Recognition of CRISPR/Cas9 off-target sites through ensemble learning of uneven mismatch distributions

Hui Peng[1], Yi Zheng[1], Zhixun Zhao[1], Tao Liu[2,3], Jinyan Li[1,*]

1. Advanced Analytics Institute, University of Technology Sydney, PO Box 123, Broadway, NSW 2007, Australia; 2. Centre for Childhood Cancer Research, University of New South Wales, Kensington, NSW 2056, Australia; 3. Children's Cancer Institute Australia, Randwick, Sydney, NSW 2031, Australia. * To whom correspondence should be addressed

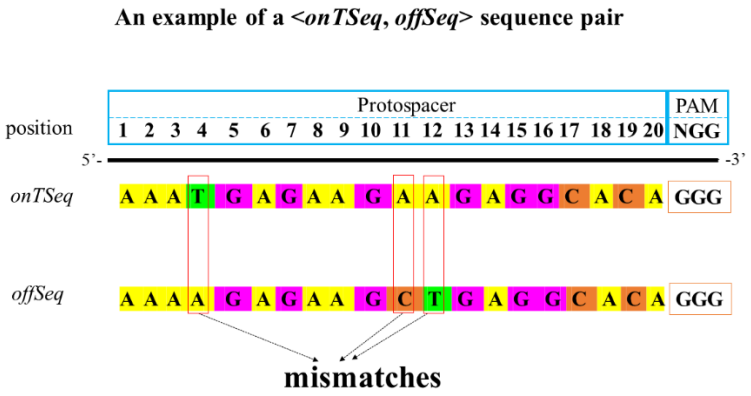
Simple description of the supplementary files

There are two supplementary files which contain the datasets we adopted in our main manuscript and some results that have not been shown in the main manuscript. The Supplementary file 1 provides the two positive sample sets and the negative samples. The Supplementary file 2 (this file) gives the additional results that have not been described in the main manuscript.

1. The observation of differences between positive and negative samples

1.1 The definition of the sequence pair

In our paper, the sample is defined as a sequence pair <*onTseq*, *canSeq*>, where *onTSeq* is the sequence corresponding to the on-target site sequence for a given sgRNA and *canSeq* is a sequence corresponding to the potential target site for this sgRNA other than the *onTSeq*. *CanSeq* will be an off-target site sequence (*offTSeq*) if it will be cut by the CRISPR/Cas9 system or it will be a no-editing site sequence (*noEdSeq*). < *onTseq*, *offTSeq*> is represented as a positive sample while < *onTseq*, *noEdSeq* > is a negative sample. **Supp. Fig. 1** shows an example of the <*onTseq*, *offTSeq*> sequence pair.



Supp. Fig. 1 An example of a <*onTseq, offTSeq*> sequence pair. The pair <*onTseq*, *canSeq*> will be labeled as positive or negative according to whether the *canSeq* will be cut by the CRISPR/Cas9 system.

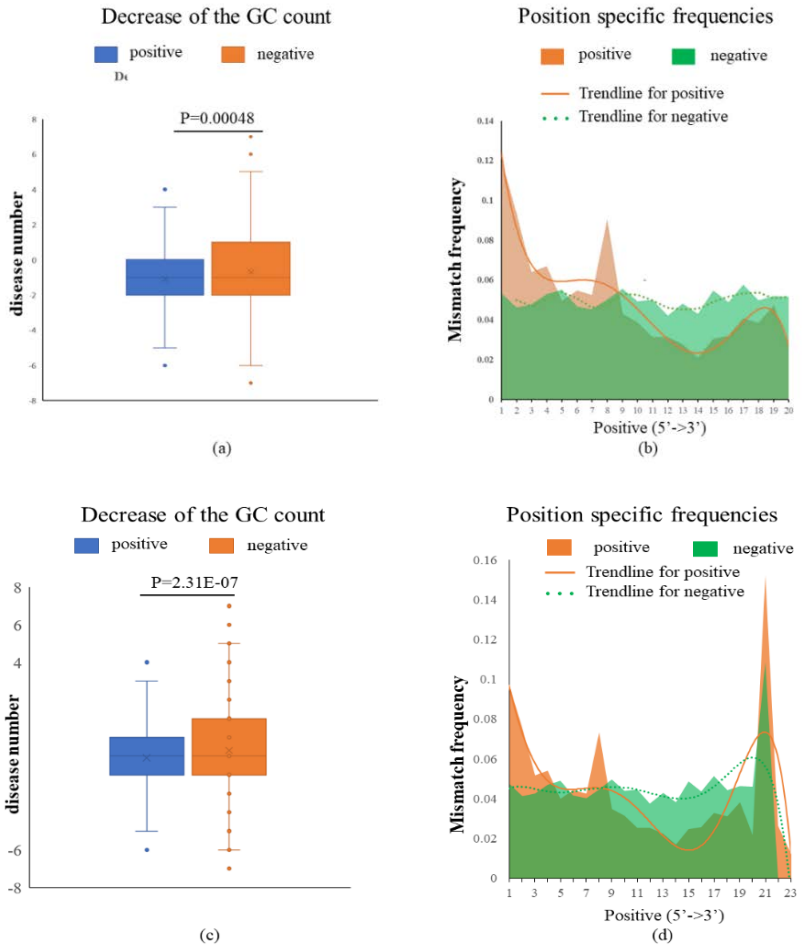
1.2 The comparison of decrease of GC count

