応用編コマンド実習

がんの全ゲノム解析

令和5年度

東京大学　医科学研究所附属病院　血液腫瘍内科　助教　**横山和明**

東京大学医科学研究所　ヒトゲノム解析センター　健康医療インテリジェンス分野　**清水英悟**

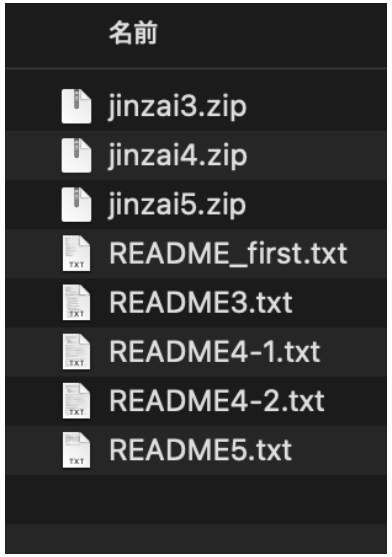
国立がん研究センター　先端医療開発センタートランスレーショナルインフォマティクス分野　ユニット長　**山下理宇**

目次

解析実習の準備 1

1. コマンド実習のサイトにログインしていただき、コマンド実習関連のファイルをダウンロードしてください。

ファイルのリストは下記になります。



2. ダウンロードの後、README_first.txt をダブルクリックして開いてください。

実習中は、テキストファイルに記述してあるコマンドをコピーして、Terminal(Macの場合）または、Windows PowerShell(Windowsの場合）、TeratermなどのSSHソフト(Windowsの場合）にペーストして、コマンドを実行することになります。

コピー、ペーストが適切に実行できるように確認してください。

Macの場合は、Cmd+C(copy), Cmd+V(paste)

Windowsの場合は、Ctrl+C(copy), Ctrl+V または terminal上で右クリック(paste)

解析実習の準備 2

3. SHIROKANEへのログイン

SHIROKANEにログインするには、README_first.txtにある、2.1 login に記述されているコマンドを実行します。

```
# 2.1. login
ssh lectXX@slogin.hgc.jp
```

実際に Mac Terminal で実行した結果(lect90がユーザー名の場合）が下記になります。受け取ったユーザーIDとパスワードでログインしてください。ログイン先は ログインノード(login node)になります。コマンド実習のスライド全体を通して、lect90のユーザーで実行しますのでご了承願います。

Windows PowerShell をご使用の場合は、同様にコマンドをペーストしていただければログインできます。

その他の SSH terminal をご使用の場合は、それぞれでログイン方法が違いますので、ご確認ください。

```
$ > ssh lect90@slogin.hgc.jp
Are you sure you want to continue connecting(yes/no(fingererprint))?  yes
lect90@slogin.hgc.jp's password:
Last failed login: Fri Jun 30 14:06:40 JST 2023 from a202175149004.at.hgc.jp on ssh:notty
There was 1 failed login attempt since the last successful login.
Last login: Fri Nov 26 14:56:15 2021 from 113x43x51x65.ap113.ftth.ucom.ne.jp

[Begin first by QLOGIN!]
Welcome to SHIROKANE supercomputer of Human Genome Center, IMS, UT.
This is a login node as the gateway with very little software.
You can type  qlogin  COMMAND to use all the features of SHIROKANE.
   |       o             ,,
,--.| ,--.,--.. ,--. |  --, / +-   -+-   .
|  || |  ||  || |  ||  //  |-- | `  |  |
`--|`-`--'`--|`-`  `  `-- `-`-'`- `-  `-
   `     `--' https://supcom.hgc.jp/internal/mediawiki/id/357

[A login node of SHIROKANE lect90@slogin2:~]$
```

解析実習の準備 3

4. 計算ノードへの qlogin

ssh でSHIROKANEにログインした後、実際にアラインメントや変異コールなどの処理を行うためには、計算ノードに qlogin しなければなりません。
README_first.txt の下記に計算ノードに qlogin するコマンドが記載されています。SHIROKANEスパコンにメモリ、CPUのリソースがない場合などに、ログインに失敗する場合はあります。その場合は再度 qlogin を実行してください。何度も失敗する場合は、指定メモリを減らして、再度 qlogin してみてください。

```
# 2.2. qlogin
# スパコンのリソースがない場合は、40Gでログインできません。
#  3章のjavaのプログラムは、30Gのメモリを使用するため、40Gを確保しますが。
# 3章のbwa、samtools及び、4章以降は、20Gで実行できます。
qlogin -l s_vmem=40G
```

実際にコマンドを実行して計算ノードに qlogin した結果

```
[A login node of SHIROKANE lect90@slogin2:~]$ qlogin -l s_vmem=40G
要求ハードリソース
  memory (s_vmem): 40G = ジョブは 1 スロットあたり 40G バイトのメモリを要求します
  slots (def_slot): 1 = ジョブは 100% の CPU を要求します
  total memory: 40G = ジョブは 40G バイトのメモリを要求します
通常の qlogin を実行します。
Your job 77275599 ("QLOGIN") has been submitted
waiting for interactive job to be scheduled ...
Your interactive job 77275599 has been successfully scheduled.
Establishing /home/geadmin/N1GE/utilbin/qlogin_wrapper session to host gc013i ...
Last login: Fri Jun 30 14:14:31 2023 from gc002i
[OS 7] You are now on OS 7 compute node.
==== あなたのグループ lect のリソース利用状況 ====  hauq command version 1.13
* Home Disk use> 3 TB / 96 TB (3.4 %)   651 kfiles / 96000 kfiles (0.7 %)
* Arch Disk use> 0.0 TB / 0.0 TB (0.0 %)   [ 0.0 TB(cache) + 0.0 TB(tape) ]
* UGE queue use> mjobs.q: 1/4096 (0 %)  ljobs.q: 0/768 (0 %)  lmem.q: 2/128 (2 %)  intr.q: 11/96 (11 %)
[lect90@gc013 ~]$
```

解析実習の準備 4

5. 解析環境の設定

コマンドを複数行ペーストして実行するために下記のコマンドを実行します。

```
＃2.3. environment set
export PROMPT_COMMAND=
```

JAVAプログラムを実行するために下記のコマンドを実行します。JAVA heap memory のサイズを 16Gに設定しています。

JAVAのコマンドによっては、メモリが足りない場合がありますので、適時メモリサイズを増やしてこのコマンドで再設定してください。

qlogin で指定しているメモリを超えて JAVA heap memory のサイズを指定すると失敗しますので、qlogin で指定したメモリサイズを考慮して JAVA heap memory を指定してください。

```
export JAVA_TOOL_OPTIONS='-XX:+UseSerialGC -Xmx16g'
```

fastqc、bwa、samtools を実行するために下記のコマンドを実行します。

```
module use /usr/local/package/modulefiles/;
module load fastqc/0.11.8;
module load bwa/0.7.17;
module load samtools/1.9;
```

解析実習の準備 5

コマンド実行後、ツールが使用できるか確認してください。

```
[lect90@gc013 ~]$ export PROMPT_COMMAND=
[lect90@gc013 ~]$ module use /usr/local/package/modulefiles/;
[lect90@gc013 ~]$ module load fastqc/0.11.8;
[lect90@gc013 ~]$ module load bwa/0.7.17;
[lect90@gc013 ~]$ module load samtools/1.9;
[lect90@gc013 ~]$
[lect90@gc013 ~]$ fastqc --help

           FastQC - A high throughput sequence QC analysis tool

SYNOPSIS

                fastqc seqfile1 seqfile2 .. seqfileN

    fastqc [-o output dir] [--(no)extract] [-f fastq|bam|sam]
           [-c contaminant file] seqfile1 .. seqfileN
…
[lect90@gc013 ~]$ bwa

Program: bwa (alignment via Burrows-Wheeler transformation)
Version: 0.7.17-r1188
Contact: Heng Li <lh3@sanger.ac.uk>

Usage:   bwa <command> [options]
…
[lect90@gc013 ~]$ samtools

Program: samtools (Tools for alignments in the SAM format)
Version: 1.9 (using htslib 1.9)

Usage:   samtools <command> [options]
…
```

