# ModkitOpt: Systematic optimisation of modkit parameters for accurate nanopore-based RNA modification detection

Alexandra Sneddon[1,2,3,4,*], Stefan Prodic[1,2,3], Eduardo Eyras[1,2,3,4]

1 EMBL Australia Partner Laboratory Network at the Australian National University, Canberra ACT 2601, Australia.
2 Centre for Computational Biomedical Sciences, The John Curtin School of Medical Research, Australian National University, Canberra ACT 2601, Australia.
3 The Shine-Dalgarno Centre for RNA Innovation, The John Curtin School of Medical Research, Australian National University, Canberra ACT 2601, Australia.
4 The Australian Research Council Centre of Excellence for the Mathematical Analysis of Cellular Systems, Australia.

*Correspondence to: Alexandra.Sneddon@anu.edu.au

## Abstract

Nanopore direct RNA sequencing enables single-molecule detection of RNA modifications, but accurate quantification relies on filtering out the modification caller's erroneous predictions. The default filtering strategy in Oxford Nanopore Technologies' modkit software is heuristic and often leads to inaccurate reporting of modification sites. We introduce ModkitOpt, a computational method for identifying the optimal modkit filtering parameters to achieve accurate detection of modification sites in any given sample. ModkitOpt systematically scans modkit filtering parameters (*--filter-threshold* and *--mod-threshold*) and stoichiometry cutoffs to classify whether a site is modified, computes precision and recall by comparing predictions against a reference set of known sites, and identifies the parameter combination that optimises the precision-sensitivity trade-off. Using mammalian N6-methyladenosine and pseudouridine datasets, we show that modkit's default parameters are frequently suboptimal and that ModkitOpt identifies optimal parameter choices that are robust across biological contexts. ModkitOpt is implemented as an efficient Nextflow workflow, available at https://github.com/comprna/modkitopt, to enable accurate reporting of nanopore RNA modification sites.

## Introduction

Nanopore sequencing has revolutionised epitranscriptome analysis by enabling direct, single-molecule measurement of native RNA[1]. As RNA transits through nanopores, biochemical modifications cause characteristic changes in the measured ionic current signal, which computational tools decode to identify modifications at single-nucleotide resolution[2–5]. Oxford Nanopore Technologies (ONT) provides dorado[6] for calling modified bases per read, and modkit[7] for aggregating per-read predictions into site-level stoichiometry estimates, defined as the

fraction of transcripts that are modified at a given reference position. In downstream analyses, a stoichiometry cutoff is typically applied to modkit's site-level estimates to classify each site as modified or unmodified. To enable accurate biological interpretation, it is crucial that modification stoichiometry is correctly quantified and an appropriate stoichiometry cutoff applied. However, modkit's default parameters are estimated heuristically from the data, without any assessment of their impact on site-level accuracy, and no quantitative approach currently exists for identifying the optimal stoichiometry cutoff for modification site detection.

Before computing stoichiometry, modkit *pileup* filters out low-confidence modification calls made by dorado. By default, modkit estimates a confidence threshold by sampling predictions from a subset of reads and defining confidence as the maximum predicted probability (modified or non-modified) for each call. The threshold is then set to the 10$^{th}$ percentile of these confidence values. Since this percentile-based threshold is not based on prediction accuracy, datasets dominated by low-confidence calls, such as those for rare modifications like pseudouridine (pseU), will be assigned an insufficiently stringent threshold, resulting in elevated false discovery rates[8]. Conversely, datasets dominated by high-confidence calls will yield an overly stringent threshold, resulting in true modification sites being erroneously excluded. Although users may override the default behaviour by manually specifying confidence thresholds for canonical bases and modifications (*--filter-threshold* and *--mod-threshold,* respectively), this process relies on subjective interpretation of probability distributions output by modkit's *sample-probs*. Selection of both the modkit thresholds and downstream stoichiometry cutoff lack a quantitative framework for optimising the precision-sensitivity trade-off.