We compared the decrease of the GC count (*ΔGC*) of the positive samples (<*onTSeq*, *offTSeq*>, *ΔGC*=GC(*offTSeq*)-GC(*onTSeq*)) and negative samples (<*onTSeq*, *noEdSeq*>, *ΔGC*=GC(*noEdSeq*)-GC(*onTSeq*)). The sample sizes of the positive samples and the negative samples are different, where there are over 500-fold of negative samples (408260) than the positive ones (742). Thus, the comparison of the properties between positive samples and negative samples

includes two parts: the original datasets and the balanced datasets. The comparison of the original datasets means all the 742 positive samples are compared with all the 408260 negative samples (two groups comparison with the Two sample Kolmogorov-Smirnov test [1]). For the balanced datasets' comparison, the positive sample group is compared with each of the 100 selected negative sample groups where there are 742 randomly selected negative samples in every negative sample group. The *onTSeq* and the *canSeq*∈{*offTSeq*, *noEdSeq*} are 23nt in length. As We fixed the PAM in those *noEdSeq* as 'NGG', we will discuss the properties comparison of the sequences with or without PAM sequences (protospacer+PAM with 23nt or protospacer only with 20nt).

For the *ΔGC* of the protospacer only comparison, we observed that in the original datasets, the positive samples decrease more of the GC count comparing to those negative ones (averagely -0.69 vs. -0.54, p-value=0.00048 with two-sample Kolmogorov-Smirnov test, see following **Supp. Fig. 2 (a)**). For the 100 balanced datasets which are constructed by the combination of all the positive samples and the same number of negative samples selected from the negative sample set, the *ΔGC* shows significant differences in 78 out of 100 of the comparisons (p-value<0.05). Most of the comparisons (97%) show that the negative sample groups have smaller decrease of the GC count than positive samples.

If we consider the protospacer+PAM sequences, the positive samples have a mean *ΔGC* = -1.09, while the negative samples have a mean *ΔGC* = -0.71. This difference is also significant with p-value=2.31E-07 by the two-sample K-S test (see **Supp. Fig. 2 (c)**). For the randomly selected 100 negative samples comparing with the positive samples, all of them had significant differences (p-value<0.05); and all of the randomly selected negative data sets had smaller decrease of the GC count than the positive samples. Thus, with or without PAM sequences have no influence that the off-target sites decrease more of the GC counts than those no-editing sites comparing to the on-target sites.



Supp. Fig. 2 The comparison of the properties between the positive and negative samples. (a) the *ΔGC* comparison of the protospacer sequences; (b) the mismatch frequencies comparison of the protospacer sequences. (c) the *ΔGC* comparison of the protospacer + PAM sequences; (d) the mismatch frequencies comparison of the protospacer + PAM sequences. Both of them are for the original datasets.

