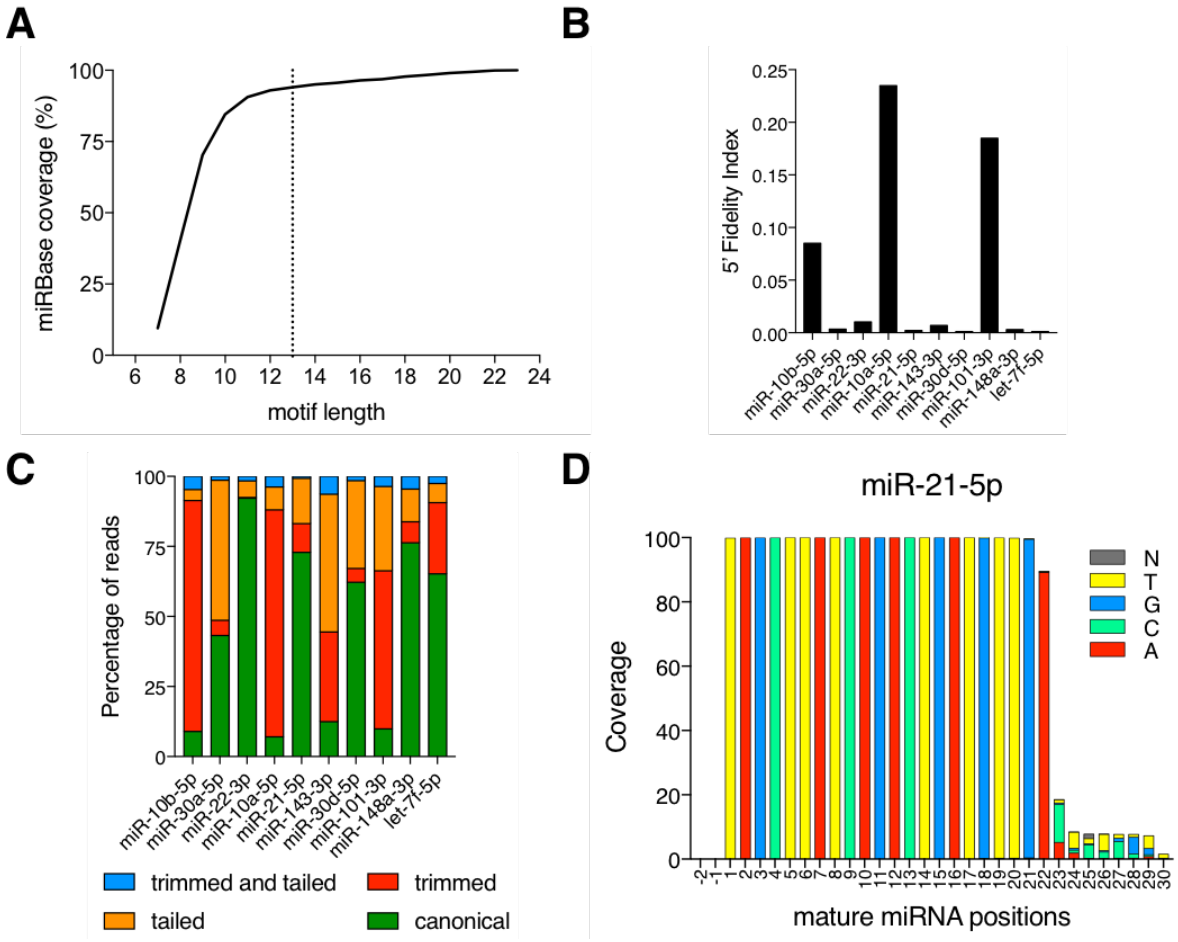
