Modifying Reference Sequence and Annotation Files Quickly and Reproducibly with *ref*orm

Mohammed Khalfan[1]*, Eric Borenstein[1], Pieter Spealman[1], Farah Abdul-Rahman[1], and David Gresham[1]*

[1]*Center for Genomics and Systems Biology, Department of Biology, New York University, New York, NY, USA 10003*

* *To whom correspondence should be addressed.* ⟨*mkhalfan@nyu.edu, dgresham@nyu.edu*⟩

July 2021

Abstract

Summary: With the increased use of genome editing tools such as CRISPR/Cas9, and the widespread use of reference genome based bioinformatic analyses, it is increasingly necessary to edit reference genome sequences and annotations to include novel sequences and features in engineered genomes (e.g. transgenes, protein fusions, and selectable markers). Whereas it is possible to manually edit sequences and annotations, the process is time-consuming and prone to error. However, failing to update reference data to match edits in corresponding genomes can result in findings that are misleading and incorrect. Here we describe *ref*orm, a novel tool that provides a simple, reproducible solution for creating modified reference data for use with standard bioinformatic analysis pipelines. *ref*orm was developed in Python3, is open-source, and freely available for download through GitHub (https://github.com/gencorefacility/reform), with detailed documentation and examples available at https://gencore.bio.nyu.edu/reform. To ensure *ref*orm is accessible to all users, a web-based application with a full graphical user interface has also been developed (https://reform.bio.nyu.edu).

1 Introduction

Next-generation DNA sequencing (NGS) has become an essential tool for biological researchers. Many popular bioinformatics tools used for the analysis of NGS data, such as HISAT2 (Kim et al., 2015), Cell Ranger (http://10xgenomics.com/), STAR (Dobin et al., 2013), BWA (Li and Durbin, 2009), and GATK (DePristo

et al., 2011), rely on reference genome-based approaches. Typically, these tools require data in the form of whole genome sequences in FASTA format and corresponding genomic feature annotations in GFF3 or GTF format, which are usually acquired from public databases such as NCBI (NCBI Resource Coordinators, 2016), Ensembl (Zerbino et al., 2018), and the UCSC Genome Browser database (Haeussler et al., 2019). NGS data visualization tools such as the Integrative Genomics Viewer (IGV) (Robinson et al., 2011) and JBrowse (Buels et al., 2016) also make use of reference sequence and annotation files to visualize data effectively.

In parallel, advances in genome editing techniques such as CRISPR/Cas9 (Cong et al. 2013) have made it easier for researchers to engineer the genome of their organism of choice. Accurate sequence analysis and visualization of data aligned to the modified genomes require modifying the reference data to reflect the engineered changes to the genome (Figure 1a). Although it is possible to edit reference sequence and annotation data manually, the process is time-consuming, tedious, and prone to human error. Moreover, data that appear correct to the eye after editing may violate file formatting rules, rendering the files unusable during analysis. Tools such as GATK FastaAlternateReferenceMaker (DePristo et al., 2011) and VCFtools vcf-consensus (Danecek et al., 2011) make the generation of alternative reference sequences possible; however, they do not operate on annotation files and thus may cause reference sequences and annotations to be out of sync. To address this commonly encountered problem we developed *ref*orm to provide a fast, easy, and reproducible means of modifying reference genome and annotation files (Figure 1b).

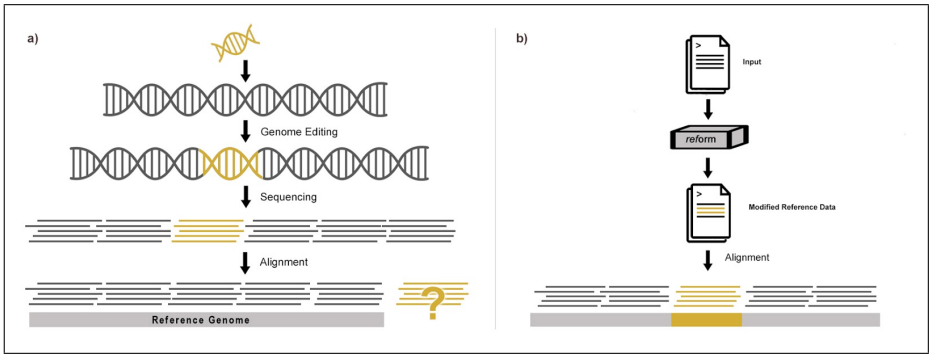

Figure 1: a) Accurate analysis of edited genomes requires editing reference data used for analysis. Foreign sequence (yellow) that is engineered into the genome of an organism is not accurately mapped using standard reference sequences and annotations. b) *ref*orm provides a solution for modifying reference genome data facilitating the accurate use of reference based alignment methods for bespoke engineered genomes.