1.3 The comparison of the mismatch frequencies at different positions

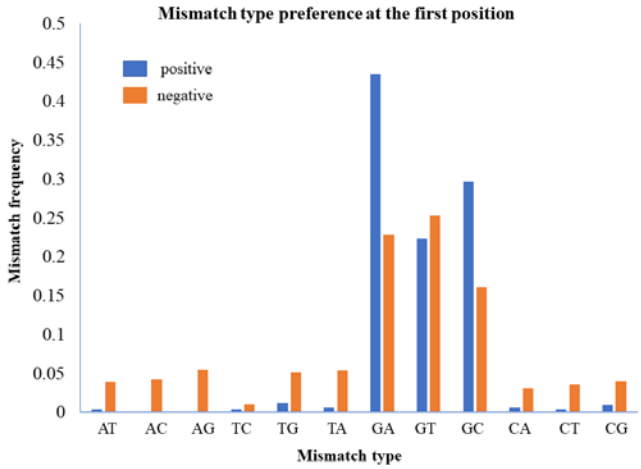
As is shown in **Supp. Fig. 2 (b)**, the mismatch distribution of the positive samples and the negative samples are significantly different (p-value=0.0082) when only the protospacer sequences were considered. The mismatches are more frequently appeared at the 5' close region in off-target sites while are about evenly distributed among all the 20 positions in no-editing sites (see the trendlines of the two groups). For the 100 balanced datasets, 98 out of the 100 comparisons show significant difference of the mismatch distribution for the 20 positions (with p-value<0.05), nearly all of the negative subgroups have about 5% of the total mismatches at each of the 20 positions.

When the protospacer and the PAM sequences are considered together, the first position of the PAM sequences contains most of the mismatches both for the positive samples and the negative samples. This is because that in fact, the first position of the PAM sequences has no conservation, while the following two nucleotides are conserved to be 'G' (sometimes PAM can be 'NRG' but will has lower cutting efficiency [2]). The other conclusions are similar to the protospacer only comparisons where the mismatch distribution of the positive samples and the negative samples are significantly different for both the original datasets or the balanced datasets (p-value= 0.0064 and all of the 100-balanced comparison's p-value<0.05).

1.4 mismatch type preference analysis

For the mismatch preference in a specific position analysis, we found that for the 20 positions (or 23 positions including the protospacer + PAM), only the 1st position shows significant difference of the mismatch type frequencies (p-value=0.0046, see **Supp. Fig. 3**). Both of the positive and negative samples contain more 'G-A', 'G-T' and 'G-C' mismatches at the 1st position (>95% for positive and >65% for negative), however, 'G-A' mismatch is the most preferred (account for 45% of the mismatches in the 1st position) for positive samples while 'G-C' is more preferred by negative samples (account for 25% of all the mismatches in the 1st position). For the balanced dataset comparison, there exist mismatch type preference differences at the 1st position for all the 100 comparisons (all with p-value<0.05). This may due to the fact that among the 29 sgRNAs, most of their spacer sequences start with the nucleotide of 'G' (19 out of 29 sgRNAs).

The three types of property differences indicate that we can characterize the samples with nucleotide composition features and mismatch distribution features to build a classification model. This lays the foundation of the idea of turning the off-target site prediction problem into a binary classification problem.



Supp. Fig. 3 Mismatch type preference at the 1st position of positive samples and negative samples.

2. parameter selection with leave-one-guide-out cross-validation on training dataset

We applied the 527 positive samples in D_{+}^{high} containing 11 sgRNAs and these 11 sgRNAs related negative samples as the training dataset to select the three super-paramenter namely penalty parameter *c*, *gamma* for RBF kernel of SVM and ensemble size *n*. Leave-one-guide-out cross-validation was applied in this process. To select *c* ($c=2^C$, $C \in$[-6, 6]) and *gamma* ($g=2^G$, $G \in$[-6, 6]), we fixed *n*=|*negative samples*|/|*positive samples*| where |*negative samples*| means the sample size of the negative samples. After obtaining the best *c* and *g* (*c*=2, *g*=0.0625), we vary *n* in {1, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 150, 200, 250} to see the performance changes during the logocv. For a given n, the logocv was repeat 10 times (thus, the negative samples in training data of each fold have also been randomly selected 10 times). The mean AUROC and mean AUPRC are shown in following **Supp. Tab. 1.**

