MitoChime: A Machine-Learning Pipeline for Detecting PCR-Induced Chimeras in Mitochondrial Illumina Reads

A Special Project Proposal

Presented to

the Faculty of the Division of Physical Sciences and Mathematics

College of Arts and Sciences

University of the Philippines Visayas

Miag-ao, Iloilo

In Partial Fulfillment

of the Requirements for the Degree of

Bachelor of Science in Computer Science

by

DURAN, Duranne

LIN, Yvonne

PAILDEN, Daniella

Adviser

Francis DIMZON

October 24, 2025

Contents

List of Figures

List of Tables

Chapter 1

0.1 Introduction

0.1.1 Overview

The rapid advancement of next-generation sequencing (NGS) technologies has transformed genomic research by enabling high-throughput and cost-effective DNA analysis (Metzker, 2010). Among current platforms, Illumina sequencing remains the most widely adopted, capable of producing millions of short reads that can be assembled into reference genomes or analyzed for genetic variation (Bentley et al., 2008; Glenn, 2011). Despite its high base-calling accuracy, Illumina sequencing is prone to artifacts introduced during library preparation, particularly polymerase chain reaction (PCR)-induced chimeras, which are artificial hybrid sequences that do not exist in the true genome (Judo, Wedel, & Wilson, 1998).

PCR chimeras form when incomplete extension products from one template anneal to an unrelated DNA fragment and are extended, creating recombinant reads (Qiu et al., 2001). In mitochondrial genome assembly, such artifacts are especially problematic because the mitochondrial genome is small, circular, and often repetitive (Boore, 1999; Cameron, 2014). Even a small number of chimeric or mis-joined reads can reduce assembly contiguity and introduce false junctions during organelle genome reconstruction (Dierckxsens, Mardulyn, & Smits, 2017; Hahn, Bachmann, & Chevreux, 2013; Jin et al., 2020). Existing assembly tools such as GetOrganelle and MITObim assume that input reads are largely free of

such artifacts (Hahn et al., 2013; Jin et al., 2020). Consequently, undetected chimeras may produce fragmented assemblies or misidentified organellar boundaries. To ensure accurate reconstruction of mitochondrial genomes, a reliable and automated method for detecting and filtering PCR-induced chimeras before assembly is essential.

This study focuses on mitochondrial sequencing data from the genus *Sardinella*, a group of small pelagic fishes widely distributed in Philippine waters. Among them, *Sardinella lemuru* (Bali sardinella) is one of the country's most abundant and economically important species, providing protein and livelihood to coastal communities (Labrador, Agmata, Palermo, Ravago-Gotanco, & Pante, 2021; Willette, Bognot, Mutia, & Santos, 2011). Accurate mitochondrial assemblies are critical for understanding its population genetics, stock structure, and evolutionary history. However, assembly pipelines often encounter errors or fail to complete due to undetected chimeric reads. To address this gap, this research introduces **MitoChime**, a machine-learning pipeline designed to detect and filter PCR-induced chimeric reads using both alignment- and sequence-derived statistical features. The tool aims to provide bioinformatics laboratories, particularly the Philippine Genome Center Visayas, with an efficient, interpretable, and resource-optimized solution for improving mitochondrial genome reconstruction.

0.1.2 Problem Statement

While NGS technologies have revolutionized genomic data acquisition, the accuracy of mitochondrial genome assembly remains limited by artifacts produced during PCR amplification. These chimeric reads can distort assembly graphs and

cause misassemblies, with especially severe effects in small, circular mitochondrial genomesBoore (1999); Cameron (2014). Existing assembly pipelines such as GetOrganelle, MITObim, and NOVOPlasty assume that sequencing reads are free of such artifactsDierckxsens et al. (2017); Hahn et al. (2013); Jin et al. (2020). At the Philippine Genome Center Visayas, several mitochondrial assemblies have failed or yielded incomplete contigs despite sufficient coverage, suggesting that undetected chimeric reads compromise assembly reliability. Meanwhile, existing chimera-detection tools such as UCHIME and VSEARCH were developed primarily for amplicon-based microbial community analysis and rely heavily on reference or taxonomic comparisonsEdgar, Haas, Clemente, Quince, and Knight (2011); Rognes, Flouri, Nichols, Quince, and Mahé (2016). These approaches are unsuitable for single-species organellar data, where complete reference genomes are often unavailable. Therefore, there is a pressing need for a reference-independent, data-driven tool capable of automatically detecting and filtering PCR-induced chimeras in mitochondrial sequencing datasets.

