University of Bari Aldo Moro

Department of Computer Science

Ph.D. Program in Computer Science and Mathematics

XXXVIII Cycle

SSD: INFO/01A – Computer Science

Design of Explainable Methods for Predicting Off-Target and On-Target Effects on gRNA Sequences in CRISPR/Cas9 Base Editing

Ph.D. Candidate:

Faiza Hasin

Supervisor: Prof. Corrado Mencar

Co-supervisor: Dr. Arianna Consiglio

Ph.D. Coordinator: Prof. Giovanna Castellano

January 31, 2026

Abstract

This PhD thesis presents a comprehensive investigation into the design of explainable methods for predicting efficiency related to off-target and on-target events on guide RNA (gRNA) sequences in CRISPR/Cas9 base editing. The research was conducted within the project "Development of an innovative platform for the design of highly effective and specific CRISPR screens", led by Tommaso Selmi (CNR Institute for Biomedical Technologies), which is part of the PNRR CN3-Spoke7-WP1 National Center for Gene Therapy and Drugs based on RNA Technology.

Background and Motivation CRISPR/Cas9 has revolutionized genome editing by enabling precise DNA modifications through customizable guide RNA (gRNA) sequences that direct the Cas9–gRNA complex toward complementary genomic loci. Upon recognition of the target sequence, the complex binds to the DNA, allowing site-specific editing. While gene knockout (CRISPR-KO) involves DNA breaks that can cause unpredictable insertions or deletions, base editing (CRISPR-BE) offers more controlled point mutations without DNA breaks. However, off-target effects—where gRNA binds to unintended DNA locations due to sequence similarity—remain a critical challenge that can cause cell malfunction or death. Current predictive models suffer from two major limitations: low accuracy in efficiency prediction and domain specificity that prevents knowledge transfer between knockout and base editing domains.

Research Questions and Methodology The thesis addresses two primary research questions: (RQ1) whether it is possible to improve prediction accuracy while preserving explainability, and (RQ2) to what extent knowledge acquired for knockout prediction can be transferred to base editing domains. The methodological approach involves designing an ensemble framework that integrates four established CRISPR prediction models—CRISPRedict (linear GLM model), CRISPRAidit (MLP with 473 features), CRISPRon (CNN-based architecture), and DeepHF (BiLSTM with biological features)—using XGBoost as a meta-learner. SHAP (SHapley Additive exPlanations) was employed as the primary explainability tool due to its linear representation that facilitates aggregation within the ensemble hierarchy.

Major Contributions The thesis comprises three interconnected studies. The first study aimed at developing a systematic scoping review following PRISMA-ScR guidelines that analyzed 72 CRISPR/Cas9 predictive models published between 2014 and 2023 from an initial pool of 357 records. The review revealed that deep learning models (43.1%) dominate the field, followed by machine learning (17.0%) and ensemble ap-

proaches. Notably, only 7 papers addressed interpretability and 2 addressed explainability explicitly, highlighting a significant gap in transparent predictive modeling. The four models selected for in-depth analysis—CRISPRedict, CRISPRAidit, CRISPRon, and DeepHF—represented diverse computational paradigms while offering different levels of explainability.

The subsequent study presents the core experimental framework evaluating gRNA efficiency prediction through an explainable ensemble. Three experiments were conducted using two datasets: the SpCas9-Endo dataset (79 gRNAs for knockout) and the Hanna dataset (5062 gRNAs for base editing with z-score ground truth values). Experiment 1 demonstrated that the XGBoost ensemble achieved superior performance on knockout data compared to individual models and baseline methods including Linear Regression, MLP, SVR, Decision Trees, and Gaussian Process. Experiment 2 tested direct transfer learning by applying knockout-trained models to base editing data without retraining, resulting in consistently poor performance, definitively showing that knowledge cannot be directly transferred between domains. Experiment 3 involved complete retraining of all base models and the meta-learner on the Hanna dataset using 10-fold cross-validation, yielding substantial improvements: CRISPRon achieved the best individual performance, while the XGBoost ensemble reached an average Pearson correlation.

In parallel to the ensemble framework, the third study investigates the impact of feature representation on off-target event prediction using XGBoost models trained on CHANGE-seq, GUIDE-seq and Hanna datasets. Multiple encoding schemes were compared, including one-hot encoding, k-mers, bulges, and position-specific encodings. Results showed that Bulges encoding achieved the best classification performance, while OH5C encoding performed best in regression tasks. SHAP analysis, combined with Accumulated Local Effects (ALE) revealed that distance features and specific nucleotide positions significantly influenced model predictions.

Key Findings Regarding RQ1, the research conclusively demonstrates that ensemble methods can improve predictive accuracy while maintaining interpretability through SHAP-based explanations. SHAP reconstructions showed nearly perfect correlation with XGBoost predictions, confirming the reliability of SHAP values in explaining ensemble decisions. Waterfall plots and feature importance analyses revealed biologically meaningful patterns: G-rich and C-rich regions near PAM and seed sites consistently showed high importance across datasets. Position-specific features such as G_{21}, C_{13}, and C_{18} emerged as critical determinants of editing efficiency.

Regarding RQ2, the findings indicate limited transferability between knockout and base editing domains. Models trained on knockout data performed poorly on base editing tasks without retraining, reflecting fundamental biological differences: knockout relies on double-strand breaks producing indels, while base editing performs single-nucleotide conversions without breaks. However, models incorporating thermodynamic and biophysical features (DeepHF, CRISPRon) demonstrated enhanced transferability potential compared to purely sequence-based approaches. Complete retraining on domain-specific data proved essential for achieving acceptable performance in base editing applications.

Future Directions The thesis concludes with several recommendations for advancing the field. Proposed extensions include incorporating diverse datasets such as CHANGE-seq and GUIDE-seq to improve model generalizability, implementing uncertainty-aware modeling through probabilistic approaches to enhance clinical safety, developing context-aware models that integrate chromatin accessibility and RNA-seq data, creating web-based interfaces with real-time SHAP visualization for community adoption, and exploring multi-task learning frameworks that jointly model indel formation, base editing efficiency, off-target effects, and cytotoxicity. The research provides a robust methodological framework that balances predictive performance with biological interpretability, offering valuable tools for biotechnologists designing optimal gRNA sequences for therapeutic applications in cancer treatment and genetic disease intervention.

Contents

1 Introduction **1**

1.1 Main Contributions 2

1.2 Structure of the thesis 4

2 Interpretable and uncertainty-aware off-target prediction methods through CRISPR/Cas9: A Scoping Review **5**

2.1 Protocol 5

2.1.1 Rationale 6

2.1.2 Definitions 6

2.1.3 Preliminary search of other reviews 7

2.1.4 Inclusion Criteria 8

2.1.5 Exclusion Criteria 9

2.1.6 Search strategy 9

2.1.7 Final selection of sources 10

2.2 Data Charting Process 10

2.3 Results 10

2.3.1 Publication frequency and temporal trends 13

2.3.2 Model type and algorithm categorization 13

2.3.3 Interpretability and explainability 14

2.4 Selection of methods for subsequent analysis 23

3 An Interpretable Ensemble Framework for Guide RNA Efficiency scores Across CRISPR Domain **25**

3.1 General Structure 25

3.2 Base Models and training strategy 26

3.2.1 **CRISPREdict** 27

3.2.2 **CRISPR-Aidit** 28

3.2.3 **CRISPRon:** CNN-based model using 121 features including sequence motifs and thermodynamic ΔG. 29

3.2.4 **DeepHF:** BiLSTM model using 11 biological features such as Tm, RNAfold MFE, and base accessibility. 30

3.3 Materials and methods 32

3.3.1 Datasets 32

3.3.2 Ensemble Architecture 33

3.3.3 Performance Metrics 35

3.4 Main Workflow and Methodology 37

3.5 Models of Baseline Machine Learning for Comparative analysis 38

3.5.1 Linear Model 38

3.5.2 Multilayer Perceptron (MLP) 39

3.5.3 Support Vector Regression (SVR): 40

3.5.4 Decision Tree 41

3.5.5 Gaussian process 42

3.6 Explainability 45

3.7 Experimental results 48

3.7.1 Framework on SpCas9-Endo dataset (KO Data) 48

3.7.2 Experiment 1: Transferring the Pre-trained Model on BE Data . . 49

3.7.3 Experiment 3: Re-training All the Models and meta learner . . . 52

3.8 Interpretability Analysis 55

3.9 Discussion of the Results 61

4 Feature Representation for Explainable CRISPR Off-Target Prediction and Base Editing Efficiency 79

4.1 Materials and Methods 79

4.1.1 Datasets 79

4.1.2 Data Pre-processing 80

4.1.3 Feature Encoding Strategies 81

4.1.4 Gradient Boosting Implementation 83

4.1.5 Experimental Setup and Evaluation 83

4.1.6 Hyperparameter Optimization 84

4.2 Results and Discussion 84

4.2.1 XGBoost Robustness on Off-Target Prediction (KO) 84

4.2.2 XGBoost Performance on Base Editing (Hanna) 85

4.2.3 Implicit Learning of Biophysical Properties 86

4.2.4 Hyperparameter Robustness and Task Consistency across Base Editors 87

4.2.5 Error Analysis: Limits of Regression at the Extremes 88

4.2.6 Explainability: Impact of Feature Representation on Model Decisions 88

4.3 Results 90

5 Conclusion 96

5.1 Summary of Contributions 96

5.1.1 Systematic Scoping Review 96

5.1.2 Explainable Ensemble and Research Questions 97

5.1.3 Feature Representation for Off-Target Prediction 98

5.2 Future Work 98

1 Introduction

The CRISPR/Cas9 system—acronym for Clustered Regularly Interspaced Short Palindromic Repeats/CRISPR-associated protein 9—is a robust genome-editing tool with a broad range of applications in research [1]. In CRISPR/Cas9 molecular complex, a customizable RNA fragment, called guide RNA (gRNA), is used for defining which portion of the DNA should be edited, and the Cas9 enzyme is used to cut the DNA. The main application is gene knockout (CRISPR-KO), the process of using genetic engineering to deactivate or eliminate one or more specific genes from an organism. This technique induces DNA breaks, which are repaired by the cell's endogenous mechanisms, often resulting in unpredictable insertions or deletions and potential false positives in gene analyses [2, 3]. A more recent tool, CRISPR/Cas9 Base Editing (BE), enables the insertion of point mutations into DNA without causing DNA breaks. CRISPR-BE is a technique with a more controllable effect, but it must be designed more carefully because not all the mutations have a significant effect on the cell. CRISPR-KO or BE assays can be designed by bioinformatics tools [4], and their effects on living cells can be tested in laboratory via CRISPR screens [5].

The main issue in CRISPR/Cas9 effectiveness are off-target events, which occur when gRNA instead of binding to the desired portion of DNA ends up in another part of the DNA (because of sequence similarity). In fact, the DNA is full of repetitions (3 billion nucleotides composed of only 4 nucleotides), therefore off-target effects can cause cell malfunction, incontrollable growth, or death. Consequently, tools for predicting off-target events and effects given a gRNA sequence [6] are crucial to help genetists designing the gRNA sequences that reduce the risk of not significant assay results.

The main objective of this thesis, which is part of a main project from the PNRR CN3-Spoke7-WP1 "National Center for Gene Therapy and Drugs based on RNA Technology" coordinated by CNR Institute for Biomedical Technologies, is to define explainable methods useful to predict on-target and off-target effects in CRISPR-BE, to help the biologists design optimal gRNA sequences for CRISPR screens on human cell lines,

with the aim of studying the impact of gene editing in treating cancer or genetic diseases.

The additional requirement of explainability enables the intelligibility of the predictions to the biologists, so that they can suggest model refining for biological plausibility, as well as gaining new knowledge from the model.

1.1 Main Contributions

To predict gRNA efficiency and off-target events, several computational tools have been proposed in literature. Some of them provide users with some form of explanation of the prediction, which could give valuable information to help biologists in designing more efficient gRNA sequences.

Two main drawbacks affect predictive models of gRNA efficiency: (i) low accuracy, and (ii) domain specificity. In fact, the complexity of the biological processes underlying the working of CRISPR/Cas9 makes the prediction of gRNA efficiency very hard. As a consequence, the correlation between the predicted efficiency and the actual efficiency may not be high enough for a practical use of the tools; moreover, the predictive model designed for the domain of KO may not be easily transferred to the domain of BE, and vice versa, because the biological processes in these two domains, though related, are significantly different.

In this thesis,the following research questions are addressed:

RQ1: Is it possible to further improve the accuracy of efficiency prediction in available models, and still preserve their explanatory abilities?

RQ2: To what extent is it possible to reuse the knowledge acquired for predicting efficiency in KO domain to BE domain?

Our approach is to design an ensemble of predictive models that share the same input information (a gRNA sequence) and provide an estimation of its efficiency. To improve accuracy—thus addressing RQ1—these base models are stacked, and their predictions are combined by another trainable model to produce the final prediction [7].

To preserve the explanatory ability, the base models and the combiner should provide explanations that can be aggregated together to generate an explanation of the

ensemble. We select SHAP as the explanation tool, because it offers a linear representation of the predictions, thus favoring easy aggregation within the ensemble hierarchy.

The ensemble framework enables the identification of three levels of knowledge: (i) the knowledge acquired by the base models; (ii) the knowledge acquired by the combiner; and (iii) the knowledge acquired by ensemble (base models + combiner). Testing these three levels of knowledge allows us to address RQ2.

In this work, we considered four predictive models that satisfy the constraints defined by the proposed approach. Moreover, these models are available as open source, which favors easy integration and analysis. The selected models are CISPRedict [8], DeepHF [9], CRISPRon [10], and CRISPR-AIdt [11]. While many other models could have been used (see, e.g., [12, 13, 14, 15]) our works tries to delineate a more general methodology that could be easily extended with other models, provided that the required constraints are satisfied.

The starting point of the first stage of this study was finding and assessing the CRISPR/Cas9 gRNA efficiency prediction models that were already in use. Based on their availability, methodological diversity, and popularity, four well-known tools were chosen from four papers that we selected for systematic review.

These tools yield different outcomes and are sometimes regarded as separate black boxes because of its deep complexity and complex patterns, despite their particular advantages. Combining their predictions into a single, coherent, and effective model that may increase accuracy and offer biological understanding was obviously lacking. The main goal of the thesis was to close this gap.

In addition to evaluating existing CRISPR prediction tools, this thesis also provides a dedicated study on how feature representation influences the behaviour of machine learning models. We compared eight different encoding strategies—NPM, Bulges, LEP, K-mer, One-Hot, OH5C, 8XL, and Match-Mismatch—to represent gRNA and off-target sequences, and we analysed how these representations affect prediction accuracy and explainability. Our results clearly show that feature representation plays a key role in model performance: some encodings capture biologically meaningful patterns better, while others simplify the sequence but reduce interpretability. This analysis helps identify which representations are most suitable for accurate and interpretable CRISPR/Cas9 prediction.

Moreover, in this work, we distinguish between machine learning and deep learning

based on the model's ability to learn features. Machine learning refers to traditional models such as linear regression, support vector machines, and tree-based methods that rely on engineered features. Deep learning refers to neural network-based models that learn hierarchical representations directly from input data. Although deep learning is a subset of machine learning, this distinction is adopted for clarity; in this study, models with more than two hidden layers are considered deep learning.

1.2 Structure of the thesis

This thesis is organized into five chapters, each of which plays a crucial role in the examination of interpretability and predictive modeling in CRISPR/Cas9 genome editing.

Chapter 2 - Interpretable and uncertainty-aware off-target prediction methods through CRISPR/Cas9: A systematic scoping review of interpretable and uncertainty-aware CRISPR prediction models is presented in this chapter. It groups tools that have been released according to their methodology, data sources, explainability strategies, and evaluation techniques. Several insights into current trends and gaps in the literature are provided by this review.

Chapter 3 - Guide RNA efficiency prediction through an explainable ensemble: This central chapter examines the effectiveness of four well-known CRISPR efficiency prediction models—CRISPRedict, CRISPRAidit, DeepHF, and CRISPRon—together with an XGBoost meta-learner that combines their outputs. The analysis uses both the Hanna and Knockout datasets to address RQ1 and RQ2.

Chapter 4 - Impact of feature representation in off-target event prediction through CRISPR/Cas9: The different feature encoding techniques for representing gRNA sequences and their biological context are examined in this chapter. It assesses how different encoding techniques—such as k-mers, bulges, mismatches, and position-specific encodings—affect model performance and interpretability, particularly when used with complex architectures like XGBoost.

Chapter 5 - Conclusion: This last chapter of the current thesis highlights the key discoveries about RQ1, RQ2 and provides a summary of the contributions made, and suggests future study areas. It considers the advantages and disadvantages of the suggested framework and makes recommendations for how it might be modified or expanded to address similar genome editing issues.

2 Interpretable and uncertainty-aware off-target prediction methods through CRISPR/Cas9: A Scoping Review

The reported literature analysis was carried out in accordance with the PRISMA-ScR (Preferred Reporting Items for Systematic Reviews and Meta-Analyses Extension for Scoping Reviews) guidelines [16]. This scoping review aims to give an in-depth analysis of the recent literature on predicting off-target effects in CRISPR/Cas9 research. This review specifically aims to define off-target effects and events, distinguish between detection and prediction approaches, examine the justifications for predicting off-target effects, and look at the interchangeability of the terms ”interpretable” and ”explainable” in relation to off-target effects in CRISPR/Cas9 gRNA sequences. This scoping review is being conducted with the goal of bringing together current information, identifying research gaps, and offering suggestions for more field study and advancement.

2.1 Protocol

The following primary and secondary research questions are the focus of this scoping review:

Primary Question:

- Which methods, techniques and tools are available for off-target prediction of CRISPR gRNA sequences?

Secondary Questions:

- Which methods, techniques and tools deal with uncertainty in off-target predic-

tion of CRISPR gRNA sequences?

- Which methods, techniques, and tools deal with explainability in off-target prediction of CRISPR gRNA sequences?
- Which deep learning and machine learning architectures are most important for CRISPR off-target prediction?

2.1.1 Rationale

Genetic engineering has undergone a revolution due to the CRISPR/Cas9 technology, which provides new prospects for precise genome editing. Researchers can target certain genomic regions and make desired alterations by using the Cas9 nuclease's RNA-guided DNA cleavage capability. CRISPR/Cas9 has wonderful potential, but it also raises questions regarding off-target effects, which are unexpected alterations that take place at genomic places that are similar to the intended target but not exactly the same [17]. For CRISPR/Cas9 to be used safely, effectively, and ethically in a variety of applications, including as gene therapy, agriculture, and fundamental research, accurate prediction and assessment of off-target consequences are essential [18] .

Although computational approaches for CRISPR/Cas9 off-target prediction have been the subject of prior investigations [19, 20, 21] there is a gap in the literature regarding the interpretability and uncertainty of these prediction methods. "Interpretability" refers to the ability to understand and explain the predictions made by computational models, while "uncertainty" deals with quantifying and managing the confidence or reliability of those predictions. This scoping review seeks to offer a thorough assessment of the present status of techniques and methodologies for forecasting off-target effects in CRISPR/Cas9 gRNA sequences by examining the interpretability and uncertainty concerns [22].

2.1.2 Definitions

To ensure clarity and understanding, it is important to define certain key terms within the context of this review:

- "Off-target event" is the occurrence of an alternative genomic targeting brought on by CRISPR/Cas9 activity

- “On-target effect” is the intended observable genomic modification affecting the functionality of the cell
- “Detection” is the identification or recognition of off-target effects.
- “Prediction” involves estimating the likelihood or probability of off-target effects occurring.
- “Off-target effects” encompass unintended modifications or alterations in the genetic sequence other than the intended target site.
- ”Interpretability” describes the capacity to comprehend and justify the predictions generated by off-target prediction techniques. It entails determining the elements or characteristics that go into the prediction and offering compelling justifications for the results. In this study, the words “interpretable” and “explainable” are used interchangeably to describe a model’s or algorithm’s capacity to offer precise justifications for its predictions or choices.

2.1.3 Preliminary search of other reviews

A preliminary search in PubMed using the search string

```
((Crispr[Title/Abstract]) AND (review [Title/Abstract]))
AND (off-target [Title])
```

was done to determine the uniqueness and relevance of this scoping review. The following criteria were followed:

- Inclusion criteria: reviews on computational methods, tools, software
- Exclusion criteria: reviews not including computational methods or tools, unreachable papers
- Additional criterion: Look at existing reviews from 2020 and collect papers that describe the methods for designing the predictive models.
- In the case of different papers describing different versions of the same method, the most recent version is kept.

The search found up published analyses of off-target prediction techniques. None of these evaluations, however, paid particular attention to the interpretability and uncertainty problems connected with off-target prediction techniques in the context of CRISPR/Cas9 gRNA sequences. In order to close this gap, this scoping review looks at the most up-to-date explainable strategies for foreseeing off-target consequences [23].

2.1.4 Inclusion Criteria

For this review, we selected papers on off-target effects in CRISPR/Cas9 gRNA sequences that are predicted using computational techniques, instruments, and software.

The following criteria were followed:

Topic Predicting off-target effects in CRISPR/Cas9 gRNA sequences should be the main objective of the research that was included.

Types of studies

1. Reviews that explicitly concentrate on computational techniques, instruments, and software for foreseeing off-target events in CRISPR/Cas9 gRNA sequences, including systematic reviews and meta-analyses.
2. Scientific journals that have produced peer-reviewed research articles.
3. Peer-reviewed research papers covering the creation or assessment of computer models or algorithms for predicting off-target events in the context of CRISPR/Cas9 gRNA sequences.

Methodological framework

- Studies on the use and assessment of reproducible techniques for predicting off-target events.
- Studies investigating interpretability and/or uncertainty management of off-target prediction systems.
- Studies that make use of software, hardware, or computational approaches for predicting off-target events.

Population Since the inclusion criteria are mainly concerned with the mechanism and context of off-target prediction in CRISPR/Cas9 gRNA sequences, no particular demographic criteria are employed.

Language Due to resource limitations and the reviewers' language skills, only studies that were published in the English language are considered.

Timeframe Published studies between 2014 and June 2023.

2.1.5 Exclusion Criteria

the following types of papers were excluded, even if resulting from the search:

1. Laboratory methods for detection of off-target effects in CRISPR/Cas9 gRNA sequences have not been addressed in studies.
2. Studies that only use CRISPR methods other than CRISPR/Cas9, or non CRISPR methods in general.
3. Languages other than English publications
4. Lack of discussion or evaluation of computational techniques, apparatus, or software for off-target prediction

The presence of alternative gene editing techniques like Transcription Activator-Like Effector Nucleases (TALEN) and Chimeric Antigen Receptor T-cell (CAR-T) treatment must be acknowledged in the context of CRISPR/Cas9. These other technologies are not taken into account in this assessment because it is primarily focused on CRISPR/Cas9. The reason TALEN and CAR-T were excluded is because of their unique modes of action, which do not rely on gRNA to direct the editing process.

2.1.6 Search strategy

First-level search: PUBMED

```
(Off-target OR scoring) AND predict* AND CRISPR AND (gRNA* OR "guide RNA*"
OR sgRNA*) AND (method OR tool OR technique OR model)
```

Second-level search: PUBMED

```
Interpretab* OR explain* OR transparen* OR Uncertain* OR probab*
```

Google Scholar We looked for relevant papers (matching the same search string used in PubMed) until an entire search page does not contain any relevant result.

2.1.7 Final selection of sources

Overall, a total of 357 publication records were found in the first search. A screening workflow was applied (see fig. 2.1):

- A total of 125 articles were eliminated at the abstract level.
- 72 articles passed through the initial screening.
- After 91 review papers were assessed, 18 were removed from consideration due to their scope.
- Two more publications were found that provide more details.

2.2 Data Charting Process

Recorded using a standardized data extraction form were:

- Author(s), year, and country
- Model type (e.g., CNN, RNN, XGBoost, regression)
- Techniques for model interpretability (e.g., SHAP, attention)
- Methods for quantifying uncertainty (such as probabilistic scoring and Bayesian dropout)
- The availability and state of maintenance of software and tools

2.3 Results

This scoping review found and examined 82 research papers that created computational models to forecast CRISPR/Cas9 activity, efficiency, or off-target consequences

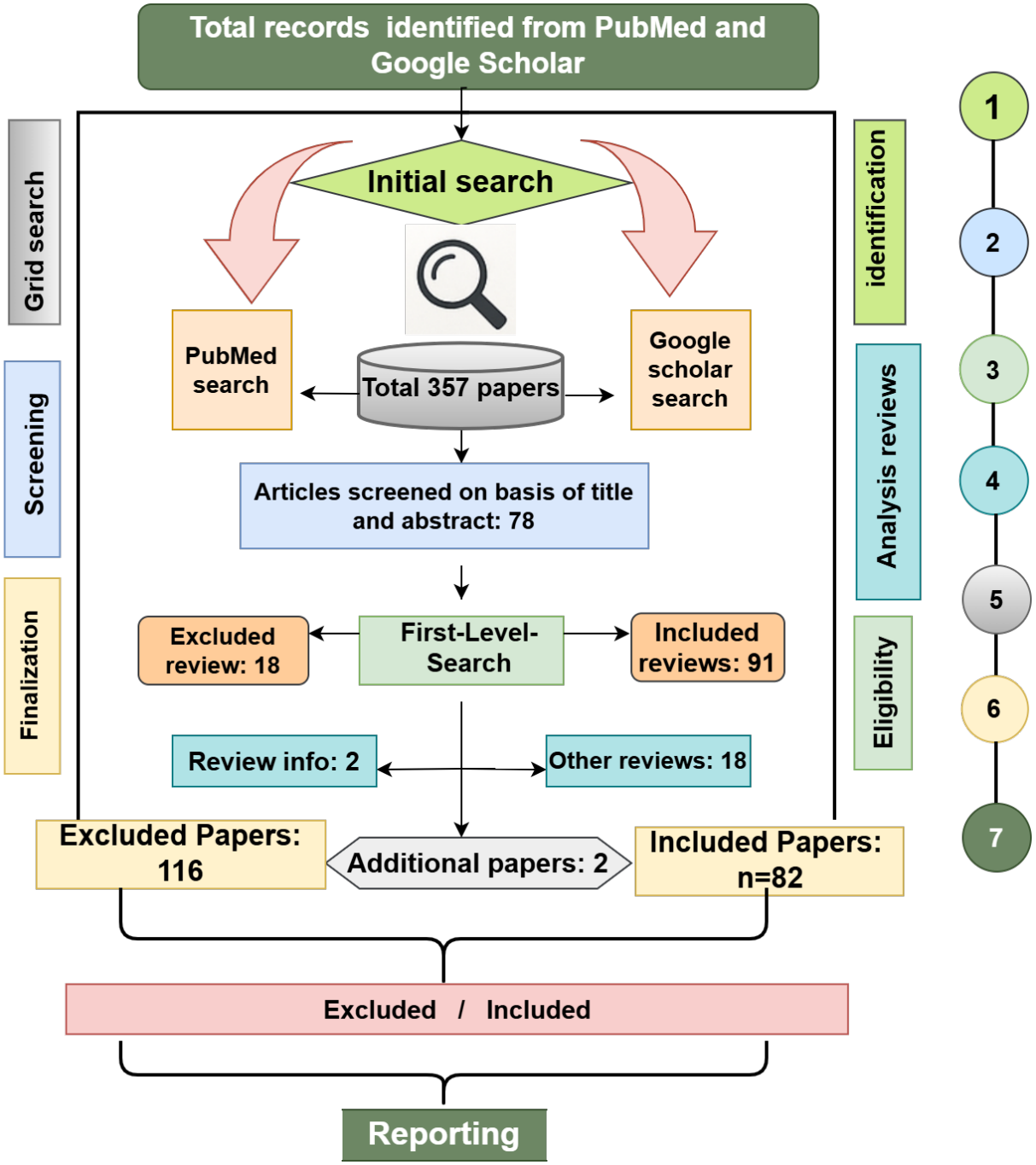


Figure 2.1: A workflow diagram that illustrates the scoping review's initial screening procedure. PubMed and Google Scholar were used to find an initial total of 357 records. 82 review papers were judged pertinent to CRISPR/Cas9 predictive modeling out of the 91 that were kept after applying inclusion and exclusion criteria. According to PRISMA-ScR guidelines, the diagram also displays excluded studies and additional records that have been identified.

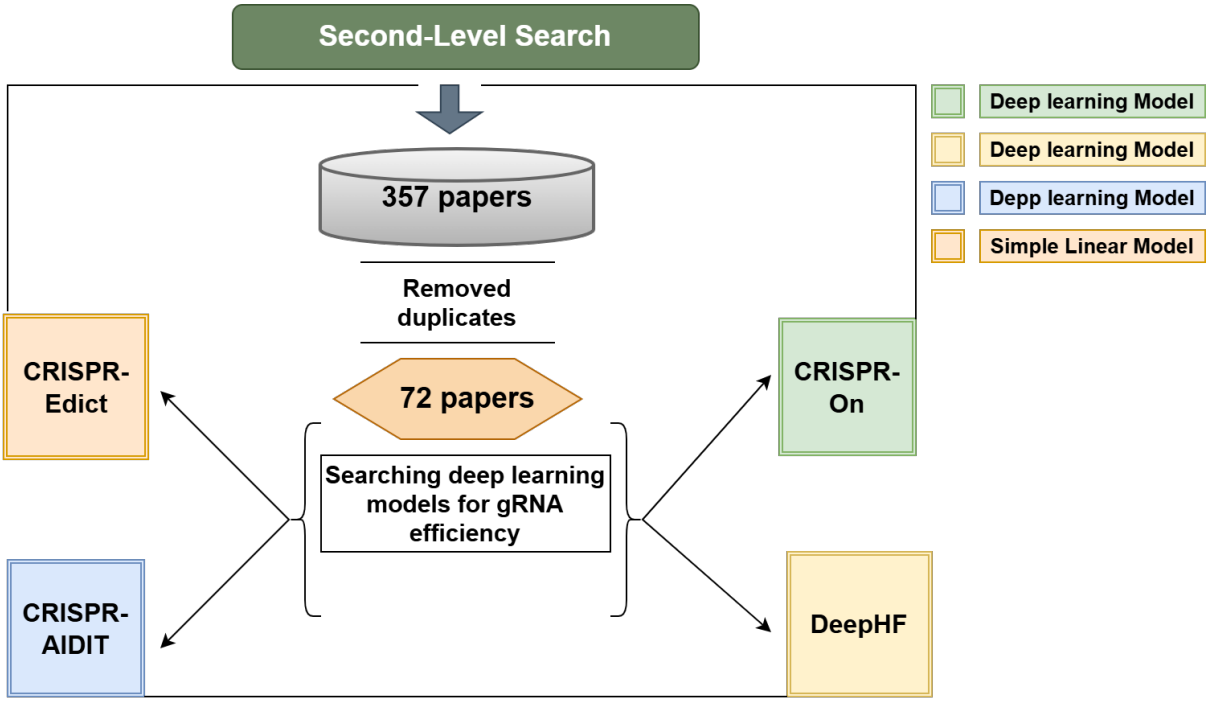


Figure 2.2: The selection of four interpretable and very effective CRISPR/Cas9 predictive tools—CRISPRedict, CRISPRAidit, CRISPRon, and DeepHF—is highlighted in this comprehensive flowchart of the second-level search method. These were selected for a thorough comparison analysis based on model interpretability, performance, and usefulness after being taken from 82 finished articles following the initial screening.

and were published between 2014 and 2023 (Figure.2.2). A total of 357 articles were first found using PubMed and Google Scholar. Following the application of inclusion/exclusion criteria and the removal of duplicates, 91 papers were kept for full-text screening. Of these, 82 papers were determined to be pertinent to predictive modeling using CRISPR/Cas9. Figures 2.1) and 2.2 (First-Level Workflow and Second-Level Workflow) show this procedure. We arranged the chosen papers according to the year of publication in order to examine patterns over time. Tables 1 to 10 indicate that the number of predictive modeling studies has been steadily increasing, with a substantial increase after 2018. This increase is indicative of the increasing accessibility and maturity of machine learning in Bioinformatics research. Based on their modeling methodologies, we categorized the 82 distinct models that were found in the literature as shown in Figures. 2.3 and 2.4. Table 11 lists the explainability techniques used in the four well-known models chosen for further analysis. The detailed summary of the predictive models are shown in 12, while the details of Knockout and Base editing is in 13.

2.3.1 Publication frequency and temporal trends

Over the past ten years, there has been a steady rise in CRISPR/Cas9 predictive modeling research. Only a small number of foundational models were presented between 2014 and 2016 (Tables 1 to 3). But starting in 2019 (Table 6), the number of publications increased dramatically, reaching a peak in 2020 (Table 7) with 13 papers, 2021 (Table 8) with 10 papers, and 2022 (Table 9) with 10 papers, respectively. This increase is a result of deep learning's broad use in bioinformatics and the growing availability of training datasets.

2.3.2 Model type and algorithm categorization

The included models were divided into seven main categories according to their logic and architectural design, as shown in Figure 2.4

- **Deep learning models**, which included CNNs, BiLSTMs, RNNs, CRNNs, and transformer-based techniques, accounted for more than half of all papers. Examples of such tools are DL-CRISPR, CRISPR-IP, CRISPRon, DeepHF and CRISPRAidit.
- **Ensemble learning models**, which integrate several learners for robust performance, were also popular. Examples of these models include GuidePro, XGBoost, and AdaBoost.
- **Hybrid models** like CNN-SVR and CNN + BiLSTM were to integrate the advantages of neural networks with traditional techniques.
- **Classical machine learning models** Because of their interpretability, random forest, SVM, and logistic regression were very common in early research.
- **Thermodynamic and heuristic/rule-based models**, like CRISPRcut and CROP, took prior biological information into consideration.
- **Mechanistic and statistical models**, such as CRISPRedict and MAGeCK, employed kinetic or regression-based modeling to produce clear predictions.

Tables 1 to 10 provide complete annual summaries that include the type of model and algorithms used in each publication from 2014 to 2023. Morovere, the classification of predictive models in the literature is not always consistent. For example, models

based on convolutional neural networks (CNNs) are described as deep learning, neural networks, or even as part of hybrid or thermodynamic frameworks depending on how they are integrated with other components in these base papers workflow. The aim was to the classification of these models as presented in the original studies, while also grouping them under broader categories to ensure a consistent comparison across problems.

2.3.3 Interpretability and explainability

Interpretability and explainability approaches were explicitly used in just a small portion of the 82 reviewed studies. The total of 7 papers are about interpretability, and 2 are about explainability. In addition, Table 11 lists the explainability techniques used in the four well-known models chosen for further analysis.