n	**AUROC**	***std*(AUROC)***	**AUPRC**	***std*(AUPRC)**
1	0.978111	0.001795144	0.355818	0.018617862
10	0.981556	0.000529509	0.393053	0.0045806
20	0.981688	0.000403103	0.396315	0.003839303
30	0.981737	0.000357377	0.394564	0.00488384
40	0.981729	0.000285634	0.395846	0.001992267
50	0.981739	0.000257807	0.396509	0.002510356
60	0.982033	0.000272282	0.396754	0.002598641
70	0.9819	0.000133014	0.397631	0.002707483
80	0.981858	0.000143883	0.396258	0.002777306
90	0.981873	0.000280168	0.396909	0.002270525
100	0.981947	0.000175073	0.396208	0.00298467
150	0.981917	9.08415E-05	0.39665	0.001942358
200	0.981944	8.70863E-05	0.397141	0.001278991
250	0.981834	0.000103914	0.397134	0.001591173
fixed**	0.981924	-	0.39805	-

* std(AUROC) means the standard deviation of the 10 times repeated logocv results; **fixed means when we fixed *n*=|*negative samples*|/|*positive samples*|, the corresponding performance (without repeat experiment)

We can find that comparing the single base classifier and the ensemble classifiers, the performances have been increased. Especially, the single base classifier achieves average mean AUPRC of 0.3558 which is about 4% smaller than the other ensemble classifiers (more than 0.39). When the ensemble size n increases, the performance increases a little (within 0.01 for both AUROC and AUPRC) however the standard deviations decrease obviously (for example, when *n* increases from 10 to 40, the *std*(AUROC) decreases about 2folds and the *std*(AUPRC) decreases more than 2folds). Thus, we draw the conclusion that the ensemble strategy exactly improves the prediction accuracy and stability. As bigger *n* may result in huge increase of running time but litter performance improvement, we finally select *n*=40.

3. Comparison of the detected off-target sites by computation methods and the high-throughput sequencing methods

To compare how the computational methods detected off-target sites overlap with the high-throughput sequencing methods, we used two examples to test five computational methods and 5 sequencing methods. The computational methods include the proposed methods, CCTOP [3], MIT-score [4], CROP-IT [5] and CFD [2]. The sequencing methods are GUIDE-seq (GUIDE) [6], Digenome-seq [7] (Digenome), HTGTS [8], multiplex Digenome-seq [9] (mDigenome) and CIRCLE-seq [10] (CIRCLE). The first example is the site EMX1 (protospacer+PAM: GAGTCCGAGCAGAAGAAGAAGGG), where all the five sequencing methods have been tested with it. As there is no other site that has been tested with all the five sequencing methods, we selected the site HEK4 (protospacer+PAM: GGCACTGCGGCTGGAGGTGGGGG) as another example.

To test our method, we trained the ensemble classification model with those samples excluding the EMX1 or HEK4 site related samples (when testing EMX1, the EMX1 related samples are excluded and when testing HEK4, HEK4 related samples are excluded). For the other four computational methods, they will compute a score for each of the candidate off-target sites of EMX1 or HEK4. Then, these scores will be ranked, and those with higher scores are more possible to be off-target sites. As there is no threshold of these methods for determining positive or negative (real off-target site or no-editing site), for a fair comparison, we defined those ranked top *N* ones as positive, where *N* is the number of off-target sites predicted by our method. In addition, the union of all the five sequencing methods' detections were computed as the 'Integrated' results. The number of overlapped detections between the computational methods and the 'Integrated' were also counted. The following **Supp. Tab. 2** shows the number of off-target sites detected by sequencing methods and the number of overlapped off-target sites that detected by both of the computational methods and the sequencing methods.

Supp. Tab. 2 The number of off-target sites detected by sequencing methods and the number of overlapped off-target sites that detected by both of the computational methods and the sequencing methods.