2 Materials and Methods

2.1 Running *ref*orm

*ref*orm is written in Python 3 and runs on the command line. The sole dependency is BioPython which is used to read and create FASTA records. Execution of *ref*orm requires four files: the reference sequence (FASTA) and corresponding annotation (GFF3 or GTF) files that are to be modified, and files containing the novel sequence (FASTA) and corresponding annotation (GFF3 or GTF) that are to be added. The user must also provide as arguments the name of the chromosome to be modified and either the coordinate, or the upstream and downstream sequences, at which the novel sequence is inserted. In the latter case, the given upstream and downstream sequences must be present and unique within the chromosome. If a coordinate position argument is given, the novel sequence is inserted at the specified position. Likewise, if a given upstream and downstream sequence are immediately adjacent to each other, the novel sequence is inserted between those sequences. However, if the upstream and downstream sequences are not directly adjacent, any sequence between them will be deleted before the novel sequence is inserted. This results in the addition of the novel sequences, and possible deletion of some reference sequence, in the modified reference fasta file. If the chromosome length is provided in the FASTA header, it is updated with the new, modified length. Additionally, any changes to the existing annotations that result from deleted or interrupted sequence are incorporated into the modified annotation information. These modifications are reported in the comment field of the affected features in the new GFF or GTF.

2.2 Reference genome feature disruption

Inserting and/or deleting nucleotides from an existing sequence can disrupt existing genomic features in several ways. In the simplest case, inserting a novel sequence into an existing chromosome will offset all features downstream of the insertion by the length of the inserted sequence. Similarly, a deletion that results in the removal of an entire feature will offset all features downstream of the deletion by the length of the deleted sequence. A more complex case is an insertion or deletion within an existing feature. In this case, *ref*orm splits the existing feature into two, renaming one side of the split feature, and adding comments to both ends indicating the feature has been split. Other complex scenarios include indels that truncate the 5' or 3' side of a feature. In all of these cases, *ref*orm will update existing genomic feature data in the reference annotation based on the changes which have occurred within the genome sequence.

2.3 Output and downstream analysis

*ref*orm produces two files as output, a modified whole genome sequence called ⟨*input*⟩_reformed.fa and a corresponding annotation file called ⟨*input*⟩_reformed.gtf or ⟨*input*⟩_reformed.gff3 (depending on the input annotation format). These

new files are produced without altering the original input files and are compatible with all standard bioinformatic pipelines. *ref*orm has been used at the NYU Center for Genomics and Systems Biology to create modified reference genome data for use with a variety of standard bioinformatics analysis tools and with the 10x Cell Ranger pipeline. Test cases in the codebase ensure the integrity of the software, and that future updates to the code do not modify the output in unexpected ways thereby maintaining reproducibility. This is achieved using a single test script that produces reformed data on the fly for different classes of sequence/feature disruptions and then compares this newly generated data to a predefined gold standard data set.

2.4 Web Service

In addition to running on the command line, *ref*orm has been developed into a web-based application with a full graphical user interface for easier accessibility that is available at https://reform.bio.nyu.edu. This allows users to produce customized genome reference data using a web browser without the need for command line knowledge. While users need to provide data corresponding to their genomic modifications, the original reference data can be provided via a url link to the reference files available in public databases.

3 Results

We describe two illustrative case studies to describe the utility of *ref*orm and highlight how failing to edit reference data can produce misleading and/or incorrect results.

3.1 Accounting for transgenes in reference files prevents incorrect evidence for increased coverage, expression, and variation

The HOXA13 transcription factor is a conserved gene among different vertebrate species and is crucial for proper limb development (Post et al., 2000). The HOXA13 gene from rat was inserted into a mouse cell line (Pinglay et al., 2021). Using the standard reference sequence, alignment of Illumina genome sequence reads provided inflated read coverage and incorrect variant data due to reads from the exogenous rat HOXA13 incorrectly mapping to the endogenous mouse gene (Fig. 2 Upper Panel). In addition, many reads were unmapped due to the absence of the sequence in the reference genome. By contrast, following modification of the reference files using *ref*orm, the same input read data mapped to the correct, heterologous gene sequence (Fig. 2 Lower Panel), and sequence variants were accurately mapped. In addition, the use of the reformed reference sequence and annotation data allowed for the creation of a custom genome in IGV enabling the accurate visualization of reads aligned to the inserted rat gene.