Explainability is still not well understood in many deep learning-based CRISPR models, despite the increased focus on model performance. Recent transformer-based or hybrid techniques, which frequently place a higher priority on predictive accuracy than transparency, make this disparity especially clear.

Table 1: Summary of CRISPR/Cas9 predictive tools published in 2014.

Paper	**Year**	**Model type**	**Algorithm used**	**Interpretability**	**Explainability**
Wang et al[24].	2014	Machine Learning Rule-based Biological Model	SVM	NO	NO
Montague et al[25].	2014	Web tool	CHOPCHOP	NO	NO
Zu et al[26].	2014	Computational tool	CRISPRseek	NO	NO
Bea et al[27].	2014	Computational tool	Cas-offinder	NO	NO

Table 2: Summary of CRISPR/Cas9 predictive tools published in 2015.

Paper	Year	Model type	Algorithm used	Interpretability	Explainability
Xu et al[28].	2015	Statistical Machine Learning	Logistic Regression (GLM with binomial distribution and logit link)	NO	NO
Naito et al[29].	2015	Web tool	CROP-IT	NO	NO
Singh et al[30].	2015	Mathematical approach	CRISPRscan	NO	NO
Stemmer et al[31].	2015	Web tool	CCtop (Bowtie)	NO	NO
Moreno-Mateos et al[32].	2015	Computational tool	Cas-offinder	NO	NO

Table 3: Summary of CRISPR/Cas9 predictive tools published in 2016.

Paper	Year	Model type	Algorithm used	Interpretability	Explainability
Heussler et al[19].	2016	None (Experimental Protocol Paper)	None—no computational model used	NO	NO
Doench et al[33].	2016	Predictive Model	Rule Set 2 (L1/L2 regression, Boosted RT)	NO	NO
Farasat et al[34].	2016	Statistical and Machine Learning	Gaussian Process Regression (GPR) and Bayesian Optimization	NO	NO

Table 4: Summary of CRISPR/Cas9 predictive tools published in 2017.

Paper	Year	Model type	Algorithm used	Interpretability	Explainability
Xiaog et al[35].	2017	Logistic Regression-Based Model	Rule-based system with integrated scoring	NO	NO
Rahman et al[36].	2017	Machine Learning	SVM	NO	NO
Abadi et al[37].	2017	Random Forest	CRISTA	NO	NO
Shvets et al[38].	2017	Mathematical Model	Discrete-state stochastic model	NO	NO

Table 5: Summary of CRISPR/Cas9 predictive tools published in 2018.

Paper	**Year**	**Model type**	**Algorithm used**	**Interpretability**	**Explainability**
Alkan et al[39].	2018	Thermodynamic Modeling (Physics-based)	Duplex Binding Free Energy Model using nearest-neighbor thermodynamic parameters for RNA:DNA hybrids	Yes	NO
Anderson et al[40].	2018	Predictive Model	CRISPR off-target analysis in genetically engineered rats and mice	NO	NO
Labuhn et al[41].	2018	Linear Model	CRISPRater	NO	NO
Concordet et al[42].	2018	Ensemble Scoring System based on ML models	Gradient Boosted Regression Trees (GBRT), Logistic Regression	NO	NO
Wilson et al[43].	2018	Random Forest	TUSCAN	NO	NO
Peng et al[44].	2018	Ensemble Learning	SVM	NO	NO
Mendoza et al[45].	2018	Computational tool	CASPER	NO	NO
Lin et al[46].	2018	Deep Learning	CNN, FNN	NO	NO
Chuai et al[47].	2018	Deep Learning	CNN (DeepCRISPR)	NO	NO

Table 6: Summary of CRISPR/Cas9 predictive tools published in 2019.

Paper	**Year**	**Model type**	**Algorithm used**	**Interpretability**	**Explainability**
Dimauro et al[48].	2019	Deep Learning	CNN (CRISPRLearner)	NO	NO
Zhang et al[49].	2019	Ensemble Learning	AdaBoost, RF, MLP, SVM	NO	NO
Wang et al[9].	2019	Deep Learning	CNN, RNN	NO	NO
Liu et al[50].	2019	Deep Learning	Transformer, 2D-CNN	Yes	NO
Fortin et al[51].	2019	Experimental Analysis	Multi-target sgRNA analysis	NO	NO
Tycko et al[52].	2019	Rule-based, ML, DL approaches	CRISPRoff, CCTop, CRISPRater – all compared	NO	NO
Labun et al[53].	2019	Rule-Based + Integrated Machine Learning Models	CHOPCHOP v3	NO	NO
Young et al[54].	2019	Statistical Modeling + Linear Regression	LASSO Regression (L1-regularized)	NO	NO
Dandage et al[55].	2019	sgRNA-library design tool	Rule-based / heuristic (not ML)	NO	NO

Table 7: Summary of CRISPR/Cas9 predictive tools published in 2020.

Paper	**Year**	**Model type**	**Algorithm used**	**Interpretability**	**Explainability**
Zhang et al[56]. (DL-CRISPR)	2020	Deep Learning	CNN, Data Augmentation	NO	NO
Zhang et al[57]. (C-RNNCrispr)	2020	Deep Learning	CNN, BGRU	NO	NO
Zhang et al[23]. (CNN-SVR)	2020	Hybrid Model	CNN, SVR	NO	NO
Chen et al[58].	2020	Deep Learning	DNA-BERT, LightGBM	NO	NO
Lin et al[59]. (CRISPR-Net)	2020	Deep Learning	CNN, GBT	NO	NO
Yan et al[60].	2020	Ensemble Learning	AdaBoost	NO	NO
Rafid et al[61].	2020	Machine Learning	SVM, Random Forest	NO	Yes
Danjal et al[62].	2020	Classification Model	CRISPRcut	NO	NO
Liu et al[63].	2020	SeqCor	Random Forest	NO	NO
Wang et al[64].	2020	Machine Learning	GBR, Rule set2, Bayesian ridge regression	NO	NO
Hwang et al[65].	2020	Statistical Modeling + Negative Binomial Regression	MAGeCK (Model-based Analysis of Genome-wide CRISPR/Cas9 Knockout)	NO	NO
Hiranniramol et al[66].	2020	Machine Learning	SVM,GB,RF, ANNs	NO	NO
Lin et al[59].	2020	Deep Learning	Hybrid Neural Network(CNN+ BiLSTM)	NO	NO

Table 8: Summary of CRISPR/Cas9 predictive tools published in 2021.

Paper	**Year**	**Model type**	**Algorithm used**	**Interpretability**	**Explainability**
Xiao et al[67].	2021	Deep Learning	Attention Network	YES	NO
Zhang et al[68]. (CRISPR-ONT)	2021	Deep Learning	CNN, CRNN	YES	NO
He et al[69].	2021	Ensemble Learning	GuidePro	NO	NO
Feng et al[70].	2021	Thermodynamic Model	CNN	YES	NO
Fan et al[71].	2021	Ensemble Learning	XGBoost	NO	NO
Charlier et al[72].	2021	Machine Learning	CNN, RNN, FNN	NO	NO
Vinodkumar et al[73].	2021	Neural Network	GCN	NO	NO
Xiang et al[10].	2021	Neural Network	CNN	NO	NO
Aprilyanto et al[74].	2021	Heuristic Rule-Based Model	CROP algorithm	NO	NO
Kim et al[75].	2021	Deep Learning	DeepPE	NO	NO

Table 9: Summary of CRISPR/Cas9 predictive tools published in 2022.

Paper	Year	Model type	Algorithm used	Interpretability	Explainability
Konstantakos et al[8].	2022	Linear Interpretable Model	GLM (CRISPREDICT)	YES	NO
Zhang et al[76]. (CRISPR-IP)	2022	Deep Learning	CNN, BiLSTM, Attention	NO	NO
Fu et al[77].	2022	Deep Learning	CNN (MOFF)	NO	Yes
Yaish et al[78].	2022	Machine Learning	XGBoost	NO	NO
Vora et al[79].	2022	Random Forest	Linear, MLP, SVR, XGBoost	YES	NO
Chen et al[80].	2022	CRISPRDB	Ensemble learning (RF)	NO	NO
Corsi et al[81].	2022	Supervised Binary Classification	Logistic Regression, Linear SVM	NO	NO

Table 10: Summary of CRISPR/Cas9 predictive tools published in 2023.

Paper	Year	Model type	Algorithm used	Interpretability	Explainability
Lin et al[82].	2023	Neural Network	CNN	NO	NO
Zhang et al[11].	2023	Deep Learning	CNN	NO	NO
Cheng et al[12].	2023	Deep Learning	CRNN, CNN, RF, XGB, SVM	NO	NO

Table 11: Explainability techniques used in the four selected CRISPR/Cas9 predictive models.

Model	**Type**	**Explainability Method**	**Interpretability Level**	**Notes**
CRISPRedict[8]	Generalized Linear Model (GLM)	Coefficient weights	High	Provides direct insight into feature importance via statistical coefficients.
DeepHF[9]	Deep Learning (BiLSTM)	SHAP (SHapley Additive Explanations)	High	Quantifies individual feature contributions for each prediction using SHAP values.
CRISPRon[10]	Deep Learning with Attention	Attention mechanism	Moderate	Highlights influential sequence regions through attention weights in the model.
CRISPRAidit[11]	MLPRegressor + Feature Engineering	Feature importance (engineered)	Moderate	Uses biologically derived features with position-aware significance.

Table 12: CRISPR/Cas9 Predictive Model Types (2014–2023): Count by Category

Model Type	**Count**	**Percentage**	**Example Papers**
Deep Learning	28	43.1%	DeepHF (2019), CRISPRon (2021), Zhang et al. (2020, 2023), Lin et al. (2020), Das et al. (2023)
Machine Learning	14	25.0%	CRISPRedict (2021), Rafid et al. (2020), Hiranniramol et al. (2020)
Hybrid Models	6	8.3%	Zhang et al. (CNN-SVR, 2020), Lin et al. (CNN + BiLSTM, 2020)
Ensemble Learning	10	8.3%	Yan et al. (2020), Cheng et al. (2023)
Classical ML (RF, SVM)	12	5.6%	CRISPRAidit (2022), Das et al. (2023)
Thermo or Heuristic	6	4.2%	CRISPRspec (2016), Doench et al. (2016), Hwang et al. (2020)
Statistical & Others	6	5.6%	CRISPR-RT (2015), Haeussler et al. (2016), Hwang et al. (2020)

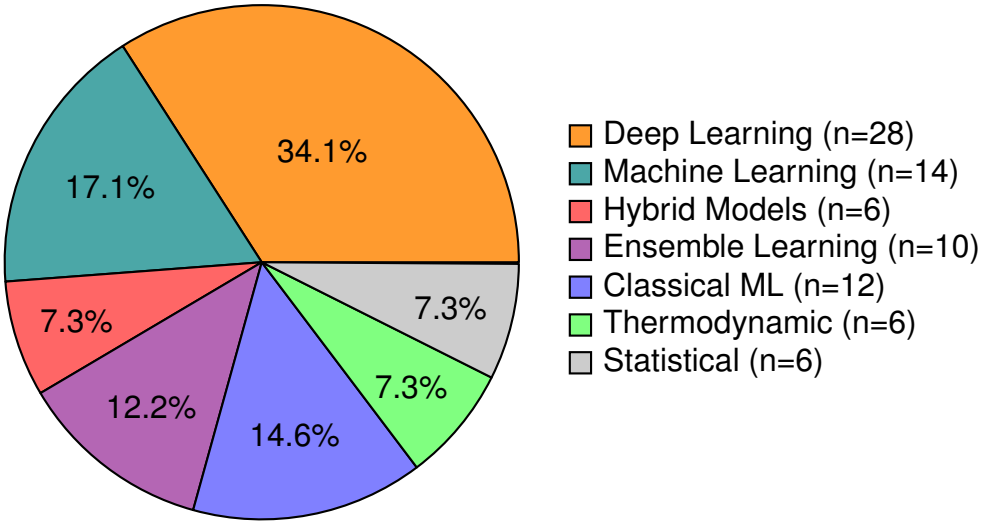


Figure 2.3: Distribution of CRISPR/Cas9 predictive model type (2014–2023) across 72 models. Percentages and counts are derived from exhaustive categorization.

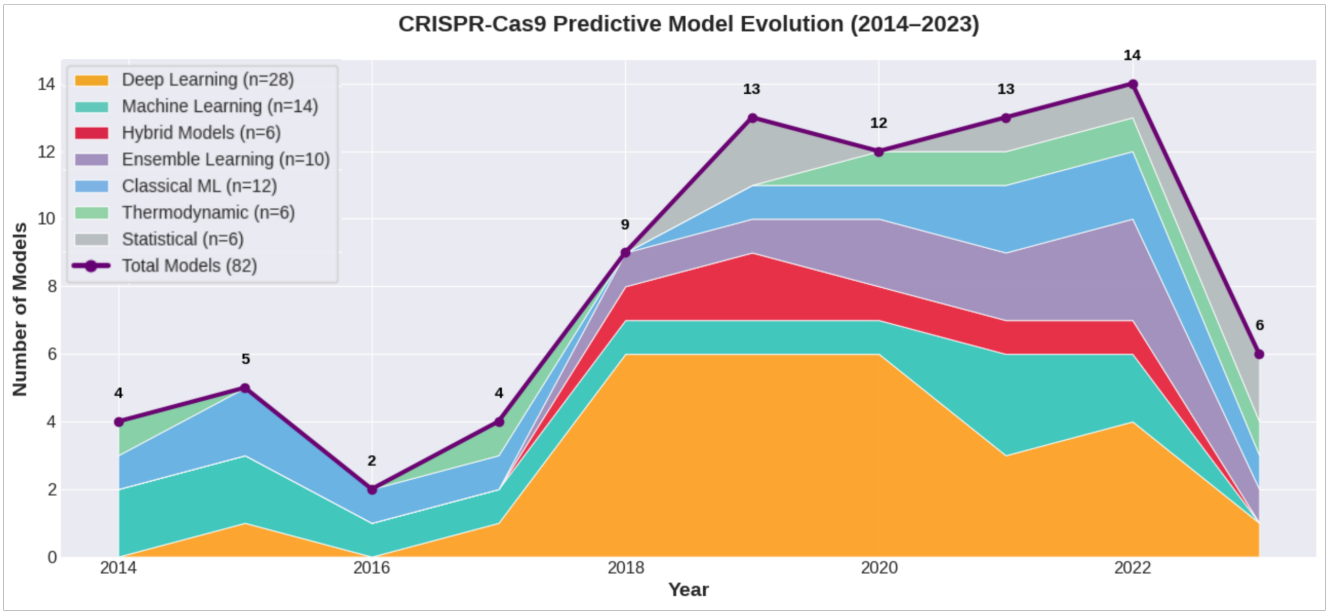


Figure 2.4: The temporal distribution of 82 CRISPR/Cas9 prediction models created between 2014 and 2023, broken down by model type, is displayed in a stacked area plot. Deep learning, machine learning, ensemble, hybrid, classical machine learning, thermodynamic, and statistical models are some of the categories. This graphic shows how deep learning models have become more and more dominant over time, but it also shows how different methodological approaches have been steadily contributing.

Table 13: Verified use of knockout or base editing datasets in CRISPR/Cas9 predictive tools.

Paper (Citation Key)	Focus Area	Evidence of Dataset Usage
Chuai et al[47]. (2018) (DeepCRISPR)	Knockout	Trained on sgRNA knockout efficiency data (from experimental CRISPR-KO assays); source: *Genome Biology* 2018
Xu et al[28]. (2015)	Knockout	Validated using knockout screens on multiple human cell lines (HL-60, KBM-7, etc.); source: *CRISPR-SSC*
Dandage et al[55]. (2019) (Beditor)	Base Editing	Focused on designing gRNAs for base editing and prime editing using a simulation-based dataset; source: GitHub and preprint
He et al[69]. (2021) (GuidePro)	Knockout	Developed using tiling-sgRNA datasets from genome-scale KO experiments; source: GitHub and literature
Wang et al[24] (2014) (Genetic screen)	Knockout	Used a genome-wide knockout screening library targeting 18,080 genes; source: *Science*, 2014

2.4 Selection of methods for subsequent analysis

This scoping review gave priority to comprehending the level of explainability and interpretability provided by CRISPR/Cas9 predictive models in addition to mapping the landscape of these models. In our second-level selection process, explainability was a crucial factor, and we looked for models that offered a clear understanding of their internal decision-making procedures. This was particularly crucial because the life sciences are increasingly focusing on responsible and interpretable models. We selected four tools—CRISPRedict, CRISPRAidit, DeepHF, and CRISPRon—for in-depth examination based on both performance and interpretability after selecting 70 pertinent models from our systematic evaluation of 357 preliminary publications. These models all exhibit explainability in different ways:

CRISPRedict: Through the use of coefficient weights and a generalized linear model (GLM), CRISPRedict allows users to immediately evaluate how characteristics contribute to predictions. As a result, the model is very visible and auditable.

DeepHF: By incorporating SHAP (SHapley Additive Explanations) into its BiLSTM-based deep learning model, DeepHF enables thorough visualization of the ways in which every input feature—such as secondary structure or GC content—influences the machine's predictions.

CRISPRon: Attention mechanisms that emphasize important gRNA sequence areas that affect cleavage efficiency are incorporated into CRISPRon. Attention maps provide visual clues into model behavior, but they are not as easily interpreted as GLM or SHAP.

CRISPRAidit: By examining which manually created features (such as mismatch positions, PAM type, and GC content) affect the predictions, CRISPRAidit, which is based on a neural network trained with biologically engineered features, offers moderate interpretability.

3 An Interpretable Ensemble Framework for Guide RNA Efficiency scores Across CRISPR Domain

3.1 General Structure

This chapter of the current thesis expands on the preceding models described an explainable machine learning framework that integrates predictions from CRISPRedict, CRISPRAidit, CRISPRon, and DeepHF model architectures and uses XGBoost as a meta-learner to boost performance and enable interpretability through SHAP (SHapley Additive exPlanations). This research aims to improve biological knowledge and prediction accuracy to assist in the best gRNA design in the presence of genetic engineering. Genetic engineering has been transformed by the CRISPR/Cas9 gene-editing system, which allows precise genome modifications. However, off-target effects pose a major challenge to its safe and efficient application. We present a comprehensive evaluation of these predictive models for guide RNA (gRNA) efficiency design, by combining machine learning and deep learning approaches with biological feature engineering. We used an explainable machine learning workflow that integrates the actual prediction logic with exact feature engineering of these 4 models, with the aim of improving both the prediction accuracy and biological understanding to support gRNA design through efficiency scores.

Two main drawbacks affect predictive models of gRNA efficiency: (i) low accuracy, and (ii) domain specificity. In fact, the complexity of the biological processes underlying the working of CRISPR/Cas9 makes the prediction of gRNA efficiency very hard. As a consequence, the correlation between the predicted efficiency and the actual efficiency may not be high enough for a practical use of the tools; moreover, the predictive model designed for the domain of KO may not be easily transferred to the domain of BE, and

vice versa, because the biological processes in these two domains, though related, are significantly different.

In this work, we address the following research questions:

RQ1: Is it possible to further improve the accuracy of efficiency prediction in available models, and still preserve their explanatory abilities?

RQ2: To what extent is it possible to reuse the knowledge acquired for predicting efficiency in KO domain to BE domain?

Our approach is to design an ensemble of predictive models that share the same input information (a gRNA sequence) and provide an estimation of its efficiency. To improve accuracy—thus addressing RQ1—these base models are stacked, and their predictions are combined by another trainable model to produce the final prediction.

To preserve the explanatory ability, the base models and the combiner should provide explanations that can be aggregated together to generate an explanation of the ensemble. We select SHAP as the explanation tool, because it offers a linear representation of the predictions, thereby favoring easy aggregation within the ensemble hierarchy.

The ensemble framework enables the identification of three levels of knowledge: (i) the knowledge acquired by the base models; (ii) the knowledge acquired by the combiner; and (iii) the knowledge acquired by ensemble (base models + combiner). Testing these three levels of knowledge allows us to address RQ2.

3.2 Base Models and training strategy

Four different predictive models were used in order to guarantee reliable and broadly applicable predictions of gRNA efficiency in the context of CRISPR/Cas9 genome editing, each of which stands for a distinct computational paradigm: BILSTM architectures, shallow machine learning, deep learning, and linear regression. Using 10-fold cross-validation on the Hanna dataset—which comprises 5062 gRNA sequences with original z-score ground truth values—all models were retrained from the beginning. Cross-validation decreased the chance of overfitting and enabled objective measurement of predictive performance.

3.2.1 CRISPREdict

CRISPRedict is a Linear Model with 29 Features. In order to improve the prediction of guide RNA (gRNA) efficiency in the setting of U6-driven expression systems, we used 29 designed sequence features in the U6 promoter variant of the **CRISPRedict** model, which is an interpretable linear regression model. The variant U6 addresses the distinct regulatory characteristics of the U6 promoter, which is widely used in CRISPR/Cas9 genome editing applications and commonly used for human organisms.

In order to ensure that the predictions remain contextually specific and biologically appropriate, the model was trained on carefully selected datasets that included gRNAs that were solely produced under the U6 promoter. The original feature set incorporated a wide range of biologically meaningful sequence characteristics. These features include motifs known to affect the effectiveness of CRISPR editing, positional nucleotide preferences, and global nucleotide composition (e.g., GC content).

To ensure model efficiency and interpretability, a rigorous multi-step feature selection process was applied. This process led to the identification of a compact yet highly informative subset of **29 features** that retained the predictive power of the original broader feature set.

The final model architecture employs a **binomial regression framework** trained on this condensed feature set to estimate the probability of efficient genome editing. The design of the CRISPRedict U6 variant reflects a deliberate trade-off between accuracy, simplicity, and computational efficiency. The resulting model accepts sequence-based inputs alongside numerically engineered features and produces quantitative predictions of gRNA editing efficiency.

By combining explainability with performance, the U6 variant of CRISPRedict provides an accessible and interpretable tool for researchers seeking to optimize gRNA design in both experimental and therapeutic settings.

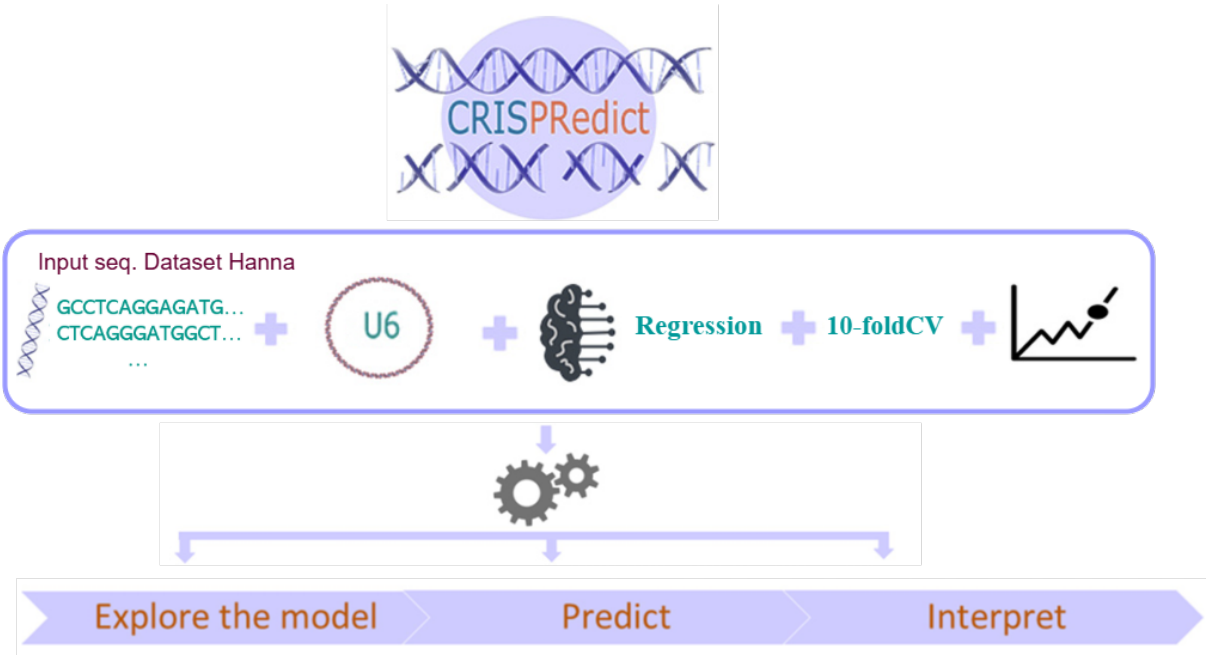


Figure 3.1: Comparison of the general performance of the CRISPRedict model. A modified workflow diagram from Konstantakos et al.[8]. Interpretable features extracted from gRNA sequences served as the foundation for the model.

3.2.2 CRISPR-Aidit

CRISPR-AIdt is based on a Multi-Layer Perceptron (MLP) model using 473 features extracted from an input sequence, including a comprehensive collection of specifically designed biological features resulting from 63-nucleotide context based on the 23-nt target sequence and the adjacent 20-nt upstream and 20-nt downstream on the reference genome, preserving strand orientation.

One-hot encoding of the 63-nt region, positional mismatch flags, nucleotide pair mismatches between the target and off-target sequences, PAM sequence encoding, and GC content were among the biologically significant descriptors used to encode each gRNA. A fixed-length vector of 473 dimensions was thereby created by combining these properties, guaranteeing uniformity across all inputs.

To learn the complex connection between these characteristics and gRNA efficiency, CRISPR-AIdt uses a fully connected MLP with five hidden layers.

Regression objectives were used for training, the Adam method was used for optimization, and an L2 penalty ($\alpha = 0.001$) was used for regularization to minimize overfitting.

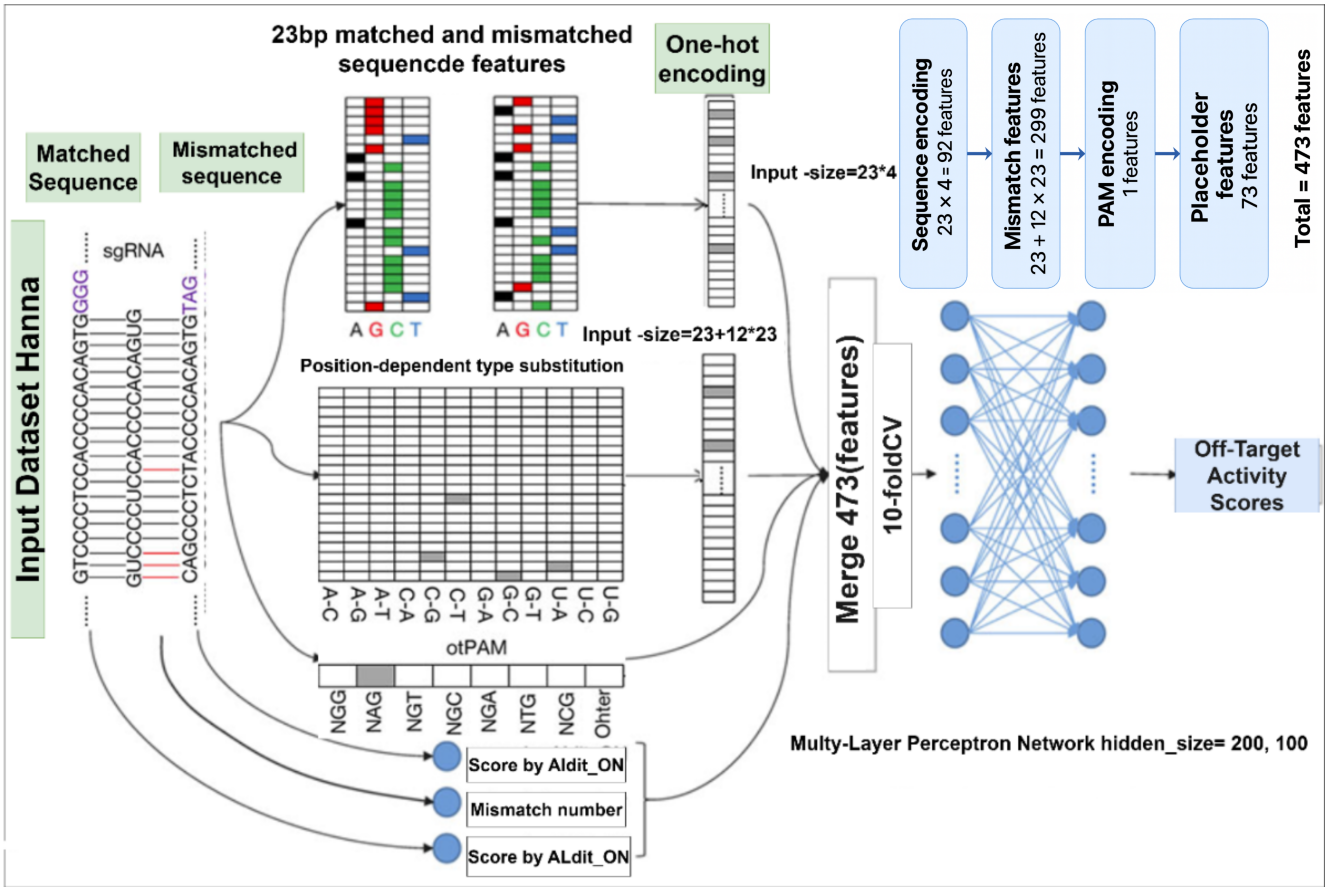


Figure 3.2: A modified Schematic of the CRISPR-AidIt workflow from Zhang et al. [11] for feature extraction and multi-layer perceptron-based off-target activity prediction. A 473-dimensional input vector processed through hidden layers (200, 100) is created by the model integrating sequence encoding (92 features), mismatch features (299), PAM encoding (8), GC content (1), and placeholders. Advanced gRNA sequence encodings and mismatch-related features were integrated into the model

3.2.3 CRISPRon: CNN-based model using 121 features including sequence motifs and thermodynamic ΔG.

In order to improve the prediction of gRNA cleavage efficiency under U6 promoter expression, we used a dataset of 30-nucleotide sequences from Knockout (KO) and Hanna (BE) dataset experiments to trained the CRISPRon model. To record nucleotide-level resolution, each sequence was one-hot encoded into a 30×4 matrix. To represent biophysical stability, a thermodynamic characteristic called ΔG was computed based on GC content. A hybrid deep learning approach that included a one-dimensional convolutional layer and global max pooling to extract significant sequence patterns was used to build the model architecture. A single neuron output for regression was obtained by concatenating the pooled output with the ΔG feature and passing it through a dense hidden layer of 64 units. The Adam optimizer and mean squared error loss were

used to train the model over 20 epochs at a learning rate of 0.001. We used SHAP (SHapley Additive exPlanations) to evaluate the impact of specific nucleotides andΔG on each prediction in order to interpret feature contributions. In addition to producing biologically significant explanations, this interpretable retraining pipeline achieved high accuracy, giving researchers a better understanding of the dynamics of gRNA creation inside the CRISPRon framework.

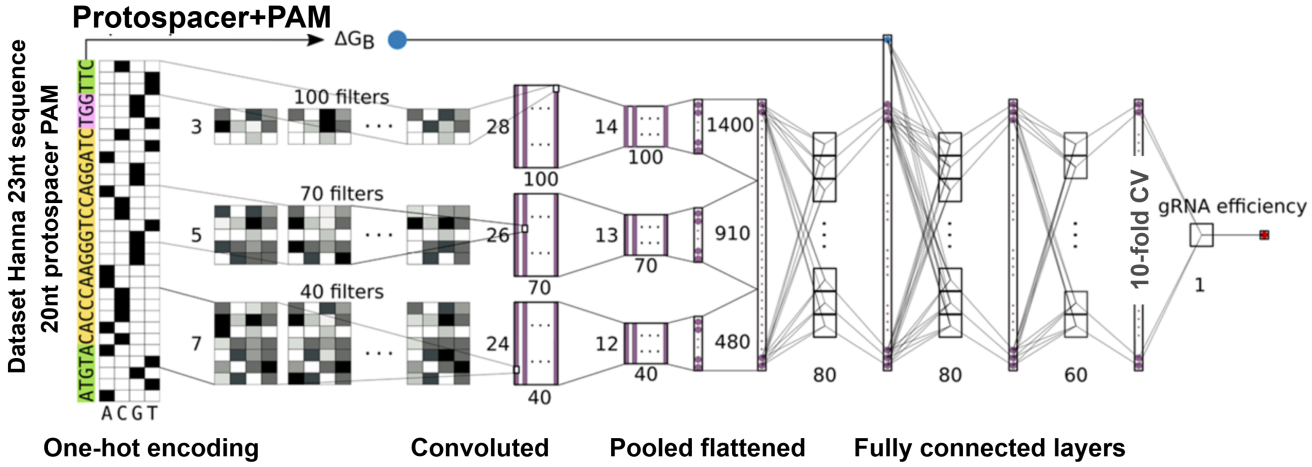


Figure 3.3: A modified Schematic illustration of the CRISPRon from Xiang et al. [10] prediction algorithm and input DNA sequence. The one-hot encoded 30mer and the binding energy ΔG are the inputs to the deep learning network. Only the three completely connected layers and the filtering (convolutional) layers are seen clearly; the thin vertical bars represent the output of one layer, which acts as the input for the subsequent layer.

3.2.4 DeepHF: BiLSTM model using 11 biological features such as Tm, RNAfold MFE, and base accessibility.

We trained this BiLSTM DeepHF model using a hybrid deep learning technique that combines both sequence embeddings and biologically designed features in order to improve gRNA efficiency prediction under standardized sequence settings. Sequence dependencies were captured by training the model using 21-nt trimmed sequences that were obtained from 23-nt gRNAs, one- hot encoded, and processed through a two-layer bidirectional LSTM network. We retrieved 11 biologically significant features, such as base accessibility, free energy, melting temperatures, and nucleotide composition, which were normalized and fed through dense layers that were totally connected. After that, the biological and sequence streams were combined and sent into a linear output layer for regression. SHAP values were calculated for every biological trait, offering information on how they affected the anticipated efficiency. The model's dual-input architecture provided a strong tool for guided gRNA design in CRISPR/Cas9

systems, enabling it to attain high accuracy while maintaining interpretability.

