# **Cirrus-NGS: Cloud-optimized next generation sequencing primary analysis pipeline**



Guorong Xu, Mustafa Guler, Mengyi Liu, Roman Sasik, Amanda Birmingham, Kathleen M. Fisch<sup>\*</sup> Center for Computational Biology & Bioinformatics, Department of Medicine University of California, San Diego, La Jolla, CA Contact: g1xu@ucsd.edu



CENTER FOR COMPUTATIONA **BIOLOGY &** BIOINFORMATICS

#### **Motivation**

ot Bioinformatics large-scale next-generation analysis sequencing (NGS) data requires significant compute resources. While cloud computing makes such processing power available on-demand, the administration of dynamic compute clusters is a daunting task for most working biologists. To address this pain

# Configurations

## Configuration file for tools in	#!/bin/bash	project: 20180601 Rana Li RNAseg
" configuration file for coold in	software dir=/shared/workspace/software	pipeline: RNASeq
# Shared steps #	reference dir=\$software dir/references	workflow: star rsem
fastac	tool dir=software dir	analysis:
script path: "fastac"		- fastoc
download suffix: ~	#22222222222222222222222222222222222222	- align count
input is output: False	# REFERENCES #	- merge counts
can be zinned: True	#2222222222222222222222222222222222222	- multiac
uses chromosomes: False	***************************************	- trim
trim.	#human ho19	date: 2018/06/12
script path: "trim"	evnort hol9 fasta="\$reference dir/Hsaniens/hol9/sequence/ussc.hol9.fasta"	upload: s3://ucsd-ccb-data-upload-1/20180601 Rana Li
download suffix: ~	export half chromosome list="chrM chr2 chr3 chr3 chr6 chr7 chr8 chr9 chr10 ch	download: s3://ucsd-ccb-data-upload-1/20180601 Rana l
input is output: False	export halo fai-"streference dir/Heaniers/halo/cequence/ussc halo fasta fai"	genome: mm10
can be zinned: True	export halo atf_sterence dir/Hapiens/halo/annotations/annotation atf	style: NA
uses chromosomes: False	export halo by index="treference dir/Hsapiens/halo/indices/bya/ussc halo fasta"	pairs
uses_enromosomes. Tacse	export half boutie index-"treference dir/Hsapiens/hg19/Indices/boutie/generg"	0
# MultiOC: aggregates results #	export half_bkallista index_"stereference_di/hadplensions/half/indices/kallista/kallista ind	sample:
wilting:	export halo STAP index—"traference dic_ut/hsapiens/hgip/indices/STAP/"	- filename: NL m -1 59 1004 R1 001.fasto.oz
script path, "multiac"	export halfs_star_index_steretence_dif/isapiens/hg19/indices/star/	description: NL m -1 S9 L004 R1 001
download suffix: ""	export half decon_liference dir (Heaping) (half decon a start and the st	aroup: aroupA
input is output. True	export hg19_doshp= %reference_dir/hsapiens/hg19/variation/doshp_156.hg19.vc1	= filename: NL m $= 2$ S7 L004 R1 001 fasta oz
cap be zipped. False	export halp_hitts= #fereference_uir/hsapiens/hgi9/variation/mitts_and_needeggotd_standard	description: NL m -2 S7 L004 R1 001
can_be_zipped: False	export hg19_nephidp= steletence_ui/hsdpiens/hg19/variation/hgmdp_3.5.hg19.sites.vcf	aroun: arounA
all complex. True	export hg19_complex steletence_uit/nsapiens/hg19/variation/luogo_complex.s.ng19.sites.vci	= filename: NL m $=$ S8 L004 R1 001 fasta oz
att_samples: True	export hg19_shp5_1000G="\$reference_oir/hsghpens/ng19/variation/1000G_phasei.snps.nlgn_	description: NL m _3 S8 L004 R1 001
t MCC stops #	export bg19_indets_ltroference_dir/Hadpiens/hg19/variation/10006_phasei.indets.ng	$\frac{1}{10} = \frac{1}{1000} = \frac{1}{$
wus steps #	export ngra_cosmic= \$rererence_urr/hsapiens/ngra/varracion/ngra_cosmic.vcr	= filename: NL m $=7$ S10 L004 R1 001 facto oz
Jwa:	thuman bainnin	description: NL m =7 S10 L004 R1 001
script_path: "DNASeq/bwa"	#numan nairpin	
download_suffix: ".trim{}"	export hairpin_numan_tasta="\$reference_dir/hsapiens/hairpin/sequence/hairpin_numan.ta"	filename: NL m _Toput_13 S1 L004 P1 001 fasta az
input_is_output: True	export hairpin_numan_bowtle2_indexes="specific human"	description: NL m _Trput=13_S1_L004_P1_001
can_be_zipped: False	export narrpin_numan_bowtrez_index="narrpin_numan"	arount around
uses_chromosomes: raise	Henry heimin	filename: NL m _Tnput_15 S2 L004 P1 001 facto oz
SOFT:	#mouse nairpin	description: NL m_Trout_15_S2_L004_R1_001.Tastq.g2
script_path: "DNASeq/sort"	export nairpin_mouse_tasta="\$reference_di/ymmuscutus/nairpin/sequence/nairpin_muscutus/	.Ta description. NL_m_=input=15_52_t004_K1_001
download_suffix: ".Dam"	export nairpin_mouse_bowtle2_indexes="\$reference_dir/mmusculus/nairpin/indices/bowtle2;	- filename: NL m - Input-22 S2 L004 P1 001 facto oz
input_is_output: True	export nairpin_mouse_bowtle2_index="nairpin_musculus"	description: NL m_Trput=22_55_L004_R1_001.Tastq.g2
can_be_zipped: False	Harris (200-20	
uses_chromosomes: False	#mouse GRCm38	filonomo: NL m Toput 22 54 1004 B1 001 fosta az
Jeaup:	export GRCm38_rasta="\$reterence_dir/Mmusculus/GRCm38/sequence/GRCm38_68.fa"	description: NL m _Tnput=23_54_L004_R1_001.1dStQ.92
script_path: "UNASeq/dedup"	export GRCm38_cnromosome_List="1 10 11 12 13 14 15 16 1/ 18 19 2 3 4 5 6 / 8 9 MI X Y"	acoupt acoupA
download_suffix: ".sort.bam"	export GRCm38_ral="sreference_dir/Mmusculus/GRCm38/sequence/GRCm38_68.fa.fa"	group: groupA filename: NL m _Toput_27 SE L004 P1 001 facts as
input_is_output: True	export GRCm38_gtt="\$reference_dir/Mmusculus/GRCm38/annotations/Mus_musculus.GRCm38.68.6	description: NL_MINPUL=27_55_L004_R1_001.TaStq.gZ
can be zipped: False	export GRLm38 bwa index="%reference dir/Mmusculus/GRCm38/indices/bwa/GRCm38 68.fa"	description: NL_MINput-27_33_L004_K1_001