Sequence variants are typically identified using specialized software, many of which depend on reference genome data as input, such as Samtools mpileup (Li et al., 2009), GATK HaplotypeCaller, and freebayes (Garrison and Marth, 2012). Using the same set of input reads for variant analysis in the endogenous mouse HOXA13 region with GATK HaplotypeCaller 3.7 resulted in 73 SNPs and 3 indels being identified when using the original, unreformed mouse reference in the analysis. By contrast, the same pipeline detected no variants when using the corrected reference data (Fig. 2 Middle Panel).

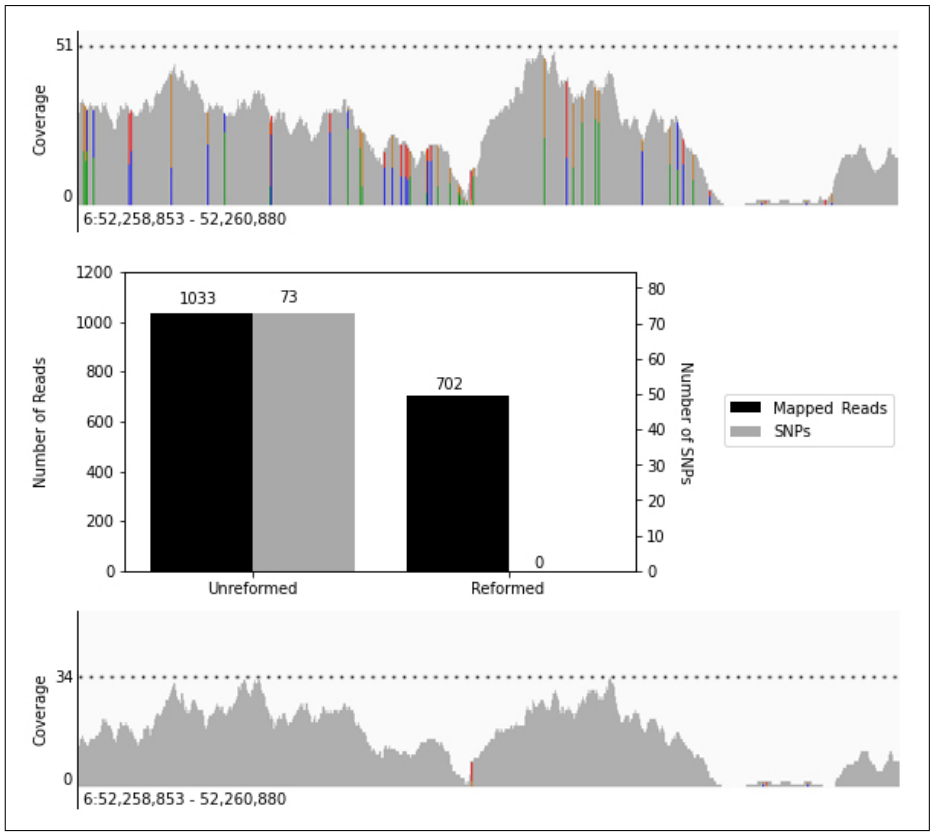

Figure 2: Upper Panel: Increased coverage and variation as a result of Illumina sequence reads from the inserted rat HOXA13 DNA incorrectly mapping to endogenous gene when using uncorrected reference data. Middle Panel: Number of reads mapped and SNPs detected in endogenous HOXA13 region using reformed vs unreformed reference data. Lower Panel: Use of the modified reference genome sequence results in more accurate read mapping at endogenous site.

