The EuroBiotech Journal
RESEARCH ARTICLE
Bioinformatics

Machine Learning-Driven Prediction of CRISPR-Cas9 Off-Target Effects and Mechanistic Insights

Anuradha Bhardwaj[1], Pradeep Tomar[2*], Vikrant Nain[1*]

Abstract

Background: The precise prediction of off-target effects in CRISPR-Cas9 genome editing is critical for ensuring the safety and efficacy of this powerful tool. This study leverages machine learning techniques to predict off-target cleavage sites and investigate the underlying mechanisms that affect cleavage efficiencies. By integrating data from Tsai et al. and Kleinsteiver et al., who employed the GUIDE-seq method, we aim to enhance our understanding of the factors influencing CRISPR-Cas9 activity.

Results: Our research analyzed datasets from Tsai et al. and Kleinsteiver et al., standardizing cleavage efficiencies to align with Tsai et al.'s comprehensive dataset. We identified a range of sequence features, including PAM sequence types, nucleotide composition, GC content, chromatin structure, CpG islands, and gene expression levels. Various machine learning models, including Artificial Neural Networks, Support Vector Machines, Naïve Bayes, k-Nearest Neighbors, Logistic Regression, and Extra Trees Classifiers, were developed and evaluated. The Extra Trees Classifier, particularly with class weighting, exhibited robust performance, achieving high accuracy, precision, recall, and F1 scores. SHAP analysis provided insights into feature importance, highlighting the significant factors contributing to model predictions.

Conclusions: The application of machine learning to predict CRISPR-Cas9 off-target effects demonstrates significant potential in enhancing the precision of genome editing. Our findings underscore the importance of considering a diverse range of sequence and genomic features to improve prediction models. The insights gained from this study can inform the development of safer and more effective CRISPR-based applications in medicine, agriculture, and biotechnology. Future work will focus on further refining these models and exploring their applicability across different genomic contexts.

Keywords: CRISPR-Cas9, Machine Learning, On-Targets, Off-Targets, Genome Editing

Introduction

Genome editing, also referred to as gene editing, involves the precise manipulation of an organism's DNA by deleting, inserting, or replacing specific sequences within its genome. To achieve this, molecular tools known as designed nucleases, often referred to as molecular scissors, are employed in laboratory settings to modify gene functions through the editing or alteration of DNA segments (1,2). Numerous gene editing techniques, including CRISPR-Cas9, ZFNs, and TALENs, have been developed and applied extensively across various cell types, tissues, and organisms (3–5). However, among these methods, CRISPR-Cas9 stands out as the most widely adopted approach by researchers globally.

CRISPR is an acronym that means Clustered Regularly Interspaced Short Palindromic Repeats, a segment of bacterial DNA containing repetitive base sequences. It forms the basis of natural immunity in bacteria against foreign DNA. When viral DNA is detected, the bacterium generates guide RNA consisting of two short RNA strands. The guide RNA associates with a nuclease enzyme called Cas9 (6) Figure 1. Basically, the CRISPR-Cas9 complex is guided to the viral DNA, where it cuts it, thus inactivating the virus. The enzyme Cas9 does not bind to DNA in the absence of a specific sequence called the Protospacer Adjacent Motif, which enables it to tell the difference between bacterial and viral DNA. Moreover, the CRISPR-Cas9 system can memorize the information about viral DNA that permits it to recognize and inactivate future viral threats. The sequences of CRISPR are present in about 50% of bacterial genomes and 90% of archaeal genomes

[1] School of Biotechnology, Gautam Buddha University, Greater Noida, Uttar Pradesh, 201312, India
[2] School of Information and Communication Technology, Gautam Buddha University, Greater Noida, Uttar Pradesh, 201312, India

***Corresponding authors:**
Vikrant Nain, Assistant Professor, School of Biotechnology, Gautam Buddha University, Greater Noida, Uttar Pradesh-201312, India
E-mail: vikrant@gbu.ac.in
Pradeep Tomar, Assistant Professor, School of Information and Communication Technology, Gautam Buddha University, Greater Noida, Uttar Pradesh-201312, India
E-mail: pradeep.tomar@gbu.ac.in

DOI: 10.2478/ebtj-2024-0020

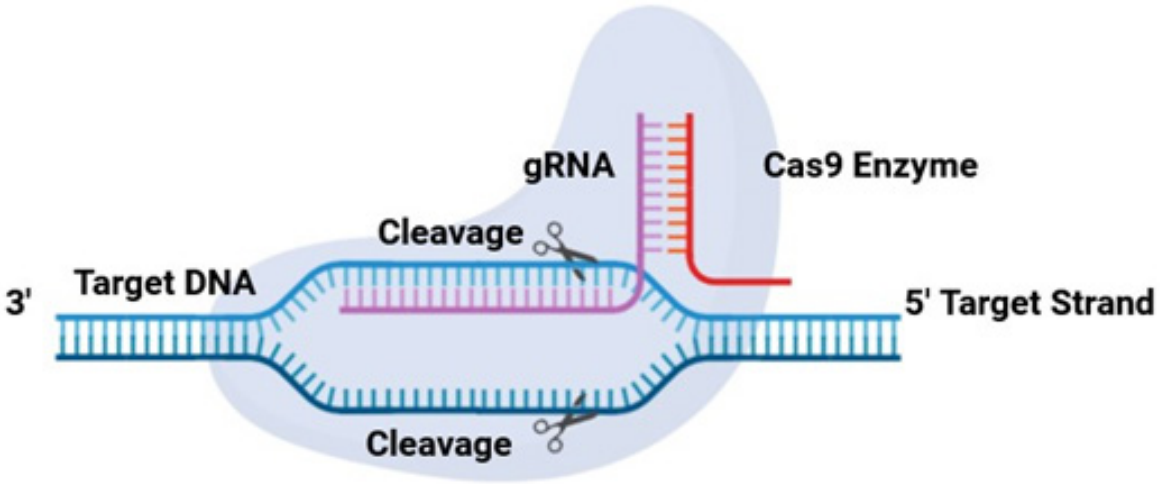

Figure 1. A schematic showing the conventional structure of the CRISPR/Cas9. The CRISPR/Cas9 system is represented by a monomeric protein whereby the Cas9 enzyme forms a complex with a guide RNA, gRNA, that comprises tracrRNA and crRNA. The latter is complementary to 20–22 base pairs of target DNA while the protospacer adjacent motif, PAM—a 2–6 bp sequence, NGG—is necessary for activity of Cas9 and located 3–4 nucleotides downstream from the cut site. The seed sequence must have sufficient homology, located 5 nucleotides upstream of the PAM, so that the gRNA will bind and the Cas9 will cut the DNA, resulting in single-stranded nicks or DSBs.

(3–5,7,8).

The versatility of the CRISPR-Cas9 system in locating and modifying specific genes has made it invaluable in various fields, including medicine, drug discovery, and agriculture. Recent advancements have introduced sequence-based genome editing for crop improvement (9,10). Notably, CRISPR-Cas9 has the potential to address the evolving challenges in crop science and agriculture. This technology is capable of modifying genomic sequences to impart desired traits to organisms, including crop species, as long as the requisite Protospacer Adjacent Motif (PAM) sequence is present. Compared to other genome editing tools like zinc finger nucleases (ZFNs) and transcriptional activator-like effector nucleases (TALENs), CRISPR-Cas9 is more efficient, cost-effective, and precise (11). Since its introduction in 2012, the CRISPR-Cas9 system has gained widespread acceptance among researchers worldwide and has been employed to target numerous crucial genes in a wide array of cell lines and organisms, including bacteria, C. elegans, Xenopus tropicalis, yeast, zebrafish, Drosophila, rabbits, plants, monkeys, humans, rats, and mice. Researchers have used this method to introduce single-point mutations, deletions, or insertions into target genes using single guide RNA (sgRNA)(4). The CRISPR-Cas9 system holds vast potential not only in therapeutics but also in diagnostics, climate change solutions, and pandemic response. It enables precise and rapid disease detection in diagnostics, offers solutions for climate change challenges in agriculture and environmental science, and contributes to pandemic preparedness through specific diagnostic tools and antiviral strategies. The versatility of CRISPR technology spans across critical domains, making it a powerful tool in addressing diverse issues in human health and the environment (12,13)

While sgRNA aims to target specific DNA segments, it can sometimes bind to unintended sites, resulting in off-target mutations. These off-target mutations can alter gene functions and lead to genomic instability, posing a significant challenge in CRISPR-Cas9 gene editing (14). To mitigate these risks, it is imperative to accurately predict off-target sites within the genome. Several off-target prediction methods are available, relying on scoring mechanisms based on mismatch positions within the guide sequence (15,16).These scores are computed through statistical analysis, such as Pearson correlation coefficient, based on prior gene editing experiments. However, many widely used methods focus primarily on basic sequence features and neglect the genomic context surrounding the target sequence (17). For precise prediction of cleavage sites, machine learning and deep learning-based tools have been developed for both humans (18–20) and plants (21,22). These tools incorporate a wide range of features, including those specific to the genomic target, thermodynamics of sgRNA, and pairwise similarity between the sgRNA and the genomic target (23).

In the last couple of years, there has been tremendous development in the application of machine learning models toward off-target prediction of CRISPR-Cas9. Extensive usage of models has been done with SVM and ANN in the prediction of off-target effects due to the abilities of these models in detecting complex patterns within genomic data (24). This has normally posed a few challenges, such as requiring intensive hyperparameter tuning and being prone to overfitting, mostly

while handling high-dimensional or imbalanced datasets (25). The recent area of research has focused on ensemble learning techniques like Random Forest and Gradient Boosting Machines because of their power in the prediction models that gave better accuracy and model robustness (26). These ensemble methods are powerful in reducing variance and bias by combining many weak learners; therefore, they are very suitable for the prediction of CRISPR-Cas9 off-targets. For example, one study in 2024 proved that ensemble methods provide a balanced approach to accuracy and interpretability, especially in genomic data sets with large scales and diverse feature sets (27). However, the extent of their effectiveness on different genomic contexts, beyond specific organisms, remains a less-explored area.

This study aims to refine the prediction of off-target effects in CRISPR-Cas9 genome editing, a critical factor for ensuring the safety and effectiveness of this technology. By synthesizing data from diverse sources and employing a range of machine learning models—including Artificial Neural Networks (ANN), Support Vector Machines (SVM), and Extra Trees Classifiers (ETC)—the research seeks to improve the accuracy of off-target predictions. We examined key features that includes PAM sequence types, nucleotide composition, GC content, chromatin structure, and gene expression levels. Among the models tested, the Extra Trees Classifier with class weighting demonstrated superior performance in terms of accuracy, precision, recall, and F1 score. This strong performance is partly due to the fact that ETC can handle sophisticated interactions between genomic features and has a robustness for overfitting, even when minimal tunings of the hyperparameters are performed (28). Also, class weighting added to the model made the ETC model very robust to class imbalance, a common problem with CRISPR off-target predictions (29). SHAP analysis further highlighted significant genomic features impacting model predictions, emphasizing the importance of feature selection in enhancing prediction reliability (30).