#### Performance

Data Type	Num of Reads (M)	Running Time (H)	Instance Type	Running Cost (\$)
WGS	680	30	r3.8xlarge	15
WES	43	5	r3.8xlarge	0.3

point, we have developed cirrus-ngs, a turn-key solution for common NGS analyses using Amazon Web Services (AWS).

## **Pipelines**

Cirrus-ngs currently supports RNA-Seq, miRNA-Seq, ChIP-Seq and WGS/WES data with multiple version of genomes.

#### **Features**:

WGS/WES pipelines include bwa gatk and bwa mutect workflows for germline and somatic variants calling. The analysis steps include fastqc, trim, align, multiqc, sort, dedup, split, postalignment, haplotype, somatic\_variant\_calling, merge, combine\_vcf.

RNA-seq pipelines include star\_rsem, star\_htseq, kallisto and star\_gatk workflows. The analysis steps include fastqc, trim, align\_count, multiqc, merge\_counts and variant\_calling.

ChiP-seq pipeline includes Homer workflow. The analysis steps include fastqc, trim, align, multiqc, make\_tag\_directory, make\_UCSC\_file, find\_peaks, annotate\_peaks, pos2bed, find\_motifs\_genome.

miRNA-seq pipeline includes bowtie2 workflow. The analysis steps include fastqc, trim, cut\_adapt, align\_and\_count, multiqc.

