CRISPR guide RNA design by deep learning

A graduation project for the students:

Rawan Sayed
Sara Adel
Reham Abdelfatah
Ahmed Elnouby

Supervised by Dr. Ibrahim Youssef

Industrial Partner: Proteinea

Contents

Table of Figures

1. Introduction

a. Biological Background

Every cell in our body has a copy of our genome, which consists of more than 20,000 genes and 3 billion letters of deoxyribonucleic acid (DNA). DNA is consisting of two strands twisted together into a double helix and held together by a basic pairing rule in which Adenine (A) bases on one strand form a double hydrogen bond with Thymine (T) bases on the opposite strand, and Cytosine (C) bases on one strand form a triple hydrogen bond with Guanine (G) bases on the opposite strand.

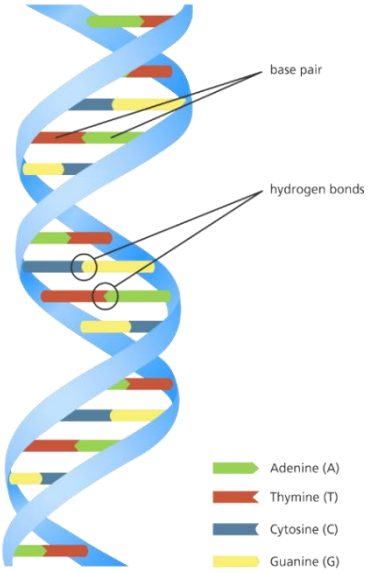

Figure 1: DNA structure

Our genes not only shape who we are as individuals and as a species, but they also have a significant impact on our health, and thanks to advances in DNA sequencing, researchers have discovered thousands of genes that influence our disease risk, but they still need ways to manipulate genes to understand how they function.

Clustered Regularly Interspaced Short Palindromic Repeats (CRISPR) is a ground-breaking genetic editing system that has been designed recently to enhance our ability to alter the genome of any living organism, although gene editing in living cells is difficult.

The CRISPR approach is based on a natural mechanism that bacteria use to defend themselves against virus infection. When bacteria detect virus 'viral' DNA, it generates two forms of short ribonucleic acid (RNA) and these two RNAs form a complex with a protein called CAS9.

Guide RNA (gRNA) is one of these two RNAs and it consists of two components:

1. The trans-activating RNA (tracrRNA) which is responsible for Cas9 endonuclease activity
2. The CRISPR RNA (crRNA) binds to the target-specific DNA region

then they joined with each other forming Single Guided Ribonucleic Acid (sgRNA)

sgRNA has been designed to locate and bind to a specific sequence in the DNA so it has a sequence (~20 nucleotides) that matches the invading virus to make sure that it will only bind to the target sequence and no other unwanted regions.

CAS9 is a nuclease which is a type of enzyme that follows the sgRNA to the desired location then it untwists the DNA and compares it to its target RNA; if the match is successful, the cas9 will cut the DNA with two tiny molecular scissors. When this occurs, the cell attempts to repair the cut, but the repair process is error-prone, leading to mutations that can disable the gene allowing the researcher to understand its function.

Researchers have been studying the CRISPR approach for a few years and have discovered that by modifying the sgRNA to fit the target sequence, it can be designed to cut any DNA sequence at a specific location not only viral DNA. This can be achieved either in a test tube or within a living cell's nucleus. Once the CRISPR be inside the nucleus, it will lock onto a short sequence called protospacer adjacent motif (PAM).

The PAM is a short DNA sequence (2-6 base pairs) that follows the DNA region targeted for cleavage by the CRISPR system and it is necessary for a Cas9 to make its cut and is generally found 3-4 nucleotides downstream from the cut site.

