1 Appendix

1.1 Parameters for mapping and alignment tools

When `Bowtie 2` was run to produce alignment results, it was run with default parameters with the exception of `-k 200` and `-no-discordant`. When timing `Bowtie 2` the the number of threads (`-p`) was set in accordance with what is mentioned in the relevant text, and the output was piped to `/dev/null`. When `Bowtie 2` was used to produce alignment results for quantification with `RSEM`, `RSEM`'s `Bowtie 2` wrapper (with its default parameters) was used to generate alignments.

When producing alignment results, `STAR` was run with the following parameters: `-outFilterMultimapNmax 200 -outFilterMismatchNmax 99999 -outFilterMismatchNoverLmax 0.2 -alignIntronMin 1000 -alignIntronMax 0 -limitOutSAMoneReadBytes 1000000 -outSAMmode SAMUnosrted`. Additionally, when timing `STAR`, it was run with the number of threads (`-runThreadN`) specified in the relevant text and with the `-outSAMMode None` flag.

To obtain the "pseudo-alignments" produces by `Kallisto`, it was run with the `-pseudobam` flag.

When producing mapping results, `RapMap` was run with the option `-m 200` to limit multi-mapping reads to 200 locations. Additionally, when timing `RapMap`, it was run with the number of threads (`-t`) specified in the relevant text and with the `-n` flag to suppress output.

1.2 Flux Simulator parameters

The Flux simulator dataset was generated using the following parameters:

```
REF_FILE_NAME   Human_Genome
GEN_DIR     protein_coding.gtf

NB_MOLECULES    5000000
TSS_MEAN    50
POLYA_SCALE NaN
POLYA_SHAPE NaN

FRAG_SUBSTRATE  RNA
FRAG_METHOD UR
FRAG_UR_ETA     350

RTRANSCRIPTION  YES
RT_MOTIF default

GC_MEAN NaN
PCR_PROBABILITY 0.05
PCR_DISTRIBUTION default

FILTERING YES

READ_NUMBER 150000000
READ_LENGTH 76
PAIRED_END  YES
ERR_FILE    76
FASTA       YES
```

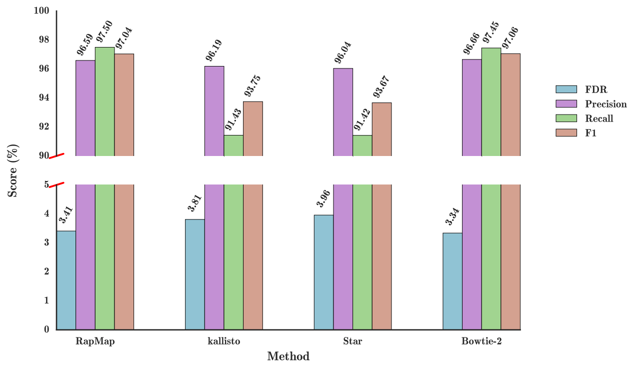
The following parameters were used to produce noise reads:

```
PAIRED_END YES
REF_FILE_NAME noisy.gtf

READ_LENGTH 76
PRO_FILE_NAME flux_simulator_noise_expression.pro
ERR_FILE 76
GEN_DIR Human_Genome/
SEQ_FILE_NAME noise_reads.bed
PCR_DISTRIBUTION none
POLYA_SCALE NaN
FASTA YES
```

```
NB_MOLECULES 2000000
READ_NUMBER 34382441
UNIQUE_IDS YES
POLYA_SHAPE NaN
```

1.3 Mapping accuracy in the presence of noisy reads



Supplementary Figure 1: Precision, recall and F1-score (top) and FDR (bottom) on the simulated dataset with noise, for the 4 different tools we consider.

We tested the effect of including background (i.e. noise) reads on the accuracy of the different mapping and alignment tools. In this experiment, we sampled 9 million reads from the 48 million read simulated data set used in Section 3.1. We then incorporated an additional 1 million "noise" reads from a simulated dataset generated with the Flux Simulator using a custom annotation. This noise annotation was created by constructing a single interval for each transcript, which contained the entire genomic range from the initial until the terminal exons (i.e. it contained all intervening intronic regions). Thus, for each annotated transcript, the noise annotation contains a nascent, un-spliced version of this transcript. This model of noise was motivated from the observation of (Gilbert *et al.*, 2004), that some RNA-seq data (e.g. human brain tissue) contains reads potentially derived from nascent, un-spliced variants of expressed transcripts.