Here, we introduce ModkitOpt, a tool for identifying the optimal modkit filtering parameters (*--filter-threshold* and *--mod-threshold*) and downstream stoichiometry cutoff for a given nanopore dataset. ModkitOpt systematically scans a comprehensive range of candidate thresholds and stoichiometry cutoffs and evaluates the predicted modification sites against a reference set of known modification sites to quantify precision and recall. Using this strategy, we demonstrate that modification-calling performance is highly sensitive to threshold choice and that the default thresholds selected by modkit are often suboptimal. We further show that the optimal stoichiometry cutoff varies across modkit thresholds. We provide reference sites for mammalian N6-methyladenosine (m6A) and pseU and show that, even when reference sites are derived from a different biological context than the nanopore data, the relative performance of modkit across the threshold space is preserved. Consequently, ModkitOpt can be applied to nanopore datasets from any experiment using the provided references, or with user-supplied reference sites for other modification types, and is made available as an efficient Nextflow pipeline.

## Implementation

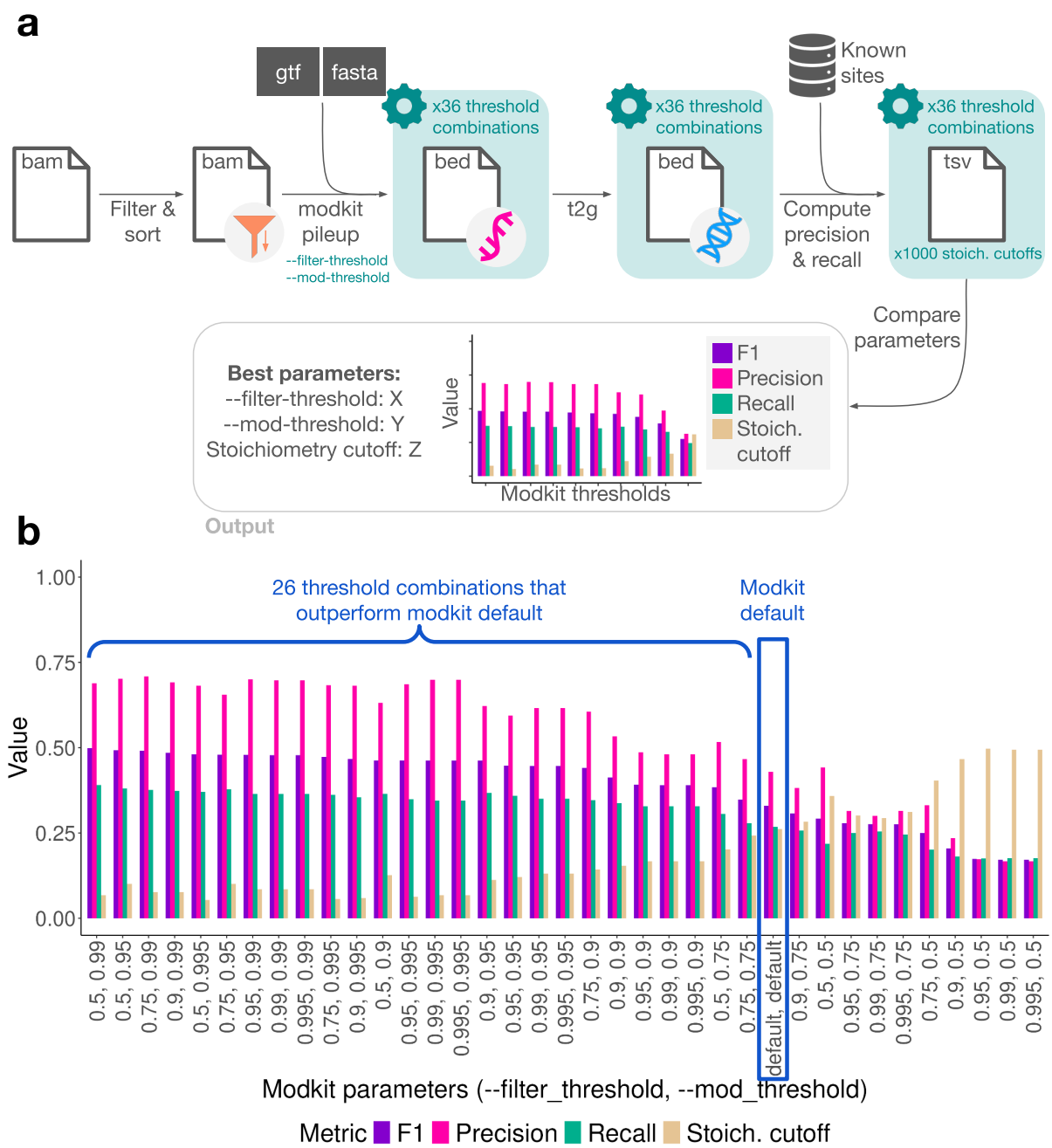


**Figure 1: a,** Graphical summary of the ModkitOpt pipeline. A transcriptome-aligned modBAM file containing per-read modification calls is filtered and sorted. Modkit *pileup* is then run across 36 combinations of *--filter-threshold* and *--mod-threshold* to generate a bedMethyl file containing stoichiometry estimates per transcript site. Transcript sites are converted to genomic coordinates, thenprecision and recall calculated against a set of known sites, across 1,000 candidate stoichiometry cutoffs uniformly spaced between 0 and 0.999. The performance of the candidate thresholds is plotted, and the parameters that maximise the F1 score are reported. **b,** Bar plot output by ModkitOpt executed on a modBAM file from HEK293T cells, using the supplied default m6A reference site set. Shown is the performance of modkit's modification calls (y-axis), in terms of F1 score (purple), precision (pink) and recall (green), across 36 threshold

combinations (x-axis). For each threshold combination, the plot shows the performance obtained at the stoichiometry cutoff (tan) that maximised F1 score**.**