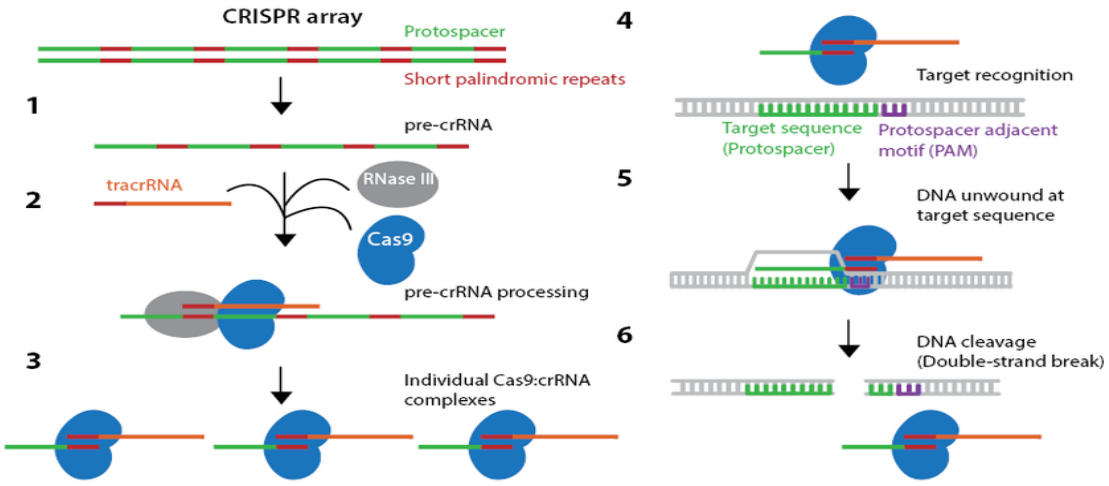

Figure 2: How CRISPR works

Although these mutations are random, the researcher can need to be more precise at times. This can be achieved by replacing a mutated gene with a healthy copy that can be done by inserting another piece of DNA with the desired sequence. This DNA template will pair up with the cut ends after the CRISPR method has made a cut, recombining and replacing the original sequence with the new version.

All of this can be achieved in cultured cells, like stem cells that can give rise to a variety of cell types, or in a fertilized egg, allowing for the development of transgenic animals with targeted mutations. Unlike previous methods, CRISPR can target several genes at once by modifying the crRNA sequences of the gRNA to suit the target location.

These methods are being improved rapidly and will have many applications in basic research, in drug development, agriculture, and, possibly, in the treatment of human patients with genetic diseases.

b. Used Approach

DeepCRISPR is our used approach that uses a well-built hybrid deep neural network for model training and prediction of on-target and off-target sites. Using a comprehensive set of genome-wide unlabelled sgRNAs, we used a deep unsupervised representation learning technique to automatically learn the underlying representation of sgRNAs.

2. Material and Methods

a. Data

CRISPR-based gene knockout is widely implemented in various cell types and organisms. CAS9 follows the sgRNA to the desired location then it untwists the DNA and compares it to its target RNA; if the match is successful, i.e., the on-target, cleavage occurs upstream of a PAM however sometimes the cleavage occurs despite the mis-match i.e., off the target.

We used a public dataset that contains on-target dataset and off-target dataset. We work on on-target dataset.

i. On-target data sources

Our on-target dataset includes ~15,000 sgRNAs containing 1071 genes from four different cell lines with redundancy removed and validated known knockout efficacy.

The four different cell lines are **HCT116** that isolated from male with colonic carcinoma, **HEK293T** that taken from kidney an aborted female fetus, **HELA** that derived from cervical cancer cell, and finally **HL60** that taken from human with leukaemia.

ii. Epigenetic features

DeepCRISPR can be easily extended to other cell types if it similar to our used epigenetic features so we included it in our encoding schema.

These epigenetic features included **CTCF** binding information from the ChIP-Seq assay, **H3K4me3** position information from the ChIP-Seq assay, chromatin-opening information from the **DNase**-Seq assay, and DNA methylation information from the **RRBS** assay, obtained from ENCODE.

b. Data Pre-processing

i. sgRNA encoding with genome and epigenetic features

The nucleotide sequence is represented by four channels in a DNA region, i.e., the A-channel, C-channel, G-channel, and T-channel, and each epigenetic feature is considered as one channel. As a result, each DNA region has an eight-channel representation.

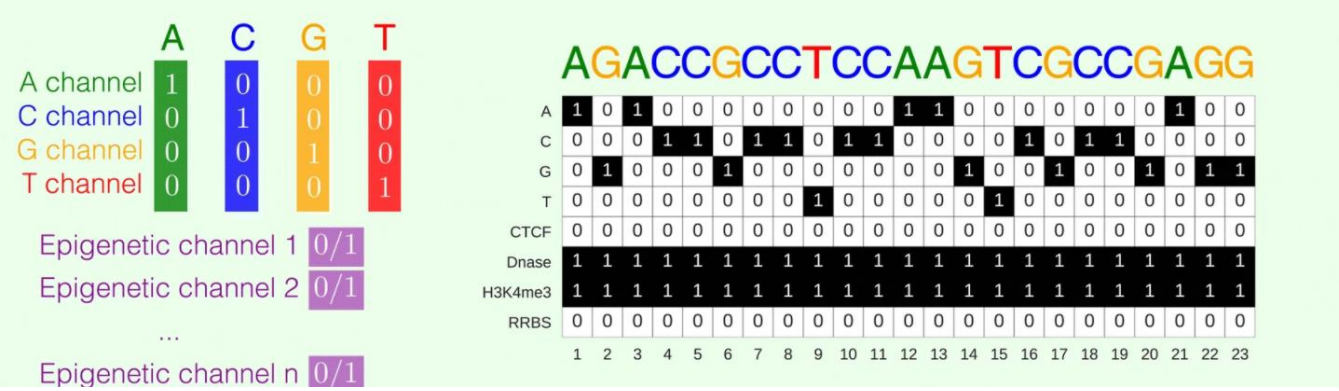

Figure 3 : sgRNA encoding schema

ii. Autoencoder Data

We concatenated the available classification and regression data for the four cell lines to provide ~70,000 data samples to train our autoencoder because the downloaded public data wasn't complete as there weren't 0.68 billion data samples for training autoencoder as mentioned in the paper we worked on.