EMX1						
	CIRCLE	GUIDE	HTGTS	Digenome	mDigenome	Integrated
*****	176	15	13	27	142	259
proposed	108**	15	13	24	59	112
CCTOP	37	11	8	15	24	39
CROP-IT	35	5	5	15	30	37
MIT-score	50	12	10	21	36	52
CFD	48	8	7	10	27	51
HEK4						
	CIRCLE	GUIDE	HTGTS	Digenome	mDigenome	Integrated
*****	980	133	-	38	215	1011
proposed	417	120	-	35	163	421
CCTOP	85	42	-	22	56	88
CROP-IT	81	43	-	13	50	85
MIT-score	182	83	-	32	103	185
CFD	248	83	-	24	95	250

*these rows show the number of sequencing methods detected off-target sites;

**108 means there are 108 common detected off-target sites by our proposed method and the CIRCLE method

For the EMX1 site, there are 673 off-target sites predicted by our proposed method while 1202 off-target sites were found for the HEK4 site. Most of the sequencing methods detected off-target sites were also detected by our method. The other four stat-of-the-art methods cannot work as well as our method according to the results. As those sequencing methods provide bona fide detections, we can draw the conclusion that our method can detect more reliable off-target sites than the other computational methods.

4. Comparison of our method with the two most recently published methods

We note that this work was started from the end of August 2017. Thus, two most recently published methods were not compared with ours in our main manuscript. We present the comparison here as a part of the supplementary results.

Abadi *et al.* published their off-target prediction tool at October 6, 2017 and named their tool CRISTA [15]. CRISTA applied more types of features for characterizing the off-target site sequences and corresponding on-target sequencings. In addition, it considers the off-target sites that contain bulges. Though, bulges should not be ignored for finding potential off-target sites, we currently don't work on this problem as we cannot obtain enough data to train a reliable prediction model. Though in Abadi et al.'s paper, they claim that 87 out of 491 targets contain bulges, most of these

bulges are produced by their alignments but not validation experiments. On the other hand, considering of bulges may increases the candidate off-target sites sharply which will make the prediction more complex, thus we hope to solve it in our future work.

To compare our method with CRISTA, we adopted the datasets used by CRISTA to test our method by three processes such as the leave-one-sgRNA-out cross-validation (logocv), the leave-one-study-out cross-validation (losocv) and the independent validation (the parameter C=1 and G=-4 is set for our method), where CRISTA has also been evaluated by these processes. As our method is proposed to solve the classification issue, we mainly compare our method with CRISTA in its binary classification format. CRISRA's prediction scores are obtained from the literature's supplementary files [15]. According to Abadi et al.'s paper, they selected 0.12 and 0.4 as two thresholds for determining cleavage sites because 95% and 50% of the positive samples in their cross-validation datasets pass these thresholds respectively. However, on the external validation dataset, the thresholds are 0.39 (95%) and 0.54 (50%) respectively. To compare our method and CRISTA fairly, we fixed their threshold as 0.4 for all the three processes. Furthermore, we also computed six more performance indexes such as specificity = TN/(TN+FP), Recall = TP/(TP+FN), Precision = TP/(TP+FP), Accuracy = (TP+TN) / (TP+FP+TN+FN), F1 = 2*Recall*Precision/(Recall + Precision) and MCC = (TP*TN - FP*FN) / SQRT((TP + FN)*(TP + FP)*(TN+FN) * (TN+FP)) to evaluate the binary classification energies of two methods (AUROC and AUPRC were all computed). Here, TP, FP, TN and FN are true positive, false positive, true negative and false negative respectively. The detail results are shown in following **Supp. Tab. 4**.

Supp. Tab. 4 Comparison of our method and CRISTA

	logocv		**losocv**		**validation**		
	our*	CRISTA	our	CRISTA	our	CRISTA	our model**
Specificity	0.87	1.00	0.91	1.00	0.02	0.15	0.01
Recall	0.88	0.71	0.86	0.56	1.00	0.91	0.99
Precision	0.88	0.99	0.90	0.99	0.40	0.41	0.39
Accuracy	0.88	0.85	0.89	0.78	0.41	0.45	0.39
F1	0.87	0.80	0.88	0.69	0.57	0.56	0.56
MCC	0.76	0.75	0.78	0.62	0.09	0.08	-0.02
AUROC	0.92	0.99	0.93	0.97	0.69	0.67	0.68
AUPRC	0.73	0.77	0.90	0.96	0.66	0.66	0.49

* our classifier trained and tested by CRISTA's datasets; ** our ensemble classifier trained by our own datasets but excluding samples in the validation dataset and tested by CRISTA's validation dataset.