解析実習の準備 6

6. 実習データをコピーする

実習用のデータを SHIROKANE の /share/lect/202210-11 からご自分のホームディレクトリにコピーします。

下記3行のcp コマンドをコピーして実行してください。コピーには少々時間がかかります(20～30分程度)。

```
3. 実習データのコピー
実習で使用しますので、スパコン上でコピーをお願いします。

cp -r /share/lect/202210-11/jinzai3 ~/;
cp -r /share/lect/202210-11/jinzai4 ~/;
cp -r /share/lect/202210-11/jinzai5 ~/;
```

実行結果。データがコピーされたか確認してください。

```
[lect90@gc013 ~]$  cp -r /share/lect/202210-11/jinzai3 ~/;
[lect90@gc013 ~]$  cp -r /share/lect/202210-11/jinzai4 ~/;
[lect90@gc013 ~]$  cp -r /share/lect/202210-11/jinzai5 ~/;
[lect90@gc013 ~]$  ls –l
合計 12
drwxr-x--- 4 lect90 lect 4096  6月 30 14:39 jinzai3
drwxr-x--- 5 lect90 lect 4096  6月 30 14:36 jinzai4
drwxr-x--- 3 lect90 lect 4096  6月 30 14:34 jinzai5
[lect90@gc013 ~]$
```

fastq ファイルのアラインメント1

1. ご自分のPCにダウンロードした README3.txt をダブルクリックして開いてください。

応用編第Ⅲ章に記載されているコマンドが抜粋されています。

コマンドをコピーし、SHIROKANE のterminal にペーストし実行してください。

2. コマンドの実行

下記の２つのコマンドは reference fasta の index ファイルを作成するコマンドです。

bwa index

samtools faidx

この２つのコマンドはすでに実行されているので、実行しなくても構いません。

fastqc は、シークエンサーから出力される fastq ファイルのクオリティチェック（QC）を行うコマンドです。

bwa mem は、fastq ファイルを reference fasta にアラインメントするコマンドです。

samtools sort は、bwa mem の結果を座標でソートして、bam ファイルに変換するコマンドです。

gatk は、duplicate reads にフラグをたて、base recalibration を行い、bam ファイルに情報を追加し、補正します。

詳しくは、応用編第Ⅲ章をご覧ください。

fastq ファイルのアラインメント2

bwa mem コマンドの実行結果です。

```
[lect90@gc013 jinzai3]$ bwa mem ¥
>       -t 20 ¥
>       -R '@RG¥tID:COLO829BL¥tLB:lib1¥tPL:illumina¥tSM:COLO829BL¥tPU:COLO829BL' ¥
>       ~/jinzai3/data/ref/Homo_sapiens_assembly38.fasta ¥
>       ~/jinzai3/data/fastq/COLO829BL/COLO829BL.BRAF_R1.fastq ¥
>       ~/jinzai3/data/fastq/COLO829BL/COLO829BL.BRAF_R2.fastq ¥
>       > ~/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38.sam
[M::bwa_idx_load_from_disk] read 0 ALT contigs
[M::process] read 50000 sequences (7500000 bp)...
[M::mem_pestat] # candidate unique pairs for (FF, FR, RF, RR): (0, 25000, 0, 0)
[M::mem_pestat] skip orientation FF as there are not enough pairs
[M::mem_pestat] analyzing insert size distribution for orientation FR...
[M::mem_pestat] (25, 50, 75) percentile: (465, 499, 532)
[M::mem_pestat] low and high boundaries for computing mean and std.dev: (331, 666)
[M::mem_pestat] mean and std.dev: (498.45, 49.65)
[M::mem_pestat] low and high boundaries for proper pairs: (264, 733)
[M::mem_pestat] skip orientation RF as there are not enough pairs
[M::mem_pestat] skip orientation RR as there are not enough pairs
[M::mem_process_seqs] Processed 50000 reads in 2.760 CPU sec, 0.149 real sec
[main] Version: 0.7.17-r1188
[main] CMD: bwa mem -t 20 -R @RG¥tID:COLO829BL¥tLB:lib1¥tPL:illumina¥tSM:COLO829BL¥tPU:COLO829BL
/home/lect90/jinzai3/data/ref/Homo_sapiens_assembly38.fasta
/home/lect90/jinzai3/data/fastq/COLO829BL/COLO829BL.BRAF_R1.fastq
/home/lect90/jinzai3/data/fastq/COLO829BL/COLO829BL.BRAF_R2.fastq
[main] Real time: 3.407 sec; CPU: 6.004 sec
```

※ Terminalの言語によっては、backslash は、円マーク ‘¥’ になります。’¥’マークは、改行せずコマンドを１行で実行するという指示になります。

fastq ファイルのアラインメント3

samtools sort コマンドの実行結果です。

```
[lect90@gc013 jinzai3]$ samtools sort -@ 4 ¥
>           -o ~/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38.bam ¥
>           ~/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38.sam
[bam_sort_core] merging from 0 files and 4 in-memory blocks...
```

gatk MarkDuplicates の実行結果です。

```
[lect90@gc013 jinzai3]$ ~/jinzai3/bin/gatk-4.2.3.0/gatk ¥
>       MarkDuplicates ¥
>       --java-options -Xmx12g ¥
>       -I ~/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38.bam ¥
>       -O ~/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38_markdup.bam ¥
>       -M ~/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38_matricx.txt
Using GATK jar /rshare1/ZETTAI_path_WA_slash_home_KARA/home/lect90/jinzai3/bin/gatk-4.2.3.0/gatk-package-4.2.3.0-local.jar
Running:
    java -Dsamjdk.use_async_io_read_samtools=false -Dsamjdk.use_async_io_write_samtools=true -Dsamjdk.use_async_io_write_tribble=false -Dsamjdk.compression_level=2 -Xmx12g
-jar /rshare1/ZETTAI_path_WA_slash_home_KARA/home/lect90/jinzai3/bin/gatk-4.2.3.0/gatk-package-4.2.3.0-local.jar MarkDuplicates -I
/home/lect90/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38.bam -O /home/lect90/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38_markdup.bam -M
/home/lect90/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38_matricx.txt
Picked up JAVA_TOOL_OPTIONS: -XX:+UseSerialGC -Xmx64m -Xms32m
14:59:27.111 INFO  NativeLibraryLoader - Loading libgkl_compression.so from jar:file:/rshare1/ZETTAI_path_WA_slash_home_KARA/home/lect90/jinzai3/bin/gatk-4.2.3.0/gatk-package-
4.2.3.0-local.jar!/com/intel/gkl/native/libgkl_compression.so
…
[Fri Jun 30 14:59:31 JST 2023] picard.sam.markduplicates.MarkDuplicates done. Elapsed time: 0.07 minutes.
Runtime.totalMemory()=6797881344
Tool returned:
0
```

※ Terminalの言語によっては、backslash は、円マーク ‘¥’ になります。

※ SHIROKANE スーパーコンピュータ上では、/home/lect90 は、下記の様なパスとして表示されます。
/rshare1/ZETTAI_path_WA_slash_home_KARA/home/lect90

fastq ファイルのアラインメント4

その他、BaseRecalibrator、ApplyBQSR、samtools index を実行します。

Tumor, Blood のサンプルに対してそれぞれ、アラインメントの全てのコマンドを実行した結果のファイルは下記４ファイルになります。

```
[lect90@gc013 bam]$ cd ~/jinzai3/data/bam
[lect90@gc013 bam]$ pwd
/home/lect90/jinzai3/data/bam

[lect90@gc013 bam]$ ls –l *.bam *.bai
-rw-r----- 1 lect90 lect 1489963  6月 30 14:42 COLO829/COLO829.BRAF.GRCh38_markdup_updated.bam
-rw-r----- 1 lect90 lect 1283985  6月 30 14:59 COLO829BL/COLO829BL.BRAF.GRCh38_markdup_updated.bam
-rw-r----- 1 lect90 lect 95736  6月 30 14:42 COLO829/COLO829.BRAF.GRCh38_markdup_updated.bam.bai
-rw-r----- 1 lect90 lect 95736  6月 30 14:42 COLO829BL/COLO829BL.BRAF.GRCh38_markdup_updated.bam.bai
```