iii. Classification and Regression Model Data

Previously, sgRNA efficacy was converted to a binary value using a log-fold change of 1 as the cut-off for classification model and collaborative filtering-based data normalization method was adopted for regression model. Table 1. This Table provide both models data sources

Table 1 : sgRNA data for Regression and Classification Model

Cell line	Classification Data	Regression Data
HCT116	~ 4,200	~ 4,200
HEK293T	~ 2,300	~ 4,500
HELA	~ 8,000	~ 8,000
HL60	~ 2,000	~ 2,000
Total	~ 16,700	~ 18,900

iv. Data Augmentation

To enhance our model performance, we needed an additional data which wasn`t available so we did data augmentation by

1. Data Augmented

   We concatenate the data of the four cell-lines to be in one file only.

2. Data Inverse

   We inversed the data augmented by swapping the start and end of our DNA sequences and epigenetic features separately for the four cell lines to gain more data without losing the meaning of the original dataset so it acts as a new dataset.

3. Data Augmented

   We concatenate the data inversed `new dataset` to data augmented in the first step so we get a new file contains ~35,000 data for each model.

c. Deep Learning Model

The Model we are trying to reimplement are the on-target classifier and the on target regressor and the auto encoder which will be used to fine-tune the previous models

i. Auto encoder Model

This is an unsupervised deep convolutionary denoising neural network (DCDNN)-based autoencoder.It consist of 2 groups of layers the encoder group and the decoder group. The encoder group consists of 5 layer of Convolution 2d and batch normalization and denoising layer. The decoder Group in the other hand have a nearly same structure but with the expectation of using deconvolution instead of a convolution.

This model is the model that requires the most data. This model trains on the whole augmented dataset to optimize the encoder as much as we could So it can improve the other models when we load its encoder layers into them.

