nature

SUPPLEMENTARY INFORMATION

https://doi.org/10.1038/s41586-018-0686-x

In the format provided by the authors and unedited.

Predictable and precise template-free CRISPR editing of pathogenic variants

Max W. Shen[1,2,12], Mandana Arbab[3,4,5,12], Jonathan Y. Hsu[6,7], Daniel Worstell[8], Sannie J. Culbertson[8], Olga Krabbe[8,9], Christopher A. Cassa[8,10], David R. Liu[3,4,5]*, David K. Gifford[2,6,10,11]* & Richard I. Sherwood[8,9]*

[1]Computational and Systems Biology Program, Massachusetts Institute of Technology, Cambridge, MA, USA. [2]Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA. [3]Merkin Institute of Transformative Technologies in Healthcare, Broad Institute of Harvard and MIT, Cambridge, MA, USA. [4]Department of Chemistry and Chemical Biology, Harvard University, Cambridge, MA, USA. [5]Howard Hughes Medical Institute, Harvard University, Cambridge, MA, USA. [6]Department of Biological Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA. [7]Molecular Pathology Unit, Center for Cancer Research, and Center for Computational and Integrative Biology, Massachusetts General Hospital, Charlestown, MA, USA. [8]Division of Genetics, Department of Medicine, Brigham and Women's Hospital and Harvard Medical School, Boston, MA, USA. [9]Hubrecht Institute for Developmental Biology and Stem Cell Research, Royal Netherlands Academy of Arts and Sciences (KNAW), Utrecht, The Netherlands. [10]Broad Institute of MIT and Harvard, Cambridge, MA, USA. [11]Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, USA. [12]These authors contributed equally: Max W. Shen, Mandana Arbab.
*e-mail: drliu@fas.harvard.edu; gifford@mit.edu; rsherwood@rics.bwh.harvard.edu

Supplementary Information

Predictable and precise template-free CRISPR editing of pathogenic variants

Max W. Shen[‡1,2], Mandana Arbab[‡3,4,5], Jonathan Hsu[6,7], Daniel Worstell[8], Olga Krabbe[8,9], Christopher A. Cassa[8,10], David R. Liu[3,4,5]*, David K. Gifford[2,6,10,11]* and Richard I. Sherwood[8,9]*

[‡] *These authors contributed equally to this work.*
[1] *Computational and Systems Biology Program, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.*
[2] *Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.*
[3] *Merkin Institute of Transformative Technologies in Healthcare, Broad Institute of Harvard and MIT, Cambridge, Massachusetts, USA.*
[4] *Department of Chemistry and Chemical Biology, Harvard University, Cambridge, Massachusetts 02138, USA.*
[5] *Howard Hughes Medical Institute, Harvard University, Cambridge, Massachusetts 02138, USA.*
[6] *Department of Biological Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.*
[7] *Molecular Pathology Unit, Center for Cancer Research, and Center for Computational and Integrative Biology, Massachusetts General Hospital, Charlestown, Massachusetts, USA.*
[8] *Division of Genetics, Department of Medicine, Brigham and Women's Hospital and Harvard Medical School, Cambridge, Massachusetts, USA.*
[9] *Hubrecht Institute, Utrecht, the Netherlands.*
[10] *Broad Institute of MIT and Harvard, Cambridge, Massachusetts, USA.*
[11] *Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.*

* Correspondence should be addressed to R.I.S. (rsherwood@rics.bwh.harvard.edu) or D.K.G. (gifford@mit.edu) or D.R.L. (drliu@fas.harvard.edu).

Table of Contents

Supplementary Discussion

Cellular repair of double-stranded DNA breaks and inDelphi

DNA double-strand breaks are detrimental to genomic stability, and as such the detection and faithful repair of genomic lesions is crucial to cellular integrity. A large number of genes have evolved to respond to and repair DNA double-strand breaks, and these genes can be broadly grouped into a set of DNA repair pathways[1], each of which differs in the biochemical steps it takes to repair DNA double-strand breaks. Accordingly, these pathways tend to produce characteristically distinguishable non-wildtype genotypic outcomes.

The goal of our machine learning algorithm, inDelphi, is to accurately predict the identities and relative frequencies of non-wildtype genotypic outcomes produced following a CRISPR/Cas9-mediated DNA double-strand break. To accomplish this goal, we developed parameters to classify three distinct categories of genotypic outcomes, microhomology deletions, microhomology-less deletions, and insertions, informed by the biochemical mechanisms underlying the DNA repair pathways that typically give rise to them.

Double strand breaks are thought to be repaired via four major pathways: classical non-homologous end-joining (c-NHEJ), alternative-NHEJ (alt-NHEJ), microhomology-mediated end-joining (MMEJ), and homology-directed repair (HDR)[1]. To create inDelphi, we developed three machine learning modules to model genotypic outcomes assuming characteristic of the c-NHEJ, microhomology mediated alt-NHEJ, and MMEJ pathways. While template-free CRISPR/Cas9 DNA double-strand break may lead to HDR repair via endogenous homology templates that exist in *trans*[2], we do not explicitly model HDR-characteristic outcomes using our algorithm.

Before proceeding, it is important to note that while specific DNA repair pathways are characteristically associated with distinct genotypic outcomes, the proteins involved in the various pathways and the resulting repair products may at times overlap. This fact has several implications. First, we cannot make conclusive statements about the role of specific proteins or pathways in specific genotypic outcomes without perturbation experiments (e.g. our comparison of wildtype and *Prkdc*$^{-/-}$*Lig4*$^{-/-}$ mESCs can illuminate the roles of these proteins, specifically). Second, because assigning genotypic outcomes to biochemical mechanisms is likely imperfect, we use machine learning methods to identify trends and patterns in genotype frequencies that refine this crude binning process.

In the first step of the inDelphi method, we separate genotypic outcomes into three classes: microhomology deletions (MH deletions), microhomology-less deletions (MH-less deletions), and single-base insertions (1-bp insertions) (Figure 1e). Below we outline the algorithmic definitions of each genotypic outcome class, the pathways associated with each class, and the DNA sequence parameters included in inDelphi training of each class. For more detailed technical algorithmic definitions of the genotypic outcome classes, see Supplementary Methods.

MH deletions are predicted from MH length, MH GC content, and deletion length

The majority of Cas9-mediated double-strand break repair genotypes we observe in our datasets are what we classify as MH deletions (53-58% in mESC, K562, HCT116, and HEK293). We hypothesize that these deletions occur through MMEJ-like processes and use known features of this pathways to inform a machine learning module to predict MH deletion outcomes. Following 5'-end resection as occurs in MMEJ, alt-NHEJ, and HDR[1], microhomologous basepairing of single-stranded DNA (ssDNA) sequences occurs across the border of the double strand breakpoint[3,4]. To restore a contiguous double-strand DNA chain, the 5'-overhangs not participating in the microhomology are removed up until the paired microhomology region, and the unpaired ssDNA sequences are extended by DNA polymerase using the opposing strand as a template (Figure 1d, Extended Data Fig. 2).

Assuming these same processes, inDelphi calculates the set of all MH deletions available given a specific sequence context and cleavage site.

As an example workflow, given the following sequence and its cleavage site:

```
ACGTG|CATGA
TGCAC|GTACT
```

for every possible deletion length from 1-bp to 60-bp deletions, we overlap the 3'-overhang downstream of the cut site under the upstream 3'-overhang and determine if there is any microhomologous basepairing. As an example, given the 4-bp deletion length:

```
ACGTG
 | ||
 GTACT
```

we see that there are three microhomologous basepairing events.

We then choose a particular microhomology (here, the highlighted C:G):

```
ACGTG
 | ||
 GTACT
```

then generate its unique repair genotype by following left-to-right along the top strand and jumping down to the complement of the bottom strand to simulate DNA polymerase fill-in.

Here, this yields:

```
ACATGA
TGTACT
```

This can also be displayed as an alignment. We note that by “jumping down” after the first base in the top strand, we can also describe this outcome using the delta-position 1. (See section on delta-positions). A deletion at delta-position 0 yields the same genotype.

```
Deletion a:   AC----ATGA
Wt:           ACGTGCATGA
```

This same sequence context and cleavage site could produce a distinct 4-bp MH deletion genotypic outcome through use of the TG:AC microhomology. This single outcome can be described as using delta-positions 2, 3, or 4. inDelphi uses only the single maximum delta-position (here, 4) to described a unique MH deletion.

```
Deletion b:   ACGTG----A
Wt:           ACGTGCATGA
```

Thus, there may be multiple MH deletion outcome genotypes for a given deletion length, and there is always a 1:1 mapping between the microhomologous basepairing used in that MH deletion and the resultant genotypic outcome. The set of MH deletions thus includes all 1-bp to 60-bp deletions that can be derived from the steps above that simulate the MMEJ mechanism.

MMEJ efficiency has been reported to depend on the thermodynamic favorability and stability of a candidate microhomology[3,4]. To parameterize MH deletions using the biochemical sequence features that influence this form of DNA repair, inDelphi calculates the MH length, MH GC content, and resulting deletion length for each possible MH deletion. These features are input into a machine learning module referred to in the Supplementary Methods as the microhomology neural network (MH-NN) to learn the relationship between these features and the frequency of an MH deletion outcome in a training CRISPR/Cas9 genotypic outcome dataset. While we predict and empirically find that favored MH deletions have long MH lengths relative to total deletion length and high MH GC-contents, we do not provide any explicit direction or comparative weighting to these parameters at the outset. inDelphi then outputs a phi-score for any MH deletion genotype (whether it was in the training data or not) that represents the favorability of that outcome as predicted by MH-NN.

It is important to emphasize that the phi-score of a particular MH deletion does not itself represent the likelihood of that MH deletion occurring in the context of all MH deletions at a given site. Some CRISPR/Cas9 target sites may have many possible favorable MH deletion outcomes while other sites have few, and thus phi-score must be normalized for a given target site to generate the fractional likelihood of that genotypic outcome at that site. Total unnormalized MH deletion phi-score is one factor that is further used to predict the relative frequency of the different repair classes: MH deletions, MH-less deletions, and insertions.

MH-less deletions are predicted from their length

We define MH-less deletions as all possible deletions that have not been accounted for by the workflow described above for MH deletions. Mechanistically, our data analysis suggests that MH deletions are associated with repair genotypes produced by c-NHEJ and microhomology-mediated alt-NHEJ pathways.

Following a double-strand break, c-NHEJ-associated proteins rapidly bind the DNA strands flanking the double-strand DNA breakpoint and recruit ligases, exonucleases, and polymerases to process and re-anneal the breakpoint in the absence of 5’-end resection (Extended Data Fig. 2)[1,5]. Commonly, c-NHEJ repair is error-free; however, in the context of Cas9-mediated cutting, faithful repair leads to repeated cutting, thereby increasing the eventual likelihood of mutagenic repair. Erroneous c-NHEJ repair products are mainly thought to consist of small insertions or deletions or combinations thereof that most frequently occur in the direct vicinity of the DNA break point[5–7]. The resulting deletions, which we refer to as medial end-joining MH-less deletions, have often lost bases both upstream and downstream of the cleavage site.

Microhomology-mediated alt-NHEJ is a distinct pathway that produces MH-less deletion products. In contrast to c-NHEJ, which is microhomology independent, this form of alt-NHEJ repair occurs following 5’-end resection and is mediated by microhomology in the sequence surrounding the double-strand break-point[1]. Microhomologous basepairing stabilizes the 3’-ssDNA overhangs following 5’-end resection, similarly to in MMEJ, allowing DNA ligases to join the break across one of the strands of this temporarily configured complex. The opposing un-annealed flap is then removed, and newly synthesized DNA templated off of the remaining strand is annealed to repair the lesion (Extended Data Fig. 2).

While alt-NHEJ uses microhomology, the repair products it produces do not follow the predictable genotypic patterns induced by MMEJ and are thus grouped into MH-less deletion genotypes. MH deletions are a direct merger of both annealed strands, in which the outcome genotype switches from top to bottom strand at the exact end-point of a microhomology. In contrast, while alt-NHEJ employs microhomology in its repair mechanism, the deletion outcomes it generates comprise bases exclusively derived from either the top or bottom strand. Mechanistically, this occurs because ligation of a 3’-overhang to its downstream ligation partner results in removal of the entire opposing ssDNA overhang up until the point of ligation. This process prevents any deletion from occurring in the 3’-overhang strand that is first attached to the DNA backbone, while inducing loss of an indeterminant length of sequence on the opposing strand. The resulting deletion genotypes, which we refer to as unilateral end-joining MH-less deletions, do not retain information on the exact microhomology causal to their occurrence, and are thus also referred to as MH-less.

Consequently, the various mechanisms that give rise to MH-less deletions are capable of generating a vast number of genotypic outcomes for any given deletion length. Having less information on the biochemical mechanisms that impact the relative frequency of

NHEJ deletion products, inDelphi models these deletions without assuming any particular mechanism.

inDelphi detects MH-less deletions from training data as the set of all deletions that are not MH deletions and parameterizes them solely by the length of the resulting deletion. This is based on the simple assumption that c-NHEJ and alt-NHEJ processes are most likely to produce short deletions, supported by our empirical observation. As with MH deletions, this assumption is not explicitly coded into the inDelphi MH-less deletion prediction module, instead allowing it to be “learned” by a neural network called MHless-NN.

MHless-NN optimizes a phi-score for a given MH-less deletion length, grounded in the frequency of MH-less deletion outcomes of that length observed in the training data. We observe that MHless-NN learns a near-exponential decaying phi-score for increasing deletion length, that reflects the sum total frequency of all MH-less deletion genotypes. The total unnormalized MH-less deletion phi-score for a given target and cut site is also employed to inform the relative frequency of different repair classes.

1-bp insertions are predicted from sequence context and deletion phi-scores

Lastly, inDelphi predicts 1-bp insertions from both the broader sequence context and the immediate vicinity of the cleavage site. We empirically find that 1-bp insertions are far more common than longer insertions, so we focus on their prediction. It is classically assumed that short sequence insertions are the result of c-NHEJ[6,7], however, little else is known about their biochemical mechanism as it pertains to local sequence context to help inform prediction. Nonetheless, we find powerful correlations between the identities of the bases surrounding the Cas9 cleavage site and the frequency and identity of the inserted base (see main text). Motivated by these empirical observations, inDelphi is fed with training data on 1-bp insertion frequencies and identities at each training site parameterized with the identities of the -3, -4, and -5 bases upstream of the NGG PAM-sequence (when the training set is sufficiently large, and the -4 base alone when training data is limited) as features. Also added as features are the precision score of the deletion length distribution and the total deletion phi-score at that site. These features are combined into a *k*-nearest neighbor algorithm that predicts the relative frequencies and identities of 1-bp insertion products at any target site.

The combination of the MH, MH-less, and insertion model predict genotype fractions

Altogether, informed by known paradigms of DNA repair, we build 2 neural networks and a *k*-nearest neighbor model to predict genotypic outcomes following Cas9 cutting. These models compete and collaborate in inDelphi to generate predictions of the relative frequencies of these products. This competition within inDelphi among repair types reflects empirical evidence from Lib-A and Lib-B that sequence contexts do influence classes of repair outcomes. Sequence contexts with high phi scores (high microhomology) have higher efficiencies of MH deletions among all editing outcomes (Figure 2d, Extended Data Fig. 3), and sequence contexts with low phi scores (low microhomology) have higher efficiencies of 1-bp insertions among all editing outcomes

(Figure 2d, Extended Data Fig. 3). While it is tempting to generalize that the competition and collaboration among outcome classes modeled by inDelphi reflects interactions among components of distinct DNA repair pathways, the classes of outcomes considered by inDelphi do not necessarily arise from distinct DNA repair pathways as they are described above. inDelphi is trained on the repair outcomes only and cannot distinguish between the nature of genotypes when they may occur through MH-mediated and MH-less mechanisms, and it is imaginable that some repair products result through more than one repair pathway.