The study concludes that machine learning has considerable potential to advance CRISPR-Cas9 technology, recommending future research to focus on model refinement, exploration of ensemble methods, and application to real-world CRISPR experiments. Overall, the manuscript highlights the essential role of machine learning in achieving safer and more precise genome editing applications.

Materials - Methods

This study aims to investigate off-target effects of CRISPR-Cas genome editing by utilizing GUIDE-seq data from Tsai et al. and Kleinsteiver et al., focusing on the cleavage efficiency of various sgRNAs in U2OS and HEK293 cell lines. The research involved data preprocessing to standardize cleavage efficiencies from different datasets and explored various machine learning models to predict CRISPR off-target sites. Models evaluated included Artificial Neural Networks, Support Vector Machines, Naïve Bayes Classifiers, k-Nearest Neighbours, Logistic Regression, and Extra Trees Classifiers. Each model's performance was assessed using metrics such as AUC-ROC, recall, accuracy, precision, and F1 Score. Feature importance was analysed using SHAP to understand the influence of different attributes on model predictions.

2.1 Data Collection

To investigate the off-target effects of CRISPR-Cas genome editing, we used both Tsai and Kleinsteiver that employed GUIDE-seq (31,32). Tsai et al. used ten different 20-nt sgRNAs, with six applied to endogenous human genes in the U2OS cell line and four in the HEK293 cell line. On the other hand, Kleinsteiver et al. utilized GUIDE-seq with ten sgRNAs in the U2OS cell line. These studies provide experimentally verified genomic targets across various genomes, each associated with the frequency of cleavage by specific sgRNAs. We chose these studies because they help us achieve our research goal of understanding the genetic factors that impact the effectiveness of CRISPR.

We also explored alternative methods for identifying cleavage sites but decided against them because they didn't align with our research goals. For example, Digenome-Seq can't give us cleavage frequencies in living cells, and the Integrase-Defective Lentiviral Vectors (IDLV) method, while good at finding off-targets in vivo, doesn't measure cleavage frequencies. Some studies also focused on specific genomic sites chosen based on existing knowledge instead of doing a complete genome-wide analysis.

2.2 Data Preprocessing

In working with CRISPR-Cas machine learning data, tackling data preprocessing challenges is crucial. This involves tasks like data reduction, cleaning, transformation, and integration (33). Dealing with specific CRISPR-Cas data brings its own complexity, especially in addressing inaccuracies, inconsistencies, and missing values in datasets, impacting the accuracy of prediction models.

Both Tsai and Kleinsteiver employed GUIDE-seq to investigate off-target effects of CRISPR-Cas genome editing. Interestingly, four of the sgRNAs used by Kleinsteiver overlapped with those used by Tsai. These cleaved sites come from various experimental conditions, making their reported cleavage efficiencies incomparable. To address this, we standardized the cleavage efficiencies by aligning them with Tsai dataset, the most comprehensive.

2.3 Calculation of Sequence Features

We explored a wide range of potential features that could explain the outcomes. These features included characteristics specific to the target site and the sgRNA, covering details like the PAM sequence type, nucleotide composition, and GC content. We also examined broader features such as chromatin structure, the existence of CpG islands, and the expression levels of genes in coding regions (19). Furthermore, we considered features

related to how well the sgRNA aligned with the target site, considering the number and distribution of mismatches.

2.4 Machine Learning Models and Experiments

In our pursuit of developing robust models for predicting CRISPR-Cas9 off-target sites, we explored several contemporary classification modelling techniques implemented in the python SciKit-learn module (34). Our experiments encompassed the use of Artificial Neural Networks (ANN), Support Vector Machines (SVM), Naïve Bayes (NB), k-Nearest Neighbors (KNN), Logistic Regression (LR), and Extra Trees classifiers (ETC) — all of which are prominent within the machine learning domain.

The model development process involved a meticulous examination of various configurations and structures associated with each modelling approach to optimize performance. Hyperparameter tuning, cross-validation, and iterative refinement were integral components of our methodology to ensure the reliability and generalizability of our models.

Our exploration of machine learning models involved a systematic approach to comprehend their nuances and intricacies. The following subsections provide a detailed overview of each model and the experiments conducted:

2.4.1 Artificial Neural Networks (ANN)

The ANN was used to model the complex patterns in the dataset. Several configurations are tested for the performance of models. Configurations include the number of hidden layers varied from one to three, and the number of neurons per layer varies. It considered configurations with a single hidden layer with 100 neurons (ANN_Default), going deeper up to two hidden layers with 100 and 50 neurons (ANN_Large) and even three hidden layers with 100, 50, and 25 neurons (ANN_Deep). In all cases, it used the ReLU activation method, and the maximum number of iterations for stochastic gradient descent during the training was 500. During training, a dropout technique for avoiding overfitting was applied.

2.4.2 Support Vector Machines (SVM)

Various Support Vector Machine (SVM) models were developed and assessed to discern their efficacy in the designated task. Support Vector Machines are adopted with different kernel functions for class separation among the data. The SVM models were systematically configured with diverse parameters, including different values for the regularization parameter (C), kernel types (linear, polynomial, and radial basis function), and class weights. The models, namely SVM_Default, SVM_C1, SVM_C0.1, SVM_KernelPoly, SVM_KernelRBF, and SVM_ClassWeight, were meticulously trained and evaluated on the dataset.

2.4.3 Naïve Bayes

For the Naïve Bayes classifier, we used the Gaussian Naïve Bayes variant, with special attention being paid to the var_smoothing parameter, which controls the variance that is added to each feature to avoid numerical errors due to data with a very low variance resulting in six different configurations and their evaluation. Each configuration, varying in the number of estimators, maximum depth, and minimum samples for splitting and leaf nodes, underwent meticulous training and assessment. The models were systematically configured with various parameters, including the default settings, different levels of variance smoothing, and custom priors. The var_smoothing parameter is tried against several valuesand specifically, 1e-1 (NB_VarSmoothing), 1, and 1e-3. Custom priors are also tried, to bias the predicted class distribution toward certain classes by setting the priors like this: [0:3,0:7] NB_Priors. The models, namely NB_Default, NB_VarSmoothing, NB_LargerVarSmoothing, NB_SmallerVarSmoothing, and NB_Priors, were meticulously trained and assessed on the dataset.

2.4.4 k-Nearest Neighbors (KNN)

K-Nearest Neighbors models were thoroughly investigated across five distinct configurations, varying in the number of neighbors and the metric for distance computation. The specific configurations were k = 5, k = 10, and k = 20, one of which used distance-based weighting. These were selected on the basis of capturing the optimal bias–variance tradeoff. Noteworthy findings included the nuanced performance variations observed with different neighbor counts, emphasizing the sensitivity of the model to this parameter. The KNN_Distance model, leveraging distance-weighted neighbors, emerged as the most promising with higher accuracy, precision, and commendable recall.

2.4.5 Logistic Regression

We used Logistic Regression as our baseline for binary classification tasks. Logistic Regression models were extensively evaluated across diverse configurations, distinguished by penalty type and regularization strength. Tested models were those with different regularization strengths: LogisticReg_C1, that is, when C = 1, and LogisticReg_C0.1, that is, when C = 0.1. We also experimented with the influence of various penalty terms: in LogisticReg_PenaltyL1, we used L1 regularization and penalty='l1'. The LogisticReg_PenaltyL1 model, employing L1 regularization, demonstrated exceptional performance, underscoring its proficiency in correctly identifying positive instances while maintaining high precision.

2.4.6 Extra Trees Classifier (ETC)

Other ways to maximize classification accuracy include the addition of Extra Trees classifiers. The Extra Trees Classifier (ETC) underwent meticulous evaluation using cross-validation and diverse configurations. Those tested include the default model: ExtraTrees_Default; 100 trees: n_estimators=100; labeled ExtraTrees_N100; constrained models, such as at a maximum depth of 10: max_depth=10; labeled ExtraTrees_MaxDepth10. We further parametrized minimum samples needed to split an internal node (min_samples_split=2, ExtraTrees_MinSamplesSplit2) and minimum samples

needed to be at a leaf node (min_samples_leaf=1, ExtraTrees_MinSamplesLeaf1). We other setup consider class weight handling for class imbalance class_weight='balanced', ExtraTrees_ClassWeight. Notable models like ET_Default and ET_ClassWeight demonstrated robust performance, providing valuable insights into the model's efficacy under different settings. The ET_ClassWeight configuration excelled in accuracy, precision, recall, and F1 Score, showcasing the importance of addressing class imbalances.

All models with different configurations were evaluated and analyzed for the most common measures such as accuracy, precision, recall, f1 score, AUC-ROC, FPR, among others, to give a complete measure of model performance

2.5 Model Evaluation Metrics and Comparative Analysis

In our comprehensive evaluation, a set of key performance metrics was employed to quantify the effectiveness of each machine learning model. These metrics include accuracy, precision, recall, F1 Score, and the Area Under the Receiver Operating Characteristic (ROC) Curve (AUC-ROC). These metrics provide a multifaceted understanding of each model's ability to discriminate between classes and generalize to new data.

2.6 Feature Selection

In this study, we employed powerful machine learning algorithm Extra Trees for predicting the target variable in our dataset. The dataset consisted of the target variable representing binary outcomes. To interpret and compare the importance of features in the model, SHAP (SHapley Additive exPlanations) analysis was employed. Specifically, we utilized the SHAP TreeExplainer for the Extra Trees model, allowing us to obtain meaningful insights into the contribution of each feature to the models' predictions. The dataset was split into training and testing sets, and the model was trained on the training set. The feature importance analysis was then conducted on the testing set to assess the global impact of each feature on the model predictions.

Results

3.1 Data Collection and Preprocessing

Two genome-wide studies (Tsai, Kleinstiver) that profiled cleavage sites using CRISPR-Cas9 provided the training dataset. These investigations supplied genomic targets linked to the frequency of cleavage by particular sgRNAs that have been experimentally validated. After data filtration, a total of 16 distinct sgRNAs were constructed, yielding 472 genomic targets. Cleavage efficiencies were standardized using linear regression using the Tsai dataset as a reference to acknowledge the differences in experimental methodology used in different research. The final dataset contained details on cleavage frequencies, on- and off-target sites, and sgRNA sequences.

The Cas-OFFinder webserver was used to create the dataset of negative off-targets (35). Cas-OFFinder showed better coverage of experimentally confirmed off-targets in comparison to other traditional sgRNA design methods. In addition to its accuracy in identifying positive off-targets, Cas-OFFinder was also able to identify genomic regions that could be off-targets but were not found to be cleaved in experiments. As a result, those anticipated by Cas-OFFinder but not included in the list of positive off-targets were included in the study's cases of negative off-targets. Following each Cas-OFFinder run, the negative dataset was produced. To ensure that the positive and negative datasets were equally represented, a predetermined number of negative off-targets were chosen at random from the shuffled list for each target.

Uncleaved and experimentally validated cleavage sites were pooled to create the combined training dataset. The training set included 472 positive off-targets and 472 negative off-targets, all of which were linked to 16 different sgRNAs. This large dataset offers a strong basis for the development and evaluation of machine learning algorithms for CRISPR-Cas9 off-target location prediction.

