Joo et al. BMC Bioinformatics (2019) 20:90
https://doi.org/10.1186/s12859-019-2676-x

BMC Bioinformatics

SOFTWARE Open Access

SEQprocess: a modularized and customizable pipeline framework for NGS processing in R package

Taewoon Joo[1,2], Ji-Hye Choi[1,2], Ji-Hye Lee[1,2], So Eun Park[1,2], Youngsic Jeon[3,4], Sae Hoon Jung[5] and Hyun Goo Woo[1,2*]

Abstract

Backgrounds: Next-Generation Sequencing (NGS) is now widely used in biomedical research for various applications. Processing of NGS data requires multiple programs and customization of the processing pipelines according to the data platforms. However, rapid progress of the NGS applications and processing methods urgently require prompt update of the pipelines. Recent clinical applications of NGS technology such as cell-free DNA, cancer panel, or exosomal RNA sequencing data also require appropriate customization of the processing pipelines. Here, we developed SEQprocess, a highly extendable framework that can provide standard as well as customized pipelines for NGS data processing.

Results: SEQprocess was implemented in an R package with fully modularized steps for data processing that can be easily customized. Currently, six pre-customized pipelines are provided that can be easily executed by non-experts such as biomedical scientists, including the National Cancer Institute's (NCI) Genomic Data Commons (GDC) pipelines as well as the popularly used pipelines for variant calling (e.g., GATK) and estimation of allele frequency, RNA abundance (e.g., TopHat2/Cufflink), or DNA copy numbers (e.g., Sequenza). In addition, optimized pipelines for the clinical sequencing from cell-free DNA or miR-Seq are also provided. The processed data were transformed into R package-compatible data type 'ExpressionSet' or 'SummarizedExperiment', which could facilitate subsequent data analysis within R environment. Finally, an automated report summarizing the processing steps are also provided to ensure reproducibility of the NGS data analysis.

Conclusion: SEQprocess provides a highly extendable and R compatible framework that can manage customized and reproducible pipelines for handling multiple legacy NGS processing tools.

Keywords: Next generation sequencing, Whole exome sequencing, RNA sequencing, Preprocessing, Pipeline

Background

Next-Generation Sequencing (NGS) technology is now widely used in biomedical research fields, and is extensively being used in the clinic [9]. Applications with NGS technology include identification of DNA or RNA sequence variants, and the quantitation of RNA abundances or DNA copy numbers. However, processing and analysis of NGS data remain difficult as data are generally processed through by multiple processing steps, and each step requires different legacy programs. To handle these complex processing steps, several pipeline programs have been released. For example, 'NGS-pipe' [18] and 'NEAT' [17] provide automated pipelines for NGS data analysis. Another tool 'systemPipeR' provides an NGS analysis workflow in R program that can be customized according to the various NGS applications such as whole-exome sequencing (WES), whole-genome sequencing (WGS) and transcriptome sequencing (RNA-seq) data [2]. However, these tools do not handle the recently updated NCI Genomic Data Commons (GDC) pipelines, which have been used as

* Correspondence: hg@ajou.ac.kr
[1]Department of Physiology, Ajou University School of Medicine, 164 Worldcup-ro, Yeongtong-gu, Suwon 16499, Republic of Korea
[2]Department of Biomedical Science, Graduate School, Ajou University, Suwon, Republic of Korea
Full list of author information is available at the end of the article

standard pipelines to process The Cancer Genome Atlas (TCGA, https://cancergenome.nih.gov) data. Moreover, recent progress in clinical applications of the NGS data has generated new platform data, such as cell free DNAs, exosomes, and cancer panels. These applications require customized analysis for data quality control and processing.

With this concern, we developed a SEQprocess that provides fully customizable NGS processing pipelines covering the GDC pipelines and new data for clinical applications. SEQprocess is implemented in an R program, providing six pre-customized pipelines that are widely used as standards in NGS data processing and can be executed easily by non-experts such as biomedical scientists.