As an additional note, while NHEJ is generally assumed to dominate double-strand break repair from environmentally induced damage[5], we find in the context of Cas9 cutting that MH deletion genotypes are more common than MH-less deletions and insertions. It is possible that error-free c-NHEJ is occurring frequently in response to Cas9 cutting but that its perfect repair allows for recurring Cas9 cutting that goes undetected by our workflow, thus skewing the observed relative frequency profile of mutagenic outcomes toward MMEJ-type repair.

Rarer CRISPR-Cas9 outcomes

Our library assay and workflow involved data processing of high-throughput sequencing data using sequence alignments and a designed procedure for categorizing sequence alignments into categories of CRISPR-related outcomes. Beyond simple deletions and insertions, we identified other rarer outcomes that were explained as indels caused by CRISPR, such as combination insertion/deletions involving and/or near the cleavage site (0.5-2% of all products) and indels near but not immediately at the cleavage site (3-5% of all products), which occurred more often on the PAM-distal side of the double-strand break (data not shown). Our library assay is unable to observe events that occur outside of our high-throughput sequencing window.

Default sequence alignment procedures can generate sequence alignments involving simple CRISPR-caused deletions and insertions that do not occur immediately at the cleavage site, but that can be transformed into an equal-scoring sequence alignment where the indel does occur immediately at the cleavage site. This straightforward processing step is not performed by the most common bioinformatic tools for sequence alignment, since they were not expressly designed for CRISPR. We note here that our sequence alignment procedure takes this into account (see Supplementary Methods for more detailed description). This attention to detail enables us to accurately identify simple indels that occur near but not immediately at the cleavage site. We observe that the frequency of these indels across target sites correlates significantly with the total on-target editing efficiency (measured by the frequency of non-wild-type outcomes out of all non-noise outcomes) at these target sites in HEK293 and mES cells. We also observe significantly higher frequencies in postCas9 treatment conditions than preCas9 control conditions. Together, these observations suggest that these indels are caused by CRISPR editing.

Prkdc$^{–/–}$Lig4$^{–/–}$ **mutants have distinct and predictable DNA repair product distributions**

While it is generally true that our work cannot establish roles for specific DNA repair pathways in specific types of Cas9-mediated outcomes, we have performed an experiment in which we measure Cas9-mediated genotypic outcomes from mESCs that are lacking *Prkdc* and *Lig4*, two proteins known to be key in c-NHEJ[5]. We find an increase in relative frequency of MH deletions as compared to MH-less deletions in *Prkdc*$^{–/–}$*Lig4*$^{–/–}$ mESCs as compared to wild-type mESCs (see main text), which is suggestive of an increase in MMEJ outcomes at the expense of NHEJ outcomes.

Intriguingly, we also find that *Prkdc*$^{–/–}$*Lig4*$^{–/–}$ mESCs are impaired in unilateral deletions, where only bases from one side of the cutsite are removed, but not medial MH-less deletion outcomes that have loss of bases on both sides of the breakpoint. (Extended Data Fig. 6). As discussed earlier, microhomology-mediated alt-NHEJ, which we hypothesize may give rise to unilateral MH-less deletions, proceeds through a mechanism in which DNA repair intermediates that mimic MMEJ-mediated repair are formed initially (Extended Data Fig. 2), as microhomology base-pairing temporarily stabilizes 3'-overhangs following 5'-end resection. Subsequently, ligation joins one 3' overhang with the sequence on the other side of the DNA double-strand break, giving rise to a unilateral deletion. If the unilateral joining products we observe in our experiments indeed arise through similar mechanisms as those described by this form of alt-NHEJ, it is conceivable that the MMEJ pathway may overtake 3'-end ligation at this microhomology-containing intermediate step when ligation is impaired through loss of Lig4. Thus, cross-talk of microhomology-mediated repair pathways could account for loss of unilateral end-joining MH-less outcomes and concomitant increase in MH deletion outcomes. Medial joining outcomes are not hypothesized to originate from intermediates that overlap with microhomology-mediated deletion products (Extended Data Fig. 2). Therefore, the repair genotypes generated via this orthogonal pathway may be afforded more time to be completed by ligases other than Lig4, thus explaining why these outcomes appear unaffected by NHEJ impairment.

While DNA repair products in *Prkdc*$^{-/-}$*Lig4*$^{-/-}$ mESCs differ substantially from those in wild-type cells, we find that these DNA repair products are also highly predictable. In particular, inDelphi performed well on held-out *Prkdc*$^{-/-}$*Lig4*$^{-/-}$ data when trained on *Prkdc*$^{-/-}$*Lig4*$^{-/-}$ data (indel genotype prediction median Pearson correlation = 0.84, indel length frequency prediction Pearson correlation = 0.80), showing that our modeling approach is robustly capable of learning accurate predictions for Cas9 editing data in not just wild-type experimental settings but also settings with significant biochemical perturbation. As such, we suggest here that inDelphi's modeling approach can be useful on additional tasks unexplored here provided that inDelphi is supplied with appropriate training data.

NU7041, DPKi3, and MLN4924 induce a distinct DNA repair product distribution

We further investigated the role of DNA repair pathways by three separate experiments involving HTS characterization of Lib-B in mESCs treated with three separate small molecules: NU7041, a DNA dependent protein kinase (DNA-PK) inhibitor; DPKi3, another DNA-PK inhibitor, and MLN4924, a NEDD8-activating enzyme (NAE) inhibitor. DNA-PK and NAE are proteins involved in c-NHEJ[5,8].

MLN4924 is thought to inhibit the release of the Ku70/Ku80 heterodimer following proper c-NHEJ repair, potentially disrupting downstream processes such as transcription and replication, which may lead to decreased cell survival and a depletion of Ku70/Ku80-dependent DNA repair genotypes in a population.[8]

DNA-PK is commonly recruited to DSBs during c-NHEJ and is known to phosphorylate *in vitro* many c-NHEJ-related factors including Ku70/80, XRCC4, DNA Ligase IV, Artemis, H2AX, p53, and itself. Inhibition of DNA-PK leads to DNA repair defects[5]. The catalytic subunit of DNA-PK is encoded by the Prkdc gene, which was knocked out in Prkdc$^{-/-}$Lig4$^{-/-}$ cells.

From HTS data, we observed that the frequency of MH deletions among all deletions clustered into three approximate groups: wild-type (median 77%) and MLN4924, then DPKi3 and NU7041 (median 81%), and lastly Prkdc$^{-/-}$Lig4$^{-/-}$ (median 90%) (Extended Data Fig. 6). These data suggest that impairing DNA-PK (via DPKi3, NU7041 and Prkdc$^{-/-}$) yields a moderate 17% reduction in the frequency of MH-less deletions (23% to 19%). This reduction appears to be non-redundant with knockout of Lig4 evidenced in Prkdc$^{-/-}$Lig4$^{-/-}$ cells with a 57% reduction (23% to 10%) in MH-less frequency. Lastly, impairing NAE did not have a significant impact on the frequency of MH-less deletions.

We observed an overall increased frequency of repair to wild-type at pathogenic microduplication alleles after treatment with DPKi3, MLN4924, and NU7041 (Extended Data Fig. 6). Along with Prkdc$^{-/-}$Lig4$^{-/-}$ cells, the change in repair efficiency was associated with deletion length ($p < 2.2 \times 10^{-3}$), with decreased efficiency compared to wild-type at short deletion lengths and increased efficiency at longer deletion lengths.

The change in repair efficiency caused by separate treatments of DPKi3, MLN4924, and NU7041 was highly consistent across different target sites (r = 0.73, 0.77, and 0.81, Extended Data Fig. 6). This is surprising since MLN4924 inhibits a different target than DPKi3 and NU7041. We observed a similar but weaker relationship between the three small molecules and *Prkdc$^{-/-}$Lig4$^{-/-}$*, with Pearson correlations of 0.09, 0.16, and 0.18. Taken together, these observations suggest a relationship between DNA sequence and the propensity of DNA repair outcomes through c-NHEJ.

In DPKi3, MLN4924, and NU7041 treated cells, the decrease in MH-less deletions primarily occurs medial joining products (Extended Data Fig. 6), suggesting that DNA-PK is a strong contributor to medial joining products. However, when both DNA-PK and Lig4 are knocked out in *Prkdc$^{-/-}$Lig4$^{-/-}$* cells, the average frequency of medial joining products is not significantly changed, and instead the primary decrease occurs in unilateral joining products.

Interestingly, MLN4924 increases the average frequency of unilateral joining events. Combined with its effect of decreasing medial joining products, the overall net effect of MMLN4924 is an absence of significant change to the frequency of MH-less deletions.

The frequency distribution of medial joining products in *Prkdc*$^{-/-}$*Lig4*$^{-/-}$ reveals a decrease in median frequency in combination with an inflation in high frequency outliers (target sites where >80% of all deletion products are MHless medial products) which skews the distribution's average to be above the median. Taken together, these data confirm that both medial and unilateral products are both generally depleted in *Prkdc*$^{-/-}$*Lig4*$^{-/-}$ cells, and suggest that knocking out DNA-PK depletes medial MHless products while knocking out Lig4 depletes unilateral MHless products.

Supplementary Methods

Library cloning protocol

Synthesized oligo library sequence

GATGGGTGCGACGCGTCAT[55bpTarget]AGATCGGAAGAGCACACGTCTG**AATATT**GTGGAAAGGACGAAACACCG[19/20-nt PROTOSPACER depending on whether it naturally starts with a G]GTTTAAGAGCTATGCTGGAAACAGC

Linker region / Oligo library amplification primer anneal region
Read 2 sequencing primer stub
SspI restriction site
U6-promoter stub
sgRNA-hairpin stub

1. Oligo library QPCR to determine number of amplification cycles for Oligo Library PCR

Notes: Amplification of oligos with relatively low GC-content is less efficient than GC-rich sequences. We found NEBNext polymerase to be the least biased in amplification of our library. Increasing the elongation time to 1 min per cycle for all cloning and sequencing library prep PCRs eliminates GC-skewing of library sequences and reduces the rate of PCR-recombination.

- Set up the following reaction:

0.4 ng	Synthesized Oligo Library
10 ul	NEBNext 2x Master Mix
0.5 ul	20uM **OligoLib_Fw**
0.5 ul	20uM **OligoLib_Rv**
0.2 ul	SybrGreen Dye (100x)
to 20 ul	H_2O

67°C annealing temperature

- Check 246bp amplicon size on 2.5% agarose gel.
- Determine the point that signal amplification has plateaued.

2. Oligo Library PCR amplification

- Set up the following reaction:

4 ng	Synthesized Oligo Library
50 ul	NEBNext 2x Master Mix
2.5 ul	20uM **OligoLib_Fw**
2.5 ul	20uM **OligoLib_Rv**
to 100 ul	H_2O

67°C annealing temperature, 1 minute extension time.
Cycle number is half the number of cycles needed to reach signal amplification plateau in the QPCR in step 1, reduced by 1 cycle to scale for DNA input.

- PCR purify amplified sequence.

3. Donor template amplification

- Set up the following reaction:

5 ng	spCas9 sgRNA plasmid *(71485)*
50 ul	NEBNext 2x Master Mix
2.5 ul	20uM **CircDonor_Fw**
2.5 ul	20uM **CircDonor_Rv**
to 100 ul	H_2O

62°C annealing temperature
20 cycles

- Gel purify 167bp band from 2.5% agarose gel.

4. Circular assembly and restriction digest linearization

Note: We use a molar ratio of donor template to amplified oligo library of 3:1. An increase in amplified oligo library compounds cross-over within library members resulting in mismatch of protospacer and target sequences.

- Set up the following reaction:

429 ng	Donor template
239 ng	Amplified Oligo Library
30 ul	Gibson Assembly 2x Master Mix
to 60 ul	H_2O

50°C incubation for 1 hour.

- Exonuclease treatment

60 ul	Circular assembly reaction
9 ul	ATP (25mM)
9 ul	10x Plasmid Safe Buffer
3 ul	Plasmid Safe Nuclease
9 ul	H_2O

37°C incubation for 1 hour.

- PCR purify and elute in 50 ul.

- Digest to linearize library

50 ul	Purified assemblies
10 ul	10x CutSmart Buffer
3 ul	SspI-HF
37 ul	H_2O

37°C incubation for ≥ 3 hours.

- Gel purify 273bp band from 2.5% agarose gel.

Note: Band is sometimes fuzzy and poorly visible. If not clearly discernible, proceed with gel isolation between 200-300bp.

5. Linearized library QPCR to determine number of amplification cycles for PCR amplification

- Set up the following reaction:

0.5 %	Purified linearized library
10 ul	NEBNext 2x Master Mix
0.5 ul	20uM **PlasmidIns_Fw**
0.5 ul	20uM **PlasmidIns_Rv**
0.2 ul	SybrGreen Dye (100x)
to 20 ul	H_2O

65°C annealing temperature

- Determine the point that signal amplification has plateaued.

6. Linearized Library PCR amplification

- Set up the following reaction:

50 %	Purified linearized library
50 ul	NEBNext 2x Master Mix
2.5 ul	20uM **PlasmidIns_Fw**
2.5 ul	20uM **PlasmidIns_Rv**
to 100 ul	H_2O

65°C annealing temperature, 1 minute extension time.
Cycle number is number of cycles needed to reach signal amplification plateau in the QPCR in step 5, reduced by 4 cycles to scale for increased DNA input.

- Gel purify 375bp band from 2.5% agarose gel.

7. Vector backbone digest

- Set up the following reaction:

2 ug	spCas9 sgRNA plasmid *(71485)*
10 ul	10x Buffer 2.1
3 ul	BbsI
2 ul	XbaI
to 100 ul	H_2O

37°C incubation for ≥ 3 hours.

- Gel purify 5.9 kb band from 1% agarose gel.

8. Vector assembly and cleanup

Note: Include a ligation with water for insert as a control.

- Set up the following reaction:

300 ng	Digested vector backbone
42 ng	Amplified Oligo Library
30 ul	Gibson Assembly 2x Master Mix
to 60 ul	H_2O

50°C incubation for 1 hour.

- Isopropanol precipitation

40 ul	Vector assembly reaction
0.4 ul	GlycoBlue Coprecipitant
0.8 ul	50mM NaCl
38.8 ul	Isopropanol

- Vortex and incubate at room temperature for 15 minutes.
- Spin down at ≥15.000g for 15 minutes, and carefully remove supernatant.
- Wash pellet with 300ul 80% EtOH and repeat spin at ≥15.000g for 5 minutes.
- Carefully remove all liquid without disturbing pellet, and let air dry for 1-3 minutes.
- Dissolve dried pellet in 10 ul H_2O at 55°C for 10 minutes.

9. Transformation

Note: Electroporation competent cells give a higher transformation efficiency than chemically competent cells. We use NEB10beta electro-competent cells, however these can be substituted for other lines and transformed according to the manufacturer's instructions.

Note: We use DRM as recovery and culture medium to enhance yield. If substituting for a less rich medium such as LB, we recommend scaling up the culture volume to obtain similar plasmid DNA quantities.

Note: Antibiotic-free recovery time should be limited to 15 minutes to prevent shedding of transformed plasmids from replicating bacteria.

Note: Also transform water ligation as control.

- Pre-warm 3.5mL recovery medium per electroporation reaction, at 37°C for 1 hour.
- Pre-warm LB-agar plates containing appropriate antibiotic.
- Per reaction, add 1 ul purified vector assembly to 25ul competent cells on ice. Perform 8 replicate reactions.
- Electroporate according to the manufacturer's instructions.
- Immediately add 100 ul pre-warmed recovery media per cuvette and pool all replicates into culture flask.
- Add 1 mL recovery media per replicate reaction to culture flask and shake at 200rpm 37°C for 10 – 15 minutes.
- Plate a dilution series from 1:10^4 – 1:10^6 on LB-agar plates containing antibiotic and grow overnight at 37°C
- Add 2 mL media per replicate reaction and admix appropriate antibiotic.
- Grow overnight in shaking incubator at 200rpm 37°C

- Assess transformation efficiency from serial dilution LB-agar plates. Expect ~10^6 clones.