3.2 Correcting reference data enables accurate CNV breakpoint analysis

Increased read depth in particular regions relative to the rest of the genome can indicate Copy Number Variations (CNVs) in the genome being studied. The start and end positions of these regions define the breakpoints of the CNV. However, when studying CNVs in edited genomes, the use of reference genome data that does not incorporate genome modifications can lead to incorrect breakpoint identification. Figure 5 shows an alignment of Oxford Nanopore Technology Minion reads to the *Saccharomyces cerevisiae* genome and variant calls using sniffles (Sedlazeck et al., 2018). The sharp coverage drop at approximately position 358kb and incongruent sequence suggest this position corresponds to a breakpoint. However, using *ref*orm to correct the reference sequence by incorporation of a heterologous construct that lies at the breakpoint reveals the correct position of the CNV (Fig. 3).

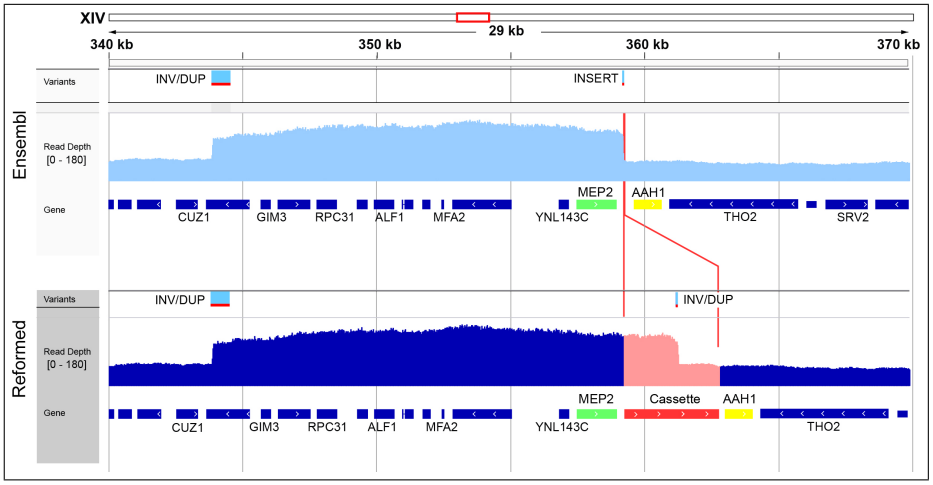

Figure 3: Schematic showing the read distribution and variant calls when aligned against the Ensembl reference genome (light blue, top track) versus those generated when aligned against the reformed reference genome (dark blue, bottom track). The increase in read depth starting near CUZ1 and ending near MEP2 is indicative of a CNV. However, when the Ensembl reference genome is used, the variant caller is only able to resolve the left INV/DUP of the CNV breakpoint. The right breakpoint is incorrectly called as an insertion due to the presence of a heterologous Cassette that had been integrated at the site (Red box and lines). Using *ref*orm to modify the reference genome, the variant caller correctly identifies that the CNV breakpoint corresponds to a INV/DUP that lies within the cassette sequence.

4 Conclusion

As the field of genome engineering advances, an increasing number of whole genome sequencing experiments are performed on modified genomes. Analyzing modified genomes accurately requires modifying the reference data used in the analysis, which is time consuming and prone to error. *ref*orm simplifies this task providing a rapid and reproducible solution, thereby allowing researchers to focus on downstream analyses.

One current limitation of *ref*orm is that it allows a single modification. For genomes that have multiple modifications, we recommend performing sequential modifications using *ref*orm.

5 Acknowledgements

The authors would like to thank Grace Avecilla, Milica Bulajic, Erika Levine, NYU Gencore, and members of the Department of Biology and Center for Genomics and Systems Biology who have used *ref*orm and provided valuable feedback.

6 Data availability

All code associated with this publication is available at:
https://github.com/gencorefacility/reform and
https://github.com/gencorefacility/reformWeb

All data associated with this publication is available at:
https://doi.org/10.17605/OSF.IO/5KS7R

7 Funding

This work was supported by an award to DG from the National Science Foundation (MCB 1818234) and The Zegar Family Foundation.

References

Buels, R., Yao, E., Diesh, C.M., Hayes, R.D., Munoz-Torres, M., Helt, G., Goodstein, D.M., Elsik, C.G., Lewis, S.E., Stein, L., Holmes, I.H., 2016. JBrowse: a dynamic web platform for genome visualization and analysis. Genome Biol. 17, 66.

Danecek, P., Auton, A., Abecasis, G., Albers, C.A., Banks, E., DePristo, M.A., Handsaker, R.E., Lunter, G., Marth, G.T., Sherry, S.T., McVean, G., Durbin, R., 1000 Genomes Project Analysis Group, 2011. The variant call format and VCFtools. Bioinformatics 27, 2156–2158.

