Mapping of RNA modifications by direct Nanopore sequencing and JACUSA2

Christoph Dieterich*[1,2,3], Amina Lemsara[1,2], and Isabel Naarmann-de Vries[1,2,3]

[1]Klaus Tschira Institute for Integrative Computational Cardiology, University Heidelberg, 69120 Heidelberg, Germany
[2]Department of Internal Medicine III (Cardiology, Angiology, and Pneumology), University Hospital Heidelberg, 69120 Heidelberg, Germany
[3]German Centre for Cardiovascular Research (DZHK)-Partner Site Heidelberg/Mannheim, 69120 Heidelberg, Germany

Abstract

to be written

Keywords: Bayesian, 10X Genomics, Cell barcode assignment, Nonsense-mediated mRNA decay (NMD)

INTRODUCTION

Chemical modifications on DNA and histones, also known as epigenetics marks, strongly impact gene expression during cell differentiation and in several other biological programs. In the 1970s, it was recognized that RNA is also subjected to extensive covalent modification, and studies in the late 1980s revealed the widespread deamination of bases (termed RNA editing), which can lead to recoding if it occurs within coding sequences. Impressive development in the RNA modification field occurred during the past eight years, with the discovery of an extensive layer of base modifications in mRNAs. These can influence gene expression and have been already shown to be involved in primary cellular programs such as stem cell differentiation, response to stress, and the circadian clock. The study of RNA modifications and their effects is now referred to as epitranscriptomics, and it reveals striking similarities to what is known for epigenomics. To date thirteen distinct modifications have been identified on mRNA transcripts [Anreiter et al., 2021]. These modifications are catalyzed by a variety of dedicated enzymes and can be divided into two classes: modifications of cap-adjacent nucleotides and internal modifications.

*christoph.dieterich@uni-heidelberg.de

In contrast to the m7G cap, the impact of internal modifications on gene regulation has been less studied apart from RNA editing, which is mediated by RNA deaminases (e.g. the ADAR family). The most widespread internal mRNA modification is N6-methyladenosine (m6A). By modulating the processing of mRNA, m6A can regulate a wide range of physiological processes and its alteration has been linked to several diseases Roignant and Soller [2017], Zaccara et al. [2019], Shi et al. [2019]. The modification is catalyzed co-transcriptionally by a Mega-Dalton methyltransferase complex, which includes the heterodimer METTL3-METTL14 and other associated subunits Garcias Morales and Reyes [2021]. This modification is reversible since two proteins of the AlkB-family demethylases can remove m6A from mRNA transcripts [Jia et al., 2011, Zheng et al., 2013]. In mammals, m6A preferentially localizes within long internal exons and at the beginning of terminal exons at so-called DRACH motif (D = A/G/U, R = A/G, H = A/C/U) sites [Dominissini et al., 2012, Meyer et al., 2012, Ke et al., 2015]. Once deposited, m6A is recognized by several reader proteins that can affect the fate of mRNA transcripts in nearly every step of the mRNA life cycle, which includes alternative splicing [Adhikari et al., 2016, Roundtree et al., 2017]. The best-described readers are the YTH domain family of proteins that decode the signal and mediate m6A functions. By affecting RNA structure, m6A can also indirectly influence the association of additional RNA-binding proteins (RBPs) and the assembly of larger messenger ribonucleoprotein particles (mRNPs).

Several approaches have been presented to map RNA modifications on RNA. Herein, we focus on mRNA modification site detection in general and on m6A in particular where antibody-based protocols (miCLIP), methylation-sensitive restriction enzyme assays (MazF) or transgenic approaches (TRIBE, DART) have been presented. All of the aforementioned approaches rely on high-throughput sequencing on the Illumina platform. This typically involves cDNA synthesis by reverse transcription and PCR-based library amplification. One recent addition to the tool is direct RNA single molecule sequencing on the Oxford Nanopore Technology platform. While or software workflow is able to deal with Illumina and Nanopore-based approaches, the latter is the principal topic of our methods article.

MATERIALS

ONT direct RNA sequencing

1. 500 ng polyA$^+$ RNA isolated from total RNA e.g. with Oligotex mRNA kit (Qiagen) or Dynabeads oligo dT$_{25}$ beads (Thermo Fisher Scientific) or *in vitro* transcriptome sample. Store RNA at -80 °C and the mRNA purification kit as recommended by the manufacturer.

2. Nuclease-free water. Store at room temperature.

3. Direct RNA-sequencing kit (SQK-RNA002, Oxford Nanopore Technologies). Store at -20 °C.

4. NEBNext Quick Ligation Reaction Buffer (New England Biolabs). Store at -20 °C.

5. T4 DNA Ligase (New England Biolabs). Store at -20 °C.

6. dNTP Mix (10 mM each). Store at -20 °C.

7. SuperScript IV Reverse Transcriptase (Thermo Fisher Scientific). Store at -20 °C.

8. Agencourt RNAClean XP beads (Beckman Coulter). Store at 4 °C.

9. 70 % ethanol, freshly prepared.

10. Qubit dsDNA HS assay kit and Qubit Fluorometer (Thermo Fisher Scientific).

11. Flow cell priming kit (EXP-FLP002, Oxford Nanopore Technologies). Store at -20 °C.

12. Thermocycler.

13. Gentle rotator mixer.

14. Magnetic stand for 1.5 ml tubes.

15. 1.5 ml DNA LoBind tubes (Eppendorf), 0.2 ml PCR tubes.

16. MinION or GridION sequencing device and MinION R9.4.1 Flow cells (FLO-MIN106D, Oxford Nanopore Technologies). Store Flow cells at 4 °C.

Preparation of an *in vitro* transcriptome sample

1. 100 ng polyA$^+$ RNA isolated from total RNA e.g. with Oligotex mRNA kit (Qiagen) or Dynabeads oligo dT_{25} beads (Thermo Fisher Scientific). Store RNA at -80 °C and the mRNA purification kit as recommended by the manufacturer

2. 10 μM oligo(dT)-VN RT primer. TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTVN. Store at -20 °C.

3. 20 μM template switching oligo (TSO). ACTCTAATACGACTCACTATAGGGAGAGGGCrGrG+G. Store at -20 °C.

4. 10 μM T7 extension primer. GCTCTAATACGACTCACTATAGG. Store at -20 °C.

5. Nuclease-free water. Store at room temperature.

6. dNTP Mix (10 mM each). Store at -20 °C.