3.1.1 Calculation of sequence descriptors

Examining several variables that might account for the effects of CRISPR-Cas9 gene editing was part of the analysis. Target site-specific information such as nucleotide composition, GC content, chromatin structure, CpG islands, and gene expression levels were among these factors. Secondary structural features were taken into consideration for sgRNA, whereas alignment features accounted for variables such the quantity and distribution of mismatches (Table 1).

Python programs were developed to extract parameters of various kinds, modulated by the pairwise alignments of sgRNA and genome target locations, so as to make this study easier. Nucleotide composition, GC content, PAM sites, and properties of neighboring genomic areas were among the variables that were retrieved. Machine learning models were constructed using these attributes as explanatory variables, labeling positive off-targets as 1 and negative off-targets as 0. With this method, CRISPR-Cas9 off-target sites can be predicted using a wide range of parameters in an organized and methodical manner.

3.1.2 One Hot Encoding of categorical values

Originally consisting of both label and numeric values, the sequence descriptors were transformed in order to make it easier to incorporate them into the machine learning models. As shown in Figure 2, One-Hot Encoding and Label Encoding techniques were used in this process. This procedure allowed for a standardized representation in the models by converting data variables with label values into binary variables.

This encoding technique allowed all of the sequence descriptors to be broken down into a complete set of 1498 features. Our prediction models were trained using these characteristics, which captured the spirit of the original sequence descriptors. This conversion not only made the data more compatible with machine learning techniques, but it also improved the dataset, enabling our models to identify complex

Table 1. Listing of all features employed in this research.

S. No	Category	Features
1	Sequence Features	'sgRNA sequence', 'chromosome'
2	Alignment Features	'pairwise alignment score'
3	Bulges and Mismatches Features	'total_bulges', '#RNA bulges', '#DNA bulges', '#mismatches', '#linked mismatches and bulges', '#mismatches - positions 1-4', '#mismatches - positions 5-8', '#mismatches - positions 9-12', '#mismatches - positions 13-16', '#mismatches - positions 17-20'
4	Base Composition Features	'#wobble total', '#YY total', '#RR total', '#Tv total', 'GC content - extended'
5	Nucleic Acid Properties Features	'DNA enthalpy', 'DNA enthalpy - extended', 'Adenosine occupancy', 'Cytosine occupancy', 'Thymine occupancy', 'Guanine occupancy'
6	Helical Structure Features	'min MGW', 'average Helical Twist', 'average Roll', 'average Propeller Twist', 'MGW NNGGN', 'Helical Twist NNGGNN', 'Roll NNGGNN', 'Propeller Twist NNGGN'
7	RNA Secondary Structure Features	'sgRNA MFE', 'sgRNA ensemble MFE', 'sgRNA MFE frequency', 'sgRNA ensemble diversity', 'sgRNA secondary structure paired nts', 'tracRNA MFE', 'tracRNA ensemble MFE', 'tracRNA MFE frequency', 'tracRNA ensemble diversity'
8	Genomic Position Features	'distance from centromere', 'distance from telomere', 'DHS signal value', 'distance from DHS', 'nucleosome occupancy', 'distance from nucleosome', 'genomic sub-compartment'
9	Expression Features	'NGG strand expression', 'non-NGG strand expression', 'in exon (NGG strand)', 'in exon (non-NGG strand)', 'transcription region', 'coding region'

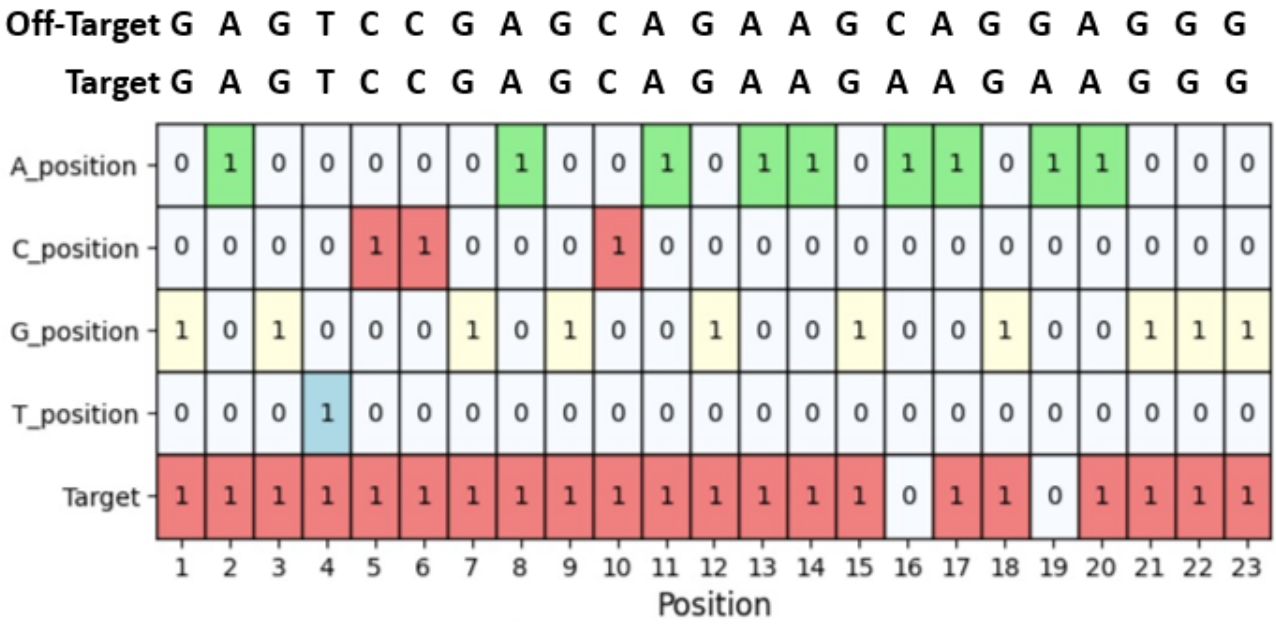

Figure 2. One-hot encoding in CRISPR/Cas9 Model.

patterns and relationships found in the genomic sequences.

3.2 Experimental Setup and Outcome Evaluation

To create reliable models, we used a variety of sophisticated classification modeling approaches in our machine learning study. Artificial Neural Network, Extra Trees, Support Vector Machine, k-Nearest, Naive Bayes, and Logistic Regression are some of the machine learning classifier models. We were able to thoroughly examine and contrast the performance of these models in a number of areas thanks to their varied diversity. The outcomes of several machine learning-based models are shown here, along with comparisons between them in diverse contexts.

3.2.1 Artificial neural network model

Five distinct architectures of artificial neural networks (ANNs) were created and carefully evaluated in order to look at machine learning models for the given challenge. Every model's efficacy was assessed using a wide range of configurations, including default values and various layer sizes, and a comprehensive set of assessment metrics. In instance, the ANN_Default model showed a poor total accuracy but a high recall of 0.971751 and an accuracy of 0.507163, suggesting that it can identify successful occurrences. Conversely, the ANN_Small model's overall performance was lower, emphasizing the influence of design on outcomes. The graphics in Figures 3 provide a quick comparison of accuracy, precision, recall, and F1 score for each model.

3.2.2 Support Vector Machine

We thoroughly examined many configurations of Support Vector Machine (SVM) models, each tailored to address a particular aspect of the classification task. We painstakingly partitioned a dataset and used a comprehensive array of SVM models with various regularization strengths (C), class weights, and kernel functions (poly and radial basis function) to evaluate each model's performance. Notable also is the SVM_C0.1 model's ability to balance recall (66.67%) and accuracy (52.72%), suggesting that it can be useful in situations when sensitivity is the primary concern. The SVM_KernelPoly model, on the other hand, lacked precision but had a high recall of 88.14%. Bar charts are used to provide a consolidated image in Figure 4. These findings help practitioners carefully select SVM configurations based on their particular needs by offering valuable information on the efficacy of SVM models in the context of the topic under study.

3.2.3 Naïve Byaes

An important way to assess the extent to which Naive Bayes (NB) models identify instances is to compare their performance in different setups. With an accuracy of roughly 50.72%, for example, the NB_VarSmoothing model exhibits balanced precision and recall of roughly 50.93% and 77.40%, respectively. Its F1 score of roughly 61.44%, however, indicates a mediocre

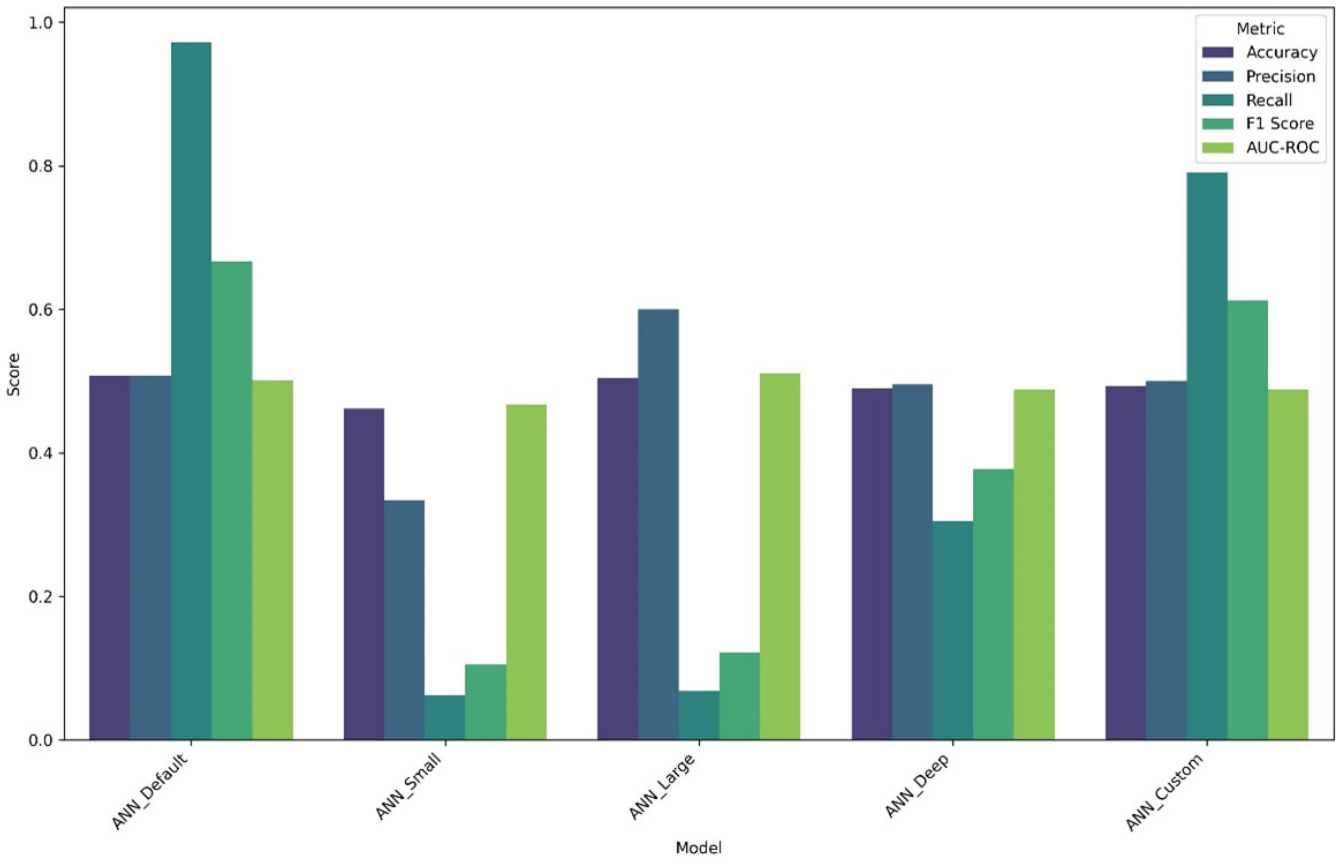

Figure 3. Graphical representation of Five ANN model's performance based on a different statistical measure.