0.1.3 Research Objectives

General Objective

To develop and evaluate a machine-learning-based pipeline (MitoChime) capable of detecting PCR-induced chimeric reads in *Sardinella* mitochondrial sequencing data to improve the accuracy of mitochondrial genome assembly.

Specific Objectives

Specifically, the researchers aim to:

1. Construct simulated and empirical *Sardinella* Illumina paired-end datasets containing both clean and PCR-induced chimeric reads.
2. Extract alignment- and sequence-based features (e.g., k-mer composition, junction complexity, split-alignment counts) from both clean and chimeric reads.
3. Train, validate, and compare supervised machine-learning models (e.g., Random Forest, XGBoost) for classifying reads as clean or chimeric.
4. Determine feature importance and identify the most informative indicators of PCR-induced chimerism.
5. Integrate the optimized classifier into a modular and interpretable pipeline deployable on standard computing environments at PGC Visayas.

0.1.4 Scope and Limitations of the Research

This study focuses on detecting PCR-induced chimeric reads in Illumina paired-end mitochondrial sequencing data from *Sardinella* species. The work emphasizes `wgsim` simulations and selected empirical data obtained from open-access genomic repositories such as the National Center for Biotechnology Information (NCBI). The study excludes naturally occurring chimeras, nuclear mitochondrial pseudogenes (NUMTs), and large-scale structural rearrangements in nuclear genomes.

Feature extraction prioritizes interpretable, shallow statistics and alignment metrics rather than deep-learning embeddings to ensure transparency and computational efficiency. Testing on long-read platforms (e.g., Nanopore, PacBio) and other taxa lies beyond the project's scope. The resulting pipeline will serve as a foundation for future, broader chimera-detection frameworks applicable to diverse organellar genomes.

0.1.5 Significance of the Research

This research provides both methodological and practical contributions to mitochondrial genomics and bioinformatics. First, MitoChime enhances assembly accuracy by filtering PCR-induced chimeras prior to genome assembly, thereby improving the contiguity and correctness of *Sardinella* mitochondrial genomes. Second, it promotes automation and reproducibility by replacing subjective manual curation with a data-driven, machine-learning-based workflow. Third, the pipeline demonstrates computational efficiency through its design, enabling implementation on modest computing infrastructures commonly available in regional laboratories. Beyond technical improvements, MitoChime contributes to local capacity building by strengthening expertise in bioinformatics and machine-learning integration, aligning with the mission of the Philippine Genome Center Visayas. Finally, accurate mitochondrial assemblies are vital for fisheries management, population genetics, and biodiversity conservation, providing reliable genomic resources for species such as *Sardinella*. Through these contributions, MitoChime advances the reliability of mitochondrial genome reconstruction and supports sustainable, data-driven research in Philippine genomics.

Chapter 3

0.2 Research Methodology

This chapter outlines and explains the specific steps and activities to be carried out in completing the project.

0.2.1 Research Activities

As illustrated in Figure 1, the researchers will carry out a sequence of computational procedures designed to detect PCR-induced chimeric reads in mitochondrial genomes. The process begins with the collection of mitochondrial reference sequences from the NCBI database, which will serve as the foundation for generating simulated chimeric reads. These datasets will then undergo bioinformatics pipeline development, which includes alignment, k-mer extraction, and homology-based filtering to prepare the data for model construction. The machine-learning model will subsequently be trained and tested using the processed datasets to assess its accuracy and reliability. Depending on the evaluation results, the model will either be refined and retrained to improve performance or, if the metrics meet the desired threshold, deployed for further validation and application.

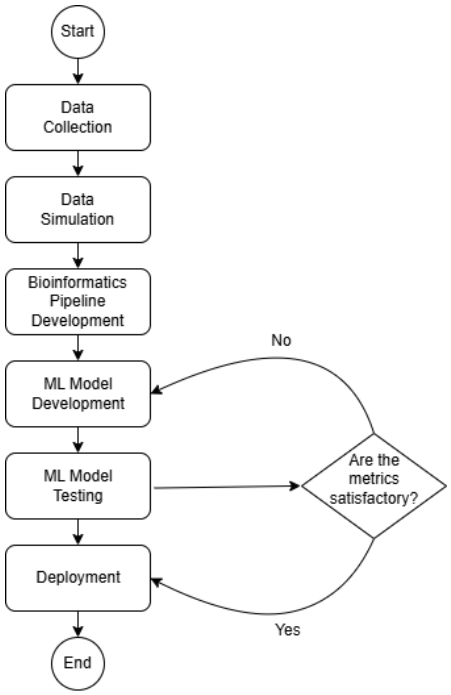

Figure 1: Process diagram of the special project.