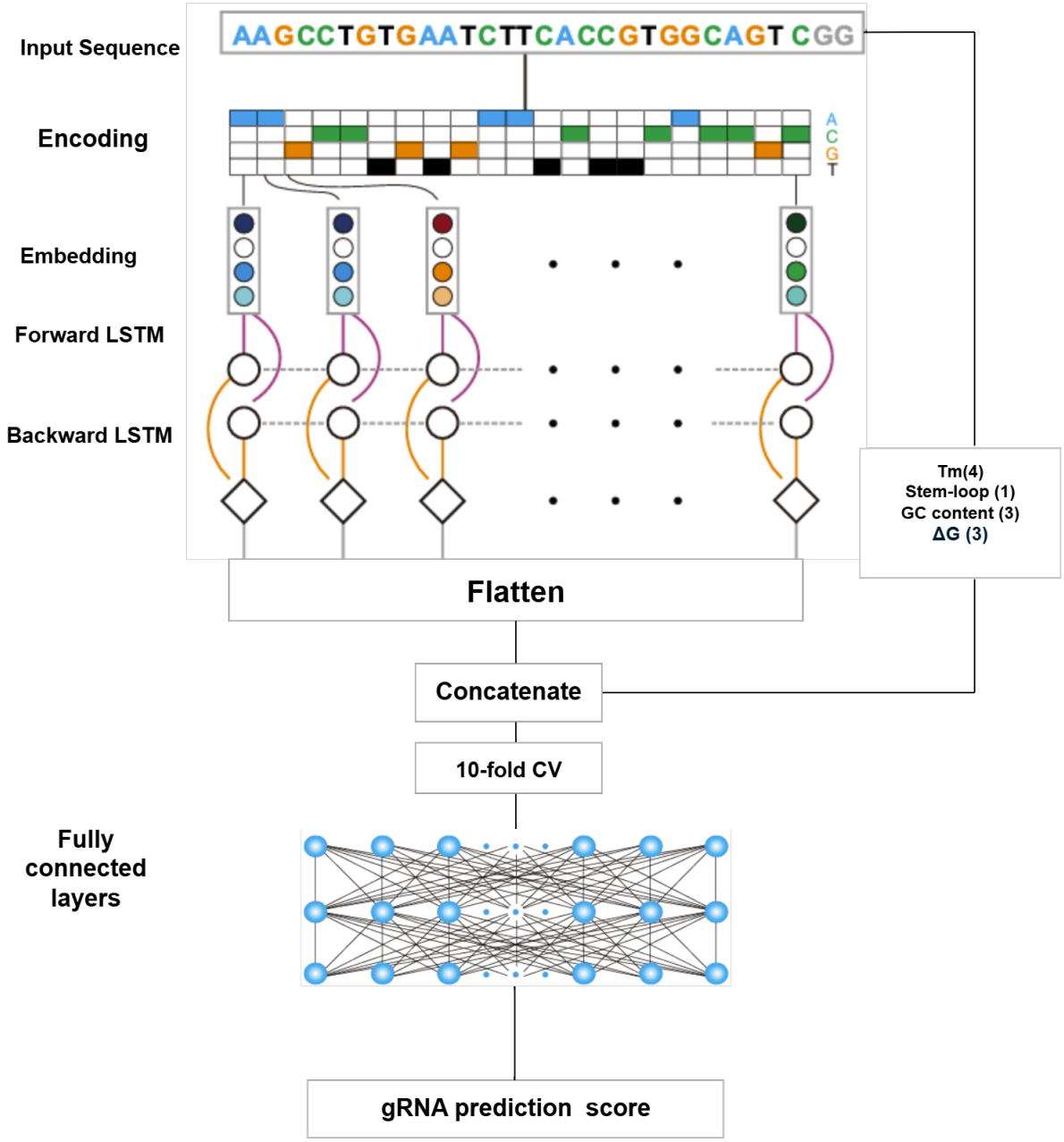


Figure 3.4: DeepHF modified workflow from Wang et al. [9], for predicting gRNA activity. A new representation is obtained by first encoding and then embedding the original gRNA sequence. The final representation, which is concatenated with manually created features to function as the input of the fully connected layer to undergo nonlinear transformation, is obtained by further processing this new representation using a BiLSTM. Lastly, the prediction score is obtained using a linear transformation.

3.3 Materials and methods

3.3.1 Datasets

We used two high throughput CRISPR off-target datasets, each representing effects of KO and BE editing variants.

SpCas9-Endo Dataset. This dataset, which was acquired from Zhang et al.[11], includes 79 CRISPR/Cas9 target sequences, each measuring 63nt, together with ground truth values that indicate the effectiveness of indel generation in HEK293T cells after CRISPR KO editing.

Hanna Dataset. This second dataset comes from Hanna et al.[83], which shows single-nucleotide alterations using CRISPR-BE (BE4max and ABE8e). Thousands of naturally occurring human single-nucleotide variations (SNVs) that have been inserted into endogenous sites throughout the genome are included in this dataset with the total samples of 11000 gRNAs. The functional impact of each SNV, as determined by a cell enrichment experiment and subsequent sequencing, was used to quantify the editing results by z-scores, which show the statistical variance in cell proliferation after editing in comparison to background controls. The `BE39:MELJUSO:zscore` feature provides standardized editing efficiencies for each gRNA. Positive z-scores correspond to above-average cell enrichment, negative scores indicate cell depletion, and zero values denote no performance. For modeling, the original z-scores were used to emphasize the magnitude of editing activity (with any effect on cell growth), ensuring that all target values were comparable across KO and BE experiments. To focus specifically on gRNAs that are expected to reduce cell proliferation—analogous to the effects observed in KO experiments—we filtered the Hanna dataset to retain only those guides predicted to introduce deleterious genomic alterations. Using the ”mutation bin” feature (predicted by Ensembl VEP) for a preliminary annotation, we selected gRNAs whose edits were classified as likely disruptive to gene function. Consequently, guides annotated as “silent” or “no edits” were excluded, while those associated with “intron,” “missense,” “nonsense,” “splice site,” and “UTR” effects were retained. This filtering resulted in a final subset of 5,062 gRNAs. For training the models, the original z-scores BE39:MELJUSO:zscore from these 5062 gRNAs were used to emphasize the magnitude of editing activity for direct comparability across KO and BE experiments workflow

.

3.3.2 Ensemble Architecture

Fig.3.5 illustrates the ensemble learning workflow employed for predicting gRNA efficiency in CRISPR/Cas9 systems. The workflow comprises four parallel pipelines that converge into a meta-learning architecture, ultimately producing a unified prediction output.

Input stage. The workflow initiates with a single gRNA sequence, which is augmented with 20nt both upstream and downstream according to CRISPR-AIdt requirements. Subsequently, the sequence is subjected to independent preprocessing tailored to the specific requirements of individual predictive models.

Base Model Layer. Following preprocessing, the gRNA sequences are fed into four prediction models that function as base learners in the ensemble framework. CRISPRedict employs a linear regression approach with biologically interpretable features, CRISPR-AIdt utilizes a multilayer perceptron with 473-dimensional feature vectors, DeepHF implements a bidirectional LSTM architecture incorporating thermodynamic properties, and CRISPRon combines convolutional neural networks with free energy calculations. Each model independently generates efficiency scores based on its learned patterns and feature representations.

Meta-Learning Integration. The individual prediction scores from all four base models serve as input features for the XGBoost meta-learner, which functions as the ensemble combiner. This stacking approach enables the meta-learner to leverage the complementary strengths of each base model while mitigating their individual limitations. The XGBoost algorithm processes the four-dimensional feature vector (comprising the prediction scores from each base model) to generate refined final predictions.

Output Generation. The final output represents the ensemble's integrated prediction of gRNA efficiency, which has demonstrated superior performance compared to individual base models across multiple evaluation metrics including Pearson correlation coefficient, coefficient of determination (R^2), and root mean squared error. The architecture's modular design facilitates interpretability through SHAP (SHapley Additive exPlanations) value aggregation, enabling biological interpretation of the ensemble's decision-making process at both the base model and meta-learner levels. This approach to handling structured data is renowned for its great precision, scalability, and

robustness. The input features formed a four-dimensional vector for each guide RNA and comprised the prediction scores from CRISPR-AIdt, CRISPRedict, DeepHF, and CRISPRon. The final predictions were subsequently used for each feature contribution analysis, and demonstrated strong predictive ability as measured by R2, RMSE, and Pearson correlation, in comparison with Linear Model, Multilayer Perceptron (MLP), Support Vector Regression (SVR), Decision Tree and Gaussian process.

XGBoost Model Configuration

The XGBoost model was configured with the following hyperparameters, optimized to balance computational efficiency and predictive performance:

- **Number of Trees (`n_estimators`):** 50: shows how many decision trees in total will be used to train the model. While more trees boost learning capacity, they also come at a larger computational cost.
- **Learning Rate:** 0.1: The learning rate regulates the speed at which the model adjusts to the data by fixing mistakes at each iteration. A trade-off of 0.1 prevents updates from being too aggressive while enabling you to progressively converge towards an ideal solution.
- **Maximum Tree Depth:** 3: specifies the model's maximum decision tree depth. More depth raises the possibility of overfitting but enables you to discover more intricate patterns. Selecting a depth of 3 strikes a compromise between generalization and learning capacity.
- **Random State:** 42 (for reproducibility):By assigning a fixed seed to any random operations in the model, this parameter in XGBoost guarantees reproducibility and guarantees that the same outcomes are produced each time the code is executed. This is necessary for comparing and evaluating model performance consistently across experiments.

Training and Validation Procedure

For each fold of the original 10-fold cross-validation:

1. The train predictions from CRISPRedict, CRISPRAidit, CRISPRon, and DeepHF were collected.

2. The test predictions from CRISPRedict, CRISPRAidit, CRISPRon, and DeepHF were collected.
3. These predictions were combined into a new training/testing set containing predictive features for each gRNA.
4. XGBoost was trained on the combined training predictions and tested on the test predictions, ensuring that the training data from the base models were not reused for evaluation.

The performance of the XGBoost model remained objective and indicative of its actual generalization skills by strictly separating the training and test sets at each fold.

3.3.3 Performance Metrics

Several statistical measures were applied to the training and test sets in order to assess the models' predictive accuracy. These consist of the coefficient of determination (R^2), mean squared error (MSE), root mean squared error (RMSE), and Pearson correlation coefficient. These supplementary measures offer a thorough evaluation of prediction error and linear correlation.

Pearson Correlation Coefficient (Test and Train)

The Pearson correlation coefficient (r) measures the strength and direction of the linear relationship between the model's predicted values and the true observed values. It is computed as:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.1}$$

where x_i are the predicted values, y_i are the observed values, and $\bar{x}$ and $\bar{y}$ are their respective means.

- **Pearson (Test):** evaluates how well predictions and ground truth match linearly on test data that hasn't been observed yet. Stronger predictive alignment is indicated by values near 1.
- **Pearson (Train):** evaluates the correlation on the training set, giving information about how well the model can identify patterns without becoming overfit.

Mean Squared Error (MSE)

The average squared error between expected and actual values is measured by MSE, which restricts greater errors more severely. It is given by:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.2}$$

A lower MSE value indicates that the model's predictions are closer to the true values.

Root Mean Squared Error (RMSE)

RMSE is the square root of MSE and represents the standard deviation of prediction errors. It is calculated as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.3}$$

More interpretability is achieved by expressing RMSE in the same units as the target variable.

Coefficient of Determination (R^2)

R^2evaluates how much of the variance in the observed data can be accounted by the model. It is computed as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.4}$$

R^2 values close to 1 signify that the model accounts for most of the variability in the data, while values closer to 0 indicate limited predictive power.

The evaluation was carried out on the concatenated test set results from all folds to provide a comprehensive assessment of the model's robustness and accuracy.

The purpose of Ensemble learning

In contrast to any single model, ensemble techniques such as XGBoost combine predictions from several models to improve generalization and robustness. By using model stacking, we aimed to reduce the variance and bias associated with these 4 individual predictive models, particularly when faced with a complex biological dataset like the Hanna dataset, which includes z-score ground truths for base editing efficiency scores.

Input Features to XGBoost

The XGBoost ensemble was trained on the predictions generated by four distinct models, CRISPRedict, CRISPRAidit, CRISPRon and DeepHF, respectively, each of which had been previously retrained using 10-fold cross-validation on the Hanna dataset.

The anticipated efficiency value from one model for each gRNA was represented by each feature in a new feature matrix created by concatenating the test set predictions from these models for each fold. To avoid any kind of data leaking and to preserve the integrity of cross-validation, this method made sure that XGBoost was trained and assessed only on out-of-sample predictions.

3.4 Main Workflow and Methodology

In this study, the predicted outputs of four separate CRISPR/Cas9 models—CRISPRedict, CRISPRAidit, CRISPRon, and DeepHF—were combined into a single ensemble prediction using the XGBoost regression model as a meta-learner. The process used a 10-fold cross-validation framework and strictly separated training and testing data. The four models' predictions for each fold made up the training/testing set, which was used as input data for the XGBoost model. The experimentally measured z-scores from biological laboratory tests served as the relevant ground truth (GT) for training. During the testing phase, the trained XGBoost model was applied to the predictions made by the same four models on the test set that was not visible. The ensemble outputs were then compared to the true z-scores for validation. This process made sure that the XGBoost model effectively captured complementary information from different models while preserving strong validation through cross-fold evaluation. It also learned to link patterns in the predictions of current CRISPR tools to actual biological outcomes.

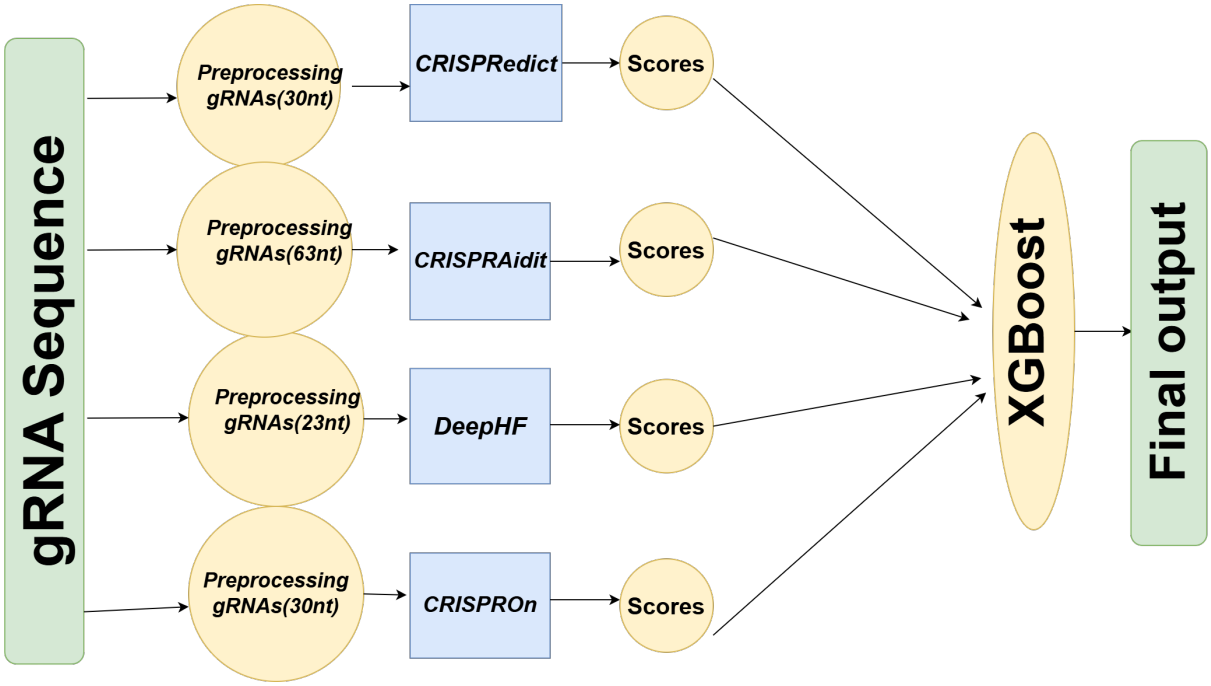


Figure 3.5: Main Workflow

3.5 Models of Baseline Machine Learning for Comparative analysis

To establish a baseline, we evaluated more basic machine learning models first , such as Gaussian Regression, Decision Tree, Support Vector Machine (SVM), and Linear Regression, before moving on to more intricate designs, such as Multilayer Perceptron (MLP) and XGBoost. We tested these models on Knockout data set with the indel efficiencies as a GT.

3.5.1 Linear Model

To train a basic linear regression model, 10-fold cross-validation was used. The general linear relationship between the tool predictions and the ground truth values was captured by the model.

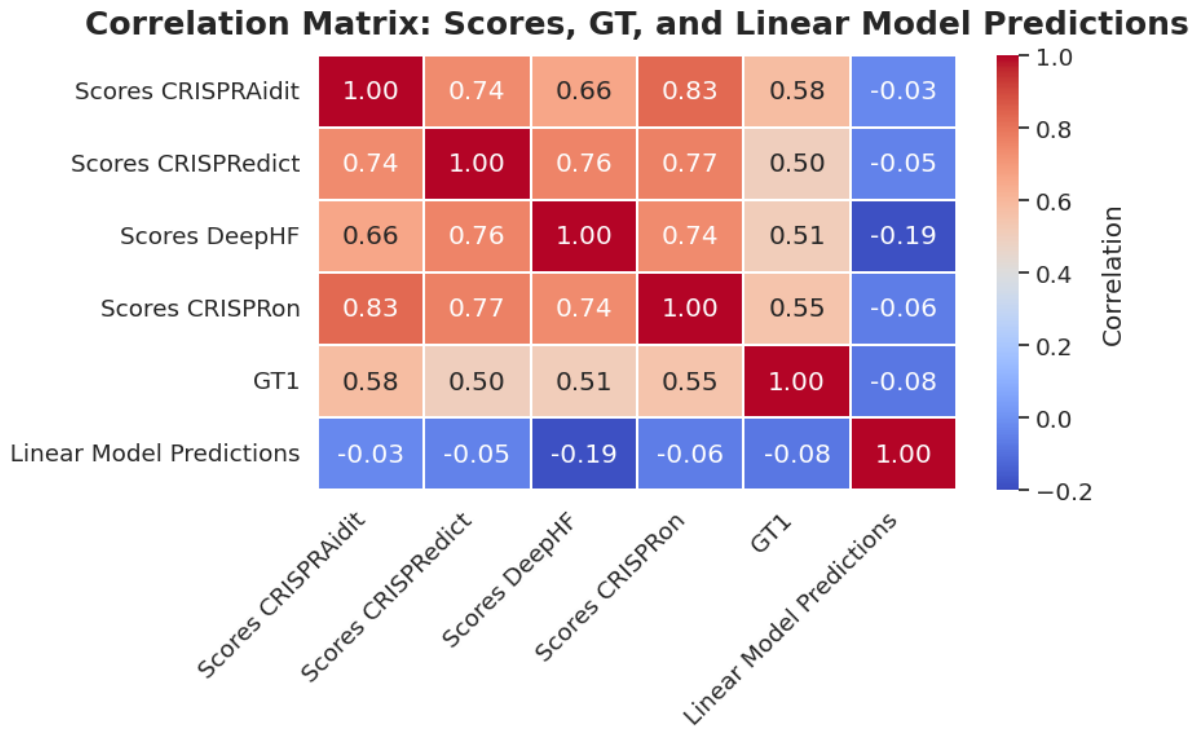


Figure 3.6: Correlation Analysis of Linear Model on SpCas9-Endo Dataset, model predictions vs GT shows very weak relation

3.5.2 Multilayer Perceptron (MLP)

The MLP model was trained with hyperparameter adjustment across several hidden layer sizes and activation functions using layered 10-fold nested cross-validation. For every fold, standardScaler was used to standardize the feature set, and a grid search was used to optimize the hyperparameters on the activation functions (relu, tanh, and logistic) and the number of hidden units (between 5 and 150). A learning rate of 0.001 and a maximum of 5000 iterations were used to train the model.

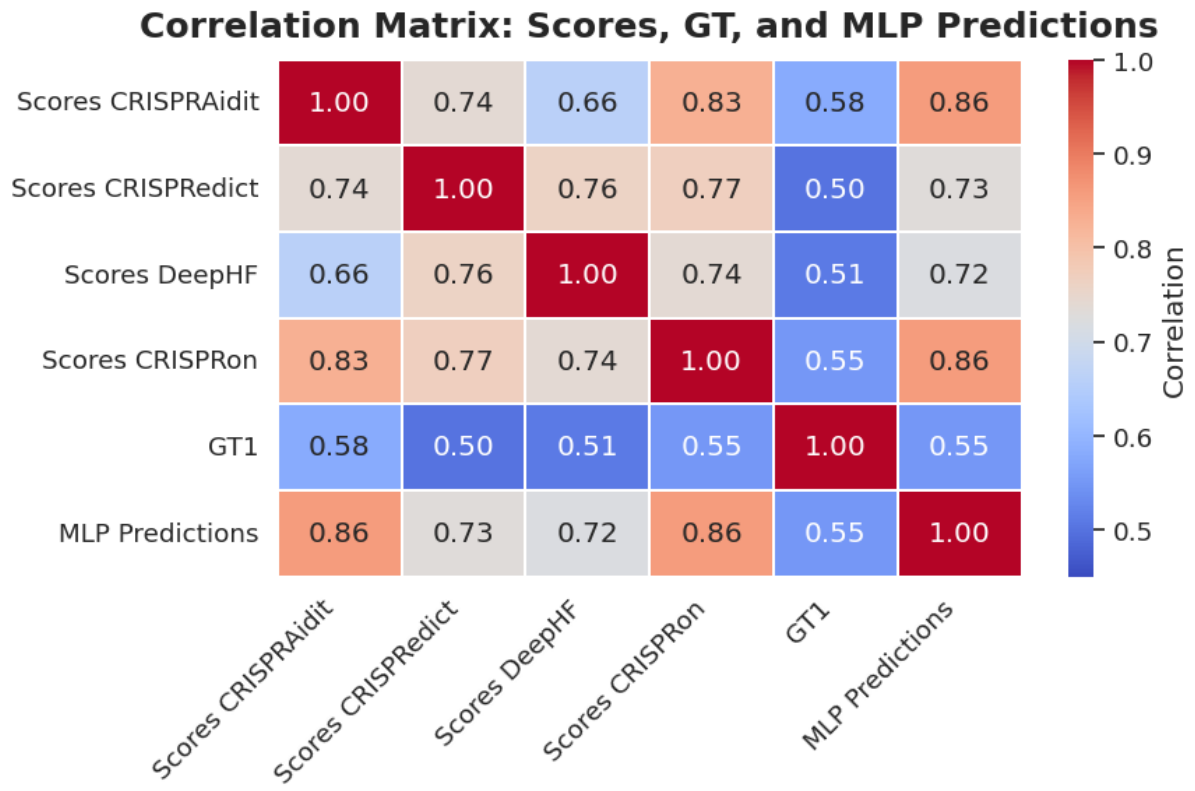


Figure 3.7: Correlation Analysis of Multilayer Perceptron (MLP) on SpCas9-Endo Dataset, model predictions vs GT shows strong relation

3.5.3 Support Vector Regression (SVR):

The predictive power of non-linear models in determining CRISPR gRNA activity scores was assessed using Support Vector Regression. The inner loop of a nested 10-fold cross-validation approach optimized across kernel types (linear, rbf), regularization parameter C, and epsilon-insensitive loss margin by hyperparameter tuning using GridSearchCV. Evaluation metrics on the test sets were calculated, and predictions were generated using the top-performing model from each fold.

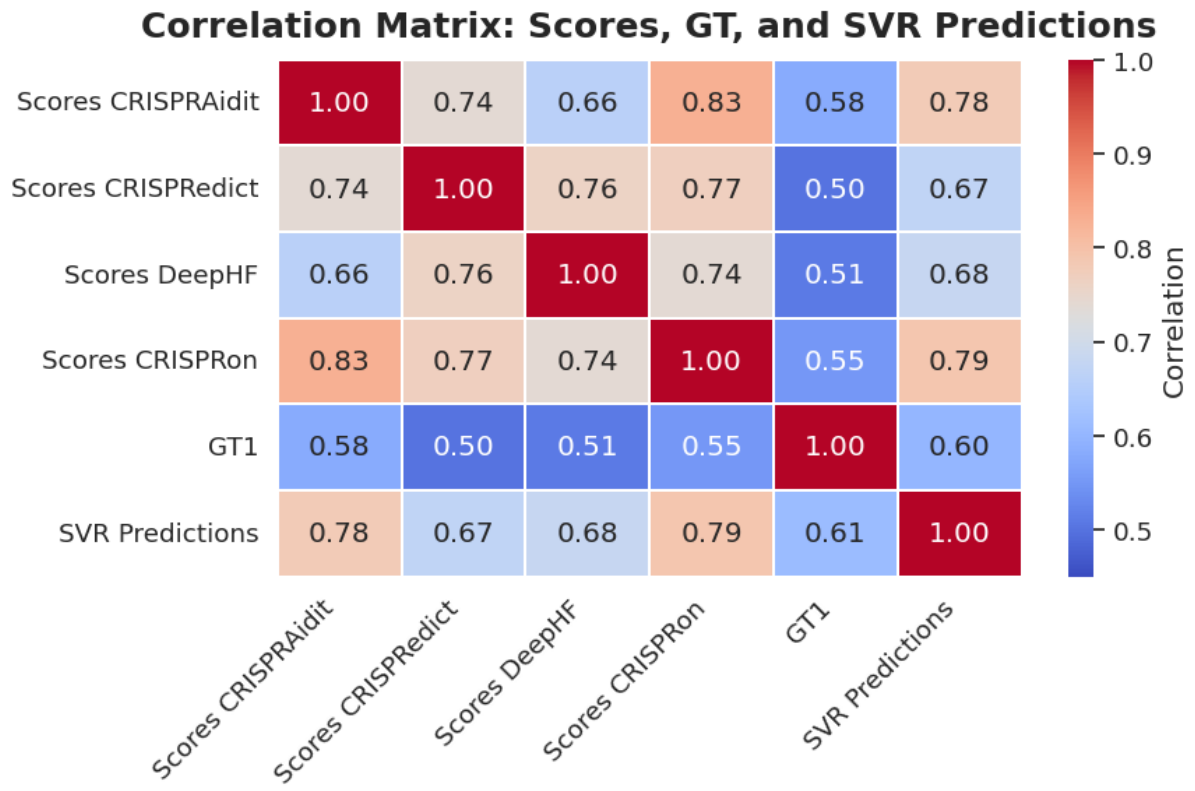


Figure 3.8: Correlation Analysis of Support Vector Regression (SVR) on SpCas9-Endo Dataset, model predictions vs GT shows strong relation.

3.5.4 Decision Tree

The effectiveness of interpretable, rule-based algorithms in forecasting CRISPR/Cas9 gRNA efficiencies was evaluated using the Decision Tree model. A nested 10-fold cross-validation approach was used, with the inner loop optimizing hyperparameters over minimum samples per split, minimum samples per leaf, and tree depth. This allowed for the selection of the best model complexity for each fold while guaranteeing objective performance estimations.

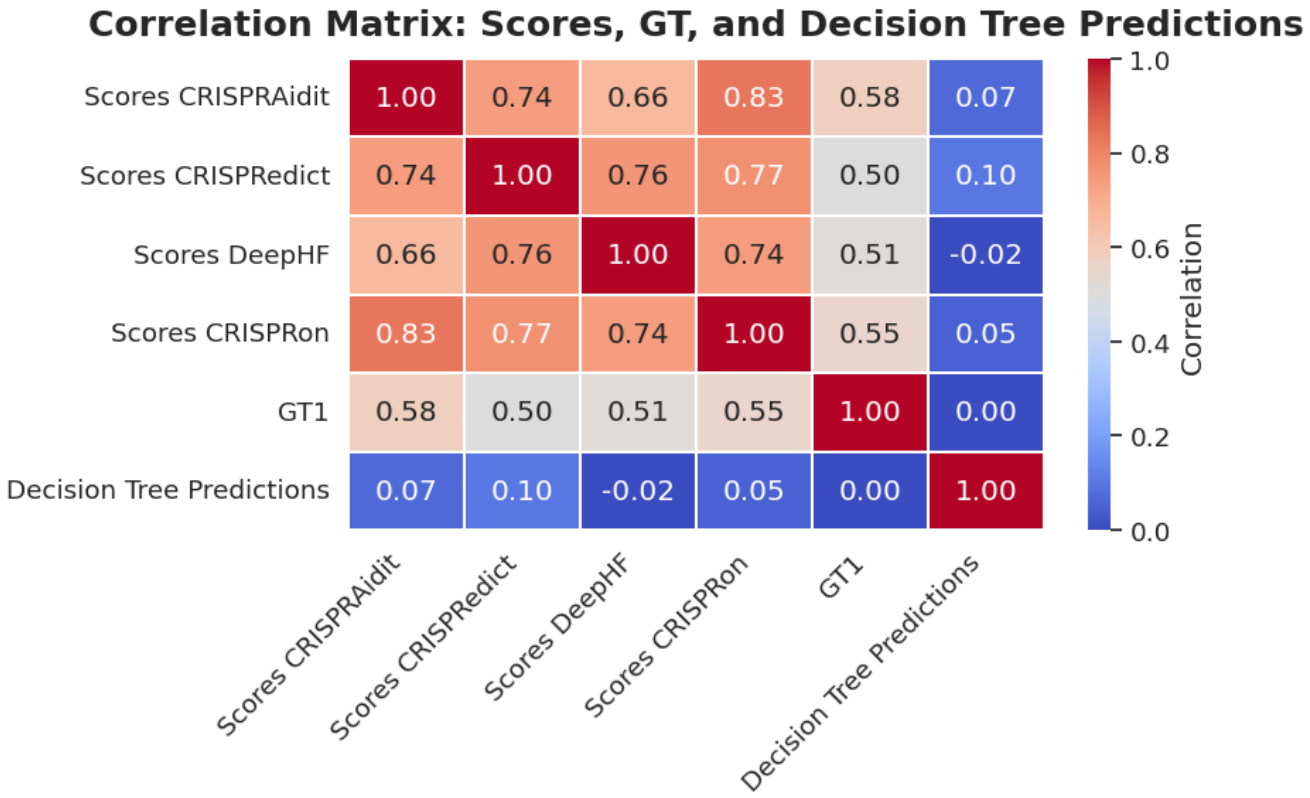


Figure 3.9: Correlation Analysis of Decision Tree onSpCas9-Endo Dataset, model predictions vs GT shows no relation.

3.5.5 Gaussian process

The capacity of the Gaussian Process (GP) model to forecast CRISPR/Cas9 gRNA efficiency while accounting for uncertainty and non-linear trends was assessed. In order to capture intricate correlations in the input features obtained from four scoring tools, it makes use of a probabilistic framework that establishes a distribution across functions and kernel functions, specifically RBF with different length scales. The noise levels (alpha) and kernel hyperparameters were tuned by inner-loop tuning and a nested 10-fold cross-validation.

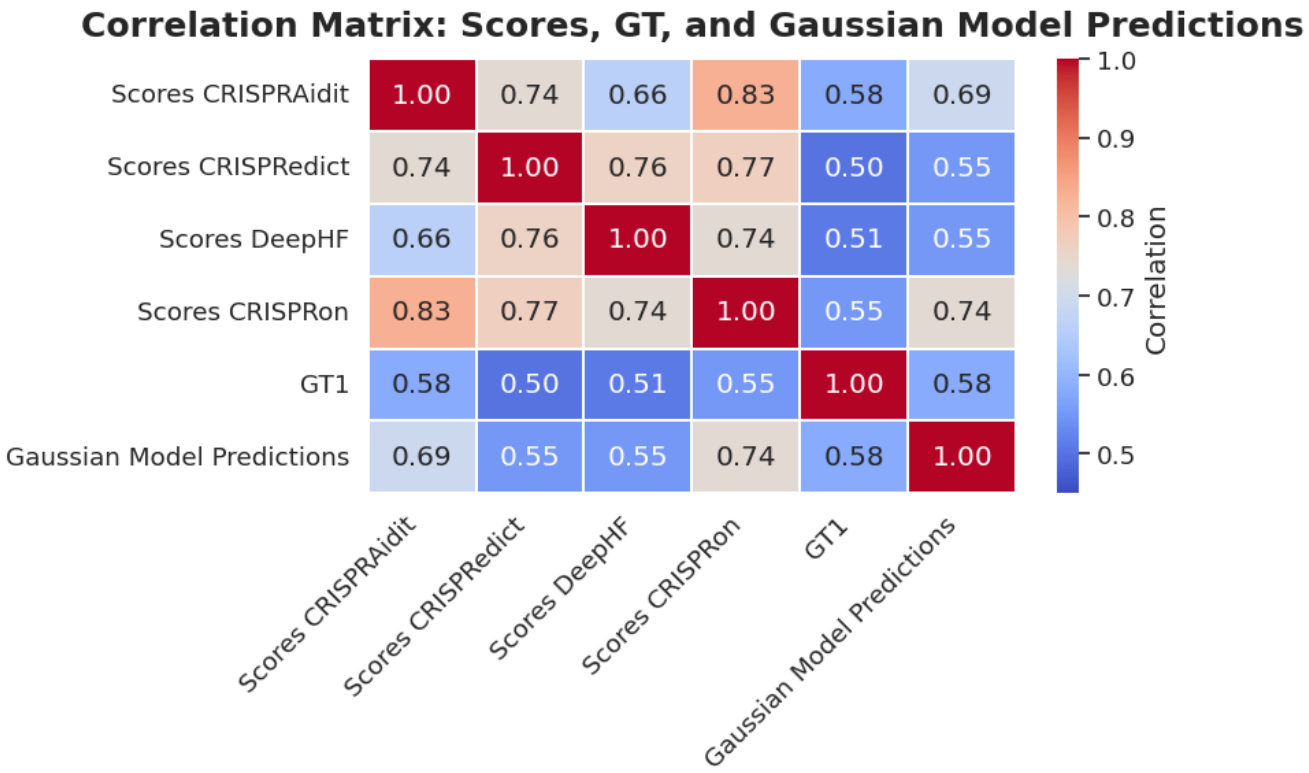


Figure 3.10: Correlation Analysis of Gaussian process on SpCas9-Endo Dataset, model predictions vs GT shows strong relation.

Choice of XG-Boost

We used XGBoost (Extreme Gradient Boosting) as the meta-regressor Table 3.1 in order to efficiently combine the results from both distinct CRISPR prediction algorithms. This approach to handling structured data is renowned for its great precision, scalability, and robustness. A 10-fold nested cross-validation architecture was used to train the model, with the inner loop using GridSearchCV to optimize hyperparameters and the outer loop assessing generalization performance. The input features formed a four-dimensional vector for each guide RNA and comprised the prediction scores from CRISPRAidit, CRISPRedict, DeepHF, and CRISPRon. In every fold, hyperparameters like the minimum sample split, learning rate, tree depth, and number of estimators were automatically adjusted. The final model generated fold-wise predictions that were subsequently utilized for each feature contribution analysis, and it demonstrated strong predictive ability as measured by R2, RMSE, and Pearson correlation as compare with Linear Model, Multilayer Perceptron (MLP), Support Vector Regression (SVR), Decision Tree and Gaussian process.