変異コール

変異コールは、Mutect2 を使用します。README3.txt に記載されているコマンドをコピーして、terminal にペーストしてください。

下記がコマンドを実行した結果です。詳しくは、応用編第Ⅲ章をご覧ください。

```
[lect90@gc013 ~]$ mkdir -p ~/jinzai3/data/vcf/COLO829
[lect90@gc013 ~]$ ~/jinzai3/bin/gatk-4.2.3.0/gatk Mutect2 ¥
>     -R ~/jinzai3/data/ref/Homo_sapiens_assembly38.fasta ¥
>     -I ~/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38_markdup_updated.bam ¥
>     -I ~/jinzai3/data/bam/COLO829/COLO829.BRAF.GRCh38_markdup_updated.bam ¥
>     --normal-sample COLO829BL ¥
>     -O ~/jinzai3/data/vcf/COLO829/result.vcf
Using GATK jar /rshare1/ZETTAI_path_WA_slash_home_KARA/home/lect90/jinzai3/bin/gatk-4.2.3.0/gatk-package-4.2.3.0-local.jar
Running:
    java -Dsamjdk.use_async_io_read_samtools=false -Dsamjdk.use_async_io_write_samtools=true -Dsamjdk.use_async_io_write_tribble=false -Dsamjdk.compression_level=2 -jar
/rshare1/ZETTAI_path_WA_slash_home_KARA/home/lect90/jinzai3/bin/gatk-4.2.3.0/gatk-package-4.2.3.0-local.jar Mutect2 -R
/home/lect90/jinzai3/data/ref/Homo_sapiens_assembly38.fasta -I /home/lect90/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38_markdup_updated.bam -I
/home/lect90/jinzai3/data/bam/COLO829/COLO829.BRAF.GRCh38_markdup_updated.bam --normal-sample COLO829BL -O /home/lect90/jinzai3/data/vcf/COLO829/result.vcf
Picked up JAVA_TOOL_OPTIONS: -XX:+UseSerialGC -Xmx64m -Xms32m
15:16:33.859 INFO  NativeLibraryLoader - Loading libgkl_compression.so from jar:file:/rshare1/ZETTAI_path_WA_slash_home_KARA/home/lect90/jinzai3/bin/gatk-4.2.3.0/gatk-
package-4.2.3.0-local.jar!/com/intel/gkl/native/libgkl_compression.so
Jun 30, 2023 3:16:34 PM shaded.cloud_nio.com.google.auth.oauth2.ComputeEngineCredentials runningOnComputeEngine
INFO: Failed to detect whether we are running on Google Compute Engine.
15:16:34.095 INFO  Mutect2 - ------------------------------------------------------------
15:16:34.095 INFO  Mutect2 - The Genome Analysis Toolkit (GATK) v4.2.3.0
15:16:34.095 INFO  Mutect2 - For support and documentation go to https://software.broadinstitute.org/gatk/
15:16:34.096 INFO  Mutect2 - Executing as lect90@gc013i on Linux v3.10.0-1127.18.2.el7.x86_64 amd64
15:16:34.096 INFO  Mutect2 - Java runtime: Java HotSpot(TM) 64-Bit Server VM v1.8.0_181-b13
15:16:34.096 INFO  Mutect2 - Start Date/Time: 2023/06/30 15:16:33 JST
15:16:34.096 INFO  Mutect2 - ------------------------------------------------------------
15:16:34.097 INFO  Mutect2 - HTSJDK Version: 2.24.1
15:16:34.097 INFO  Mutect2 - Picard Version: 2.25.4
15:16:34.097 INFO  Mutect2 - Built for Spark Version: 2.4.5
15:16:34.097 INFO  Mutect2 - HTSJDK Defaults.COMPRESSION_LEVEL : 2
```

※ Terminalの言語によっては、backslash は、円マーク ‘¥’ になります。

※ SHIROKANE スーパーコンピュータ上では、/home/lect90 は、下記の様なパスとして表示されます。
/rshare1/ZETTAI_path_WA_slash_home_KARA/home/lect90

構造変異同定、コピー数解析

構造変異同定、コピー数解析はそれぞれ、manta、cnvkit を使用します。

コマンドは、README3.txt に記載してありますので、同様にコピー、ペーストで実行してください。

cnvkit に関しては、python でライブラリをインストールして、Rのパッケージもインストールする必要があります。

module use, module load で python/3.8 を使用できるように設定し、

pip install で cnvkit 関連の pythobn のライブラリをインストールします。

その後、cnvkit でのコピー数解析を実行します。

コマンドに関しての詳しい説明は、応用編第Ⅲ章をご覧ください。

構造変異同定 manta 実行結果

```
[lect90@gc016 ~]$ ~/jinzai3/bin/manta-1.6.0.centos6_x86_64/bin/configManta.py      --normalBam
~/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38_markdup_updated.bam      --tumorBam
~/jinzai3/data/bam/COLO829/COLO829.BRAF.GRCh38_markdup_updated.bam      --referenceFasta ~/jinzai3/data/ref/Homo_sapiens_assembly38.fasta      --
runDir ~/jinzai3/data/manta/COLO829

Successfully created workflow run script.
To execute the workflow, run the following script and set appropriate options:

/home/lect90/jinzai3/data/manta/COLO829/runWorkflow.py
[lect90@gc016 ~]$ ~/jinzai3/data/manta/COLO829/runWorkflow.py
[2023-07-03T02:50:47.401279Z] [gc016i] [34126_1] [WorkflowRunner] Initiating pyFlow run
[2023-07-03T02:50:47.413459Z] [gc016i] [34126_1] [WorkflowRunner] pyFlowClientWorkflowClass: MantaWorkflow
[2023-07-03T02:50:47.414082Z] [gc016i] [34126_1] [WorkflowRunner] pyFlowVersion: 1.1.20
[2023-07-03T02:50:47.414697Z] [gc016i] [34126_1] [WorkflowRunner] pythonVersion: 2.7.15.final.0
[2023-07-03T02:50:47.415379Z] [gc016i] [34126_1] [WorkflowRunner] WorkingDir: '/rshare1/ZETTAI_path_WA_slash_home_KARA/home/lect90'
[2023-07-03T02:50:47.415934Z] [gc016i] [34126_1] [WorkflowRunner] ProcessCmdLine: '/home/lect90/jinzai3/data/manta/COLO829/runWorkflow.py'
[2023-07-03T02:50:47.416499Z] [gc016i] [34126_1] [WorkflowRunner] [RunParameters] mode: local
[2023-07-03T02:50:47.417036Z] [gc016i] [34126_1] [WorkflowRunner] [RunParameters] nCores: 72
[2023-07-03T02:50:47.417569Z] [gc016i] [34126_1] [WorkflowRunner] [RunParameters] memMb: 191770
…
…
[2023-07-03T02:51:23.195295Z] [gc016i] [34126_1] [WorkflowRunner] Manta workflow successfully completed.
[2023-07-03T02:51:23.195295Z] [gc016i] [34126_1] [WorkflowRunner]
[2023-07-03T02:51:23.195295Z] [gc016i] [34126_1] [WorkflowRunner]                workflow version: 1.6.0
[2023-07-03T02:51:23.196326Z] [gc016i] [34126_1] [WorkflowRunner]
[2023-07-03T02:51:23.197013Z] [gc016i] [34126_1] [WorkflowRunner] Workflow successfully completed all tasks
[2023-07-03T02:51:23.197727Z] [gc016i] [34126_1] [WorkflowRunner] Elapsed time for full workflow: 36 sec
[lect90@gc016 ~]$
```