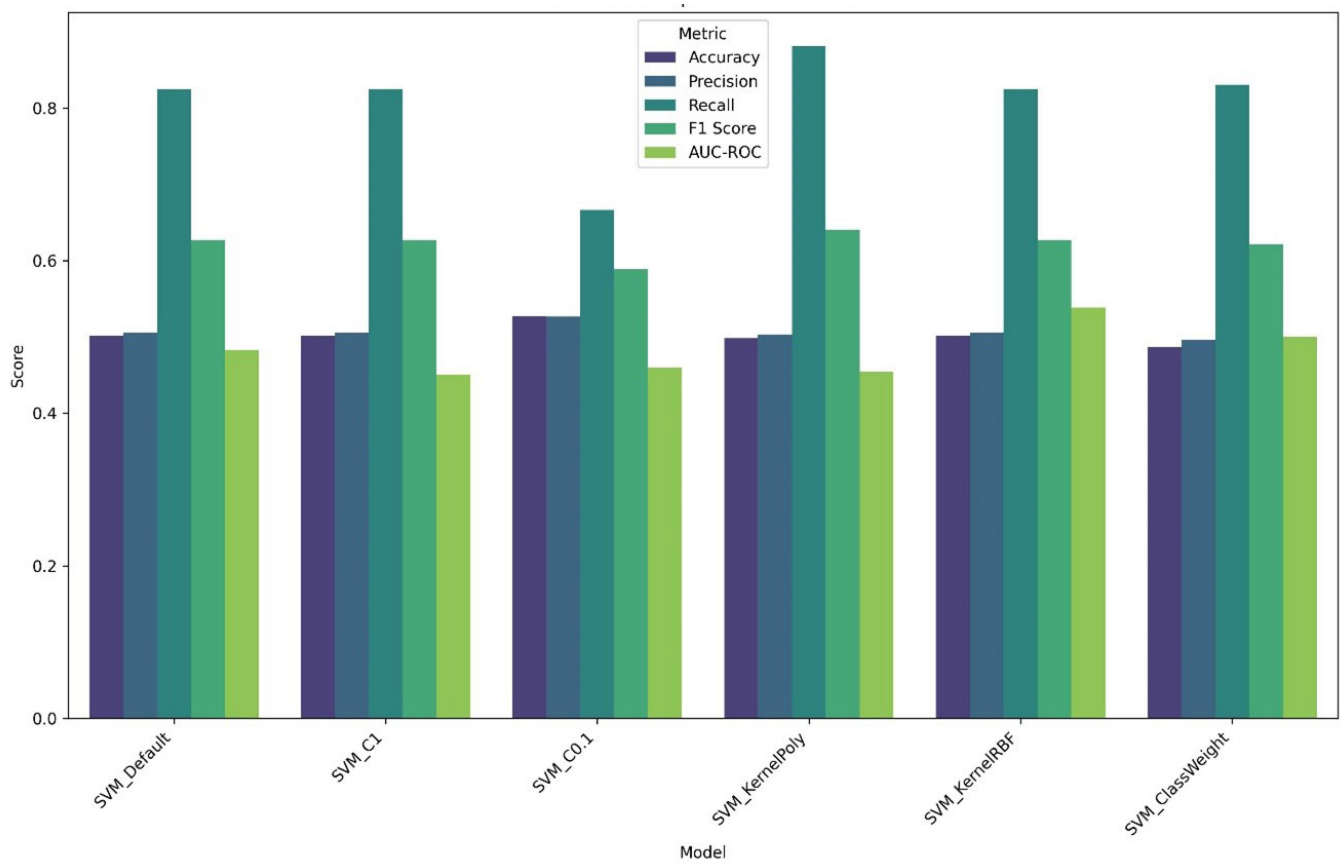

Figure 4. Graphical representation of developed SVM models performance based on evaluation parameters.

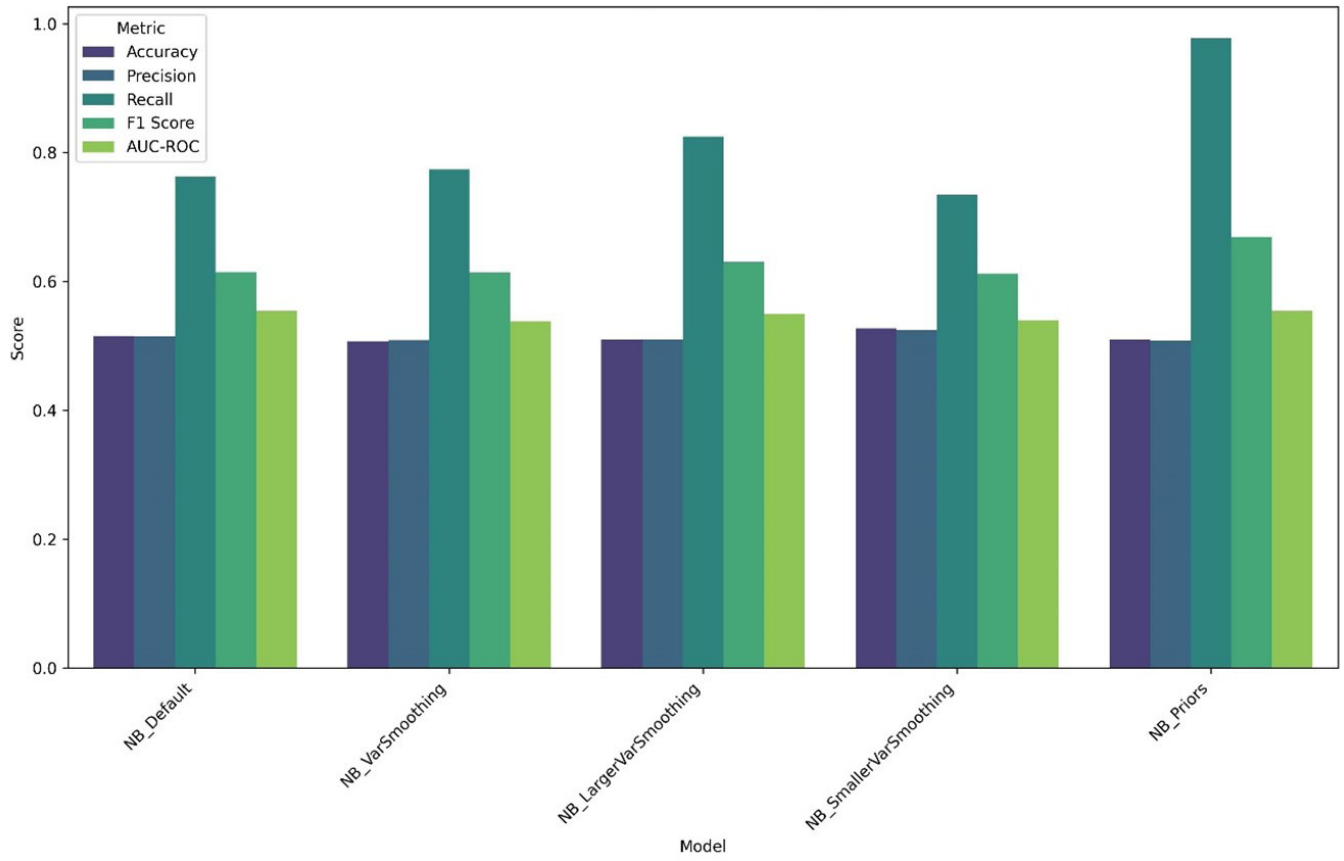

Figure 5. Graphical representation of developed Naïve Bayes model performance based on evaluation parameters.

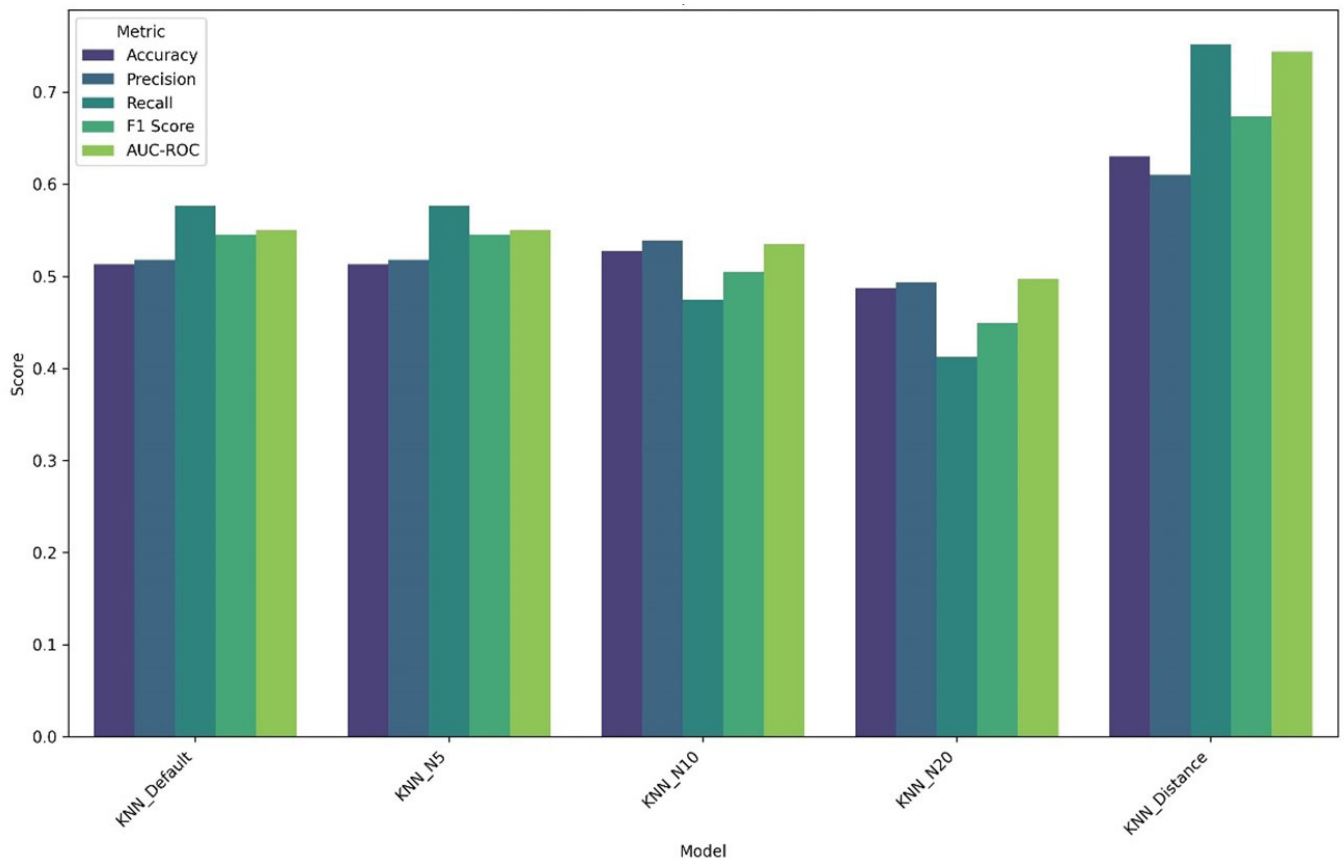

Figure 6. Graphical representation of developed KNN model performance based on evaluation parameters.