Key Outcomes

The XGBoost model demonstrated superior performance compared to individual base models in both Pearson correlation and RMSE metrics, confirming the hypothesis that model stacking can effectively enhance predictive power in complex biological regression tasks. The use of strictly separated test set predictions from 10-fold cross-validation was critical in ensuring that the reported improvements were genuine and not artifacts of overfitting or data leakage. The goal is to optimize model performance for each model, balancing the model's ability to capture complex patterns without overfitting training data.

Table 3.1: Evaluation of machine learning models' performance in predicting CRISPR/Cas9 efficiency on SpCas9-Endo dataset. Metrics include MSE, RMSE, and Pearson correlations.

Model	MSE	RMSE	Pearson (Test)	Pearson (Train)
Linear Regression	0.0397	0.1992	0.6021	0.6171
MLP Regressor	0.0441	0.2099	0.5480	0.6268
Decision Tree	0.0467	0.2162	0.5417	0.8038
Gaussian Process	0.0460	0.2144	0.5787	0.9058
SVR	0.0393	0.1984	0.6092	0.8181
XGBoost	**0.0391**	**0.1977**	**0.6111**	**0.9462**

Table 3.2: XGBoost best hyperparameters for each fold in the process 10-fold cross-validation in SpCas9-Endo dataset.

Model	n_estimators	learning_rate	max_depth	min_samples_split
Fold 1	10	0.2	3	10
Fold 2	10	0.2	3	10
Fold 3	50	0.1	3	2
Fold 4	10	0.1	3	10
Fold 5	10	0.1	7	10
Fold 6	50	0.1	3	10
Fold 7	10	0.1	3	10
Fold 8	100	0.01	3	10
Fold 9	10	0.2	3	2
Fold 10	10	0.2	3	10

3.6 Explainability

Explainability plays a critical role in genome editing by fostering confidence in model results and revealing the molecular underpinnings of gRNA efficiency. Although many ensemble and deep learning models perform well in terms of prediction, they are sometimes regarded as ”black-boxes,” providing little insight into the ways in which input attributes influence results. We used a game-theoretic method called SHAP (SHapley Additive exPlanations), which measures each feature's contribution to individual predictions, to get around this problem.

With this configuration, we were able to evaluate the ways in which each tool affected the outcome in addition to combining the predictive capacity of several models. We retrieved and displayed SHAP values for every test sample and throughout cross-validation folds in order to promote explainability at scale. Summary plots helped determine which models consistently contributed the most to the forecasts and showed the relative relevance of the input tools (features). The majority of the time, CRISPRAidit and DeepHF demonstrated greater SHAP contributions, demonstrating their hegemonic role in the ensemble decision-making process.

Furthermore, we used the SHAP additive formula to confirm the linearity of SHAP contributions to interpret the ensemble model; we recreated the XGBoost prediction using the SHAP attributions of the various base models. In three phases, we construct this derivation.

Step 1. SHAP explanation of each base model. Each of the four base predictors—CRISPR-AIdt (CA), CRISPRedict (CE), DeepHF (DH), and CRISPRon (CO)—can be locally explained using SHAP. For a given input guide, SHAP expresses the model's prediction as the sum of: a base value (bias term) and the contributions of all input features.

$$\begin{aligned} y_{\text{CA}} &= \phi_{\text{CA}} + \sum \text{SHAP}_{\text{CA}} \\ y_{\text{CE}} &= \phi_{\text{CE}} + \sum \text{SHAP}_{\text{CE}} \\ y_{\text{DH}} &= \phi_{\text{DH}} + \sum \text{SHAP}_{\text{DH}} \\ y_{\text{CO}} &= \phi_{\text{CO}} + \sum \text{SHAP}_{\text{CO}} \end{aligned} \tag{A}$$

Here, $y_{\text{CA}}, y_{\text{CE}}, y_{\text{DH}}, y_{\text{CO}}$ are the output scores predicted by CRISPR-AIdt, CRISPRedict, DeepHF, and CRISPRon, respectively. The terms $\phi_{\text{CA}}, \phi_{\text{CE}}, \phi_{\text{DH}}, \phi_{\text{CO}}$ are the corresponding base values (bias terms), and each $\sum \text{SHAP}_{\text{XX}}$ is the total contribution of

all input features within that model.

Step 2. How XGBoost combines the base models. The meta-learner (XGBoost) uses the outputs of the four base models as its inputs and produces the final ensemble score y_{XG}. Because SHAP provides an additive explanation of any tree-based model, the final prediction of XGBoost can be written as a linear combination of the contributions coming from its four input models (CA, CE, DH, CO). This means that the output y_{XG} can be decomposed into a base value and four weighted contributions—one for each base predictor—allowing the meta-learner to be expressed in an interpretable linear form. We can express this as:

$$y_{\mathrm{XG}} = \phi_0 + \alpha_{\mathrm{CA}}\, y_{\mathrm{CA}} + \alpha_{\mathrm{CE}}\, y_{\mathrm{CE}} + \alpha_{\mathrm{DH}}\, y_{\mathrm{DH}} + \alpha_{\mathrm{CO}}\, y_{\mathrm{CO}} \tag{B}$$

In this formulation, ϕ_0 is the global base value (bias) of the XGBoost ensemble model, and $\alpha_{\mathrm{CA}}, \alpha_{\mathrm{CE}}, \alpha_{\mathrm{DH}}, \alpha_{\mathrm{CO}}$ are the relative contribution weights of each base model. Each summation term corresponds to the total contribution of the respective model to the ensemble prediction. This formulation provides an interpretable estimate of each model's influence on the ensemble: higher α_{XX} values indicate stronger contribution of that model to the XGBoost prediction for the corresponding guide RNA. Thus, the corresponding coefficient α_{XX} from Eq(B) is computed as:

$$\alpha_{\mathrm{XX}} = \frac{\mathrm{SHAP}_{\mathrm{XX}}^{(\mathrm{XG})}}{y_{\mathrm{XX}}} \tag{C}$$

where each coefficient α_{XX} ($\mathrm{XX} \in \{\mathrm{CA}, \mathrm{CE}, \mathrm{DH}, \mathrm{CO}\}$) represents the relative contribution of a base model—CRISPR-AIdt, CRISPRedict, DeepHF, or CRISPRon—to the final ensemble prediction. These coefficients are not fixed constants but sample-specific empirical weights estimated from the SHAP values of the XGBoost meta-learner. For each sample, the contribution of a base model XX to the ensemble output is given by its SHAP value $\mathrm{SHAP}_{\mathrm{XX}}^{(\mathrm{XG})}$.

Step 3. Substituting the SHAP forms of the base models. Replacing each y_{XX} in Eq. (B) with its SHAP decomposition from Eq. (A) gives:

$$\begin{aligned} y_{\mathrm{XG}} = \phi_0 &+ \alpha_{\mathrm{CA}} \left(\phi_{\mathrm{CA}} + \sum \mathrm{SHAP}_{\mathrm{CA}}\right) + \alpha_{\mathrm{CE}} \left(\phi_{\mathrm{CE}} + \sum \mathrm{SHAP}_{\mathrm{CE}}\right) \\ &+ \alpha_{\mathrm{DH}} \left(\phi_{\mathrm{DH}} + \sum \mathrm{SHAP}_{\mathrm{DH}}\right) + \alpha_{\mathrm{CO}} \left(\phi_{\mathrm{CO}} + \sum \mathrm{SHAP}_{\mathrm{CO}}\right) \end{aligned} \tag{3.5}$$

Equation (3.5) is the final SHAP aggregation formula. It shows that the ensemble prediction y_{XG} can be reconstructed as a **weighted sum of two components for each base model**: its base value ϕ_{XX} and the sum of its feature-level SHAP contributions $\sum \text{SHAP}_{XX}$, each scaled by its corresponding coefficient α_{XX}, together with the ensemble bias ϕ_0.

This relationship persisted across samples, demonstrating that the XGBoost output was reliably broken down into interpretable feature effects by the SHAP values. Explainability was not restricted to a particular analytical point. To help both expert and non-technical audiences understand the underlying workings of our model, we created SHAP waterfall plots, summary plot, and Pearson correlation plots.

Overall, our multi-model CRISPR/Cas9 system gained an interpretable layer of analysis that connects biological insight and prediction performance through the incorporation of SHAP explainability to interpret model predictions. SHAP assigns each feature a contribution value, allowing the prediction to be expressed as the sum of a base value and feature contributions. This additive property enables clear interpretation of both individual models and the ensemble, while preserving biological relevance.

Method validation

We used 10-fold cross-validation, a commonly used statistical validation technique in machine learning and bioinformatics, to provide a reliable and objective assessment of the predictive performance of our models. Mitigating overfitting and generalizing model performance across unknown data are the main goals of cross-validation. The complete dataset is divided into ten equal-sized subgroups (or ”folds”) at random for 10-fold cross-validation. Nine of the folds are utilized for training, while one is kept out as the test set for each iteration. Every fold is used exactly once as the test set during the ten repetitions of this procedure. This lowers the variation related to performance measurements by guaranteeing that all data points are used for both training and evaluation.

Based on the predicted scores from four CRISPR tools—CRISPRedict, CRISPRAidit, DeepHF, and CRISPRon—we employed 10-fold cross-validation for our analysis across all models, including linear regression, decision trees, Gaussian Process Regression, MLP, SVM and XGBoost. We used measures like the R2 score, RMSE, and Pearson correlation coefficient to assess the model on the matching test set after training it on the training data inside each fold. We were also able to monitor fold-specific hyperparameters through this technique, as demonstrated by GridSearchCV for XGBoost

and Guassian process regressor. In order to reduce reliance on a single data split and provide a thorough evaluation of generalization ability, the final performance of each model was presented as the average of the evaluation metrics across all 10 folds.

This approach guarantees that our published findings are representative of the actual model capacity across the dataset and are statistically sound.

3.7 Experimental results

3.7.1 Framework on SpCas9-Endo dataset (KO Data)

In the study's first experiment, four cutting-edge CRISPR/Cas9 prediction tools—CRISPRAidit, CRISPRedict, DeepHF, and CRISPRon—were tested on a curated SpCas9-Endo dataset made up of 79 guide RNAs (gRNAs). These tools are frequently used to forecast CRISPR efficacy in a variety of biological systems and were first trained on extensive indel efficiency datasets. The goal was to comprehend the individual and aggregate prediction behavior of these pretrained models and assess how well they generalize to independent knockout data. An XGBoost-based meta-model was used to aggregate the predictions made by the four tools after they were compared to the ground truth (GT). Correlation matrices were used to further assess the results, and the Gradient Boosting model's final predictions showed a significant improvement in correlation with GT1 when compared to all tools.

For example, the meta-model prediction obtained a correlation of 0.611, showing a very accurate and consistent output, whereas CRISPRAidit and CRISPRon demonstrated the strongest individual connection with GT1. This experiment supported the idea that the combination of several scoring methods increases the reliability of the prediction, especially when rigorous valid biological techniques and ensemble learning are used for the analysis.

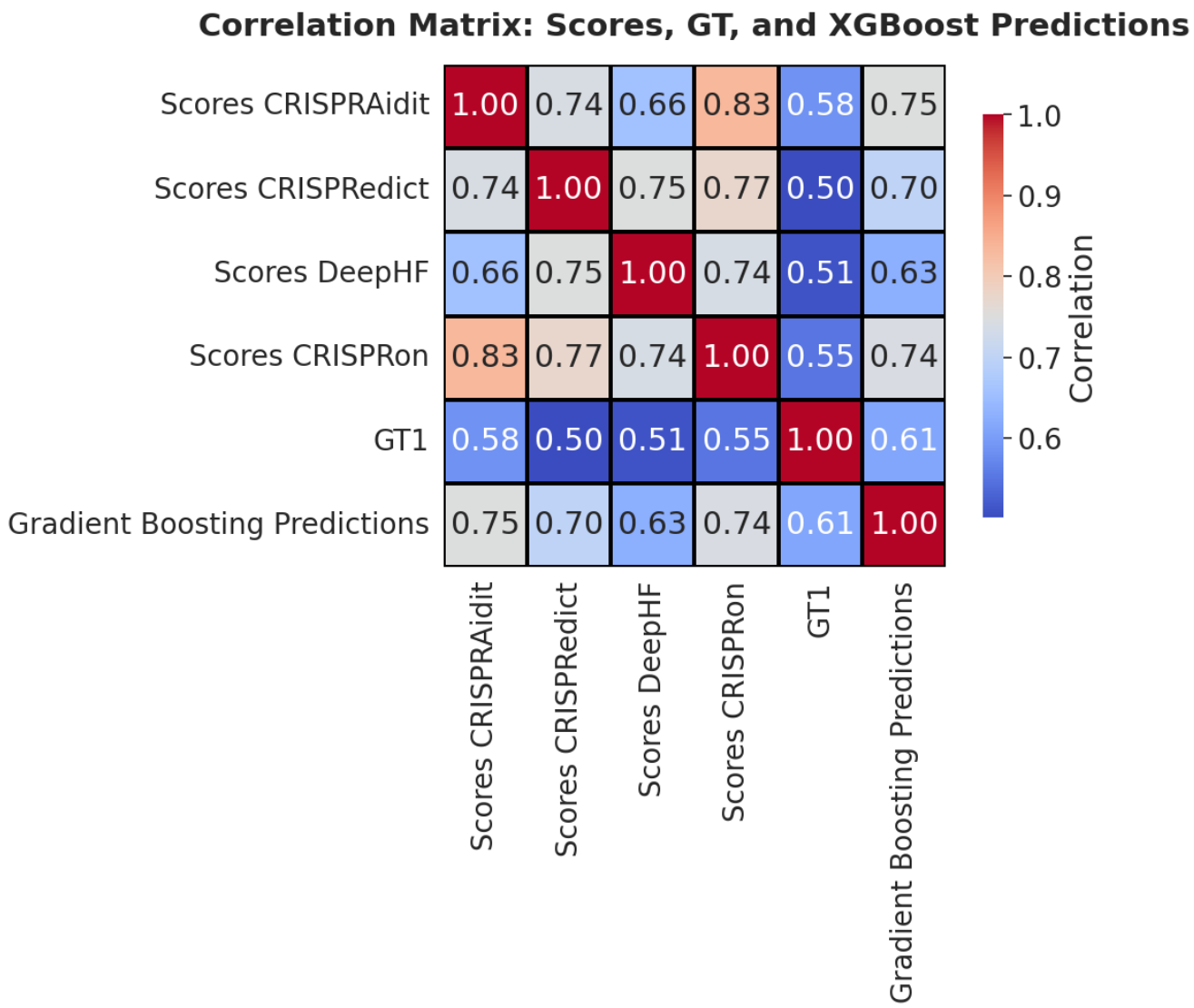


Figure 3.11: Correlation matrix on SpCas9-Endo dataset. The figure shows that XGBoost predictions have the highest correlation with the ground truth, indicating superior predictive performance compared to individual models.

3.7.2 Experiment 1: Transferring the Pre-trained Model on BE Data

In this experiment, we tested the efficacy of applying pretrained models created for CRISPR knockout (KO) editing to CRISPR base editing (BE) data in this experiment. In particular, without retraining or fine-tuning, we applied models that had already been trained on the Knockout dataset to the Hanna dataset. As ground truth values, z-scores are included in the Hanna dataset to measure editing efficiency in base editors. Using the pretrained models, we produced predictions and assessed their performance using the conventional regression metrics of R2 (coefficient of determination), RMSE (root mean squared error), and Pearson correlation coefficient across ten folds

as shown in Table 3.3. The outcomes were consistently poor in each fold. The models' performance was worse to that of a horizontal mean predictor, as indicated by the majority of R2 values being negative. Pearson correlation coefficients were near zero, indicating a nonlinear relationship between expected and actual values, while RMSE values were high, highlighting prediction error.

These poor results clearly shows a significant mismatch between the source (KO) and target (BE) domains. This supports the theory that **knowledge acquired for in/del frequency prediction in CRISPR Knockout cannot be reused for CRISPR base editing**.

This is probably because the biological mechanisms at play are essentially different. CRISPR KO editing mainly uses double-strand breaks to introduce insertions and deletions (indels), while CRISPR base editors convert single nucleotides without doing so. As a results, Predictive patterns derived from KO data are therefore not transferable to base editing scenarios.

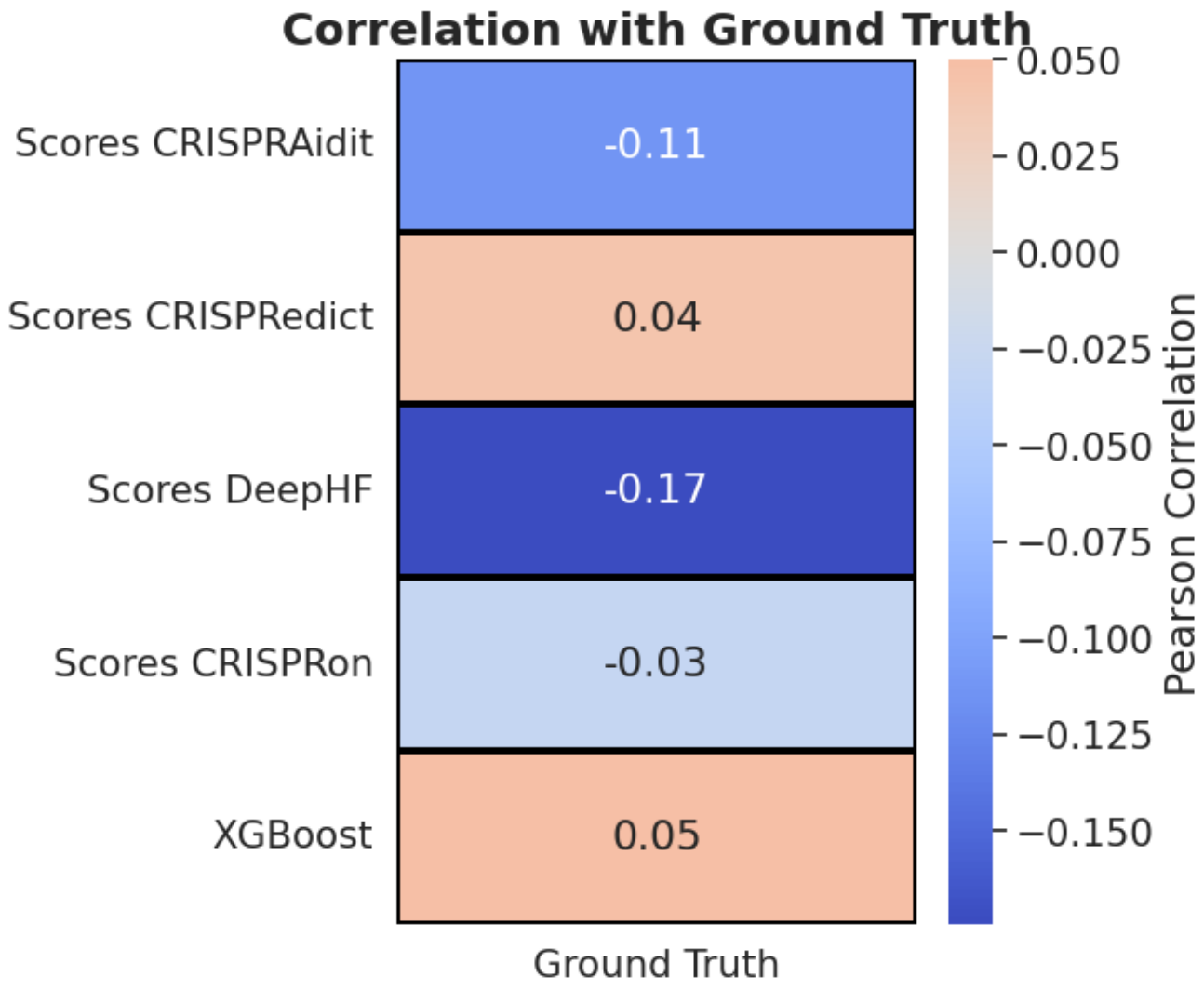


Figure 3.12: Correlation matrix on Hanna dataset using pre-trained crispr models vs XG-Bosst and GT showing week corelation.

Table 3.3: Per-fold performance metrics of the XGBoost model applied to predictions taken from the Hanna dataset after it was trained on the Knockout dataset. The model was assessed directly using its pretrained state without retraining on Hanna.

Fold	R^2 (Test)	RMSE (Test)	Pearson (Train)
Fold 1	-0.2452	1.7567	0.0499
Fold 2	-0.2948	1.6832	0.0471
Fold 3	-0.2199	1.199	-0.0258
Fold 4	-0.2340	1.6609	0.0405
Fold 5	-0.2095	1.5267	0.0406
Fold 6	-0.1828	1.5161	0.0357
Fold 7	-0.2471	1.7319	-0.0465
Fold 8	-0.2119	1.8624	0.0390
Fold 9	-0.1875	1.5619	0.0310
Fold 10	-0.1823	1.5176	0.0372
Average	**-0.2214**	**2.7078**	**0.0402**

3.7.3 Experiment 3: Re-training All the Models and meta learner

As an attempt to reuse the knowledge of base models in the BE domain, we conducted an intermediate experiment in which the four base models are pretrained on the KO domain, and only the XGBoost meta-learner was first used as a pretrained metaleraner with pretrained base models, the retrained on the retrained base models BE predictions (Fig.3.13). This setup produced only limited improvement compared to direct KO$\rightarrow$BE transfer: the meta-learner adapted slightly, but overall performance remained very little from negitive corelation to no effect that of the fully retrained models. This confirms that effective BE prediction requires not only retraining the meta-learner but also retraining the base models themselves. To address RQ2 and overcome the domain mismatch identified in the cross-domain transfer experiment, all base models as well as the XGBoost meta-learner were re- trained on the Hanna dataset using 10-fold cross-validation. For each fold of the cross-validation procedure, the following protocol was imple- mented:

1. Training predictions from the base models were collected independently.

2. Test predictions from these four base models were similarly obtained.

3. These predictions were aggregated into a new training-testing feature set, with each guide RNA represented by a four-dimensional vector of base model predic- tions.

4. The XGBoost meta-learner was trained on the combined training predictions and evaluated on the corresponding test predictions, thereby ensuring strict separation between training and evaluation data to prevent data leakage. The results demonstrated marked improvement across all models following domain-specific retraining (Table 3.4. The results demonstrated marked improvement across all models following domain-specific retraining. Among the individual base models, CRISPRon achieved the highest Pearson correlation coefficient (r = 0.6915) with the lowest root mean squared error (RMSE = 1.1940), indicating superior generalization capacity. DeepHF exhibited the second-best performance (r = 0.5545, R^2 = 0.4561), suggesting that architectures incorporating thermodynamic and biophysical features demonstrate enhanced transferability to the base editing domain. In contrast, CRISPRedict and CRISPR-AIdt yielded modest correlations (r = 0.4152 and r = 0.1650, respectively), reflecting greater domain specificity in their learned representations.

The XGBoost ensemble meta-learner outperformed remaining other three base models, achieving the highest Pearson correlation coefficient (r = 0.6220) after crisprOn. From (Table 3.4), the correlation analysis revealed strong positive associations be-

tween the ensemble predictions and ground truth values, confirming the efficacy of the stacking approach.

In order to get cover the above limitations, all models were methodically retrained on the Hanna dataset, which comprises 5062 guide RNAs (gRNAs) with empirically determined z-score-based indel efficiencies. We used 10-fold cross-validation to completely retrain each model. The goal was to evaluate these models' ability to forecast and adjust to the Hanna dataset when retrained from scratch as opposed to using pre-trained weights from other datasets.

Four representative models were selected for this retraining procedure: **CRISPRedict, CRISPRAidit, DeepHF, and CRISPRon.** Every model was retrained according to the reasoning specified in its relevant publication, utilizing its original feature encoding and model architecture. Using a stratified K-Fold technique, the Hanna dataset was divided into 10 folds for all models, guaranteeing a fair distribution of efficiency scores among each fold. The procedure was repeated ten times to guarantee statistical robustness, using nine folds for training and one for testing in each iteration.

Table 3.4: The model's overall performance was assessed using 10-fold cross-validation on the Hanna dataset after it was retrained. The table shows the average results across all folds.

Model	Pearson (Test)	RMSE (Test)	MSE (Test)	R^2 (Test)
CRISPR-AIdt	0.4152	1.3577	1.8535	0.1650
CRISPRedict	0.4891	1.2996	1.6888	0.2391
CRISPRon	0.6915	1.0870	1.1940	0.3753
DeepHF	0.5545	0.8822	0.7783	0.4561
XGBoost	0.6220	0.8222	0.6950	0.3705

We used a meta-learning approach was used using XGBoost, to combine and enhance the predictive performance of four distinct CRISPR scoring tools: CRISPRAidit, CRISPRedict, DeepHF, and CRISPRon. The objective was to retrain this meta-learner with each tool's outputs as features and assess how well it predicted actual base editing efficiencies. Using distinct training and test datasets, individual models were trained for each of the ten folds in the cross-validation pipeline. For the training set, the features comprised the prediction scores from the four tools (Scores-CRISPRAidit-train, Scores-CRISPRedict-train, Scores-DeepHF-train, and Scores-CRISPRon-train) and corresponding results for the test set. The experimentally observed base editing efficiency, expressed as a z-score (from the Hanna dataset), was the target variable in each case.

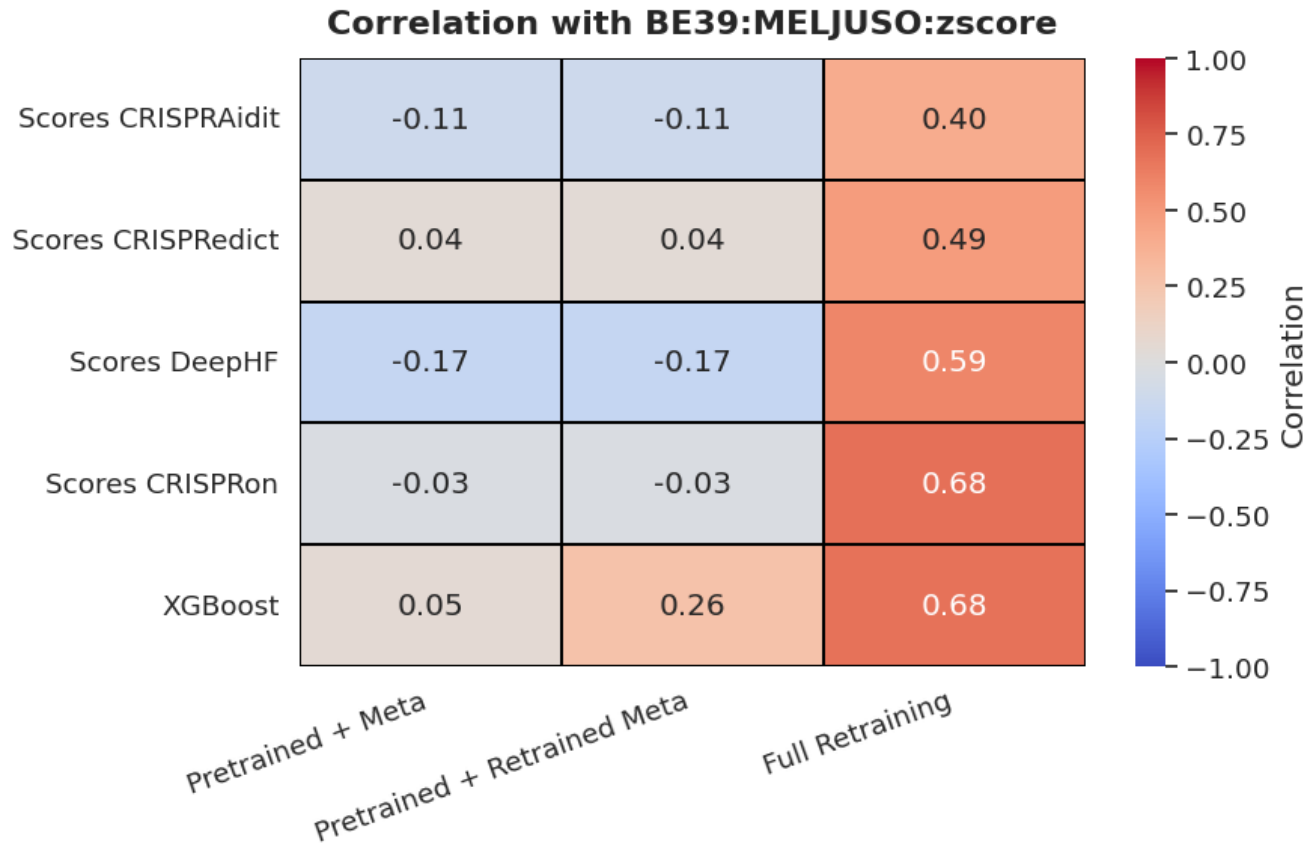


Figure 3.13: Correlation matrix on Hanna dataset using retrained CRISPR models on 10-foldCV Vs XG-Bosst and GT which showing the improved performance of XGBoost in three experimnts.

Table 3.5: Per-fold performance metrics of the XGBoost model trained on test predictions from CRISPRAidit, CRISPRedict, DeepHF, and CRISPRon tools, evaluated using 10-fold CV on the Hanna dataset.

Fold	R^2 (Test)	MSE (Test)	RMSE (Test)	Pearson (Test)	Pearson (Train)
Fold 1	0.454446	1.352097	1.162797	0.677624	0.771662
Fold 2	0.398043	0.697716	0.835294	0.631384	0.752011
Fold 3	0.409383	0.542049	0.736240	0.640865	0.776358
Fold 4	0.376490	0.650345	0.806440	0.615105	0.762607
Fold 5	0.411177	0.526763	0.725784	0.641790	0.766609
Fold 6	0.097215	0.858472	0.926538	0.497104	0.763463
Fold 7	0.392044	0.754107	0.868394	0.628103	0.758758
Fold 8	0.391467	0.570544	0.755344	0.629012	0.774860
Fold 9	0.415159	0.398285	0.631098	0.657322	0.749032
Fold 10	0.359902	0.599123	0.774030	0.601612	0.772938
Average	**0.3705**	**0.6950**	**0.8222**	**0.6220**	**0.7648**

3.8 Interpretability Analysis

We performed SHAP (SHapley Additive exPlanations) analysis on the Knockout and Hanna datasets for each of the individual tools, CRISPRedict, CRISPRAidit, CRISPRon, and DeepHF, to improve the interpretability of our ensemble prediction framework. The SHAP waterfall plots that show how top-ranked features locally contribute to individual sequence predictions are shown in Figures 3.14 to 3.21.

CRISPRedict Interpretability Results Analysis:

The SHAP waterfall plot for CRISPRedict on the Knockout dataset is shown in Figures 3.14. The Hanna dataset's comparable plot is displayed in figure 3.15. The features **G_20** and **C_18** were consistently found to be influential in both datasets. **A_middle, polynucleotides**, and **AA count** all have a negative impact on prediction in the knockout plot, which lowers the expected efficiency. Interestingly, the projected score was marginally raised by the existence of motifs and the AAA count, suggesting a possible stabilizing sequence feature. Conversely, the Hanna dataset highlighted the positive impact of **G_middle**, while **ATT_ count, A_middle, AG_ count**, and **AC_end** made substantial contributions to decreasing the prediction. This suggests that middle-position nucleotides and short motif-based sequence patterns are important for defining editing efficiency across contexts.

These shap water fall plots demonstrate that the simple linear model of CRISPRedict provides very interpretable results, with patterns that are biologically believable in both datasets.

CRISPRAidit Interpretability Results Analysis:

SHAP values for CRISPRAidit on the Knockout dataset are shown in Figure 3.16. While SHAP values for CRISPRAidit on the Hanna dataset are shown in Figure 3.17. A greater range of contributions is displayed by CRISPRAidit's model (473 features), which is indicative of the non-linear MLP architecture and deep feature set. In the Knockout dataset, **AG17, A17** significantly raised the score, whereas **TA05, GA19**, and **GA13** had the greatest negative impacts. It's interesting to note that the total contribution of 464 features was significantly negative, highlighting the great influence of a small number of important motif-encoded features on the model's output. The distribution on the Hanna dataset is more evenly spread. The score was lowered mainly by features such as **GC_Content, T22, M22, A21, C20, and CG20**, while it was slightly raised by **G21** and **C22**. Notably, 464 additional features still showed an overall negative contribution, suggesting that the model is influenced by a broad set

of weak signals when handling less expressive input patterns. These results indicate that CRISPRAidit captures complex position-specific and composition-related effects, but also highlight the importance of SHAP for identifying which features truly drive the prediction.

CRISPRon Interpretability Analysis Results:

The CRISPRon SHAP waterfall visualization on the Knockout dataset is displayed in (Fig. 3.18), while in (Fig. 3.19), the Hanna dataset SHAP waterfall visualization is represented.

From Knockout SHAP plot (Fig. 3.18), a strong negative contributions were made by the features at positions **24_G, 24_C**, and **22_C**, highlighted the importance of 3' end bases. Bases at positions **30_G and 21_A**, on the other hand, raised the prediction score, which is consistent with other research that links Cas9 efficiency to the protospacer's distal end.

The SHAP values are more detailed in the Hanna dataset (Fig. 3.19). Features like **Pos20_T, Pos22_C, Pos18_T, Pos6_A, and Pos17_A** showed notable negative contributions, while **Pos14_G, Pos3_T, Pos19_A, and Pos11_A** demonstrated mildly favorable effects. The cumulative contribution of 102 additional features was slightly positive, indicating a more distributed and balanced influence of features compared to the Knockout dataset. Overall, CRISPRon exhibits strong position-sensitive behavior, with specific nucleotides across central and distal regions contributing to prediction, which is scientifically meaningful for both biologists and bioinformaticians.

DeepHF Interpretability Results Analysis:

SHAP values for DeepHF on the Knockout dataset are shown in (Fig. 3.20), While SHAP values for DeepHF on the Hanna dataset are shown in (Fig. 3.21).

DeepHF's SHAP waterfall plots, with their 11-feature input space, provide a more condensed interpretation.**Tm_PAM_Proximal, GC_Content** and **Tm_Middle** all had a negative effect on the score in the Knockout dataset, indicating that excessively stable melting temperatures around the PAM lower efficiency. It's interesting to see that **Stem_Loop** made a small positive contribution, suggesting that the projected gRNA secondary structure may be advantageous. A strong negative effect from **Tm_Middle** and **T_Content** was observed in the Hanna dataset, indicating reduced efficiency associated with unfavorable thermodynamic profiles. Additional features such as **Base_Accessibility, Tm_Global,** and **Stem_Loop** also contributed negatively, highlighting the importance of structural and thermodynamic stability for successful editing. Moreover, **GC_Content**

and **G_Content** showed decreasing effects, suggesting that excessive GC richness may hinder CRISPR binding efficiency. In contrast, **C_Content** showed a slight positive contribution. These results emphasize that both sequence composition and structural accessibility are key factors influencing DeepHF predictions, particularly as genomic accessibility and thermodynamic balance directly affect CRISPR–DNA interactions. These findings demonstrate that DeepHF provides strong cross-dataset generalization by capturing biophysical and thermodynamic concepts.