DePristo, M.A., Banks, E., Poplin, R., Garimella, K.V., Maguire, J.R., Hartl, C., Philippakis, A.A., del Angel, G., Rivas, M.A., Hanna, M., McKenna, A., Fennell, T.J., Kernytsky, A.M., Sivachenko, A.Y., Cibulskis, K., Gabriel, S.B., Altshuler, D., Daly, M.J., 2011. A framework for variation discovery and genotyping using next-generation DNA sequencing data. Nat. Genet. 43, 491–498.

Dobin, A., Davis, C.A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., Gingeras, T.R., 2013. STAR: ultrafast universal RNA-seq aligner. Bioinformatics 29, 15–21.

Garrison, E., Marth, G., 2012. Haplotype-based variant detection from short-read sequencing. `arXiv:1207.3907`.

Haeussler, M., Zweig, A.S., Tyner, C., Speir, M.L., Rosenbloom, K.R., Raney, B.J., Lee, C.M., Lee, B.T., Hinrichs, A.S., Gonzalez, J.N., Gibson, D., Diekhans, M., Clawson, H., Casper, J., Barber, G.P., Haussler, D., Kuhn, R.M., Kent, W.J., 2019. The UCSC genome browser database: 2019 update. Nucleic Acids Res. 47, D853–D858.

Kim, D., Langmead, B., Salzberg, S.L., 2015. HISAT: a fast spliced aligner with low memory requirements. Nat. Methods 12, 357–360.

Li, H., Durbin, R., 2009. Fast and accurate short read alignment with Burrows-Wheeler transform. Bioinformatics 25, 1754–1760.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R., 1000 Genome Project Data Processing Subgroup, 2009. The sequence Alignment/Map format and SAMtools. Bioinformatics 25, 2078–2079.

NCBI Resource Coordinators, 2016. Database resources of the national center for biotechnology information. Nucleic Acids Res. 44, D7–19.

Pinglay, S., Bulajić, M., Rahe, D.P., Huang, E., Brosh, R., German, S., Cadley, J.A., Rieber, L., Easo, N., Mahony, S., Maurano, M.T., Holt, L.J., Mazzoni, E.O., Boeke, J.D., 2021. Synthetic genomic reconstitution reveals principles of mammalian hox cluster regulation. bioRxiv URL: `https://www.biorxiv.org/content/early/2021/07/07/2021.07.07.451065`, doi:10.1101/2021.07.07.451065.

Post, L.C., Margulies, E.H., Kuo, A., Innis, J.W., 2000. Severe limb defects in hypodactyly mice result from the expression of a novel, mutant HOXA13 protein. Dev. Biol. 217, 290–300.

Robinson, J.T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E.S., Getz, G., Mesirov, J.P., 2011. Integrative genomics viewer.

Sedlazeck, F.J., Rescheneder, P., Smolka, M., Fang, H., Nattestad, M., von Haeseler, A., Schatz, M.C., 2018. Accurate detection of complex structural variations using single-molecule sequencing. Nat. Methods 15, 461–468.

Zerbino, D.R., Achuthan, P., Akanni, W., Amode, M.R., Barrell, D., Bhai, J., Billis, K., Cummins, C., Gall, A., Girón, C.G., Gil, L., Gordon, L., Haggerty, L., Haskell, E., Hourlier, T., Izuogu, O.G., Janacek, S.H., Juettemann, T., To, J.K., Laird, M.R., Lavidas, I., Liu, Z., Loveland, J.E., Maurel, T., McLaren, W., Moore, B., Mudge, J., Murphy, D.N., Newman, V., Nuhn, M., Ogeh, D., Ong, C.K., Parker, A., Patricio, M., Riat, H.S., Schuilenburg, H., Sheppard, D., Sparrow, H., Taylor, K., Thormann, A., Vullo, A., Walts, B., Zadissa, A., Frankish, A., Hunt, S.E., Kostadima, M., Langridge, N., Martin, F.J., Muffato, M., Perry, E., Ruffier, M., Staines, D.M., Trevanion, S.J., Aken, B.L., Cunningham, F., Yates, A., Flicek, P., 2018. Ensembl 2018. Nucleic Acids Res. 46, D754–D761.