The development of this cloning protocol was guided by work described in Videgal et al. 2015.

Sequence alignment and data processing

For library data, each sequenced pair of gRNA fragment and target was associated with a set of designed sequence contexts *G* by finding the designed sequence contexts for all gRNAs whose beginning section perfectly matches the gRNA fragment (read 1 in general does not fully sequence the gRNA), and by using locality sensitive hashing (LSH) with 7-mers on the sequenced target to search for similar designed targets. An LSH score on 7-mers between a reference and a sequenced context reflects the number of shared 7-mers between the two. If the best reference candidate scored, through LSH, greater than 5 higher than the best LSH score of the reference candidates obtained from the gRNA-fragment, the LSH candidate is also added to *G*. LSH was used due to extensive (~33% rate) PCR recombination between read1 and read2 which in sequenced data appears as mismatched read1 and read2 pairs. The sequenced target was aligned to each candidate in *G* and the alignment with the highest number of matches is kept. Sequence alignment was performed using the Needleman-Wunsch algorithm using the parameters: +1 match, -1 mismatch, -5 gap open, -0 gap extend. For library data, starting gaps cost 0. For all other data, starting and ending gaps cost 0. For VO data, sequence alignments were derived from SAM files from SRA.

Alignments with low-accuracy or short matching sections flanked by long (10 bp+) insertions and deletions were filtered out as PCR recombination products (observed frequency of ~5%). These PCR recombination products are different than that occurring between read1 and read2; these occur strictly in read2. Alignments with low matching rates were removed. Deletions and insertions were shifted towards the expected cleavage site while preserving total alignment score. CRISPR-associated DNA repair events were defined as any alignment with deletions or insertions occurring within a 4 bp window centered at the expected cut site and any alignment with both deletions and insertions (combination indel) occurring with a 10 bp window centered at the expected cut site. All CRISPR-associated DNA repair events observed in control data had their frequencies subtracted from treatment data to a minimum of 0.

We carried out replicate experiments for library data in each cell type. For each cell-type, each target site not fulfilling the following data quality criteria was filtered: in each replicate, data at this target site must have a total of at least 1,000 reads for all CRISPR editing outcomes at that target site (see section on “Calling CRISPR editing outcomes with high confidence” below for a discussion on the 1,000 reads threshold), and a Pearson correlation of at least 0.85 in the frequency of microhomology-based deletion events. The class of microhomology-based deletion events was used for this criterion since it is a major repair class with the highest average replicability across experiments.

Details on alignment processing

All alignments with gaps were shifted as much as possible towards the cleavage site while preserving the overall alignment score. Then, the following criteria were used to categorize the alignments into noise, not-noise but not CRISPR-associated (for example, wildtype); as well as primary and secondary CRISPR activity. All data used in modeling and analysis derive solely from outcomes binned into primary CRISPR activity.

The following criteria was used to filter library alignments into “noise” categories.

Homopolymer: Entire read is homopolymer of a single nucleotide. Not considered a CRISPR repair product.
Has N: Read contains at least one N. Discarded as noise, not considered a CRISPR repair product.
PCR Recombination: Contains recombination alignment signature: (1) if a long indel (10 bp+) followed by chance overlap followed by long indel (10 bp+) of the opposite type, e.g., insertion-randommatch-deletion and deletion-randommatch-insertion. OR, if one of these two indels is 30 bp+, the other can be arbitrarily short. If either criteria is true, and if the chance overlap is length 5 or less, or any length with less than 80% match rate, then it satisfies the recombination signature. In addition, if both indels are 30 bp+, regardless of the middle match region, it satisfies the recombination signature. Finally, if randommatch is length 0, then indel is allowed to be any length. Not considered a CRISPR repair product.
Poor-Matches: 55bp designed sequence context has less than 5 bp representation (could occur from 50 bp+ deletions or severe recombination) or less than 80% match rate. Not considered a CRISPR repair product.
Cutsite-Not-Sequenced: The read does not contain the expected cleavage site.
Other: An alignment with multiple indels where at least one non-gap region has lower than an 80% match rate. Or generally, any alignment not matching any defined category above or below. In practice, can include near-homopolymers. Not considered a CRISPR repair product.

The following criteria was used to filter library alignments into “main” categories.
Wildtype: No indels in all of alignment. Not considered a CRISPR repair product.

Deletion: An alignment with only a single deletion event. Subdivided into:
Deletion - Not CRISPR: Single deletion occurs outside of 4 bp window centered around cleavage site. Not considered a CRISPR repair product.
Deletion - Not at cut: Single deletion occurring within 4 bp window centered around cleavage site, but not immediately at cleavage site. Considered a CRISPR repair product.
Deletion: Single deletion occurring immediately at cleavage site. Considered a CRISPR repair product.

Insertion: An alignment with only a single insertion event. Subdivided into:
Insertion - Not CRISPR: Single insertion occurs outside of 10 bp window around cleavage site. Not considered a CRISPR repair product.
Insertion - Not at cut: Single insertion occurring within 4 bp window centered around cleavage site, but not immediately at cleavage site. Considered a CRISPR repair product.
Insertion: Single insertion occurring immediately at cleavage site. Considered a CRISPR repair product.

Combination indel: An alignment with multiple indels where all non-gap regions have at least 80% match rate. Subdivided into:

Combination Indel: All indels are within a 10 bp window around the cleavage site. Considered a primary CRISPR repair product.
Forgiven Combination Indel: At least two indels, but not all, are within a 10 bp window around the cleavage site. Considered a rarer secondary CRISPR repair product, ignored.
Forgiven Single Indel: Exactly one indel is within a 10 bp window around the cleavage site. Considered a rarer secondary CRISPR repair product, ignored.
Combination Indel - Not CRISPR: No indels are within a 10 bp window around the cleavage site. Not considered a CRISPR repair product.

We note that deletion and insertion events, even those spanning many bases, are defined to occur at a single location between bases. As such, events occurring up to 5 bp away from the cleavage site are defined as events where there are five or fewer matched/mismatched alignment positions between the event and the cleavage site, irrespective of the number of gap dashes in the alignment.

Calling CRISPR editing outcomes with high confidence

Following the processing steps above, we performed the following further analysis and processing steps to call high-confidence CRISPR editing outcomes. These steps largely follow heuristics, and we believe that a thorough and unbiased methodological standardization in counting CRISPR editing outcomes will be valuable future work.

DNA repair at Cas9-mediated double-strand breaks is known to result in a large diversity of outcomes, with indels of varying length and positions around the cleavage site. The frequencies of many of these editing outcomes, though enriched in Cas9-treatment data over control data, are rare (<0.5% of all edited products) and can be challenging to assign as a CRISPR editing outcome due to a lack of foundational biological or computational models on the exact mechanisms of DNA repair. In addition, rare outcomes can sometimes be attributed to sequencing errors.

In this work, we focus on CRISPR editing outcomes that are enriched in treatment data over control data and agree with a relatively conservative and strict model of DNA repair, in order to ensure a high degree of confidence in the editing outcomes that we call. As a result, we underestimate the total number of unique CRISPR editing outcomes, though we believe this underestimation is not by an order of magnitude, though it may be by a factor of 2x or so.

We define high-confidence CRISPR editing outcomes as bins of alignments categorized by the previously described pipeline into CRISPR-associated categories that have no mismatches. Each unique deletion genotype consistent with microhomology is treated as a single unique outcome, though we note that microhomology deletions may arise by noise or chance though we expect this to be a rare event. Each unique insertion genotype is also treated as a single unique outcome, and as with MH deletions, we note that some insertions may arise by noise or chance though we anticipate this to be rare. In sum, we likely overestimate by a slight amount the true number of unique microhomology deletion and insertion events.

All microhomology-less deletion genotypes are binned together for a particular deletion length, which almost always will bin together multiple unique MH-less deletion genotypes. However, the class of MH-less deletions, in general, has lower replicate consistency and higher stochasticity than MH deletions, and the space of all possible MH-less deletions is orders of magnitude larger than that of MH deletions. The class of MH-less deletions is also less frequent than MH deletions in all five human and mouse cell types we examined. In sum, we characterize MH-less deletions as comprising a large number of rare genotypes that lack high replicate consistency. As such, we conservatively count all binned MH-less deletions for a particular deletion length as a single unique outcome. In sum, we likely underestimate by a moderate amount the true number of unique microhomology-less deletion events.

As MH-less deletions represents the larger space of possible unique genotypes, in total, we are likely underestimating the total number of unique outcomes in our procedure that calls unique outcomes with high confidence. We provide statistics on the total number of high-confidence unique outcomes in the manuscript and in Extended Data Fig. 1 and 5.

Based on a computational simulation of subsampling the data, we empirically set 1,000 reads per target site as a minimum quality threshold. The diversity of editing outcomes requires some minimum read count to consider the data as representative of editing outcomes at that target site. In Lib-A and Lib-B data in U2OS and mESCs, we empirically observe that 1,000 reads per target site lies above the “elbow” in the curve plotting the number of unique high-confidence outcomes and subsampled read count. We recommend this quality filtering methodology in general for future work studying CRISPR editing outcomes, and based on our data, empirically suggest that 1,000 reads per target site may be a useful guideline for future experimental design.

Controlling for cell-type specific 1-bp insertion frequencies when measuring replicability of indel frequencies across cell-types

All indels not belonging to “all major repair outcomes” were filtered out. To adjust the frequencies of all 1-bp insertion genotypes in a target site in two cell-types, the average of the total 1-bp insertion frequency among all major repair outcomes was calculated between the two cell-types, then frequencies of each 1-bp insertion genotype was adjusted proportionally such that the resulting total 1-bp insertion frequency in that is equal to the aforementioned average, and thereby equal to the adjusted 1-bp insertion frequency in the data from the other cell-type.

Selection of variants from disease databases

Disease variants were selected from the NCBI ClinVar database (downloaded September 9, 2017)[9] and the Human Gene Mutation Database (publicly available variant data from before 2014.3)[10] for computational screening and subsequent experimental correction.

A total of 4,935 unique variants were selected from Clinvar submissions where the functional consequence is described as complete insertions, deletions, or duplications where the reference or alternate allele is of length less than or equal to 30 nucleotides. Variants were included where at least one submitting lab designated the clinical significance as ‘pathogenic’ or ‘likely pathogenic’ and no submitting lab had designated the variant as ‘benign’ or ‘likely benign’, including variants will all disease associations. More complex indels and somatic variants were included. A total of 18,083 unique insertion variants were selected from HGMD which were between 2 to 30 nucleotides in length. Variants were included with any disease association with the HGMD classification of ‘DM’ or disease-causing mutation.

SpCas9 gRNAs and their cleavage sites were enumerated for each disease allele. Using a previous version of inDelphi, genotype frequency and indel length distributions were predicted for each tuple of disease variant and unique cleavage site. Among each unique disease, the single best gRNA was identified as the gRNA inducing the highest predicted frequency of repair to wildtype genotype, and if this was impossible (due to, for example, a disease allele with 2+ bp deletion), then the single best gRNA was identified as the gRNA inducing the highest predicted frameshift repair rate. 1327 sequence contexts were designed in this manner for Lib-B. An additional 265 sequence contexts were designed by taking the 265 sequence contexts in any disease in decreasing order of predicted wildtype repair rate, starting with Clinvar, stopping at 45% wildtype repair rate, then continuing with HGMD. This yielded 1592 total sequences derived from Clinvar and HGMD.

Definition of Delta-Positions

Using the MMEJ mechanism, deletion events can be predicted at single-base resolution. For computational convenience, we use the tuple (deletion length, delta-position) to construct a unique identifier for deletion genotypes. A delta-position associated with a deletion length N is an integer between 0 and N inclusive (Extended Data Fig. 2). In a sequence alignment, a delta-position describes the starting position of the deletion gap in the read with respect to the reference sequence relative to the cleavage site. For a deletion length N and a cleavage site at position C such that seq[:C] and seq[C:] yield the expected DSB products where the vector slicing operation vector[index1:index2] is inclusive on the first index and exclusive on the second index (python style), a delta-position of 0 corresponds to a deletion gap at seq[C-N+0 : C+0], and generally with a delta-position of D, the deletion gap occurs at seq[C-N+D : C+D]. Microhomologies can be described with multiple delta-positions. To uniquely identify microhomology-based deletion genotypes, the single maximum delta-position in the redundant set is used. Microhomology-less deletion genotypes are associated with only a single delta position and deletion length tuple; we use this as its unique identifier.

Another way to define delta-positions can be motivated by the example workflow in the Supplementary Discussion on MH deletions describing how each microhomology is associated with a deletion genotype. In that workflow, the delta-position is the number of bases included on the top strand before “jumping down” to the bottom strand.

MH-less medial end-joining products correspond to all MH-less genotypes with delta-position between 1 and N-1 where N is the deletion length. MH-less unilateral end-joining products correspond to MH-less genotypes with delta-position 0 or N. We note that a deletion genotype with delta position N does not immediately imply that it is a microhomology-less unilateral end-joining product since it may contain microhomology (it’s possible that delta-positions N-j, N-j+1, .., N all correspond to the same MH deletion.)

Definition of Precision Score

For a distribution X, where |X| indicates its cardinality (or length when represented as a vector):

$$PrecisionScore(X) = 1 - \frac{-\sum_{i=1}^{n} P(x_i)\log(P(x_i))}{\log(|X|)}$$

This precision score ranges between zero (minimally precise, or highest entropy) to one (maximally precise, or lowest entropy).

inDelphi Deletion Modeling: Neural network input and output

inDelphi receives as input a sequence context and a cleavage site location, and outputs two objects: a frequency distribution on deletion genotypes, and a frequency distribution on deletion lengths.

To model deletions, inDelphi trains two neural networks: MH-NN and MHless-NN. MH-NN receives as input a microhomology that is described by two features: microhomology length and GC fraction in the microhomology. Using these features, MH-NN outputs a number (psi). MHless-NN receives as input the deletion length. Using this feature, MHless-NN outputs a number (psi).

A phi score is obtained from a psi score using: phi_i = exp(psi_i – 0.25*deletion_length), where 0.25 is a “redundant” hyperparameter that serves to increase training speed by helpful scaling. This relationship between psi and phi is differentiable and encodes the assumption that the frequency of an event exponentially increases with neural network output psi (which empirically appears to reflect MH strength) and exponentially decreases with its minimum necessary resection length (deletion length).

The architecture of the MH-NN and MHless-NN networks are *input-dimension -> 16 -> 16 -> 1* for a total of two hidden layers where all nodes are fully connected. Sigmoidal activations are used in all layers except the output layer. All neural network parameters are initialized with Gaussian noise centered around 0. No regularization or dropout was used.

inDelphi Deletion Modeling: Making predictions

Given a sequence context and cleavage site, inDelphi enumerates all unique deletion genotypes as a tuple of its deletion length and its delta-position for deletion lengths from 1 bp to 60 bp. For each microhomology enumerated, an MH-phi score is obtained using MH-NN. In addition, for each deletion length from 1 bp to 60 bp, an MHindep-phi score is obtained using MHless-NN.

inDelphi combines all MH-phi and MHindep-phi scores for a particular sequence context into two objects – a frequency distribution on deletion genotypes, and a frequency distribution on deletion lengths – which are both compared to observations for training. The model is designed to output two separate objects because both are of biological interest, and separate but intertwined modeling approaches are useful for generating both. By learning to generate both objects, inDelphi jointly learns about microhomology-based deletion repair and microhomology-less deletion repair.