Validation of SHAP Linearity Across SpCas9-Endo and Hanna Datasets:

The linear relationship between SHAP-reconstructed outputs (aggregated from CRISPRedict, CRISPRAidit, CRISPRon, and DeepHF) and the real XGBoost predictions was examined in order to assess the dependability of SHAP values in reconstructing the XGBoost meta-learner's predictions. Both the SpCas9-Endo and Hanna datasets were used for this comparison, as seen in Figures 3.22 and 3.23, respectively. An almost perfect linear relationship is seen for the SpCas9-Endo dataset, with a Pearson correlation coefficient of 0.9987 and R^2 = 0.9966. This strong agreement attests to the SHAP-aggregation formula (Eq.3.5).

In low-complexity conditions, it successfully reconstructs the XGBoost predictions. The very small dataset size (79 gRNAs) and reduced biological variability, which enable the model to learn more linear and readily explicable patterns, are responsible for the virtually perfect alignment.

Moreover, the results for the Hanna dataset, which includes 5062 gRNAs, the SHAP-reconstructed predictions nevertheless exhibit good agreement with the XGBoost outputs, despite the perfect linearity, as evidenced by a Pearson correlation of 0.9976 and R^2 0.9952. Although the explanation is still strong, the little decrease in correlation is a reflection of the higher sequence variety and non-linear interactions found in large-scale biological datasets. These findings demonstrate that SHAP values are an excellent interpretability tool in ensemble modeling since they scale well from simple to complicated input spaces and retain high fidelity with XGBoost outputs across datasets.

Common SHAP Features for All Tools and Datasets:

We performed the positional and nucleotide-level SHAP analyses for both the datsets SPCas9-Endo (KO) and Hanna (BE) datasets which show clear and dataset-dependent patterns in how sequence features influence model predictions. In the SpCas9-Endo dataset, the positional heatmap in 3.24a indicates that a small num-

ber of guide positions—particularly those close to the PAM—dominate the model's contribution profile, reflecting the known positional sensitivity of Cas9. The nucleotide-specific aggregation in Figure 3.24b further shows that bases A and T contribute more strongly than G and C, suggesting a compositional bias learned from this dataset.

For the Hanna dataset, the positional importance for features are shown in Figure (Fig. 3.25a), which shows only a subset of positions drives most of the predictive signal with respect to each nucleotide. However, the nucleotide-level contributions in Figure 3.25b are more evenly balanced across A, T, G, and C compared with SpCas9-Endo. This indicates fewer nucleotide-specific biases in the Hanna dataset, likely due to its broader sequence diversity and the characteristics of base-editing data. Overall, these two figures demonstrates that positional effects consistently dominate across datasets, while nucleotide-level contributions vary depending on the underlying experimental context from KO to BE knowledge.

The top 20 SHAP features for the SpCas9-Endo and Hanna datasets were taken using four prediction tools (CRISPRedict, CRISPRAidit, CRISPRon and DeepHF) and are shown in Figures 3.26 and 3.27 respectively. These features were chosen because they were present in our four models. When they were the same or had biological similarities (for example, position-based or motif-based), their SHAP values were added together and combined through SHAP*ALPHA contribution then sum and then absolute. The highest contributions were associated with specific positional features, like *T_end (Combined)*, *G_18 (Combined)*, *C_11 (Combined)* and several C- and G-related positions (e.g., *CA_TA05*, *CA_CA18*, *G_20 (Combined)*). These results indicate that both 3' terminal composition and certain PAM-proximal nucleotides play important roles in the model's predictions from the Knockout dataset in Figure 3.26. These positions are close to the guide RNA's 5' or 3' ends, which are essential for the stability and activity of Cas9 binding. This dataset's SHAP contributions are often lower, which reflects the reduced sample size while still emphasizing important biological signals. The literature has consistently demonstrated that certain characteristics, such as G_20 and C_19, can impact Cas9 efficiency by altering structural accessibility or guide-target hybridization. On the other hand, Figure 3.27 shown the most common features on the Hanna dataset, which has 5062 gRNAs. Because the filtered dataset focuses on functionally relevant guides, the SHAP absolute sum values are more pronounced and informative in this case. The top features that dominate the landscape are **G_21 (combined), CO_Pos18_G, CO_Pos23_A, and G_20 (combined)**, along with positional and compositional features such as **CA_GC_Content** and **CO_Pos5_G**. These results highlight the importance of both positional nucleotides and sequence composition, partic-

ularly across PAM-distal and central gRNA regions, illustrating a more intricate yet reliable identification of the biological signals influencing base editing efficiency.

SHAP explanation analysis of the ensemble

Single sequence analysis

SHAP (SHapley Additive exPlanations) analysis was used to assess the models' interpretability in both KO and BE domains by examining the predictions made on a single gRNA sequence of 23nt, extended with 20 nucleotides upstream and downstream, from which the input for all the base models is determined. As an example, we considered the sequence:

```
CAGATTAGTCGATGCAAATG TTAAAAACGTGTTGGCAAGCTGG ATGAGCAGGTTTGTATTTTA
```

where the middle part corresponds to the 23nt gRNA sequence. We first aggregated the SHAP values from the four base models (CRISPRAidit, CRISPRedict, DeepHF, CRISPRon) into the XGBoost using the linear SHAP aggregation formula (Eq. 3.5) for both the datasets KO and BE.

For the KO dataset, the waterfall plot shows how the top–10 features in the XGBoost meta-learner contribute to the final prediction for a single gRNA. (Fig. 3.28)

Each bar in the waterfall plot represents the contribution of a single feature (i.e., $\alpha \times \mathrm{SHAP}$), added on top of the baseline term $(\phi_0 + \Sigma\alpha \cdot \mathrm{Base})$. Adding the baseline term and all feature contributions exactly reconstructs the final XGBoost output for the sequence:

$$y_{\mathrm{xg}} = 0.3147.$$

This confirms the perfect linearity of the ensemble and clearly highlights which KO features most strongly influence the model's prediction.

For the BE each bar in the waterfall plot represents a single feature's weighted contribution $(\alpha \times \mathrm{SHAP})$, showing how much it increases or decreases the model output relative to the baseline. The top 10 BE features exert the strongest directional influence, either pushing the prediction upward or pulling it downward 3.29.

Adding the baseline term and all cumulative feature contributions perfectly reconstructs

the final XGBoost prediction for this sequence

$$y_{\mathrm{xg}} = -10.2151,$$

confirming the linear and interpretable structure of the ensemble. Overall, the BE waterfall plot clearly identifies the dominant nucleotide–level signals that most strongly guide the XGBoost estimator for this sample.

The SHAP analysis of the ensemble indicated that the models behave the same in the two editing domains interestengly. We identify the best-performing tool in the XGBoost meta-learner by summing the absolute SHAP values of each tool across all features and all sequences thoughout the datsets both KO and BE, where the tool with the highest total SHAP impact is considered the most influential and best performing. In the KO dataset (SPCas9-Endo) crisprOn was the best-performing model (Fig.3.30) and its SHAP values were spread across different features. Other KO models, such as CRISPRedict and CRISPR-AIdt, focused more on clear biological signals around the PAM and seed region, while DeepHF relied mainly on thermodynamic stability features.

In the Hanna dataset(BE), the best performer was crisprOn also, and its SHAP patterns showed meaningful contributions from both positional and contextual sequence features. BE explanations were more dispersed overall, reflecting the more complex sequence requirements of base editing (Fig. 3.31).

The SHAP analysis of the XGBoost ensemble showed that it use all four tools equally just based on its alpha shap contribution towards the meta learner. Instead, it gives the highest importance to CRISPRon (the best model in the KO and BE domain) This means the ensemble learns to rely on the most reliable models, producing stronger and more stable predictions than any single tool alone.

The top combined SHAP features across the four tools are interpreted in Table 3.6 to help better understand the biological importance of these shared common features. The table displays a feature group, the tool in which it was found, and a biological explanation for each row. We initially collected these SHAP features from four models: 473 from CRISPR-AIdit, 29 from CRISPRedict, 121 from CRISPRon, and 11 from DeepHF, resulting in a total of 634 SHAP features. To identify common patterns across models, we matched features based on shared positional and nucleotide information. For example, features such as *CA_G20*, *CO_Pos20_G*, and *CE_G_20* were grouped together as **G_20_combined**, a known hotspot near the seed region linked to cleavage efficiency, also, C_19_combined, which may influence specificity near the

PAM-proximal region, and T_end_combined, which may be related to 3′-end stability or transcription termination. This process allowed us to merge equivalent features from different tools into unified combined features, enabling consistent comparison and interpretation across models.

Overall, these results show that biologically significant traits are consistently given priority across various datasets and technologies. These common features and common tools are supported by established biochemical and structural mechanisms linked to Cas9 targeting efficiency in addition to being computationally significant (as measured by SHAP scores). This convergence improves the ensemble prediction framework's interpretability and reliability.

3.9 Discussion of the Results

We evaluated each model's predicted accuracy, interpretability, and feature contributions before combining their predictions in a meta-learner that was trained with XGBoost. Both the Hanna and the Knockout datasets were analyzed, with an emphasis on statistical validation, feature convergence, and interpretability based on SHAP. Different charts provide a visual representation ofperfomance metrics of the models average performance while the Pearson correlation index for regression tasks were obtained , allowing to observe variations in performance across different configurations. We focused on pearson corelation coefficient matrices because we were intrested to sort gRNAs to its specific efficiencies not to approximate thes efficiencies scores.

Evaluation of the Baseline Machine Learning Model (SpCas9-Endo Dataset): We started by analyzing a number of traditional machine learning regressors on the Knockout dataset, Linear Regression, Multi-Layer Perceptron (MLP), Support Vector Regression (SVR), Decision Tree, Gaussian Process and XGBoost as shown in figures 3.6 to 3.11 and in table 3.1 which clearly shows that XGBoost performed better than all other models, with the lowest MSE (0.0391) and greatest test Pearson (0.6111).

Optimization of XGBoost Hyperparameter: The XGBoost model was trained with 10-fold cross-validation, and hyperparameter were adjusted for each fold to ensure a fair evaluation (table 3.2). Although the majority of folds favored a straightforward setup (e.g., 10 estimators, depth of 3), Fold 3 (50 estimators) and Fold 8 (100 estimators, lower learning rate) showed variances, indicating that dataset-specific tuning enhances generalization.

Transfer Learning Assessment: SpCas9-Endo-Trained Model on Hanna Data set:

We used the XGBoost model that was trained on the SpCas9-Endo dataset to demonstrate a workflow by applying it directly to the Hanna dataset without retraining (table 3.3). The results were consistently unsatisfactory across all folds, with high RMSE (1.28) and negative R2 values (average R2 = -0.4083), suggesting that domain shift in sequence distributions or labeling schemes causes direct transfer learning without fine-tuning to fail.

CRISPR Tool and XGBoost Retraining on the Hanna Dataset:

To address the domain shift approach we used 10-fold cross-validation to retrain four distinct tools—CRISPRedict, CRISPRAidit, DeepHF, and CRISPRon—on the Hanna dataset (table 3.4). CRISPRedict produced acceptable outcomes (R2 = 0.2391; Pearson = 0.4891). Despite its feature richness, CRISPRAidit performed poorly (R2 = 0.1050; Pearson = 0.4152), DeepHF fared better than the other two, with R2 = 0.4762 and Pearson = 0.6275, indicating improved generalization using biologically informed features with the lowest RMSE (0.9234) among the individual tools, CRISPRon is the most successful independently model with Pearson = 0.6815 and R2 = 0.7424. According to these findings, retraining on the target dataset is essential, and methods that incorporate biological knowledge—like DeepHF and CRISPRon—generalize more effectively. We trained an XGBoost meta-model employing predictions from CRISPRAidit, CRISPRedict, DeepHF, and CRISPRon as input features, assessed using 10-fold cross validation, building on the outputs of the retrained tools (table3.5). The ensemble regularly outperformed individual tools, with an average R2 of 0.4562, RMSE of 1.0628, and Pearson correlation of 0.6876. Most folds achieved R2 $>$0.41 and Pearson $>$0.7686, indicating the meta-learner's adaptability, while fold-wise metrics stayed consistent. Moreover, the corelation analysis is shown in (fig3.13), indicating that XGBoost is performing well on predictive models as shown by the correlation of 0.704 with CRISPRedict and a high strongest correlation with crisprOn (0.988), which indicates the strongest positive relation.

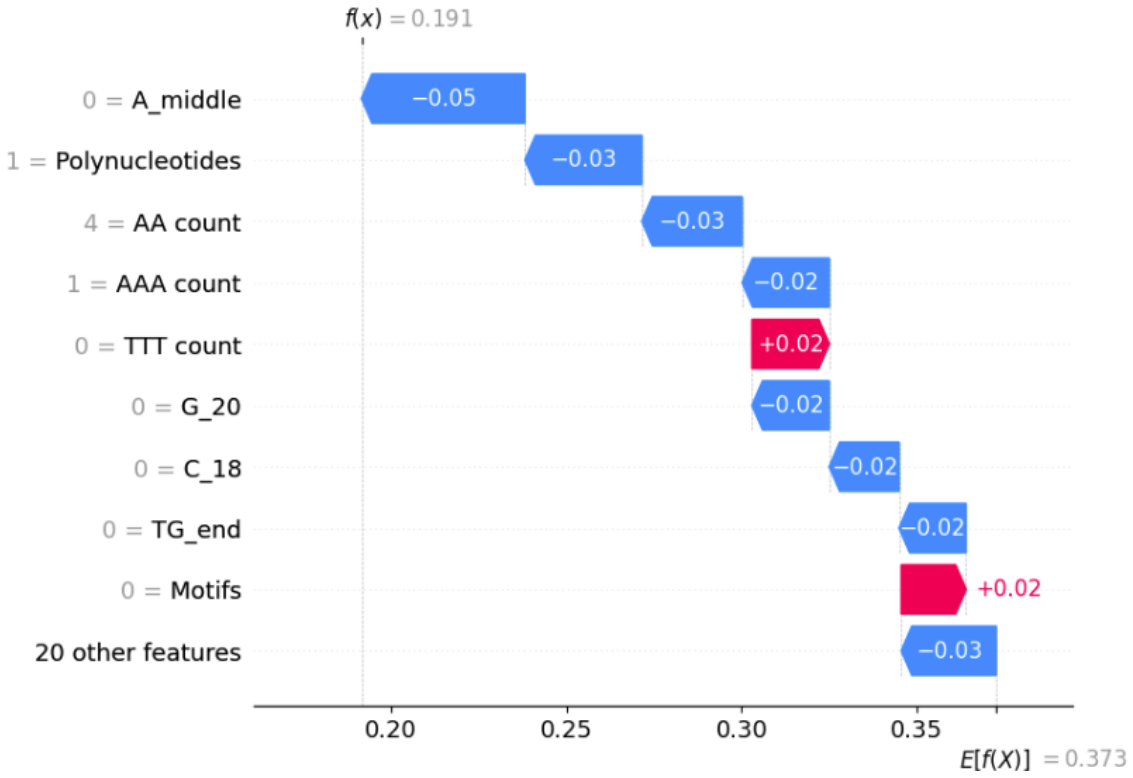


Figure 3.14: SHAP waterfall plot for the CRISPRedict SpCas9-Endo dataset showing the performance of each 29 features towards the model prediction.

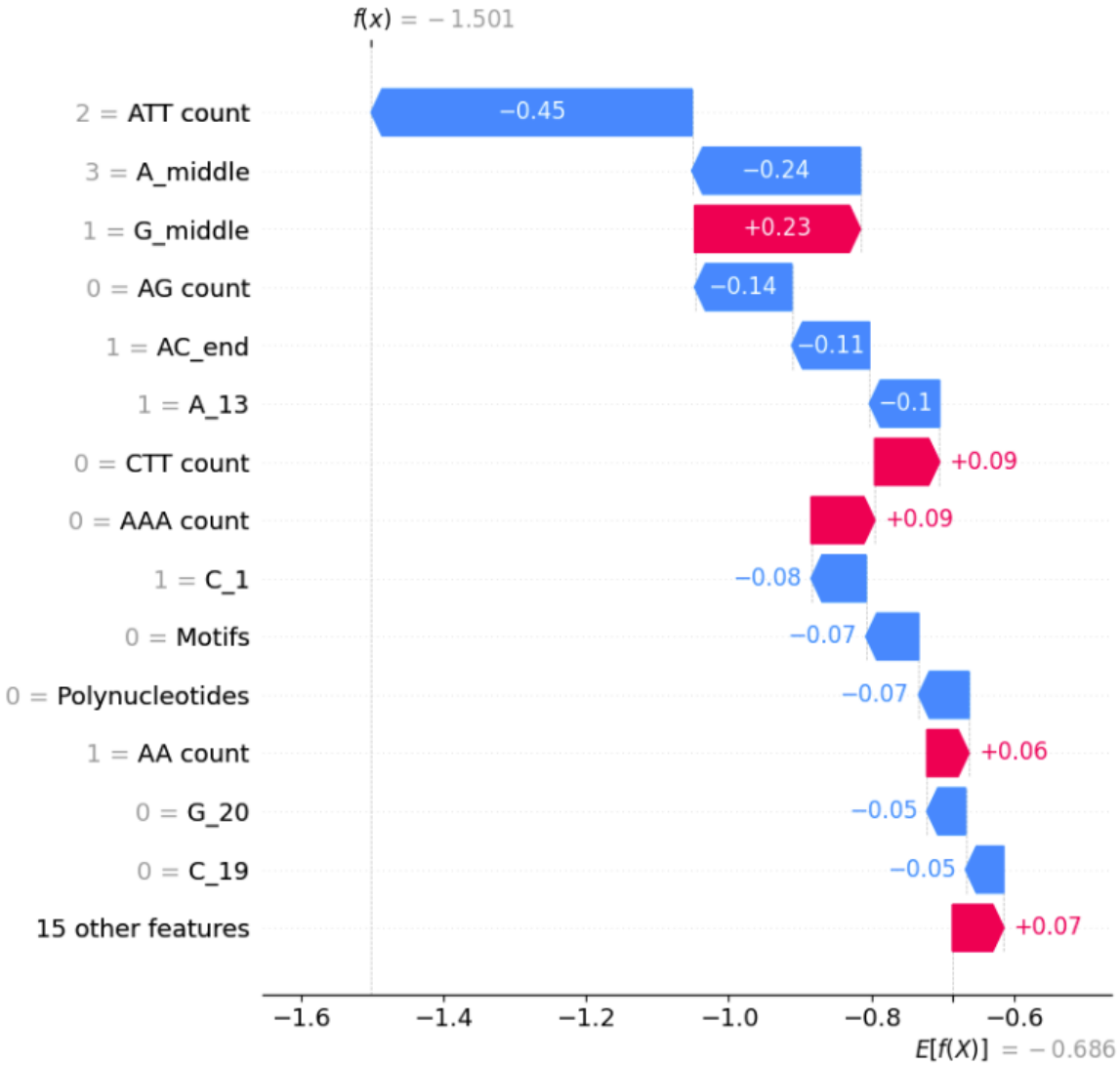


Figure 3.15: Shap waterfall plot for crisprEdict Hanna dataset showing the performance of each 29 features towards the model prediction.

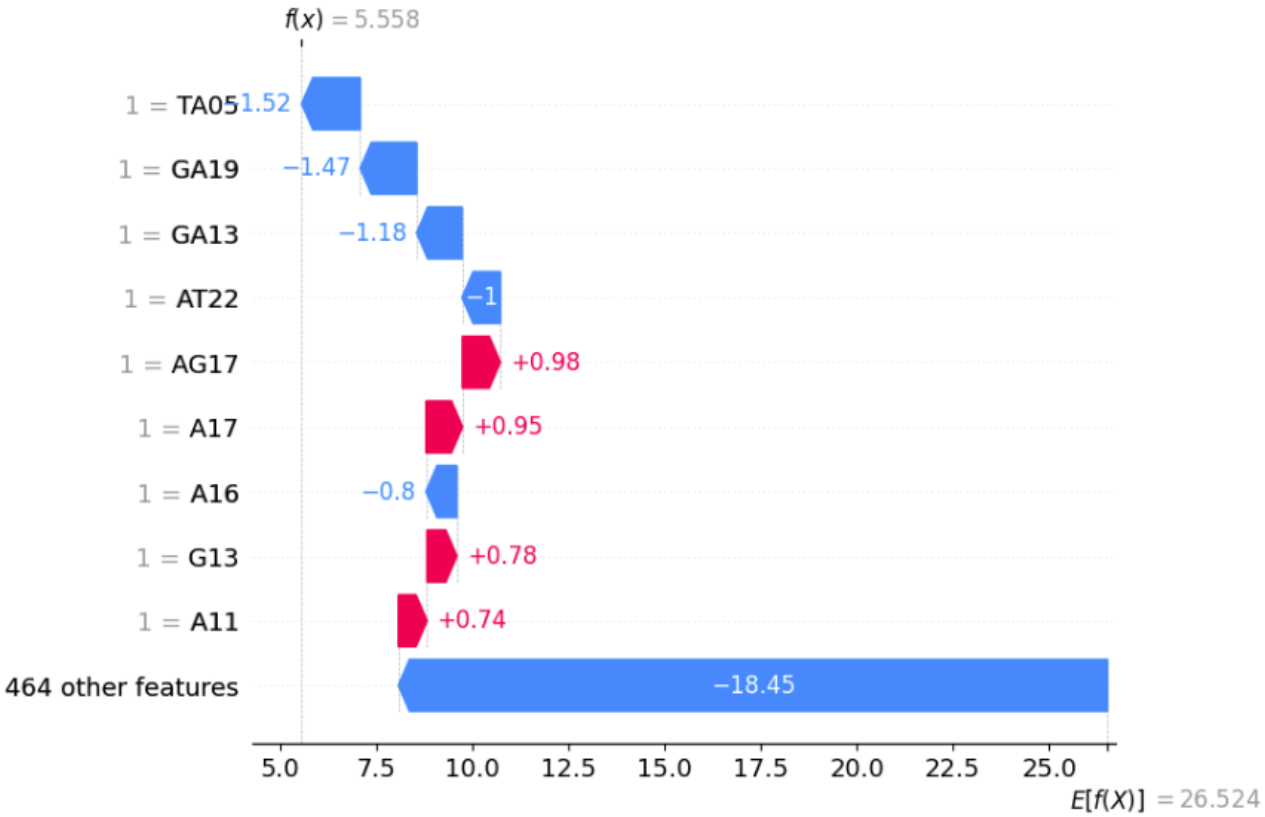


Figure 3.16: SHAP waterfall plot for CRISPRAidit on SpCas9-Endo dataset showing the performance of each 464 features towards the model prediction.

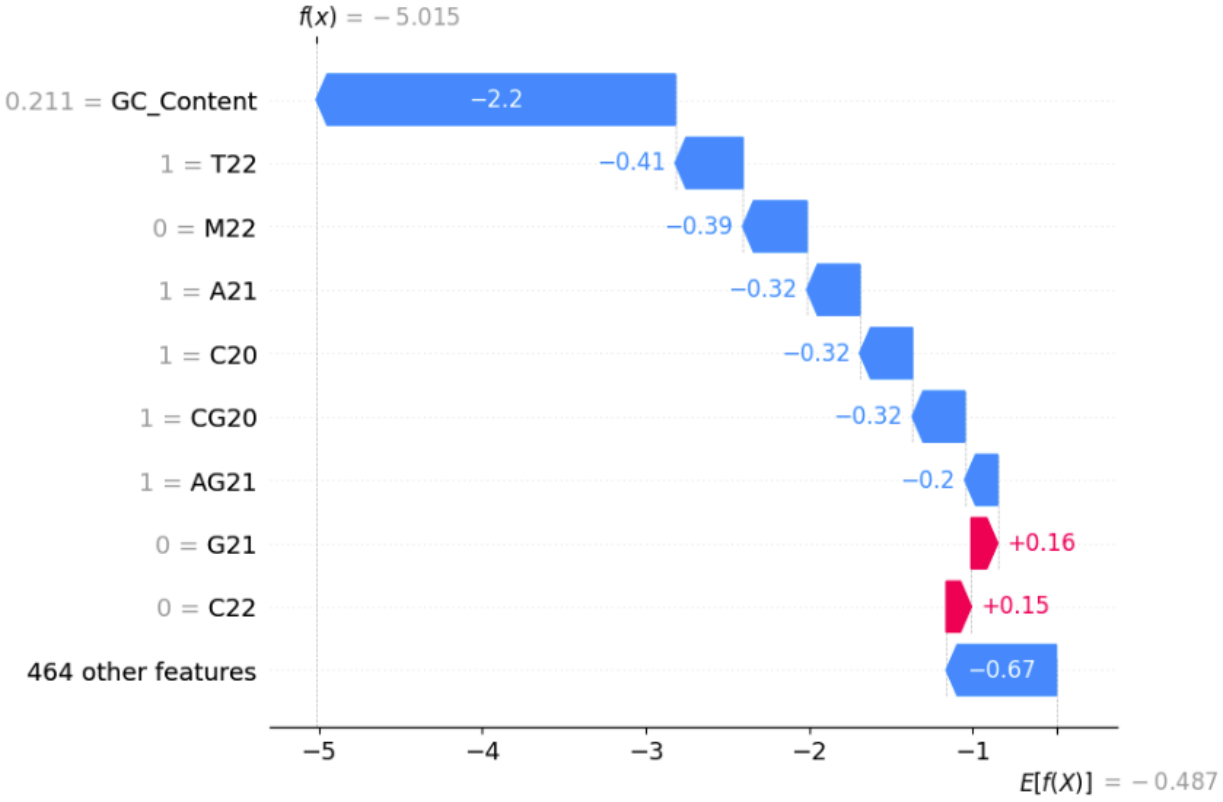


Figure 3.17: SHAP waterfall plot for CRISPRAidit on Hanna dataset showing the performance of each 464 features towards the model prediction.

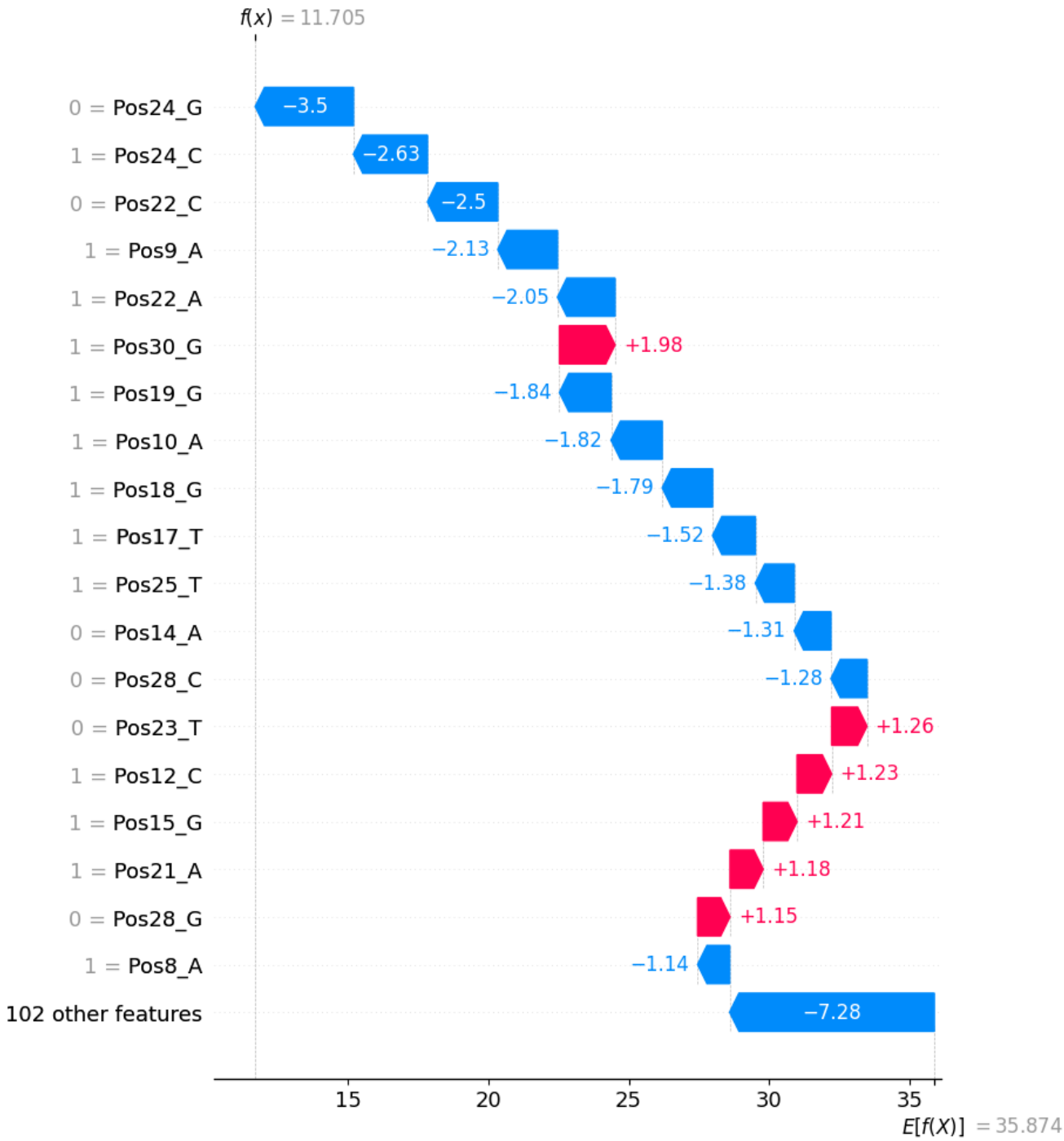


Figure 3.18: Shap waterfall plot for CrisprOn SpCas9-Endo dataset showing the performance of each 121 features towards the model prediction.

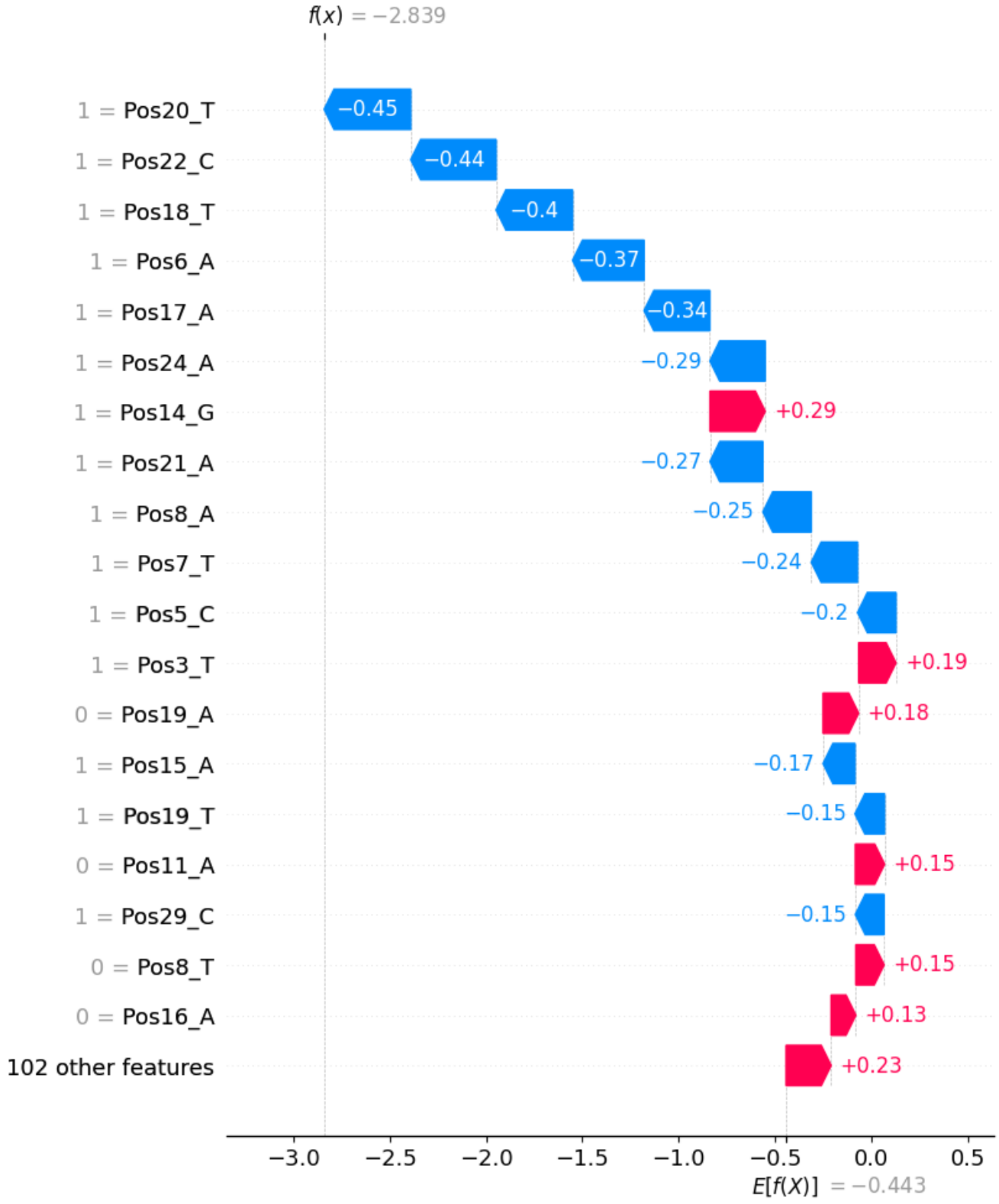


Figure 3.19: Shap waterfall plot for CrisprOn Hanna dataset showing the performance of each 121 features towards the model prediction.

