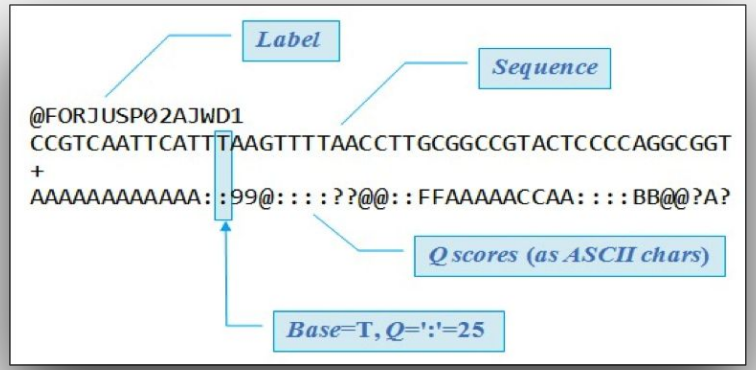
Transcriptomics in Crop Reseach

Introduction to RNAseq technology

Mary-Ann Blätke
JJ Szymanski

Fastq -> gene expression

Fastq



Kallisto

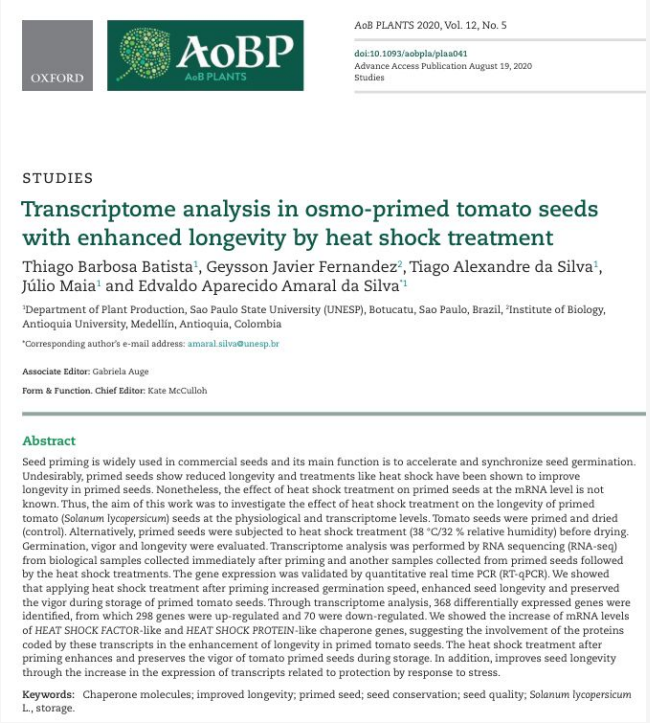
-

>

Gene expression

	A	B	C	D
1	geneID	cond1	cond2	cond3
2	ADD3	5.46	5.16	5.14
3	ADGRE1	4.24	4.17	4.21
4	ADGRE5	4.97	5.09	5.03
5	ADGRG6	6.10	6.13	5.98
6	ADGRL2	3.84	4.15	4.26
7	ADH5	3.83	3.75	3.67
8	ADHFE1	6.25	6.11	6.12
9	ADI1	6.23	5.89	5.93
10	ADIPOR1	7.45	7.30	7.26

Our dataset



AoB PLANTS 2020, Vol. 12, No. 5

doi:10.1093/aobpla/plaa041
Advance Access Publication August 19, 2020
Studies

STUDIES

Transcriptome analysis in osmo-primed tomato seeds with enhanced longevity by heat shock treatment

Thiago Barbosa Batista[1], Geysson Javier Fernandez[2], Tiago Alexandre da Silva[1], Júlio Maia[1] and Edvaldo Aparecido Amaral da Silva[*1]

[1]Department of Plant Production, Sao Paulo State University (UNESP), Botucatu, Sao Paulo, Brazil, [2]Institute of Biology, Antioquia University, Medellín, Antioquia, Colombia

*Corresponding author's e-mail address: amaral.silva@unesp.br

Associate Editor: Gabriela Auge

Form & Function. Chief Editor: Kate McCulloh

Abstract

Seed priming is widely used in commercial seeds and its main function is to accelerate and synchronize seed germination. Undesirably, primed seeds show reduced longevity and treatments like heat shock have been shown to improve longevity in primed seeds. Nonetheless, the effect of heat shock treatment on primed seeds at the mRNA level is not known. Thus, the aim of this work was to investigate the effect of heat shock treatment on the longevity of primed tomato (*Solanum lycopersicum*) seeds at the physiological and transcriptome levels. Tomato seeds were primed and dried (control). Alternatively, primed seeds were subjected to heat shock treatment (38 °C/32 % relative humidity) before drying. Germination, vigor and longevity were evaluated. Transcriptome analysis was performed by RNA sequencing (RNA-seq) from biological samples collected immediately after priming and another samples collected from primed seeds followed by the heat shock treatments. The gene expression was validated by quantitative real time PCR (RT-qPCR). We showed that applying heat shock treatment after priming increased germination speed, enhanced seed longevity and preserved the vigor during storage of primed tomato seeds. Through transcriptome analysis, 368 differentially expressed genes were identified, from which 298 genes were up-regulated and 70 were down-regulated. We showed the increase of mRNA levels of *HEAT SHOCK FACTOR*-like and *HEAT SHOCK PROTEIN*-like chaperone genes, suggesting the involvement of the proteins coded by these transcripts in the enhancement of longevity in primed tomato seeds. The heat shock treatment after priming enhances and preserves the vigor of tomato primed seeds during storage. In addition, improves seed longevity through the increase in the expression of transcripts related to protection by response to stress.

Keywords: Chaperone molecules; improved longevity; primed seed; seed conservation; seed quality; *Solanum lycopersicum* L., storage.

Physiological assays

Seed germination and vigor. Four replications of 50 seeds were germinated in 9 cm Petri dishes with substrate of paper towel moistened with distilled water equivalent to 2.5 times its weight, at 25 °C, under 8 h of light and 16 h in the dark. The length of the primary root, ≥2mm was used as the germination criterion. Data collection was done in different times after sowing; and ended when the germination rate reached 100 % or at 14 days. Seed vigor was determined by the calculation of the time to 50 % of germination (t50) through the analysis of cumulative germination data using the curve fitting module of the Germinator software package (Joosen *et al.* 2010).

Longevity. We used ageing protocol to assess seed [illegible] in which the seeds were placed in a support over a saturated solution of NaCl (75 % RH) at 35 °C in glass bottles hermetically sealed. During storage, the water content of *S. lycopersicum* seeds stabilized at 0.10 ± 0.007 g H_2O/g DW^{-1}, corresponding to ±9.5 % on wet basis. At different time spans, seeds were imbibed and viability was assessed using the germination assay as described earlier. The different time spans were carried out considering the viability loss behaviour of each treatment group during storage. The viability data were transformed into probit to

libraries were 100 base pair (bp) paired-end sequenced. The data output in fastq file format contained sequence information, including the sequencing quality (Phred quality score). Average Phred scores of ≥20 per position were used for the alignment.

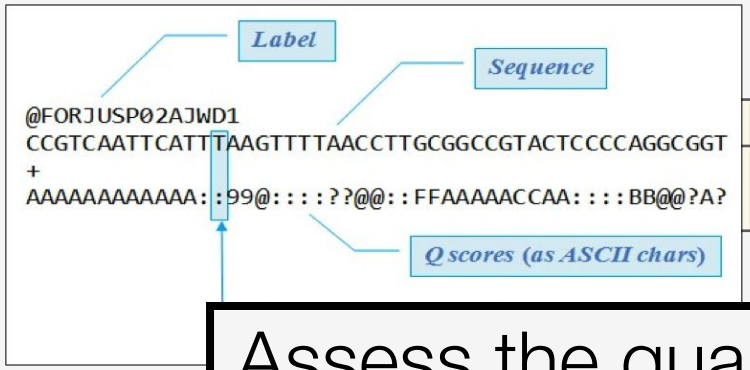
Read alignment and differentially expressed genes. Paired-end reads for mRNA were mapped to the *Solanum lycopersicum* release 39 reference genome using the default parameters of TopHat2 (Kim *et al.* 2013). Counts for RefSeq genes were obtained using HTSeq (Anders *et al.* 2015) and DESeq2 (Love *et al.* 2014) was used to normalize expression counts. The changes in gene expression were considered statistically significant when fold change ≥2 and *P*-values ≤ 0.05. The RNAseq data was deposited in NCBI (BioProject PRJNA562700: https://trace.ncbi.nlm.nih.gov/Traces/sra/?study=SRP220280).

The analysis of principal components was made using all the genes expressed on the RNA seq data. The normalized count per gene was used and transformed to Z-score. This matrix was used was used to perform the PCA. For plotting the PCA results, we used the principal component one and two. The heatmap was generated using the normalized counts of the differentially expressed genes. Then we transformed it to z-score and plotted it using the package pheatmaps of R.