To generate a frequency distribution on deletion genotypes, inDelphi assigns a score for each microhomology. Score assignment considers the concept of “full” microhomology and treats full and not full MHs differently.

A microhomology is “full” if the length of the microhomology is equal to its deletion length. The biological significance of full microhomologies is that there is only a single deletion genotype possible for the entire deletion length, while in general, a single deletion length is consistent with multiple genotypes. In addition, this single genotype can be generated through not just the MH-dependent MMEJ mechanism but also through MH-less end-joining, for example as mediated by Lig4. Therefore, we model full microhomologies as receiving contributions from both MH-containing and MH-less mechanisms by scoring full microhomologies as MH-phi[i] + MHindep-phi[j] for deletion length *j* and microhomology index *i.*

Microhomologies that are not “full” are assigned a score of MH-phi[i] for MH index *i*.

Scores for all deletion genotypes assigned this way are normalized to sum to 1 to produce a predicted frequency distribution on deletion genotypes.

To generate a frequency distribution on deletion lengths, inDelphi assigns a score for each deletion length. Score assignment integrates contributions from both MH-dependent and MH-independent mechanisms via the following procedure: For each deletion length *j*, its score is assigned as MHindep-phi[j] plus the sum of MH-phi for each microhomology with that deletion length. Scores for all deletion lengths are normalized to sum to 1 to produce a frequency distribution.

inDelphi trains its parameters using a single sequence context by producing both a predicted frequency distribution on deletion genotypes and deletion lengths and minimizing the negative of the sum of two values: the mean squared Pearson correlation for the deletion genotype frequency distribution at each target site in the training set plus the mean squared Pearson correlation for the deletion length frequency distribution at each target site in the training set. This represents a multitask learning framework.

In practice, deletion genotype frequency distributions are formed from observations for deletion lengths 1-60, and deletion length frequency distributions are formed from observations for deletion lengths 1-28. Both neural networks are trained jointly and simultaneously on both tasks. inDelphi is trained with stochastic gradient descent with batched training sets. inDelphi is implemented in Python using the autograd library. We used a batch size of 200, an initial weight scaling factor of 0.10, an initial step size of 0.10, and an exponential decaying factor for the step size of 0.999 per step. We observed performance convergence within about 50 epochs.

inDelphi Deletion Modeling: Multitask learning improves performance

Over the course of developing our model, at an intermediate stage we considered a simpler model for predicting the frequencies of MH deletions. This model featurizes all sequence microhomologies at a target site using MH length and GC content and uses a 2x16x16x1 neural network with sigmoidal activations except at the output layer to output psi. This psi value is adjusted using exp(psi – 0.25 * deletion length) to obtain phi for a particular microhomology, which are normalized across all microhomologies to sum to 1 to achieve a predicted distribution of frequencies. Altogether, this model is identical to the MH module used in inDelphi, with the notable difference of not including contributions of MH-less phi at “full” microhomologies.

This simple model, henceforth known as the baseline model, does not recognize the possibility that a MH genotype may arise from both MH and MH-independent repair pathways. We compared the performance of the baseline model to inDelphi’s MH module and observed a statistically significant relative improvement of 10% in model performance as measured on test set data (p ~ 0.02). These measurements were performed using Lib-A target sites in mESCs.

We note that the multitask model used in inDelphi also jointly trains MHless-NN and, in addition to predicting MH deletion frequencies more accurately than the baseline, also provides strong performance on deletion length frequency prediction.

Using random seed A, the baseline model mean Pearson *r* on held-out data was .905, while the multitask model mean Pearson *r* on the same held-out data was 0.913, for a 8.5% relative improvement (*p* = 0.009, one-sided *t*-test). Using random seed B, the baseline model mean Pearson *r* on held-out data was .924, while the multitask model mean Pearson *r* on the same held-out data was 0.928, for a 5.3% relative improvement (*p* = 0.02, one-sided *t*-test). Using random seed C, the baseline model mean Pearson *r* on held-out data was .912, while the multitask model mean Pearson *r* on the same held-out data was 0.917, for a 5.7% relative improvement (*p* = 0.03, one-sided *t*-test).

inDelphi Deletion Modeling: Summary and Revisiting Assumptions

In summary, inDelphi trains MH-NN, which uses as input (microhomology length, microhomology GC content) to output a psi score which is translated into a phi score using deletion length. This phi score represents the “strength” of the microhomology corresponding to a particular MH deletion genotype. It also trains MHless-NN which uses as input (deletion length) to directly output a phi score representing the “total strength” of all MH-independent activity for a particular deletion length.

While the model assumes that microhomology and microhomology-less repair can overlap in contributions to a single repair genotype, this assumption is made conservatively by assuming that their contributions overlap only when there is no alternative. Specifically, in the context of a deletion length with full microhomology, the model assumes that they must overlap, while in the context of a deletion length without full microhomology, inDelphi allows MHindep-phi to represent all MH-less repair genotypes and none of the MH-dependent repair genotypes which are represented solely using their MH-phi scores. This can be seen by noting that at a deletion length j without full microhomology, MH genotypes are scored using their MH-phi scores, while the length j is scored by MHindep-phi[j] plus the sum of MH-phi for each microhomology. Therefore, the subset of MH-less genotypes at this deletion length have a score MHindep-phi[j].

When the subset of MH-less genotypes includes only one MH-less genotype, this single genotype’s score is equal to MHindep-phi[j]. In general, multiple MH-less genotypes are possible, in which case the total score of all of the MH-less genotypes is equal to MHindep-phi[j].

The relative frequency of MH deletions and MH-less deletions is learned implicitly by the balancing between the sum of all MH-phi and MHindep-phi. Since MHindep-phi does not vary by sequence context while MH-phi does, the model assumes that variation in the fraction of deletions that use MH can at least partially be explained by varying sequence microhomology as represented by MH-NN.

inDelphi Insertion Modeling

Once inDelphi is trained on both deletion tasks, it predicts insertions from a sequence context and cleavage site by using the precision score of the predicted deletion length distribution and total deletion phi (from all MH-phi and MHindep-phi). inDelphi also uses one-hot-encoded binary vectors encoding nucleotides -4 and -3. In a training set, these features are collected and normalized to zero mean and unit variance, and the fraction of 1-bp insertions over the sum counts of 1-bp insertions and all deletions are tabulated as the prediction goal. A *k*-nearest neighbor model is built using the training data. inDelphi uses the default parameter *k* = 5.

On test data, the above procedure is used to predict the frequency of 1-bp insertions out of 1-bp insertions and all deletions for a particular sequence context. Once this frequency is predicted, it is used to make frequency predictions for each of the 4 possible insertion genotypes, which are predicted by deriving from the training set the average insertion frequency for each base given its local sequence context. When the training set is small, only the -4 nucleotide is used. When the training set is relatively large, nucleotides -5, -4, and -3 are used.

To produce a frequency distribution on 1-bp insertions and 1-60 bp deletion genotypes, scores for all deletion genotypes and all 1-bp insertions are normalized to sum to 1. To produce a frequency distribution on indel lengths (+1 to -60), scores for all deletion lengths and 1-bp insertions are normalized to sum to 1.

inDelphi: Repair classes predicted at varying resolution

inDelphi predicts MH-deletions and 1-bp insertions at single base resolution. Measuring performance on the task of genotype frequency prediction considers this subset of repair outcomes only (about 60-70% of all outcomes).

inDelphi predicts MH-less deletions to the resolution of deletion length. That is, inDelphi predicts a single frequency corresponding to the sum total frequency of all unique MH-less deletion genotypes possible for a particular deletion length. This modeling choice was made because genotype frequency replicability among MH-less deletions is substantially lower than among MH deletions.

Measuring performance on the task of indel length frequency considers MH deletions, MH-less deletions, and 1-bp insertions (90% of all outcomes).

In practice, if end-users desire, they can extend inDelphi predictions to frequency predictions for specific MH-less deletion genotypes by noting that MH-less deletions are distributed uniformly between 0 delta-position genotypes, medial genotypes, and N delta-position genotypes.

Comparison with a linear baseline model

We compared inDelphi to a baseline model with the same model structure but replacing the deep neural networks with linear models. We compared using Lib-A mESC data. While inDelphi achieves a mean held-out Pearson correlation of 0.851 on deletion genotype frequency prediction and 0.837 on deletion length frequency prediction, the linear baseline model achieves a mean held-out Pearson correlation of 0.816 on deletion genotype frequency prediction and 0.796 on deletion length frequency prediction. When including the third model component for 1-bp insertion modeling and testing on genotype frequency prediction for 1-bp insertions and all deletions, inDelphi achieves a median held-out Pearson correlation of 0.937 and 0.910 on the task of indel length frequency prediction. The linear baseline model achieves a median held-out Pearson correlation of 0.919 and 0.900 on the two tasks respectively.

From these results, we can see that much of the model's power is derived from its designed structure which is independent of the choice of linear or non-linear modeling. While the baseline does not significantly cripple the model, the use of deep nonlinear neural networks offers a substantial performance improvement (10-24%) above linear modeling. In addition, the strong performance of the linear baseline model highlights that the prediction task, given the model structure, is relatively straightforward. This suggests that our model should be able to generalize well to unseen data.

The deep neural network version of MH-NN learns that microhomology length is more important than % GC (Extended Data Fig. 2). The linear version learns the same concept, with a weight of 1.1585 for MH length and 0.332 for % GC.

Comparison with a baseline model lacking microhomology length as a feature

Microhomology length is an important feature for MH-NN (Extended Data Fig. 2). We trained a model that uses only % GC as input to MH-NN while keeping the rest of the model structure identical. On held-out data at Lib-A target sites in mESCs, this baseline model at convergence achieves to a mean Pearson correlation of 0.59 on the task of predicting deletion genotype frequencies, and a mean Pearson correlation of 0.58 on the task of predicting deletion length frequencies. Notably, a model at iteration 0 with randomly initialized weights achieves mean Pearson correlations of 0.55 and 0.54 on the two respective tasks on held-out data. This basal Pearson correlation is relatively high due to the model structure, in particular, the exponential penalty on deletion length. In sum, removing MH length as a feature severely impacts model performance, restricting it to predictive performance not appreciably better than random chance.

inDelphi training and testing on data from varying cell-types

For predicting genotype and indel length frequencies in any particular cell-type *C* where data *D* is available, we first trained inDelphi's deletion component on a subset of Lib-A mESC data. Then, we apply k-fold cross-validation on *D* where *D* is iteratively split into training and test datasets. For each cross-validation iteration, the training set is used to train the insertion frequency model (k-nearest neighbors) and insertion genotype model (matrix of observed probabilities of each inserted base given local sequence context, which is just the -4 nucleotide when the training dataset is small, and -5, -4 and -3 nucleotides when the training dataset is large). For each cross-validation iteration, predictions are made at each sequence context in the test set which are compared to observations for each sequence context to yield a Pearson correlation. For any particular sequence context, the median test-time Pearson correlation across all cross-validation iterations is used as a single number summary of the overall performance of inDelphi. For all reported results, we used 100-fold cross-validation with 80%/20% training and testing splits. Empirically, we observed small variance in test-time Pearson correlation, highlighting the stability of inDelphi's modeling approach.

inDelphi testing on endogenous VO data

On this task, the deletion component of inDelphi was trained on a subset of the Lib-A mESC data. For each cell type in HCT116, K562, and HEK293T, all VO sequence contexts (about 100) were randomly split into training and test datasets 100 times. During each split, the training set was used for *k*-nearest neighbor modeling of 1-bp insertion frequencies. Feature normalization to zero mean and unit variance was not performed. The average frequency of each 1-bp insertion genotype was derived from the training set as well. For each of the ~100 sequence contexts, the median test-time Pearson correlation was used for plotting in Figure 3. Due to the small size of the training set, only the -4 nucleotide was used for modeling both the insertion frequency and insertion genotype frequencies.

inDelphi testing on library data

On this task, the deletion component of inDelphi was trained on a subset of the Lib-A mESC data. The remaining test set was used for measuring test-time prediction performance on Lib-A. Nucleotides -5, -4, and -3 were used for the insertion genotype model. For testing on Lib-B, Lib-B was split into training and test datasets in the same manner as with VO data. Nucleotide -4 was used for the insertion genotype model. The median test-time Pearson correlation is used as a single number summary of the overall performance of inDelphi on any particular sequence context. For reporting predictive results in Figure 4, sequence contexts with low replicability (less than 0.85 Pearson correlation) in observed editing outcome frequencies were first removed.

inDelphi training and testing on *Prkdc*$^{-/-}$*Lig4*$^{-/-}$ data

inDelphi was trained on data from 946 Lib-A sequence contexts and tested on 168 held-out Lib-A sequence contexts. Nucleotide -4 was used for insertion rate modeling, all other modeling choices were standard as described above. On held-out data, this version of inDelphi achieved a median Pearson correlation of 0.84 on predicting indel genotype frequencies, and 0.80 on predicting indel length frequencies.

Training the online public version of inDelphi and its expected properties

For general-use on arbitrary cell types, we trained a version of inDelphi using additional data from diverse types of cells. Deletion modeling was trained using data from 2,464 sequence contexts from high-replicability Lib-A and Lib-B data (including clinical variants and microduplications, fourbp, and longdup) in mES and data from VO sequence contexts in HEK293 and K562. Insertion frequency modeling is implemented as above. Insertion genotype modeling uses nucleotides -5, -4, and -3. The insertion frequency model and insertion genotype model are trained on VO endogenous data in K562 and HEK293T, Lib-A data in mESC, and Lib-B data (including clinical variants and microduplications, fourbp, and longdup) in mESC and U2OS.

Though MHless-NN, as trained on library data, never receives information on deletion lengths beyond 28, we allow it to generalize its learned function and make predictions on deletion lengths up to 60 bp to match the supported range of MH-NN.

inDelphi makes predictions on 1-bp insertions and 1-60-bp deletions, which we empirically show to consist of higher than 90% of all Cas9 editing outcomes in data from multiple human and mouse cell lines. Nevertheless, there is a subset of repair (about 8% on average) that inDelphi does not attempt to predict. We suggest that end-users, depending on what predictive quantities are of interest, take this into account when using inDelphi. For example, if inDelphi predicts that 60% of 1-bp insertions and 1-60-bp deletions at a disease allele correspond to repair to wildtype genotype, a quantity of interest may be the rate of wildtype repair in all Cas9 editing outcomes (including the 8% not predicted by inDelphi). In such a situation, this quantity can be calculated as (92%*60%) = 55.2%.

By the design of 1872 sequence contexts in Lib-A, our training dataset has rich and uniform representation across all quintiles of several major axes of variation including GC content, precision, and number of bases participating in microhomology as measured empirically in the human genome. This design strategy enables inDelphi to generalize well to arbitrary sequence contexts from the human genome.

These training data further include data in the outlier range of statistics of interest, including extremely high and low precision repair distributions, and extremely weak and strong microhomology (minimal microhomology and extensive microduplication microhomology sequences). The availability of such sequences in our training data enables inDelphi to generalize well to sequence contexts of clinical interest and sequence contexts supporting unusually high frequencies of precision repair. In particular, by training on more than 1000 examples of repair at clinical microduplications, inDelphi has received strong preparation for accurate prediction on other clinical microduplications.

By training on data from many cell-types, we enable inDelphi to make predictions that are generally applicable to many human cell-types. We note that the HCT116 human colon cancer cell line experiences a markedly higher frequency of single base insertions compared to all other cell lines we studied, possibly due to the MLH1 deficiency of this cell line leading to impaired DNA mismatch repair. For this reason, we excluded HCT116 data from our training dataset. For best results, we suggest end-users keep in mind that repair class frequencies can be cell type-dependent, and this issue has not been well-characterized thus far.