Data Collection

The researchers will collect mitochondrial genome reference sequences of *Sardinella lemuru* from the National Center for Biotechnology Information (NCBI) database. The downloaded files will be in FASTA format to ensure compatibility with bioinformatics tools and subsequent analysis. The gathered sequences will serve as the basis for generating simulated chimeric reads to be used in model development.

The expected outcome of this process is a comprehensive dataset of *Sardinella lemuru* mitochondrial reference sequences that will serve as the foundation for the succeeding stages of the study. This step is scheduled to start in the first week of November 2025 and is expected to be completed by the last week of

November 2025, with a total duration of approximately one (1) month.

Data Simulation

The researchers will simulate sequencing data using the reference sequences collected from NCBI. Using `wgsim`, a total of 5,000 paired-end reads (R1 and R2) will be generated from the reference genome and designated as clean reads. These reads will be saved in FASTQ (`.fastq`) format. From the same reference, a Bash script will be created to deliberately cut and reconnect portions of the sequence, introducing artificial junctions that mimic chimeric regions. The manipulated reference file, saved in FASTA (`.fasta`) format, will then be processed in `wgsim` to simulate an additional 5,000 paired-end chimeric reads, also stored in FASTQ (`.fastq`) format. The resulting read files will be aligned to the original reference genome using SAMtools, generating SAM (`.sam`) or BAM (`.bam`) alignment files. During this alignment process, clean reads will be labeled as "0," while chimeric reads will be labeled as "1" in a corresponding CSV (`.csv`) file.

The expected outcome of this process is a complete set of clean and chimeric paired-end reads prepared for subsequent analysis and model development. This step is scheduled to start in the first week of November 2025 and is expected to be completed by the last week of November 2025, with a total duration of approximately one (1) month.

Bioinformatics Tools Pipeline

The researchers will obtain the necessary analytical features through the development and implementation of a bioinformatics pipeline. This pipeline will serve as a reproducible and modular workflow that accepts FASTQ and BAM inputs, processes these through a series of analytical stages, and outputs tabular feature matrices (TSV/CSV) for downstream machine learning. All scripts will be version-controlled through GitHub, and computational environments will be standardized using Conda to ensure cross-platform reproducibility. To promote transparency and replicability, the exact software versions, parameters, and command-line arguments used in each stage will be documented. To further ensure correctness and adherence to best practices, the researchers will consult with bioinformatics experts in Philippine Genome Center Visayas for validation of pipeline design, feature extraction logic, and overall data integrity. This stage of the study is scheduled to begin in the last week of November 2025 and conclude by the last week of January 2026, with an estimated total duration of approximately two (2) months.

The bioinformatics pipeline focuses on three principal features from the simulated and aligned sequencing data: (1) supplementary alignment count (SA count), (2) k-mer composition difference between read segments, and (3) microhomology length at potential junctions. Each of these features captures a distinct biological or computational signature associated with PCR-induced chimeras.

Alignment and Supplementary Alignment Count

This will be derived through sequence alignment using Minimap2, with subsequent

processing performed using SAMtools and `pysam` in Python. Sequencing reads will be aligned to the *Sardinella lemuru* mitochondrial reference genome using Minimap2 with the `-ax sr` preset (optimized for short reads). The output will be converted and sorted using SAMtools, producing an indexed BAM file which will be parsed using `pysam` to count the number of supplementary alignments (SA tags) per read. Each read's mapping quality, number of split segments, and alignment characteristics will be recorded in a corresponding TSV file. The presence of multiple alignment loci within a single read, as reflected by a nonzero SA count, serves as direct computational evidence of chimerism. Reads that contain supplementary alignments or soft-clipped regions are strong candidates for chimeric artifacts arising from PCR template switching or improper assembly during sequencing.