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7494243/

Read quality

Fastq

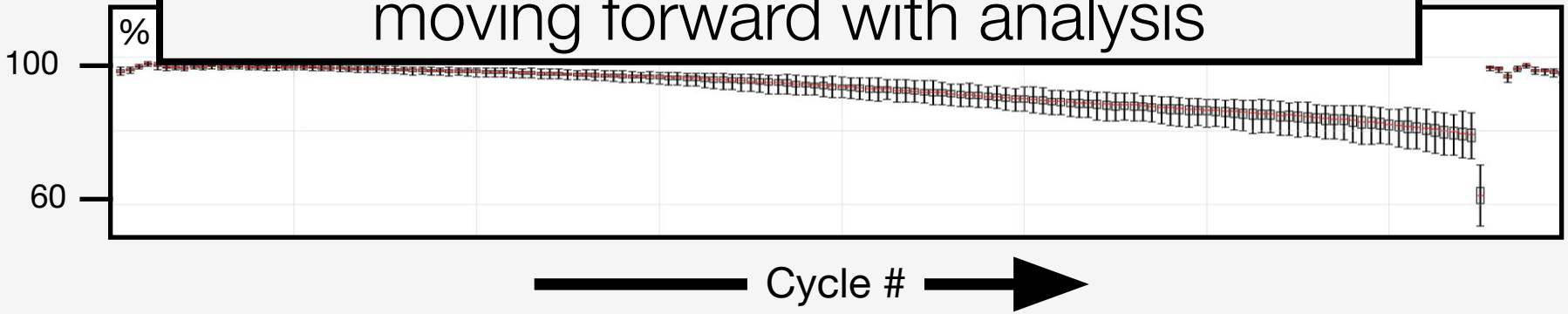


ASCII characters

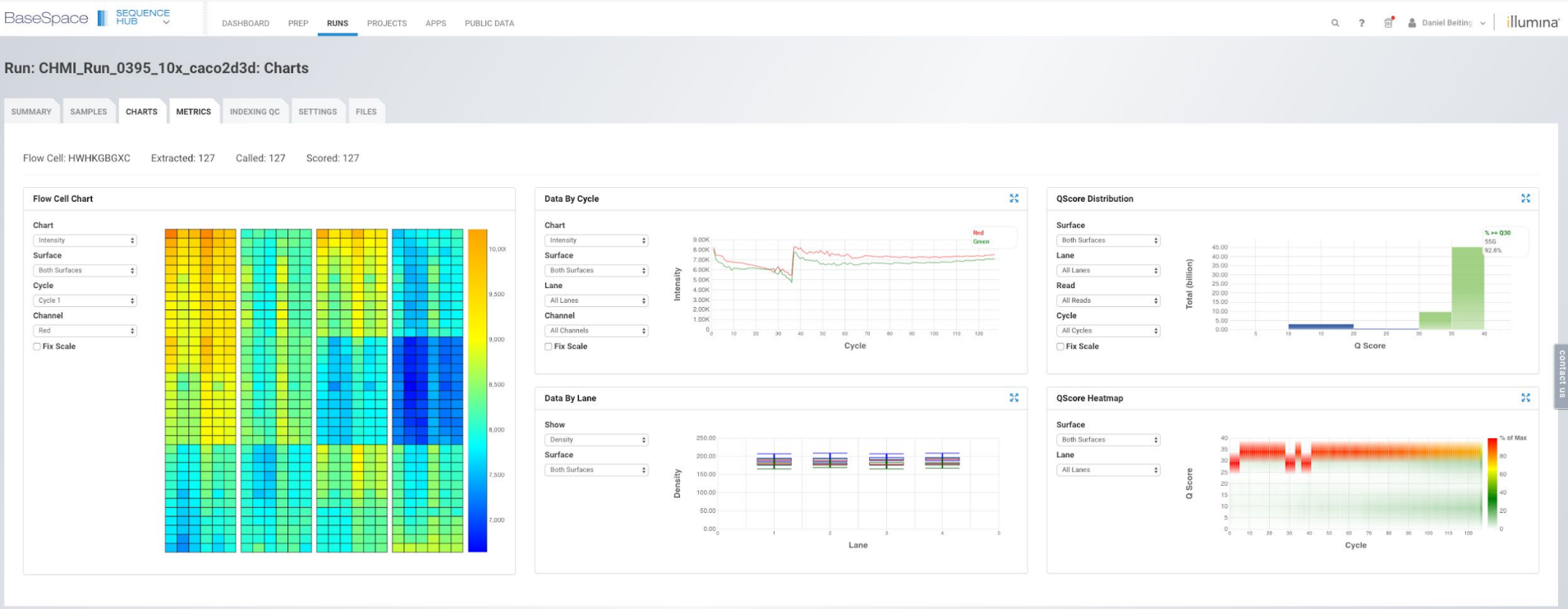
-	.	/	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	@	A	B	C	D	E	F	G	H	I
12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40

Quality score

Assess the quality of your raw data before moving forward with analysis

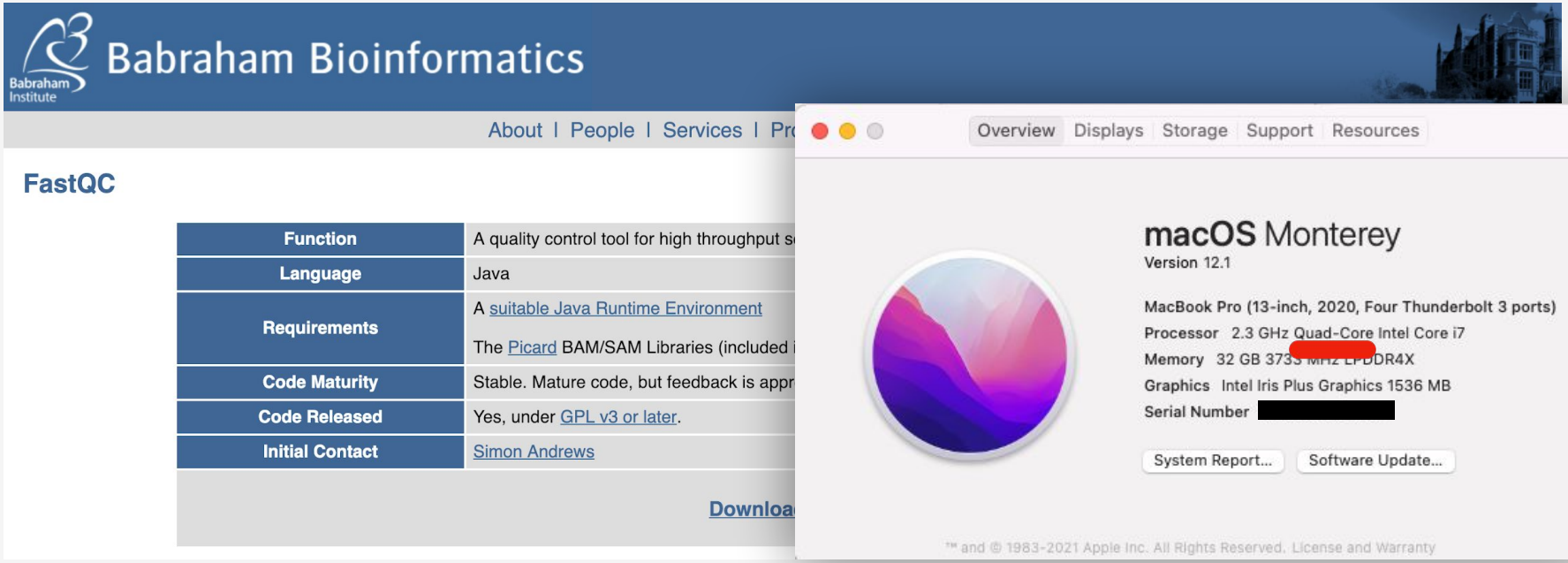


Quality assessment via Illumina's BaseSpace



Quality assessment of fastq files using fastqc

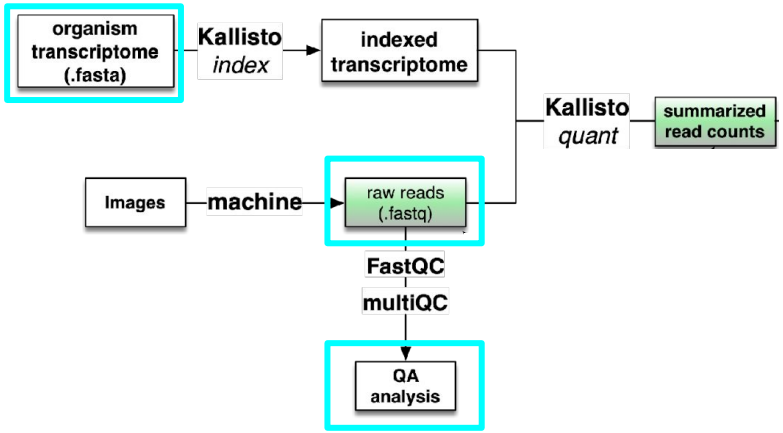
Quality assessment of fastq files using fastqc