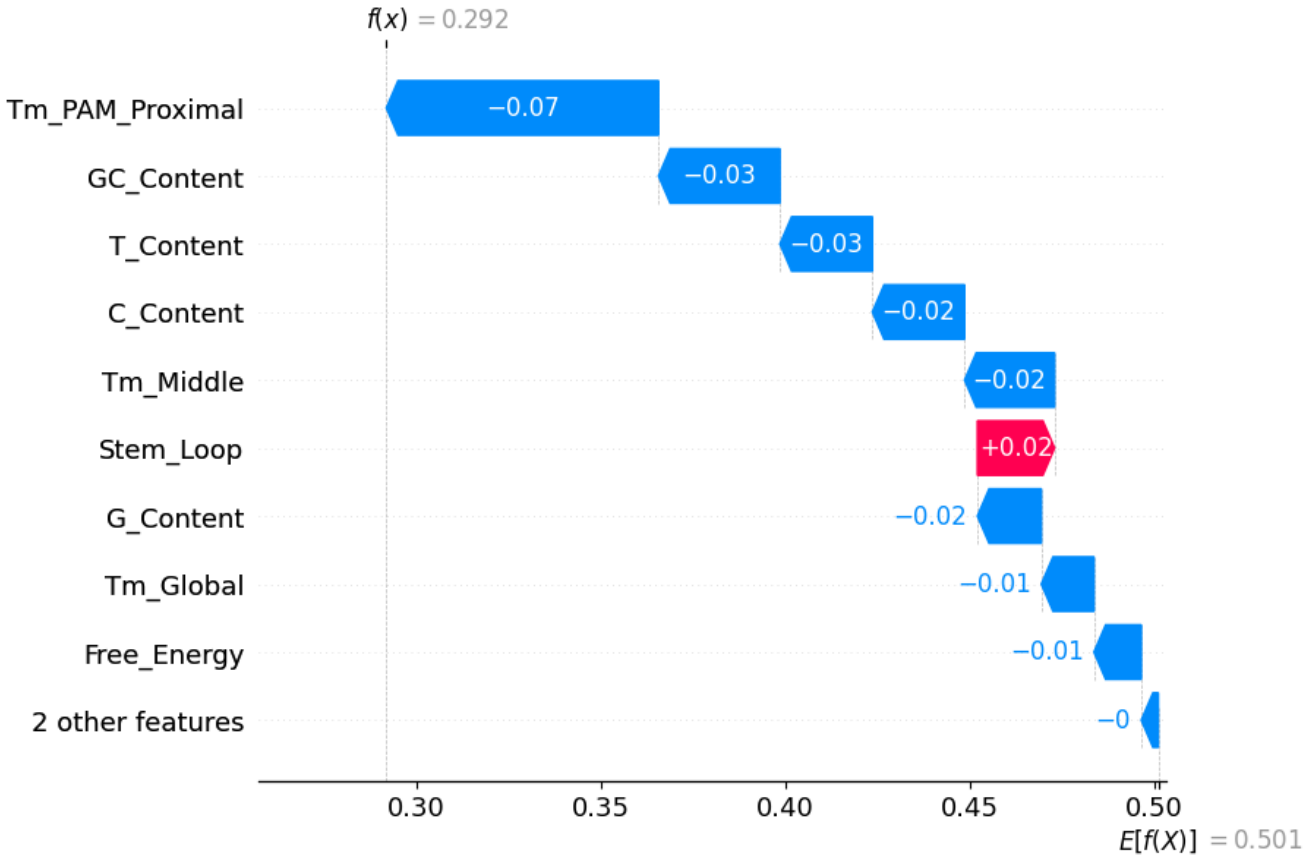


Figure 3.20: Shap waterfall plot for DeepHF SpCas9-Endo dataset showing the performance of each 11 biophysical features towards the model prediction.

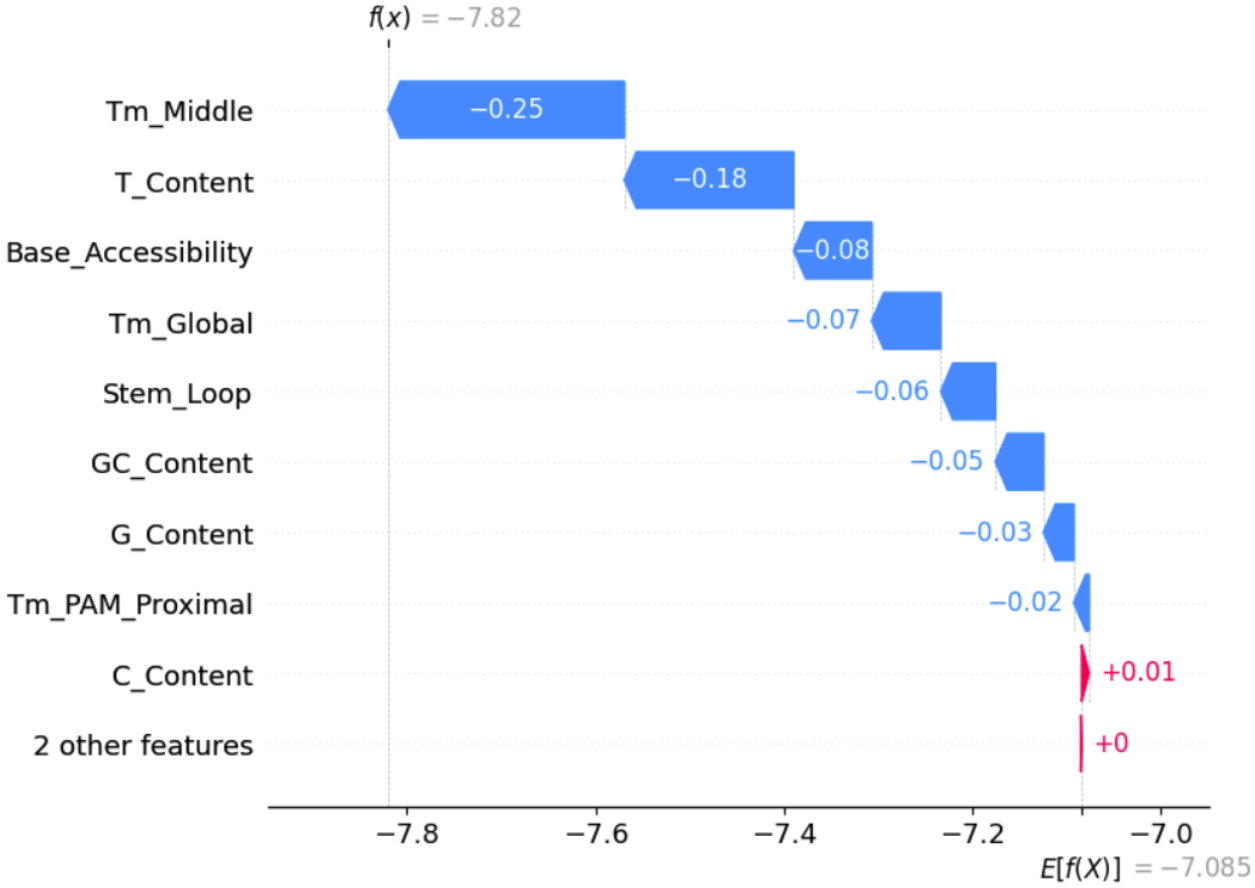


Figure 3.21: Shap waterfall plot for DeepHF Hanna dataset showing the performance of each 11 biophysical features towards the model prediction.

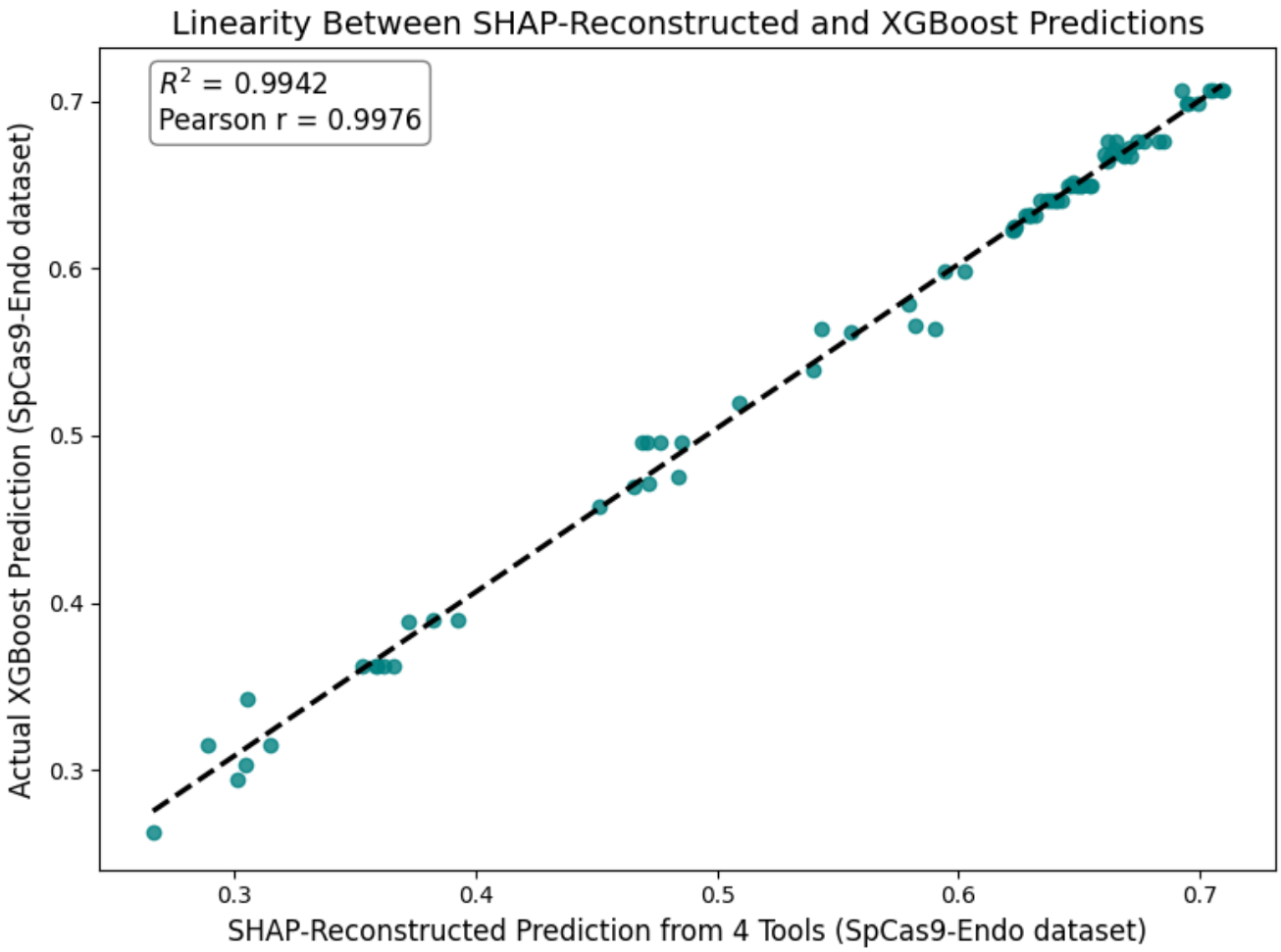


Figure 3.22: A linearity plot comparing the XGBoost predictions on theSpCas9-Endo dataset with the SHAP-reconstructed predictions from the four base tools through shap aggregation formula.

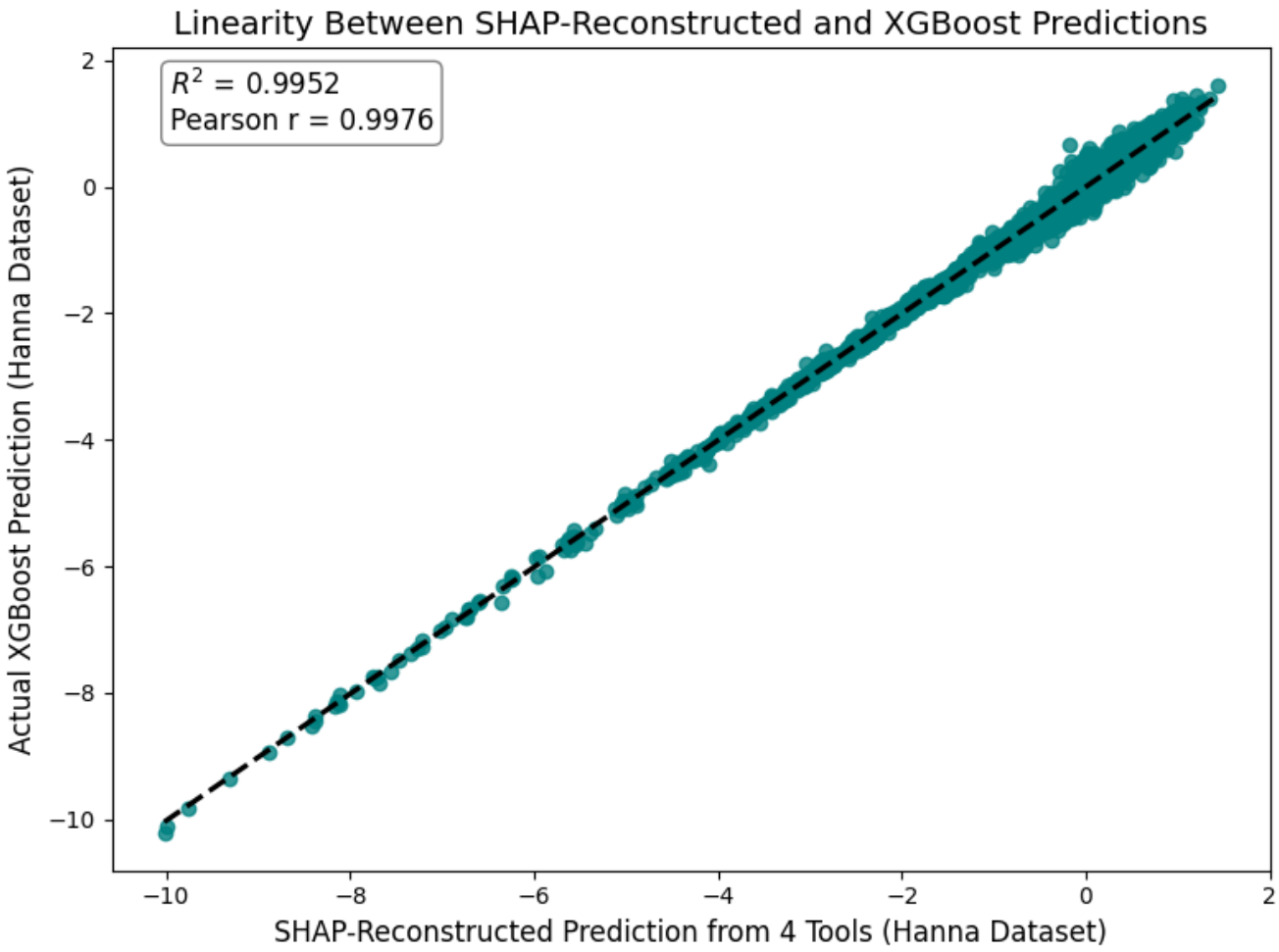
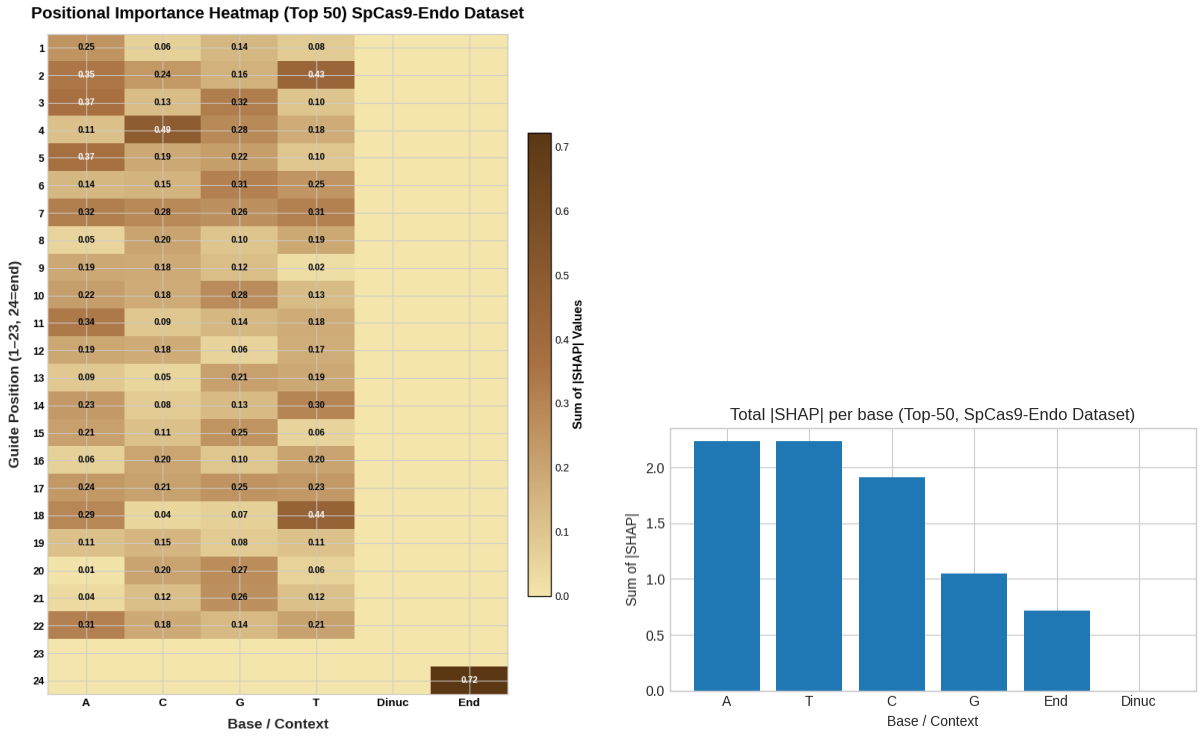
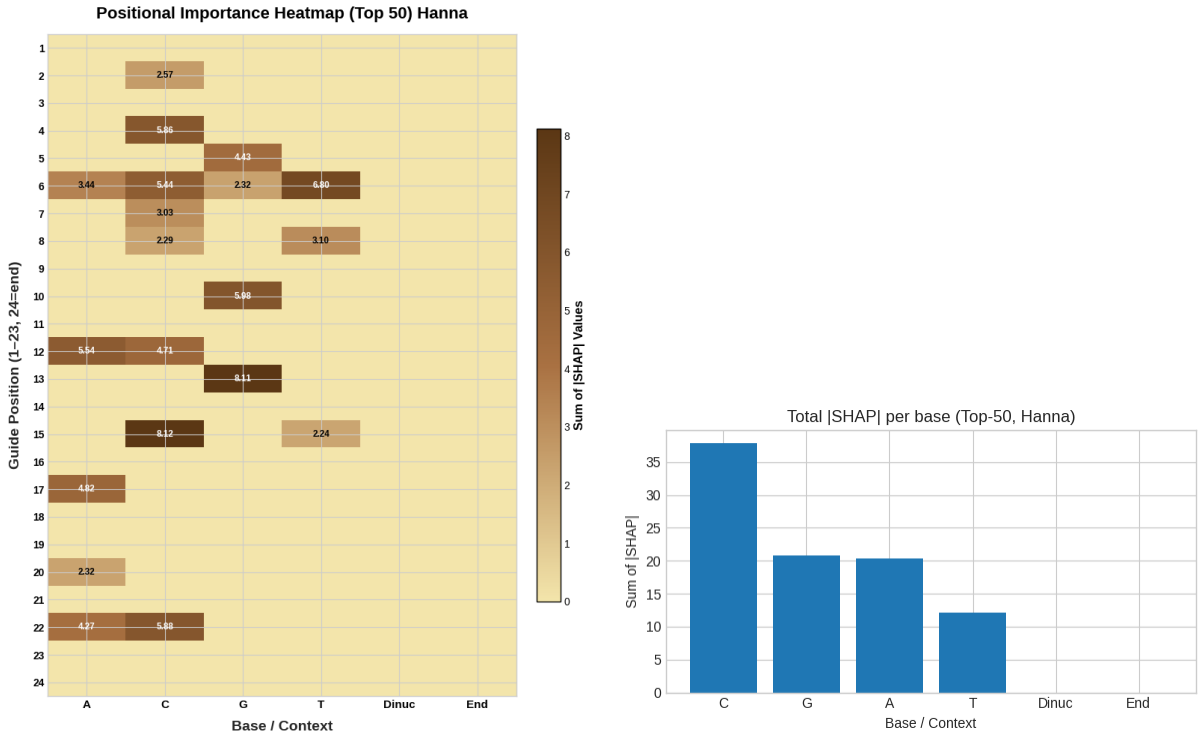


Figure 3.23: A linearity plot comparing the XGBoost predictions on the Hanna dataset with the SHAP-reconstructed predictions from the four base tools through shap aggregation formula.



(a) (b)

Figure 3.24: Summed absolute SHAP values per guide position (a) and nucleotide-specific SHAP values (b) for the SpCas9-Endo dataset.



(a) (b)

Figure 3.25: Summed absolute SHAP values per guide position (c) and nucleotide-specific SHAP values (d) for the Hanna dataset.

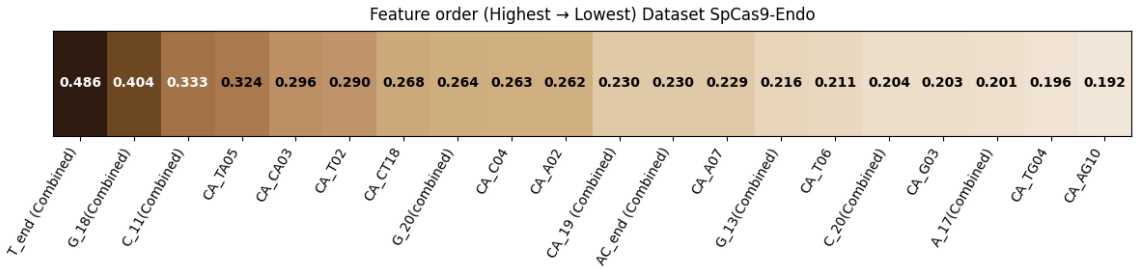


Figure 3.26: SHAP Summary plot (top-20 features) for SpCas9-Endo data

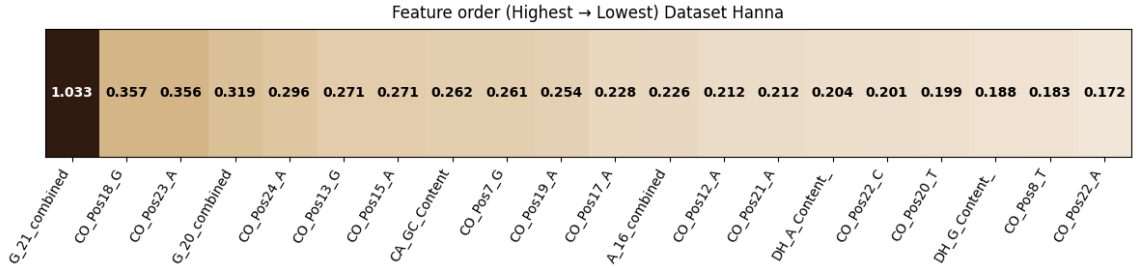


Figure 3.27: SHAP Summary plot (top-20 features) for Hanna data

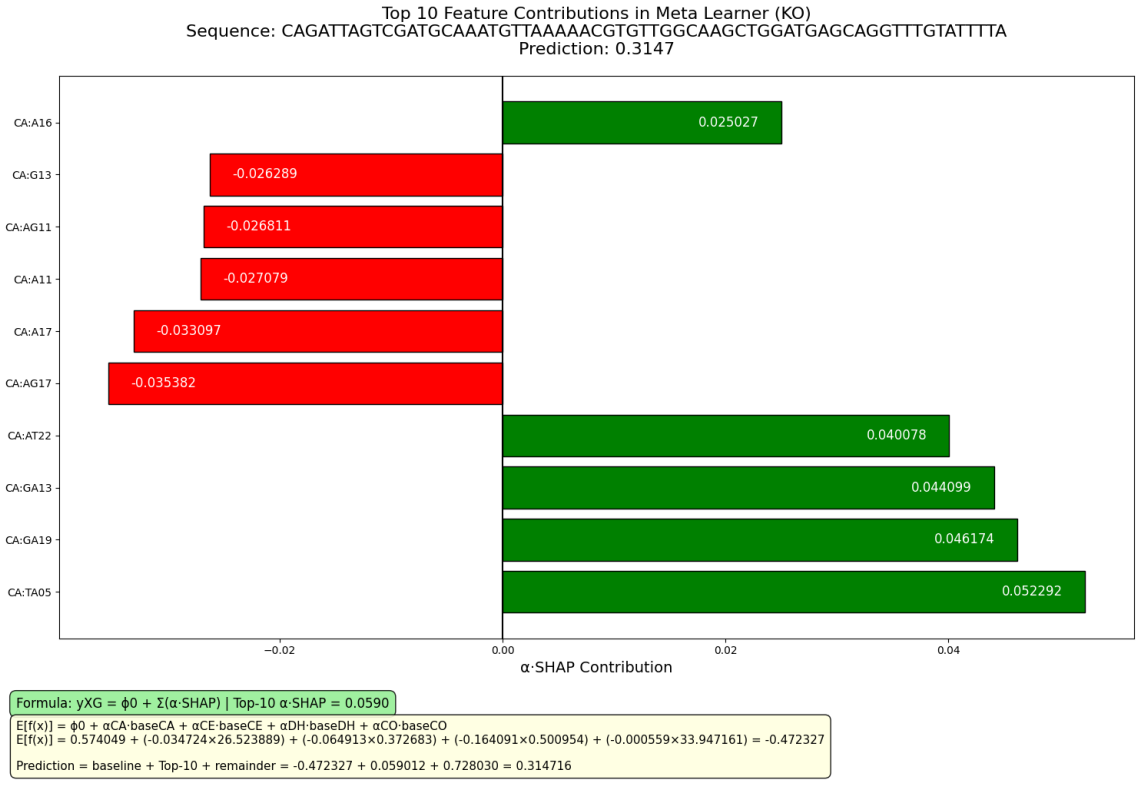


Figure 3.28: KO features contribution in meta learner

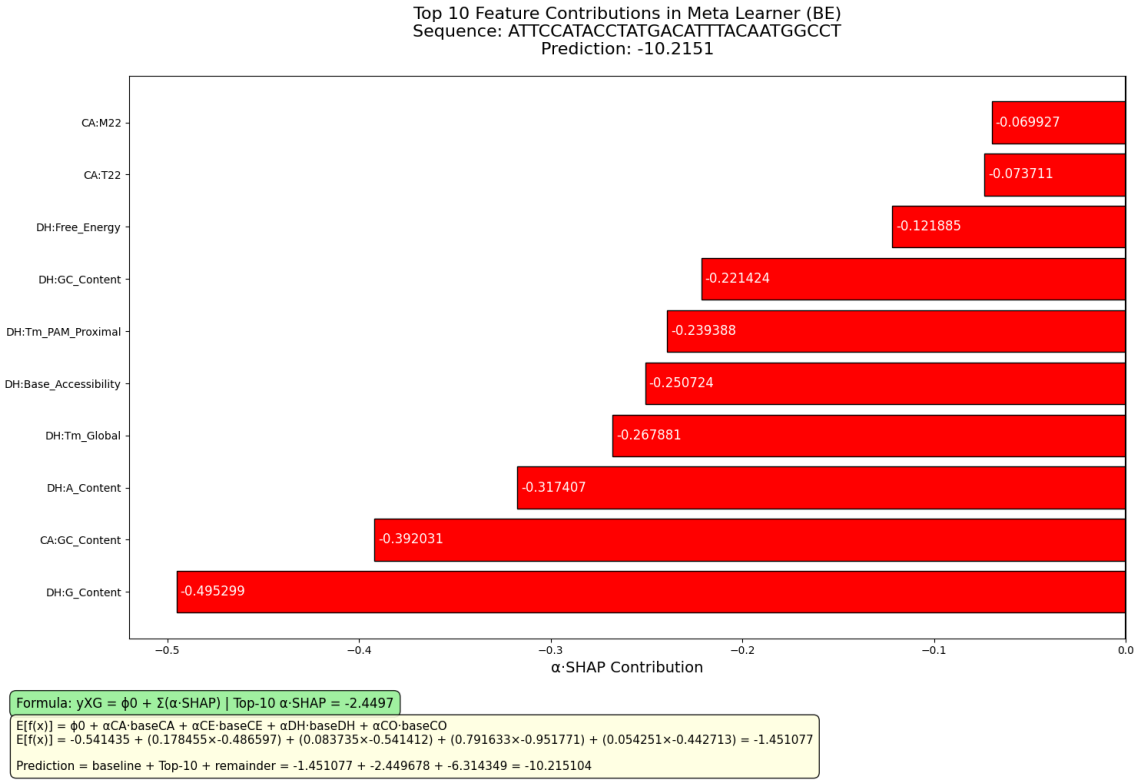


Figure 3.29: BE features contribution in meta learner

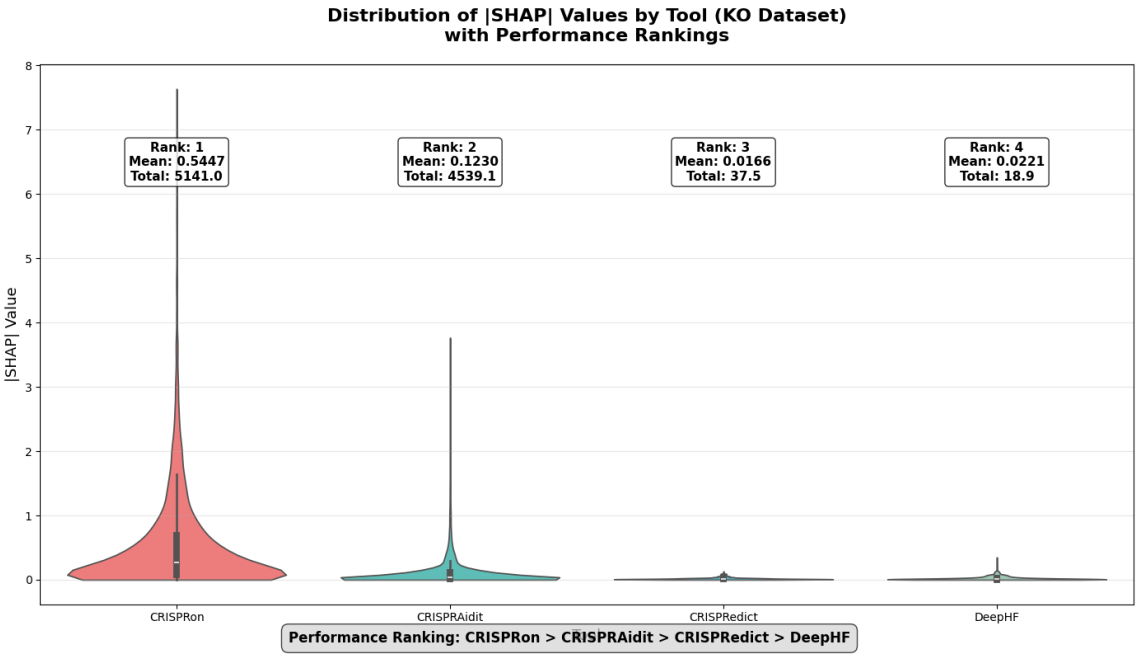


Figure 3.30: KO tools contribution in meta learner

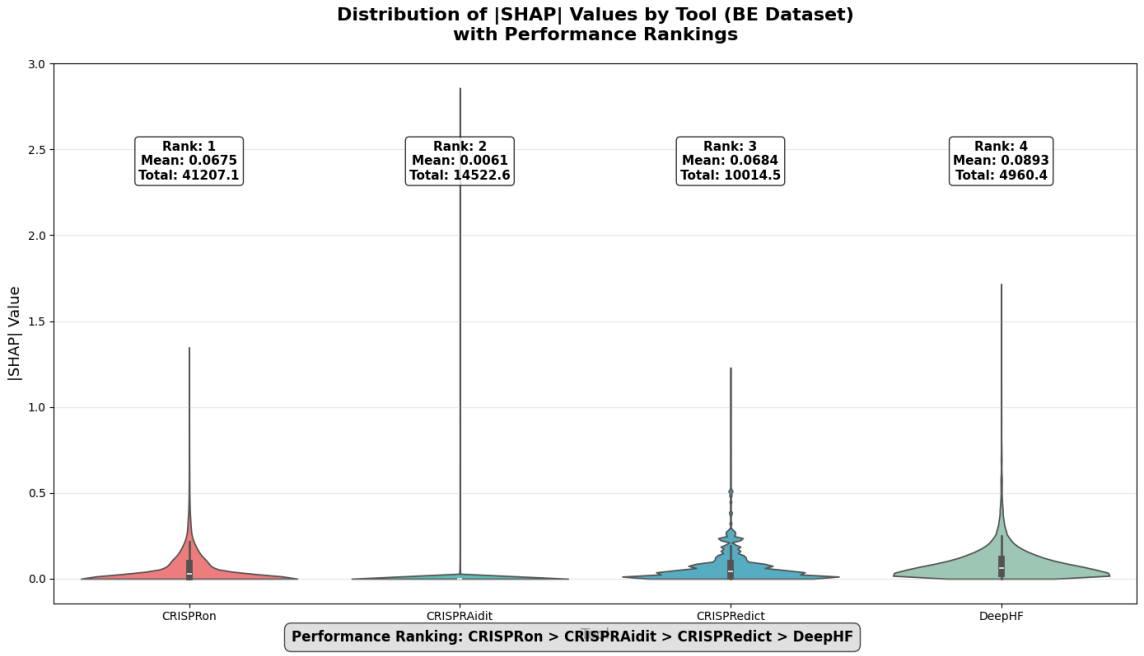


Figure 3.31: BE tools contribution in meta learner

Table 3.6: Biological interpretation of combined top SHAP features from four models

Feature Group	Combined From four Tools	Biological Interpretation
G_24_combined	CE_G-24, CO_pos-24-G	Guanine at position -24, possibly linked to PAM-distal structure or transcriptional noise buffering
G_21_combined	CE_G_21, CA_G21, CO_Pos21_G	Conserved G near 5' end may affect Cas9 binding affinity and sgRNA folding
G_20_combined	CE_G_20, CA_G20, CO_Pos20_G	A well-documented hotspot near the seed region that influences cleavage rate
C_19_combined	CE_C_19, CA_C19, CO_Pos19_C	Cytosine presence near the PAM-proximal site can affect Cas9 specificity
C_18_combined	CE_C_18, CA_C18, CO_Pos18_C	May impact local DNA secondary structure and melting temperature
C_13_combined	CA_C13, CO_Pos13_C	Possibly contributes to gRNA stability in mid-sequence regions
C_1_combined	CE_C_1, CA_C01, CO_Pos1_C	One of the most PAM-proximal positions; crucial for target recognition
A_16_combined	CE_A_16, CA_A16, CO_Pos16_A	Adenine effects in mid-sequence could affect local DNA-RNA hybridization
A_13_combined	CE_A_13, CA_A13, CO_Pos13_A	Repetitive adenines might impact secondary structure or Cas9 tolerance
T_end_combined	CE_T_end, CA_T00	Terminal thymine motifs may destabilize 3' end or affect transcription termination
AC_end_combined	CE_AC_end, CA_AC00	Di-nucleotide motifs (e.g., AC) at terminal ends could modulate stability
CA_19_combined	CE_CA_19, CA_CA19	Specific CA motifs near cleavage site are known to affect editing efficiency
CT_13_combined	CE_CT_13, CA_CT_13	Cytosine at position -13 may affect Cas9 binding efficiency and gRNA stability