7. Template Switching RT Enzyme Mix (New England Biolabs). Store at -20 °C.

8. Q5 Hot Start High-Fidelity 2X Master Mix (New England Biolabs). Store at -20 °C.

9. RNase H (5,000 U/ml) (New England Biolabs). Store at -20 °C.

10. NucleoSpin Gel and PCR Clean-up, Mini kit for gel extraction and PCR clean up (Macherey-Nagel) or equivalent. Store at room temperature.

11. MEGAscript T7 transcription kit (Thermo Fisher Scientific). Store at -20 °C.

12. RNA Clean & Concentrator-25 kit (Zymo Research). Store at room temperature.

13. Thermocycler.

14. Table top centrifuge for 1.5 ml tubes.

15. Nanodrop spectrophotometer or equivalent.

16. 0.2 ml PCR tubes, 1.5 ml DNA LoBind tubes (Eppendorf).

Hardware requirements

All analyses have been performed/tested on two alternative hardware systems: a standard Linux desktop computer or an Apple iMac (Retina 5K, ultimo 2014). The workflow requires a multi-core processor system with minimal main memory of 16GB RAM and several GBs of free disk space (depending on data set size).

Software dependencies and installation

Our analysis workflow has few requirements, which are detailed in Table 2. Specifically, to execute our workflow, the following prerequisites are necessary: a BASH shell, a JAVA runtime environment, a working PERL and R installation. Additional i.e. non-standard software to process and map Nanopore reads (bedtools, samtools and Minimap2) are obligatory, while

the installation of a Nanopore read simulator (NanoSim) is optional and depends on your use case. Table **??** lists some additional R packages, which are required to run the R code. Detailed instructions on how to setup are found under `https://github.com/dieterich-lab/MiMB_JACUSA2_chapter`

METHODS

Overview Figure 1

Nanopore direct RNA sequencing

1. Adjust 500 ng polyA$^+$ RNA to a total volume of 9 μl with nuclease-free water. Complete RT adapter ligation reaction (in 0.2 ml PCR tube) with 3 μl NEBNext Quick Ligation Reaction Buffer, 0.5 μl RNA CS (RCS, from SQK-RNA002), 1 μl RT-Adapter (RTA, from SQK-RNA002) and 1.5 μl T4 DNA Ligase. Incubate 10 min at room temperature.

2. Prepare reverse transcription master mix on ice during ligation: 9 μl nuclease-free water, 2 μl 10 mM dNTPs, 8 μl 5x SuperScript IV first strand buffer, 4 μl 0.1 mM DTT.

3. Add the reverse transcription master mix to the ligation reaction and mix by pipetting. Add 2 μl SuperScript IV reverse transcriptase and mix by pipetting. Incubate in a thermocycler with the following protocol: 50 min at 50 °C, 10 min at 70 °C, cool down to 4 °C.

4. Let the Agencourt RNAClean XP beads come to room temperature during reverse transcription. Carefully resuspend beads before use. Transfer reaction to a 1.5 ml DNA LoBind tube and mix with 72 μl Agencourt RNAClean XP beads. Incubate 5 min at room temperature on a gentle rotator mixer.

5. Collect beads on a magnetic stand and remove supernatant. Wash pelleted beads two times (30 sec) with 200 μl freshly prepared 70 % ethanol. Remove supernatant. Spin sample down and place on magnet again. Remove any residual ethanol.

6. Resuspend beads in 20 μl nuclease-free water by gentle flicking and incubate 5 min at room temperature on a gentle rotator mixer. Collect beads on a magnetic stand and transfer 20 μl eluate in a fresh 1.5 ml DNA LoBind tube.

7. For ligation of the RMX adapter, add the following to 20 μl eluate: 8 μl NEBNext Quick Ligation Reaction Buffer, 6 μl RMX (from SQK-RNA002), 3 μl nuclease-free water, 3 μl T4 DNA Ligase. Mix by pipetting and incubate 10 min at room temperature.

8. Add 40 μl carefully resuspended Agencourt RNAClean XP beads to the reaction and mix by pipetting. Incubate 5 min at room temperature on a gentle rotator mixer.

9. Collect beads on a magnetic stand and remove supernatant. Wash pelleted beads two times with 150 μl wash buffer (WSB, from SQK-RNA002). Resuspend beads by flicking, spin down and return to magnetic stand. Remove supernatant from pelleted beads.

10. Resuspend beads in 21 μl elution buffer (EB, from SQK-RNA002) by gentle flicking and incubate 5 min at room temperature on a gentle rotator mixer. Pellet beads on a magnetic stand and transfer 21 μl eluate in a fresh 1.5 ml DNA LoBind tube.

11. Quantify 1 μl of the library on a Qubit fluorometer with the Qubit dsDNA HS kit according to the manufacturerers protocol. Concentration should be usually in the range of 5 - 10 ng/μl.

12. Insert MinION R9.4.1 Flow cell in the MinION or GridION sequencing device and perform Flow cell check in the MinKNOW software. For successful sequencing of mammalian polyA$^+$ RNA at least 1,000 available pores are recommended.

13. Prepare Priming Mix by adding 30 μl flush tether (FLT, from EXP-FLP002) to a vial of flush buffer (FB, from EXP-FLP002) and mix by pipetting. Open priming port. Remove air bubble from priming port by inserting the tip of a P1000 pipette into the priming port and slowly dialing up, until a small volume of storage buffer enters the pipette tip. Load 800 μl Priming Mix via the priming port and carefully avoid introduction of air bubbles. Close the priming port and wait for 5 min.

14. Mix 20 μl library with 17.5 μl nuclease-free water and 37.5 μl RNA running buffer (RRB, from SQK-RNA002) and mix by pipetting. Open the priming port and the sample port. Load 200 μl Priming Mix via the priming port. Mix library by pipetting just before loading and load dropwise via the sample port. Carefully avoid introduction of air bubbles. Close the sample port and the priming port.

15. Start sequencing for 48 to 72 h in the MinKNOW software. Choose direct RNA-sequencing kit and high-accuracy basecalling as parameters. We recommend to adjust the output filter to a minimum Q score of 7 (instead of 9).

Preparation of an *in vitro* transcriptome sample

The *in vitro* transcriptome sample is prepared based on a protocol published by Zhang *et al.* Zhang et al. [2021] with some modifications.

1. Adjust 100 ng polyA$^+$ RNA to a total volume of 6 μl with nuclease-free water. Add 1 μl each of 10 μM oligo(dT)-VN RT primer and 10 mM dNTPs. Mix by pipetting and incubate in a thermocycler: 5 min at 75 °C, 2 min at 42 °C, cool to 4 °C.