Implementation

SEQprocess is a framework implemented in R package, providing pipelines for NGS data processing operated by multiple programs. It can be run from start-to-end with a single command in the R console, or through stepwise customization with an interactive mode. The pipelines are designed to support processing pipelines for DNA and RNA sequencing data, including the data processing steps for quality control of raw sequencing data, trimming, alignment, variant calling, annotation, DNA copy number estimation and RNA quantitation. Each pipeline is modularized to run sequentially or separately. The following programs are supported by the pipelines. Quality control of raw data is assessed by FastQC (https://www.bioinformatics.babraham.ac.uk). Sequence trimming is performed by TrimGalore (https://github.com/broadinstitute/picard) or Cutadapt [14]. Sequence alignment is supported by BWA [12], STAR[3], TopHat2 [7], bowtie2 [10], or samtools [13]. Removal of duplicates is performed by Picard (https://github.com/broadinstitute/picard) and re-alignment by GATK [15]. Variants calling is supported by GATK, VarScan2 [8], MuSE [4], or SomaticSniper [11]. Variant annotation is supported by VEP [16] or ANNOVAR [20]. For RNA-seq data, SEQprocess performs RNA quantitation by HTSeq [1] or Cufflinks[19], and DNA copy number estimation is conducted by Sequenza [5]. These programs are implemented as modularized functions with optimized default parameters. These external programs can be installed easily using Conda package manager (https://conda.io/en/latest). Subsequent steps for NGS data processing can be easily included or excluded in the pipeline. This modular framework provides a highly flexible and extendable platform; thus, new pipelines for upcoming data types such as single cell RNA-Seq data can be implemented.

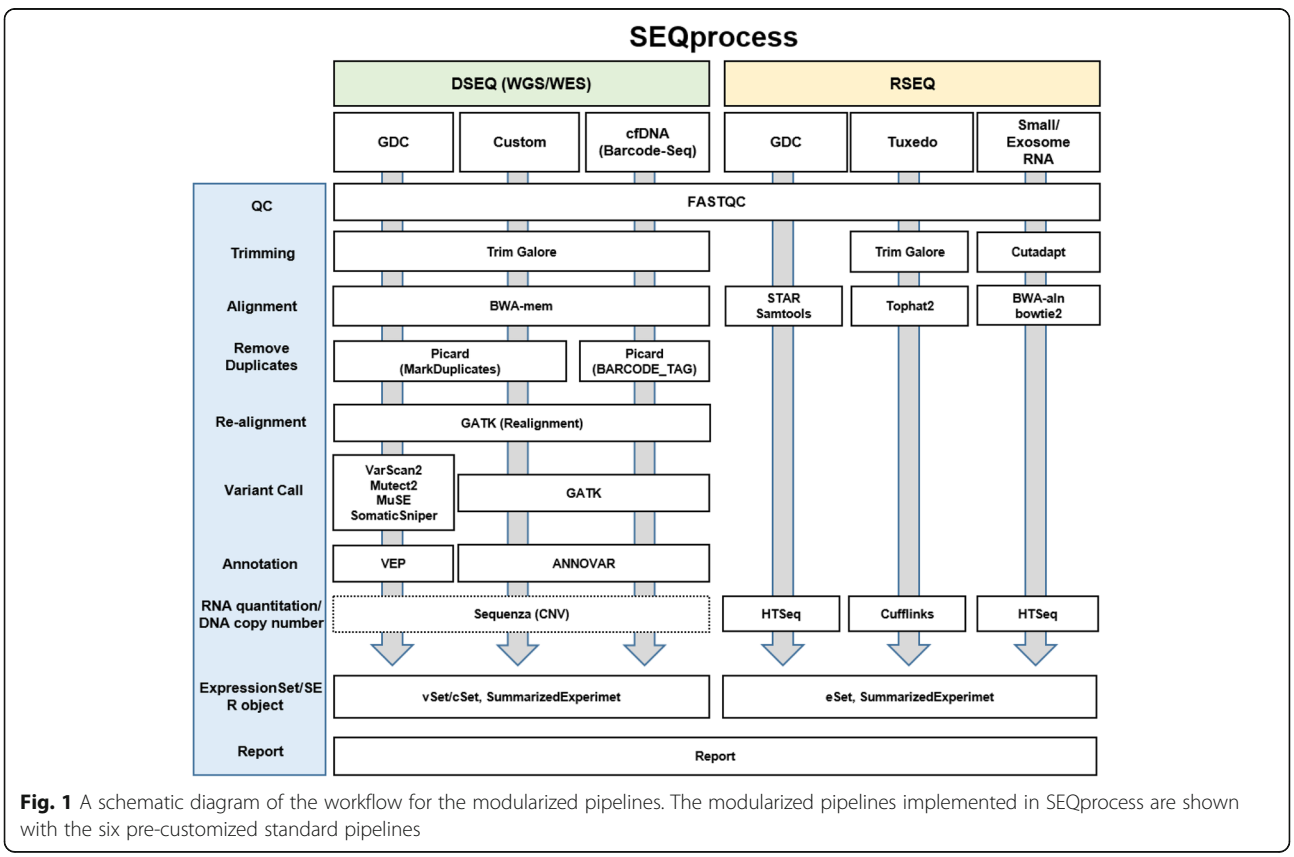

Fig. 1 A schematic diagram of the workflow for the modularized pipelines. The modularized pipelines implemented in SEQprocess are shown with the six pre-customized standard pipelines