K-mer Composition Difference

Chimeric reads often comprise fragments from distinct genomic regions, resulting in a compositional discontinuity between segments. Comparing k-mer frequency profiles between the left and right halves of a read allows detection of such abrupt compositional shifts, independent of alignment information. This will be obtained using Jellyfish, a fast k-mer counting software. For each read, the sequence will be divided into two segments, either at the midpoint or at empirically determined breakpoints inferred from supplementary alignment data, to generate left and right sequence segments. Jellyfish will then compute k-mer frequency profiles (with $k =$ 5 or 6) for each segment. The resulting k-mer frequency vectors will be normalized and compared using distance metrics such as cosine similarity or Jensen–Shannon divergence to quantify compositional disparity between the two halves of the same

read. The resulting difference scores will be stored in a structured TSV file.

Micro-homology Length

The micro-homology length will be computed using a custom Python script that detects the longest exact suffix–prefix overlap within ±30 base pairs surrounding a candidate breakpoint. This analysis identifies the number of consecutive bases shared between the end of one segment and the beginning of another. The presence and length of such micro-homology are classic molecular signatures of PCR-induced template switching, where short identical regions (typically 3–15 base pairs) promote premature termination and recombination of DNA synthesis on a different template strand. By quantifying micro-homology, the researchers can assess whether the suspected breakpoint exhibits characteristics consistent with PCR artifacts rather than true biological variants. Each read will therefore be annotated with its corresponding micro-homology length, overlap sequence, and GC content.

After extracting the three primary features, all resulting TSV files will be joined using the read identifier as a common key to generate a unified feature matrix. Additional read-level metadata such as read length, mean base quality, and number of clipped bases will also be included to provide contextual information. This consolidated dataset will serve as the input for subsequent machine-learning model development and evaluation.

Machine-Learning Model Development

The classification component of MitoChime will employ two ensemble algorithms—Random Forest (RF) and Extreme Gradient Boosting (XGBoost)—to evaluate complementary learning paradigms. Random Forest applies bootstrap aggregation (bagging) to reduce model variance and improve stability, whereas XGBoost implements gradient boosting to minimize bias and capture complex non-linear relationships among genomic features. Using both models enables a balanced assessment of predictive performance and interpretability.

The dataset will be divided into training (80%) and testing (20%) subsets. The training data will be used for model fitting and hyperparameter optimization through five-fold cross-validation, in which the data are partitioned into five folds; four folds are used for training and one for validation in each iteration. Performance metrics will be averaged across folds, and the optimal parameters will be selected based on mean cross-validation accuracy. The final models will then be evaluated on the held-out test set to obtain unbiased performance estimates.

Model development and evaluation will be implemented in Python (version 3.11) using the `scikit-learn` and `xgboost` libraries. Standard metrics including accuracy, precision, recall, F1-score, and area under the ROC curve (AUC) will be computed to quantify predictive performance. Feature-importance analyses will be performed to identify the most discriminative variables contributing to chimera detection.

Validation and Testing

Validation will involve both internal and external evaluations. Internal validation will be achieved through five-fold cross-validation on the training data to verify model generalization and reduce variance due to random sampling. External validation will be achieved through testing on the 20% hold-out dataset derived from the simulated reads, which will serve as an unbiased benchmark to evaluate how well the trained models generalize to unseen data. All feature extraction and preprocessing steps will be performed using the same bioinformatics pipeline to ensure consistency and comparability across validation stages.

Comparative evaluation between the Random Forest and XGBoost classifiers will establish which model achieves superior predictive accuracy and computational efficiency under identical data conditions.

Documentation

Comprehensive documentation will be maintained throughout the study to ensure transparency, reproducibility, and scientific integrity. All stages of the research—including data acquisition, preprocessing, feature extraction, model training, and validation—will be systematically recorded. For each analytical step, the corresponding parameters, software versions, and command-line scripts will be documented to enable exact replication of results.

Version control and collaborative management will be implemented through GitHub, which will serve as the central repository for all project files, including Python scripts, configuration settings, and Jupyter notebooks. The repository

structure will follow standard research data management practices, with clear directories for datasets, processed outputs, and analysis scripts. Changes will be tracked through commit histories to ensure traceability and accountability.

Computational environments will be standardized using Conda, with environment files specifying dependencies and package versions to maintain consistency across systems. Experimental workflows and exploratory analyses will be conducted in Jupyter Notebooks, which facilitate real-time visualization, annotation, and incremental testing of results.

For the preparation of the final manuscript and supplementary materials, Overleaf (LaTeX) will be utilized to produce publication-quality formatting, consistent referencing, and reproducible document compilation. The documentation process will also include a project timeline outlining major milestones such as data collection, simulation, feature extraction, model evaluation, and reporting to ensure systematic progress and adherence to the research schedule.