2. Assemble 2.5 μl 4x template switching RT buffer, 0.5 μl 20 μM TSO, 1 μl 10x template switching RT enzyme mix and mix by pipetting. Combine with 6 μl RNA and incubate in a thermocycler: 90 min at 42 °C, 10 min at 68 °C, cool to 4 °C.

3. For Second strand synthesis add to First strand synthesis reaction: 50 μl Q5 Hot Start High-Fidelity 2X Master Mix, 5 μl RNase H, 2 μl 10 μM T7 extension primer, 33 μl nuclease-free water. Mix by pipetting and incubate in a thermocycler: 15 min at 37 °C, 1 min at 95 °C, 10 min at 65 °C, cool to 4 °C.

4. Purify double stranded cDNA with NucleoSpin Gel and PCR Clean-up kit according to the manufacturerers protocol and elute in 20 μl elution buffer. Determine concentration on a Nanodrop spectrophotometer. cDNA may be stored at -20 °C.

5. Combine 8 μl cDNA for *in vitro* transcription with 2 μl each of ATP, GTP, CTP, UTP, 10x reaction buffer and enzyme mix from the MEGAscript T7 transcription kit. Incubate 3 h at 37 °C.

6. Digest template DNA by addition of 1 μl Turbo DNase. Mix by pipetting and incubate 15 min at 37 °C.

7. Adjust reaction volume to 100 μl with nuclease-free water and clean up with RNA Clean & Concentrator-25 kit according to the manufacturers protocol, using two volumes of adjusted RNA binding buffer (1:1 RNA binding buffer : ethanol). Elute RNA in 25 μl nuclease-free water. Determine RNA concentration on a Nanodrop spectrophotometer. Store at -80 °C.

Nanopore read processing

1. Base call the ionic current signal stored in FAST5 file using Guppy. For the IVT readout, we adopted real-time base calling with the MinKNOW-embedded Guppy basecaller. Otherwise, Guppy basecaller software can be used; in this case, the basecaller requires the path to FAST5 files, the output folder, and the config file or the flowcell/kit combination. The output is FASTQ files that can be compressed using the option "–compress_fastq".

```
$ guppy_basecaller --compress_fastq -i path_to_fast5 -s path_to_output
-c config_file.cfg --cpu_threads_per_caller 14 --num_callers
1
```

Set the number of threads "cpu_threads_per_caller" and the number of parallel basecallers "num_caller" according to your resources. Additional details can be found in Gup [2019].

2. Align reads to the transcriptome using Minimap2 software. The output is a SAM file that has to be converted to a compressed form as BAM file using SAMtools command. The alignment requires the reference sequence. Here, we used GRCh38 Ensembl annotation and FASTA file release version 96. To reduce the indexing time of the human genome, save the index with the option "-d" before the mapping and use the index instead of the reference file in the minimap2 command line.

```
$ minimap2 -d reference.mmi reference.fa
```

To allow spliced alignments, use the setting "-ax splice –junc-bed annotation.bed –junc-bonus" where "–junc-bonus" allows to tune the bonus score and the BED file "–junc-bed annotation.bed" provides the splice junctions. The BED file can be generated using the following command:

```
$paftools.js gff2bed annotation.gtf > annotation.bed
```

Use "-ub" to allow alignment to both strands or '-uf' to force the alignment to only forward strand. For Direct RNA Sequencing, it is recommended to set a small k-mer size "-k [=14]" to enhance sensitivity. We recommend outputting primary alignments "–secondary=no". Use the parameter '–MD' to add the reference sequence information to the alignment; this is recommended for the downstream analysis. Customize the number of threads "-t" according to your resources. Check Minimap2 manual for more details [Min].

```
$ minimap2 -t 5 --MD -ax splice --junc-bonus 1 -k14 --secondary=no
--junc-bed final_annotation_96.bed -ub reference.mmi Reads.fastq.gz
|samtools view -bS > mapping.bam
```

3. Map RNA modifications using JACUSA2 pipeline. JACUSA2 [Piechotta et al., 2021] rapidly detects RNA modifications based on a comparative strategy where the mapping features (mismatch, insertion and deletion) of a sample of interest is compared to a reference sequence (call-1) or against a sample without RNA modifications, e.g. a knock-out of an RNA modifying enzyme or an IVT (call-2). Moreover, it allows the integration of information from replicate experiments. The output

of JACUSA2 variant calling is a set of scores reflecting the read signatures involving mismatch, insertion and deletion. The analysis of read signature can be used for RNA modification detection. We integrate JACUSA2, in particular call-2 method, with the downstream analysis in one pipeline using the Python-based workflow management system Snakemake [Köster and Rahmann, 2012]. The Snakemake pipeline involves rules for the variant calling using JACUSA2 call-2, detection of RNA modification patterns, prediction of new modified sites and other intermediate rules as shown in figures 4. The input of the pipeline are BAM files from paired conditions with different replicates. BAM files need to be sorted and may be subjected to many filters before being used by JACUSA2 call2 rule. Here, we suggest to filter out secondary and poor alignments. The output of JACUSA2 call2 is preprocessed (get_features) and subjected to a learning process to extract and visualize modification patterns (resp. get_pattern, visualize_pattern) and make predictions (predict_modification). "split_trani_test" rule allows splitting input data into a training set and a test set. To use our snakemake-based JACUSA2 pipeline a set of parameters should be defined in the "config.yaml" file; mainly: the label of the analysis 'label', the input bam files under 'data', the reference sequence 'reference', a file containing size of chromosomes 'chr_size', JACUSA2 jar file 'jar', plus the path to inputs and outputs under 'path_inp' and 'path_out' fields respectively. Further details on how to use JACUSA2 pipeline is presented within the use cases in the next section. The pipeline could be executed on a high-performance-computing cluster (HPC) using the following command by specifying the number of cores to be used "–cores [=all]" and the rule name:

```
$ srun snakemake --cores all rule_name
```

Check Snakemake documentation for more details [sna].

Use Case 1: Comparison of wild-type and knock-out samples

The conventional way to detect RNA modifications using direct RNA sequencing is to compare a modified sample to an unmodified control sample. To assess the ability of JACUSA2 in this case, we used a published dataset of HEK293 cell lines to detect m6A modification [Pratanwanich et al., 2021]. The benchmark is composed of two sample sets from two conditions: wild-type cells (modified RNAs) and Mettl3 knockout cells (unmodified RNAs) with two replicates (2 and 3). The FASTQ files are mapped using Minimap2 as described in the previous section. The following analysis is validated against reported m6A sites in the three miCLIP-based studies Boulias et al. [2019], Koh et al. [2019], Körtel et al. [2021] (figure 5).