fastqc demonstration

fastqc -o ***~/my_path/QC/*** *-t 8* ***~/my_path/raw_reads*.fastq.gz***

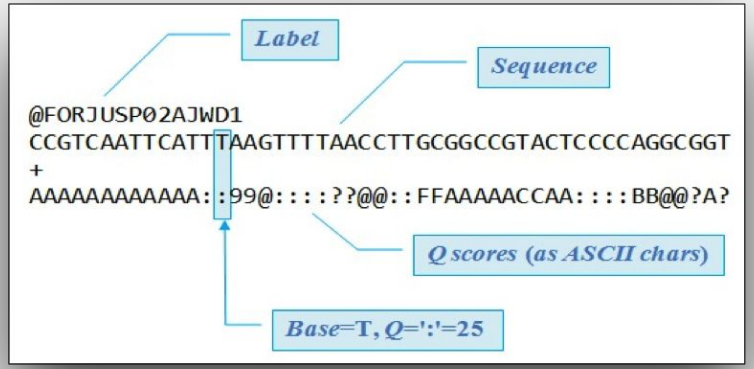
RNAseq analysis pipeline




```
~/Documents/my_study
├── scripts
│   ├── get_raw_reads.sh
│   ├── get_and_index_genome.sh
│   ├── quality_check.sh
│   └── map_reads.sh
├── genome
│   ├── cDNA_Solanum_lycopersicum.fasta
│   └── cDNA_Solanum_lycopersicum.index
├── raw_reads
│   ├── SRR15245007_1.fastq.gz
│   ├── SRR15245007_2.fastq.gz
│   ├── SRR15245013_1.fastq.gz
│   ├── SRR15245013_2.fastq.gz
│   └── etc…
├── publication
│   └── manuscript.pdf
├── QC
│   ├── qc_SRR15245007_1.html
│   └── etc….
└── mapped_reads
    ├── control1
    │   ├── abundance.h5
    │   ├── abundance.tsv
    │   └── run_info.json
    ├── control2
    │   ├── abundance.h5
    │   ├── abundance.tsv
    │   └── run_info.json
    └── etc…
        ├── abundance.h5
        ├── abundance.tsv
        └── run_info.json
```

Fastq -> gene expression

Fastq



Kallisto

->

Gene expression

	A	B	C	D
1	geneID	cond1	cond2	cond3
2	ADD3	5.46	5.16	5.14
3	ADGRE1	4.24	4.17	4.21
4	ADGRE5	4.97	5.09	5.03
5	ADGRG6	6.10	6.13	5.98
6	ADGRL2	3.84	4.15	4.26
7	ADH5	3.83	3.75	3.67
8	ADHFE1	6.25	6.11	6.12
9	ADI1	6.23	5.89	5.93
10	ADIPOR1	7.45	7.30	7.26

Ingredients for an alignment

our data
(or your data)

.fastq

reference sequences

.fasta

Ingredients for an alignment

our data

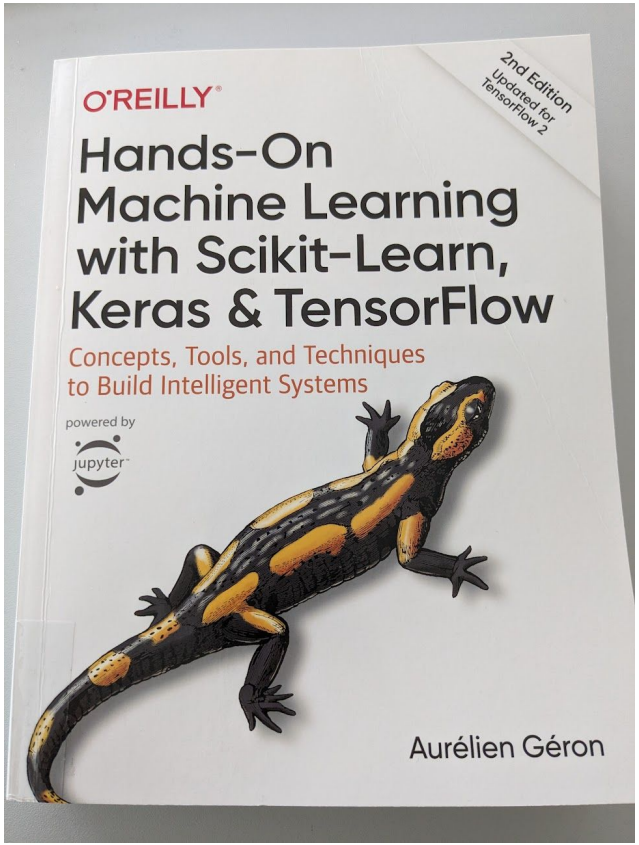
```
@SRR10056916.5114340 5114340 length=101
GATTAACATCATTTTGCATATTGTAAGCTGAATATTCTTGAGCGGTTCTACAACATTGCTTACACCCTCCTCCTCGATAATATCCACCGCAACATCCGTGG

@SRR10056916.5114341 5114341 length=101
GCCGCTTCCGCAGAGAGGTGTTCAGAGCTGGAGTTATCCAAGAGAGTAAACGCCGAAGATTTTTTTGAATCTACTCAAGAAAAGAAGAAAAGGAAGTGTCG

@SRR10056916.5114342 5114342 length=101
GTGAAAGTTACTACCGCCCTGGTGGCCGTGGTAGGGGCCGCGGCCGCGGAAGAGGTAATGGTGGTGGATATGGTGGAATCAGCAGTGTTTCAGCCCCATCC

@SRR10056916.5114343 5114343 length=101
CTAGGGATCTTTTTGATCTCATCAGAGATTTATACGTCAATAGAAAAAGTAGAGATGACCTAAGAGTAAAGAATCTAATATATTGAATATTGAAATGAGGG

@SRR10056916.5114344 5114344 length=100
AGACAGAGCTGTTTTTGCCCTGCAGAACTGCAACGGGATAGTTGAGGAGGCCACCGAGGAGGAGTTGATGGATGCTATGGCTCAGGCAGACTCCACTGGG

@SRR10056916.5114345 5114345 length=99
AGTATGTTAACACCTATCGCCGAACATTCACAAACAAAAAGGGGAAAGAGGTTAGCAAGGCTCCTAAGATCCAGAGGCTGGTAACACCATTGACTCTTC

@SRR10056916.5114346 5114346 length=100
ACTAGAAGATCATCACCCACTTTTACATCTTCAGCAAAACCATCATAGTTAACAGTGATGGTGCGTTCTGGAAGCGGTGGATCAAATGCACGAACAGTGA

@SRR10056916.5114347 5114347 length=101
GCTTAAGAAGGTGAAGAAGAGCAACGGTCAGATGCTGGCTATTAATGAGATTTTTGAGAAGAACCCAACAACAATCAAGAATTATGGTATTTGGCTCCGTT

@SRR10056916.5114348 5114348 length=101
TGAGTGGCGTCCTTGGATTCTGTCCCACACTTCTCTCGATTCCCCGGGTTCATCCTGAACATATAATGTGGAGAAATGCCATGAGGAAGGAGGCCAGTATT
```