We note that inDelphi's main error tendency is on the side of overestimating rather than underestimating the precision of repair (Figure 4). In general, this tendency can be explained by noting that inDelphi only considers sequence microhomology as a factor, while it's plausible and likely in biological experimental settings that even sequence contexts with very strong sequence microhomology may not yield precise results due to noise factors that are not considered by inDelphi. For best results, we recommend end-users take this tendency into account when using inDelphi predictions for further experiments. In particular, if gRNAs are designed by using a minimum precision threshold, end-users should recognize that observed repair outcomes may have empirical precision under this threshold. However, conversely, it is unlikely that a gRNA will have precision higher than what inDelphi predicts.

Lib-A design

All designed sequence contexts were 55 bp in length with cutting between the 27th and 28th base.

1872 sequence contexts were designed by empirically determining the distribution of four statistics in sequence contexts from the human genome. These four statistics are GC content, total sum of bases participating in microhomology for 3-27-bp deletions, Azimuth predicted on-target efficiency score, and the statistical entropy of the predicted 3-27-bp deletion length distribution from a previous version of inDelphi. For each of these statistics, empirical quintiles were derived by calculating these statistics in a large number of sequence contexts from the human genome. For the library, sequence contexts were designed by randomly generated DNA that categorized into each combination of quintiles across each of the four statistics. For example, a sequence context falling into the 1st quintile in GC, 2nd quintile of total MH, 1st quintile of Azimuth score, and 5th quintile of entropy, was found by random search. With four statistics and five bins each (due to quintiles), there are 5^4 = 625 possible combinations. For each combination, we attempted to design three sequence contexts for a total of 1875; 3 sequences could not be designed (for a total of 1872) though each bin was filled. 90 sequence contexts were designed from VO sequence contexts. Other sequence contexts were also designed for a total of 2000 sequence contexts in Lib-A. Lib-A sequence names, gRNAs, and sequence contexts are listed in Supplementary Table 2.

Lib-B design

All designed sequence contexts were 55 bp in length with cutting between the 27th and 28th base.

1592 sequence contexts were designed from Clinvar and HGMD (see section on Selection of variants from disease databases). Some disease sequence contexts were designed that such that the corrected wildtype or frameshift allele supports further cutting by the original gRNA; data from such sequence contexts were ignored during analysis. 57 “longdup” sequence contexts were designed by repeating the following procedure three times: for N = 7 to 25, an N-mer was randomly generated, then duplicating and surrounded by randomly generated sequences, while ensuring that SpCas9 NGG was included and appropriately positioned for cutting between positions 27 and 28. 90 sequence contexts were designed from VO sequence contexts. 228 “fourbp” sequence contexts were designed at 3 contexts with random sequences (with total phi score on average lower than VO sequence contexts) while varying positions -5 to -2; for each of the 3 “low-microhomology” contexts,76 four bases were randomly designed while ensuring representation from all possible 2 bp microhomology patterns including no microhomology, one base of microhomology at either position, and full two bases of microhomology. Other sequence contexts were also designed for a total of 2000 sequence contexts in Lib-B. Lib-B sequence names, gRNAs, and sequence contexts are listed in Supplementary Table 3.

1bpInsDisLib design

12 sequence contexts were designed from Clinvar and HGMD. Pathogenic alleles were selected for a high predicted frequency of correction to the wild-type genotype via a Cas9-mediated 1-bp insertion. Sequence names, gRNAs, and sequence contexts are listed in Supplementary Table 4.

PHG design

18 sequence contexts were designed using inDelphi to select SpCas9 gRNAs targeting the coding regions of genes including VEGFA, VEGFR2, PDCD1, APOB, CCR5, CD274, CXCR4, PCSK9, and APOBEC3B, such that a frameshift would be induced with higher frequency than typical SpCas9 gRNAs. Of these 18 frameshift designs, 10 were designed to induce a single deletion genotype with high precision, and 8 were designed to induce a single 1-bp insertion genotype with high precision. 6 sequence contexts were designed using inDelphi from Clinvar and HGMD where pathogenic 1-bp insertion alleles were selected based on a high predicted frequency of induction from Cas9 editing of the wild-type allele. Sequence names, gRNAs, and sequence contexts are listed in Supplementary Table 5.

Generating a DNA motif for 1-bp insertion frequencies

Nucleotides from positions -7 to 0 were one-hot-encoded and used in ridge regression to predict the observed frequency of 1-bp insertions out of all Cas9 editing events in 1996 sequence contexts from Lib-A mESC data. The data were split into training and testing sets (80/20 split) 10,000 times to calculate a bootstrapped estimate of linear regression weights and test-set predictive Pearson correlation. The median test-set Pearson correlation was found to be 0.62. To generate a DNA motif, any features that included 0 within the bootstrapped weight range were excluded (probability that the weight is zero > 1e-4). The average bootstrapped weight estimate was used as the “logo height” for all remaining features. Each feature is independent; vertical stacking of features follows the published tradition of DNA motifs.

Predicting precision repair of genomic SpCas9 gRNAs

In this work, we determined the distributions of the most frequent deletion and insertion outcomes among major editing outcomes at SpCas9 gRNAs targeting human exons and introns as predicted by inDelphi trained on data from Lib-A target sites in mESCs and U2OS cells separately (Fig. 3f, Extended Data Table 1). A combination of computational constraint (the inability to make predictions at ~350 million target sites comprising all SpCas9 gRNAs in the human genome), uncertainty in the exact predictions of the model and a preference for avoiding overfitting our training data, and lack of sufficient held-out data to verify our predictions and identify potential bias, motivated us to smooth the exact predictions made by the model. We resampled each predicted value from a Gaussian centered at the predicted value with a specified standard deviation. For mESCs, we set the standard deviation as the predicted value divided by 4, up to a maximum of 3% for insertions, while for deletions we used the predicted value divided by

4 with a minimum of 6%. For U2OS cells, we set the standard deviation as the predicted value divided by 4 for insertions, and the predicted value divided by 4 with a minimum of 6% for deletions. The scaling of standard deviation at higher predicted values reflects the abundance of data and therefore higher relative confidence at lower predicted values. The use of symmetrical noise reflects our prior belief that our predictions are equally likely to underestimate and overestimate the true value.

Plasmid and insert sequences

P2T-CAG-MCS-P2A-GFP-PuroR complete plasmid sequence

```
CCACCTAAATTGTAAGCGTTAATATTTTGTTAAAATTCGCGTTAAATTTTTGTTAAAT
CAGCTCATTTTTTAACCAATAGGCCGAAATCGGCAAAATCCCTTATAAATCAAAAGA
ATAGACCGAGATAGGGTTGAGTGTTGTTCCAGTTTGGAACAAGAGTCCACTATTAA
AGAACGTGGACTCCAACGTCAAAGGGCGAAAAACCGTCTATCAGGGCGATGGCCC
ACTACGTGAACCATCACCCTAATCAAGTTTTTTGGGGTCGAGGTGCCGTAAAGCAC
TAAATCGGAACCCTAAAGGGAGCCCCCGATTTAGAGCTTGACGGGGAAAGCCGGC
GAACGTGGCGAGAAAGGAAGGGAAGAAAGCGAAAGGAGCGGGCGCTAGGGCGC
TGGCAAGTGTAGCGGTCACGCTGCGCGTAACCACCACACCCGCCGCGCTTAATGC
GCCGCTACAGGGCGCGTCCCATTCGCCATTCAGGCTGCGCAACTGTTGGGAAGG
GCGATCGGTGCGGGCCTCTTCGCTATTACGCCAGCTGGCGAAAGGGGGATGTGC
TGCAAGGCGATTAAGTTGGGTAACGCCAGGGTTTTCCCAGTCACGACGTTGTAAAA
CGACGGCCAGTGAGCGCGCGTAATACGACTCACTATAGGGCGAATTGGGTACCG
GCATATGGTTCTTGACAGAGGTGTAAAAAGTACTCAAAAATTTTACTCAAGTGAAAG
TACAAGTACTTAGGGAAAATTTTACTCAATTAAAAGTAAAAGTATCTGGCTAGAATC
TTACTTGAGTAAAAGTAAAAAAGTACTCCATTAAAATTGTACTTGAGTATTAAGGAA
GTAAAAGTAAAAGCAAGAAAGATCGATCTCGAAGGATCTGGAGGCCACCATGGTG
TCGATAACTTCGTATAGCATACATTATACGAAGTTATCGTGCTCGACATTGATTATT
GACTAGTTATTAATAGTAATCAATTACGGGGTCATTAGTTCATAGCCCATATATGGA
GTTCCGCGTTACATAACTTACGGTAAATGGCCCGCCTGGCTGACCGCCCAACGAC
CCCCGCCCATTGACGTCAATAATGACGTATGTTCCCATAGTAACGCCAATAGGGAC
TTTCCATTGACGTCAATGGGTGGAGTATTTACGGTAAACTGCCCACTTGGCAGTAC
ATCAAGTGTATCATATGCCAAGTACGCCCCCTATTGACGTCAATGACGGTAAATGG
CCCGCCTGGCATTATGCCCAGTACATGACCTTATGGGACTTTCCTACTTGGCAGTA
CATCTACGTATTAGTCATCGCTATTACCATGGTCGAGGTGAGCCCCACGTTCTGCT
TCACTCTCCCCATCTCCCCCCCCTCCCCACCCCCAATTTTGTATTTATTTATTTTTTA
ATTATTTTGTGCAGCGATGGGGGCGGGGGGGGGGGGGGGGGCGCGCGCCAGGCG
GGGCGGGGCGGGGCGAGGGGCGGGGCGGGGCGAGGCGGAGAGGTGCGGCGG
CAGCCAATCAGAGCGGCGCGCTCCGAAAGTTTCCTTTTATGGCGAGGCGGCGGC
GGCGGCGGCCCTATAAAAAGCGAAGCGCGCGGCGGGCGGGGAGTCGCTGCGAC
GCTGCCTTCGCCCCGTGCCCCGCTCCGCCGCCGCCTCGCGCCGCCCGCCCCGG
CTCTGACTGACCGCGTTACTCCCACAGGTGAGCGGGCGGGACGGCCCTTCTCCTC
CGGGCTGTAATTAGCGCTTGGTTTAATGACGGCTTGTTTCTTTTCTGTGGCTGCGT
GAAAGCCTTGAGGGGCTCCGGGAGGGCCCTTTGTGCGGGGGGAGCGGCTCGGG
GGGTGCGTGCGTGTGTGTGTGCGTGGGGAGCGCCGCGTGCGGCTCCGCGCTGC
CCGGCGGCTGTGAGCGCTGCGGGCGCGGCGCGGGGCTTTGTGCGCTCCGCAGT
GTGCGCGAGGGGAGCGCGGCCGGGGGCGGTGCCCCGCGGTGCGGGGGGGGCT
```

```
GCGAGGGGAACAAAGGCTGCGTGCGGGGTGTGTGCGTGGGGGGGTGAGCAGGG
GGTGTGGGCGCGTCGGTCGGGCTGCAACCCCCCCTGCACCCCCCTCCCCGAGTT
GCTGAGCACGGCCCGGCTTCGGGTGCGGGGCTCCGTACGGGGCGTGGCGCGGG
GCTCGCCGTGCCGGGCGGGGGGTGGCGGCAGGTGGGGGTGCCGGGCGGGGCG
GGGCCGCCTCGGGCCGGGGAGGGCTCGGGGGAGGGGCGCGGCGGCCCCCGGA
GCGCCGGCGGCTGTCGAGGCGCGGCGAGCCGCAGCCATTGCCTTTTATGGTAAT
CGTGCGAGAGGGCGCAGGGACTTCCTTTGTCCCAAATCTGTGCGGAGCCGAAATC
TGGGAGGCGCCGCCGCACCCCCTCTAGCGGGCGCGGGGCGAAGCGGTGCGGCG
CCGGCAGGAAGGAAATGGGCGGGGAGGGCCTTCGTGCGTCGCCGCGCCGCCGT
CCCCTTCTCCCTCTCCAGCCTCGGGGCTGTCCGCGGGGGGACGGCTGCCTTCGG
GGGGGACGGGGCAGGGCGGGGTTCGGCTTCTGGCGTGTGACCGGCGGCTCTAG
AGCCTCTGCTAACCATGTTCATGCCTTCTTCTTTTTCCTACAGCTCCTGGGCAACGT
GCTGGTTATTGTGCTGTCTCATCATTTTGGCAAAGAATTCCTCGAGCGGCCGCCAG
TGTGATGGATATCGGATCCGCTAGCGCTACTAACTTCAGCCTGCTGAAGCAGGCT
GGAGACGTGGAGGAGAACCCTGGACCTGGACCGGTCGCCACCATGGTGAGCAAG
GGCGAGGAGCTGTTCACCGGGGTGGTGCCCATCCTGGTCGAGCTGGACGGCGAC
GTAAACGGCCACAAGTTCAGCGTGTCCGGCGAGGGCGAGGGCGATGCCACCTAC
GGCAAGCTGACCCTGAAGTTCATCTGCACCACCGGCAAGCTGCCCGTGCCCTGG
CCCACCCTCGTGACCACCCTGACCTACGGCGTGCAGTGCTTCAGCCGCTACCCCG
ACCACATGAAGCAGCACGACTTCTTCAAGTCCGCCATGCCCGAAGGCTACGTCCA
GGAGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGACCCGCGCCGAGGT
GAAGTTCGAGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTT
CAAGGAGGACGGCAACATCCTGGGGCACAAGCTGGAGTACAACTACAACAGCCAC
AACGTCTATATCATGGCCGACAAGCAGAAGAACGGCATCAAGGTGAACTTCAAGAT
CCGCCACAACATCGAGGACGGCAGCGTGCAGCTCGCCGACCACTACCAGCAGAA
CACCCCCATCGGCGACGGCCCCGTGCTGCTGCCCGACAACCACTACCTGAGCAC
CCAGTCCGCCCTGAGCAAAGACCCCAACGAGAAGCGCGATCACATGGTCCTGCTG
GAGTTCGTGACCGCCGCCGGGATCACTCTCGGCATGGACGAGCTGTACAAGTAAA
GCGGCCGCCACCGCGGTGGAGCTCGAATTAATTCATCGATGATGATCCAGACATG
ATAAGATACATTGATGAGTTTGGACAAACCACAACTAGAATGCAGTGAAAAAAATG
CTTTATTTGTGAAATTTGTGATGCTATTGCTTTATTTGTAACCATTATAAGCTGCAAT
AAACAAGTTAACAACAACAATTGCATTCATTTTATGTTTCAGGTTCAGGGGGAGGTG
TGGGAGGTTTTTTAAAGCAAGTAAAACCTCTACAAATGTGGTATGGCTGATTATGAT
CCTCTAGAGTCGGTGGGCCTCGGGGGCGGGTGCGGGGTCGGCGGGGCCGCCCC
GGGTGGCTTCGGTCGGAGCCATGGGGTCGTGCGCTCCTTTCGGTCGGGCGCTGC
GGGTCGTGGGGCGGGCGTCAGGCACCGGGCTTGCGGGTCATGCACCAGGTGCG
CGGTCCTTCGGGCACCTCGACGTCGGCGGTGACGGTGAAGCCGAGCCGCTCGTA
GAAGGGGAGGTTGCGGGGCGCGGAGGTCTCCAGGAAGGCGGGCACCCCGGCGC
GCTCGGCCGCCTCCACTCCGGGGAGCACGACGGCGCTGCCCAGACCCTTGCCCT
GGTGGTCGGGCGAGACGCCGACGGTGGCCAGGAACCACGCGGGCTCCTTGGGC
```

```
CGGTGCGGCGCCAGGAGGCCTTCCATCTGTTGCTGCGCGGCCAGCCGGGAACCG
CTCAACTCGGCCATGCGCGGGCCGATCTCGGCGAACACCGCCCCCGCTTCGACG
CTCTCCGGCGTGGTCCAGACCGCCACCGCGGCGCCGTCGTCCGCGACCCACACC
TTGCCGATGTCGAGCCCGACGCGCGTGAGGAAGAGTTCTTGCAGCTCGGTGACC
CGCTCGATGTGGCGGTCCGGGTCGACGGTGTGGCGCGTGGCGGGGTAGTCGGC
GAACGCGGCGGCGAGGGTGCGTACGGCCCGGGGGACGTCGTCGCGGGTGGCGA
GGCGCACCGTGGGCTTGTACTCGGTCATGGAAGGTCGTCTCCTTGTGAGGGGTCA
GGGGCGTGGGTCAGGGGATGGTGGCGGCACCGGTCGTGGCGGCCGACCTGCAG
GCATGCAAGCTTTTTGCAAAAGCCTAGGCCTCCAAAAAAGCCTCCTCACTACTTCT
GGAATAGCTCAGAGGCCGAGGCGGCCTCGGCCTCTGCATAAATAAAAAAAATTAG
TCAGCCATGGGGCGGAGAATGGGCGGAACTGGGCGGAGTTAGGGGCGGGATGG
GCGGAGTTAGGGGCGGGACTATGGTTGCTGACTAATTGAGATGCATGCTTTGCAT
ACTTCTGCCTGCTGGGGAGCCTGGGGACTTTCCACACCTGGTTGCTGACTAATTG
AGATGCATGCTTTGCATACTTCTGCCTGCTGGGGAGCCTGGGGACTTTCCACACC
CTAACTGACACACATTCCACAGAATTCAAGTGATCTCCAAAAAATAAGTACTTTTTG
ACTGTAAATAAAATTGTAAGGAGTAAAAAGTACTTTTTTTTCTAAAAAAATGTAATTA
AGTAAAAGTAAAAGTATTGATTTTTAATTGTACTCAAGTAAAGTAAAAATCCCCAAAA
ATAATACTTAAGTACAGTAATCAAGTAAAATTACTCAAGTACTTTACACCTCTGGTTC
TTGACCCCCTACCTTCAGCAAGCCCAGCAGATCCGAGCTCCAGCTTTTGTTCCCTT
TAGTGAGGGTTAATTGCGCGCTTGGCGTAATCATGGTCATAGCTGTTTCCTGTGTG
AAATTGTTATCCGCTCACAATTCCACACAACATACGAGCCGGAAGCATAAAGTGTA
AAGCCTGGGGTGCCTAATGAGTGAGCTAACTCACATTAATTGCGTTGCGCTCACTG
CCCGCTTTCCAGTCGGGAAACCTGTCGTGCCAGCTGCATTAATGAATCGGCCAAC
GCGCGGGGAGAGGCGGTTTGCGTATTGGGCGCTCTTCCGCTTCCTCGCTCACTGA
CTCGCTGCGCTCGGTCGTTCGGCTGCGGCGAGCGGTATCAGCTCACTCAAAGGC
GGTAATACGGTTATCCACAGAATCAGGGGATAACGCAGGAAAGAACATGTGAGCA
AAAGGCCAGCAAAAGGCCAGGAACCGTAAAAAGGCCGCGTTGCTGGCGTTTTTCC
ATAGGCTCCGCCCCCCTGACGAGCATCACAAAAATCGACGCTCAAGTCAGAGGTG
GCGAAACCCGACAGGACTATAAAGATACCAGGCGTTTCCCCCTGGAAGCTCCCTC
GTGCGCTCTCCTGTTCCGACCCTGCCGCTTACCGGATACCTGTCCGCCTTTCTCC
CTTCGGGAAGCGTGGCGCTTTCTCATAGCTCACGCTGTAGGTATCTCAGTTCGGT
GTAGGTCGTTCGCTCCAAGCTGGGCTGTGTGCACGAACCCCCCGTTCAGCCCGAC
CGCTGCGCCTTATCCGGTAACTATCGTCTTGAGTCCAACCCGGTAAGACACGACTT
ATCGCCACTGGCAGCAGCCACTGGTAACAGGATTAGCAGAGCGAGGTATGTAGGC
GGTGCTACAGAGTTCTTGAAGTGGTGGCCTAACTACGGCTACACTAGAAGGACAG
TATTTGGTATCTGCGCTCTGCTGAAGCCAGTTACCTTCGGAAAAAGAGTTGGTAGC
TCTTGATCCGGCAAACAAACCACCGCTGGTAGCGGTGGTTTTTTTGTTTGCAAGCA
GCAGATTACGCGCAGAAAAAAAGGATCTCAAGAAGATCCTTTGATCTTTTCTACGG
GGTCTGACGCTCAGTGGAACGAAAACTCACGTTAAGGGATTTTGGTCATGAGATTA
TCAAAAAGGATCTTCACCTAGATCCTTTTAAATTAAAAATGAAGTTTTAAATCAATCT
```