Given the preprocessed mapped reads as inputs (BAM files): 'HEK293T-WT-rep2.bam' and 'HEK293T-WT-rep3.bam' representing the wildtype replicates and 'HEK293T-KO-rep2.bam' and 'HEK293T-KO-rep3.bam' as the control replicates,

1. Identify read error profile: use "jacusa2_call2" rule to run JACUSA2 in pairwise conditions mode (call-2). The method requires BAM files of the paired conditions and the corresponding library information "-P1" and "-P2". In addition to the mismatch score, add "-D" and "-I" to output the deletion and insertion scores. JACUSA2 allows filtering reads according to many parameters. Here, we consider all sites with base calling quality "-q [> 1]", mapping quality "-m [> 1] and read coverage "-c [> 4]". Plus, it provides a filter feature to improve sensitivity. Here, we consider filtering sites within homopolymer regions "-a [=Y]". The output (named here, "Cond1vsCond2Call2.out") consists of a read error profile where the format is a combination of BED6 with JACUSA2 call-2 specific columns and common info columns: info, filter, and ref. Check JACUSA2 manual for more details on JACUSA2 filter and output options [JAC, 2021]. The number of threads can be customized via the parameter "-p". All parameters related to JACUSA2 method can be added under the field "jacusa_params" in the config file by setting the name of the parameter followed by the corresponding value [key: value]. Be aware to set all parameters before running the pipeline.

   ```
   $ srun snakemake --cores all jacusa2_call2  $
   ```

2. Preprocess JACUSA2 output: from JACUSA2 call-2 output, we select all sites within 5-mer of a central nucleotide 'A' flanked by 2 random nucleotides (NNANN) and we filter out sites of the homo-polymer regions (JACUSA filter: Y). Then, we rebuild the tabular features such that the observations are only sites with a reference base 'A'. Each site is characterized by 15 features corresponding to the mismatch, insertion and deletion scores for the observed site and its two flanking positions from both sides. The rule "get_features" performs the preprocessing step. Use the parameter 'region' with a file containing target 5-mers to limit the analysis to specific sites. The output is an R object "features/features.rds", representing the matrix of Sites×15 features.

   ```
   $ srun snakemake --cores all get_features
   ```

3. Extract m6A modification pattern: given the matrix of Sites×Features, the next step is to learn a model representing the m6A modification pattern. To this end, the conventional non-negative matrix factorization (NMF) analysis is suggested [Lee and Seung, 1999]. Briefly, NMF

factorizes a non-negative data matrix X (here: n sites and m features) into two non-negative matrices as $X \approx WH$, such that W is an $n \times k$ matrix containing basis vectors and H is an $k \times m$ matrix containing coefficient vectors. The coefficient vectors and their combination can be viewed as a pattern for m6A modification. The rank of factorization k is a critical parameter that affects the performance substantially. We suggest to select the rank k according to the method of Frigyesi and Höglund [2008] by looking at silhouette [Rousseeuw, 1987] and cophenetic correlation [Brunet et al., 2004] indices. Accordingly, the performance indices are computed for different choices of rank ($k < n, m$) and compared to the performance of a random permutation of the original data. Subsequently, the chosen rank corresponds to the value with the largest difference between slopes of the original and the randomized data. Here, the unsupervised pattern training is based on the consensus set of $2,401$ m6A sites reported in the three miCLIP-based studies mentioned earlier. Based on the silhouette and cophenetic correlation indices, we could identify an optimal factorization rank of 6 (figure 6A). We then analyzed the identified patterns. According to the membership indicator of each site in matrix W, more than 80% of m6A modification sites can be represented by five patterns (Patterns 1,2,3,4,6) (figure 6B). Interestingly, the linear combination of these five patterns in figure (6C) highlights the importance of position 3 and eventually the implication of all scores.

Using the JACUSA2 pipeline, run rule "get_pattern" to generate patterns and provide the set of modified sites as a ground truth under the field "modified_sites" in the config file. Here, the "miCLIP_union.bed" file contains the m6A sites from the three miCLIP-based studies. A miCLIP annotation, reflecting studies (hence, the consensus) wherein the modification is reported, is added to each site. A subset of modified sites could be used to generate patterns. Accordingly, the "internal_pattern" field should refer to the annotation of selected sites from the "modified_sites" file. Plus, multiple combinations of patterns can be defined and appended to the field "combined_pattern" as new patterns. The corresponding outputs are under "patterns" folder.

```
$ srun snakemake --cores all get_pattern
```

The produced patterns and their combinations can be visualized using "visualize_pattern" rule. The corresponding outputs are under "pattern/viz" folder.

```
$ srun snakemake --cores all visualize_pattern
```

4. Predict m6A modifications: the additive linear combination of the coefficient vectors (patterns) with the 15 features can be used to predict

m6A modification. We examine the ability of prediction on a subset of data of more than 1, 98 million sites with 22, 248 miCLIP m6A sites. We opt for the linear combination of the five important patterns described in step 3. The empirical Cumulative Distribution Function (eCDF) of the inferred scores shows clearly a significant difference between the different miCLIP m6A categories (miCLIP annotation) and the unmodified sites (figure 6D). As the number of negative samples is much larger than the number of positive samples, we particularly recommend investigating the Positive Predictive Value (PPV) of the predictions. Here, figure 6E shows a moderate PPV that increases with the cut-off.

To perform prediction based on selected patterns using JACUSA2 pipeline, run rule "predict_modification". The patterns can be generated from a subset of the input data according to the field "internal_pattern" or predefined patterns indicated in the "external_pattern" field. The output is a BED file containing the estimated scores and the corresponding eCDF and PPV plots. The corresponding outputs are under a new folder called "prediction".

```
$ srun snakemake --cores all predict_modification
```

Note that the rules are linked so that the workflow are determined from top (e.g. predict_modification) to bottom (e.g. sort_bam) and executed accordingly from bottom to top 4. Therefore, running "predict_modification" rule leads to excuting all rules in its pipeline.

Use Case 2: Comparison of wild-type and IVT samples

An alternative way to detect RNA modification is to compare a modified sample to an *in-vitro* (IVT) synthesized control sample. Therefore, we benchmark JACUSA2 on a sample set of wild-type HEK293 cell lines (modified sample) with two replicates (2 and 3) from Pratanwanich et al. [2021] and a modification-free RNA synthesized sample (control sample). The analysis steps are similar to case 1. We evaluate the analysis against miCLIP m6A sites (figure 5).