We can see that our method also obtains good performance when trained and tested with CRISTA datasets. Though CRISTA achieves better AUROC and AUPRC during logocv and losocv, our method can obtain better Recall, accuracy, F1 and MCC. In addition, in the independent validation, our method has bigger values of Recall, F1, MCC, AUROC and similar AUPRC comparing with CRISTA. Especially, our method always can predict more off-target sites correctly (bigger Recall) but less non-off-target site correctly (smaller Specificity) comparing with CRISTA. It should be noted that in the application of gene editing, false positives may be more tolerated than false negatives as we need to avoid real off-target sites and the false off-target site has no impact to our targets. In addition, CRISTA selected those negative samples with the rule that the alignment score of the sites greater than 14.75. This may induce bias as the real off-target sites breaking this rule may be ignored as the same time (this is why we adopted genome wide negative samples in our method). In the last column of **Supp. Tab. 4**, we also test our ensemble classifier that trained by our own dataset with the external validation dataset. We can see that it also can achieve higher recall and AUROC and similar F1 comparing with CRISTA. We will also test CRISTA with our datasets for a fair comparison in our following work.

Another tool published on January 20, 2018 by Listgarten et al. [16] is named the Elevation. This tool also applies machine learning to predict off-target sites. However, we cannot get accessible to their article up to now. Although

we can visit their web server, we cannot get the detail off-target scores for a given on-target site from their server. It's impossible for us to compare their method with ours currently.

5. Case studies of applying our method for selecting sgRNAs to cure genetic diseases

Two case studies were presented here to show that our method can be applied to assist the sgRNA design in practical usage. One is the application of CRISPR/Cas9 system for preventing retinal degeneration [11] and another is the use of CRISPR/Cas9 system to cure autosomal dominant hearing loss [12]. The off-target prediction methods MIT-score and CFD were also compared with our method. These two methods have been integrated into their web servers CRISPR Design [4] (http://crispr.mit.edu/, the off-target prediction method is previous MIT-score) and sgRNA Designer [2] (https://portals.broadinstitute.org/gpp/public/analysis-tools/sgrna-design, it uses the CFD to predict the off-target sites) respectively.

In the first case, there are five potential sgRNAs for selection where their cutting efficiencies have already been tested. Our objective is to detect off-target sites of these potential sgRNAs and determine which one can work better with higher cutting efficiency but lower off-target potential. The five sgRNAs' protospacer + PAM sequences are: (1) NT1: GAGCCTTCTGAGGGCCGATCTGG, cut efficiency = 21.9% (Indel); (2) NT2: GTATGGTGTGGAGCCCA-ACGAGG, cut efficiency = 22.7%; (3) NT3: AGGCACCGAGCTGTATGGTGTGG, cut efficiency = 22.5%; (4) NT4: GGTGGCGAGGCCCCTAGGCCAGG, cut efficiency = 23.2%; (5) NT5: CCTGGAGGAGCTATATTGGCTGG, cut efficiency = 18.3%. For the web servers CRISPR Design and sgRNA Designer, the mouse Nrl gene transcript (Ensembl ID: ENSMUST00000062232.13, downloaded from ensembl database [13]) was used as input. As to our method, we obtained all the potential off-target sites by the offline tool Cas-OFFinder [14] with the protospacer + PAM sequences as the inputs.

In the second case, there are originally four potential sgRNAs designed by the authors namely: (1) Tmc1-mut1: GGGACAGAACTTCCCCAGGAGGG, cut efficiency = 4.1% (Indel); (2) Tmc1-mut2: TGGGACAGAACTTCCCC-AGGAGG, cut efficiency = 0.74%; (3) Tmc1-mut3: GGGTGGGACAGAACTTCCCCAGG, cut efficiency = 10%; and (4) Tmc1-mut4: TGGGACAGAACTTCCCCAGG, cut efficiency = ~2%. The Tmc1-mut4 is a truncated sequence which has only 17nt, as our method and the other two tools are not supporting the truncated sequences, this sgRNA was not tested. In addition, the given region being targeted contains mutation (T1235A mutation), we didn't use the original gene sequence as input but use the "AAACCTTTCCAACCGTGTCTCCTTGTAGATGAACATGGT AATGTCCCTCCTGGGGAAGTTCTGTCCCACCCTGTTTGACTTATTTGCTGAACTGGAAGATTACCATCC TCTCATTGCTCT" which include all the sgRNAs' spacer and the mutation instead for the inputs of CRISPR Design and sgRNA Designer. The inputs for our method is similar to above Nrl case. The detail results of recommended sgRNAs by different methods were described in our main manuscript.