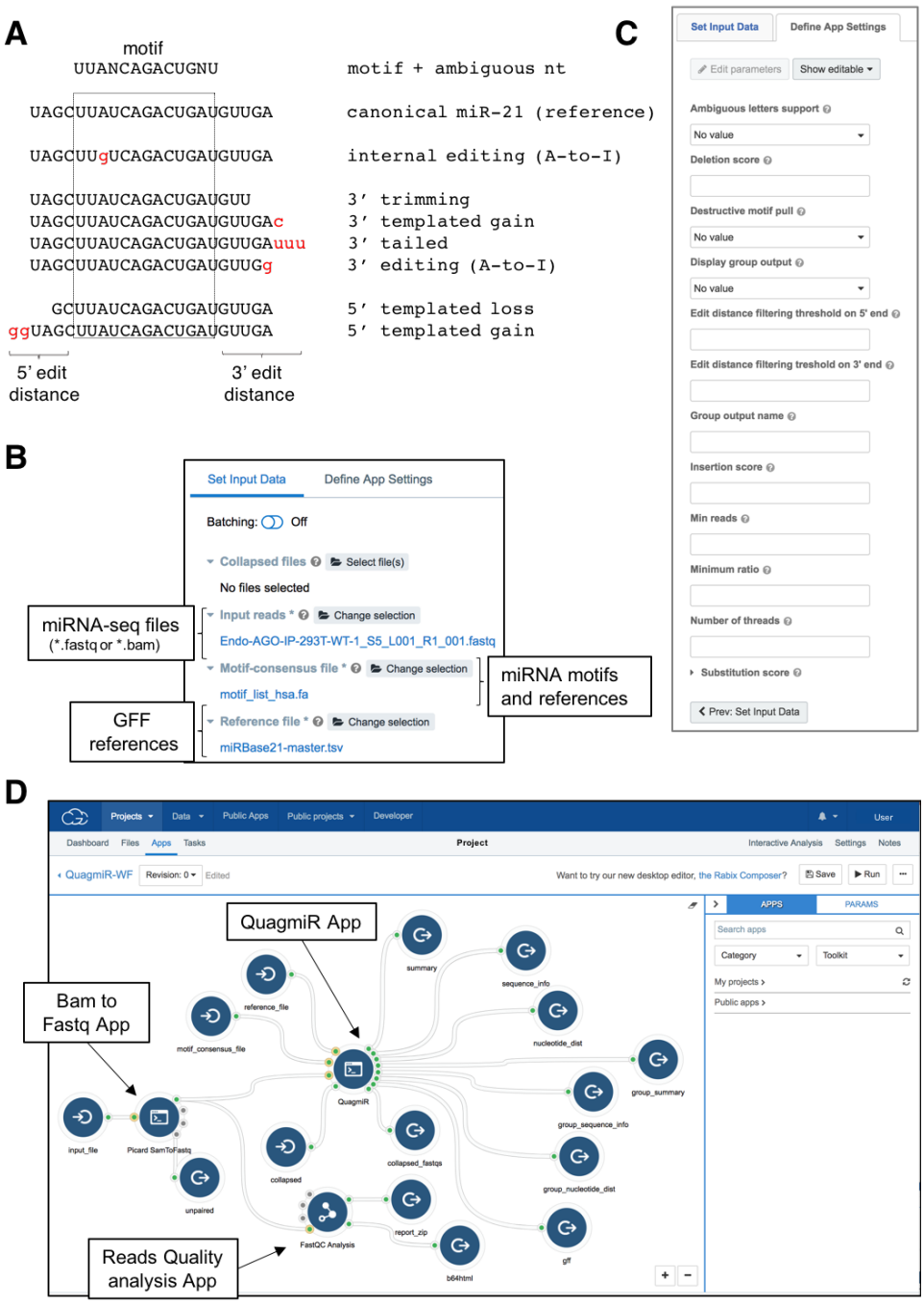
Supplementary Figures