ModkitOpt **(Figure 1a)** takes as input a transcriptome-aligned modBAM[9] file generated by dorado, along with a reference transcriptome FASTA file, GTF or GFF3 annotation, and a specified modification type. Input modBAM files are preprocessed to retain primary alignments only, followed by sorting and indexing using samtools (v1.21)[10]. Modkit (v0.6.0) pileup is subsequently executed across a predefined grid of filtering parameters, spanning all combinations of *--filter-threshold* and *--mod-threshold* values (0.5, 0.75, 0.9, 0.95, 0.99, 0.995). For each parameter combination, modkit produces transcriptome-coordinate bedMethyl files containing modification calls and estimated stoichiometries. We also allow advanced users to optionally specify the --filter-threshold and --mod-threshold values to test.Since reference sets of known modification sites are typically defined in genomic coordinates, predicted modification sites are converted from transcriptomic to genomic coordinates. When multiple transcriptomic positions map to the same genomic site, stoichiometry is aggregated using a weighted average across transcripts resulting genomic modification calls are then compared against the reference set to assess prediction accuracy. Precision (fraction of the predicted sites that are in the reference set), recall (fraction of the reference sites that are predicted) and F1 score ($2 \cdot \frac{(Pecision \cdot Recall)}{(Pecision + Recall)}$) are computed across a range of stoichiometry cutoffs. ModkitOpt systematically tests 1,000 stoichiometry cutoffs uniformly spaced between 0 and 0.999. At each cutoff, nanopore-called sites are retained if their estimated stoichiometry exceeds the cutoff. For each parameter combination, the stoichiometry cutoff that maximises the F1 score is identified, and performance across all parameter combinations is compared to determine the combination of *--filter-threshold, --mod-threshold,* and stoichiometry cutoff that achieves the highest F1 score.

ModkitOpt reports the optimal *--filter-threshold*, *--mod-threshold* and stoichiometry cutoff to the user and produces a bar plot depicting the precision, recall and F1 score across the parameter space (**Figure 1b**)[11]. The bar plot's data is also written in .tsv format. More detailed plots are additionally provided; a scatter plot visualising the F1 score across all combinations of *--mod-threshold* and *--filter-threshold,* using the optimal stoichiometry cutoff per threshold combination (**Supplementary Figure 1**); and the set of precision-recall curves for each threshold combination, annotated with the maximum F1 score per curve (**Supplementary Figure 2**). Reference site sets are provided for mammalian m6A and pseU, derived from GLORI-identified m6A sites in HEK293T cells[12] and BACS-identified pseU sites in HeLa cells[13], respectively. ModkitOpt is distributed as a Nextflow workflow, enabling reproducible execution on local machines or HPC systems, with resource allocation automatically managed for efficient and scalable analysis.