コピー数解析、cnvkit 実行結果

```
[lect90@gc016 ~]$ ~/.local/bin/cnvkit.py batch ¥
>     ~/jinzai3/data/bam/COLO829/COLO829.BRAF.GRCh38_markdup_updated.bam ¥
>     --normal ~/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38_markdup_updated.bam ¥
>     -m wgs ¥
>     --fasta  ~/jinzai3/data/ref/Homo_sapiens_assembly38.fasta ¥
>     --output-reference ~/jinzai3/data/cnvkit/my_reference.cnn ¥
>     --output-dir  ~/jinzai3/data/cnvkit/COLO829 ¥
>     --diagram --scatter
CNVkit 0.9.10
WGS protocol: recommend '--annotate' option (e.g. refFlat.txt) to help locate genes in output files.
chr1: Scanning for accessible regions
                    Accessible region chr1:10000-207666 (size 197666)
                    Accessible region chr1:257666-297968 (size 40302)
                    Accessible region chr1:347968-535988 (size 188020)
                    Accessible region chr1:585988-2702781 (size 2116793)
…
Wrote Homo_sapiens_assembly38.bed with 239 regions
Indexing BAM file /home/lect90/jinzai3/data/bam/COLO829BL/COLO829BL.BRAF.GRCh38_markdup_updated.bam
Estimated read length 150.0
Limiting est. bin size 1059840 to given max. 50000
WGS average depth 0.05 --> using bin size 50000
Detected file format: bed
Splitting large targets
Wrote /home/lect90/jinzai3/data/cnvkit/COLO829/Homo_sapiens_assembly38.target.bed with 58496 regions
Wrote /home/lect90/jinzai3/data/cnvkit/COLO829/Homo_sapiens_assembly38.antitarget.bed with 0 regions
Building a copy number reference from normal samples...
Processing reads in COLO829BL.BRAF.GRCh38_markdup_updated.bam
Time: 0.344 seconds (115735 reads/sec, 169974 bins/sec)
Summary: #bins=58496, #reads=39830, mean=0.6809, min=0.0, max=39830.0
Percent reads in regions: 79.660 (of 50000 mapped)
Wrote /home/lect90/jinzai3/data/cnvkit/COLO829/COLO829BL.BRAF.GRCh38_markdup_updated.targetcoverage.cnn with 58496 regions
Skip processing COLO829BL.BRAF.GRCh38_markdup_updated.bam with empty regions file /home/lect90/jinzai3/data/cnvkit/COLO829/Homo_sapiens_assembly38.antitarget.bed
Wrote /home/lect90/jinzai3/data/cnvkit/COLO829/COLO829BL.BRAF.GRCh38_markdup_updated.antitargetcoverage.cnn with 0 regions
Relative log2 coverage of chrX=-0.00049, chrY=-0.00049 (maleness=0 x 300 = 0) --> assuming female
Loading /home/lect90/jinzai3/data/cnvkit/COLO829/COLO829BL.BRAF.GRCh38_markdup_updated.targetcoverage.cnn
Calculating GC and RepeatMasker content in /home/lect90/jinzai3/data/ref/Homo_sapiens_assembly38.fasta ...
Extracting sequences from chromosome chr1
Extracting sequences from chromosome chr2
….
Extracting sequences from chromosome chrY
WARNING: most bins have no or very low coverage; check that the right BED file was used
Loading /home/lect90/jinzai3/data/cnvkit/COLO829/COLO829BL.BRAF.GRCh38_markdup_updated.antitargetcoverage.cnn
Calculating average bin coverages
Calculating bin spreads
….
```

※ Terminalの言語によっては、backslashは、円マーク ‘¥’ になります。

FASTQCの結果、IGVでの変異の確認1

1. ご自分のPCにダウンロードした jinzai3.zip をダブルクリックして展開してください。

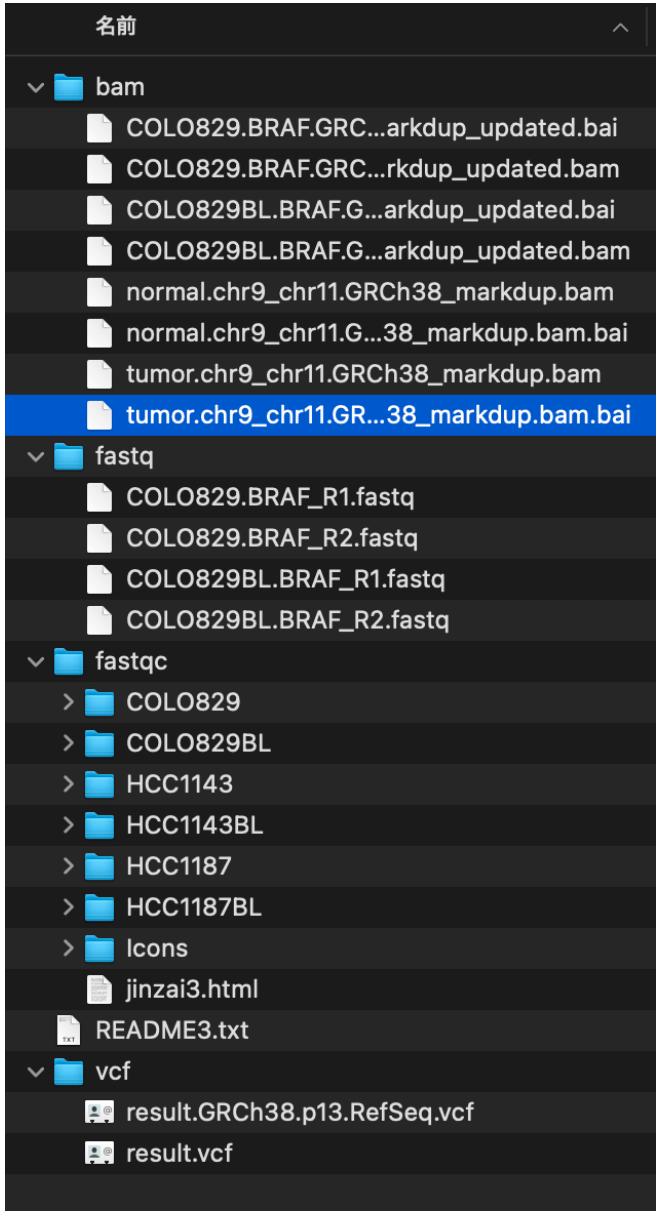
展開されたフォルダー jinzai3 には右記にようなファイルが含まれています。

2. fastqc フォルダーにある、jinzai3.html をダブルクリックすると、ブラウザーにFASTQCの結果が表示されます。

fastqc jinzai3

	Basic Statistics	Per base sequence quality	Per tile sequence quality	Per sequence quality scores	Per base sequence content	Per sequence GC content	Per base N content	Sequence Length Distribution	Sequence Duplication Levels	Overrepresented sequences	Adapter Content	Kmer Content
COLO829_R1_fastqc	✓	✓		✓	✓	!	✓	✓	!	✓	✓	✗
COLO829_R2_fastqc	✓	!		✓	✓	!	✓	✓	!	✓	✓	✗
COLO829BL_R1_fastqc	✓	✓		✓	✓	!	✓	✓	!	✓	✓	✗
COLO829BL_R2_fastqc	✓	!		✓	✓	!	✓	✓	!	✓	✓	✗
HCC1143_R1_fastqc	✓	✓		✓	✓	!	✓	✓	!	✓	✓	✗
HCC1143_R2_fastqc	✓	!		✓	✓	!	✓	✓	!	✓	✓	✗
HCC1143BL_R1_fastqc	✓	✓		✓	✓	!	✓	✓	✓	✓	✓	✗
HCC1143BL_R2_fastqc	✓	!		✓	✓	!	✓	✓	✓	✓	✓	✗
HCC1187_R1_fastqc	✓	✓		✓	✓	!	✓	✓	!	✓	✓	✗
HCC1187_R2_fastqc	✓	!		✓	✓	!	✓	✓	!	✓	✓	✗
HCC1187BL_R1_fastqc	✓	✓		✓	✓	!	✓	✓	✓	✓	✓	✗
HCC1187BL_R2_fastqc	✓	!		✓	✓	!	✓	✓	✓	✓	✓	✗