#### Notebooks

asicCFNClusterSetup Last Checkpoint: Last Thursday at 4:43 PM (autosaved)	P Logout	Ċ jupyter	r WES	WGS Pi	ipelin	e Last Checkpo	int: a few seconds	ago (autosaved)				n Cogo
w Insert Cell Kernel Widgets Help Tru	sted Python 3 C	File Edit	View	Insert	Cell	Kernel W	dgets Help				Trusted	Python 3
		8 + %	46	↑ ↓	M I	C Code	÷					
<pre>sport os sport sys s.path.append("/./src/cirrus_ngs") Input the AWS account access keys s_access_key_id = "AKIXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX</pre>		In [1]:	<pre>import import sys.ps from c from c from c from c #s3 ac s3_ing s3_out ## CFI your_c ## The privat</pre>	: OS : Sys tth.appen. :fnCluste :fnCluste :fnCluste out_files put_file :Cluster_n :luster_n :private :e_private	d(" rr imp ort Pi ort De addr s_addr s_add name ame = e key "/pat	//src/cirn ort CFNClust pelineManage signFileLoad t files and ess = "s3:/ ress = "s3:/ "clusternar pair for acc h/to/your_at	rus_ngs") eerManager, Co rr output files /path/to/fastq //path/to/outp ne" eessing cluste rs_key.pem"	nnectionManager "" ut"				
Install CFNCluster			## Pro projec #bwa_c	<pre>pject inf ct_name = gatk or b low = "bw</pre>	Format tes wa_mu	ion t_status" tect ect"						
tice: The CFNCluster package can be only installed on Linux box which supports pip installation.			#hg19	or GRCm3	18							
s.path.append("/./src/cirrus_ngs") m cfnCluster import CFNClusterManager KClusterManager.install_cfn_cluster()			genome ## Pos ##bwa_	e = "GRCm ssible an _gatk: "f	38" alysi astqc	s_steps inpu ", "trim",	its for the tw "align", "mult	o workflows. Or	der of input doe edup", "split",	s not matter. "postalignment",		
Upgrade CFNCluster			#"hap] ##bwa_	<pre>lotype", _mutect:</pre>	"merg "fast	e", "combine qc", "trim"	e_vcf", "filte , "align", "m	er" multigc", "sort"	, "dedup", "spli	t", "postalignment",		
m cfnCluster import CFNClusterManager ClusterManager.upgrade cfn cluster()			#"soma	atic_vari sis steps	ant_c	alling", "me	erge"					
Configure CFNCluster configure CFNCluster settings, you need to import the package CFNCluster. The below functions te ert AWS access keys, configure instance types, spot price and S3 resource.	ll you how to					"fastqc" ,"trim" ,"align" ,"multiqc" ,"sort" ,"dedup" ,"split" ,"postalign "bapletype	ument"					
m cfnCluster import CFNClusterManager Configure cfncluster settings aws access keys are the AWS credentials section (required). These settings apply to all clusters.						<pre>, "mutect" , "merge" , "merge_vcd , "combine , "filter"</pre>	_pairwise" _vcf"					
<pre>KClusterManager.insert_access_keys(aws_access_key_id=aws_access_key_id, aws_secret_access_key=aws_secret "private_key" is to specify the name of an existing EC2 KeyPair to enable SSH access to the instances KClusterManager.config_key_name(private_key) "master_instance_type" is to specify the EC2 instance type use for the master node. "compute_instance_type" is to specify the EC2 instance type used for the cluster compute nodes.</pre>	et_access_key)		#add of #examp design	design fi ples in c n_file =	} ile pa irrus "/hom	th here _root/data/co e/mustafa/co t")	<i>cirrus-ngs/</i> :bb/cirrus-ngs	/data/cirrus-ng	s/mouse_test.txt			
			variat	oles set								
Α	nal	ys	is	5								

RNA-seq	35	2	r3.8xlarge	0.12
ChiP-seq	30	1.3	r3.8xlarge	0.08
miRNA-seq	15	0.5	r3.4xlarge	0.04

Note: the numbers are calculated per sample

# Conclusion

Cirrus-ngs is a lightweight, reproducible tool to perform scalable NGS primary analyses on the cloud. Cirrus-ngs has been optimized for use on AWS HPC clusters, which dynamically scale depending on the compute resources required for the various steps in the pipelines to minimize the per sample compute cost. All computation is performed using AWS EC2 compute instances and all results are uploaded to AWS S3 storage.

#### **Availability**

Cirrus-ngs is developed in Python and is available for free use and extension under the MIT License. Source code and extensive documentation are on GitHub at https://github.com/ucsd-ccbb/cirrus-ngs

#### **Reliability:**

To avoid data loss, the intermediate results of each step are saved to S3 and users can rerun any step of the pipeline if one step is failed because of spot instances loss or unknown reasons.

#### **Extensibility**:

Each workflow is designed independently but shares the common steps. The developers/users can easily modify or integrate their own codes to Cirrus-ngs by following the design and configuration of framework.

# **Schematic**





Copy table Configure Columns Plot Showing 6/6 rows and 8/10 column



 
 1906780
 1659502
 1941302
 2094120
 2166610
 1866043

 18919498
 17428562
 17889833
 19438339
 17881171
 18478954
 6228177 6428896 6892358 6396395 NSMUSG0000000131 ENSMUST00000009344 NSMUSG0000000142\_ENSMUST000005291 NSMUSG0000000148 ENSMUST0000004158 3416 1336.08 NSMUSG0000000154 ENSMUST00000052348



#### **Funding/Disclosures**

Supported by the UC San Diego Clinical and Translational Research Institute Grant UL1TR001442.

## References

- [1] 1000 Genomes Project Consortium. 2015. A global reference for human genetic variation. Nature 526:68-74. doi: 10.1038/nature15393 [2] cfncluster (https://github.com/awslabs/cfncluster) [3] H. Li, R. Durbin Fast and accurate short read alignment with Burrows-Wheeler transform
- **Bioinformatics**, 25 (2009), pp. 1754-1760 [4] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G.Abecasis, R. Durbin,
- 1000 Genome Project Data Processing Subgroup The Sequence Alignment/Map format and SAMtools. **Bioinformatics**, 25 (2009), pp. 2078-2079
- [5] A. McKenna, M. Hanna, E. Banks, A. Sivachenko, K. Cibulskis, A. Kernytsky, K.Garimella, D. Altshuler, S. Gabriel, M. Daly, M.A. DePristoThe Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data Genome Res., 20 (2010), pp. 1297-1303
- [6] D. Koboldt, D. Larson, R. Wilson. Using VarScan 2 for Germline Variant Calling and Somatic Mutation Detection. Curr Protoc **Bioinformatics**. 2013;44:15.14.11–17. doi: 10.1002/0471250953.bi1504s44.