Feature Group	Combined From four Tools	Biological Interpretation
A_10_combined	CA_A10, CO_Pos10_A	May contribute to local sequence context
C_10_combined	CA_C10, CO_Pos10_C	May support local structural stability
G_10_combined	CA_G10, CO_Pos10_G	May affect guide composition and binding
T_10_combined	CA_T10, CO_Pos10_T	May have a minor positional effect
A_11_combined	CA_A11, CO_Pos11_A	May influence local guide conformation
C_11_combined	CA_C11, CO_Pos11_C	May strengthen sequence-specific stability
G_11_combined	CA_G11, CO_Pos11_G	May affect sgRNA folding and binding
T_11_combined	CA_T11, CO_Pos11_T	May contribute to local sequence context
A_12_combined	CA_A12, CO_Pos12_A	May influence guide flexibility
C_12_combined	CA_C12, CO_Pos12_C	May support local DNA-RNA stability
G_12_combined	CA_G12, CO_Pos12_G	May affect sequence-dependent binding
T_12_combined	CA_T12, CO_Pos12_T	May influence local structural context
G_13_combined	CA_G13, CO_Pos13_G	May affect guide stability and folding
T_13_combined	CA_T13, CO_Pos13_T	May influence sequence flexibility
A_14_combined	CA_A14, CO_Pos14_A	May contribute to local sequence context
C_14_combined	CA_C14, CO_Pos14_C	May support DNA structural stability
G_14_combined	CA_G14, CO_Pos14_G	May affect guide conformation
T_14_combined	CA_T14, CO_Pos14_T	May influence cleavage-associated context
A_15_combined	CA_A15, CO_Pos15_A	May affect local hybridization dynamics
C_15_combined	CA_C15, CO_Pos15_C	May support local sequence stability
G_15_combined	CA_G15, CO_Pos15_G	May influence guide folding
T_15_combined	CA_T15, CO_Pos15_T	May affect local sequence context
C_16_combined	CA_C16, CO_Pos16_C	May support local DNA-RNA stability

Feature Group	Combined From four Tools	Biological Interpretation
G_16_combined	CA_G16, CO_Pos16_G	May influence guide structure
T_16_combined	CA_T16, CO_Pos16_T	May affect local conformational flexibility
A_17_combined	CA_A17, CO_Pos17_A	May influence guide conformation
C_17_combined	CA_C17, CO_Pos17_C	May influence local DNA structure and guide RNA stability
G_17_combined	CA_G17, CO_Pos17_G	May affect guide stability near the seed-adjacent region
T_17_combined	CA_T17, CO_Pos17_T	May contribute to local sequence flexibility
A_18_combined	CA_A18, CO_Pos18_A	May influence local hybridization context
G_18_combined	CA_G18, CO_Pos18_G	May contribute to guide stability and efficiency
T_18_combined	CA_T18, CO_Pos18_T	May affect local structural balance
A_19_combined	CA_A19, CO_Pos19_A	May influence cleavage-site context
G_19_combined	CA_G19, CO_Pos19_G	May affect guide sequence preference
T_19_combined	CA_T19, CO_Pos19_T	May influence cleavage-region flexibility
A_20_combined	CA_A20, CO_Pos20_A	May affect local sequence composition near the seed region
C_20_combined	CA_C20, CO_Pos20_C	May influence cleavage-region precision and specificity
T_20_combined	CA_T20, CO_Pos20_T	May affect cleavage efficiency
A_21_combined	CA_A21, CO_Pos21_A	May affect distal sequence flexibility
C_21_combined	CA_C21, CO_Pos21_C	May influence distal specificity
T_21_combined	CA_T21, CO_Pos21_T	Influences distal guide stability
A_22_combined	CA_A22, CO_Pos22_A	May influence terminal binding efficiency
C_22_combined	CA_C22, CO_Pos22_C	May impact distal structural stability
G_22_combined	CA_G22, CO_Pos22_G	May influence distal guide structure
T_22_combined	CA_T22, CO_Pos22_T	Influences terminal-region stability
CA_13_combined	CE_CA_13, CA_CA13	May influence local sequence flexibility

4 Feature Representation for Explainable CRISPR Off-Target Prediction and Base Editing Efficiency

This chapter reports a study on the impact of feature representation on accuracy and interpretability of a Machine Learning (ML) model trained to predict off-target events in CRISPR/Cas9 applications. The study expands upon our earlier research regarding interpretability and feature representation in CRISPR off-target prediction. In our original study, we trained an XGBoost model on the CHANGE-seq and GUIDE-seq datasets to evaluate how different sequence encodings affect model performance. In this expanded version, we address two CRISPR applications: gene knockout (KO) and base editing (BE) using distinct benchmark datasets. For the KO problem, we utilized CHANGE-seq and GUIDE-seq to evaluate paired sequence representations. To extend the analysis to BE, we used the Hanna dataset, focusing on One-Hot and K-mer encodings. Results indicate that One-Hot encoding consistently outperforms K-mer representations. Furthermore, our analysis demonstrates comparable predictive performance across both gene knockout and base editing tasks, confirming the robustness of the framework in distinct editing domains. Interpretability analysis using SHAP reveals that despite different mechanisms, the PAM-proximal region remains a critical feature for prediction for both editing mechanism.

4.1 Materials and Methods

4.1.1 Datasets

Our experimental framework utilizes weighted XGBoost models for both off-target event prediction and efficiency estimation. We employed three high-throughput datasets

covering two distinct biological domains.

CHANGE-seq and GUIDE-seq (Knockout Domain)

Two datasets were used to examine off-target effects in the CRISPR/Cas9 system:

- **CHANGE-seq:** Includes 202,044 off-target sites across 110 gRNAs cameron2017mapping.
- **GUIDE-seq:** Includes 1,072 off-target sites across 58 gRNAs tsai2015guide.

To train and test the models, we defined Active and Inactive Off-Target Sites (OTS) as follows:

- **CHANGE-seq:** Active OTS were defined as those with cleavage counts over 100, while sites with 100 or fewer counts were considered inactive. This resulted in **67,476 active** and **2,806,152 inactive** off-targets.
- **GUIDE-seq:** All experimentally identified off-targets were marked as active. Inactive OTS were identified using Cas-OFFinder as potential Cas9 mismatches (up to six mismatches), excluding the known active sites. This dataset comprises **1,702 active** and **1,476,301 inactive** off-targets.

Hanna Dataset (Base Editing Domain)

To extend the analysis to Base Editing (BE), we used the dataset from [83]. This dataset contains 11,088 sgRNAs targeting human single-nucleotide variations (SNVs) using BE4max and ABE8e editors. Unlike the KO datasets, Hanna provides a single sgRNA sequence associated with a z-score representing editing efficiency.

4.1.2 Data Pre-processing

Given the high variance in sequencing data and the severe class imbalance in the off-target datasets, specific pre-processing steps were applied to stabilize the model training.

Logarithmic Transformation: Specifically for the Knockout (KO) datasets (CHANGE-seq and GUIDE-seq), where the target variable represents raw cleavage counts, we

applied a logarithmic transformation to reduce the impact of outliers and normalize the distribution. The transformation is defined as:

$$y' = \log(y + 1) \tag{4.1}$$

where y is the raw read count. Note that this transformation was not applied to the Hanna dataset, as its target variable (z-score) is already normalized.

Class Weighting: Both CHANGE-seq and GUIDE-seq datasets are highly imbalanced (approx. 97% negative). To prevent the model from biasing towards the majority class, we applied sample weighting during training. The weight for the active class was calculated as:

$$W_{active} = \frac{|N_{inactive}|}{|N_{inactive}| + |N_{active}|} \tag{4.2}$$

This ensures that both active and inactive samples contribute proportionally to the learning process.

4.1.3 Feature Encoding Strategies

Feature representation is central to our analysis. We categorize the encodings into two groups based on whether they explicitly model the alignment relationship between paired sequences or represent the nucleotide content directly.

Alignment-Based Encodings (CHANGE-seq & GUIDE-seq)

These encodings are exclusively applicable to the paired datasets (KO task), as they rely on the explicit alignment between the gRNA and the Off-Target sequence to extract mismatch patterns. We evaluated:

- **Nucleotides to Position Mapping (NPM):** A $23 \times 4 \times 4$ tensor representing the specific pairing between the gRNA nucleotide and the DNA off-target nucleotide at each position. This results in **368 features** that capture exactly which nucleotide substitution occurred (e.g., A-to-G vs. A-to-C).
- **Bulges Encoding:** An extension of NPM that accounts for DNA/RNA bulges (insertions/deletions) by adding a specific channel to the matrix, allowing the model to handle gapped alignments.
- **Label Encoding Pairwise (LEP):** Assigns a unique integer $(0-15)$ to each of the

16 possible nucleotide pairs (e.g., A-A, A-T, G-C) at each position. This results in a compact vector of **23 features**.

- **Match-Mismatch (MM):** A binary vector of length 23 indicating simply the presence (1) or absence (0) of a mismatch at each position, ignoring nucleotide identity.
- **8XL Encoding:** A flattened concatenation of the One-Hot encoded gRNA and the One-Hot encoded Off-Target sequence, resulting in a vector of $23 \times 8 = 184$ features charlier2021accurate.

Sequence-Content Encodings

Unlike the alignment-based methods, these strategies focus on representing the nucleotide identity or compositional motifs directly. These encodings are versatile, as they were applied to the single-sequence Hanna dataset (BE) and adapted for the paired sequences in CHANGE-seq and GUIDE-seq (KO) through aggregation operations.

- **One-Hot Encoding (OH):** This strategy captures precise positional information. Each nucleotide is represented by a binary vector of length 4.
  - *For Base Editing (Hanna):* The single sgRNA is directly encoded, resulting in a flattened vector of $23 \times 4 =$ **92 features**.
  - *For Knockout (KO):* Both the target gRNA and the off-target sequence are encoded separately. The two resulting vectors are then combined via a **bitwise logical OR** operation. This allows the model to capture the presence of different nucleotides at the same position (effectively representing mismatches implicitly) while maintaining the same 92-feature dimensionality.
- **K-mer Encoding (KMER):** This strategy captures local sequence composition by counting the frequency of overlapping subsequences of length $k = 3$. This yields a vector of **63 features**, losing exact positional context but retaining motif information.
  - *For Base Editing (Hanna):* K-mers are counted on the sgRNA sequence to capture motifs favored by the deaminase.
  - *For Knockout (KO):* K-mers are extracted from the off-target sequence to represent the compositional context of the cleavage site.

4.1.4 Gradient Boosting Implementation

We implemented the models using XGBoost (eXtreme Gradient Boosting). This algorithm optimizes a loss function $L^{(t)}$ at iteration t using the second-order Taylor expansion:

$$L^{(t)} \approx \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \Omega(f_t) \tag{4.3}$$

where g_i and h_i are the gradient and hessian of the loss function, and Ω represents the regularization term (L1/L2) to control tree complexity and prevent overfitting. The model is updated iteratively:

$$F_m(x) = F_{m-1}(x) + \eta \cdot h_m(x) \tag{4.4}$$

where η is the learning rate.

4.1.5 Experimental Setup and Evaluation

Knockout (KO) Framework (CHANGE/GUIDE-seq)

We established two categories of predictive models for the KO datasets:

1. **Classification:** Aimed at determining whether an OTS is active or inactive.
2. **Regression:** Designed to estimate the exact cleavage count for an OTS.

For every encoding strategy, we trained two model variants to isolate the contribution of sequence information versus mismatch count:

- **Sequence-Only (-seq):** Uses only the encoded sequence features.
- **Sequence + Distance (-seq-dist):** Uses the encoded sequence features plus an explicit integer feature representing the Hamming distance (number of mismatches).

Cross-Validation Strategy:

- **CHANGE-seq:** We employed a “leave-11-sgRNAs-out” approach. The 110 gRNAs were divided into 10 mutually exclusive sets. In each iteration, one set served as the test set.

- **GUIDE-seq:** We utilized a Leave-One-Group-Out cross-validation.

Metrics: For classification, we calculated the **Area Under the Precision-Recall curve (AUPR)**. For regression, we evaluated **Pearson's** and **Spearman's** correlation coefficients on the read counts pertaining exclusively to active off-target sites.

Base Editing (BE) Framework (Hanna)

For the Hanna dataset, we maintained a similar dual-task approach but adapted for single sequences:

1. **Regression:** Predicting the continuous z-score directly (Metric: Pearson Correlation).
2. **Classification:** Discretizing the z-score to distinguish highly effective sgRNAs (Active) from the rest (Inactive). We adopted the specific threshold of $z < -2.0$ to maintain methodological consistency with the original study hanna2021massively, where this cutoff was established.

4.1.6 Hyperparameter Optimization

To ensure the models achieved optimal performance, we employed a **Randomized Search** strategy combined with cross-validation. This approach allowed us to explore a wide range of model configurations—adjusting parameters such as learning rate (η), maximum tree depth, and regularization terms (λ, α)—without the computationally prohibitive cost of a full grid search.

4.2 Results and Discussion

4.2.1 XGBoost Robustness on Off-Target Prediction (KO)

We first evaluated XGBoost on the classical off-target prediction task. Table 4.2 (Columns 1-2) summarizes the performance on GUIDE-seq and CHANGE-seq. Consistent with previous literature, encoding strategies that explicitly model mismatches (Bulges, NPM) perform strongly, achieving AUPRs of 0.673 and 0.661 respectively on GUIDE-seq.

However, a key observation is the performance of **One-Hot (OH)** encoding. On GUIDE-seq, OH achieves an AUPR of 0.611, significantly outperforming K-mer encoding

(0.577). This suggests that even without explicit mismatch features, XGBoost can learn the importance of specific nucleotide positions (e.g., the PAM-proximal region) directly from the raw sequence representation.

To further investigate the model's stability and the specific contribution of mismatch information, we analyzed the distribution of performance metrics across validation sets using One-Hot encoding (Figure 4.1); comprehensive performance distributions for all other evaluated encodings are available in the Supplementary Data. The boxplots reveal a crucial insight regarding the "Distance" feature. While sequence-only models (-seq) achieve respectable scores, the inclusion of the distance feature (-dist) yields a substantial performance boost, particularly in the classification tasks. For instance, in the CHANGE-seq classification task (Figure 4.1A), the median AUPR increased from 0.318 to 0.543 upon adding the distance feature. Similarly, for GUIDE-seq (Figure 4.1C), the median AUPR improved from 0.535 to 0.611. Beyond the median improvements, the inclusion of explicit mismatch counts consistently reduces the variance of the predictions (narrower interquartile ranges), suggesting that this feature acts as a robust regularizer, helping the model generalize better across diverse off-target samples where sequence context alone might be ambiguous.

4.2.2 XGBoost Performance on Base Editing (Hanna)

We then applied the XGBoost framework to the Base Editing task using the Hanna dataset. Since this dataset contains only sgRNA sequences, we compared One-Hot vs. K-mer encodings.

To determine the impact of model optimization and feature selection, we analyzed the performance of XGBoost in both its default configuration and after hyperparameter tuning across four feature sets. The results, detailed in Table Table 4.1 , reveal several key insights.

First, **One-Hot encoding consistently outperforms K-mer encoding**, Regardless of the tuning state or the presence of the distance feature, One-Hot models achieved significantly higher Pearson correlations (up to 0.746) compared to K-mer models (peaking at 0.632). This supports the hypothesis that Base Editing efficiency is highly position-dependent, a property better captured by the rigid structure of One-Hot encoding than by the frequency-based K-mer approach.

Second, **Hyperparameter tuning is critical for XGBoost**. As shown in Table 4.1, optimization yielded substantial improvements across all configurations. For the best-

performing model (One-Hot without distance), tuning increased the Pearson correlation from 0.671 to 0.746, representing a relative improvement of approximately 11%.

Table 4.1: Detailed performance comparison (Pearson Correlation) of XGBoost models on the Hanna dataset before and after hyperparameter tuning. The table reports values for both One-Hot and K-mer encodings, with and without the explicit distance feature.

Encoding	Distance Feature	Default Pearson	Tuned Pearson
One-Hot	Included	0.658	0.742
One-Hot	Excluded	0.671	**0.746**
K-mer	Included	0.583	0.628
K-mer	Excluded	0.586	0.632

Interestingly, unlike the KO datasets, the inclusion of the "Distance" feature (mismatch count) did not improve performance for the Hanna dataset; in fact, the models without the explicit distance feature performed marginally better (e.g., 0.746 vs. 0.742). This suggests that for single-sequence input tasks where the target is efficiency rather than off-target specificity, the raw sequence data contains sufficient signal, and the explicit distance may introduce slight noise or redundancy.

4.2.3 Implicit Learning of Biophysical Properties

To further investigate the model's capability to capture biological mechanisms, we tested whether explicitly providing biophysical features could enhance prediction accuracy on the Base Editing task (Hanna dataset). Specifically, we engineered a set of thermodynamic and compositional features, including:

- **GC Content:** The percentage of Guanine and Cytosine bases in the sgRNA.
- **Melting Temperature (T_m):** The temperature at which the DNA double helix dissociates.
- **Gibbs Free Energy (ΔG):** A measure of the thermodynamic stability of the RNA secondary structure.

Our experiments demonstrated that adding these engineered features to the One-Hot representation yielded **no significant improvement** in performance (neither in AUPR nor Pearson correlation). This result is highly significant: it indicates that the

XGBoost model, when trained on raw sequence data (One-Hot), is sufficiently powerful to implicitly infer these thermodynamic properties.

Table 4.2: Comparison of best performance metrics (AUPR for Classification, Pearson for Regression) across datasets and encodings.

Encoding	Classification (AUPR)			Regression (Pearson)		
	GUIDE	CHANGE	Hanna	GUIDE	CHANGE	Hanna
8XL	0.586	0.516	-	0.750	0.469	-
BULGES	**0.673**	0.663	-	0.753	0.498	-
K-MER	0.577	0.391	0.517	0.518	0.390	0.632
LEP	0.619	0.573	-	0.807	0.436	-
MM	0.566	0.390	-	0.527	0.388	-
NPM	0.661	**0.640**	-	0.758	**0.516**	-
OH	0.611	0.543	**0.723**	0.822	0.459	**0.746**
OH5C	0.600	0.590	-	**0.833**	0.481	-

4.2.4 Hyperparameter Robustness and Task Consistency across Base Editors

To assess the robustness of our strategy, we evaluated whether the hyperparameters tuned on the primary dataset (`BE39_MELJUSO`) could be successfully transferred to other Base Editing contexts included in the Hanna dataset. It is important to clarify the experimental setup: while the model hyperparameters (e.g., learning rate, tree depth, regularization) were fixed to the values optimized for `BE39_MELJUSO`, we performed a full **retraining** of the XGBoost model for each specific target editor and cell line. This approach allows us to verify if the structural complexity learned for one Base Editor configuration is generalizable to others without requiring a new expensive search for hyperparameters.

Performance Stability. As illustrated in Figure 4.2, the models demonstrated strong performance across all Base Editor targets. The regression models consistently maintained Pearson correlations > 0.70 across different editors (BE39 and BE4max) and cell lines. This indicates that the optimal configuration found for the Meljuso cell line captures generalizable learning dynamics suitable for Base Editing data in general, and that One-Hot encoding provides a robust feature set for different deaminases.

Consistency Between Tasks. Furthermore, we analyzed the relationship between the Regression (predicting continuous z-score) and Classification (predicting active guides with $z < -2.0$) tasks. Our results show a high degree of consistency between the two formulations for Base Editing. As shown in Figure 4.3, targets that yielded

high Pearson correlations in the regression task also achieved high AUPR scores in the classification setting. This alignment confirms that the predictive signal regarding editing efficiency is strong and stable, allowing XGBoost to effectively rank sgRNAs regardless of whether the problem is framed as a continuous prediction or a binary selection.

4.2.5 Error Analysis: Limits of Regression at the Extremes

To understand the performance ceiling of the regression model (Pearson ≈ 0.75), we analyzed the distribution of prediction errors. Figure 4.4 visualizes the relationship between predicted and true z-scores.

The scatter plot (Figure 4.4, left) reveals a distinct pattern: the model performs reliably within the standard range (z-scores between -2 and +2), where data is abundant. However, for the "Worst 10%" of predictions (highlighted in red), which correspond to extreme outliers (true z-scores < -4.0), the model consistently under-predicts the magnitude of the effect. Instead of predicting extreme negative values, the model regresses towards the mean.

This behavior is characteristic of gradient boosting algorithms when facing data scarcity in the tails of the distribution. Since high-efficiency guides are rare events, the model lacks sufficient training examples to learn the specific sequence rules governing these extreme phenotypes, preferring a conservative prediction. This limitation suggests that while the *ranking* of these guides remains generally correct (they are predicted as low, just not low enough), the *exact quantitative prediction* is inherently limited by the dataset distribution.

4.2.6 Explainability: Impact of Feature Representation on Model Decisions

To verify that the models rely on biologically meaningful patterns rather than statistical artifacts, we employed a multi-faceted explainability strategy based on SHAP (SHapley Additive exPlanations) and Accumulated Local Effects (ALE).

Global Feature Importance (SHAP). Across all tasks and encoding strategies, SHAP analysis consistently identifies the **Distance** feature (mismatch count) as the dominant driver of model predictions (Figs. 4.5 and 4.6). High distance values are associated with strongly negative SHAP values, indicating a pronounced reduction in predicted cleavage probability as the number of mismatches increases. This behavior is fully

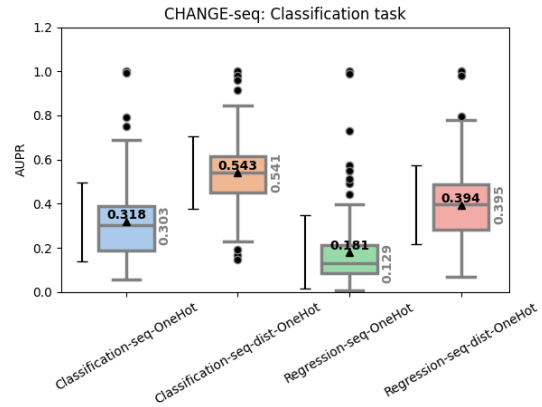
consistent with the established biological principle that increasing mismatches progressively destabilize Cas9–DNA binding.

Quantifying the Impact of Mismatch Count (ALE). To isolate and quantify the effect of mismatches on model predictions while accounting for feature correlations, we analyzed the *Distance* feature using ALE plots (Fig.4.8). The ALE plot reveals a clear trend: even small numbers of mismatches (distance 1–2) produce a measurable negative effect on the predicted score, which intensifies sharply as the distance increases. This confirms that the model has learned a stable and biologically plausible relationship between mismatch count and cleavage efficiency, correctly penalizing off-targets proportional to their divergence from the target sequence.

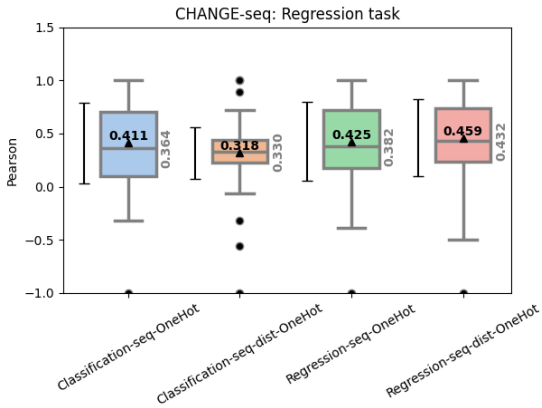
Sequence-Level Interpretability and Encoding Effects. Beyond the global influence of mismatch count, the choice of sequence encoding substantially affects the interpretability of learned patterns, as revealed by SHAP analyses (Figs. 4.5 and 4.6). Models trained with One-Hot encoding highlight positional nucleotide effects, particularly at distal positions (e.g., position 1) and within the seed region (e.g., position 9), indicating that specific locations along the guide sequence contribute disproportionately to model decisions. In contrast, the Bulges encoding provides higher-resolution biological insight by explicitly modeling mismatch types. In both classification and regression tasks, the model identifies specific transitions (e.g., G$\rightarrow$A at position 11) as highly influential features. Since position 11 lies within the seed region, the ability to distinguish mismatch types at this location suggests that the Bulges representation enables the model to capture fine-grained, thermodynamically relevant effects that are not accessible through simpler One-Hot encodings.

Interpretability on Base Editing (Hanna Dataset). We extended the SHAP analysis to the Hanna dataset (Fig. 4.7) to compare how the model interprets single-sequence data for base editing efficiency classification. The **One-Hot** model (Fig. 4.7, left) clearly identifies specific nucleotides in the PAM-proximal region (e.g., `pos_20_A`, `pos_17_A`) as key determinants, consistent with the known biology of the Cas9 seed region. Conversely, the **K-mer** model (Fig. 4.7, right) relies on the frequency of short subsequences (e.g., `kmer_GGC`, `kmer_AAA`). While K-mer encoding captures local composition, it loses precise positional information. This explains the superior performance of One-Hot encoding observed in Table 1: for base editing, the *exact position* of a nucleotide relative to the PAM is more predictive than its mere presence in a local motif.

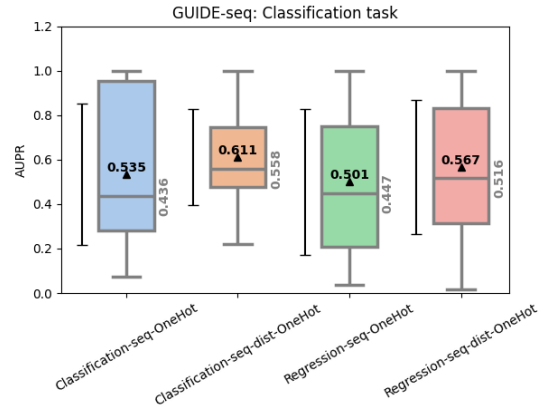
4.3 Results



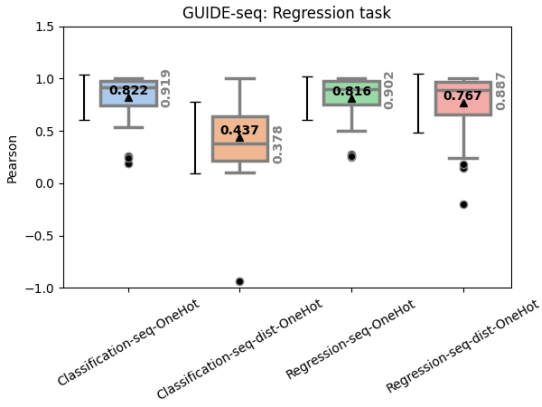
(A) CHANGE-seq Classification (AUPR)



(B) CHANGE-seq Regression (Pearson)



(C) GUIDE-seq Classification (AUPR)



(D) GUIDE-seq Regression (Pearson)

Figure 4.1: **Performance distribution of XGBoost models trained with One-Hot encoding.** The boxplots compare models using only sequence features (`-seq`) versus those including the mismatch count (`-seq-dist`).

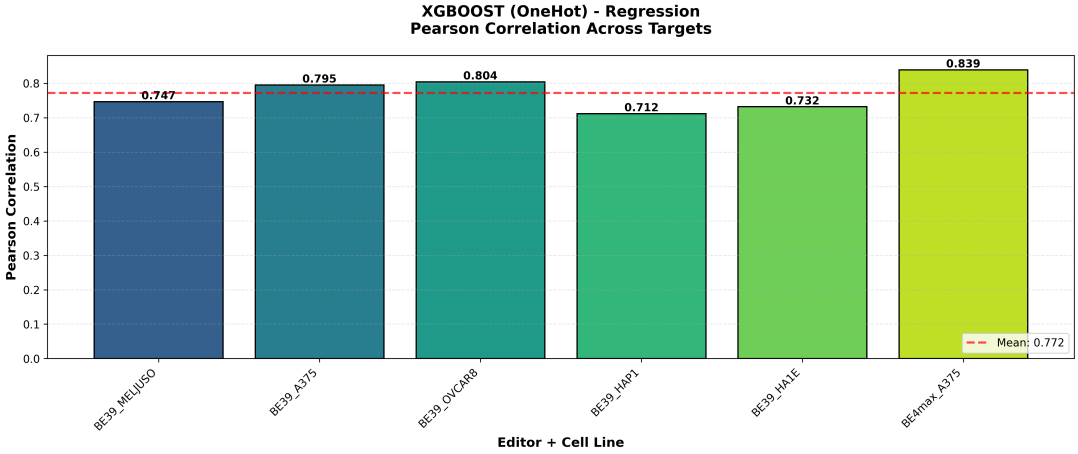


Figure 4.2: **XGBoost (One-Hot) regression performance across targets.** The bar chart shows the Pearson correlation of models trained and tested on various cell lines and editors. The model demonstrates strong generalization capabilities, maintaining high correlations across different Base Editor contexts (BE39, BE4max), confirming the robustness of the learned features.

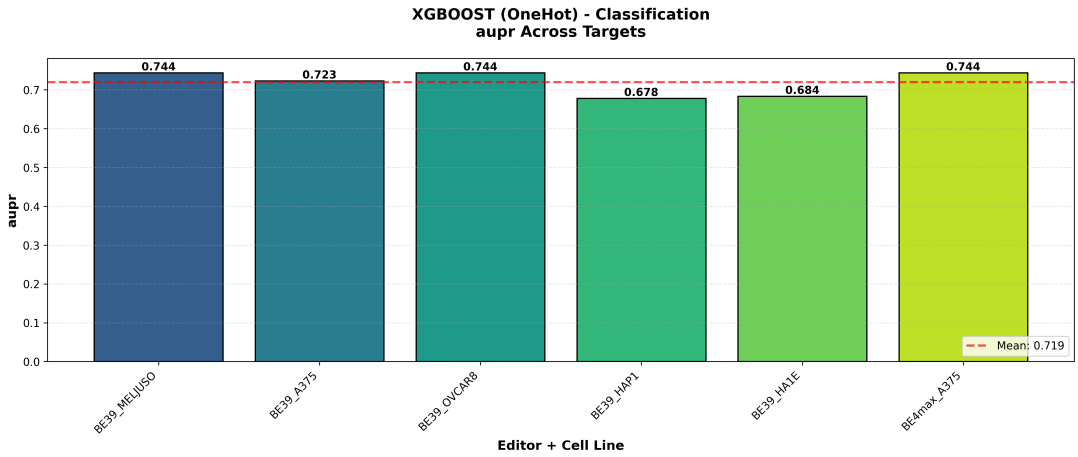


Figure 4.3: **XGBoost (One-Hot) classification performance across targets.** The classification model (AUPR) maintains strong performance across all targets. The results show strong alignment with the regression metrics, confirming that the model consistently identifies effective guides across different Base Editing contexts.

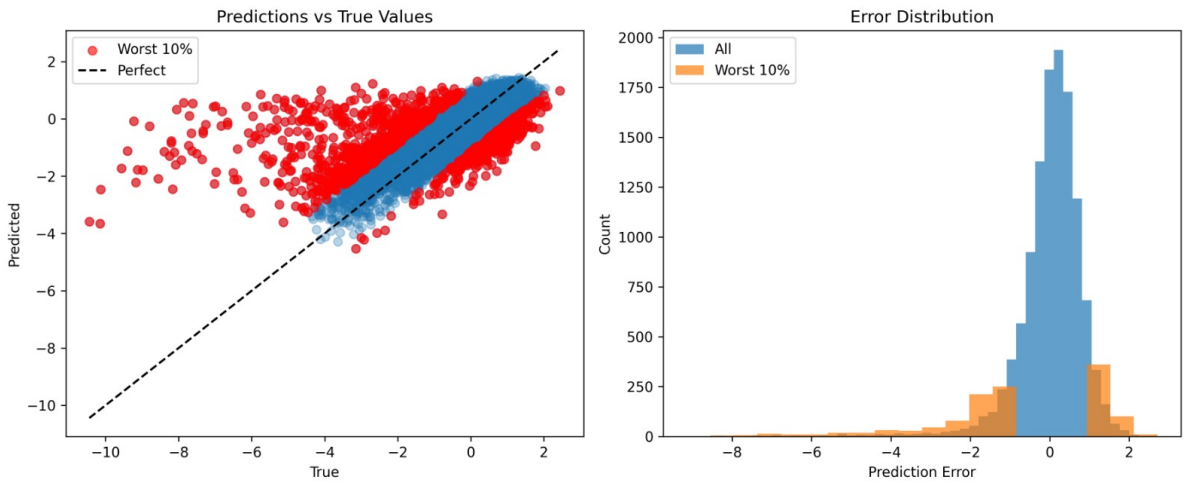


Figure 4.4: **Analysis of Prediction Errors on the Hanna Dataset.** The scatter plot (left) compares predicted vs. true z-scores, with the top 10% largest errors highlighted in red. The model struggles to predict extreme negative values (outliers < -4), showing a tendency to regress towards the mean due to data scarcity in the tails. The histogram (right) confirms that these outliers constitute the long tail of the error distribution.