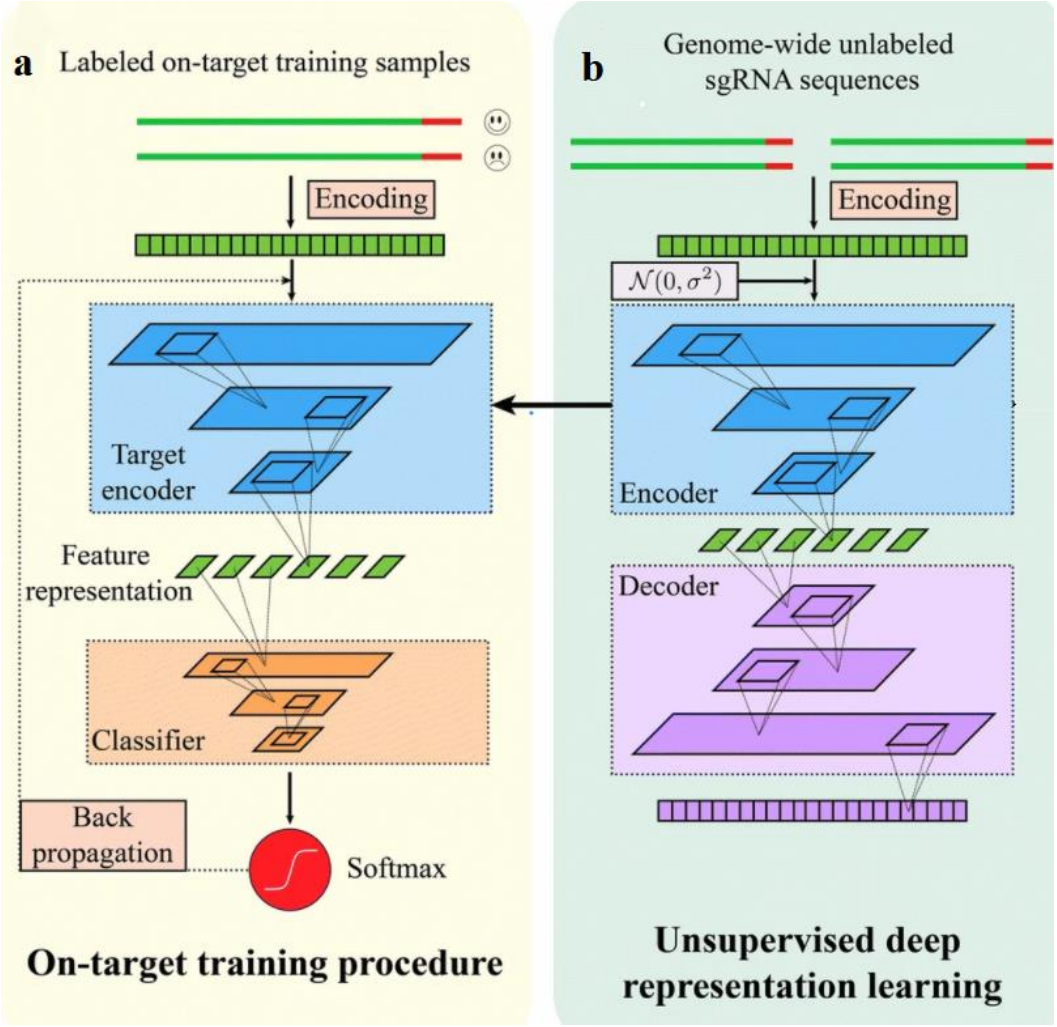

Figure 4 : a) Training details of DeepCRISPR for sgRNA on-target efficacy prediction. The Softmax and Identity functions correspond to classification and regression models, respectively.b) Unsupervised deep representation learning based on billions of genome-wide sgRNA sequences.

ii. Classifier and Regressor Models

These are the primary models are the classifier model and the regressor model. These models consist – like the autoencoder- from 2 Layer Groups: 1. The encoder layer 2. The classifier or regressor.

Both of these models have the same architecture for the encoder part as the encoder in the auto encoder ,Since they will load the Encoder layers from the auto encoder.

1. The encoder group:

It Primary goal is to encode the 23*8 DNA sequence format we got to a 1*256 sequence of numbers that contain important information about the DNA for the classifier or regressor to work on. These layers be loaded from the auto encoder and over tuned on this model again.

2. The Classifier or Regressor group:

This is the model' last layer this will take the 256-information outputted by the encoder layers and output the classification or regression output. These layers can differ but until now they have nearly identical layer configuration with expectation of the last layer.

Table 2 : Model Layers

Input Data of size: 1*23*8					
Auto-Encoder model		Classifier model		Regressor model	
Encoder group		Encoder group		Encoder group	
layer	output Size	layer	output Size	layer	output Size
convolution	1*23*32	convolution	1*23*32	convolution	1*23*32
Batch normalization					
Adding normal distributed noise					
convolution	1*12*64	convolution	1*12*64	convolution	1*12*64
Batch normalization					
Adding normal distributed noise					
convolution	1*12*64	convolution	1*12*64	convolution	1*12*64
Batch normalization					
Adding normal distributed noise					
convolution	1*6*256	convolution	1*6*256	convolution	1*6*256
Batch normalization					
Adding normal distributed noise					
convolution	1*6*256	convolution	1*6*256	convolution	1*6*256
Batch normalization					
Adding normal distributed noise					
Decoder group		Classifier group		Regressor group	
De-convolution	1*12*256	convolution	1*3*512	convolution	1*3*512
Batch normalization					
De-convolution	1*12*256	convolution	1*3*512	convolution	1*3*512
Batch normalization					
De-convolution	1*24*64	convolution	1*1*1024	convolution	1*1*1024
Batch normalization		SoftMax	1*1*2	squeeze	1
De-convolution	1*24*64	squeeze	1		
Batch normalization					
De-convolution					
Batch normalization					
De-convolution	1*23*8				

Table 2. This Table provide layers structures. on the left the layers structure for auto encoder. on the middle the layers structures for classifier. On the right the layers structures for regressor.

d. Model Enhancement

i. Dropout

Dropout can be used to have a large number of different network architectures by randomly dropping out nodes during training. This offers a computational, cheap, remarkable and effective regularization method to reduce overfitting.

We used a keep_prob = 0.9, the keep_prob is the percentage of nodes kept and dropped out which reduced the overfitting of our model with about 85%.