AAAGTATATATGAGTAAACTTGGTCTGACAGTTACCAATGCTTAATCAGTGAGGCAC
CTATCTCAGCGATCTGTCTATTTCGTTCATCCATAGTTGCCTGACTCCCCGTCGTGT
AGATAACTACGATACGGGAGGGCTTACCATCTGGCCCCAGTGCTGCAATGATACC
GCGAGACCCACGCTCACCGGCTCCAGATTTATCAGCAATAAACCAGCCAGCCGGA
AGGGCCGAGCGCAGAAGTGGTCCTGCAACTTTATCCGCCTCCATCCAGTCTATTA
ATTGTTGCCGGGAAGCTAGAGTAAGTAGTTCGCCAGTTAATAGTTTGCGCAACGTT
GTTGCCATTGCTACAGGCATCGTGGTGTCACGCTCGTCGTTTGGTATGGCTTCATT
CAGCTCCGGTTCCCAACGATCAAGGCGAGTTACATGATCCCCCATGTTGTGCAAAA
AAGCGGTTAGCTCCTTCGGTCCTCCGATCGTTGTCAGAAGTAAGTTGGCCGCAGT
GTTATCACTCATGGTTATGGCAGCACTGCATAATTCTCTTACTGTCATGCCATCCGT
AAGATGCTTTTCTGTGACTGGTGAGTACTCAACCAAGTCATTCTGAGAATAGTGTAT
GCGGCGACCGAGTTGCTCTTGCCCGGCGTCAATACGGGATAATACCGCGCCACAT
AGCAGAACTTTAAAAGTGCTCATCATTGGAAAACGTTCTTCGGGGCGAAAACTCTC
AAGGATCTTACCGCTGTTGAGATCCAGTTCGATGTAACCCACTCGTGCACCCAACT
GATCTTCAGCATCTTTTACTTTCACCAGCGTTTCTGGGTGAGCAAAAACAGGAAGG
CAAAATGCCGCAAAAAAGGGAATAAGGGCGACACGGAAATGTTGAATACTCATACT
CTTCCTTTTTCAATATTATTGAAGCATTTATCAGGGTTATTGTCTCATGAGCGGATAC
ATATTTGAATGTATTTAGAAAAATAAACAAATAGGGGTTCCGCGCACATTTCCCCGA
AAAGTG

LDLRwt

ATGGGGCCCTGGGGCTGGAAATTGCGCTGGACCGTCGCCTTGCTCCTCGCCGCG
GCGGGGACTGCAGTGGGCGACAGATGCGAAAGAAACGAGTTCCAGTGCCAAGAC
GGGAAATGCATCTCCTACAAGTGGGTCTGCGATGGCAGCGCTGAGTGCCAGGATG
GCTCTGATGAGTCCCAGGAGACGTGCTTGTCTGTCACCTGCAAATCCGGGGACTT
CAGCTGTGGGGGCCGTGTCAACCGCTGCATTCCTCAGTTCTGGAGGTGCGATGGC
CAAGTGGACTGCGACAACGGCTCAGACGAGCAAGGCTGTCCCCCCAAGACGTGC
TCCCAGGACGAGTTTCGCTGCCACGATGGGAAGTGCATCTCTCGGCAGTTCGTCT
GTGACTCAGACCGGGACTGCTTGGACGGCTCAGACGAGGCCTCCTGCCCGGTGC
TCACCTGTGGTCCCGCCAGCTTCCAGTGCAACAGCTCCACCTGCATCCCCCAGCT
GTGGGCCTGCGACAACGACCCCGACTGCGAAGATGGCTCGGATGAGTGGCCGCA
GCGCTGTAGGGGTCTTTACGTGTTCCAAGGGGACAGTAGCCCCTGCTCGGCCTTC
GAGTTCCACTGCCTAAGTGGCGAGTGCATCCACTCCAGCTGGCGCTGTGATGGTG
GCCCCGACTGCAAGGACAAATCTGACGAGGAAAACTGCGCTGTGGCCACCTGTCG
CCCTGACGAATTCCAGTGCTCTGATGGAAACTGCATCCATGGCAGCCGGCAGTGT
GACCGGGAATATGACTGCAAGGACATGAGCGATGAAGTTGGCTGCGTTAATGTGA
CACTCTGCGAGGGACCCAACAAGTTCAAGTGTCACAGCGGCGAATGCATCACCCT
GGACAAAGTCTGCAACATGGCTAGAGACTGCCGGGACTGGTCAGATGAACCCATC
AAAGAGTGCGGGACCAACGAATGCTTGGACAACAACGGCGGCTGTTCCCACGTCT
GCAATGACCTTAAGATCGGCTACGAGTGCCTGTGCCCCGACGGCTTCCAGCTGGT
GGCCCAGCGAAGATGCGAAGATATCGATGAGTGTCAGGATCCCGACACCTGCAGC
CAGCTCTGCGTGAACCTGGAGGGTGGCTACAAGTGCCAGTGTGAGGAAGGCTTC
CAGCTGGACCCCCACACGAAGGCCTGCAAGGCTGTGGGCTCCATCGCCTACCTCT
TCTTCACCAACCGGCACGAGGTCAGGAAGATGACGCTGGACCGGAGCGAGTACA
CCAGCCTCATCCCCAACCTGAGGAACGTGGTCGCTCTGGACACGGAGGTGGCCA
GCAATAGAATCTACTGGTCTGACCTGTCCCAGAGAATGATCTGCAGCACCCAGCTT
GACAGAGCCCACGGCGTCTCTTCCTATGACACCGTCATCAGCAGAGACATCCAGG
CCCCCGACGGGCTGGCTGTGGACTGGATCCACAGCAACATCTACTGGACCGACTC
TGTCCTGGGCACTGTCTCTGTTGCGGATACCAAGGGCGTGAAGAGGAAAACGTTA
TTCAGGGAGAACGGCTCCAAGCCAAGGGCCATCGTGGTGGATCCTGTTCATGGCT
TCATGTACTGGACTGACTGGGGAACTCCCGCCAAGATCAAGAAAGGGGGCCTGAA
TGGTGTGGACATCTACTCGCTGGTGACTGAAAACATTCAGTGGCCCAATGGCATCA
CCCTAGATCTCCTCAGTGGCCGCCTCTACTGGGTTGACTCCAAACTTCACTCCATC
TCAAGCATCGATGTCAATGGGGGCAACCGGAAGACCATCTTGGAGGATGAAAAGA
GGCTGGCCCACCCCTTCTCCTTGGCCGTCTTTGAGGACAAAGTATTTTGGACAGAT
ATCATCAACGAAGCCATTTTCAGTGCCAACCGCCTCACAGGTTCCGATGTCAACTT
GTTGGCTGAAAACCTACTGTCCCCAGAGGATATGGTCCTCTTCCACAACCTCACCC
AGCCAAGAGGAGTGAACTGGTGTGAGAGGACCACCCTGAGCAATGGCGGCTGCC
AGTATCTGTGCCTCCCTGCCCCGCAGATCAACCCCCACTCGCCCAAGTTTACCTG
CGCCTGCCCGGACGGCATGCTGCTGGCCAGGGACATGAGGAGCTGCCTCACAGA
GGCTGAGGCTGCAGTGGCCACCCAGGAGACATCCACCGTCAGGCTAAAGGTCAG

CTCCACAGCCGTAAGGACACAGCACACAACCACCCGGCCTGTTCCCGACACCTCCCGGCTGCCTGGGGCCACCCCTGGGCTCACCACGGTGGAGATAGTGACAATGTCTCACCAAGCTCTGGGCGACGTTGCTGGCAGAGGAAATGAGAAGAAGCCCAGTAGCGTGAGGGCTCTGTCCATTGTCCTCCCCATCGTGCTCCTCGTCTTCCTTTGCCTGGGGGTCTTCCTTCTATGGAAGAACTGGCGGCTTAAGAACATCAACAGCATCAACTTTGACAACCCCGTCTATCAGAAGACCACAGAGGATGAGGTCCACATTTGCCACAACCAGGACGGCTACAGCTACCCCTCGAGACAGATGGTCAGTCTGGAGGATGACGTGGCG

LDLRDup252 with surrounding region

CCCCCAAGACGTGCTCCCAGGACGAGTTTCGCTGCCACGATGGGAAGTGCATCTCTCGGCAGTTCGTCTGTGACTCAGACCGGGACTGCTTGGACGGCTCAGACGAGGCCTCCTGCCCGGTGCTCACCTGTGGTCCCGCCAGCTTCCAGTGCAACAGCTCCACCTGCATCCCCCAGCTGTGGGCCTGCGACAACGACCCCGACTGCGAAGATGGCTCGGAGGCTCGGATGAGTGGCCGCAGCGCTGTAGGGGTCTTTACGTGTTCCAAGGGGACAGTAGCCCCTGCTCGGCCTTCGAGTTCCACTGCCTAAGTGGCGAGTGCATCCACTCCAGCTGGCGCTGTGATGGTGGCCCCGACTGCAAGGACAAATCTGACGAGGAAAACTGCG

LDLRDup254/255 with surrounding region

CCCCCAAGACGTGCTCCCAGGACGAGTTTCGCTGCCACGATGGGAAGTGCATCTCTCGGCAGTTCGTCTGTGACTCAGACCGGGACTGCTTGGACGGCTCAGACGAGGCCTCCTGCCCGGTGCTCACCTGTGGTCCCGCCAGCTTCCAGTGCAACAGCTCCACCTGCATCCCCCAGCTGTGGGCCTGCGACAACGACCCCGACTGCGAAGATGGCTCGGATGAGTGGCCGCAGCGCTGTAGGGGTCTTTACGTGTTCCAAGGGGACAGTAGCCCCTGCTCGGCCTTCGAGTTCCACTGCCTAAGTGGCGAGTGCATCCACTCCAGCTGGCGCTGTGATGGTGGCCCCGACTGCAAGGACAAATCTGACAGGACAAATCTGACGAGGAAAACTGCGCTGTGGCCACCTGTCGCCCTGACGAATTCCAGTGCTCTGATGGAAACTGCATCCATG

LDLRDup258 with surrounding region

CCCCCAAGACGTGCTCCCAGGACGAGTTTCGCTGCCACGATGGGAAGTGCATCTCTCGGCAGTTCGTCTGTGACTCAGACCGGGACTGCTTGGACGGCTCAGACGAGGCCTCCTGCCCGGTGCTCACCTGTGGTCCCGCCAGCTTCCAGTGCAACAGCTCCACCTGCATCCCCCAGCTGTGGGCCTGCGACAACGACCCCGACTGCGAAGATGGCTCGGATGAGTGGCCGCAGCGCTGTAGGGGTCTTTACGTGTTCCAAGGGGACAGTAGCCCCTGCTCGGCCTTCGAGTTCCACTGCCTAAGTGGCGAGTGCATCCACTCCAGCTGGCGCTGTGATGGTGGCCCCGACTGCAAGGACAAATCTGAGGACAAATCTGACGAGGAAAACTGCGCTGTGGCCACCTGTCGCCCTGACGAATTCCAGTGCTCTGATGGAAACTGCATCCATG