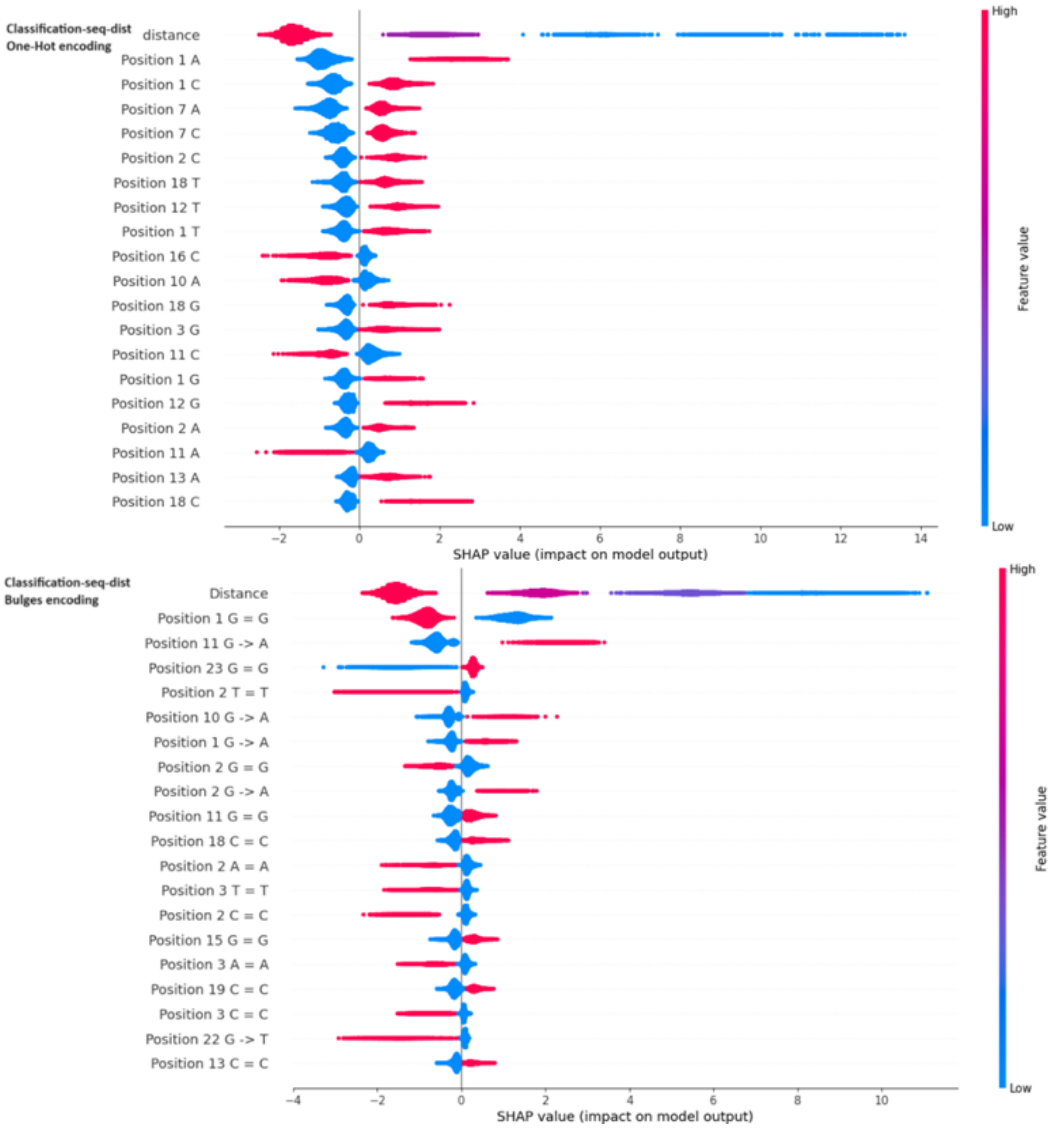


Figure 4.5: **SHAP summary plots for classification.** Top: One-Hot encoding. Bottom: Bulges encoding. Distance is the most influential feature in both models.

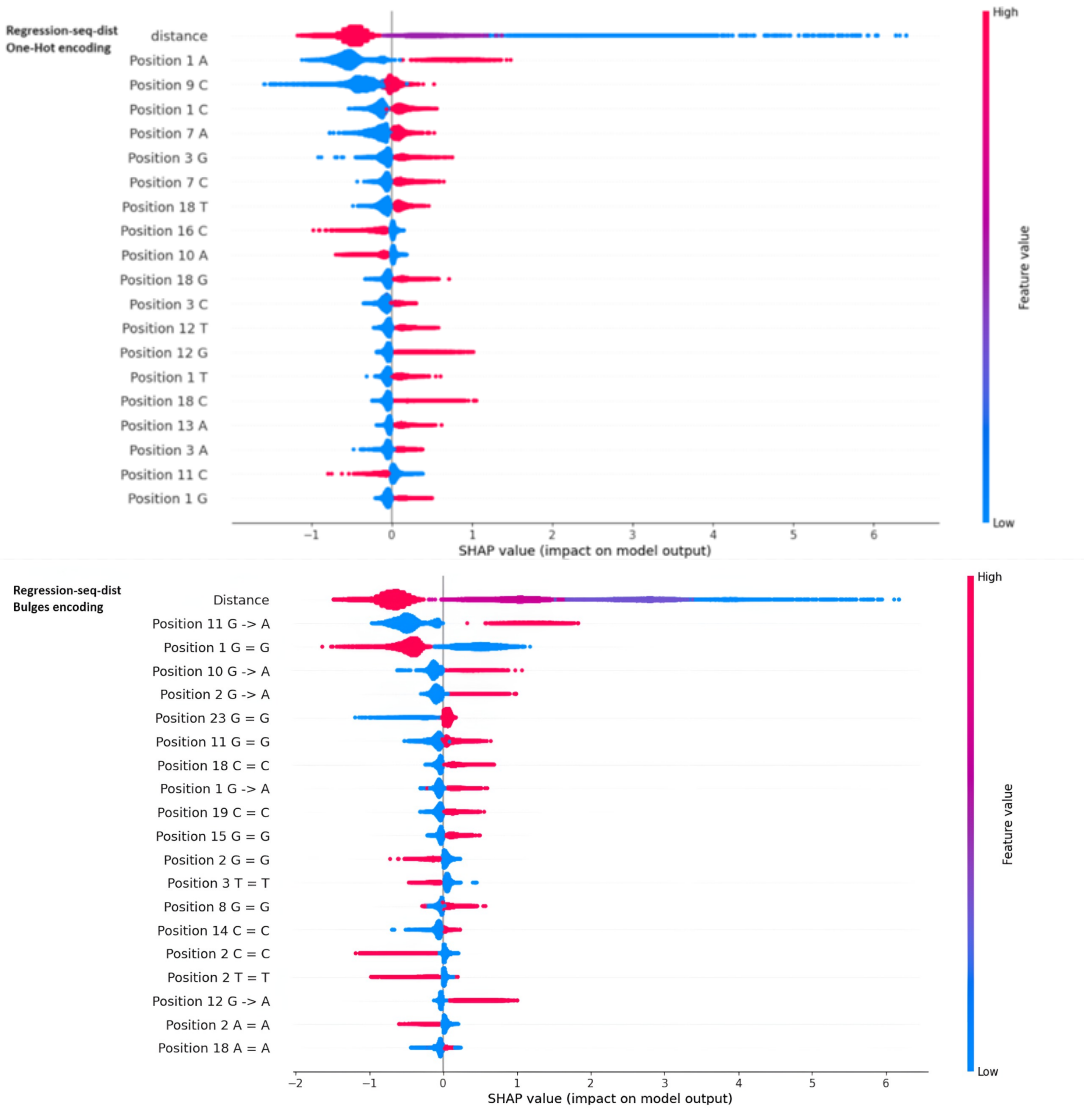
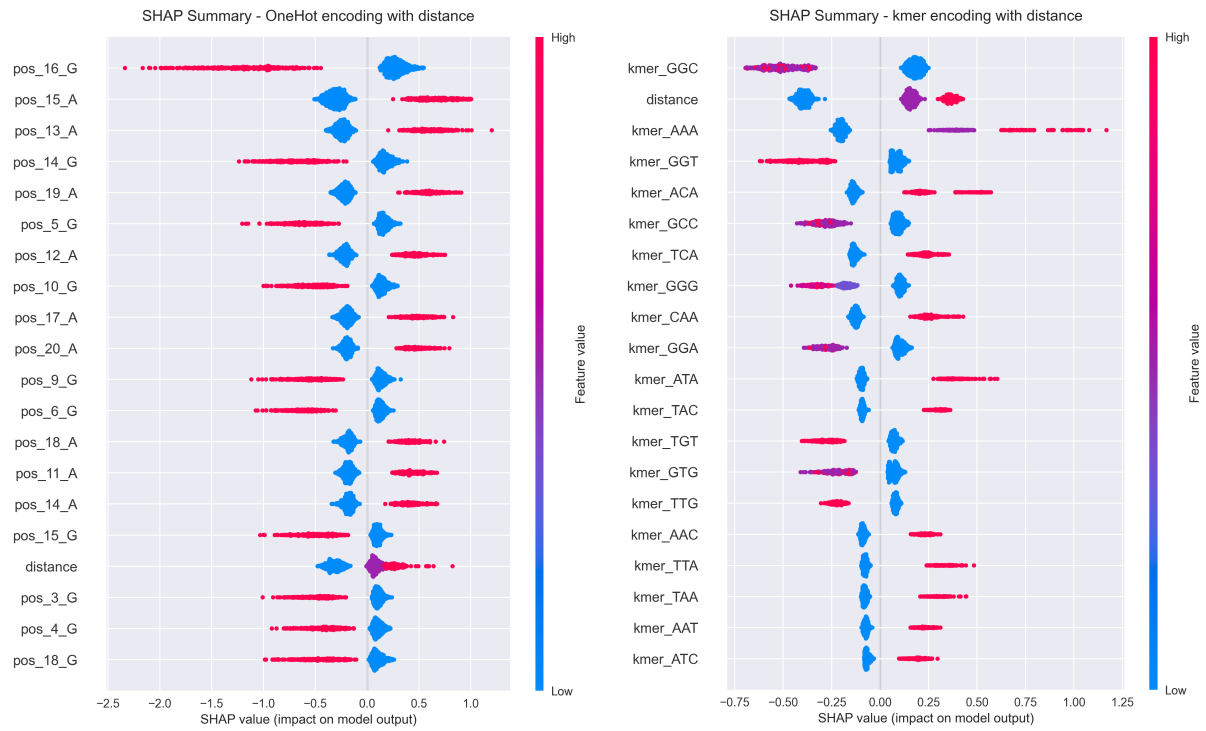


Figure 4.6: **SHAP summary plots for regression.** Top: One-Hot encoding. Bottom: Bulges encoding. Bulges encoding improves biological interpretability by highlighting mismatch types.



(A) One-Hot encoding **(B)** K-mer encoding

Figure 4.7: **SHAP summary plots for Hanna dataset classification.** (A) The One-Hot model highlights strong positional dependencies (e.g., pos_20, pos_17), consistent with the biological relevance of the seed region. (B) The K-mer model captures compositional sequence motifs but lacks explicit positional resolution, resulting in reduced interpretability.

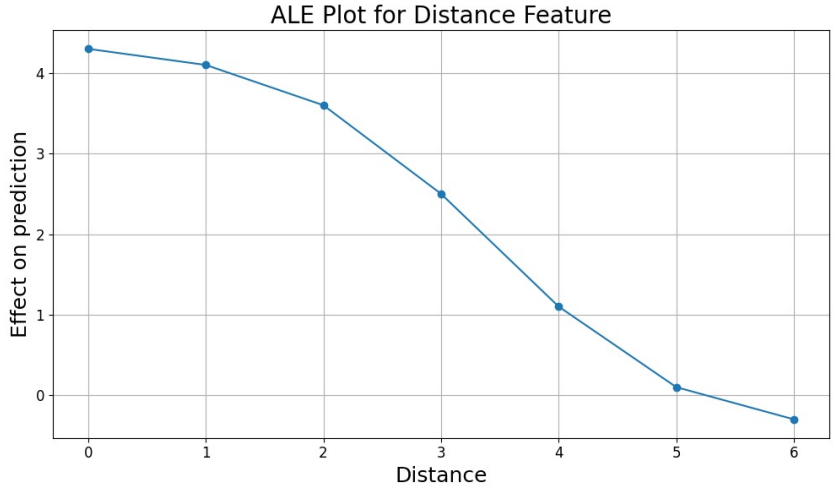


Figure 4.8: **Accumulated Local Effects (ALE) analysis of the ”Distance” feature.** The plot quantifies the marginal effect of mismatch count on the prediction while accounting for feature correlations. It highlights the model's sensitivity to even small numbers of mismatches (1–3), showing a sharp decline in predicted activity as genetic distance increases.

5 Conclusion

This chapter presents the overall conclusions of the thesis by summarizing the main contributions, addressing the research questions, and highlighting how each part of the work advances the understanding of interpretable and explainable CRISPR/Cas9 predictive modeling. The thesis demonstrates how predictive accuracy and biological interpretability can be effectively combined to support reliable and meaningful CRISPR applications.

5.1 Summary of Contributions

This thesis is structured around three main components: a systematic scoping review of CRISPR prediction tools, the development of an explainable ensemble framework for gRNA efficiency prediction, and an analysis of feature representations for off-target and base editing modeling. Together, these components provide a comprehensive investigation into the integration of accuracy and interpretability in CRISPR/Cas9 predictive systems.

5.1.1 Systematic Scoping Review

Chapter 2 presented a detailed scoping review of CRISPR/Cas9 predictive models published between 2014 and 2023. A total of 357 studies were screened, of which 82 models satisfied the inclusion criteria. The analysis revealed a rapid increase in deep learning–based approaches, particularly after 2019, with models such as CNNs, RNNs, BiLSTMs, and transformers becoming dominant.

Despite these advances, a significant limitation identified in the literature is the lack of interpretability. Only a small number of models provided interpretable outputs, and even fewer offered explicit explainability mechanisms. As a result, most existing tools behave as black boxes, limiting their applicability for biological interpretation and clini-

cal decision-making. Furthermore, the majority of models are strongly biased toward knockout datasets, highlighting a lack of diversity in training data.

Based on interpretability, methodological diversity, and availability, four representative models—CRISPRedict, CRISPRAidit, DeepHF, and CRISPRon—were selected for further investigation. These models provide a balanced combination of linear, deep learning, and feature-engineered approaches.

5.1.2 Explainable Ensemble and Research Questions

Chapter 3 addressed the two main research questions of the thesis through the development of an explainable ensemble framework based on XGBoost.

With respect to the first research question (RQ1), the results demonstrate that predictive accuracy can be significantly improved without compromising interpretability. The XGBoost meta-learner, trained on the outputs of the four base models, consistently outperformed all individual models, including traditional approaches such as linear regression, support vector machines, and multilayer pperceptrons,and showed that the prediction obtained a correlation of 0.611, showing a very accurate and consistent output, whereas CRISPRAidit and CRISPRon demonstrated the strongest individual connection with GT1. This experiment supported the idea that the combination of several scoring methods increases the reliability of the prediction, especially when rigorous valid biological techniques and ensemble learning are used for the analysis.

Importantly, SHAP-based analysis enabled full interpretability of the ensemble predictions, allowing the identification of key biologically meaningful features such as G21, G20, C18, and C13. Moreover, the additive structure of SHAP allowed reconstruction of the model predictions with near-perfect correlation, confirming the transparency of the ensemble.

These findings confirm that it is indeed possible to improve prediction performance while preserving interpretability, providing a strong positive answer to RQ1.

Regarding the second research question (RQ2), the results clearly show that knowledge learned in the knockout (KO) domain cannot be directly transferred to the base editing (BE) domain. Models trained on KO data performed poorly when applied to BE datasets, as reflected by negative R^2 values. This is explained by fundamental biological differences between the two domains: KO relies on double-strand breaks and indel formation, whereas BE involves targeted nucleotide modifications.

Although models' performance remained insufficient without retraining. When all models were retrained on BE data, strong performance was recovered, with the ensemble achieving high correlation values with a correlation of 0.68 with the original z-scores GT, with CRISPROn 0.68 as well, with deepHF 0.59, with crisprEdict 0.49 while with CRISPRAidit was 0.40. This demonstrates that effective BE prediction requires domain-specific retraining, although partial transferability may exist in models with biologically rich feature representations.

5.1.3 Feature Representation for Off-Target Prediction

Chapter 4 investigated the impact of different feature encoding strategies on model performance and interpretability using datasets such as CHANGE-seq, GUIDE-seq, and Hanna.

The results demonstrate that for both classification and regression, One-Hot and its variants (OH, OH5C) achieve the best results on GUIDE-seq (AUPR=0.661, Pearson=0.756), while the Bulges representation performs best on CHANGE-seq (AUPR=0.612, Pearson=0.602). In the case of BE, One-hot encoding consistently outperforms K-mer representation for predictive accuracy both as regression and classification (AUPR=0.723, Pearson=0.746). Furthermore, our analysis demonstrates comparable predictive performance across both gene knockout and base editing tasks, confirming the robustness of the framework in distinct editing domains. Interpretability analysis using SHapley Additive exPlanations (SHAP) reveals that despite different mechanisms, the Protospacer Adjacent Motif (PAM)-proximal region remains a critical feature for prediction for both editing mechanisms.

Furthermore, XGBoost showed strong versatility across different task formulations. A high level of consistency was observed between regression and classification settings, indicating that the model effectively captures the underlying biological signal and can reliably rank gRNA sequences. These findings strengthen confidence in the model, as they demonstrate that predictive performance is grounded in biologically relevant mechanisms rather than arbitrary patterns.

5.2 Future Work

Several directions for future research can further enhance the framework proposed in this thesis. First, expanding the methodology to incorporate a broader range of

datasets, including CHANGE-seq and GUIDE-seq, would improve model generalizability and enable more robust biological applications. Second, the integration of uncertainty-aware modeling approaches would allow the estimation of prediction confidence, which is particularly important for clinical and therapeutic use cases.

In addition, future work should explore more context-aware models for base editing, incorporating biological information such as chromatin accessibility and transcriptomic data (e.g., RNA-seq), which may improve both predictive accuracy and biological interpretability. The development of interactive visualization tools, such as web-based platforms for real-time SHAP analysis, would also facilitate model usability and adoption by the broader research community.

Finally, extending the framework to multi-task learning represents a promising direction. Training models jointly on multiple CRISPR outcomes, including indel formation, base editing efficiency, off-target effects, and potential toxicity, could provide a more comprehensive and practical solution for real-world CRISPR guide design.

Funding

This project, including Faiza Hasin PhD Fellowship, is funded by the European Union - Next Generation EU, Mission 4, Component 2, CUP B93D21010860004.

Availability of data and software code

Our software codes are available at the following URL:
`https://github.com/faizamaqsood782-ship-it/Explainable-Methods`.

Bibliography

[1] Dana Carroll. “Collateral damage: benchmarking off-target effects in genome editing”. In: *Genome biology* 20.1 (2019), p. 114.

[2] Tim Wang et al. “Identification and characterization of essential genes in the human genome”. In: *Science* 350.6264 (2015), pp. 1096–1101.

[3] In Sokra et al. “CRISPR–Cas9 Genome Editing in Microorganisms: Principles, Tools, and Applications”. In: *Journal of Agriculture and Technology* 2.1 (2026), pp. 118–128.

[4] Pooja Saraswat and Rajiv Ranjan. “Unlocking the potential of CRISPR tools and databases for precision genome editing”. In: *Frontiers in Plant Science* 16 (2025), p. 1563711.

[5] Martin Jinek et al. “A programmable dual-RNA–guided DNA endonuclease in adaptive bacterial immunity”. In: *science* 337.6096 (2012), pp. 816–821.

[6] Ciaran M Lee et al. “Nuclease target site selection for maximizing on-target activity and minimizing off-target effects in genome editing”. In: *Molecular Therapy* 24.3 (2016), pp. 475–487.

[7] David H Wolpert. “Stacked generalization”. In: *Neural networks* 5.2 (1992), pp. 241–259.

[8] Vasileios Konstantakos et al. “CRISPRedict: a CRISPR-Cas9 web tool for interpretable efficiency predictions”. In: *Nucleic Acids Research* 50.W1 (2022), W191–W198.

[9] Daqi Wang et al. “Optimized CRISPR guide RNA design for two high-fidelity Cas9 variants by deep learning”. In: *Nature communications* 10.1 (2019), p. 4284.

[10] X Xiang et al. *Enhancing CRISPR-Cas9 gRNA efficiency prediction by data integration and deep learning. Nat. Commun. 12, 3238*. 2021.

[11] Heng Zhang et al. “Deep sampling of gRNA in the human genome and deep-learning-informed prediction of gRNA activities”. In: *Cell discovery* 9.1 (2023), p. 48.

[12] Xiaolong Cheng et al. “Modeling CRISPR-Cas13d on-target and off-target effects using machine learning approaches”. In: *Nature communications* 14.1 (2023), p. 752.

[13] Oscar A Zarate et al. “BoostMEC: predicting CRISPR-Cas9 cleavage efficiency through boosting models”. In: *BMC bioinformatics* 23.1 (2022), p. 446.

[14] Yanyi Zheng et al. “CRISPR-MFH: A Lightweight Hybrid Deep Learning Framework with Multi-Feature Encoding for Improved CRISPR-Cas9 Off-Target Prediction”. In: *Genes* 16.4 (2025), p. 387.

[15] Condy Bao and Fuxiao Liu. “DeepFM-Crispr: Prediction of CRISPR On-Target Effects via Deep Learning”. In: *arXiv preprint arXiv:2409.05938* (2024).

[16] Hilary Arksey and Lisa O’malley. “Scoping studies: towards a methodological framework”. In: *International journal of social research methodology* 8.1 (2005), pp. 19–32.

[17] Yanfang Fu et al. “High-frequency off-target mutagenesis induced by CRISPR-Cas nucleases in human cells”. In: *Nature biotechnology* 31.9 (2013), pp. 822–826.

[18] Jennifer A Doudna and Emmanuelle Charpentier. “The new frontier of genome engineering with CRISPR-Cas9”. In: *Science* 346.6213 (2014), p. 1258096.

[19] Maximilian Haeussler et al. “Evaluation of off-target and on-target scoring algorithms and integration into the guide RNA selection tool CRISPOR”. In: *Genome biology* 17 (2016), pp. 1–12.

[20] Daesik Kim et al. “Digenome-seq: genome-wide profiling of CRISPR-Cas9 off-target effects in human cells”. In: *Nature methods* 12.3 (2015), pp. 237–243.

[21] Jennifer Listgarten et al. “Prediction of off-target activities for the end-to-end design of CRISPR guide RNAs”. In: *Nature biomedical engineering* 2.1 (2018), pp. 38–47.

[22] Scott M Lundberg and Su-In Lee. “A unified approach to interpreting model predictions”. In: *Advances in neural information processing systems* 30 (2017).

[23] Guishan Zhang, Zhiming Dai, and Xianhua Dai. “A novel hybrid CNN-SVR for CRISPR/Cas9 guide RNA activity prediction”. In: *Frontiers in genetics* 10 (2020), p. 1303.

[24] Tim Wang et al. “Genetic screens in human cells using the CRISPR-Cas9 system”. In: *Science* 343.6166 (2014), pp. 80–84.

[25] Tessa G Montague et al. “CHOPCHOP: a CRISPR/Cas9 and TALEN web tool for genome editing”. In: *Nucleic acids research* 42.W1 (2014), W401–W407.

[26] Lihua J Zhu et al. “CRISPRseek: a bioconductor package to identify target-specific guide RNAs for CRISPR-Cas9 genome-editing systems”. In: *PloS one* 9.9 (2014), e108424.

[27] Sangsu Bae, Jeongbin Park, and Jin-Soo Kim. “Cas-OFFinder: a fast and versatile algorithm that searches for potential off-target sites of Cas9 RNA-guided endonucleases”. In: *Bioinformatics* 30.10 (2014), pp. 1473–1475.

[28] Han Xu et al. “Sequence determinants of improved CRISPR sgRNA design”. In: *Genome research* 25.8 (2015), pp. 1147–1157.

[29] Yuki Naito et al. “CRISPRdirect: software for designing CRISPR/Cas guide RNA with reduced off-target sites”. In: *Bioinformatics* 31.7 (2015), pp. 1120–1123.

[30] Ritambhara Singh et al. “Cas9-chromatin binding information enables more accurate CRISPR off-target prediction”. In: *Nucleic acids research* 43.18 (2015), e118–e118.

[31] Manuel Stemmer et al. “CCTop: an intuitive, flexible and reliable CRISPR/Cas9 target prediction tool”. In: *PloS one* 10.4 (2015), e0124633.

[32] Miguel A Moreno-Mateos et al. “CRISPRscan: designing highly efficient sgRNAs for CRISPR-Cas9 targeting in vivo”. In: *Nature methods* 12.10 (2015), pp. 982–988.

[33] John G Doench et al. “Optimized sgRNA design to maximize activity and minimize off-target effects of CRISPR-Cas9”. In: *Nature biotechnology* 34.2 (2016), pp. 184–191.

[34] Iman Farasat and Howard M Salis. “A biophysical model of CRISPR/Cas9 activity for rational design of genome editing and gene regulation”. In: *PLoS computational biology* 12.1 (2016), e1004724.

[35] Yuanyan Xiong et al. “pgRNAFinder: a web-based tool to design distance independent paired-gRNA”. In: *Bioinformatics* 33.22 (2017), pp. 3642–3644.

[36] Md Khaledur Rahman and M Sohel Rahman. “CRISPRpred: a flexible and efficient tool for sgRNAs on-target activity prediction in CRISPR/Cas9 systems”. In: *PloS one* 12.8 (2017), e0181943.

[37] Shiran Abadi et al. “A machine learning approach for predicting CRISPR-Cas9 cleavage efficiencies and patterns underlying its mechanism of action”. In: *PLoS computational biology* 13.10 (2017), e1005807.

[38] Alexey A Shvets and Anatoly B Kolomeisky. “Mechanism of genome interrogation: How CRISPR RNA-guided Cas9 proteins locate specific targets on DNA”. In: *Biophysical journal* 113.7 (2017), pp. 1416–1424.

[39] Ferhat Alkan et al. “CRISPR-Cas9 off-targeting assessment with nucleic acid duplex energy parameters”. In: *Genome biology* 19.1 (2018), p. 177.

[40] Keith R Anderson et al. “CRISPR off-target analysis in genetically engineered rats and mice”. In: *Nature methods* 15.7 (2018), pp. 512–514.

[41] Maurice Labuhn et al. “Refined sgRNA efficacy prediction improves large-and small-scale CRISPR–Cas9 applications”. In: *Nucleic acids research* 46.3 (2018), pp. 1375–1385.

[42] Jean-Paul Concordet and Maximilian Haeussler. “CRISPOR: intuitive guide selection for CRISPR/Cas9 genome editing experiments and screens”. In: *Nucleic acids research* 46.W1 (2018), W242–W245.

[43] Laurence OW Wilson et al. “High activity target-site identification using phenotypic independent CRISPR-Cas9 core functionality”. In: *The CRISPR Journal* 1.2 (2018), pp. 182–190.

[44] Hui Peng et al. “CRISPR/Cas9 cleavage efficiency regression through boosting algorithms and Markov sequence profiling”. In: *Bioinformatics* 34.18 (2018), pp. 3069–3077.

[45] Brian J Mendoza and Cong T Trinh. “Enhanced guide-RNA design and targeting analysis for precise CRISPR genome editing of single and consortia of industrially relevant and non-model organisms”. In: *Bioinformatics* 34.1 (2018), pp. 16–23.

[46] Jiecong Lin and Ka-Chun Wong. “Off-target predictions in CRISPR-Cas9 gene editing using deep learning”. In: *Bioinformatics* 34.17 (2018), pp. i656–i663.

[47] Guohui Chuai et al. “DeepCRISPR: optimized CRISPR guide RNA design by deep learning”. In: *Genome biology* 19.1 (2018), p. 80.

[48] Giovanni Dimauro et al. “Crisprlearner: a deep learning-based system to predict crispr/cas9 sgrna on-target cleavage efficiency”. In: *Electronics* 8.12 (2019), p. 1478.

[49] Shixiong Zhang et al. “Synergizing CRISPR/Cas9 off-target predictions for ensemble insights and practical applications”. In: *Bioinformatics* 35.7 (2019), pp. 1108–1115.

[50] Qiao Liu, Di He, and Lei Xie. “Prediction of off-target specificity and cell-specific fitness of CRISPR-Cas System using attention boosted deep learning and network-based gene feature”. In: *PLoS computational biology* 15.10 (2019), e1007480.

[51] Jean-Philippe Fortin et al. “Multiple-gene targeting and mismatch tolerance can confound analysis of genome-wide pooled CRISPR screens”. In: *Genome biology* 20.1 (2019), p. 21.

[52] Josh Tycko et al. “Mitigation of off-target toxicity in CRISPR-Cas9 screens for essential non-coding elements”. In: *Nature communications* 10.1 (2019), p. 4063.

[53] Kornel Labun et al. “CHOPCHOP v3: expanding the CRISPR web toolbox beyond genome editing”. In: *Nucleic acids research* 47.W1 (2019), W171–W174.

[54] Joshua Young et al. “CRISPR-Cas9 editing in maize: systematic evaluation of off-target activity and its relevance in crop improvement”. In: *Scientific reports* 9.1 (2019), p. 6729.

[55] Rohan Dandage et al. “beditor: a computational workflow for designing libraries of guide RNAs for CRISPR-mediated base editing”. In: *Genetics* 212.2 (2019), pp. 377–385.

[56] Yu Zhang et al. “DL-CRISPR: a deep learning method for off-target activity prediction in CRISPR/Cas9 with data augmentation”. In: *IEEE Access* 8 (2020), pp. 76610–76617.

[57] Guishan Zhang, Zhiming Dai, and Xianhua Dai. “C-RNNCrispr: Prediction of CRISPR/Cas9 sgRNA activity using convolutional and recurrent neural networks”. In: *Computational and structural biotechnology journal* 18 (2020), pp. 344–354.

[58] Dong Chen, Wenjie Shu, and Shaoliang Peng. “Predicting crispr-cas9 off-target with self-supervised neural networks”. In: *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE. 2020, pp. 245–250.

[59] Jiecong Lin et al. “CRISPR-net: a recurrent convolutional network quantifies CRISPR off-target activities with mismatches and indels”. In: *Advanced science* 7.13 (2020), p. 1903562.

[60] Jifang Yan et al. “Benchmarking and integrating genome-wide CRISPR off-target detection and prediction”. In: *Nucleic acids research* 48.20 (2020), pp. 11370–11379.

[61] Ali Haisam Muhammad Rafid et al. “CRISPRpred (SEQ): a sequence-based method for sgRNA on target activity prediction using traditional machine learning”. In: *BMC bioinformatics* 21.1 (2020), p. 223.

[62] Jaspreet Kaur Dhanjal et al. “Evaluation of off-targets predicted by sgRNA design tools”. In: *Genomics* 112.5 (2020), pp. 3609–3614.

[63] Xiaojian Liu et al. “SeqCor: correct the effect of guide RNA sequences in clustered regularly interspaced short palindromic repeats/Cas9 screening by machine learning algorithm”. In: *Journal of Genetics and Genomics* 47.11 (2020), pp. 672–680.

[64] Meng Wang et al. “Fibrinogen alpha chain knockout promotes tumor growth and metastasis through integrin–AKT signaling pathway in lung cancer”. In: *Molecular Cancer Research* 18.7 (2020), pp. 943–954.

[65] Gue-Ho Hwang et al. “CRISPR-sub: Analysis of DNA substitution mutations caused by CRISPR-Cas9 in human cells”. In: *Computational and structural biotechnology journal* 18 (2020), pp. 1686–1694.

[66] Kasidet Hiranniramol et al. “Generalizable sgRNA design for improved CRISPR/Cas9 editing efficiency”. In: *Bioinformatics* 36.9 (2020), pp. 2684–2689.

[67] Li-Ming Xiao, Yun-Qi Wan, and Zhen-Ran Jiang. “AttCRISPR: a spacetime interpretable model for prediction of sgRNA on-target activity”. In: *BMC bioinformatics* 22.1 (2021), p. 589.

[68] Guishan Zhang et al. “Prediction of CRISPR/Cas9 single guide RNA cleavage efficiency and specificity by attention-based convolutional neural networks”. In: *Computational and structural biotechnology journal* 19 (2021), pp. 1445–1457.

[69] Wei He et al. “GuidePro: a multi-source ensemble predictor for prioritizing sgRNAs in CRISPR/Cas9 protein knockouts”. In: *Bioinformatics* 37.1 (2021), pp. 134–136.

[70] Huibao Feng et al. “Guide-target mismatch effects on dCas9–sgRNA binding activity in living bacterial cells”. In: *Nucleic Acids Research* 49.3 (2021), pp. 1263–1277.

[71] Yongxian Fan and Haibo Xu. “Prediction of off-target effects in CRISPR/Cas9 system by ensemble learning”. In: *Current Bioinformatics* 16.9 (2021), pp. 1169–1178.

[72] Jeremy Charlier, Robert Nadon, and Vladimir Makarenkov. “Accurate deep learning off-target prediction with novel sgRNA-DNA sequence encoding in CRISPR-Cas9 gene editing”. In: *Bioinformatics* 37.16 (2021), pp. 2299–2307.

[73] Prasoon Kumar Vinodkumar, Cagri Ozcinar, and Gholamreza Anbarjafari. “Prediction of sgrna off-target activity in crispr/cas9 gene editing using graph convolution network”. In: *Entropy* 23.5 (2021), p. 608.

[74] Victor Aprilyanto et al. “CROP: a CRISPR/Cas9 guide selection program based on mapping guide variants”. In: *Scientific reports* 11.1 (2021), p. 1504.

[75] Hui Kwon Kim et al. “Predicting the efficiency of prime editing guide RNAs in human cells”. In: *Nature Biotechnology* 39.2 (2021), pp. 198–206.

[76] Zhong-Rui Zhang and Zhen-Ran Jiang. “Effective use of sequence information to predict CRISPR-Cas9 off-target”. In: *Computational and structural biotechnology journal* 20 (2022), pp. 650–661.

[77] Rongjie Fu et al. “Systematic decomposition of sequence determinants governing CRISPR/Cas9 specificity”. In: *Nature communications* 13.1 (2022), p. 474.

[78] Ofir Yaish, Maor Asif, and Yaron Orenstein. “A systematic evaluation of data processing and problem formulation of CRISPR off-target site prediction”. In: *Briefings in bioinformatics* 23.5 (2022), bbac157.

[79] Dhvani Sandip Vora, Yugesh Verma, and Durai Sundar. “A machine learning approach to identify the importance of novel features for CRISPR/Cas9 activity prediction”. In: *Biomolecules* 12.8 (2022), p. 1123.

[80] Yuhao Chen and Xiaowei Wang. “Evaluation of efficiency prediction algorithms and development of ensemble model for CRISPR/Cas9 gRNA selection”. In: *Bioinformatics* 38.23 (2022), pp. 5175–5181.

[81] Giulia I Corsi et al. “CRISPRroots: on-and off-target assessment of RNA-seq data in CRISPR–Cas9 edited cells”. In: *Nucleic Acids Research* 50.4 (2022), e20–e20.

[82] Jiecong Lin, Xingjian Chen, and Ka-Chun Wong. “An artificial intelligence approach for gene editing off-target quantification: Convolutional self-attention neural network designs and considerations”. In: *Statistics in Biosciences* 15.3 (2023), pp. 657–668.

[83] Ruth E Hanna et al. “Massively parallel assessment of human variants with base editor screens”. In: *Cell* 184.4 (2021), pp. 1064–1080.