.fastq

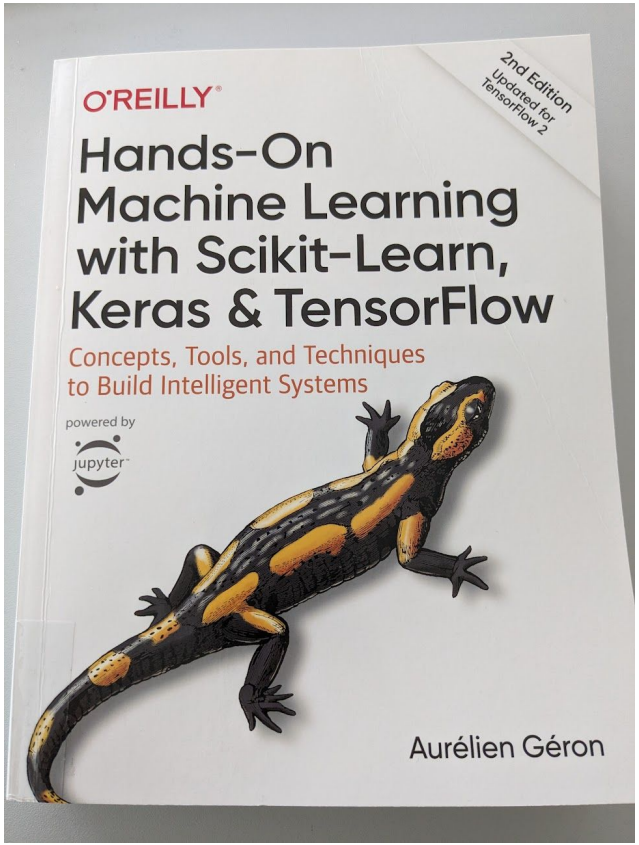
reference

```
ACGTTCTGGAGTGGCTGCAGCTCACCATGTATGCCTGAACCAGGCTGGCCTCATCCAAAG
CCATGCAGGATCACTCAGGATTTCAGTTTCACCCTCTATTTCCAAAGCCATTTACCTCAA
AGGACCCAGCAGCTACACCCCTACAGGCTTCCAGGCACCTCATCAGTCATGCTCCTCCTC
CATTTTACCCCCTACCCATCCTTGATCGGTCATGCCTAGCCTGACCCTTTAGTAAAGCAA
TGAGGTAGGAAGAACAAACCCTTGTCCCTTTGCCATGTGGAGGAAAGTGCCTGCCTCTGG
TCCGAGCCGCCTCGGTTCTGAAGCGAGTGCTCCTGCTTACCTTGCTCTAGGCTGTCTGCA
GAAGCACCTGCCGGTGGCACTCAGCACCTCCTTGTGCTAGAGCCCTCCATCACCTTCACG
CTGTCCCACCATGGGCCAGGAACCAAACCAGCACTGGGTTCTACTGCTGTGGGGTAAACT
AACTCAGTGGAATAGGGCTGGTTACTTTGGGCTGTCCAACTCATAAGTTTGGCTGCATTT
TGAAAAAAGCTGATCTAAATAAAGGCATGTGTATGGCTGGTCCCC
>ENST00000397163.8 cdna chromosome:GRCh38:15:42359501:42412317:1 gene:ENSG00000092529.25
gene_biotype:protein_coding transcript_biotype:protein_coding gene_symbol:CAPN3 description:calpain 3
[Source:HGNC Symbol;Acc:HGNC:1480]
ACTCTCTTTCTCTCTCCCTCTGGCATGCATGCTGCTGGTAGGAGACCCCCAAGTCAACAT
TGCTTCAGAAATCCTTTAGCACTCATTTCTCAGGAGAACTTATGGCTTCAGAATCACAGC
TCGGTTTTTAAGATGGACATAACCTGTACGACCTTCTGATGGGCTTTCAACTTTGAACTG
GATGTGGACACTTTTCTCTCAGATGACAGAATTACTCCAACTTCCCCTTTGCAGTTGCTT
CCTTTCCTTGAAGGTAGCTGTATCTTATTTTCTTTAAAAAGCTTTTTCTTCCAAAGCCAC
TTGCCATGCCGACCGTCATTAGCGCATCTGTGGCTCCAAGGACAGCGGCTGAGCCCCGGT
CCCCAGGGCCAGTTCCTCACCCGGCCCAGAGCAAGGCCACTGAGGCTGGGGGTGGAAACC
CAAGTGGCATCTATTCAGCCATCATCAGCCGCAATTTTCCTATTATCGGAGTGAAAGAGA
AGACATTCGAGCAACTTCACAAGAAATGTCTAGAAAAGAAAGTTCTTTATGTGGACCCTG
AGTTCCCACCGGATGAGACCTCTCTCTTTTATAGCCAGAAGTTCCCCATCCAGTTCGTCT
GGAAGAGACCTCCGGAAATTTGCGAGAATCCCCGATTTATCATTGATGGAGCCAACAGAA
CTGACATCTGTCAAGGAGAGCTAGGGGACTGCTGGTTTCTCGCAGCCATTGCCTGCCTGA
CCCTGAACCAGCACCTTCTTTTCCGAGTCATACCCCATGATCAAAGTTTCATCGAAAACT
```

.fasta



“tensor arrays”



Ingredients for an alignment

our data
(or <u>your</u> data)

.fastq

reference
~~sequences~~
index

.fasta

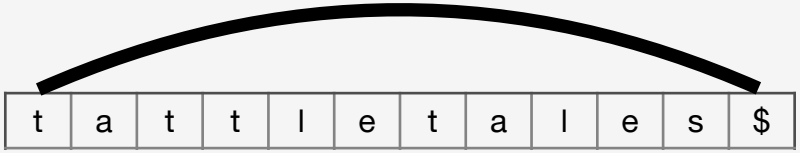
Building an index

- **Use Kallisto to build index from reference fasta**
  - *kallisto index -i inputFastaName.index inputFastaName.fa*
  - *kallisto index -i Homo_sapiens.GRCh38.cdna.all.index Homo_sapiens.GRCh38.cdna.all.fa*
- **Only need to build an index once**
- **Be careful with file names!**
  - *Long and meaningful file names are better than short ambiguous ones.*
  - *No spaces. ‘my_file_name’, or ’myFileName’, or ‘my.file.name’….just not, ‘my file name’*

Let’s construct an index from our reference fasta file

Burrows-Wheeler Transform (BWT)

‘tattletales’



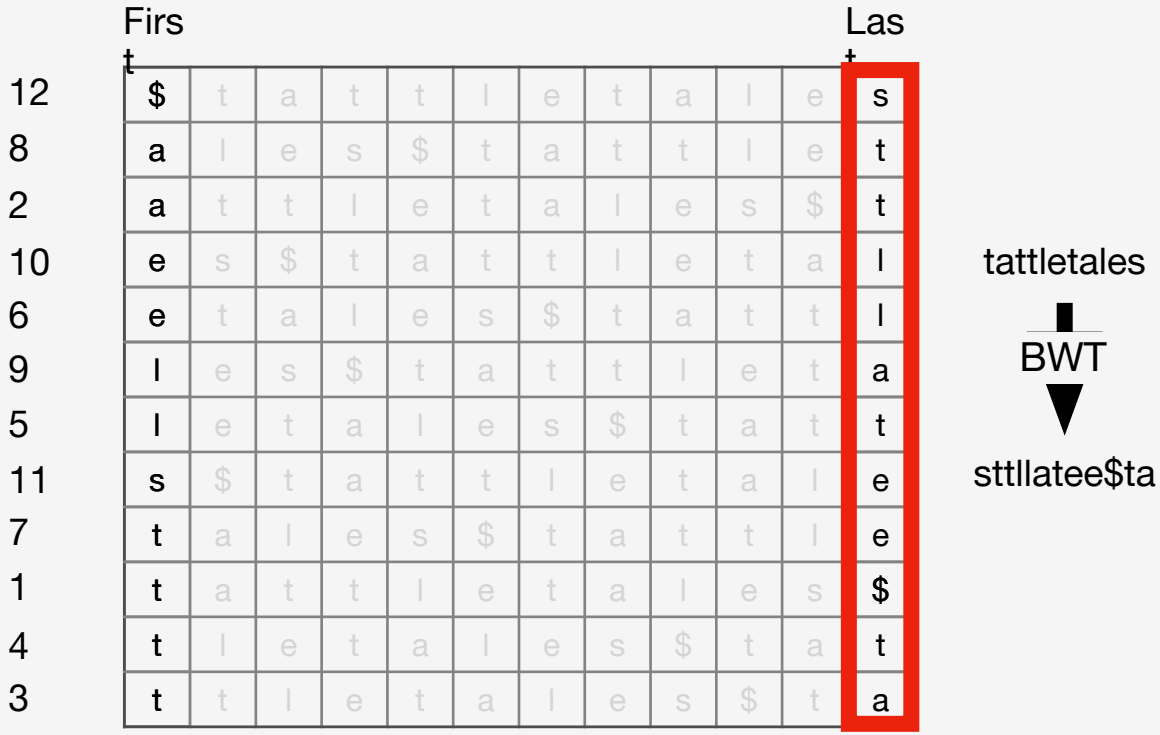
Burrows-Wheeler Transform (BWT)

‘tattletales’

12	$	t	a	t	t	l	e	t	a	l	e	s
8	a	l	e	s	$	t	a	t	t	l	e	t
2	a	t	t	l	e	t	a	l	e	s	$	t
10	e	s	$	t	a	t	t	l	e	t	a	l
6	e	t	a	l	e	s	$	t	a	t	t	l
9	l	e	s	$	t	a	t	t	l	e	t	a
5	l	e	t	a	l	e	s	$	t	a	t	t
11	s	$	t	a	t	t	l	e	t	a	l	e
7	t	a	l	e	s	$	t	a	t	t	l	e
1	t	a	t	t	l	e	t	a	l	e	s	$
4	t	l	e	t	a	l	e	s	$	t	a	t
3	t	t	l	e	t	a	l	e	s	$	t	a

Burrows-Wheeler Transform (BWT)

‘tattletales’