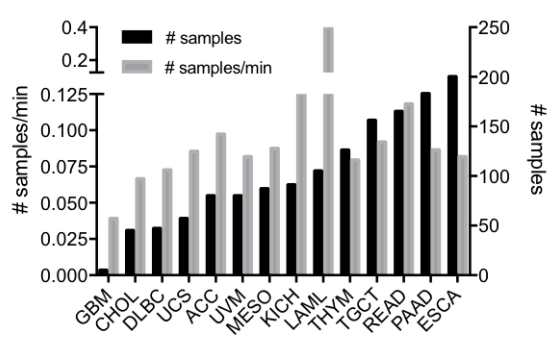


Supplementary Fig. 1. Examples of QuagmiR outputs. (**A**) Percentage of unique motifs relative to the unique number of miRBase entries using a motif of *n* nucleotides in length, centered in the middle of the mature miRNA sequence. (**B**) Example of a 5' Fidelity Index plot, as reported in QuagmiR's summary miRNA analysis report. (**C**) Example of the percentage of reads with 3' end modifications of 10 highly expressed miRNAs in one patient sample, as reported in QuagmiR's summary report. (**D**) Example of the coverage and nucleotide composition at each position relative to the canonical miR-21-5p miRNA sequence, as reported in QuagmiR's sequence composition report.



Supplementary Fig. 2. Customizable use of QuagmiR on the CGC. (**A**) Examples of "relaxed motifs" for miR-21 and of the modifications detected by QuagmiR. QuagmiR permits the use of motifs with IUPAC ambiguous nucleotides (NRYSWKMBDHV) at defined positions. (**B**) Setup of the run on the CGC, which requires selection of the input files to analyze (*.bam or *.fastq files after adaptor sequence removal), the GFF reference and the motif list (*.fa). The computational run time of re-runs using differents motifs can be shortened by selecting the collapsed files generated on previous runs. (**C**) The "Define App Settings" tab in the workflow configuration panel on the CGC. This tab allows users to set advanced parameters for the run (as described in more detail at https://github.com/Gu-Lab-RBL-NCI/QuagmiR/wiki). (**D**) Example of a workflow consisting of *QuagmiR* and other applications, including *Picard SamToFastq* to convert from bam to fastq files and *FastQC Analysis* to evaluate the quality of the reads. Workflows such as the one shown here can be generated easily using the web-based workflow editor on the CGC.