## Performance evaluation

We first assessed whether modkit *pileup's* default behaviour, whereby confidence thresholds are estimated heuristically from the data, yields optimal stoichiometry estimates. To ensure a fair comparison, these analyses were performed using modBAM files generated from the same cell line as the corresponding reference site sets, HEK293T for m6A[12] and HeLa for pseU[13]. Using ModkitOpt, we systematically evaluated sensitivity and precision across all 36 combinations of

--*filter-threshold* and --*mod-threshold,* and compared these results to those obtained using modkit's default implementation. We evaluated each threshold combination, as well as the modkit default, across each of the 1,000 stoichiometry cutoffs, and selected for downstream analysis the cutoffs achieving maximum F1 score. For both modification types, m6A and pseU, the default thresholds were suboptimal – for m6A, 26 of the 36 parameter combinations outperformed the default (**Figure 1b**), while 23 of the 36 parameter combinations were superior for pseU (**Supplementary Figure 3**). The optimal thresholds were --*filter-threshold=0.5* and --*mod-threshold=0.99* for m6A, and --*filter-threshold=0.5* and --*mod-threshold=0.995* for pseU, resulting in F1 score improvements of 51% and 1251%, respectively, relative to the default. The significant performance improvement for pseU reflects the particularly low confidence of pseU calls under default thresholds.

The F1 score, which quantifies the precision-recall trade-off, was dominated by --*mod_threshold*, with more stringent values increasing precision by reducing the number of false-positive sites, at the expense of recall, resulting in more false-negative site predictions. For m6A, moderately stringent --*mod_thresholds* outperformed both stricter thresholds that compromise recall, as well as looser thresholds that compromise precision. For pseU, the most stringent --*mod_threshold* was optimal, as expected given the overall lower confidence of pseU modification calls made by dorado, which necessitates stricter filtering. Overall, the optimal stoichiometry cutoff increased across threshold combinations as F1 score decreased, suggesting that looser --*mod_threshold* necessitates stricter stoichiometry cutoffs to counteract the increase in false predictions. The impact of --*filter_threshold* was secondary, effectively tuning the F1 score per --*mod_threshold,* such that, in general, a stricter --*filter_threshold* reduced the F1 score.

We next investigated whether ModkitOpt remains effective when the reference site set originates from a different biological context to the nanopore dataset. Although such reference sets may not fully overlap with the true modification landscape of a given sample, we reasoned that a shared subset of true sites should still be sufficient to identify thresholds that optimise the precision-recall trade-off. To test this, we evaluated modkit *pileup* across the same 36 threshold combinations and 1,000 stoichiometry cutoffs using nanopore datasets from HepG2 and K562 cells[8], while retaining the original reference site sets. For both m6A and pseU, the relative performance landscape across the threshold space was mostly conserved (**Supplementary Figures 4-7**), and the optimal --*filter-threshold* and --*mod-threshold* values matched those identified in the previous analyses using matched cell lines. These results indicate that ModkitOpt's supplied reference site sets can be used to optimise modkit parameters in mammalian nanopore datasets, even when the biological context differs from that of the reference, provided that a subset of modification sites is conserved.

Finally, we evaluated the computational performance of ModkitOpt. When executed on an HPC system (PBSPro Scheduler) using default resource settings (8 CPUs and 8GB RAM for samtools filter and samtools sort; 8 CPUs and 30GB RAM for modkit pileup; 4 CPUs and 8GB RAM for samtools index; 1 CPU, 30GB RAM and 5GB job filesystem disk space for converting transcriptomic to genomic coordinates; and 1 CPU and 8GB RAM for all other tasks), ModkitOpt completed in 22 minutes for a 7GB modBAM file and 33 minutes for a 21GB modBAM file (**Supplementary Table 1**), demonstrating that comprehensive parameter optimisation can be

performed efficiently on typical nanopore datasets.