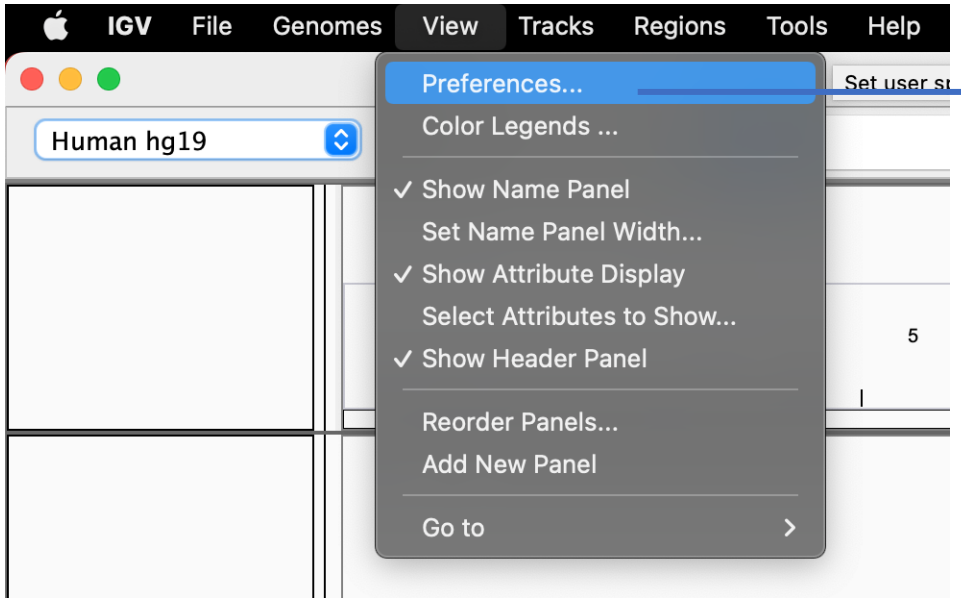
それぞれのサンプルをクリックすると、実際の FASTQCの結果が表示されます。

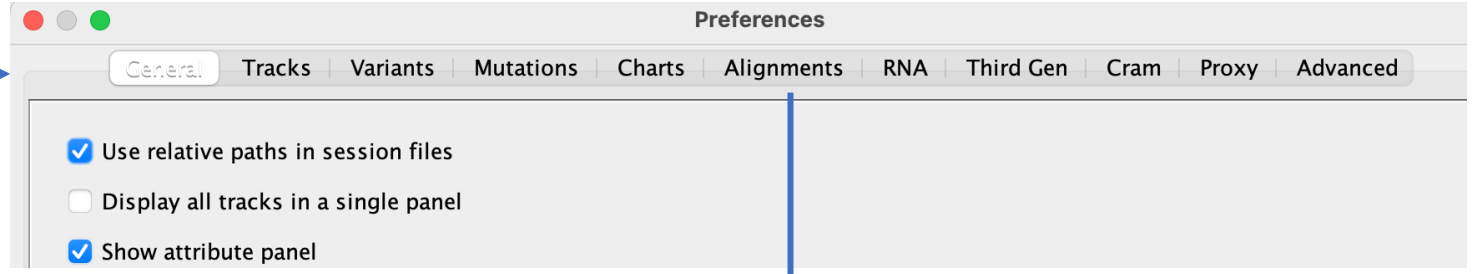


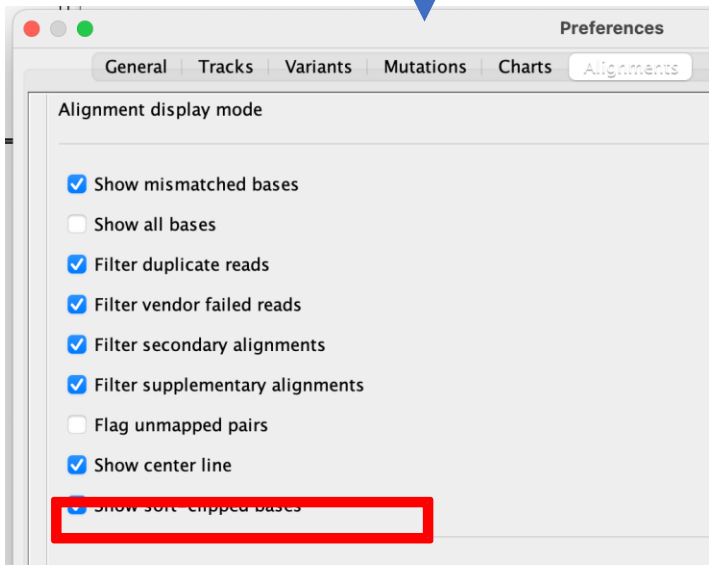
FASTQCの結果、IGVでの変異の確認2

3. IGVを立ち上げてください。

View → Preferences… → Alignments tab を選択





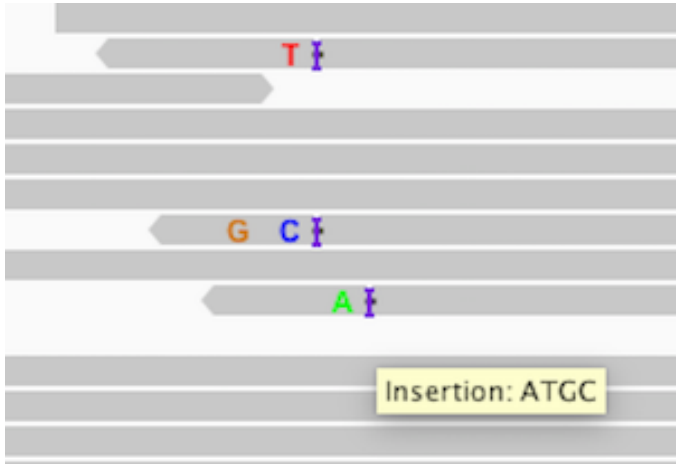


Alignment display mode の Show soft-clipped bases をチェックして、Save してください。

これで soft clip read が表示されるようになります。

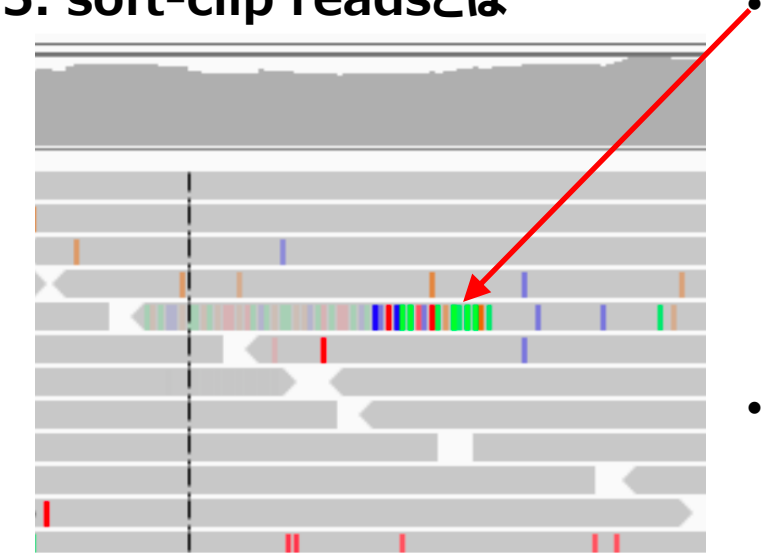
FASTQCの結果、IGVでの変異の確認3

4. IGVにおける、SNV(point mutation）、indel の表示



- **Reference** と違う塩基は、ACGTでそれぞれ、緑、青、茶色、赤、で表示されます。
- **Insertion** は、”I”で表示されます。
- **Deletion** は、”–”で表示されます。

5. soft-clip readsとは



- **Soft-clip reads** とは、リードの一部で reference とは違う塩基が複数続いている、ある程度長さがある塩基配列で、reference とは違うために ACGT それぞれの色で表示れています。Soft-clip の部分は、ゲノム上の他の部分にアラインメントされているため、bam ファイルの sequence の項に情報として残されており、追加情報で、**supplementary alignment** としてアラインメントされている reference の座標情報が記載されています。

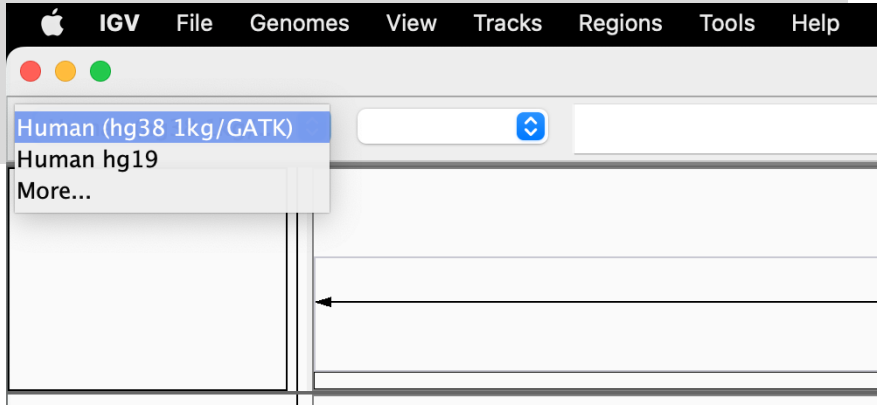
- **Hard-clip reads** というものもあり、これは reference に全くアラインメントされなかったシークエンスで、bam ファイルの sequence の項には情報として残っていません。