As shown in Supplementary Figure 1 we observe that, in the presence of noise, the precision for all the tools decreases slightly compared to the "clean", 48 million read dataset described in Section 3.1. This is because some small fraction of noisy reads are assigned as false positives, as they map to the mature version of their corresponding transcript of origin that appears in the reference. Overall, however, the results follow a very similar trend both with and without noisy reads. Specifically, `RapMap` (quasi-mapping) performs almost identically to `Bowtie 2`, while `Kallisto` and `STAR` yield very similar results — somewhat under-performing `RapMap` and `Bowtie 2`. This clearly demonstrates that, in the presence of noisy reads, all of the tools degrade gracefully and still perform reasonably well, with no discernible difference between mapping and alignment-based tools.

1.4 Quantification results using TPM

In addition to computing the error metrics based on the estimated versus true number of reads originating from each transcript (as provided in Table 2), we also evaluated the same metrics based instead on the TPM of each transcript. That is, all of the metrics defined in Section 4.3 and appendix 1.5 remain the same, except that x_i now denotes the true TPM value for transcript i and y_i denotes the estimated TPM of transcript i. We note that the Flux Simulator provides neither effective lengths nor TPM estimates directly. To obtain the ground truth TPM values for the Flux Simulator dataset, we first computed the effective length of each transcript (by convolving the characteristic function over the transcript with the *true* fragment length distribution), and then computed the TPM value for each transcript using Equation (3). The results are generally similar to what was observed at the read level, except that TIGAR 2 seems to perform considerably worse under a number of metrics on the RSEM-sim dataset when considering the TPM measure of abundance.

1.5 Error Metrics

We define the error metrics reported in Section 4.3 below, letting x_i denote the true number of reads originating from transcript i and y_i denote the estimated number of reads.

The relative error for transcript i (RE$_i$) is given by $\mathrm{RE_i} = \frac{x_i - y_i}{x_i}$ and the error indicator for transcript i (EI$_i$) is given by

$$\mathrm{EI_i} = \begin{cases} 1 & \text{if } |RE_i| > 0.1 \\ 0 & \text{otherwise} \end{cases}, \tag{4}$$

Table 4. Performance evaluation of different tools along with quasi enabled sailfish (q-Sailfish) with other tools on synthetic data generated by Flux simulator and RSEM simulator

	Flux simulation				RSEM-sim simulation			
	Kallisto	RSEM	q-Sailfish	Tigar 2	Kallisto	RSEM	q-Sailfish	Tigar 2
Proportionality corr.	0.79	0.80	0.80	0.80	0.94	0.96	0.94	0.93
Spearman corr.	0.69	0.73	0.71	0.60	0.91	0.93	0.91	0.89
TPEF	0.87	0.88	0.84	0.94	0.51	0.47	0.50	0.95
TPME	0.07	0.13	0.12	-0.40	0.00	0.00	0.00	0.21
MARD	0.35	0.27	0.31	0.35	0.28	0.25	0.28	0.48
wMARD	0.67	1.22	0.69	1.76	-0.74	-0.73	-0.74	0.12

and it is equal to 1 if the estimated count for this truly expressed transcript (it is undefined, as is RE_i, when $x_i = 0$) differs from the true count by more than 10%. Given RE_i and EI_i, the aggregate true positive error fraction (TPEF) is defined as $\mathrm{TPEF} = \frac{1}{|X^+|}\sum_{i\in X^+} EI_i$. Here, X^+ is the set of "truly expressed" transcripts (those having at least 1 read originating from them in the ground truth). Similarly, the true positive median error is define as $\mathrm{TPME} = \mathrm{median}\left(\{\mathrm{RE}_i\}_{i\in X^+}\right)$. Finally, the absolute relative difference for transcript i (ARD_i) is defined as

$$\mathrm{ARD}_i = \begin{cases} 0 & \text{if } x_i + y_i = 0 \\ \frac{|x_i - y_i|}{0.5(x_i+y_i)} & \text{otherwise} \end{cases}. \tag{5}$$

Consequently, the mean absolute relative difference (MARD) is defined as $\mathrm{MARD} = \frac{1}{M}\sum_i \mathrm{ARD}_i$, and the weighted mean absolute relative difference (wMARD) is defined as

$$\mathrm{wMARD} = \sum_{i\in\mathrm{ARD}^+} \frac{\log\left(\max\left(x_i, y_i\right)\right)\mathrm{ARD}_i}{M}, \tag{6}$$

where, $\mathrm{ARD}^+ = \{i|ARD_i > 0\}$, and M is the total number of transcripts.