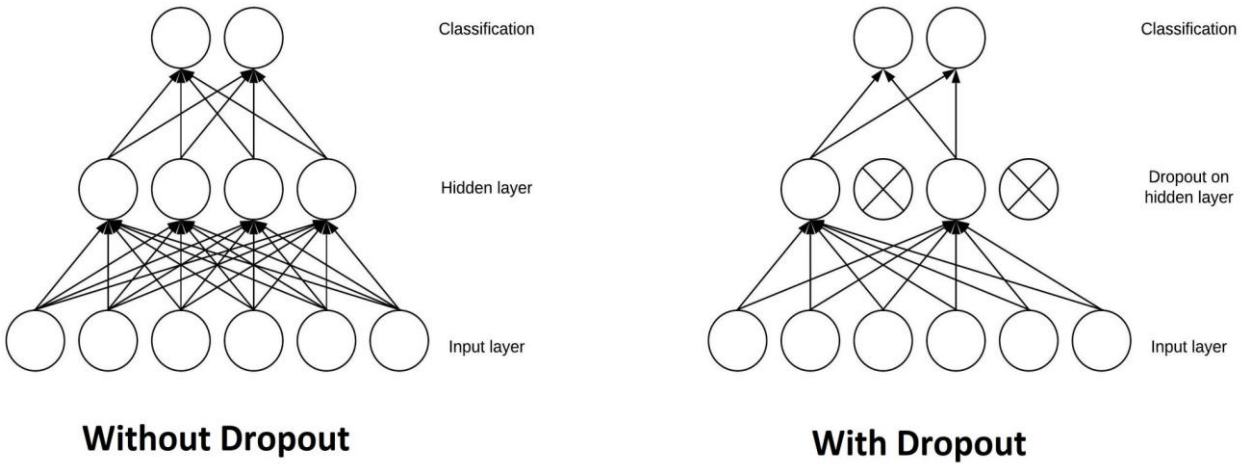

Figure 5 : with and without Dropout

ii. LSTM

It is an abbreviation for Long Short-Term Memory. LSTM has feedback connections it can an entire sequences of data not only single data points.
We tried to use LSTM to enhance the AUC-ROC accuracy of or model but we failed to implement it because our data is 4D data while the library of LSTM uses a 3D data and when we try to reshape our input data and the reshape the output data the AUC-ROC accuracy of our model decreases to 48%.

iii. Attention

Attention in deep learning is used as a vector of important weights to predict or infer an element, we estimate using the attention vector how strongly it is correlated to other elements and take the sum of their values weighted by the attention vector as the approximation of the target.

It is similar to dropout but it estimates the best keep_prob value after calculating it through trial and error.

The attention mechanism we used reduced the overfitting in our model but doesn't enhance the AUC-ROC accuracy of the model.

3. Results

For comprehensive and objective comparisons of DeepCRISPR with varying ROC-AUC/Spearman Correlation scores, six separate testing scenarios were carefully created. These comparisons show that:

1. The deep learning models (without unsupervised pre-training) are superior to shallow learning models
2. The unsupervised pre-training strategy boosts model performance
3. The data augmentation further improves model performance and model robustness
4. DeepCRISPR generalized generally well in new cell types for sgRNA on-target knockout efficacy prediction