For the comparison of the computational methods detected off-target sites of the Tmc1-mut3, we again adopt the implemented methods MIT-score and CFD instead of their web servers as the web server's outputs cannot provide detail off-target information. Similar to the process of computing the overlapping of the detected off-target sites between computational methods and sequencing methods, we also compared our method, MIT-score and CFD detected off-target sites with the literature mentioned GUIDE-seq's detections. The 10 GUIDE-seq detected off-target sites for Tmc1-mut3 are listed in **Supp. Tab. 4** (data are obtained from the literature supplementary file). The predicted scores and the ranks by computational methods for the 10 off-target sites detected by GUIDE-seq are shown in the following **Supp. Tab. 5**.

Supp. Tab. 4 The 10 GUIDE-seq detected off-target sites for Tmc1-mut3

targets	5'-Sequence-3'	Mismatches (MMs)	NCBI accession	Predicted function	Location	Indels in Bth/Bth
Bth	GGGTGGGACAGAACTTC CCCAGG	0MMs	N/A		chr9	31%
Off-T1	GGGAGGGACAGAGCTTC CCCAGG	2MMs [4:13]	N/A		chr1	8.10%
Off-T2	GTGAGGGAGAGAACTTC CCCTGG	3MMs [2:4:9]	N/A		chr16	4.40%

Off-T3	AGTTGGTACAGAACTTC CCCAGG	3MMs [1:3:7]	NC_000068.7	CD82 antigen	chr2	2.60%
Off-T4	TTGTGGGACAGAAATTC CCCAGG	3MMs [1:2:14]	N/A		chr12	3.90%
Off-T5	AGAGGAGACAGAACTCC CCCAGG	5MMs [1:3:4:6:16]	N/A		chr13	3.40%
Off-T6	GGGTGGGACAGATCTTC CCAGGG	2MMs [13:20]	NC_000067.6	hemicentin-1 isoform	chr1	0.68%
Off-T7	GTGTAGGACAGAACTTC GCCAGG	3MMs [2:5:18]	XM_0065070 26.3	inositol 1,4,5-triphosphate receptor 2	chr6	1.50%
Off-T8	GGTGAGACCAGAGCTTC CCCTGG	6MMs [3:4:5:7:8:13]	XR_389309.3	unknown	chr5	1.20%
Off-T9	AGGTGGGAAAGAACTTC TCCGGG	3MMs [1:9:18]	NC_000070.6	paralemmin A kinase anchor protein	chr4	1.40%
Off-T10	GGGTGGTAAAGAACTTC TCCTGG	3MMs [7:9:18]	N/A		chr10	0.05%

Supp. Tab. 5 The predicted scores and the ranks of the 10 GUIDE-seq detected off-target sites for the Tmc1-mut3

proposed		**CCTOP**		**MIT-score**		**CROP-IT**		**CFD**	
rank	**score**	**rank**	**score**	**rank**	**score**	**rank**	**score**	**rank**	**score**
4	1.00	10	-12.77	2	3.12	436	538	30	0.31
26	0.96	2	-8.67	4	1.55	1	628	176	0.19
2	1.00	1	-6.51	3	1.71	2	628	58	0.26
5	1.00	32	-15.48	35	0.44	509	530	101	0.22
16	0.97	348	-26.48	639	0.09	117	545	39	0.28
8	1.00	2295	-49.04	7	1.14	1186	470	284	0.15
6	1.00	561	-30.55	16	0.66	70	560	1334	0.07
660	0.08	287	-24.87	281	0.16	689	510	1289	0.08
7	1.00	714	-32.98	37	0.40	68	560	5	0.50
24	0.99	872	-35.37	123	0.23	64	560	26	0.31

Reference

[1] Lilliefors, H. W. (1967). On the Kolmogorov-Smirnov test for normality with mean and variance unknown. Journal of the American statistical Association, 62(318), 399-402.