0.2.2 Calendar of Activities

Table 1 presents the project timeline in the form of a Gantt chart, where each bullet point corresponds to approximately one week of planned activity.

Table 1: Timetable of Activities

Activities (2025)	**Nov**	**Dec**	**Jan**	**Feb**	**Mar**	**Apr**	**May**
Data Collection and Simulation	●●●●						
Bioinformatics Tools Pipeline	●●	●●●●	●●●●				
Machine Learning Development			●●	●●●●	●●●●	●●	
Testing and Validation						●●	●●●●
Documentation	●●●●	●●●●	●●●●	●●●●	●●●●	●●●●	●●●●

References

Bentley, D. R., Balasubramanian, S., Swerdlow, H. P., Smith, G. P., Milton, J., Brown, C. G., ... Smith, A. J. (2008). Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*, *456*(7218), 53–59. doi: 10.1038/nature07517

Boore, J. L. (1999). Animal mitochondrial genomes. *Nucleic Acids Research*, *27*(8), 1767–1780. doi: 10.1093/nar/27.8.1767

Cameron, S. L. (2014). Insect mitochondrial genomics: Implications for evolution and phylogeny. *Annual Review of Entomology*, *59*, 95–117. doi: 10.1146/annurev-ento-011613-162007

Dierckxsens, N., Mardulyn, P., & Smits, G. (2017). Novoplasty: de novo assembly of organelle genomes from whole genome data. *Nucleic Acids Research*, *45*(4), e18. doi: 10.1093/nar/gkw955

Edgar, R. C., Haas, B. J., Clemente, J. C., Quince, C., & Knight, R. (2011). Uchime improves sensitivity and speed of chimera detection. *Bioinformatics*, *27*(16), 2194–2200. doi: 10.1093/bioinformatics/btr381

Glenn, T. C. (2011). Field guide to next-generation dna sequencers. *Molecular Ecology Resources*, *11*(5), 759–769. doi: 10.1111/j.1755-0998.2011.03024.x

Hahn, C., Bachmann, L., & Chevreux, B. (2013). Reconstructing mitochondrial

genomes directly from genomic next-generation sequencing reads—a baiting and iterative mapping approach. *Nucleic Acids Research*, *41*(13), e129. doi: 10.1093/nar/gkt371

Jin, J.-J., Yu, W.-B., Yang, J., Song, Y., dePamphilis, C. W., Yi, T.-S., & Li, D.-Z. (2020). Getorganelle: a fast and versatile toolkit for accurate de novo assembly of organelle genomes. *Genome Biology*, *21*(1), 241. doi: 10.1186/s13059-020-02154-5

Judo, M. S. B., Wedel, W. R., & Wilson, B. H. (1998). Stimulation and suppression of pcr-mediated recombination. *Nucleic Acids Research*, *26*(7), 1819–1825. doi: 10.1093/nar/26.7.1819

Labrador, K., Agmata, A., Palermo, J. D., Ravago-Gotanco, R., & Pante, M. J. (2021). Mitochondrial dna reveals genetically structured haplogroups of bali sardinella (sardinella lemuru) in philippine waters. *Regional Studies in Marine Science*, *41*, 101588. doi: 10.1016/j.rsma.2020.101588

Metzker, M. L. (2010). Sequencing technologies — the next generation. *Nature Reviews Genetics*, *11*(1), 31–46. doi: 10.1038/nrg2626

Qiu, X., Wu, L., Huang, H., McDonel, P. E., Palumbo, A. V., Tiedje, J. M., & Zhou, J. (2001). Evaluation of pcr-generated chimeras, mutations, and heteroduplexes with 16s rrna gene-based cloning. *Applied and Environmental Microbiology*, *67*(2), 880–887. doi: 10.1128/AEM.67.2.880-887.2001

Rognes, T., Flouri, T., Nichols, B., Quince, C., & Mahé, F. (2016). Vsearch: a versatile open source tool for metagenomics. *PeerJ*, *4*, e2584. doi: 10.7717/peerj.2584

Willette, D., Bognot, E., Mutia, M. T., & Santos, M. (2011). *Biology and ecology of sardines in the philippines: A review* (Vol. 13; Tech. Rep. No. 1). NFRDI Technical Paper Series. Retrieved from `https://nfrdi.da.gov.ph/tpjf/`

362 etc/Willette%20et%20al.%20Sardines%20Review.pdf