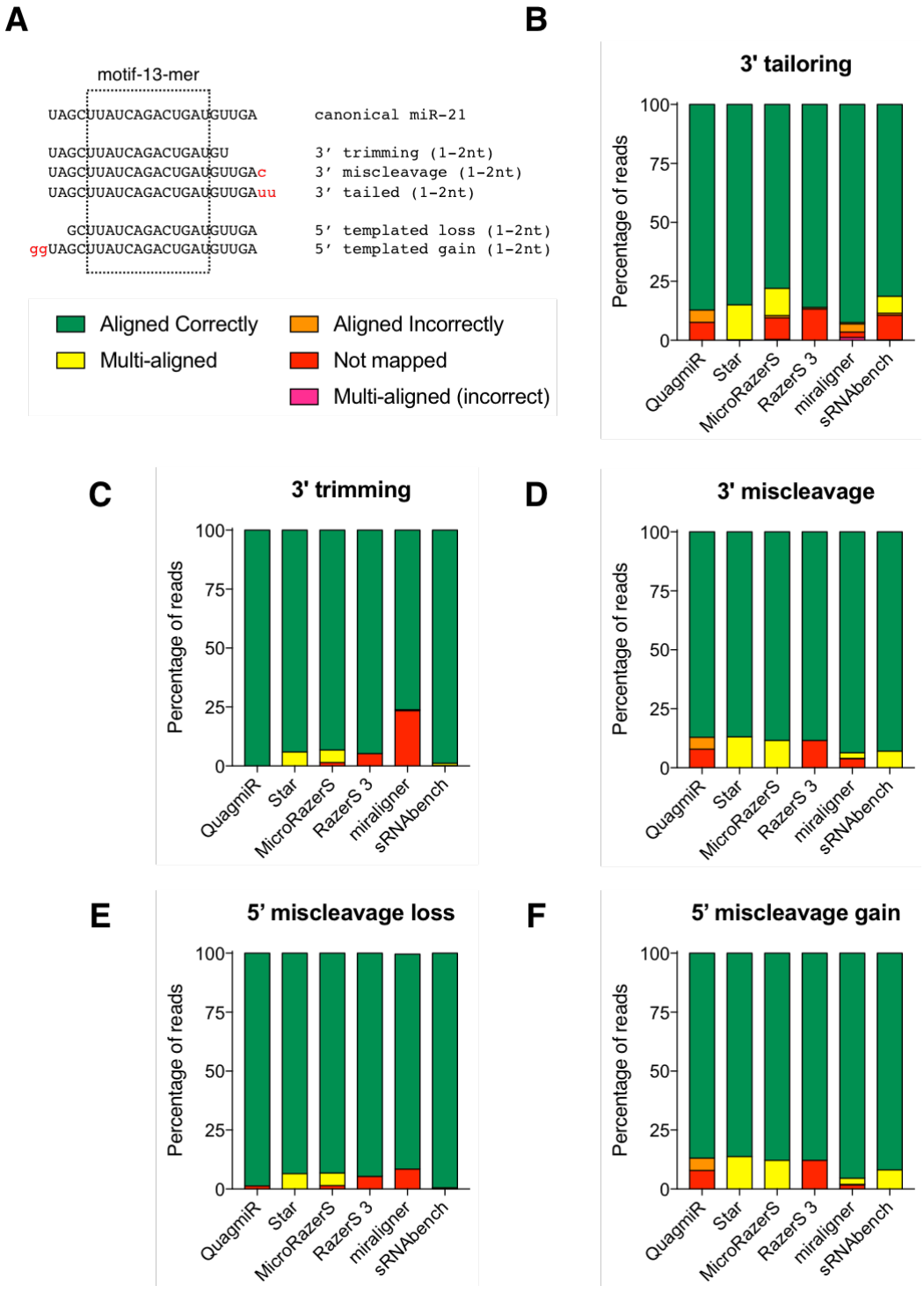
A



B

# samples	File size	Ambiguous nucleotides	Read Length	Time	Instance (AWS)
1	864 MB	No	15 - 30	17 min	c4.2xlarge
1	864 MB	Yes	15 - 30	1h 19 min	c4.2xlarge
72	599MB - 872MB	Yes	15 - 30	2h 19 min	c4.8xlarge

Supplementary Fig. 3. QuagmiR algorithm performance on the CGC. (**A**) Examples of TCGA samples analyzed by QuagmiR on the CGC. The right axis displays the number of samples analyzed in a single batch. The left axis displays the average number of samples per minute analyzed by QuagmiR. (**B**) Table describing the performance of QuagmiR on the CGC with different numbers of samples per batch, usage of ambiguous nucleotides in the reference motif, and usage of AWS instances.



Supplementary Fig. 4. Benchmarking of QuagmiR. (**A**) Examples of synthetic reads used to benchmark QuagmiR using the default parameters relative to other isomiR mappers: Star (Dobin *et al.*, 2013), MicroRazerS (Emde *et al.*, 2010), RazerS3 (Weese *et al.*, 2012), miraligner (Pantano *et al.*, 2010) and sRNAbench (Barturen *et al.*, 2014). (**B**) Detection of random untemplated tailoring of the 3' part (1-2 nt). (**C**) Detection of random trimming of the 3' part (1-2 nt). (**D**) Detection of miscleavage (gain of templated nucleotides) on the 3' part (1-2 nt). (**E**) Detection of random miscleavage loss of the 5' part (1-2 nt). (F) Detection of random miscleavage templated gain of the 5' part (1-2 nt). Note that all programs benchmarked were run using standard parameters which may not reflect their optimal performance on the simulated isomiRs.