LDLRDup261 with surrounding region

CCCCCAAGACGTGCTCCCAGGACGAGTTTCGCTGCCACGATGGGAAGTGCATCTC
TCGGCAGTTCGTCTGTGACTCAGACCGGGACTGCTTGGACGGCTCAGACGAGGC
CTCCTGCCCGGTGCTCACCTGTGGTCCCGCCAGCTTCCAGTGCAACAGCTCCACC
TGCATCCCCCAGCTGTGGGCCTGCGACAACGACCCCGACTGCGAAGATGGCTCG
GATGAGTGGCCGCAGCGCTGTAGGGGTCTTTACGTGTTCCAAGGGGACAGTAGC
CCCTGCTCGGCCTTCGAGTTCCACTGCCTAAGTGGCGAGTGCATCCACTCCAGCT
GGCGCTGTGATGGTGGCCCCGACTGCAAGGACAAATCTGACGACAAATCTGACGA
GGAAAACTGCGCTGTGGCCACCTGTCGCCCTGACGAATTCCAGTGCTCTGATGGA
AACTGCATCCATG

LDLRDup264 with surrounding region

CTTCATGTACTGGACTGACTGGGGAACTCCCGCCAAGATCAAGAAAGGGGGCCTG
AATGGTGTGGACATCTACTCGCTGGTGAGCTGGTGACTGAAAACATTCAGTGGCC
CAATGGCATCACCCTAG

GAAwt

```
ATGGGAGTGAGGCACCCGCCCTGCTCCCACCGGCTCCTGGCCGTCTGCGCCCTC
GTGTCCTTGGCAACCGCTGCACTCCTGGGGCACATCCTACTCCATGATTTCCTGCT
GGTTCCCCGAGAGCTGAGTGGCTCCTCCCCAGTCCTGGAGGAGACTCACCCAGCT
CACCAGCAGGGAGCCAGCAGACCAGGGCCCCGGGATGCCCAGGCACACCCCGG
CCGTCCCAGAGCAGTGCCCACACAGTGCGACGTCCCCCCCAACAGCCGCTTCGA
TTGCGCCCCTGACAAGGCCATCACCCAGGAACAGTGCGAGGCCCGCGGCTGTTG
CTACATCCCTGCAAAGCAGGGGCTGCAGGGAGCCCAGATGGGGCAGCCCTGGTG
CTTCTTCCCACCCAGCTACCCCAGCTACAAGCTGGAGAACCTGAGCTCCTCTGAAA
TGGGCTACACGGCCACCCTGACCCGTACCACCCCCACCTTCTTCCCCAAGGACAT
CCTGACCCTGCGGCTGGACGTGATGATGGAGACTGAGAACCGCCTCCACTTCACG
ATCAAAGATCCAGCTAACAGGCGCTACGAGGTGCCCTTGGAGACCCCGCATGTCC
ACAGCCGGGCACCGTCCCCACTCTACAGCGTGGAGTTCTCCGAGGAGCCCTTCG
GGGTGATCGTGCGCCGGCAGCTGGACGGCCGCGTGCTGCTGAACACGACGGTG
GCGCCCCTGTTCTTTGCGGACCAGTTCCTTCAGCTGTCCACCTCGCTGCCCTCGC
AGTATATCACAGGCCTCGCCGAGCACCTCAGTCCCCTGATGCTCAGCACCAGCTG
GACCAGGATCACCCTGTGGAACCGGGACCTTGCGCCCACGCCCGGTGCGAACCT
CTACGGGTCTCACCCTTTCTACCTGGCGCTGGAGGACGGCGGGTCGGCACACGG
GGTGTTCCTGCTAAACAGCAATGCCATGGATGTGGTCCTGCAGCCGAGCCCTGCC
CTTAGCTGGAGGTCGACAGGTGGGATCCTGGATGTCTACATCTTCCTGGGCCCAG
AGCCCAAGAGCGTGGTGCAGCAGTACCTGGACGTTGTGGGATACCCGTTCATGCC
GCCATACTGGGGCCTGGGCTTCCACCTGTGCCGCTGGGGCTACTCCTCCACCGCT
ATCACCCGCCAGGTGGTGGAGAACATGACCAGGGCCCACTTCCCCCTGGACGTC
CAGTGGAACGACCTGGACTACATGGACTCCCGGAGGGACTTCACGTTCAACAAGG
ATGGCTTCCGGGACTTCCCGGCCATGGTGCAGGAGCTGCACCAGGGCGGCCGGC
GCTACATGATGATCGTGGATCCTGCCATCAGCAGCTCGGGCCCTGCCGGGAGCTA
CAGGCCCTACGACGAGGGTCTGCGGAGGGGGGTTTTCATCACCAACGAGACCGG
CCAGCCGCTGATTGGGAAGGTATGGCCCGGGTCCACTGCCTTCCCCGACTTCACC
AACCCCACAGCCCTGGCCTGGTGGGAGGACATGGTGGCTGAGTTCCATGACCAG
GTGCCCTTCGACGGCATGTGGATTGACATGAACGAGCCTTCCAACTTCATCAGGG
GCTCTGAGGACGGCTGCCCCAACAATGAGCTGGAGAACCCACCCTACGTGCCTG
GGGTGGTTGGGGGGACCCTCCAGGCGGCCACCATCTGTGCCTCCAGCCACCAGT
TTCTCTCCACACACTACAACCTGCACAACCTCTACGGCCTGACCGAAGCCATCGCC
TCCCACAGGGCGCTGGTGAAGGCTCGGGGGACACGCCCATTTGTGATCTCCCGC
TCGACCTTTGCTGGCCACGGCCGATACGCCGGCCACTGGACGGGGGACGTGTGG
AGCTCCTGGGAGCAGCTCGCCTCCTCCGTGCCAGAAATCCTGCAGTTTAACCTGC
TGGGGGTGCCTCTGGTCGGGGCCGACGTCTGCGGCTTCCTGGGCAACACCTCAG
AGGAGCTGTGTGTGCGCTGGACCCAGCTGGGGGCCTTCTACCCCTTCATGCGGAA
CCACAACAGCCTGCTCAGTCTGCCCCAGGAGCCGTACAGCTTCAGCGAGCCGGC
CCAGCAGGCCATGAGGAAGGCCCTCACCCTGCGCTACGCACTCCTCCCCCACCT
```

CTACACACTGTTCCACCAGGCCCACGTCGCGGGGGAGACCGTGGCCCGGCCCCT
CTTCCTGGAGTTCCCCAAGGACTCTAGCACCTGGACTGTGGACCACCAGCTCCTG
TGGGGGGAGGCCCTGCTCATCACCCCAGTGCTCCAGGCCGGGAAGGCCGAAGTG
ACTGGCTACTTCCCCTTGGGCACATGGTACGACCTGCAGACGGTGCCAGTAGAGG
CCCTTGGCAGCCTCCCACCCCCACCTGCAGCTCCCCGTGAGCCAGCCATCCACAG
CGAGGGGCAGTGGGTGACGCTGCCGGCCCCCTGGACACCATCAACGTCCACCT
CCGGGCTGGGTACATCATCCCCCTGCAGGGCCCTGGCCTCACAACCACAGAGTC
CCGCCAGCAGCCCATGGCCCTGGCTGTGGCCCTGACCAAGGGTGGGGAGGCCC
GAGGGGAGCTGTTCTGGGACGATGGAGAGAGCCTGGAAGTGCTGGAGCGAGGG
GCCTACACACAGGTCATCTTCCTGGCCAGGAATAACACGATCGTGAATGAGCTGG
TACGTGTGACCAGTGAGGGAGCTGGCCTGCAGCTGCAGAAGGTGACTGTCCTGG
GCGTGGCCACGGCGCCCCAGCAGGTCCTCTCCAACGGTGTCCCTGTCTCCAACTT
CACCTACAGCCCCGACACCAAGGTCCTGGACATCTGTGTCTCGCTGTTGATGGGA
GAGCAGTTTCTCGTCAGCTGGTGT

GAADup327/328

ATGGGAGTGAGGCACCCGCCCTGCTCCCACCGGCTCCTGGCCGTCTGCGCCCTC
GTGTCCTTGGCAACCGCTGCACTCCTGGGGCACATCCTACTCCATGATTTCCTGCT
GGTTCCCCGAGAGCTGAGTGGCTCCTCCCCAGTCCTGGAGGAGACTCACCCAGCT
CACCAGCAGGGAGCCAGCAGACCAGGGCCCCGGGATGCCCAGGCACACCCCGG
CCGTCCCAGAGCAGTGCCCACACAGTGCGACGTCCCCCCCAACAGCCGCTTCGA
TTGCGCCCCTGACAAGGCCATCACCCAGGAACAGTGCGAGGCCCGCGGCTGTTG
CTACATCCCTGCAAAGCAGGGGCTGCAGGGAGCCCAGATGGGGCAGCCCTGGTG
CTTCTTCCCACCCAGCTACCCCAGCTACAAGCTGGAGAACCTGAGCTCCTCTGAAA
TGGGCTACACGGCCACCCTGACCCGTACCACCCCCACCTTCTTCCCCAAGGACAT
CCTGACCCTGCGGCTGGACGTGATGATGGAGACTGAGAACCGCCTCCACTTCACG
ATCAAAGATCCAGCTAACAGGCGCTACGAGGTGCCCTTGGAGACCCCGCATGTCC
ACAGCCGGGCACCGTCCCCACTCTACAGCGTGGAGTTCTCCGAGGAGCCCTTCG
GGGTGATCGTGCGCCGGCAGCTGGACGGCCGCGTGCTGCTGAACACGACGGTG
GCGCCCCTGTTCTTTGCGGACCAGTTCCTTCAGCTGTCCACCTCGCTGCCCTCGC
AGTATATCACAGGCCTCGCCGAGCACCTCAGTCCCCTGATGCTCAGCACCAGCTG
GACCAGGATCACCCTGTGGAACCGGGACCTTGCGCCCACGCCCGGTGCGAACCT
CTACGGGTCTCACCCTTTCTACCTGGCGCTGGAGGACGGCGGGTCGGCACACGG
GGTGTTCCTGCTAAACAGCAATGCCATGGATGTGGTCCTGCAGCCGAGCCCTGCC
CTTAGCTGGAGGTCGACAGGTGGGATCCTGGATGTCTACATCTTCCTGGGCCCAG
AGCCCAAGAGCGTGGTGCAGCAGTACCTGGACGTTGTGGGATACCCGTTCATGCC
GCCATACTGGGGCCTGGGCTTCCACCTGTGCCGCTGGGGCTACTCCTCCACCGCT
ATCACCCGCCAGGTGGTGGAGAACATGACCAGGGCCCACTTCCCCCTGGACGTC
CAGTGGAACGACCTGGACTACATGGACTCCCGGAGGGACTTCACGTTCAACAAGG
ATGGCTTCCGGGACTTCCCGGCCATGGTGCAGGAGCTGCACCAGGGCGGCCGGC

GCTACATGATGATCGTGGATCCTGCCATCAGCAGCTCGGGCCCTGCCGGGAGCTA
CAGGCCCTACGACGAGGGTCTGCGGAGGGGGGTTTTCATCACCAACGAGACCGG
CCAGCCGCTGATTGGGAAGGTATGGCCCGGGTCCACTGCCTTCCCCGACTTCACC
AACCCCACAGCCCTGGCCTGGTGGGAGGACATGGTGGCTGAGTTCCATGACCAG
GTGCCCTTCGACGGCATGTGGATTGACATGAACGAGCCTTCCAACTTCATCAGGG
GCTCTGAGGACGGCTGCCCCAACAATGAGCTGGAGAACCCACCCTACGTGCCTG
GGGTGGTTGGGGGGACCCTCCAGGCGGCCACCATCTGTGCCTCCAGCCACCAGT
TTCTCTCCACACACTACAACCTGCACAACCTCTACGGCCTGACCGAAGCCATCGCC
TCCCACAGGGCGCTGGTGAAGGCTCGGGGGACACGCCCATTTGTGATCTCCCGC
TCGACCTTTGCTGGCCACGGCCGATACGCCGGCCACTGGACGGGGGACGTGTGG
AGCTCCTGGGAGCAGCTCGCCTCCTCCGTGCCAGAAATCCTGCAGTTTAACCTGC
TGGGGGTGCCTCTGGTCGGGGCCGACGTCTGCGGCTTCCTGGGCAACACCTCAG
AGGAGCTGTGTGTGCGCTGGACCCAGCTGGGGGCCTTCTACCCCTTCATGCGGAA
CCACAACAGCCTGCTCAGTCTGCCCCAGGAGCCGTACAGCTTCAGCGAGCCGGC
CCAGCAGGCCATGAGGAAGGCCCTCACCCTGCGCTACGCACTCCTCCCCCACCT
CTACACACTGTTCCACCAGGCCCACGTCGCGGGGGAGACCGTGGCCCGGCCCCT
CTTCCTGGAGTTCCCCAAGGACTCTAGCACCTGGACTGTGGACCACCAGCTCCTG
TGGGGGGAGGCCCTGCTCATCACCCCAGTGCTCCAGGCCGGGAAGGCCGAAGTG
ACTGGCTACTTCCCCTTGGGCACATGGTACGACCTGCAGACGGTGCCAGTAGAGG
CCCTTGGCAGCCTCCCACCCCCACCTGCAGCTCCCCGTGAGCCAGCCATCCACAG
CGAGGGGCAGTGGGTGACGCTGCCGGCCCCCCTGGACACCATCAACGTCCACCT
CCGGGCTGGGTACATCATCCCCCTGCAGGGCCCTGGCCTCACAACCACAGAGTC
CCGCCAGCAGCCCATGGCCCTGGCTGTGGCCCTGACCAAGGGTGGGGAGGCCC
GAGGGGAGCTGTTCTGGGACGATGGAGAGAGCCTGGAAGTGCTGGAGCGAGGG
GCCTACACACAGGTCATCTTCCTGGCCAGGAATAACACGATCGTGAATGAGCTGG
TACGTGTGACCAGTGAGGGAGCTGGCCTGCAGCTGCAGAAGGTGACTGCAGAAG
GTGACTGTCCTGGGCGTGGCCACGGCGCCCCAGCAGGTCCTCTCCAACGGTGTC
CCTGTCTCCAACTTCACCTACAGCCCCGACACCAAGGTCCTGGACATCTGTGTCTC
GCTGTTGATGGGAGAGCAGTTTCTCGTCAGCTGGTGT