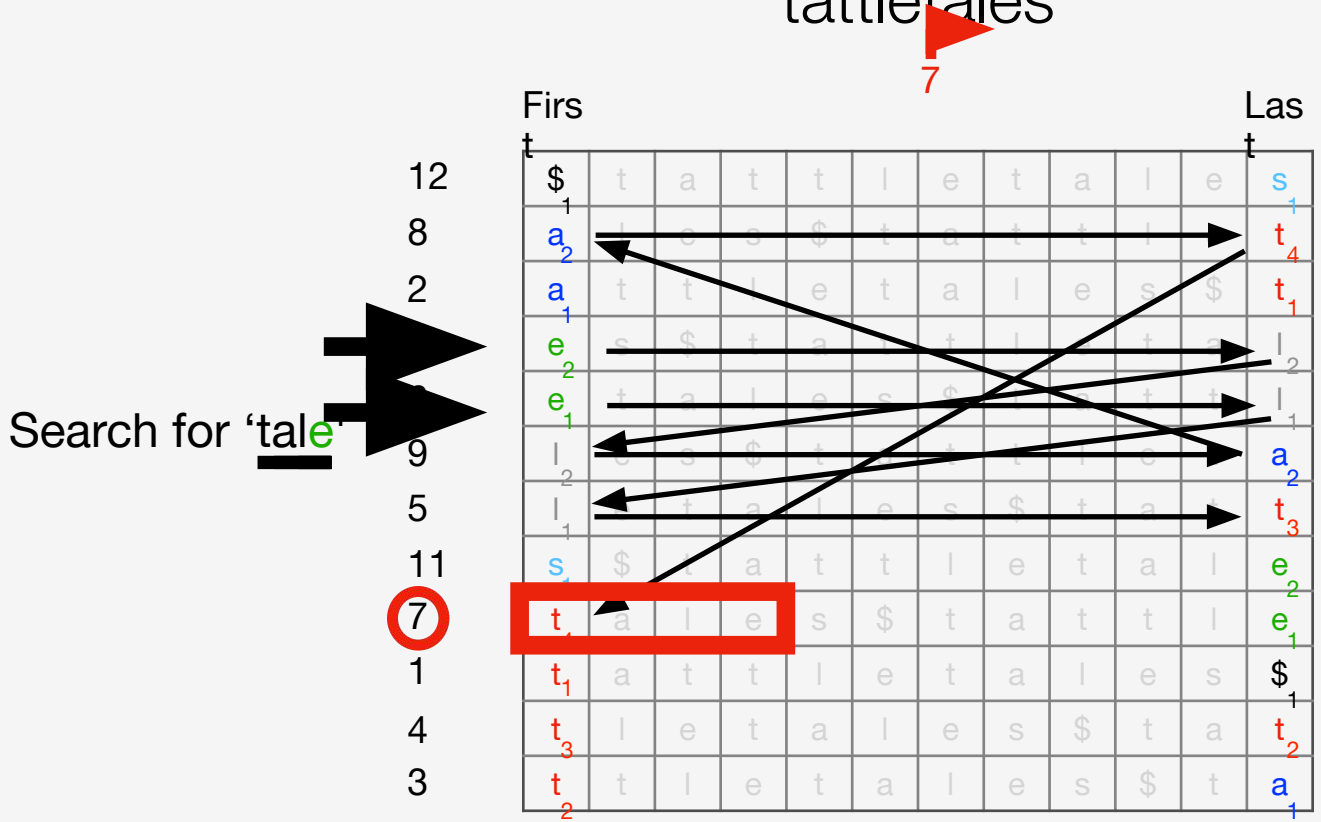
Burrows-Wheeler Transform (BWT)

‘tattletales’

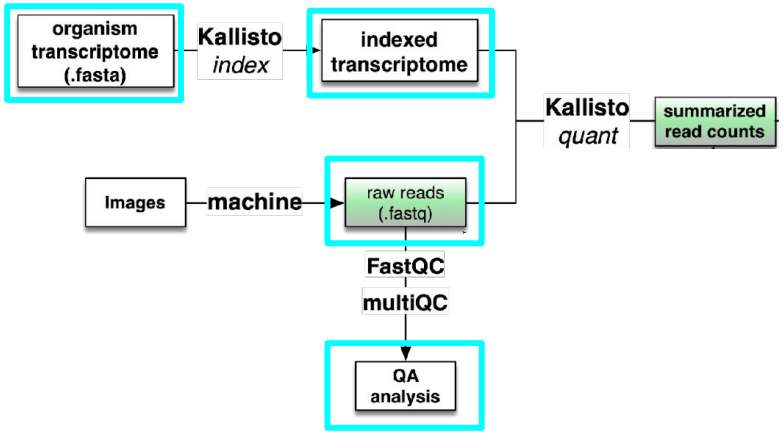
	First											Last
12	$	t	a	t	t	l	e	t	a	l	e	s
8	a	l	e	s	$	t	a	t	t	l	e	t_4
2	a	t	t	l	e	t	a	l	e	s	$	t_1
10	e	s	$	t	a	t	t	l	e	t	a	l
6	e	t	a	l	e	s	$	t	a	t	t	l
9	l	e	s	$	t	a	t	t	l	e	t	a
5	l	e	t	a	l	e	s	$	t	a	t	t_3
11	s	$	t	a	t	t	l	e	t	a	l	e
7	t_4	a	l	e	s	$	t	a	t	t	l	e
1	t_1	a	t	t	l	e	t	a	l	e	s	$
4	t_3	l	e	t	a	l	e	s	$	t	a	t_2
3	t_2	t	l	e	t	a	l	e	s	$	t	a

Burrows-Wheeler Transform (BWT)

‘tattletales’

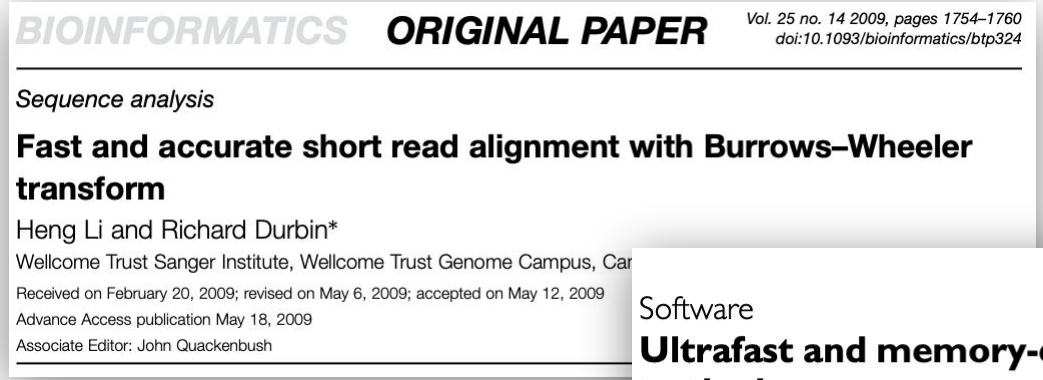


RNAseq analysis pipeline




```
~/Documents/my_study
├── scripts
│   ├── get_raw_reads.sh
│   ├── get_and_index_genome.sh
│   ├── quality_check.sh
│   └── map_reads.sh
├── genome
│   ├── cDNA_Solanum_lycopersicum.fasta
│   └── cDNA_Solanum_lycopersicum.index
├── raw_reads
│   ├── SRR15245007_1.fastq.gz
│   ├── SRR15245007_2.fastq.gz
│   ├── SRR15245013_1.fastq.gz
│   ├── SRR15245013_2.fastq.gz
│   └── etc…
├── publication
│   └── manuscript.pdf
├── QC
│   ├── qc_SRR15245007_1.html
│   └── etc….
└── mapped_reads
    ├── control1
    │   ├── abundance.h5
    │   ├── abundance.tsv
    │   └── run_info.json
    ├── control2
    │   ├── abundance.h5
    │   ├── abundance.tsv
    │   └── run_info.json
    └── etc…
        ├── abundance.h5
        ├── abundance.tsv
        └── run_info.json
```

Application of BWT in genomics



BIOINFORMATICS ORIGINAL PAPER

Vol. 25 no. 14 2009, pages 1754–1760
doi:10.1093/bioinformatics/btp324

Sequence analysis

Fast and accurate short read alignment with Burrows–Wheeler transform

Heng Li and Richard Durbin*

Wellcome Trust Sanger Institute, Wellcome Trust Genome Campus, Ca

Received on February 20, 2009; revised on May 6, 2009; accepted on May 12, 2009

Advance Access publication May 18, 2009

Associate Editor: John Quackenbush

~23,000 citations

Software

Open Access

Ultrafast and memory-efficient alignment of short DNA sequences to the human genome

Ben Langmead, Cole Trapnell, Mihai Pop and Steven L Salzberg

Address: Center for Bioinformatics and Computational Biology, Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA.

Correspondence: Ben Langmead. Email: langmead@cs.umd.edu

Published: 4 March 2009

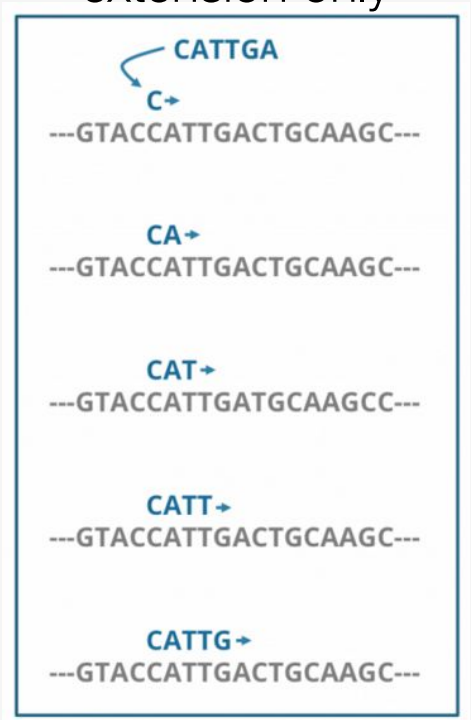
Genome ***Biology*** 2009, **10:**R25 (doi:10.1186/gb-2009-10-3-r25)