a. Auto Encoder
b. Classification

Figure 6

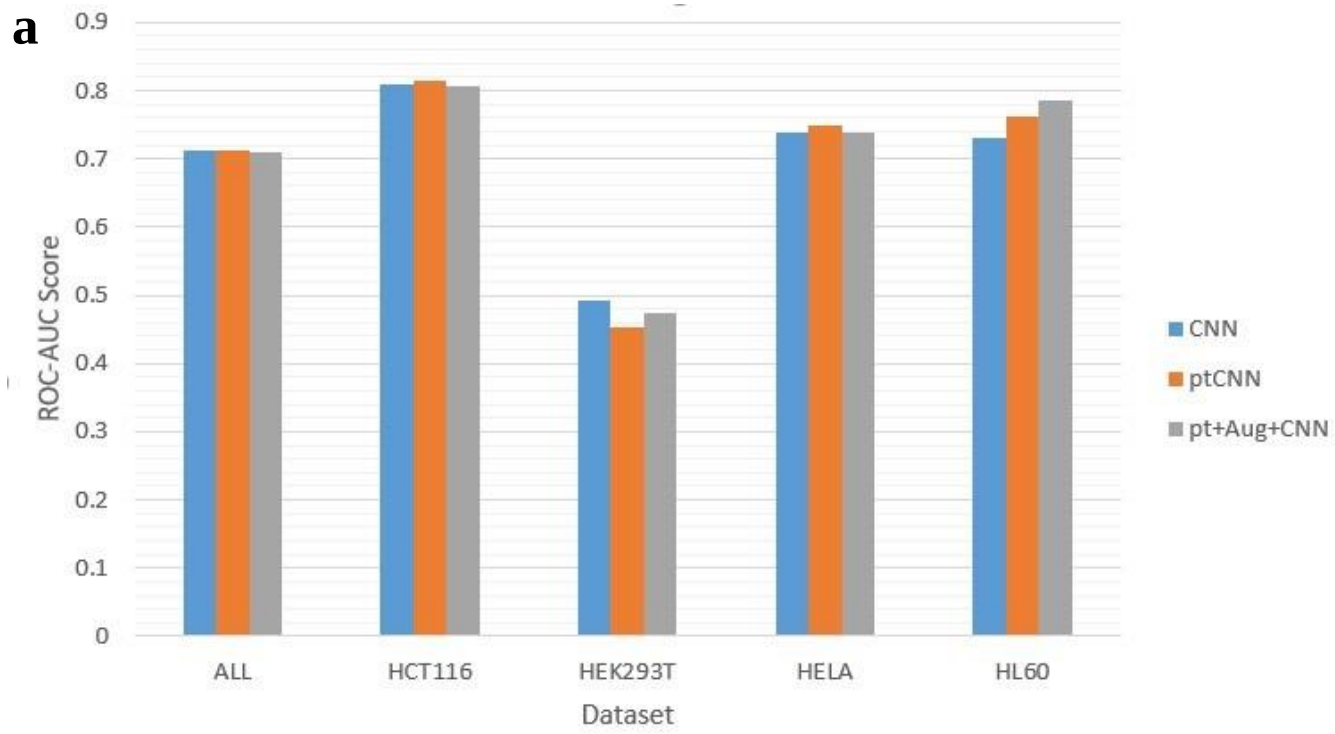

Fig 6.a: Comparison of sgRNA on-target efficacy predictions in a classification schema for various datasets, i.e., hct116 cell line, hek293t cell line, hela cell line, hl60 cell line, and the overall testing dataset using ROC-AUC Score

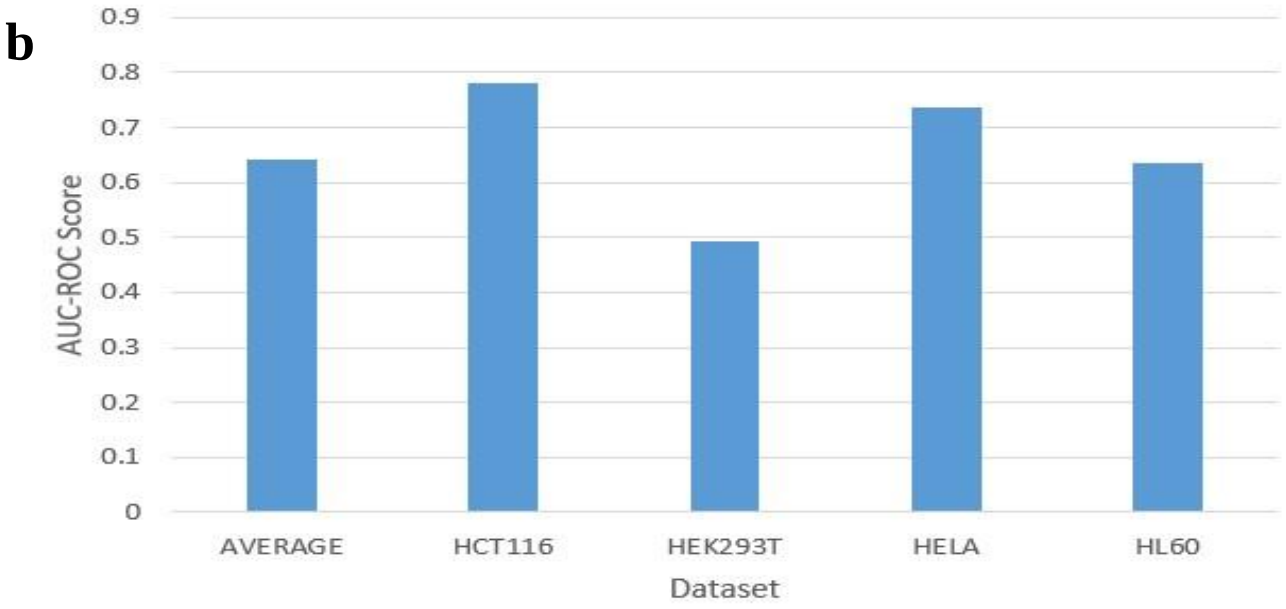

Fig 6.b: Leave cell type out comparison of sgRNA on-target efficacy prediction in a classification schema.