GLB1wt

ATGCCGGGGTTCCTGGTTCGCATCCTCCCTCTGTTGCTGGTTCTGCTGCTTCTGG
GCCCTACGCGCGGCTTGCGCAATGCCACCCAGAGGATGTTTGAAATTGACTATAG
CCGGGACTCCTTCCTCAAGGATGGCCAGCCATTTCGCTACATCTCAGGAAGCATTC
ACTACTCCCGTGTGCCCCGCTTCTACTGGAAGGACCGGCTGCTGAAGATGAAGAT
GGCTGGGCTGAACGCCATCCAGACGTATGTGCCCTGGAACTTTCATGAGCCCTGG
CCAGGACAGTACCAGTTTTCTGAGGACCATGATGTGGAATATTTTCTTCGGCTGGC
TCATGAGCTGGGACTGCTGGTTATCCTGAGGCCCGGGCCCTACATCTGTGCAGAG
TGGGAAATGGGAGGATTACCTGCTTGGCTGCTAGAGAAAGAGTCTATTCTTCTCCG

```
CTCCTCCGACCCAGATTACCTGGCAGCTGTGGACAAGTGGTTGGGAGTCCTTCTG
CCCAAGATGAAGCCTCTCCTCTATCAGAATGGAGGGCCAGTTATAACAGTGCAGG
TTGAAAATGAATATGGCAGCTACTTTGCCTGTGATTTTGACTACCTGCGCTTCCTGC
AGAAGCGCTTTCGCCACCATCTGGGGGATGATGTGGTTCTGTTTACCACTGATGGA
GCACATAAAACATTCCTGAAATGTGGGGCCCTGCAGGGCCTCTACACCACGGTGG
ACTTTGGAACAGGCAGCAACATCACAGATGCTTTCCTAAGCCAGAGGAAGTGTGA
GCCCAAAGGACCCTTGATCAATTCTGAATTCTATACTGGCTGGCTAGATCACTGGG
GCCAACCTCACTCCACAATCAAGACCGAAGCAGTGGCTTCCTCCCTCTATGATATA
CTTGCCCGTGGGGCGAGTGTGAACTTGTACATGTTTATAGGTGGGACCAATTTTGC
CTATTGGAATGGGGCCAACTCACCCTATGCAGCACAGCCCACCAGCTACGACTAT
GATGCCCCACTGAGTGAGGCTGGGGACCTCACTGAGAAGTATTTTGCTCTGCGAA
ACATCATCCAGAAGTTTGAAAAAGTACCAGAAGGTCCTATCCCTCCATCTACACCA
AAGTTTGCATATGGAAAGGTCACTTTGGAAAAGTTAAAGACAGTGGGAGCAGCTCT
GGACATTCTGTGTCCCTCTGGGCCCATCAAAAGCCTTTATCCCTTGACATTTATCCA
GGTGAAACAGCATTATGGGTTTGTGCTGTACCGGACAACACTTCCTCAAGATTGCA
GCAACCCAGCACCTCTCTCTTCACCCCTCAATGGAGTCCACGATCGAGCATATGTT
GCTGTGGATGGGATCCCCCAGGGAGTCCTTGAGCGAAACAATGTGATCACTCTGA
ACATAACAGGGAAAGCTGGAGCCACTCTGGACCTTCTGGTAGAGAACATGGGACG
TGTGAACTATGGTGCATATATCAACGATTTTAAGGGTTTGGTTTCTAACCTGACTCT
CAGTTCCAATATCCTCACGGACTGGACGATCTTTCCACTGGACACTGAGGATGCAG
TGTGCAGCCACCTGGGGGGCTGGGGACACCGTGACAGTGGCCACCATGATGAAG
CCTGGGCCCACAACTCATCCAACTACACGCTCCCGGCCTTTTATATGGGGAACTTC
TCCATTCCCAGTGGGATCCCAGACTTGCCCCAGGACACCTTTATCCAGTTTCCTGG
ATGGACCAAGGGCCAGGTCTGGATTAATGGCTTTAACCTTGGCCGCTATTGGCCA
GCCCGGGGCCCTCAGTTGACCTTGTTTGTGCCCCAGCACATCCTGATGACCTCGG
CCCCAAACACCATCACCGTGCTGGAACTGGAGTGGGCACCCTGCAGCAGTGATGA
TCCAGAACTATGTGCTGTGACGTTCGTGGACAGGCCAGTTATTGGCTCATCTGTGA
CCTACGATCATCCCTCCAAACCTGTTGAAAAAAGACTCATGCCCCCACCCCCGCAA
AAAAACAAAGATTCATGGCTGGACCATGTA
```

GLB1Dup84

ATGCCGGGGTTCCTGGTTCGCATCCTCCCTCTGTTGCTGGTTCTGCTGCTTCTGG
GCCCTACGCGCGGCTTGCGCAATGCCACCCAGAGGATGTTTGAAATTGACTATAG
CCGGGACTCCTTCCTCAAGGATGGCCAGCCATTTCGCTACATCTCAGGAAGCATTC
ACTACTCCCGTGTGCCCCGCTTCTACTGGAAGGACCGGCTGCTGAAGATGAAGAT
GGCTGGGCTGAACGCCATCCAGACGTATGTGCCCTGGAACTTTCATGAGCCCTGG
CCAGGACAGTACCAGTTTTCTGAGGACCATGATGTGGAATATTTTCTTCGGCTGGC
TCATGAGCTGGGACTGCTGGTTATCCTGAGGCCCGGGCCCTACATCTGTGCAGAG
TGGGAAATGGGAGGATTACCTGCTTGGCTGCTAGAGAAAGAGTCTATTCTTCTCCG
CTCCTCCGACCCAGATTACCTGGCAGCTGTGGACAAGTGGTTGGGAGTCCTTCTG
CCCAAGATGAAGCCTCTCCTCTATCAGAATGGAGGGCCAGTTATAACAGTGCAGG
TTGAAAATGAATATGGCAGCTACTTTGCCTGTGATTTTGACTACCTGCGCTTCCTGC
AGAAGCGCTTTCGCCACCATCTGGGGGATGATGTGGTTCTGTTTACCACTGATGGA
GCACATAAAACATTCCTGAAATGTGGGGCCCTGCAGGGCCTCTACACCACGGTGG
ACTTTGGAACAGGCAGCAACATCACAGATGCTTTCCTAAGCCAGAGGAAGTGTGA
GCCCAAAGGACCCTTGATCAATTCTGAATTCTATACTGGCTGGCTAGATCACTGGG
GCCAACCTCACTCCACAATCAAGACCGAAGCAGTGGCTTCCTCCCTCTATGATATA
CTTGCCCGTGGGGCGAGTGTGAACTTGTACATGTTTATAGGTGGGACCAATTTTGC
CTATTGGAATGGGGCCAACTCACCCTATGCAGCACAGCCCACCAGCTACGACTAT
GATGCCCCACTGAGTGAGGCTGGGGACCTCACTGAGAAGTATTTTGCTCTGCGAA
ACATCATCCAGAAGTTTGAAAAAGTACCAGAAGGTCCTATCCCTCCATCTACACCA
AAGTTTGCATATGGAAAGGTCACTTTGGAAAAGTTAAAGACAGTGGGAGCAGCTCT
GGACATTCTGTGTCCCTCTGGGCCCATCAAAAGCCTTTATCCCTTGACATTTATCCA
GGTGAAACAGCATTATGGGTTTGTGCTGTACCGGACAACACTTCCTCAAGATTGCA
GCAACCCAGCACCTCTCTCTTCACCCCTCAATGGAGTCCACGATCGAGCATATGTT
GCTGTGGATGGGATCCCCCAGGGAGTCCTTGAGCGAAACAATGTGATCACTCTGA
ACATAACAGGGAAAGCTGGAGCCACTCTGGACCTTCTGGTAGAGAACATGGGACG
TGTGAACTATGGTGCATATATGGTGCATATATCAACGATTTTAAGGGTTTGGTTTCT
AACCTGACTCTCAGTTCCAATATCCTCACGGACTGGACGATCTTTCCACTGGACAC
TGAGGATGCAGTGTGCAGCCACCTGGGGGGCTGGGGACACCGTGACAGTGGCCA
CCATGATGAAGCCTGGGCCCACAACTCATCCAACTACACGCTCCCGGCCTTTTATA
TGGGGAACTTCTCCATTCCCAGTGGGATCCCAGACTTGCCCCAGGACACCTTTATC
CAGTTTCCTGGATGGACCAAGGGCCAGGTCTGGATTAATGGCTTTAACCTTGGCC
GCTATTGGCCAGCCCGGGGCCCTCAGTTGACCTTGTTTGTGCCCCAGCACATCCT
GATGACCTCGGCCCCAAACACCATCACCGTGCTGGAACTGGAGTGGGCACCCTG
CAGCAGTGATGATCCAGAACTATGTGCTGTGACGTTCGTGGACAGGCCAGTTATT
GGCTCATCTGTGACCTACGATCATCCCTCCAAACCTGTTGAAAAAAGACTCATGCC
CCCACCCCCGCAAAAAAACAAAGATTCATGGCTGGACCATGTA

PORCNwt

```
ATGGCCACCTTTAGCCGCCAGGAATTTTTCCAGCAGCTACTGCAAGGCTGTCTCCT
GCCTACTGCCCAGCAGGGCCTTGACCAGATCTGGCTGCTCCTTGCCATCTGCCTC
GCCTGCCGCCTCCTCTGGAGGCTCGGGTTGCCATCCTACCTGAAGCATGCAAGCA
CCGTGGCAGGCGGGTTCTTCAGCCTCTACCACTTCTTCCAGCTGCACATGGTTTG
GGTCGTGCTGCTCAGCCTCCTGTGCTACCTCGTGCTGTTCCTCTGCCGACATTCCT
CCCATCGAGGCGTCTTCCTATCCGTCACCATCCTCATCTACCTACTCATGGGTGAG
ATGCACATGGTAGACACCGTGACATGGCACAAGATGCGAGGGGCACAGATGATTG
TGGCCATGAAGGCAGTGTCTCTGGGCTTCGACCTGGACCGGGGCGAGGTGGGTA
CGGTGCCCTCGCCAGTGGAGTTCATGGGCTACCTCTACTTCGTGGGCACCATCGT
CTTCGGGCCCTGGATATCCTTCCACAGCTACCTACAAGCTGTCCAAGGCCGCCCA
CTGAGCTGCCGGTGGCTGCAGAAGGTGGCCCGGAGCCTGGCACTGGCCCTGCTG
TGCCTTGTGCTGTCCACTTGCGTGGGCCCCTACCTCTTCCCGTACTTCATCCCCCT
CAACGGTGACCGCCTCCTTCGCAAGGGCACCATGGTAAGGTGGCTGCGAGCCTA
CGAGAGTGCTGTCTCCTTCCACTTCAGCAACTATTTTGTGGGCTTTCTTTCCGAGG
CCACGGCCACGTTGGCGGGGGCTGGCTTTACCGAGGAGAAGGATCACCTGGAAT
GGGACCTGACGGTGTCCAAGCCACTGAATGTGGAGCTGCCTCGGTCAATGGTGG
AAGTTGTCACAAGCTGGAACCTGCCCATGTCTTATTGGCTAAATAACTATGTTTTCA
AGAATGCTCTCCGCCTGGGGACCTTCTCGGCTGTGCTGGTCACCTATGCAGCCAG
CGCCCTCCTACATGGCTTCAGTTTCCACCTGGCTGCGGTCCTGCTGTCCCTGGCT
TTTATCACTTACGTGGAGCATGTCCTCCGGAAGCGCCTGGCTCGGATCCTCAGTG
CCTGTGTCTTGTCAAAGCGGTGCCCGCCAGACTGTTCGCACCAGCATCGCTTGGG
CCTGGGGGTGCGAGCCTTAAACTTGCTCTTTGGAGCTCTGGCCATCTTCCACCTG
GCCTACCTGGGCTCCCTGTTTGATGTCGATGTGGATGACACCACAGAGGAGCAGG
GCTACGGCATGGCATACACTGTCCACAAGTGGTCAGAGCTCAGCTGGGCCAGTCA
CTGGGTCACTTTTGGATGCTGGATCTTCTACCGTCTCATAGGC
```

PORCNDup20

```
ATGGCCACCTTTAGCCGCCAGGAATTTTTCCAGCAGCTACTGCAAGGCTGTCTCCT
GCCTACTGCCCAGCAGGGCCTTGACCAGATCTGGCTGCTCCTTGCCATCTGCCTC
GCCTGCCGCCTCCTCTGGAGGCTCGGGTTGCCATCCTACCTGAAGCATGCAAGCA
CCGTGGCAGGCGGGTTCTTCAGCCTCTACCACTTCTTCCAGCTGCACATGGTTTG
GGTCGTGCTGCTCAGCCTCCTGTGCTACCTCGTGCTGTTCCTCTGCCGACATTCCT
CCCATCGAGGCGTCTTCCTATCCGTCACCATCCTCATCTACCTACTCATGGGTGAG
ATGCACATGGTAGACACCGTGACATGGCACAAGATGCGAGGGGCACAGATGATTG
TGGCCATGAAGGCAGTGTCTCTGGGCTTCGACCTGGACCGGGGCGAGGTGGGTA
CGGTGCCCTCGCCAGTGGAGTTCATGGGCTACCTCTACTTCGTGGGCACCATCGT
CTTCGGGCCCTGGATATCCTTCCACAGCTACCTACAAGCTGTCCAAGGCCGCCCA
CTGAGCTGCCGGTGGCTGCAGAAGGTGGCCCGGAGCCTGGCACTGGCCCTGCTG
TGCCTTGTGCTGTCCACTTGCGTGGGCCCCTACCTCTTCCCGTACTTCATCCCCCT
```

```
CAACGGTGACCGCCTCCTTCGCAAGGGCACCATGGTAAGGTGGCTGCGAGCCTA
CGAGAGTGCTGTCTCCTTCCACTTCAGCAACTATTTTGTGGGCTTTCTTTCCGAGG
CCACGGCCACGTTGGCGGGGGCTGGCTTTACCGAGGAGAAGGATCACCTGGAAT
GGGACCTGACGGTGTCCAAGCCACTGAATGTGGAGCTGCCTCGGTCAATGGTGG
AAGTTGTCACAAGCTGGAACCTGCCCATGTCTTATTGGCTAAATAACTATGTTTTCA
AGAATGCTCTCCGCCTGGGGACCTTCTCGGCTGTGCTGGTCACCTATGCAGCCAG
CGCCCTCCTACATGGCTTCAGTTTCCACCTGGCTGCGGTCCTGCTGTCCCTGGCT
TTTATCCCTGGCTTTTATCACTTACGTGGAGCATGTCCTCCGGAAGCGCCTGGCTC
GGATCCTCAGTGCCTGTGTCTTGTCAAAGCGGTGCCCGCCAGACTGTTCGCACCA
GCATCGCTTGGGCCTGGGGGTGCGAGCCTTAAACTTGCTCTTTGGAGCTCTGGCC
ATCTTCCACCTGGCCTACCTGGGCTCCCTGTTTGATGTCGATGTGGATGACACCAC
AGAGGAGCAGGGCTACGGCATGGCATACACTGTCCACAAGTGGTCAGAGCTCAG
CTGGGCCAGTCACTGGGTCACTTTTGGATGCTGGATCTTCTACCGTCTCATAGGC
```

References

1. Ceccaldi, R., Rondinelli, B. & D'Andrea, A. D. Repair Pathway Choices and Consequences at the Double-Strand Break. *Spec. Issue Qual. Control* **26,** 52–64 (2016).
2. DiCarlo, J. E., Chavez, A., Dietz, S. L., Esvelt, K. M. & Church, G. M. Safeguarding CRISPR-Cas9 gene drives in yeast. *Nat. Biotechnol.* **33,** 1250 (2015).
3. McVey, M. & Lee, S. E. MMEJ repair of double-strand breaks (director's cut): deleted sequences and alternative endings. *Trends Genet.* **24,** 529–538 (2008).
4. Yu, A. M. & McVey, M. Synthesis-dependent microhomology-mediated end joining accounts for multiple types of repair junctions. *Nucleic Acids Res.* **38,** 5706–5717 (2010).
5. Davis, A. J. & Chen, D. J. DNA double strand break repair via non-homologous end-joining. *Transl. Cancer Res.* **2,** 130–143 (2013).
6. Heidenreich, E., Novotny, R., Kneidinger, B., Holzmann, V. & Wintersberger, U. Non-homologous end joining as an important mutagenic process in cell cycle-arrested cells. *EMBO J.* **22,** 2274 (2003).
7. Pfeiffer, P., Goedecke, W. & Obe, G. Mechanisms of DNA double-strand break repair and their potential to induce chromosomal aberrations. *Mutagenesis* **15,** 289–302 (2000).
8. Brown, J. S. *et al.* Neddylation Promotes Ubiquitylation and Release of Ku from DNA-Damage Sites. *Cell Rep.* **11,** 704–714 (2015).
9. Landrum, M. J. *et al.* ClinVar: Public archive of interpretations of clinically relevant variants. *Nucleic Acids Res.* **44,** D862–D868 (2016).
10. Stenson, P. D. *et al.* Human Gene Mutation Database: towards a comprehensive central mutation database. *J. Med. Genet.* **45,** 124 (2008).