FASTQCの結果、IGVでの変異の確認4

6. 展開した jinzai3 のフォルダーにある、README3.txt をダブルクリックして、開いてください。

```
##########################################################################
##
# IGV GRCh38
#
#IGV fusion
# normal.chr9_chr11.GRCh38_markdup.bam
# tumor.chr9_chr11.GRCh38_markdup.bam
9:20377605
11:118488581

##########################################################################
##
#IGV BRAF V600E
# COLO829.BRAF.GRCh38_markdup_updated.bam
# COLO829BL.BRAF.GRCh38_markdup_updated.bam
7:140753336
```



IGVのリファレンスに GRCh38を選択してください。

README3.txt に記載されている変異は、２つ、融合遺伝子 と BRAF V600E の変異です。

FASTQCの結果、IGVでの変異の確認5

6. README3.txt に記載されているファイルを開いて、記載されている座標に飛んでください。

File → Load from File からファイルを開く。

jinzai3.zip を展開したjinzai3/bam フォルダー内にある下記の２つのファイルを開いて、

normal.chr9_chr11.GRCh38_markdup.bam

tumor.chr9_chr11.GRCh38_markdup.bam

README3.txt に記述してある座標それぞれが、

融合部分の座標なので、１つずつコピーして、

9:20377605

11:118488581

Go ボタンの左側にペーストして、**Go** ボタンを押して、

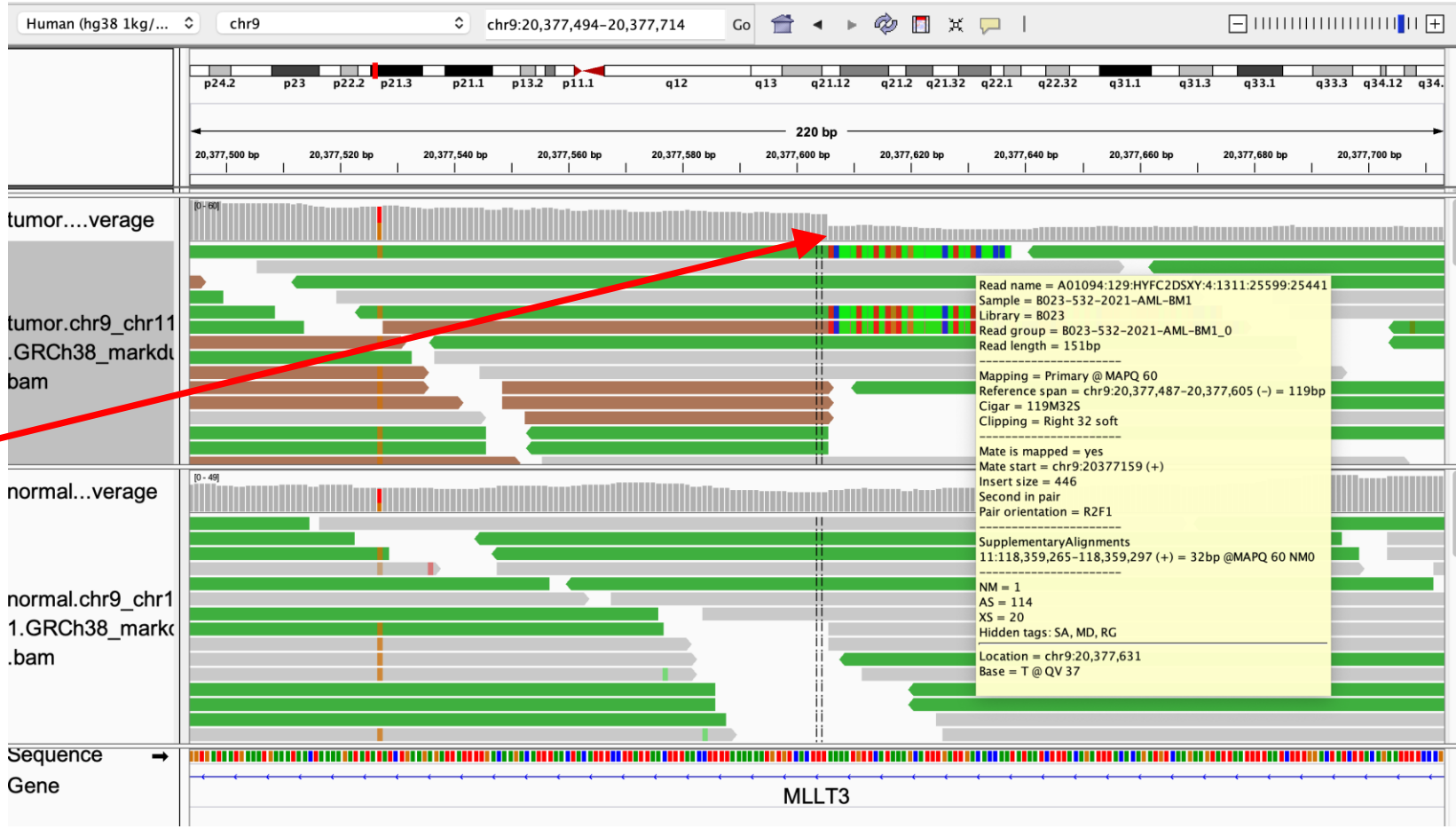
座標に飛んでください。

FASTQCの結果、IGVでの変異の確認6

6. soft-clip read の上にマウスポインターを持っていく（Macの場合）または、右クリック（Windowsの場合）すると、ダイアログが表示されます。

Reference span の部分に
アラインメントされている座標と
Supplementary Alignment の部分に
Soft-clip の部分がアラインメントされている
座標が示されています。

Break point が矢印で示されている座標で、
この座標の左側の部分が chr9 に
アラインメントされ、
左側の部分が chr11 に
アラインメントされています。Chr11 の表示でも同様に、融合部分が確認できます。