Results

The current version of SEQprocess provided six different pre-customized standard pipelines, including the pipelines for GDC processing and the newly adapted clinical applications for cell-free DNAs (cfDNA) or exosomal miRNAs (Fig. 1). These pipelines ran by a one-step command that could be executed easily by non-expert users. For WGS/WES, a GDC compatible pipeline of TrimGalore-BWA--Picard- VarScan2-VEP was implemented. We also implemented a popularly used standard Custom pipeline of TrimGalore-BWA-Picard-GATK–ANNOVAR. In addition, SEQprocess could estimate allele frequencies for each variant by calculating the sequence read depths of the mutated and wild-type sequences with a GATK function 'DepthOfCoverage'. For liquid-biopsied cfDNA or targeted sequencing data, such as a cancer panel, an optimized pipeline excluding the duplicate removal step was provided, because cfDNA sequence reads usually have the same sequences. For barcoded data (BarSeq), the duplicate removal step was performed using the barcodes. For RNA-Seq data, a GDC pipeline (STAR-Samtools-HT-Seq) was implemented. A popularly used standard pipeline Tuxedo (i.e., Tophat2-Cufflinks) was also implemented. For miR-Seq data from exosomes, cells, or tissues, the Cutadapt-BWA/bowtie2-HTSeq pipeline was implemented with optimized parameters.

Table 1 Parameters implemented in SEQprocess

Analysis Steps	Parameters	Description	Values
None	fastq.dir	Fastq file path	File path
	output.dir	Output directory	File path
	config.fn	Configure file path	File path
	project.name	Project name	Name
	type	Data type	WGS, WES, BarSEQ, RSEQ, miRSEQ
	pipeline	Select data processing pipeline	none, GDC, GATK, BarSEQ, Tuxedo, miRSEQ
	mc.cores	Number of multi core	Numeric
	run.cmd	Whether to execute the command line	Logical
QC	QC	Quality Check (FastQC)	Logical
Trimming	trim.method	Trimming (Cutadapt, TrimGalore)	trim.galore, cutadapt, none
Alignment	align.method	Alignment (BWA, Tophat2, STAR, Bowtie2)	bwa, tophat2, star, bowtie2, none
	build.transcriptome.idx	Transcriptome criterion generation in tophat	Logical
	tophat.thread.number	Number of threads	Numeric
	bwa.method	Select BWA method	mem, aln
	bwa.thread.number	Number of threads	Numeric
	star.thread.number	Number of threads	Numeric
Remove Duplicates	rm.dup	Whether to execute Picard MarkDuplicates	MarkDuplicates, BARCODE, none
Re-alignment	realign	Whether Re-alignment	Logical
Variant Call	variant.call.method	Select variant calling method	gatk, varscan2, mutect2, muse, somaticsniper, none
	gatk.thread.number	Number of threads	Numeric
	mut.cnt.cutoff	Read depth criterion determining the presence or absence of mutation	Numeric
Annotation	annotation.method	Select variant annotation method	annovar, vep
	ref	Reference version	Default = hg38
RNA quantitation	rseq.abundance.method	Select RNA quantitation method	cufflinks, htseq, none
	cufflinks.gtf	Whether detection novel genes and isoforms	-G, –g
	cufflinks.thread.number	Number of threads	Numeric
	RNAtype	Type of RNA	mRNA, miRNA
DNA copy number	CNV	Whether quantitation CNV	Logical
ExpressionSet/SE R object	make.eSet	Make ExpressionSet Rdata	Logical
	eset2SummarizedExperiment	Convert eSet to SE	Logical
Report	report.mode	Creating report file	Logical