1. Identify read error profile: we use JACUSA2 call-2 with the same parameters as the previously described case. The input BAM files (HEK293T-WT-rep2.bam, HEK293T-WT-rep3.bam) and (HEK293T-IVT-rep1.bam, HEK293T-IVT-rep2.bam) are associated to the wild-type and IVT replicate samples respectively.

   ```
   $ srun snakemake --cores all jacusa2_call2
   ```

2. Preprocess JACUSA2 output: we select all sites within the specific 5-mer (NNANN) and we consider the Y filter that excludes sites within

the homo-polymer regions. Then, we extract 5-mer features such that the selected sites are represented by the three scores: mismatch, deletion and insertion for the observed site and its two flanking positions from both sides.

```
$ srun snakemake --cores all get_features
```

3. Extract m6A modification pattern: using NMF factorization, we extract patterns from $1,905$ sites reported as modified in the three miCLIP-based studies. Based on the silhouette and cophenetic correlation indices, we could identify an optimal factorization rank of 6 (figure 7A). We determined the predominant factors from matrix W; accordingly, more than 80% of m6A modification sites can be represented by four patterns (Patterns: 1,2,3,6) (figure 7B). In agreement with case 1, the linear combination of the four patterns confirms the importance of position 3 and the implication of all scores as shown in figure (7C).

```
$ srun snakemake --cores all get_pattern
```

4. Predict m6A modifications: we evaluate the prediction ability of the detected patterns on a test set of almost $1,52$ million sites where $17,021$ are miCLIP-m6A modified. We consider the linear combination of the four important patterns (1,2,3,6). Figure 7D shows the eCDF of the inferred scores. The difference between the cumulative distribution of non miCLIP sites and miCLIP sites can be nicely observed, while, the PPV plot shows a lower performance as compared to case 1 (figure 7E). The decrease in performance is likely explained by the absence of all modifications and not exclusively m6A in the control condition, which may induce noise to the score estimation by JACUSA2 call-2 .

CD:to be confirmed

```
$ srun snakemake --cores all predict_modification
```

NOTES

Tips and Tricks

ACKNOWLEDGMENTS

The authors would like to thank Etienne Boileau, Thiago Britto Borges, Tobias Jakobi for proof-reading and comments. The authors are grateful to Marek Franitza for running the experiments on the 10x platform and to Christian Becker for running ONT sequencing. This work was supported by Informatics for Life funded by the Klaus Tschira Foundation.

REFERENCES

Minimap2. https://github.com/lh3/minimap2. Accessed: 2022-01-19.

Snakemake. https://snakemake.readthedocs.io. Accessed: 2022-01-26.

Basecalling with guppy. https://github.com/metagenomics/denbi-nanopore-training/blob/master/docs/basecalling/basecalling.rst, 2019. Accessed: 2022-01-19.

Jacusa2 manual. https://github.com/dieterich-lab/JACUSA2, 2021. Accessed: 2022-01-15.

Samir Adhikari, Wen Xiao, Yong-Liang Zhao, and Yun-Gui Yang. m(6)a: Signaling for mrna splicing. *RNA biology*, 13:756–759, September 2016. ISSN 1555-8584. doi: 10.1080/15476286.2016.1201628.

Ina Anreiter, Quoseena Mir, Jared T. Simpson, Sarath C. Janga, and Matthias Soller. New twists in detecting mrna modification dynamics. *Trends in biotechnology*, 39:72–89, January 2021. ISSN 1879-3096. doi: 10.1016/j.tibtech.2020.06.002.

Konstantinos Boulias, Diana Toczydłowska-Socha, Ben R Hawley, Noa Liberman, Ken Takashima, Sara Zaccara, Théo Guez, Jean-Jacques Vasseur, Françoise Debart, L Aravind, et al. Identification of the m6am methyltransferase pcif1 reveals the location and functions of m6am in the transcriptome. *Molecular cell*, 75(3):631–643, 2019.

Jean-Philippe Brunet, Pablo Tamayo, Todd R Golub, and Jill P Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the national academy of sciences*, 101(12):4164–4169, 2004.

Dan Dominissini, Sharon Moshitch-Moshkovitz, Schraga Schwartz, Mali Salmon-Divon, Lior Ungar, Sivan Osenberg, Karen Cesarkas, Jasmine Jacob-Hirsch, Ninette Amariglio, Martin Kupiec, Rotem Sorek, and Gideon Rechavi. Topology of the human and mouse m6a rna methylomes revealed by m6a-seq. *Nature*, 485:201–206, April 2012. ISSN 1476-4687. doi: 10.1038/nature11112.

Attila Frigyesi and Mattias Höglund. Non-negative matrix factorization for the analysis of complex gene expression data: identification of clinically relevant tumor subtypes. *Cancer informatics*, 6:CIN–S606, 2008.

David Garcias Morales and José L. Reyes. A birds'-eye view of the activity and specificity of the mrna m, javax.xml.bind.jaxbelement@6d66739e, a methyltransferase complex. *Wiley interdisciplinary reviews. RNA*, 12: e1618, January 2021. ISSN 1757-7012. doi: 10.1002/wrna.1618.

Guifang Jia, Ye Fu, Xu Zhao, Qing Dai, Guanqun Zheng, Ying Yang, Chengqi Yi, Tomas Lindahl, Tao Pan, Yun-Gui Yang, and Chuan He. N6-methyladenosine in nuclear rna is a major substrate of the obesity-associated fto. *Nature chemical biology*, 7:885–887, October 2011. ISSN 1552-4469. doi: 10.1038/nchembio.687.

Shengdong Ke, Endalkachew A. Alemu, Claudia Mertens, Emily Conn Gantman, John J. Fak, Aldo Mele, Bhagwattie Haripal, Ilana Zucker-Scharff, Michael J. Moore, Christopher Y. Park, Cathrine Broberg Vågbø, Anna Kusśnierczyk, Arne Klungland, James E. Darnell, and Robert B. Darnell. A majority of m6a residues are in the last exons, allowing the potential for 3' utr regulation. *Genes & development*, 29:2037–2053, October 2015. ISSN 1549-5477. doi: 10.1101/gad.269415.115.

Casslynn WQ Koh, Yeek Teck Goh, and WS Sho Goh. Atlas of quantitative single-base-resolution n 6-methyl-adenine methylomes. *Nature communications*, 10(1):1–15, 2019.