Received: 21 October 2008
Revised: 19 December 2008
Accepted: 4 March 2009

~16,000 citations

‘Seed and Extend’ alignments

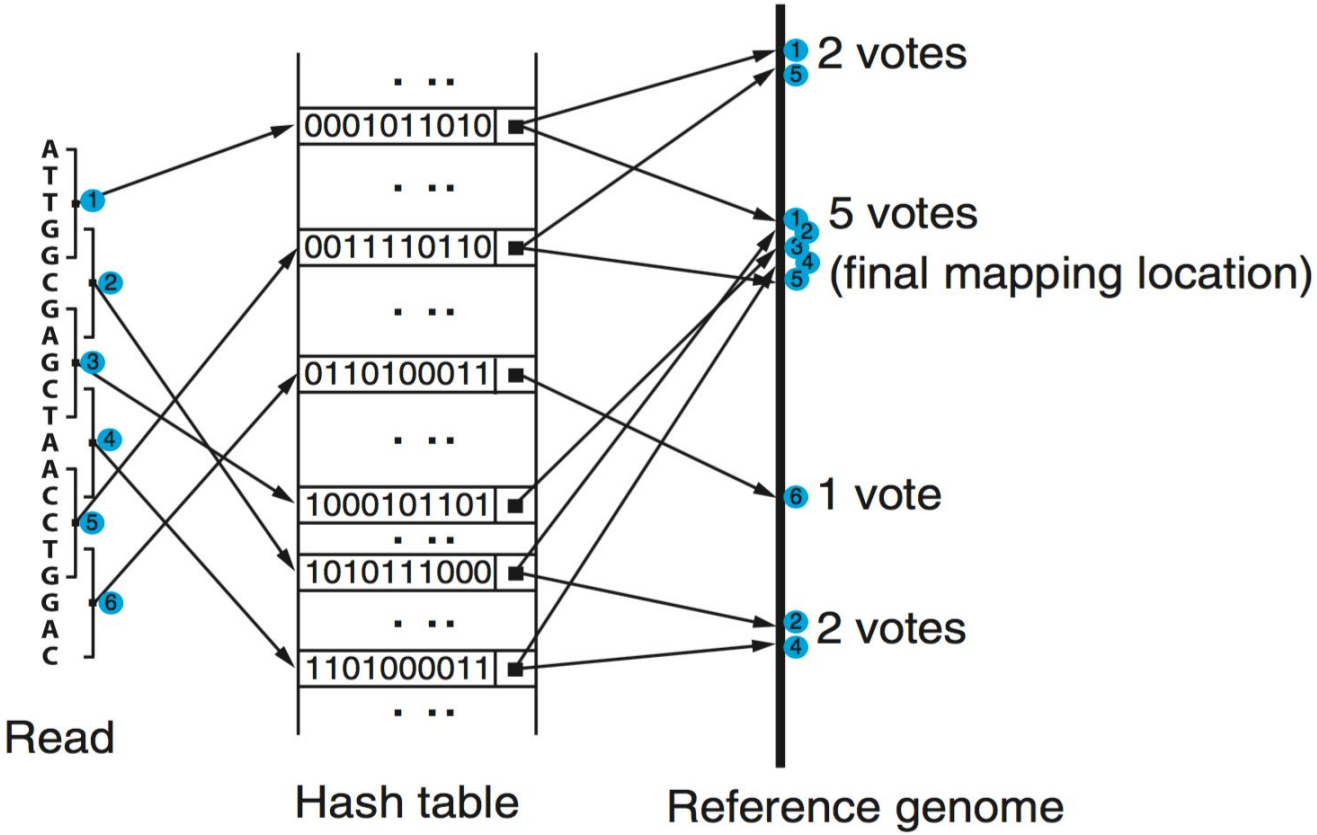
extension only



brute force, but slow

https://blog.sbgenomics.com/short-read-alignment-seeding/

R Subread aligner



Liao et al., NAR, 2013

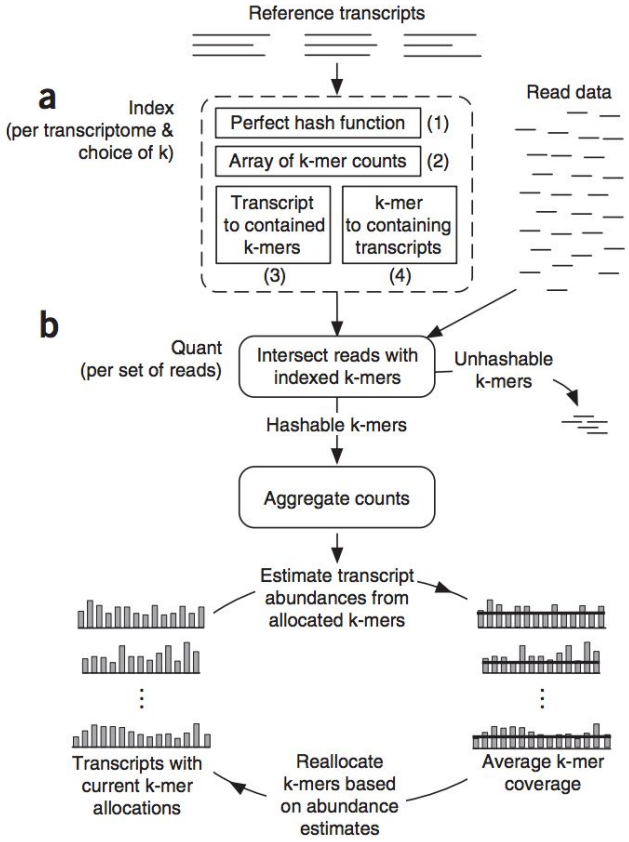
Reads are hard to align, but that's a function of their length. Why not approach the problem by shredding reads into Kmers - enter "Sailfish"

Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms

Rob Patro[1], Stephen M Mount[2,3] & Carl Kingsford[1]

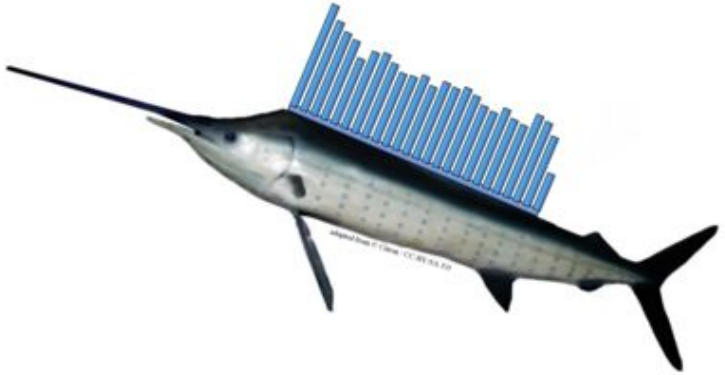
Nature Biotechnology **32**, 462–464 (2014)

Sailfish replaces *approximate* alignment of (error prone) *reads* with *exact* alignment of short *k-mers*



Sailfish

'lightweight read alignment'

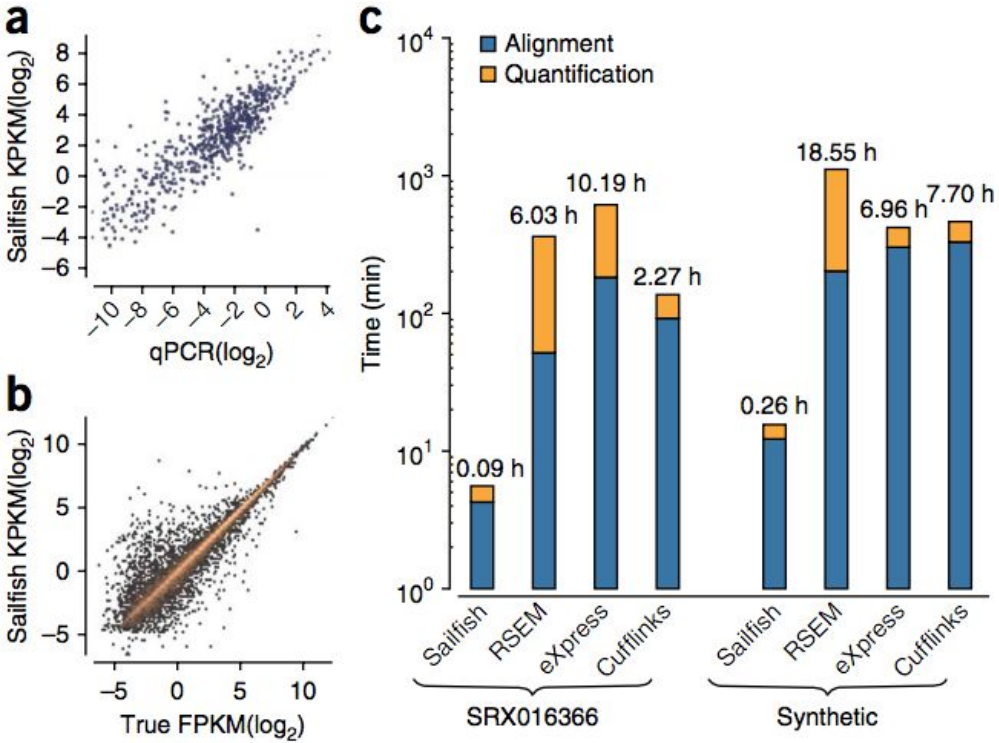


"By not requiring read mapping, **Sailfish avoids parameters specifying**, for example, the number of mismatches to tolerate, total allowable quality of mismatched bases, gap open and extension penalties, whether and how much to trim reads, number and quality of alignments to report from the aligner and pass into the estimation procedure."