FASTQCの結果、IGVでの変異の確認7

6. BRAF V600E の確認

File → New Session で データを unload し、今度は、 README3.txt に記載されている下記の２つのファイルを開いてください。

COLO829.BRAF.GRCh38_markdup_updated.bam

COLO829BL.BRAF.GRCh38_markdup_updated.bam

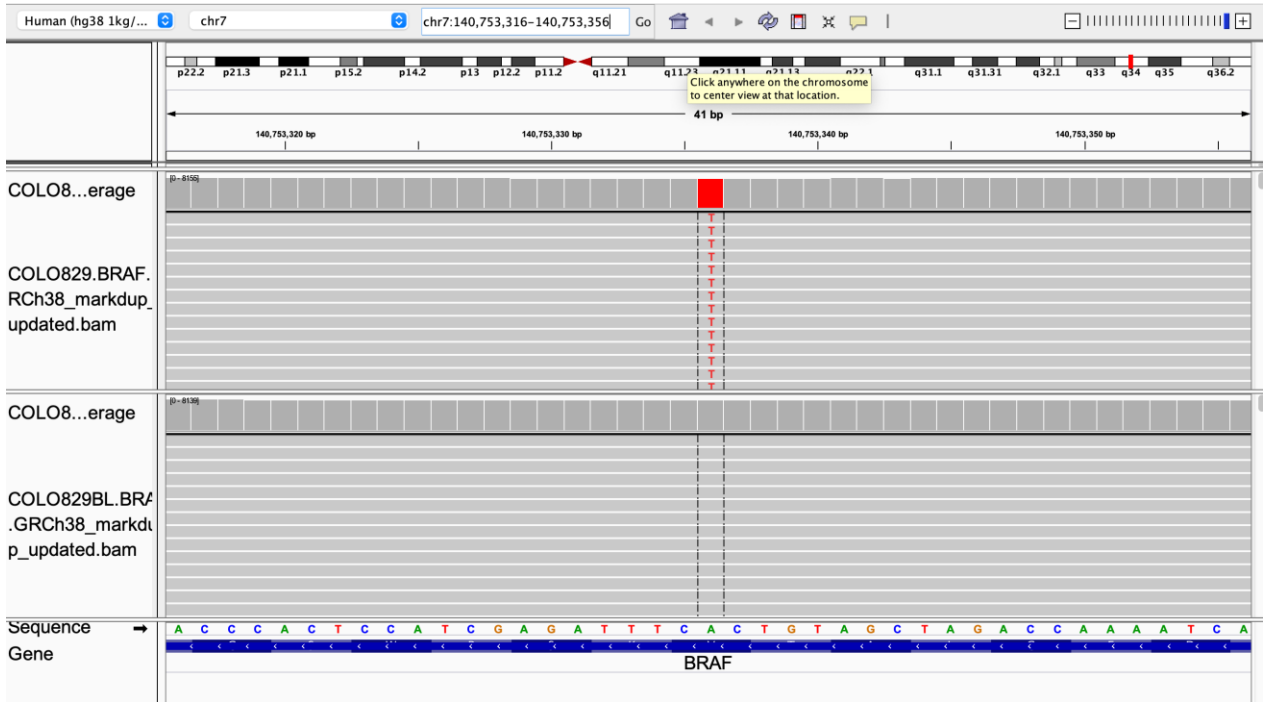
記されている 座標をコピーして、

Go ボタンの左側にペーストして、 **Go** ボタンを押して、

座標に飛んでください。

COLO829.BRAF.GRCh38_markdup_updated.bam

で、SNV が確認できます。



VCFファイルのアノテーション、フィルタリング1

1. ご自分のPCにダウンロードした README4-1.txt をダブルクリックして開いてください。

コマンド実行に関しては、jinza4/data1 にあるデータを使用しております。

data1 に対応した、応用編第4章に記載されているコマンドが記載されています。

コマンドをコピーし、SHIROKANE の terminal にペーストし実行してください。

応用編第Ⅳ章には、ソフトウェアのインストールなども記述されていますが、/share/lect/202210-11/jinzai4 にすでにソフトウェアインストールされていますので、ご自分のホームディレクトリにコピーしていただければコマンドを実行できます。

2. コマンドの実行

Mutect2による変異コール

snpEffによるアノテーション

snpSiftによるアノテーション

snpSiftによるフィルタリング（技術的フィルタリング、生物学的フィルタリング）

Vcfファイルからの変異情報の抽出により、Excelで変異を確認

となっています。

詳しくは、応用編第4章をご覧ください。

VCFファイルのアノテーション、フィルタリング2

1. ご自分のPCにダウンロードした README4-2.txt をダブルクリックして開いてください。

実行に関しては、jjinza4/data2 のデータを使用しており、

data2 に対応した応用編第4章に記載されているコマンドが記載されています。

コマンドをコピーし、SHIROKANE のterminal にペーストし実行してください。

詳しくは、応用編第4章をご覧ください。

第4章のコマンドの実行に関しては、fastqのアラインメント、変異コールと同様に、基本的にコピー、ペーストで実行します。

応用編第4章のオンデマンド動画でも詳しく紹介されているので、割愛させていただきます。

Tumor Mutation Burden1

1. ご自分のPCにダウンロードした README5.txt を開いてください。

SHIROKANEにログインして、さらに qlogin して、

Tumor mutation burden のコマンドを README5.txt からコピーして、ペーストしてください。

```
$ > ssh lect90@slogin.hgc.jp
lect90@slogin.hgc.jp's password:
Last login: Fri Jun 30 11:55:17 2023 from a202175149004.at.hgc.jp

[Begin first by QLOGIN!]
Welcome to SHIROKANE supercomputer of Human Genome Center, IMS, UT.
This is a login node as the gateway with very little software.
You can type  qlogin  COMMAND to use all the features of SHIROKANE.
    |        o                ,,
,--.| ,--.,--.. ,--. |  --, / +-   -+-   .
|  || |  ||  || |  | |   / /  |-- | ` |  |
`--|`-`--'`--|`-`  ` `-- `-`-` '`- `-  `-
   `      `--' https://supcom.hgc.jp/internal/mediawiki/id/357

[A login node of SHIROKANE lect90@slogin1:~]$ q
bash: q: コマンドが見つかりません
ログインノードにはないコマンドかもしれません。qlogin して作業してください。
[A login node of SHIROKANE lect90@slogin1:~]$ qlogin
要求ハードリソース
  memory (s_vmem): 2.1G = ジョブは 1 スロットあたり 2.1G バイトのメモリを要求します
  slots (def_slot): 1 = ジョブは 100% の CPU を要求します
  total memory: 2.1G = ジョブは 2.1G バイトのメモリを要求します
通常の qlogin を実行します。
Your job 77758379 ("QLOGIN") has been submitted
waiting for interactive job to be scheduled ...
Your interactive job 77758379 has been successfully scheduled.
Establishing /home/geadmin/N1GE/utilbin/qlogin_wrapper session to host gc013i ...
export PROMPT_COMMAND=Last login: Fri Jun 30 14:16:51 2023 from gc002i
[OS 7] You are now on OS 7 compute node.
==== あなたのグループ lect のリソース利用状況 ====  hauq command version 1.13
* Home Disk use> 3 TB / 96 TB (3.6 %)    694 kfiles / 96000 kfiles (0.7 %)
* Arch Disk use> 0.0 TB / 0.0 TB (0.0 %)    [ 0.0 TB(cache) + 0.0 TB(tape) ]
* UGE queue use> mjobs.q: 2/4096 (0 %)  ljobs.q: 0/768 (0 %)  lmem.q: 0/128 (0 %)  intr.q: 3/96 (3 %)
resize: unknown character, exiting.
[lect90@gc013 ~]$ export PROMPT_COMMAND=
```

Tumor Mutation Burden2

1. ご自分のPCにダウンロードした README4.txt を開いてください。

SHIROKANEにログインして、さらに qlogin して、

Tumor mutation burden のコマンドを README5.txt からコピーして、ペーストしてください。

応用編第5章 Tumor Mutation Burden を求める（２）のデータが作成されることが確認できると思います。

```
[lect90@gc013 ~]$ grep -a 'protein_coding' ~/jinzai5/data/COLO-829--COLO-829BL.snv.indel.final.v6.annotated.vcf | ¥
> grep -v 'intron_variant' | ¥
> cut -f 8 | ¥
> perl -pe 's/¥|/¥t/g' |¥
> cut -f 4 | ¥
> sort | ¥
> uniq -c | ¥
> awk '{print $2"¥t"$1}'
3_prime_UTR_variant    220
5_prime_UTR_variant    107
coding_sequence_variant&5_prime_UTR_variant   1
downstream_gene_variant772
frameshift_variant     5
inframe_deletion       4
missense_variant       221
missense_variant&splice_region_variant        7
splice_acceptor_variant3
splice_donor_variant   1
splice_region_variant&synonymous_variant      1
start_lost  7
stop_gained 10
stop_gained&splice_region_variant  1
stop_lost&3_prime_UTR_variant      1
synonymous_variant     115
upstream_gene_variant  1003
```