Figure 7

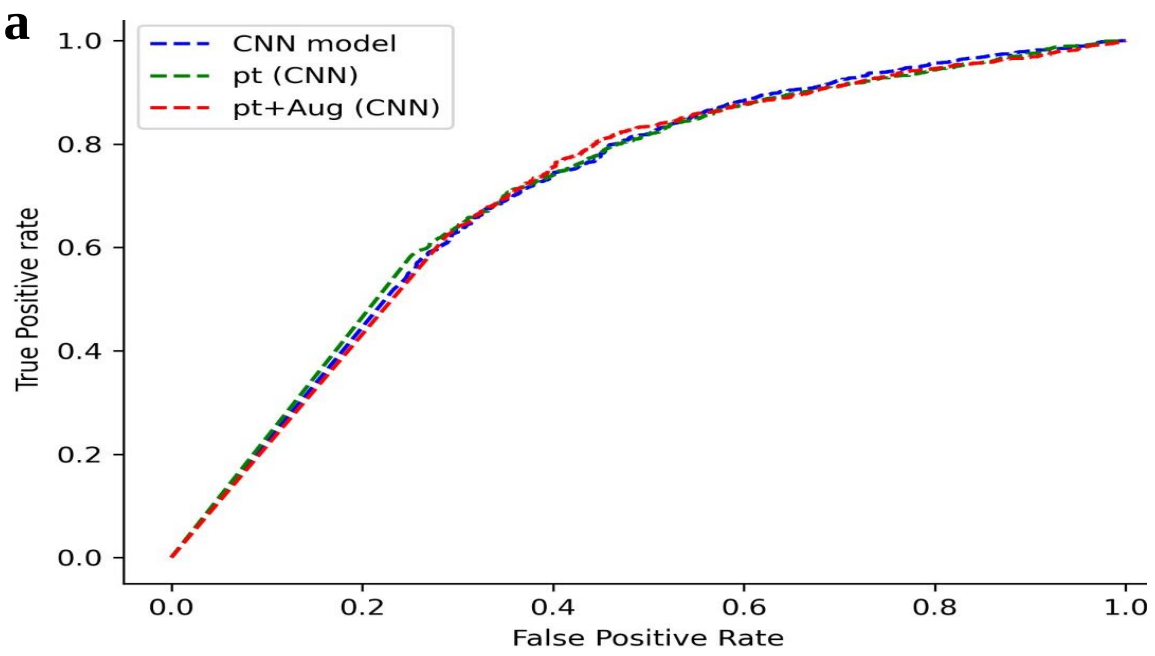

Fig 7.a: Benchmarking sgRNA on-target efficacy predictions on all testing data with ROC curve for first three classification scenarios

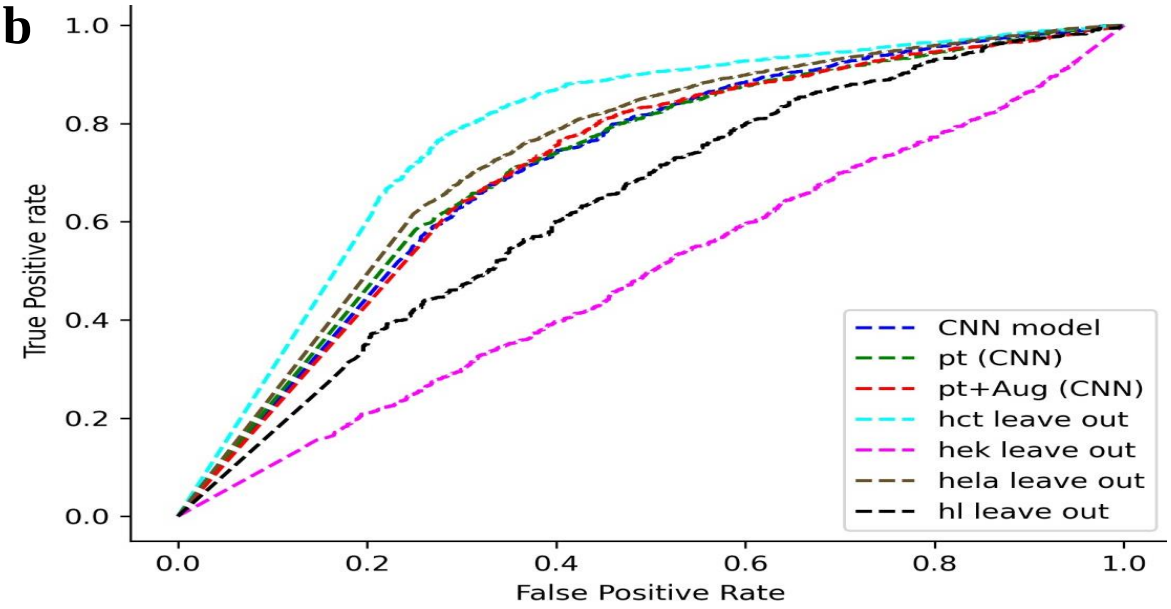

Fig 7.b: Benchmarking sgRNA on-target efficacy predictions with ROC curve for all four classification scenarios