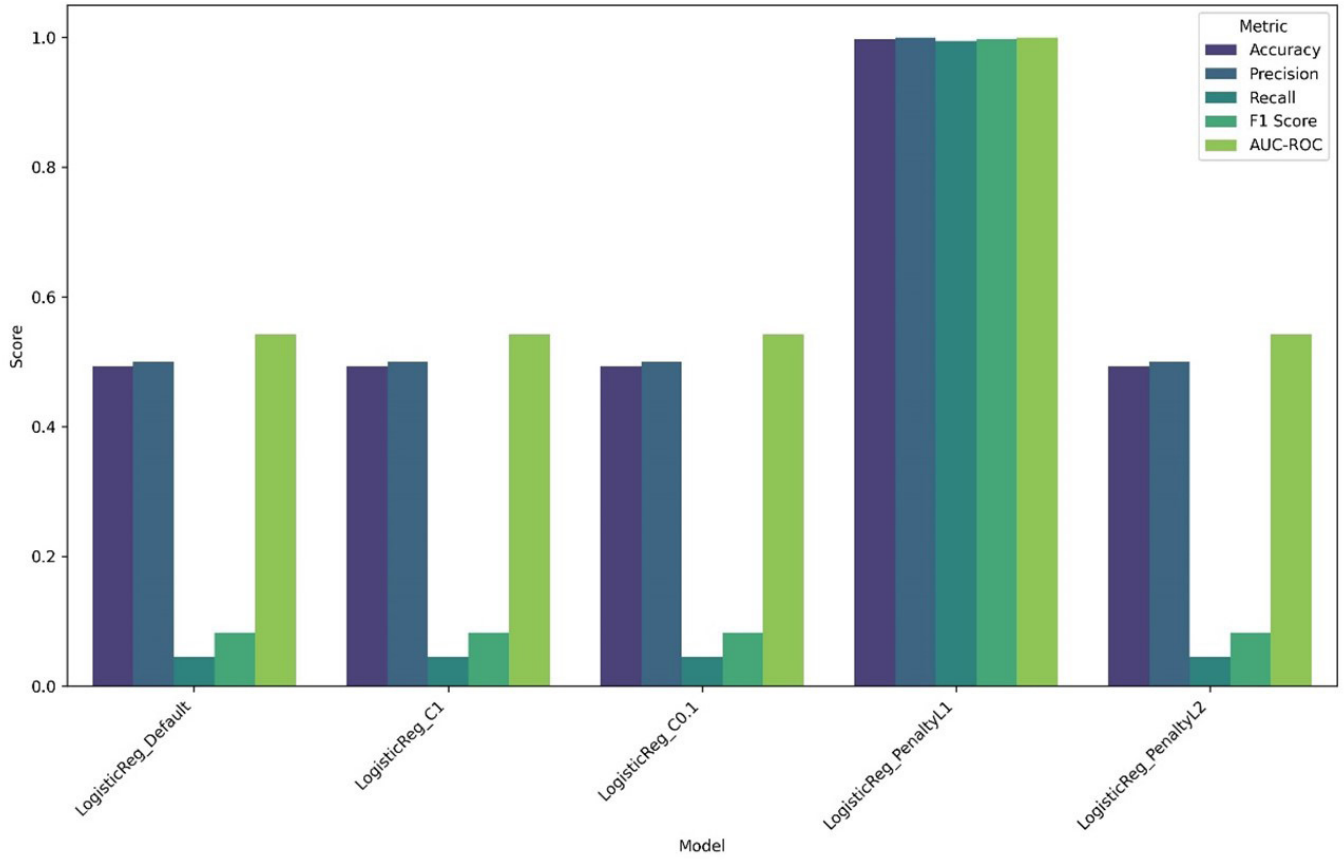

Figure 7. Graphical representation of developed Logistic Regression models performance based on evaluation parameters.

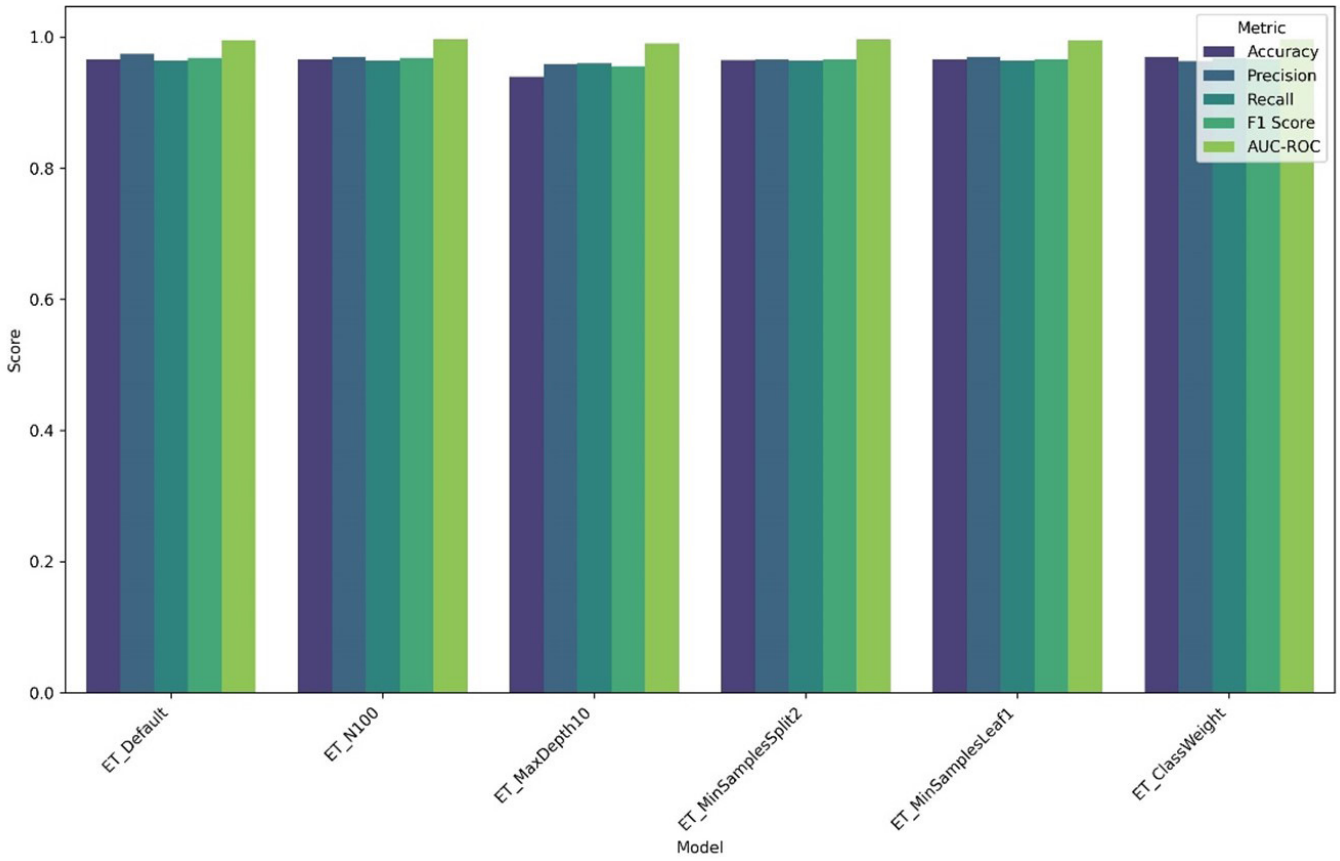

Figure 8. Graphical representation of developed Extra Tree Classifiers models performance based on evaluation parameters.

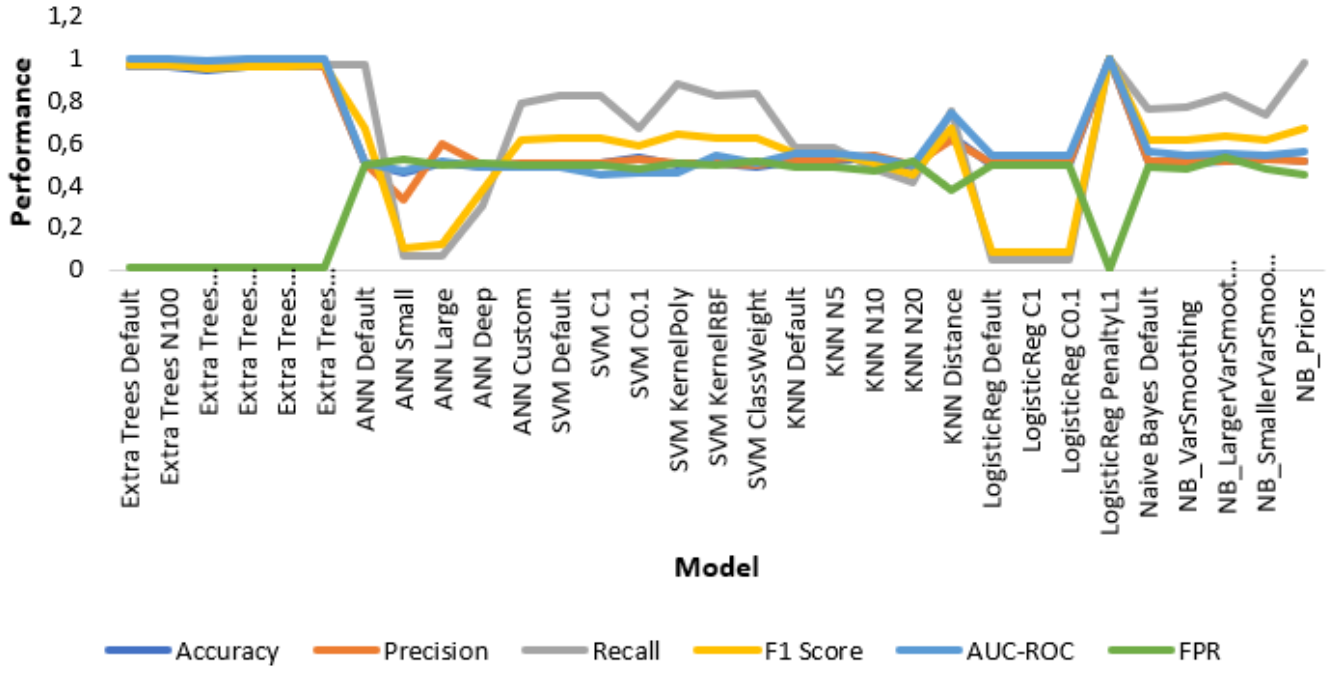

Figure 9. Graphical representation of the performance of all models for off-target prediction developed using three groups of machine learning techniques.