Signature解析1

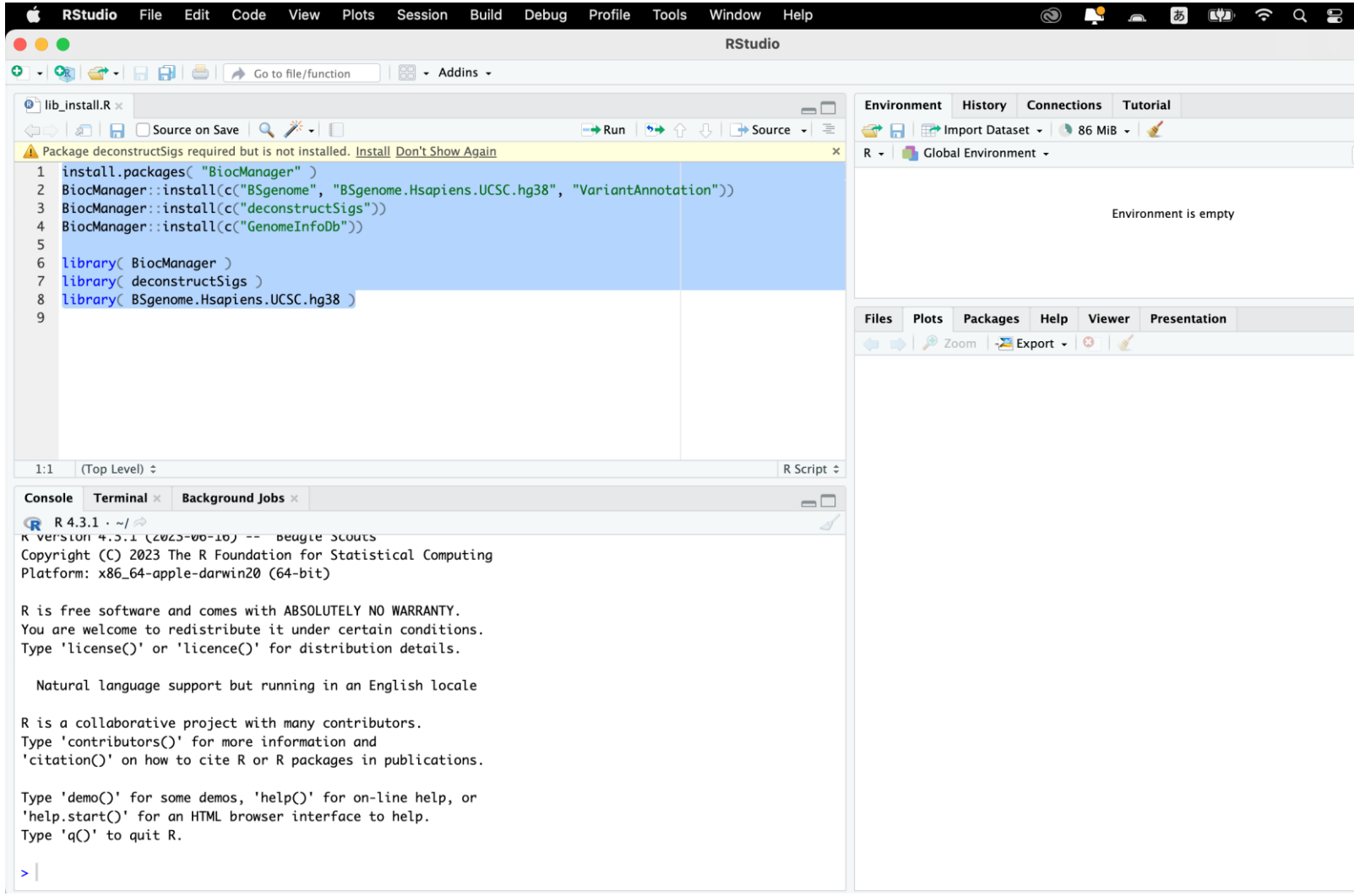
1. サイトからダウンロードした、jinzai5.zip を展開してください。

2. RStudio を立ち上げてください。画面は、R4.3.1を使用しています。

1. File → Open File から 展開した jinzai5 フォルダーにある、lib_install.R を開いてください。

4. lib_install.R の上から1行ずつ先頭にカーソルを合わせて、真ん中上にある、→Run ボタンを押して、1行づつ実行してライブラリーをインストールしてください。
インターネットからライブラリーをインストールするので、インターネットに接続する必要があります。

5. 最後のlibrary の3行を実行して、ライブラリーをロードしてください。



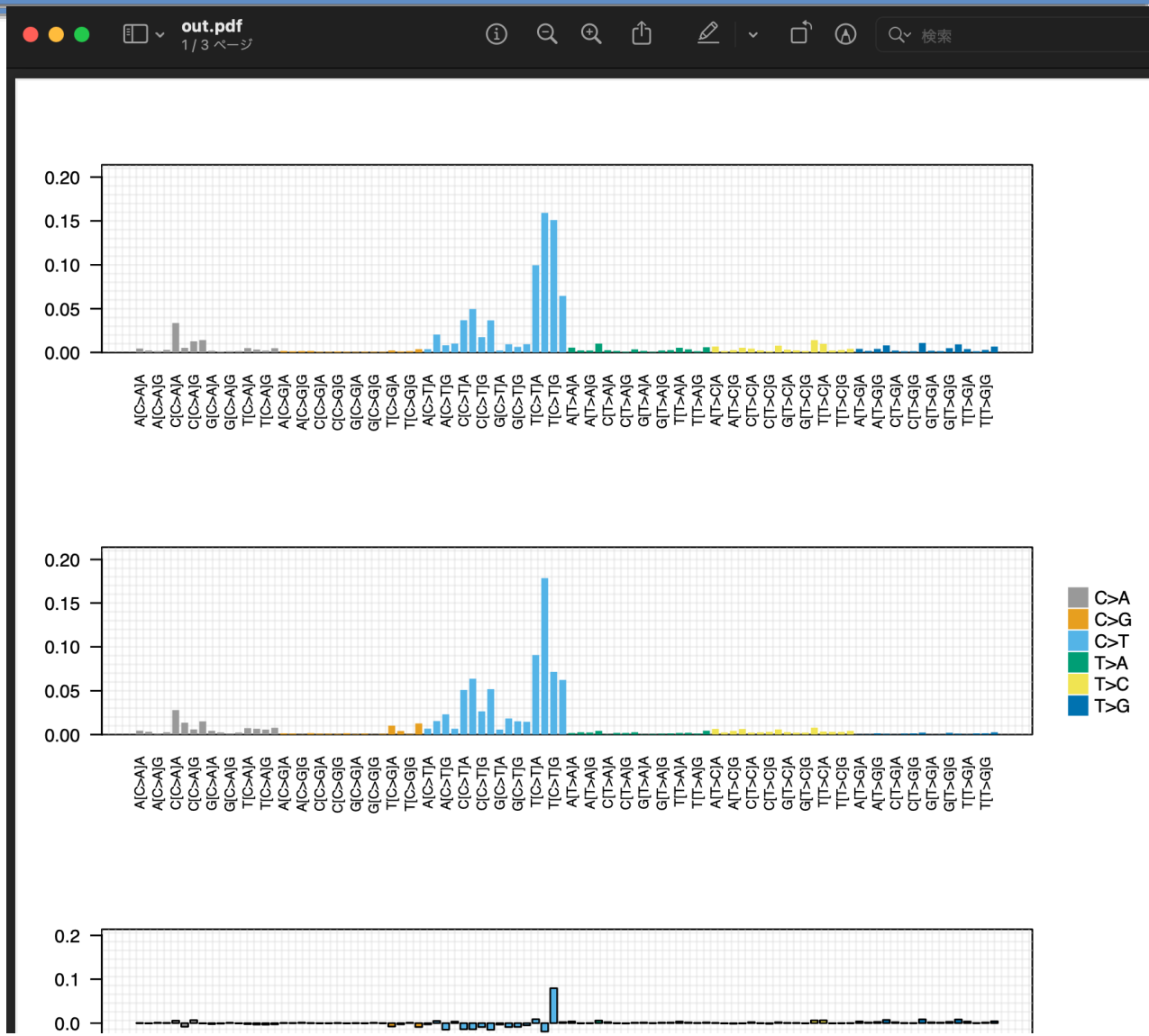
Signature解析2

1. File → Open File から 展開した jinzai5
フォルダーから、Mac PC をご使用の方は、signature.Mac.R を
Windows PC をご使用の方は、signature.Windows.R を開いてください。

2. ファイルを編集します。
home_dir に指定してあるディレクトリを、展開した jinzai5 のフォルダー
のあるディレクトリに変更します。

3. Rのスクリプトの全ての行を選択して、 →Run ボタンを押して、
実行してください。

4. うまく動作すれば、結果のファイル out.pdf が
jinzai5/data フォルダーに作成されていることが確認できると思います。



監修者

本テキストの監修者は下記の通りとなります。

監修者	所属
井元清哉	東京大学医科学研究所　ヒトゲノム解析センター　健康医療インテリジェンス分野　教授