, B., Braberg, H., Yamamoto, N., Takeuchi, R., Wanner, B.L., et al. (2008). High-throughput, quantitative analyses of genetic interactions in E. coli. Nat. Methods *5*, 781–787. .

Vialetto, E., Yu, Y., Collins, S.P., Wandera, K.G., Barquist, L., and Beisel, C.L. (2021). A target expression

threshold dictates invader defense and autoimmunity by CRISPR-Cas13.

Vigouroux, A., and Bikard, D. (2020). CRISPR Tools To Control Gene Expression in Bacteria. Microbiol. Mol. Biol. Rev. *84*, e00077–19. .

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat. Methods *17*, 261–272. .

Wang, D., Zhang, C., Wang, B., Li, B., Wang, Q., Liu, D., Wang, H., Zhou, Y., Shi, L., Lan, F., et al. (2019). Optimized CRISPR guide RNA design for two high-fidelity Cas9 variants by deep learning. Nat. Commun. *10*, 4284. .

Wang, T., Guan, C., Guo, J., Liu, B., Wu, Y., Xie, Z., Zhang, C., and Xing, X.-H. (2018). Pooled CRISPR interference screening enables genome-scale functional genomics study in bacteria with superior performance. Nat. Commun. *9*, 2475. .

Wilson, L.O.W., Reti, D., O'Brien, A.R., Dunne, R.A., and Bauer, D.C. (2018). High Activity Target-Site Identification Using Phenotypic Independent CRISPR-Cas9 Core Functionality. CRISPR J *1*, 182–190. .

Wong, N., Liu, W., and Wang, X. (2015). WU-CRISPR: characteristics of functional guide RNAs for the CRISPR/Cas9 system. Genome Biol. *16*, 218. .

Xiang, X., Corsi, G.I., Anthon, C., Qu, K., Pan, X., Liang, X., Han, P., Dong, Z., Liu, L., Zhong, J., et al. (2021). Enhancing CRISPR-Cas9 gRNA efficiency prediction by data integration and deep learning. Nat. Commun. *12*, 3238. .