## Conclusion

Accurate estimation of RNA modification stoichiometry is essential for extracting meaningful biological insight from nanopore direct RNA sequencing data. Here, we present ModkitOpt, a systematic and reproducible computational method for optimising modkit filtering thresholds based on quantitative evaluation of modification calls against reference modification sites. By exhaustively scanning the modkit filtering threshold space and identifying threshold combinations that maximise precision and sensitivity, ModkitOpt overcomes the limitations of modkit's heuristic default filtering approach and the subjective manual tuning of modkit thresholds.

Using mammalian m6A and pseU datasets, we demonstrate that modkit's default filtering thresholds are frequently suboptimal and that modification calling performance is highly sensitive to parameter choice. Importantly, we show that optimal parameter combinations are preserved even when reference site sets originate from a different biological context to the nanopore dataset, supporting the broad applicability of ModkitOpt across experiments. Implemented as an efficient Nextflow workflow, ModkitOpt integrates seamlessly into existing nanopore analysis pipelines and incurs minimal computational overhead. We anticipate that ModkitOpt will improve the accurate reporting of nanopore RNA modification calls, thereby enabling more reliable downstream analyses of RNA modifications across diverse biological and experimental contexts. ModkitOpt is available at https://github.com/comprna/modkitopt.

## References


1. Garalde, D. R. *et al.* Highly parallel direct RN A sequencing on an array of nanopores. *Nat Methods* 15, 201–206 (2018).

2. Leger, A. *et al.* RNA modifications detection by comparative Nanopore direct RNA sequencing. *Nature Communications 2021 12:1* 12, 7198- (2021).

3. Liu, H., Begik, O. & Novoa, E. M. EpiNano: Detection of m6A RNA Modifications Using Oxford Nanopore Direct RNA Sequencing. *Methods Mol Biol* 2298, 31–52 (2021).

4. Hendra, C. *et al.* Detection of m6A from direct RNA sequencing using a multiple instance learning framework. *Nature Methods 2022 19:12* 19, 1590–1598 (2022).

5. Acera Mateos, P. *et al.* Prediction of m6A and m5C at single-molecule resolution reveals a transcriptome-wide co-occurrence of RNA modifications. *Nature Communications 2024 15:1* 15, 1–17 (2024).

6. nanoporetech/dorado: Oxford Nanopore's Basecaller. https://github.com/nanoporetech/dorado.

7. nanoporetech/modkit: A bioinformatics tool for working with modified bases. https://github.com/nanoporetech/modkit.

8. Zou, Y. *et al.* A comparative evaluation of computational models for RNA modification detection using nanopore sequencing with RNA004 chemistry. *Brief Bioinform* 26, (2025).

9. Li, H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* 34, 3094–3100 (2018).

10. Li, H. *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics* 25, 2078–2079 (2009).

11. Wickham, H. *et al.* Welcome to the Tidyverse. *J Open Source Softw* 4, 1686 (2019).

12. Liu, C. *et al.* Absolute quantification of single-base m6A methylation in the mammalian transcriptome using GLORI. *Nat Biotechnol* 41, 355–366 (2023).

13. Xu, H. *et al.* Absolute quantitative and base-resolution sequencing reveals comprehensive landscape of pseudouridine across the human transcriptome. *Nat Methods* 21, 2024–2033 (2024).

## Funding

This research was supported by the Australian Research Council (ARC) Discovery Project grants DP250100070 and DP250103133; by the National Health and Medical Research Council (NHMRC) Ideas Grant 2018833; by a PhD scholarship and an Innovator Grant from the ANU TALO Computational Biology Talent Accelerator, and by the ARC Centre of Excellence for the Mathematical Analysis of Cell Systems (MACSYS) (CE230100001). This research was also indirectly supported by the Australian Government's National Collaborative Research Infrastructure Strategy (NCRIS) through access to computational resources provided by the National Computational Infrastructure (NCI) through the National Computational Merit Allocation Scheme (NCMAS) and the ANU Merit Allocation Scheme (ANUMAS). The funding bodies had no role in study design, data collection, or data analysis.

## Author Contributions

AS, SP and EE conceptualised the study. AS led the software development and data analysis, with essential contributions from SP & EE. AS wrote the manuscript, and SP and EE reviewed the manuscript.