Patro et al., Nat. Biotech., 2014

Sailfish

'lightweight read alignment'



Patro et al., Nat. Biotech., 2014

What if the idea of alignment was altogether **abandoned** enter “Kallisto”

nature biotechnology

Nature Biotechnology **34**, 525–527 (2016)

Near-optimal probabilistic RNA-seq quantification

Nicolas L Bray[1], Harold Pimentel[2], Páll Melsted[3] & Lior Pachter[2,4,5]

We present kallisto, an RNA-seq quantification program that is two orders of magnitude faster than previous approaches and achieves similar accuracy. Kallisto pseudoaligns reads to a reference, producing a list of transcripts that are compatible with each read while avoiding alignment of individual bases. We use kallisto to analyze 30 million unaligned paired-end RNA-seq reads in <10 min on a standard laptop computer. This removes a major computational bottleneck in RNA-seq analysis.

The first two steps in typical transcript-level RNA-seq processing

this information, we develop a method based on pseudoalignment of reads and fragments, which focuses only on identifying the transcripts from which the reads could have originated and does not try to pinpoint exactly how the sequences of the reads and transcripts align.

A pseudoalignment of a read to a set of transcripts, T, is a subset, $S \subseteq T$, without specific coordinates mapping each base in the read to specific positions in each of the transcripts in S. Accurate pseudoalignments of reads to a transcriptome can be obtained using fast hashing of k-mers together with the transcriptome de Bruijn graph (T-DBG). de Bruijn graphs have been crucial for DNA and RNA assembly[8], where they are usually constructed from reads. Kallisto uses a T-DBG, which is a de Bruijn graph constructed from k-mers present in the transcriptome (**Fig. 1a**), and a path covering of the graph, a set of paths whose union covers all edges of the graph, where the paths correspond to transcripts (**Fig. 1b**). This path covering of a T-DBG induces multi-sets on the vertices, called k-compatibility classes. A compatibility class can be associated to an error-free read by representing it as a path in the graph and defining the k-compatibility

Quantitative Biology > Genomics

Sailfish: Alignment-free Isoform Quantification from RNA-seq Reads using Lightweight Algorithms

Rob Patro (1), Stephen M. Mount (2), Carl Kingsford (1) ((1) Lane Center for Computational Biology, School of Computer Science, Carnegie Mellon University, (2) Department of Cell Biology and Molecular Genetics and Center for Bioinformatics and Computational Biology,

(Submitted on 16 Aug 2013)

RNA-seq is rapidly become the de facto technique to measure gene expression. However, the time required for analysis has not kept up with the pace of data generation. Here we introduce Sailfish, a novel computational method for quantifying the abundance of previously annotated RNA isoforms from RNA-seq data. Sailfish entirely avoids mapping reads, which is a time-consuming step in all current methods. Sailfish provides

Cornell University Library

We gratefully acknowledge support from the Simons Foundation and University of Pennsylvania

arXiv.org > q-bio > arXiv:1505.02710

Search or Article ID All fields

(Help | Advanced search)

Quantitative Biology > Quantitative Methods

Near-optimal RNA-Seq quantification

Páll Melsted, Lior Pachter

(Submitted on 11 May 2015 (last revised 15 May 2015 (this version, v2))

approach to RNA-Seq quantification that is near optimal in speed and accuracy. Software implementing the approach, called kallisto, can be used to analyze 30 million unaligned paired-end RNA-Seq reads in less than 5 minutes on a standard laptop computer while providing results as accurate as those of the best existing tools. This removes a major computational bottleneck in RNA-Seq analysis.

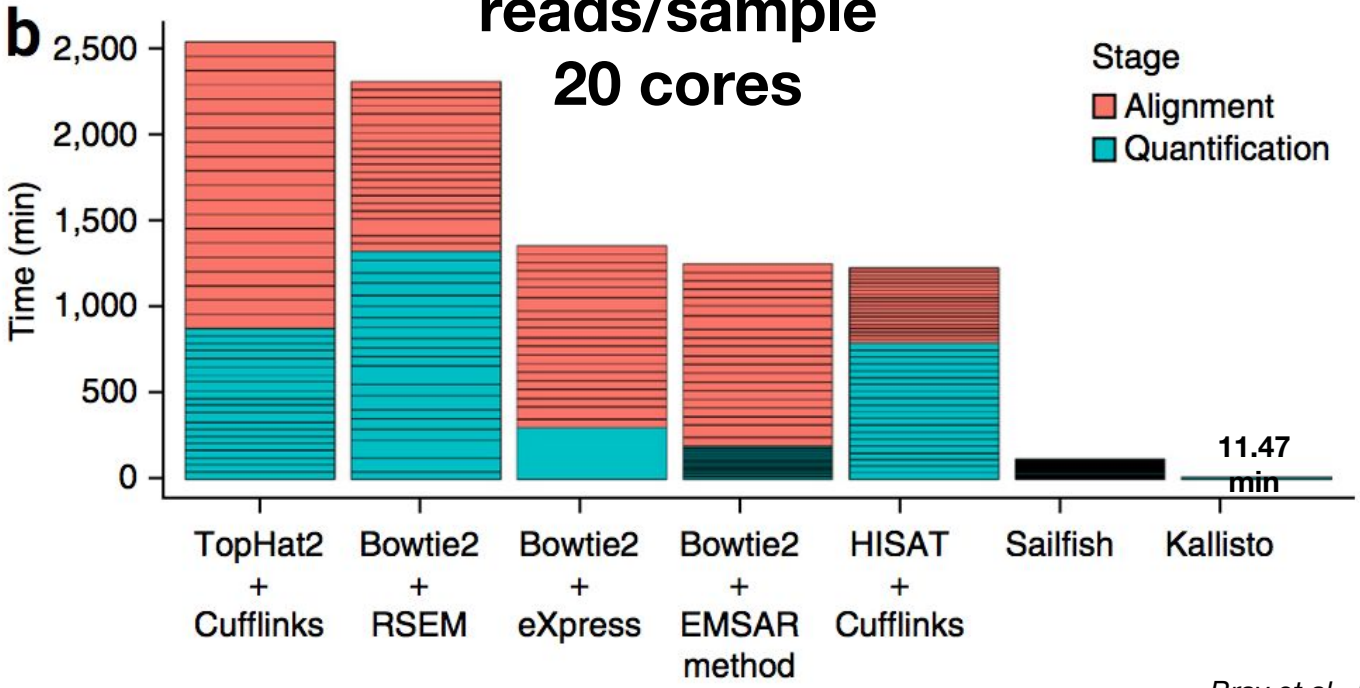
Comments: - Added some results (paralog analysis, allele specific expression analysis, alignment comparison, accuracy analysis with TPMs) - Switched bootstrap analysis to human sample from SEQC-MAQCIII - Provided link to a snakefile that allows for reproducibility of all results and figures in the paper
Subjects: **Quantitative Methods (q-bio.QM)**; Computational Engineering, Finance, and Science (cs.CE); Data Structures and Algorithms (cs.DS); Genomics (q-bio.GN)
Cite as: **arXiv:1505.02710 [q-bio.QM]**
(or **arXiv:1505.02710v2 [q-bio.QM]** for this version)

Submission history

From: Lior Pachter [view email]
[v1] Mon, 11 May 2015 17:42:04 GMT (3410kb)
[v2] Fri, 15 May 2015 17:12:58 GMT (4940kb)