Table 2. Performance evaluation of developed models on dataset based on accuracy, precession, recall, F1 score, AUC-ROC and FPR.

S. No	Model	Accuracy	Precision	Recall	F1 Score	AUC-ROC	FPR
1	Extra Trees Default	0.966	0.974	0.964	0.968	0.996	0.009
2	Extra Trees N100	0.966	0.970	0.964	0.968	0.997	0.007
3	Extra Trees MaxDepth10	0.940	0.958	0.960	0.956	0.990	0.010
4	ExtraTrees MinSamplesSplit2	0.965	0.966	0.964	0.965	0.996	0.008
5	ExtraTrees MinSamplesLeaf1	0.966	0.970	0.964	0.966	0.995	0.005
6	Extra Trees ClassWeight	0.969	0.963	0.968	0.967	0.996	0.011
7	ANN Default	0.507	0.507	0.972	0.667	0.501	0.494
8	ANN Small	0.461	0.333	0.062	0.105	0.467	0.519
9	ANN Large	0.504	0.600	0.068	0.122	0.511	0.496
10	ANN Deep	0.490	0.495	0.305	0.378	0.489	0.503
11	ANN Custom	0.493	0.500	0.791	0.613	0.488	0.498
12	SVM Default	0.501	0.505	0.825	0.627	0.483	0.499
13	SVM C1	0.501	0.505	0.825	0.627	0.451	0.499
14	SVM C0.1	0.527	0.527	0.667	0.589	0.459	0.481
15	SVM KernelPoly	0.499	0.503	0.881	0.641	0.454	0.501
16	SVM KernelRBF	0.501	0.505	0.825	0.627	0.539	0.499
17	SVM ClassWeight	0.487	0.497	0.831	0.622	0.500	0.513
18	KNN Default	0.513	0.518	0.576	0.545	0.550	0.487
19	KNN N5	0.513	0.518	0.576	0.545	0.550	0.487
20	KNN N10	0.527	0.538	0.475	0.505	0.535	0.463
21	KNN N20	0.487	0.493	0.412	0.449	0.497	0.514
22	KNN Distance	0.630	0.610	0.751	0.673	0.744	0.372
23	LogisticReg Default	0.493	0.500	0.045	0.083	0.542	0.496
24	LogisticReg C1	0.493	0.500	0.045	0.083	0.542	0.496
25	LogisticReg C0.1	0.493	0.500	0.045	0.083	0.542	0.496
26	LogisticReg PenaltyL1	0.997	1.000	0.994	0.997	1.000	0.000
27	Naive Bayes Default	0.516	0.515	0.763	0.615	0.554	0.485
28	NB_VarSmoothing	0.507	0.509	0.774	0.614	0.539	0.476
29	NB_LargerVarSmoothing	0.510	0.510	0.825	0.631	0.550	0.534
30	NB_SmallerVarSmoothing	0.527	0.524	0.734	0.612	0.540	0.480
31	NB_Priors	0.510	0.509	0.977	0.669	0.554	0.447

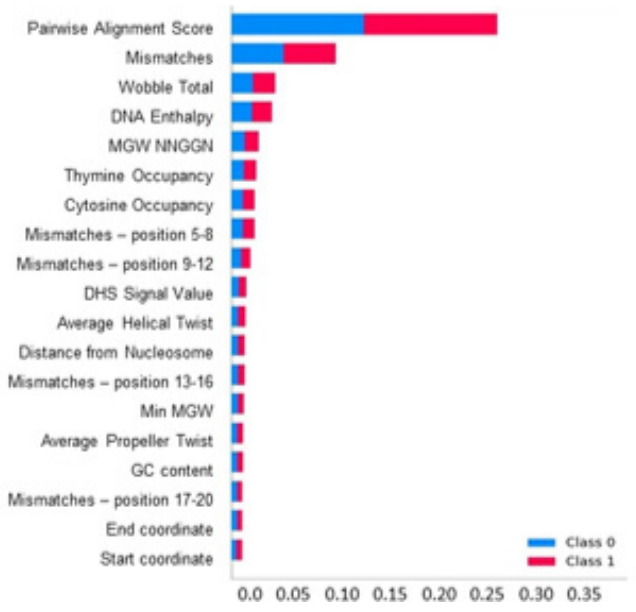

Figure 10. SHAP summary plot shows the impact of the features on each model output, negative class shown in blue and positive class shown in red, as stacked bars, in decreasing order of impact on output

effort in striking a balance between recall and precision. Conversely, the NB_Priors model is notable for having a recall of roughly 97.74%, which shows that it is effective at detecting true positive cases. Even yet, compared to other models, its precision of about 50.88% yields a little lower F1 score (Figure 5).

3.2.4 K-Nearest Neighbors

Five different configurations with different numbers of neighbors and distance metrics were thoroughly investigated in the study of K-Nearest Neighbors (KNN) models. Similar performances were shown by the KNN_Default and KNN_N5 models, which had five neighbors and a default, respectively. Both models produced balanced precision and recall values with an accuracy of roughly 0.512894. When using KNN_N10 to expand the neighborhood to 10 neighbors, accuracy increased somewhat (0.527221) but precision decreased, highlighting the model's sensitivity to neighbor selection. In contrast, the KNN_N20 model, which has 20 neighbors, showed a balanced trade-off between precision and recall while having a lower accuracy (0.487106). Among them, the KNN_Distance model with distance-weighted neighbors proved to be the most promising, with a remarkable recall of 0.751412, a greater accuracy of 0.630372, and a precision of 0.610092. These complex results provide insightful information on how neighbor selection affects overall predicting performance. These important performance measures are graphically summarized in the related Performance Comparison of KNN Models plot, which makes for an easy-to-understand and thorough explanation (Figure 6.). This in-depth analysis provides subtle insights that direct the tactical implementation of KNN models in the particular field of this research.

3.2.5 Logistic Regression

In the analysis of Logistic Regression models, a detailed study was carried out on a variety of configurations that differed in terms of regularization strength and penalty type. The models LogisticReg_Default, LogisticReg_C1, and LogisticReg_C0.1, which correspond to the default configurations and different regularization strengths, demonstrated similar performance metrics, highlighting the restricted influence of regularization strength in this situation. Prominently, the LogisticReg_PenaltyL1 model with L1 regularization performed extraordinarily well, scoring an F1 Score of 0.997167, accuracy of 0.997135, precision of 1.0, and recall of 0.994350.

This highlights how the model can accurately and with high precision identify positive events. On the other hand, using L2 regularization, the LogisticReg_PenaltyL2 model replicated the default setups' performance. The comprehensive results of the Logistic Regression Evaluation offer a nuanced picture of the unique advantages and disadvantages present in each model. These important parameters are graphically illustrated in the Performance Comparison of Logistic Regression Models plot that goes with it, making an in-depth and easy-to-understand interpretation possible (Figure 7.). This thorough investigation provides subtle insights that direct the thoughtful implementation of logistic regression models in the particular study environment.

3.2.6 Extra tree classifier

Throughout this study, cross-validation was used to carefully assess the Extra Trees Classifier (ET), utilizing a variety of configurations to determine how well it performed in estimating off-target sites. The accuracy of the ET_Default configuration was 96.56%, with noteworthy results for recall, precision, and F1 Score of 96.82%, 96.40%, and 97.41%, respectively. More investigation into modifications of hyperparameters like ET_N100 and ET_MaxDepth10 yielded more insights into the behavior of the model. With an accuracy of 96.63%, ET_N100 showed strong performance; however, ET_MaxDepth10, which has a shallower tree structure, nevertheless performed well with an accuracy of 93.98%, highlighting the model's versatility. The ET_MinSamplesSplit2 and ET_MinSamplesLeaf1 settings show minor differences in accuracy and other measures, demonstrating the impact of sample splitting schemes.

The configuration that performed the best, ET_ClassWeight, achieved an accuracy of 96.92% and demonstrated the effectiveness of resolving class imbalances by doing well in

precision, recall, and F1 Score (Figure 8).

3.2.7 Comparative Performance Analysis

The comparative performance analysis of various machine learning models on CRISPR-Cas9 data reveals noteworthy insights (Table 2.) (Figure 9.). Among the Extra Trees Classifier configurations, the default setting, and configurations such as N100 and MinSamplesLeaf1 exhibit high accuracy, precision, recall, F1 score, and AUC-ROC values, underscoring their robust performance. Notably, the Extra Trees ClassWeight configuration stands out with the highest accuracy of 0.969176, emphasizing its efficacy in addressing class imbalances. In contrast, the Artificial Neural Network (ANN) models demonstrate varying performance, with the default configuration displaying limited accuracy, while the custom configuration excels in recall and F1 score. Support Vector Machine (SVM) models, particularly those with C0.1 and KernelPoly, showcase competitive performance in terms of accuracy and recall. However, some SVM configurations, such as Default and C1, exhibit lower AUC-ROC values. The k-Nearest Neighbors (KNN) model with Distance metric stands out with the highest accuracy of 0.630372, indicating its effectiveness. Naive Bayes and Logistic Regression models demonstrate moderate performance, while the Logistic Regression PenaltyL1 configuration achieves near-perfect scores, suggesting its suitability for the given task. Overall, the comparative analysis highlights the varying strengths and weaknesses of different models, providing valuable insights for selecting the most appropriate model for CRISPR-Cas9 off-target prediction.

3.2.8 Feature importance

The feature importance analysis revealed intriguing insights into the significance of various features in predicting the target variable using Extra Trees model. The top 20 features, determined by mean SHAP values, exhibited notable differences in their contributions to the models. "Pairwise Alignment Score" emerged as the most influential feature in both models, constituting 100% of the mean SHAP value in Extra Trees. In contrast, other features, such as "PAM type" and "Helical Twist NNGGNN," demonstrated minimal importance, reflected by a 0% contribution to the mean SHAP value (Figure 10.).

Discussion

In the CRISPR-Cas9 gene editing, the potential occurrence of off-target effects, where sgRNA unintentionally binds to non-targeted DNA locations, poses a significant challenge due to resulting off-target mutations. These mutations have the capacity to disrupt gene functionalities, instigating substantial genomic instability and raising serious concerns about the application of CRISPR-Cas9 techniques. Consequently, achieving accurate predictions of off-target sites related to sgRNA becomes imperative.

Machine learning models have become an intrinsic part of CRISPR-Cas9 genome editing for predicting off-target effects, as they are capable of processing large volumes of genomic data and tracing complex patterns. However, all of these models have their own strengths and limitations, which affect their efficiency in predicting off-targets.