Table 2 External programs and data files used in SEQprocess

Pipeline	Required R package	Programs path	Reference path
No matter	parallel		
Report	Limma, data.table, fastqcr, pander, knitr, png, grid, gridExtra, ggplot2, reshape2		
QC		fastqc.dir	
Trimming	.	trim_galore.path cutadapt.path	
Alignment	.	bwa.path tophat2.path bowtie2.path STAR.path samtools.path	ref.fa chrom.fa bwa.idx bowtie.idx star.idx.dir transcriptome.idx
Remove Duplicates	.	picard.path	ref.fa chrom.fa
Re-alignment	.	GATK.path	ref.fa ref.gold_indel
Variant Call		varscan.path MuSE.path somaticsniper.path	ref.gold_indel ref.dbSNP cosmic.vcf
Annotation	.	vep.path vcf2annovar.pl table_annovar.pl	annovar.db.dir vep.dir
RNA quantitation	GenomicRanges	cufflinks.path htseq.path	ref.gtf mir.gff refGene.path
DNA copy number	sequenza	sequenza.util	ref.fa
ExpressionSet/SE R object	Biobase, GenomicRanges, SummarizedExperiment		

SEQprocess operates multiple legacy programs and reference data, which might require installation in the system. Configuration of the installed programs and data could be managed simply by editing the 'data/config.R' file (Table 1). The current version of SEQprocess supported the Linux-operating system because some of the required programs only support the Linux-operating system. Parallel computation on multi-core machines was also supported by using the 'parallel' R package. In addition, multi-threading support in each program of GATK, TopHat2, BWA, STAR, and Cufflinks could be controlled by the program arguments.

Each step of these pipelines are modularized as a wrapper function in R package to provide an easy customization platform. Step-by-step pipelines could be conducted by a single command 'SEQprocess', and which

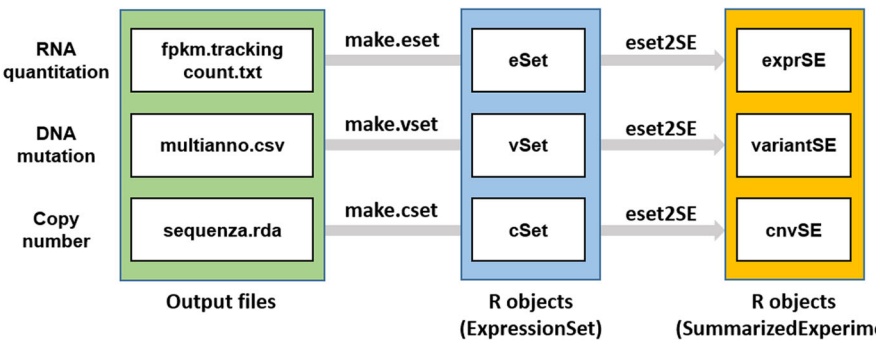

Fig. 2 Workflows for formatting output files by SEQprocess. Output files generated by the pipelines are transformed into Bioconductor-compatible data types of 'ExpressionSet' or 'SummarizedExperiment'. Different data types of RNA abundance, mutation, and DNA copy numbers are transformed into an 'ExpressionSet' with different names of eSet, vSet, and cSet, respectively. Each of 'ExpressionSet' data can be further transformed into another data type 'SummarizedExperiment'