Nadine Körtel, Cornelia Rücklé, You Zhou, Anke Busch, Peter Hoch-Kraft, Reymond FX Sutandy, Jacob Haase, Mihika Pradhan, Michael Musheev, Dirk Ostareck, et al. Deep and accurate detection of m6a rna modifications using miclip2 and m6aboost machine learning. *bioRxiv*, pages 2020–12, 2021.

Johannes Köster and Sven Rahmann. Snakemake—a scalable bioinformatics workflow engine. *Bioinformatics*, 28(19):2520–2522, 2012.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

Kate D. Meyer, Yogesh Saletore, Paul Zumbo, Olivier Elemento, Christopher E. Mason, and Samie R. Jaffrey. Comprehensive analysis of mrna methylation reveals enrichment in 3' utrs and near stop codons. *Cell*, 149: 1635–1646, June 2012. ISSN 1097-4172. doi: 10.1016/j.cell.2012.05.003.

Michael Piechotta, Qi Wang, Janine Altmüller, and Christoph Dieterich. Rna modification mapping with jacusa2. *bioRxiv*, 2021.

Ploy N Pratanwanich, Fei Yao, Ying Chen, Casslynn WQ Koh, Yuk Kei Wan, Christopher Hendra, Polly Poon, Yeek Teck Goh, Phoebe ML Yap, Jing Yuan Chooi, et al. Identification of differential rna modifications from nanopore direct rna sequencing with xpore. *Nature Biotechnology*, 39(11):1394–1402, 2021.

Jean-Yves Roignant and Matthias Soller. m, javax.xml.bind.jaxbelement@8cec19d, a in mrna: An ancient mechanism for fine-tuning gene expression. *Trends in genetics : TIG*, 33: 380–390, June 2017. ISSN 0168-9525. doi: 10.1016/j.tig.2017.04.003.

Ian A. Roundtree, Molly E. Evans, Tao Pan, and Chuan He. Dynamic rna modifications in gene expression regulation. *Cell*, 169:1187–1200, June 2017. ISSN 1097-4172. doi: 10.1016/j.cell.2017.05.045.

Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

Hailing Shi, Jiangbo Wei, and Chuan He. Where, when, and how: Context-dependent functions of rna methylation writers, readers, and erasers. *Molecular cell*, 74:640–650, May 2019. ISSN 1097-4164. doi: 10.1016/j.molcel.2019.04.025.

Sara Zaccara, Ryan J. Ries, and Samie R. Jaffrey. Reading, writing and erasing mrna methylation. *Nature reviews. Molecular cell biology*, 20:608–624, October 2019. ISSN 1471-0080. doi: 10.1038/s41580-019-0168-5.

Zhang Zhang, Tao Chen, Hong-Xuan Chen, Ying-Yuan Xie, Li-Qian Chen, Yu-Li Zhao, Biao-Di Liu, Lingmei Jin, Wutong Zhang, Chang Liu, et al. Systematic calibration of epitranscriptomic maps using a synthetic modification-free rna library. *Nature Methods*, 18(10):1213–1222, 2021.

Guanqun Zheng, John Arne Dahl, Yamei Niu, Peter Fedorcsak, Chun-Min Huang, Charles J. Li, Cathrine B. Vågbø, Yue Shi, Wen-Ling Wang, Shu-Hui Song, Zhike Lu, Ralph P. G. Bosmans, Qing Dai, Ya-Juan Hao, Xin Yang, Wen-Ming Zhao, Wei-Min Tong, Xiu-Jie Wang, Florian Bogdan, Kari Furu, Ye Fu, Guifang Jia, Xu Zhao, Jun Liu, Hans E. Krokan, Arne Klungland, Yun-Gui Yang, and Chuan He. Alkbh5 is a mammalian rna demethylase that impacts rna metabolism and mouse fertility. *Molecular cell*, 49:18–29, January 2013. ISSN 1097-4164. doi: 10.1016/j.molcel.2012.10.015.

FIGURE CAPTIONS

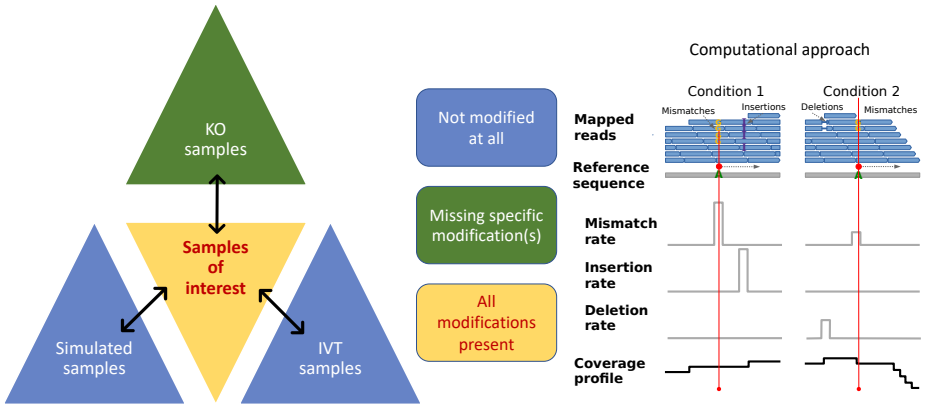

Figure 1: **General outline of RNA modification detection by JACUSA2.** A key feature of our approach is that multiple replicates can be compared as shown on the left. Samples of interests where all modifications are present could be compared with either KO samples where the modification of interest is missing or IVT/simulated samples where all modifications are absent. Read stacks (in blue) are compared head-to-head as shown on the right.

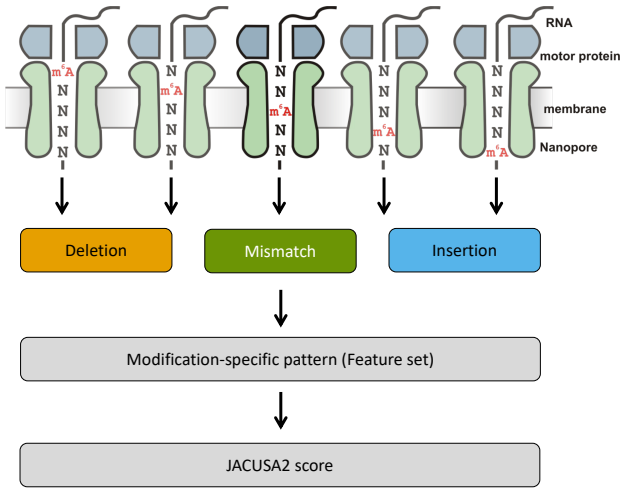

Figure 2: **Motivation of 5-mer context for RNA modification mapping**. The nanopore covers 5 consecutive RNA residues. That is why we consider a 5-mer context and derive 3 principal features for every position within a given 5-mer (15 features in total, with a central A residue in this example). We evaluate each feature set by previously learned patterns and compute a final score for modification site detection.