Our study build on this approach employed machine learning to predict off-targets in the CRISPR-Cas9 gene editing process. The comprehensive evaluation of six machine learning models for CRISPR-Cas9 off-target prediction offers valuable insights into their strengths and limitations. Machine learning models, such as Artificial Neural Networks, Support Vector Machines, k-Nearest Neighbors, Naïve Bayes, and Logistic Regression, have been applied to CRISPR-Cas9 off-target prediction. All of them come with their own strengths and weaknesses. ANNs are a very powerful tool for modeling non-linear relationships, although they tend to overfit smaller numbers of training datasets and require extensive tuning. SVMs work well in high-dimensional spaces but are computationally intensive and much less effective with imbalanced data. K-NN is very simple and intuitive, but the model suffers mainly when it comes to working on high-dimensional settings and with imbalanced classes, where it might reduce the predictive accuracy. Though naïve Bayes is quite computationally efficient, feature independence is assumed; therefore, it does pretty much remain within the bounds of moderate-performance contexts. While Logistic Regression is interpretable, it will not learn the non-linear interactions necessary for accurate prediction. In contrast, ETC performed the best in managing complex feature interactions, reducing noise through ensemble learning, and dealing with large, unbalanced datasets with minimal tuning. The ETC model with class weights outperformed the traditional models regarding accuracy, precision, recall, and F1 score and was, therefore, rated the most effective in this study for the CRISPR–Cas9 off-target prediction. The ensuing discussion delves into a comparative performance analysis of various machine learning models applied to CRISPR-Cas9 data for off-target prediction.

The results underscore the critical importance of thoughtful model selection, revealing significant variability in performance among the examined models. The Extra Trees Classifier exhibited excellent performance in different settings in this study and proved to be the most effective model for predictions of off-target sites. The default ETC model demonstrated an accuracy of 96.56%, a precision of 97.41%, a recall of 96.40%, and an F1 equal to 96.82%. In total, such metrics mean that the model is balancing very well to identify correctly both positive and negative cases and is thus very reliable. Further study of the ETC configurations gave more insight into the behavior of that model. ETC_N100, raising the number of trees in the ensemble to 100, slightly improved the obtained accuracy, at about 96.63%, thus proving the idea that increasing the number of the estimator can improve the strength of the model. Configuration ETC_MaxDepth10 limited the maximum depth of trees to 10 and resulted in reduced accuracy to 93.98%. The model retained quite strong precision and recall, showing that

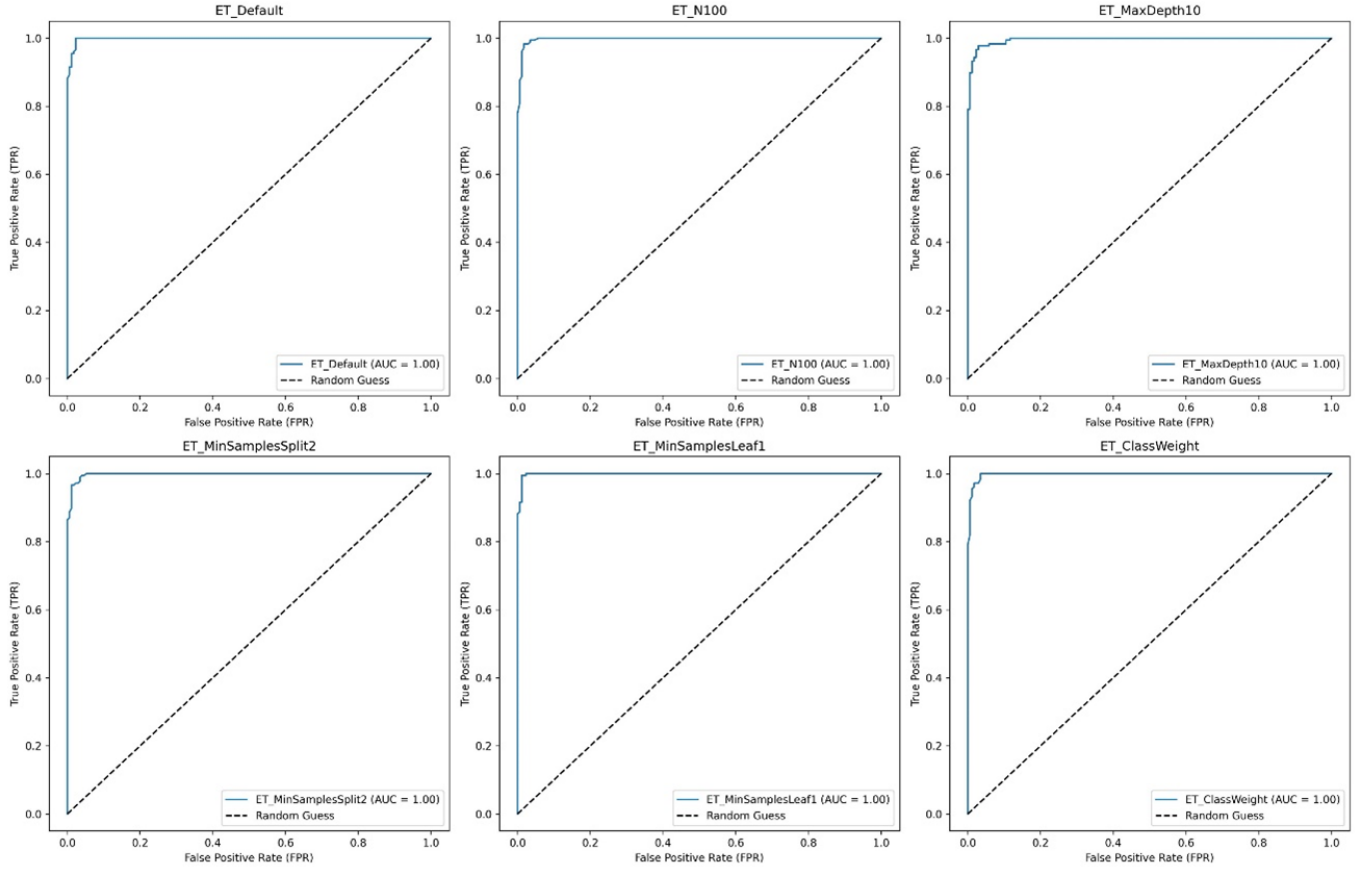

Figure 11. ROC curve of ETC model performance based on AUC score.

even when not deep, trees in the ETC will still remain with efficiency, albeit some performance tradeoffs. Configurations like ETC_MinSamplesSplit2 and ETC_MinSamplesLeaf1, in contrast, showed much less variation in accuracy and all other measures; in fact, only ETCmMin-LeafSize1, with accuracy 96.63% and considerably lower FPR of 0.005, showed a significantly lower FPR. These observations give further evidence that performance of this model is quite robust to changes in the criteria used to split the samples, only minor effects being noticed. The best performance of 96.92% accuracy was obtained with the ETC_ClassWeight setup. This setup balanced the class weights because of class imbalance and achieved an optimum balance between precision, 96.83%, and recall, 96.34%, with a resulting F1 score of 96.68%. This underlines the importance of class balancing in boosting the power of this classifier.

On the other hand, the performance of the Artificial Neural Network (ANN) models varies. The default setup demonstrates limitations in accuracy, but the custom configuration performs exceptionally well in recall and F1 score. This highlights how sensitive ANN models are to configurations, highlighting the necessity of meticulous optimization. In terms of accuracy and recall, Support Vector Machine (SVM) models perform competitively, particularly those with C0.1 and KernelPoly. Certain SVM setups, however, provide lower AUC-ROC values, indicating possible trade-offs between various performance indicators. As it concerns CRISPR-Cas9 off-target prediction, the k-Nearest Neighbors (KNN) model using the Distance measure stands out with the highest accuracy, demonstrating its effectiveness. While the Logistic Regression PenaltyL1 configuration earns nearly perfect scores, indicating its possible fit for the particular task at hand, the Naive Bayes and Logistic Regression models perform moderately overall.

The practical use of CRISPR-Cas9 in genome editing is directly affected by these findings. Precise off-target prediction is essential for reducing accidental genetic alterations, and the recognized models—especially the Extra Trees Classifier and KNN with the Distance metric—help researchers choose the

best instrument depending on their own requirements and preferences. The sensitivity of different models to these factors is used in the discussion to highlight the significance of setup optimization and hyperparameter adjustment. All things considered, this thorough analysis adds insightful knowledge to the field of CRISPR-Cas9 off-target prediction, assisting researchers in choosing the right model and configuring it to produce the best results possible in genome editing applications.

Conclusions

The paper presents a comprehensive analysis of various machine learning models designed to predict the off-targets of the CRISPR-Cas9, focusing on the performance of the Extra Trees Classifier. The results confirm that ETC is among the most efficient models that have great performance under various settings, hence making it a preferred choice for genome editing researchers. It also provided high accuracy, precision, recall, and the F1 score for different configurations of ETC, proving it to be versatile and reliable. The accuracy from the default configuration was already very high, but further exploration showed that some hyperparameters—for example, the number of estimators, maximum tree depth, and class weights—can bring marginal improvements in model performance. Of these, ETC_ClassWeight turned out to be the best for providing the highest accuracy and showing the critical role dealing with class imbalance has for model effectiveness. The SHAP value-based importance analysis did improve our understanding of the predictive capabilities of the ETC even further. In this study, it is realized that "Pairwise Alignment Score" is the most influencing feature, contributing much towards the predictions by the model, while others, such as "PAM type" and "Helical Twist NNGGNN", had very minimal impacts. This result shows that proper feature selection is very critical in improving the accuracy and reliability of machine learning models in genome editing.

In conclusion, Extra Trees Classifier represents a powerful and multi-affinity tool to predict off-targets in the highly specific genome editing technique of CRISPR-Cas9. Its good performance in all different configurations makes it a trustworthy way for researchers who need to find a good trade-off among accuracy, precision, and computational efficiency. This study therefore not only points out the potential but also provides a framework for further exploration and optimization of machine learning models in genome editing. The model improvement, ensemble techniques, and application to real-world CRISPR experiments should be focused by further studies for further improvement in precision and safety in genome editing technologies.

Future research directions could focus on further refining machine learning models, exploring ensemble methods, and integrating additional features for improved predictive accuracy. Additionally, the application of these models to real-world CRISPR-Cas9 experiments and validation on diverse genomes could enhance their practical utility.

Acknowledgements

Figure 1 created with Biorender.com.

List of Abbreviations

CRISPR – *Clustered Regularly Interspaced Short Palindromic Repeats*
TALEN - Transcription Activator-Like Effector Nuclease
ZFN - Zinc-Finger Nuclease
GUIDE-Seq - Genome-wide, Unbiased Identification of DSBs Enabled by Sequencing
ANN - Artificial Neural Network
ET - Extra Trees
SVM - Support Vector Machine
KNN - k-Nearest
NB - Naive Bayes
LR - Logistic Regression

Conflict of Interest

The authors declare that they have no competing interests.

Authors' contributions

AB has performed the analysis and written the manuscript under the guidance and supervision of PT and VN. All authors read and approved the final manuscript.