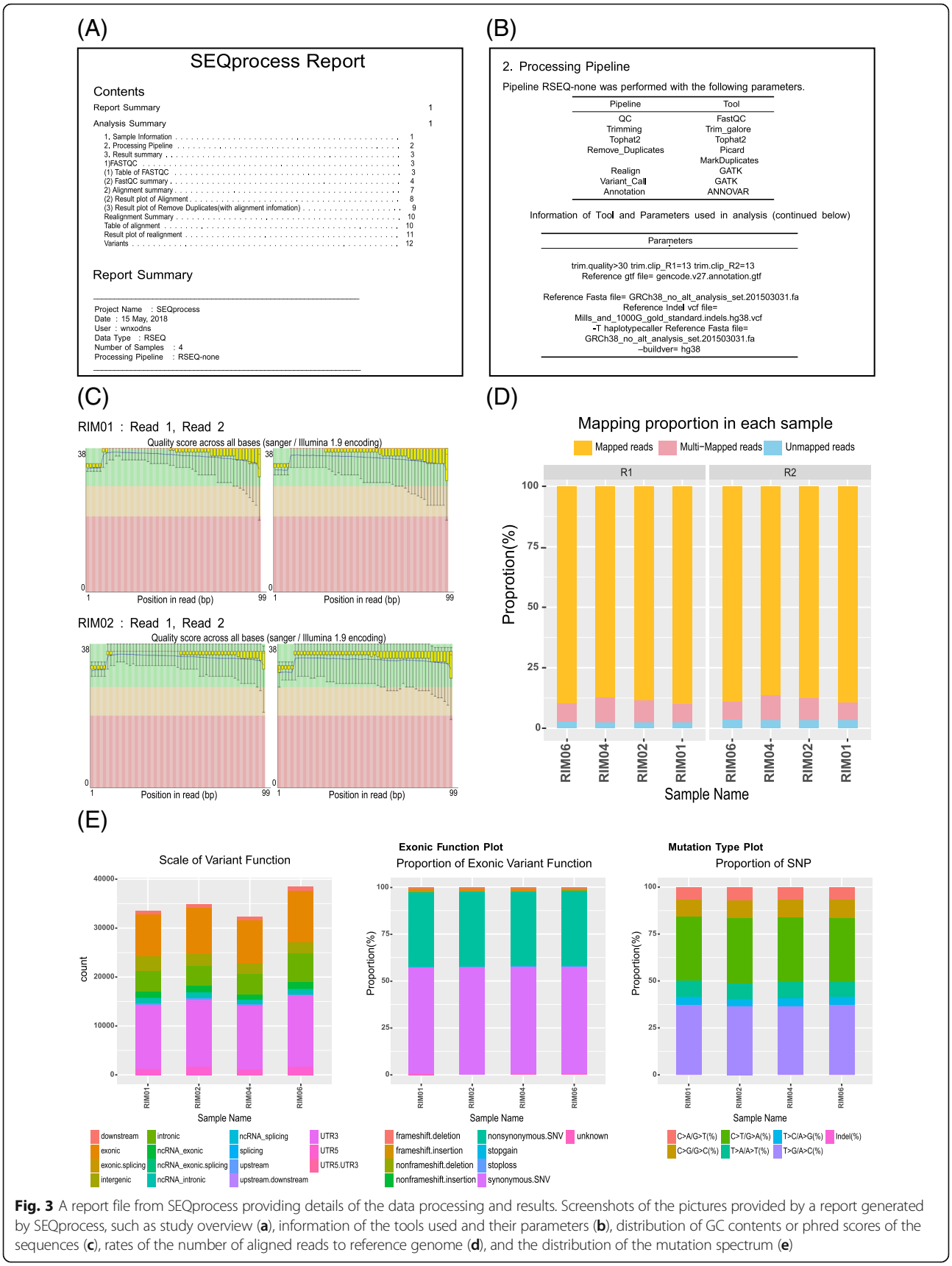

Fig. 3 A report file from SEQprocess providing details of the data processing and results. Screenshots of the pictures provided by a report generated by SEQprocess, such as study overview (**a**), information of the tools used and their parameters (**b**), distribution of GC contents or phred scores of the sequences (**c**), rates of the number of aligned reads to reference genome (**d**), and the distribution of the mutation spectrum (**e**)

could be readily customized by setting the function parameters (Table 2). The processed data were transformed into an R/Bioconductor compatible data type (i.e. 'ExpressionSet'), which is popularly used for the subsequent NGS data analysis for biological interpretation [6]-. Each data object for RNA expression, variant, and DNA copy number was provided with the filename extensions of '.eSet', '.vSet', or 'cSet', respectively. These ExpressionSet data types could be transformed into another data type 'SummariazedExperiment', i.e. a modified data type of 'ExpressionSet' containing 'GenomicRanges' data type (Fig. 2). These will serve as a framework facilitating the subsequent analyses in the R environment.

In addition, SEQprocess provided a report summarizing the processing steps and visualized tables and plots for the processed results (Fig. 3). The report file is automatically generated workflow records for data processing steps, arguments, and outcome results. Moreover, users can find error and processing messages from the log file in each program. These reporting systems will ensure the reproducibility of the data analysis. We have also provided an example data ('inst/example') and a script ('example/example.R').

Conclusions

In summary, SEQprocess provides a highly extendable and R-compatible framework that can be manage customized and reproducible pipelines for handling multiple legacy NGS processing tools.

Availability and requirements

Project name: SEQprocess.

Project home page: https://github.com/omicsCore/SEQprocess

Operating systems: Linux dependent.

Programming language: R language.

Other requirements: Java 1.8.0 or higher, Perl v5.10.1 or higher, Python 2.6.6 or higher.

License: GPL2.