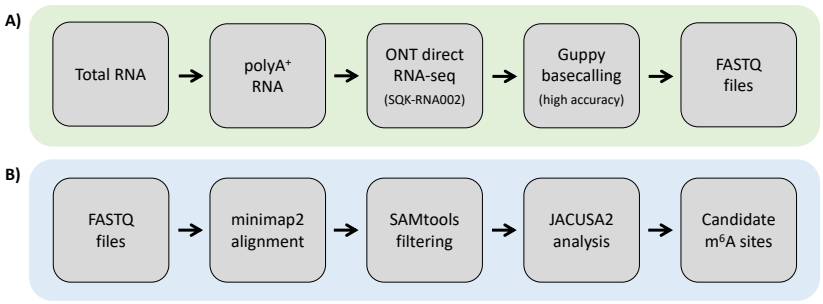

Figure 3: **Experimental and computational workflow**. tbd

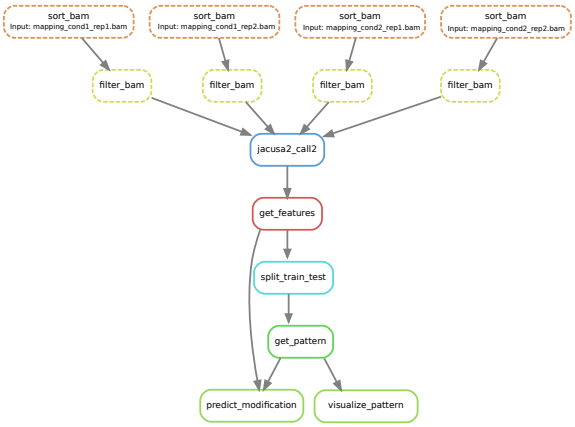

Figure 4: **Computational workflow**. Snakemake workflow for RNA modification detection based on JACUSA2 variant calling.

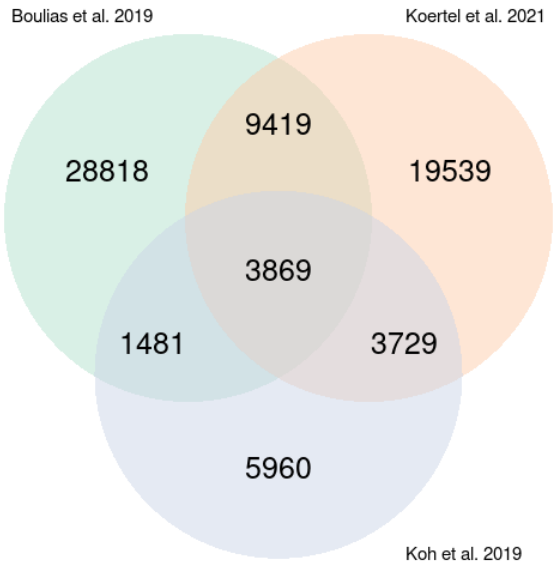

Figure 5: m6A sites reported in the three miCLIP-based studies: Boulias et al. [2019], Koh et al. [2019] and Körtel et al. [2021].

Software	Version	Description
Minimap2	https://github.com/lh3/minimap2 v2.22 or later	https://lh3.github.io/minimap2/
samtools	https://github.com/samtools/samtools v1.12 or later	http://samtools.github.io/
JAVA	openjdk 11.0.12 2021-07-20 - JAVA 11 or later	OpenJDK Runtime Environment
R	https://www.r-project.org/ version 3.5.1 or later	The R Project for Statistical Computing
PERL	https://www.perl.org/ version 5.28.1 or later	Perl is a highly capable, feature-rich programming language
BASH, sed, awk	should be part of your Linux distribution	Misc.
bedtools	https://github.com/arq5x/bedtools2 version 2.29.2 or later	Perl is a highly capable, feature-rich programming language
NanoSim	https://github.com/bcgsc/NanoSim version 3.0.2 or later (optional)	NanoSim is a fast and scalable read simulator that captures the technology-specific features of ONT data

Table 1: **Software dependencies** blubba

TABLE CAPTIONS

TABLES

R Packages	Version	Description
ggplot2	https://cran.r-project.org/web/packages/ggplot2/index.html - ggplot2_3.3.0 or later	ggplot2 is a system for declaratively creating graphics, based on The Grammar of Graphics.
NMF	https://cran.r-project.org/web/packages/NMF/index.html - NMF_0.22.0 or later	Provides a framework to perform Non-negative Matrix Factorization (NMF).