c. Regression

Figure 8

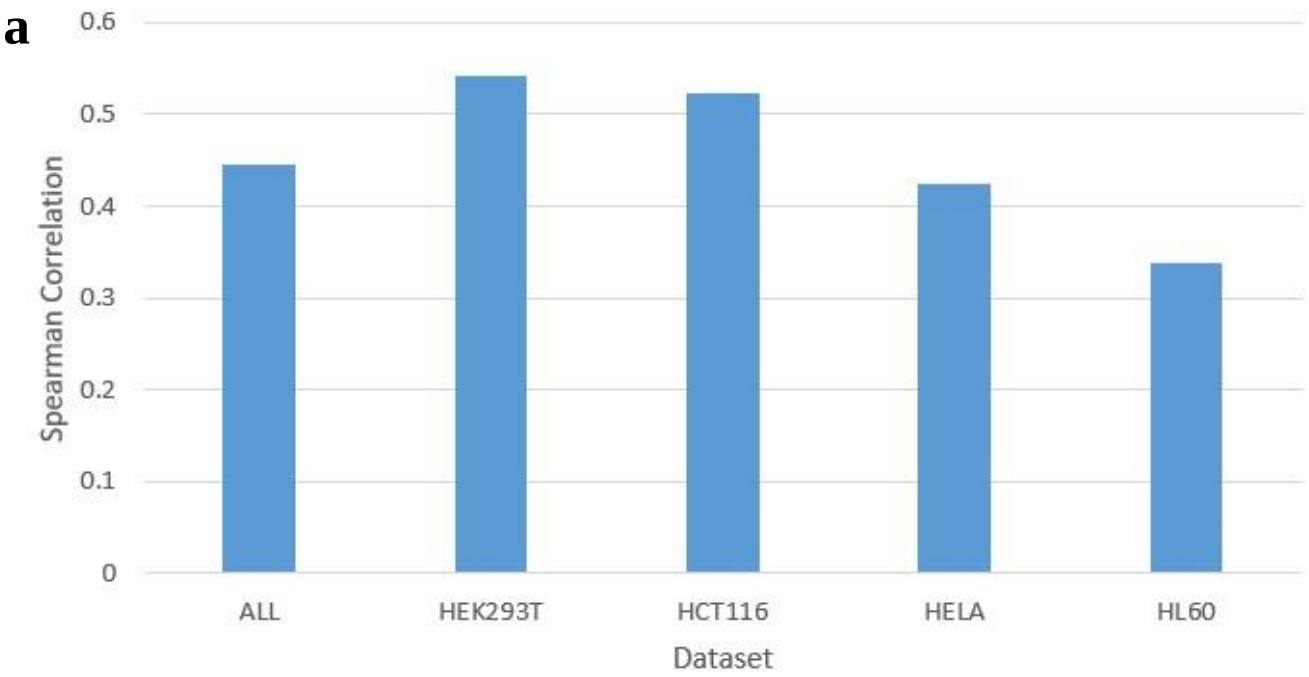

Fig 8.a: Comparison of sgRNA on-target efficacy predictions in a regression schema for various datasets, i.e., hct116 cell line, hek293t cell line, hela cell line, hl60 cell line, and the overall testing dataset using Spearman Correlation

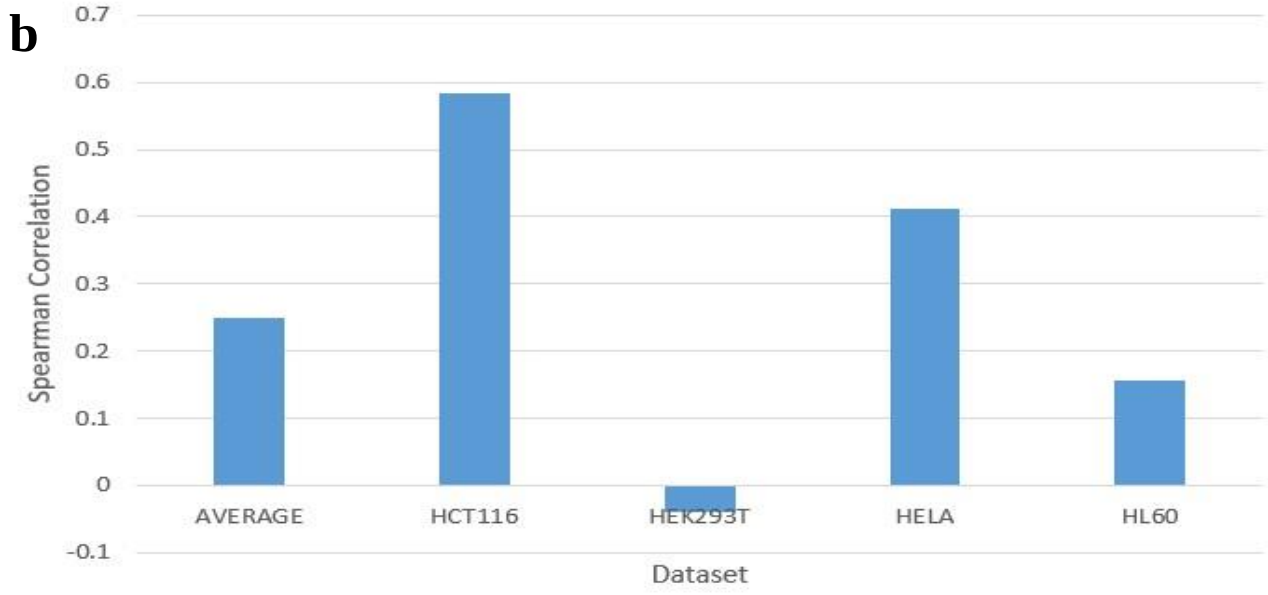

Fig 8.b: Leave cell type out comparison of sgRNA on-target efficacy prediction in a regression schema.