References

1. Urnov FD, Rebar EJ, Holmes MC, Zhang HS, Gregory PD. Genome editing with engineered zinc finger nucleases. Nature Reviews Genetics 2010 11:9 [Internet]. 2010 Sep [cited 2024 May 15];11(9):636–46. Available from: https://www.nature.com/articles/nrg2842
2. Perez-Pinera P, Kocak DD, Vockley CM, Adler AF, Kabadi AM, Polstein LR, et al. RNA-guided gene activation by CRISPR-Cas9-based transcription factors. Nat Methods [Internet]. 2013 Oct [cited 2024 May 15];10(10):973–6. Available from: https://pubmed.ncbi.nlm.nih.gov/23892895/
3. Ma H, Marti-Gutierrez N, Park SW, Wu J, Lee Y, Suzuki K, et al. Correction of a pathogenic gene mutation in human embryos. Nature 2017 548:7668 [Internet]. 2017 Aug 2 [cited 2024 May 15];548(7668):413–9. Available from: https://www.nature.com/articles/nature23305
4. Sander JD, Joung JK. CRISPR-Cas systems for editing, regulating and targeting genomes. Nat Biotechnol [Internet]. 2014 [cited 2024 May 15];32(4):347–50. Available from: https://pubmed.ncbi.nlm.nih.gov/24584096/
5. Musunuru K. The Hope and Hype of CRISPR-Cas9 Genome Editing: A Review. JAMA Cardiol [Internet]. 2017 Aug 1 [cited 2024 May 15];2(8):914–9. Available from: https://pubmed.ncbi.nlm.nih.gov/28614576/

6. Terns MP, Terns RM. CRISPR-based adaptive immune systems. Curr Opin Microbiol [Internet]. 2011 Jun [cited 2024 May 15];14(3):321–7. Available from: https://pubmed.ncbi.nlm.nih.gov/21531607/
7. Bortesi L, Fischer R. The CRISPR/Cas9 system for plant genome editing and beyond. Biotechnol Adv [Internet]. 2015 Jan 1 [cited 2024 May 15];33(1):41–52. Available from: https://pubmed.ncbi.nlm.nih.gov/25536441/
8. Westra ER, Buckling A, Fineran PC. CRISPR-Cas systems: beyond adaptive immunity. Nat Rev Microbiol [Internet]. 2014 [cited 2024 May 15];12(5):317–26. Available from: https://pubmed.ncbi.nlm.nih.gov/24704746/
9. Georges F, Ray H. Genome editing of crops: A renewed opportunity for food security. GM Crops Food [Internet]. 2017 Jan 2 [cited 2024 May 15];8(1):1–12. Available from: https://pubmed.ncbi.nlm.nih.gov/28075688/
10. Ebrahimi S, Khosravi MA, Raz A, Karimipoor M, Parvizi* P. CRISPR-Cas Technology as a Revolutionary Genome Editing tool: Mechanisms and Biomedical Applications. Iran Biomed J [Internet]. 2023 Sep 1 [cited 2024 Aug 23];27(5):219. Available from: /pmc/articles/PMC10707817/
11. Ferreira P, Choupina AB. CRISPR/Cas9 a simple, inexpensive and effective technique for gene editing. Mol Biol Rep [Internet]. 2022 Jul 1 [cited 2024 May 15];49(7):7079. Available from: /pmc/articles/PMC9206401/
12. Haque E, Taniguchi H, Hassan MM, Bhowmik P, Karim MR, Śmiech M, et al. Application of CRISPR/Cas9 Genome Editing Technology for the Improvement of Crops Cultivated in Tropical Climates: Recent Progress, Prospects, and Challenges. Front Plant Sci [Internet]. 2018 May 8 [cited 2024 May 15];9. Available from: https://pubmed.ncbi.nlm.nih.gov/29868073/
13. Bortesi L, Zhu C, Zischewski J, Perez L, Bassié L, Nadi R, et al. Patterns of CRISPR/Cas9 activity in plants, animals and microbes. Plant Biotechnol J [Internet]. 2016 Dec 1 [cited 2024 May 15];14(12):2203. Available from: /pmc/articles/PMC5103219/
14. Cho SW, Kim S, Kim Y, Kweon J, Kim HS, Bae S, et al. Analysis of off-target effects of CRISPR/Cas-derived RNA-guided endonucleases and nickases. Genome Res [Internet]. 2014 Jan [cited 2024 May 15];24(1):132. Available from: /pmc/articles/PMC3875854/
15. Xu X, Duan D, Chen SJ. CRISPR-Cas9 cleavage efficiency correlates strongly with target-sgRNA folding stability: from physical mechanism to off-target assessment. Scientific Reports 2017 7:1 [Internet]. 2017 Mar 10 [cited 2024 May 15];7(1):1–9. Available from: https://www.nature.com/articles/s41598-017-00180-1
16. Haeussler M, Schönig K, Eckert H, Eschstruth A, Mianné J, Renaud JB, et al. Evaluation of off-target and on-target scoring algorithms and integration into the guide RNA selection tool CRISPOR. Genome Biol [Internet]. 2016 Jul 5 [cited 2024 May 15];17(1):1–12. Available from: https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1012-2
17. Sanjana NE, Shalem O, Zhang F. Improved vectors and genome-wide libraries for CRISPR screening. Nat Methods [Internet]. 2014 [cited 2024 May 15];11(8):783. Available from: /pmc/articles/PMC4486245/
18. Lin J, Wong KC. Off-target predictions in CRISPR-Cas9 gene editing using deep learning. Bioinformatics [Internet]. 2018 Sep 1 [cited 2024 May 15];34(17):i656–63. Available from: https://pubmed.ncbi.nlm.nih.gov/30423072/
19. Abadi S, Yan WX, Amar D, Mayrose I. A machine learning approach for predicting CRISPR-Cas9 cleavage efficiencies and patterns underlying its mechanism of action. PLoS Comput Biol [Internet]. 2017 Oct 1 [cited 2024 May 15];13(10):e1005807. Available from: https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005807
20. Bhardwaj A, Tomar P, Nain V. Identification and evaluation of machine learning classification algorithm to predict the efficacy of gRNA in CRISPR/Cas9 genome editing system using WEKA. AIP Conf Proc [Internet]. 2023 Dec 22 [cited 2024 May 15];2938(1). Available from: https://www.researchgate.net/publication/376775027_Identification_and_evaluation_of_machine_learning_classification_algorithm_to_predict_the_efficacy_of_gRNA_in_CRISPRCas9_genome_editing_system_using_WEKA
21. Niu M, Lin Y, Zou Q. sgRNACNN: identifying sgRNA on-target activity in four crops using ensembles of convolutional neural networks. Plant Mol Biol [Internet]. 2021 Mar 1 [cited 2024 May 15];105(4–5):483–95. Available from: https://link.springer.com/article/10.1007/s11103-020-01102-y
22. Hesami M, Yoosefzadeh Najafabadi M, Adamek K, Torkamaneh D, Jones AMP. Synergizing Off-Target Predictions for In Silico Insights of CENH3 Knockout in Cannabis through CRISPR/Cas. Molecules [Internet]. 2021 [cited 2024 May 15];26(7). Available from: https://pubmed.ncbi.nlm.nih.gov/33916717/
23. Sherkatghanad Z, Abdar M, Charlier J, Makarenkov V. Using traditional machine learning and deep learning methods for on- and off-target prediction in CRISPR/Cas9: a review. Brief Bioinform [Internet]. 2023 May 1 [cited 2024 Aug 23];24(3). Available from: /pmc/articles/PMC10199778/
24. Muhammad Rafid AH, Toufikuzzaman M, Rahman MS, Rahman MS. CRISPRpred(SEQ): a sequence-based method for sgRNA on target activity prediction using traditional machine learning. BMC Bioinformatics [Internet]. 2020 Jun 1 [cited 2024 Aug 23];21(1). Available from: https://pubmed.ncbi.nlm.nih.gov/32487025/
25. Khan AA, Chaudhari O, Chandra R. A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation. Expert Syst Appl. 2024 Jun 15;244:122778.
26. Chen Y, Wang X. Evaluation of efficiency prediction algorithms and development of ensemble model for CRISPR/

Cas9 gRNA selection. Bioinformatics [Internet]. 2022 Nov 30 [cited 2024 Aug 23];38(23):5175–81. Available from: https://dx.doi.org/10.1093/bioinformatics/btac681
27. Yaish O, Orenstein Y. Generating, modeling and evaluating a large-scale set of CRISPR/Cas9 off-target sites with bulges. Nucleic Acids Res [Internet]. 2024 Jul 7 [cited 2024 Aug 23];52(12):6777. Available from: /pmc/articles/PMC11229338/
28. Pedregosa FABIANPEDREGOSA F, Michel V, Grisel OLIVIERGRISEL O, Blondel M, Prettenhofer P, Weiss R, et al. Scikit-learn: Machine Learning in Python Gaël Varoquaux Bertrand Thirion Vincent Dubourg Alexandre Passos PEDREGOSA, VAROQUAUX, GRAMFORT ET AL. Matthieu Perrot. Journal of Machine Learning Research [Internet]. 2011 [cited 2024 Aug 23];12:2825–30. Available from: http://scikit-learn.sourceforge.net.
29. Mitrofanov A, Alkhnbashi OS, Shmakov SA, Makarova KS, Koonin E V., Backofen R. CRISPRidentify: identification of CRISPR arrays using machine learning approach. Nucleic Acids Res [Internet]. 2021 Feb 26 [cited 2024 Aug 23];49(4):e20–e20. Available from: https://dx.doi.org/10.1093/nar/gkaa1158
30. Lundberg SM, Allen PG, Lee SI. A Unified Approach to Interpreting Model Predictions. Adv Neural Inf Process Syst [Internet]. 2017 [cited 2024 Aug 23];30. Available from: https://github.com/slundberg/shap
31. Kleinstiver BP, Pattanayak V, Prew MS, Tsai SQ, Nguyen NT, Zheng Z, et al. High-fidelity CRISPR-Cas9 nucleases with no detectable genome-wide off-target effects. Nature [Internet]. 2016 Jan 28 [cited 2024 May 15];529(7587):490–5. Available from: https://pubmed.ncbi.nlm.nih.gov/26735016/
32. Tsai SQ, Zheng Z, Nguyen NT, Liebers M, Topkar V V., Thapar V, et al. GUIDE-seq enables genome-wide profiling of off-target cleavage by CRISPR-Cas nucleases. Nat Biotechnol [Internet]. 2015 [cited 2024 May 15];33(2):187–98. Available from: https://pubmed.ncbi.nlm.nih.gov/25513782/
33. Zhang S, Zhang C, Yang Q. Data preparation for data mining. Applied Artificial Intelligence. 2003 May 1;17(5–6):375–81.
34. Pedregosa FABIANPEDREGOSA F, Michel V, Grisel OLIVIERGRISEL O, Blondel M, Prettenhofer P, Weiss R, et al. Scikit-learn: Machine learning in Python. jmlr.orgF Pedregosa, G Varoquaux, A Gramfort, V Michel, B Thirion, O Grisel, M Blondelthe Journal of machine Learning research, 2011•jmlr.org [Internet]. 2011 [cited 2024 May 15];12:2825–30. Available from: https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf?ref=https:/
35. Bae S, Park J, Kim JS. Cas-OFFinder: a fast and versatile algorithm that searches for potential off-target sites of Cas9 RNA-guided endonucleases. Bioinformatics [Internet]. 2014 May 15 [cited 2024 May 15];30(10):1473–5. Available from: https://pubmed.ncbi.nlm.nih.gov/24463181/