Table 2: **R Package dependencies** blubba

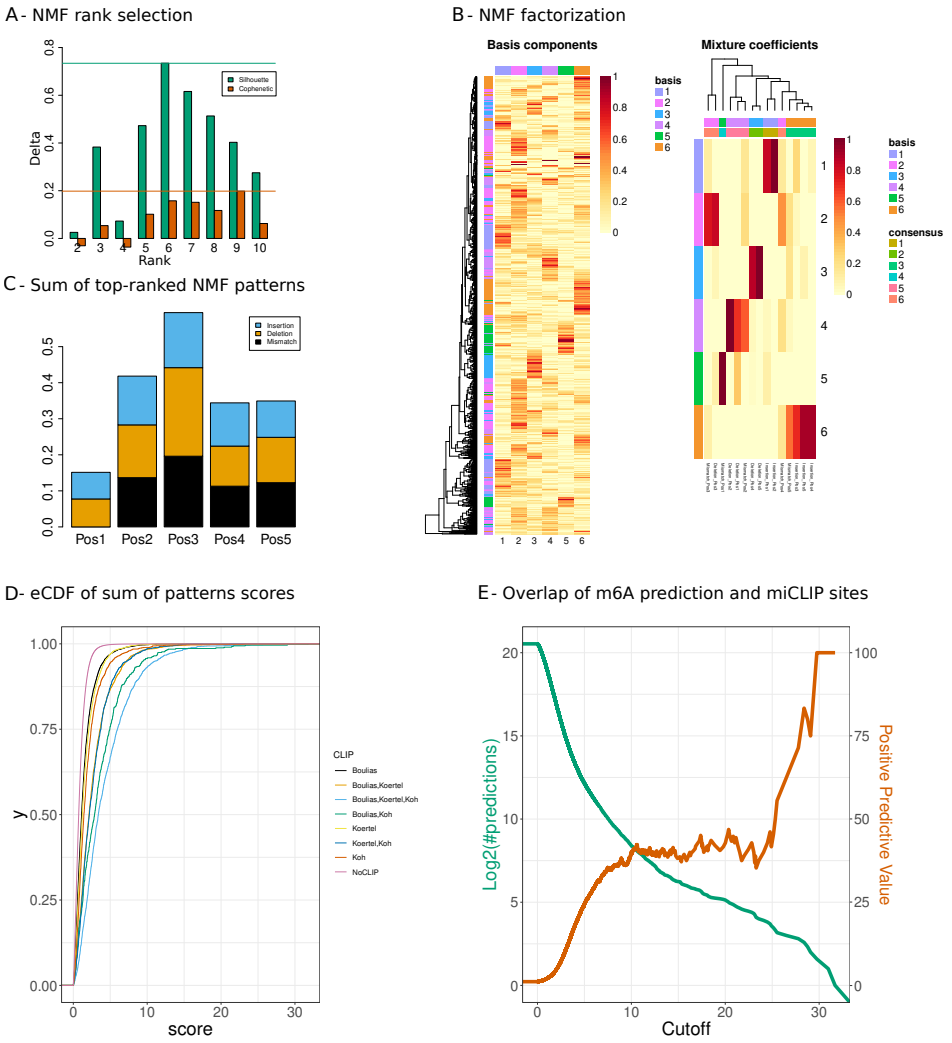

Figure 6: **Case 1. WT versus KO. A**: NMF rank selection. Barplots representing the difference of cophenetic correlation and silhouette indices between the NMF factorization of the original and the randomized data. Ranks with the largest value for both indices are determined, then the smallest rank is selected for the NMF decomposition. **B**: NMF result represented by the basis matrix W and the coefficient matrix H. The matrix H induces the RNA modification pattern. **C**: Barplots representing the linear combination of the top 5 patterns (y-axis) by the number of position in the specific 5-mer context (x-axis) and the score type: mismatch, deletion and insertion (resp. black, orange and blue). The 5 patterns (coefficient vectors: 1,2,3,4,6) are selected according to the predominant columns in matrix W. **D**: Score distribution inferred from the combined patterns, stratified by the different categories of miCLIP validated sites and non miCLIP sites. **E**: Number of predicted m6A sites (green) and Positive Predictive Value (PPV) of predicted m6A sites that overlap with miCLIP sites (orange).

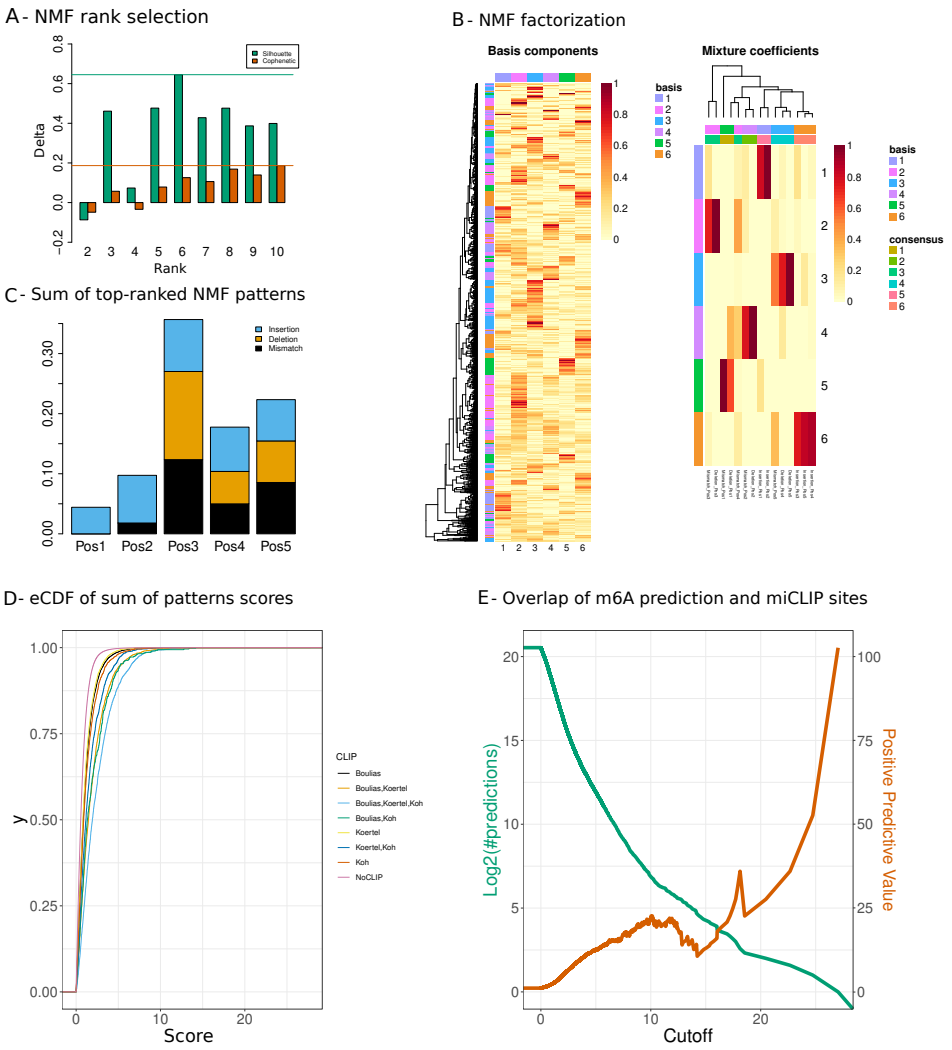

Figure 7: **Case 2. WT versus IVT. A**: NMF rank selection. Barplots representing the difference of cophenetic correlation and silhouette indices between the NMF factorization of the original and the randomized data. Ranks with the largest value for both indices are determined, then the smallest rank is selected for the NMF decomposition. **B**: NMF result represented by the basis matrix W and the coefficient matrix H. The matrix H induces the RNA modification pattern. **C**: Barplots representing the linear combination of the top 4 patterns (y-axis) by by the number of position in the specific 5-mer context (x-axis) and the score type: mismatch, deletion and insertion (resp. black, orange and blue). The 4 patterns (coefficient vectors: 1,2,3,6) are selected according to the predominant columns in matrix W. **D**: Score distribution inferred from the combined patterns, stratified by the different categories of miCLIP validated sites and non miCLIP sites. **E**: Number of predicted m6A sites (green) and Positive Predictive Value (PPV) of predicted m6A sites that overlap with miCLIP sites (orange).