[2] Doench, J. G., Fusi, N., Sullender, M., Hegde, M., Vaimberg, E. W., Donovan, K. F., ... & Virgin, H. W. (2016). Optimized sgRNA design to maximize activity and minimize off-target effects of CRISPR-Cas9. Nature Biotechnology, 34(2), 184.

[3] Stemmer, M., Thumberger, T., del Sol Keyer, M., Wittbrodt, J., & Mateo, J. L. (2015). CCTop: an intuitive, flexible and reliable CRISPR/Cas9 target prediction tool. PLOS ONE, 10(4), e0124633.

[4] Hsu, P. D., Scott, D. A., Weinstein, J. A., Ran, F. A., Konermann, S., Agarwala, V., ... & Cradick, T. J. (2013). DNA targeting specificity of RNA-guided Cas9 nucleases. Nature Biotechnology, 31(9), 827.

[5] Singh, R., Kuscu, C., Quinlan, A., Qi, Y., & Adli, M. (2015). Cas9-chromatin binding information enables more accurate CRISPR off-target prediction. Nucleic Acids Research, 43(18), e118-e118.

[6] Tsai, S. Q., Zheng, Z., Nguyen, N. T., Liebers, M., Topkar, V. V., Thapar, V., ... & Aryee, M. J. (2015). GUIDE-seq enables genome-wide profiling of off-target cleavage by CRISPR-Cas nucleases. Nature Biotechnology, 33(2), 187.

[7] Kim, D., Bae, S., Park, J., Kim, E., Kim, S., Yu, H. R., ... & Kim, J. S. (2015). Digenome-seq: genome-wide profiling of CRISPR-Cas9 off-target effects in human cells. Nature Methods, 12(3), 237.

[8] Frock, R. L., Hu, J., Meyers, R. M., Ho, Y. J., Kii, E., & Alt, F. W. (2015). Genome-wide detection of DNA double-stranded breaks induced by engineered nucleases. Nature Biotechnology, 33(2), 179.

[9] Kim, D., Kim, S., Kim, S., Park, J., & Kim, J. S. (2016). Genome-wide target specificities of CRISPR-Cas9 nucleases revealed by multiplex Digenome-seq. Genome Research, 26(3), 406-415.

[10] Tsai, S. Q., Nguyen, N. T., Malagon-Lopez, J., Topkar, V. V., Aryee, M. J., & Joung, J. K. (2017). CIRCLE-seq: a highly sensitive in vitro screen for genome-wide CRISPR–Cas9 nuclease off-targets. Nature Methods, 14(6), 607.

[11] Yu, W., Mookherjee, S., Chaitankar, V., Hiriyanna, S., Kim, J. W., Brooks, M., ... & Swaroop, A. (2017). Nrl knockdown by AAV-delivered CRISPR/Cas9 prevents retinal degeneration in mice. Nature Communications, 8, 14716.

[12] Gao, X., Tao, Y., Lamas, V., Huang, M., Yeh, W. H., Pan, B., ... & Li, Y. (2018). Treatment of autosomal dominant hearing loss by in vivo delivery of genome editing agents. Nature, 553(7687), 217.

[13] Aken, B. L., Achuthan, P., Akanni, W., Amode, M. R., Bernsdorff, F., Bhai, J., ... & Gil, L. (2016). Ensembl 2017. Nucleic acids research, 45(D1), D635-D642.

[14] Bae, S., Park, J., & Kim, J. S. (2014). Cas-OFFinder: a fast and versatile algorithm that searches for potential off-target sites of Cas9 RNA-guided endonucleases. Bioinformatics, 30(10), 1473-1475.

[15] Abadi, S., Yan, W. X., Amar, D., & Mayrose, I. (2017). A machine learning approach for predicting CRISPR-Cas9 cleavage efficiencies and patterns underlying its mechanism of action. PLoS computational biology, 13(10), e1005807.

[16] Listgarten, J., Weinstein, M., Kleinstiver, B. P., Sousa, A. A., Joung, J. K., Crawford, J., ... & Fusi, N. (2018). Prediction of off-target activities for the end-to-end design of CRISPR guide RNAs. Nature Biomedical Engineering, 2(1), 38.