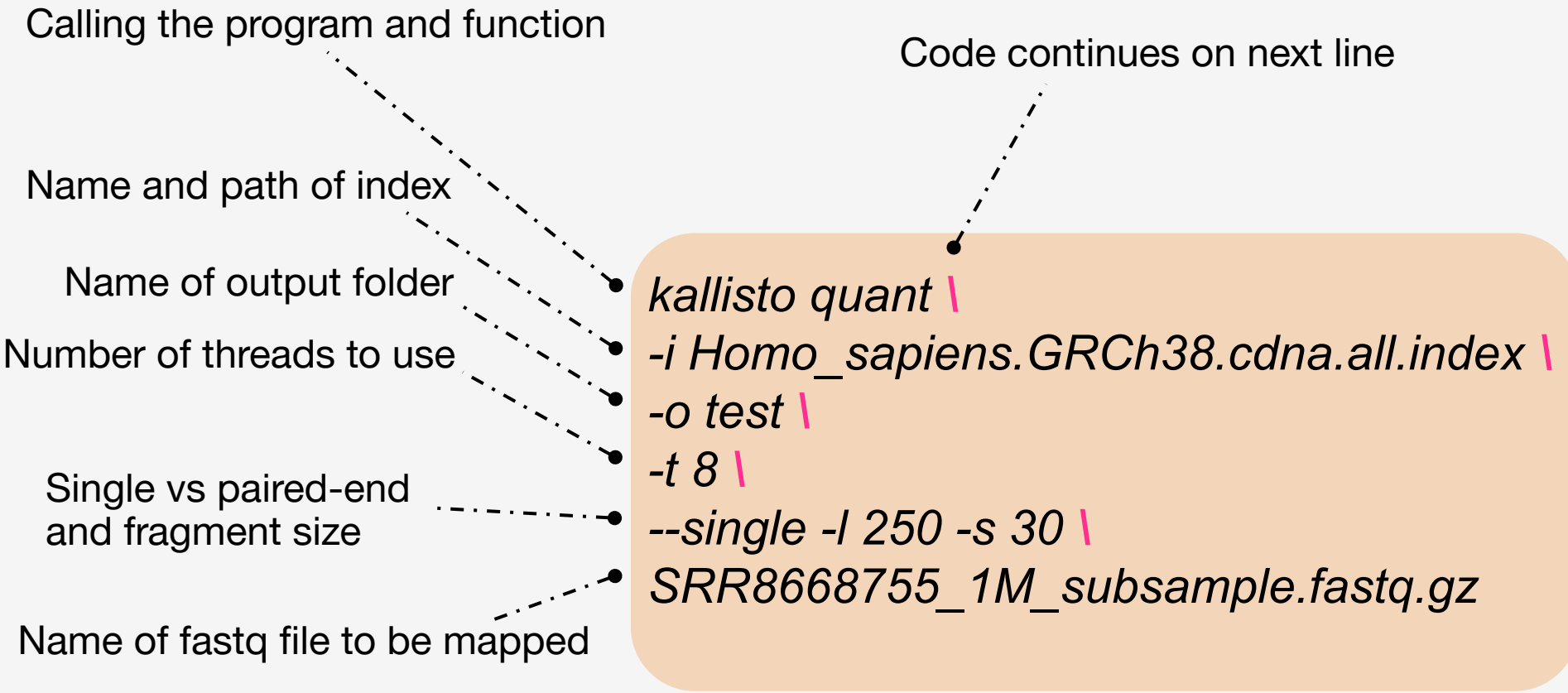
Kallisto - it's fast

20 samples
30 million PE reads/sample
20 cores

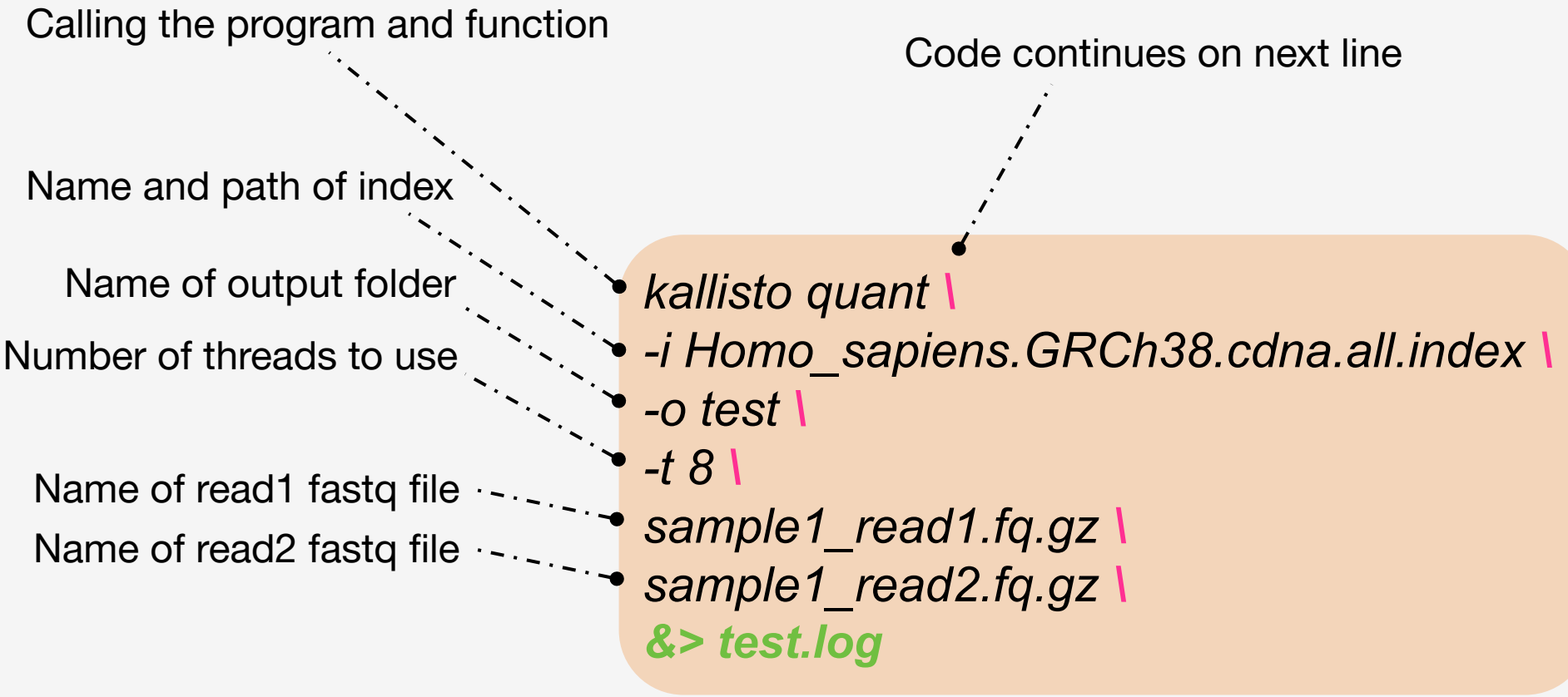


Bray et al., Nat. Biotech., 2016

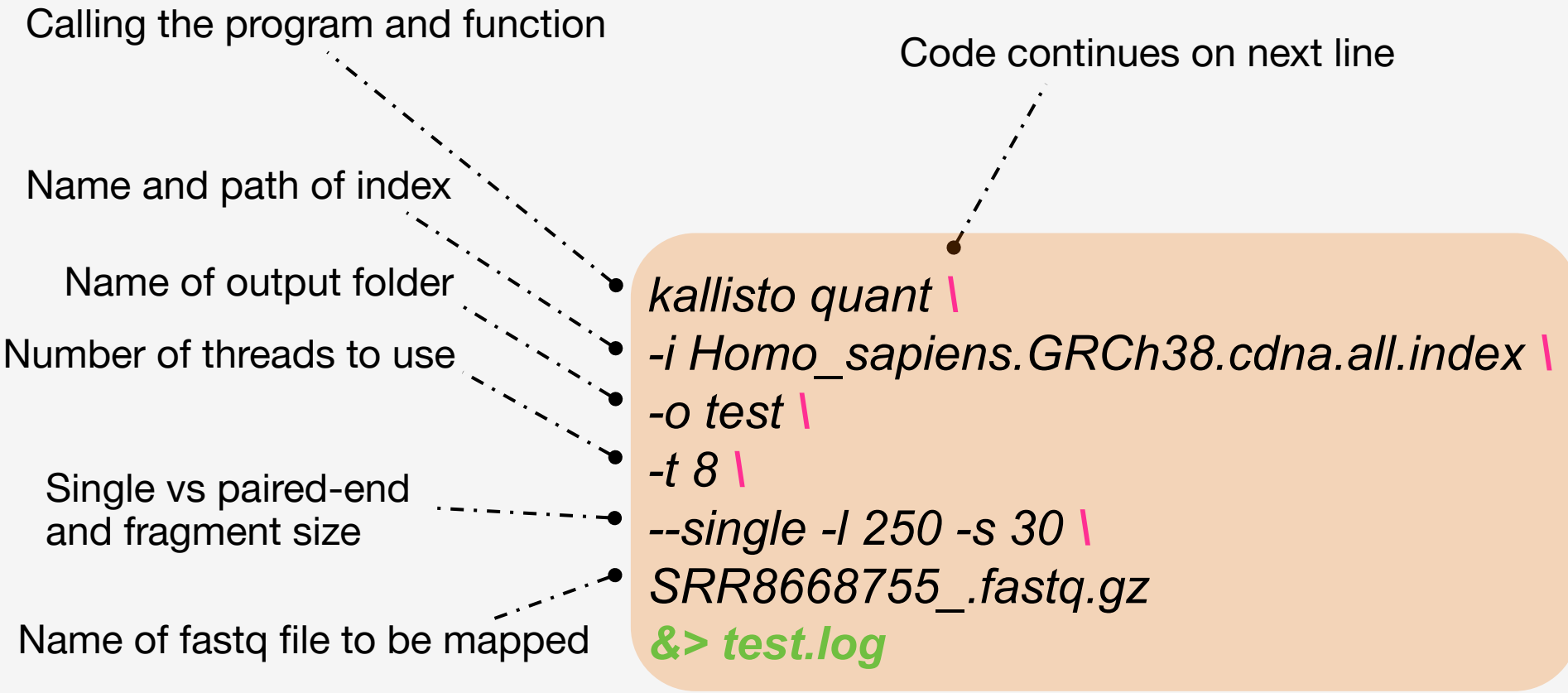
Mapping single-end data



Mapping paired-end data



Mapping single-end data



Kallisto demonstration

```
kallisto quant \
-i Homo_sapiens.GRCh38.cdna.all.index \
-o test \
-t 8 \
sample1_read1.fq.gz \
sample1_read2.fq.gz \
&> test.log
```