Abbreviations

cfDNA: Cell-free DNA; GDC: Genomic Data Commons; miRNA: Mircro RNA; miR-Seq: Micro RNA sequencing; NCI: National Cancer Institute; NGS: Next Generation Sequencing; RNA-seq: RNA sequencing; TCGA: The Cancer Genome Atlas; WES: Whole Exome Sequencing; WGS: Whole Genome Sequencing

Acknowledgements

Not applicable.

Funding

This work was supported by grants from the Korea Health Technology R&D Project through the Korea Health Industry Development Institute (KHIDI) funded by the Ministry of Health & Welfare, Republic of Korea (H15C1551) and the National Research Foundation of Korea (NRF) funded by the Korea government (MSIP) (NRF-2017R1E1A1A01074733, NRF-2017M3C9A6047620, and NRF- 2017M3A9B6061509).
Funding institutes did not play any roles in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

Author's contributions

TJ implemented pipelines and R functions, and wrote the manuscript. JHC implemented pipelines and R functions. JHL implemented report ability. SEP, YJ and SHJ wrote manuals and vignettes. HGW implemented pipelines and R functions, wrote the manuscript, and conducted a thorough review, correction and revision. All authors read and approved the final manuscript.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Author details

[1]Department of Physiology, Ajou University School of Medicine, 164 Worldcup-ro, Yeongtong-gu, Suwon 16499, Republic of Korea. [2]Department of Biomedical Science, Graduate School, Ajou University, Suwon, Republic of Korea. [3]Department of Pathology, Yonsei University College of Medicine, Seoul, Republic of Korea. [4]BK21 PLUS Project for Medical Science, Yonsei University College of Medicine, Seoul, Republic of Korea. [5]Ajou University School of Medicine, Suwon, Republic of Korea.

Received: 19 October 2018 Accepted: 12 February 2019
Published online: 20 February 2019

References

1. Anders S, Pyl PT, Huber W. HTSeq--a Python framework to work with high-throughput sequencing data. Bioinformatics. 2015;31(2):166–9.
2. Backman TWH, Girke T. systemPipeR: NGS workflow and report generation environment. BMC Bioinformatics. 2016;17(1):388.
3. Dobin A, et al. STAR: ultrafast universal RNA-seq aligner. Bioinformatics. 2013;29(1):15–21.
4. Fan Y, et al. MuSE: accounting for tumor heterogeneity using a sample-specific error model improves sensitivity and specificity in mutation calling from sequencing data. Genome Biol. 2016;17(1):178.
5. Favero F, et al. Sequenza: allele-specific copy number and mutation profiles from tumor sequencing data. *Annals of oncology : official journal of the European Society for*. Med Oncol. 2015;26(1):64–70.
6. Huber W, et al. Orchestrating high-throughput genomic analysis with Bioconductor. Nat Methods. 2015;12(2):115–21.
7. Kim D, et al. TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. Genome Biol. 2013;14(4):R36.
8. Koboldt DC, et al. VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. Genome Res. 2012;22(3):568–76.
9. Kwon SM, et al. Perspectives of integrative cancer genomics in next generation sequencing era. Genomics Inform. 2012;10(2):69–73.
10. Langmead B, Salzberg SL. Fast gapped-read alignment with bowtie 2. Nat Methods. 2012;9(4):357–9.
11. Larson DE, et al. SomaticSniper: identification of somatic point mutations in whole genome sequencing data. Bioinformatics. 2012;28(3):311–7.
12. Li H, Durbin R. Fast and accurate short read alignment with burrows-wheeler transform. Bioinformatics. 2009;25(14):1754–60.
13. Li H, et al. The sequence alignment/map format and SAMtools. Bioinformatics. 2009;25(16):2078–9.
14. Martin M. Cutadapt removes adapter sequences from high-throughput sequencing reads. 2011 2011;17(1):3.
15. McKenna A, et al. The genome analysis toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. Genome Res. 2010;20(9):1297–303.
16. McLaren W, et al. The Ensembl variant effect predictor. Genome Biol. 2016;17(1):122.
17. Schorderet P. NEAT: a framework for building fully automated NGS pipelines and analyses. BMC Bioinformatics. 2016;17:53.

18. Singer J, et al. NGS-pipe: a flexible, easily extendable and highly configurable framework for NGS analysis. Bioinformatics. 2018;34(1):107–8.
19. Trapnell C, et al. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. Nat Biotechnol. 2010;28(5):511–5.
20. Wang K, Li M, Hakonarson H. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. Nucleic Acids Res. 2010; 38(16):e164.