

Esercitazione di genotipizzazione

Corso di Genomica Computazionale, Laurea Specialistica in Genomica Funzionale,
Universita' di Trieste, Fabrizio Mafessoni, Tutor: Daniela Eugenia Nerelli

Installiamo bcftools

Usando conda

```
conda install -c bioconda bcftools
```

Oppure con brew (per mac) o apt per linux/windows

```
sudo apt install bcftools
```

```
brew install bcftools
```

Genotipizzazione a singolo campione

Scarichiamo le prime reads allineate (bam) di un individuo caucasico dal 1000 Genomes project. Utilizziamo solo il cromosoma 20 per velocizzare i calcoli.

```
wget
```

```
https://1000genomes.s3.amazonaws.com/phase3/data/NA12878/alignment/NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam
```

Indicizziamo il bam

```
samtools index
```

```
NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam
```

Scarichiamo il genoma di riferimento

```
wget https://storage.googleapis.com/gatk-best-practices/somatic-b37/Homo_sapiens_assembly19.fasta
```

Per lavorare più velocemente concentriamoci solo sul cromosoma 20, indicizzando il genoma di riferimento ed estraendo il cromosoma 20

```
samtools faidx Homo_sapiens_assembly19.fasta
```

```
samtools faidx Homo_sapiens_assembly19.fasta 20 > chr20.fa
```

```
samtools faidx chr20.fa
```

Esploriamo il nostro file bam e verifichiamo che abbiamo solo reads sul cromosoma 20 con `idxstats`.

```
samtools view
```

```
NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam | head
```

```
samtools idxstats
NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam
```

Osserviamo le nostre reads nel bam in formato mpileup

```
samtools mpileup
NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam
```

Notate che mpileup **non chiama genotipi**.

Costruisce una **rappresentazione base-per-base** di cosa mostrano le letture rispetto al riferimento in un dato sito. È quindi un formato di **evidenza grezza**, che poi viene passato a **bcftools call** per fare la vera chiamata di variante/genotipo.

Significato delle colonne:

Colonna	Nome	Significato
1	Chromosome	Nome del contig (es. 20, chr20)
2	Position	Posizione 1-based sul riferimento
3	Reference base (REF)	La base attesa dal genoma di riferimento
4	Depth (DP)	Numero totale di letture che coprono la posizione
5	READS column	Cosa mostrano le letture
6	QUAL column	Qualità dei nucleotidi (ASCII 33-based, come FASTQ)

Notate che in READ:

A C G T (maiuscole) base letta **su filamento forward**

a c g t (minuscole) base letta **su filamento reverse**

\$ indica la fine di una lettura di una sequenza, mentre ^ l'inizio.

Riguardo agli INDELS invece:

Esempio Significato

+3ACT inserzione di lunghezza 3 con sequenza ACT

-2GA delezione di lunghezza 2 di GA

Usiamo il Pileup di bcftools per creare le genotype likelihood, senza ancora chiamare il campo GT (genotipo)

```
bcftools mpileup -f chr20.fa
NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam |
head
```

Queste hanno il seguente formato:

Campo	Nome	Significato
PL	Phred-scaled Genotype Likelihoods	3 numeri (0/0, 0/1, 1/1)
GL (a volte)	Genotype Likelihoods log10	più rari; PL è lo standard
DP	Depth	numero letture
AD	Allele Depths	numero REF, ALT

Ad esempio:

GT:PL:DP:AD

0/1:135,0,161:11:6,5

Valore Significato

GT = 0/1 Chiamato come eterozigote

PL = 135,0,161 Likelihoods **PHRED** per 0/0, 0/1, 1/1

DP = 11 11 letture totali

AD = 6,5 6 REF e 5 ALT

Come si leggono i PL (phred-scaled likelihoods)

$PL = -10 \log_{10} P(\text{dati} | \text{genotipo})$

→ più piccolo = più probabile.

Nell'esempio:

- 0/0 ha PL = 135 → molto improbabile
- **0/1 ha PL = 0 → genotipo più probabile**
- 1/1 ha PL = 161 → molto improbabile

Mentre mpileup calcola quanto sono probabili i dati per ogni genotipo (0/0,0/1,1/1), call sceglie il genotipo più probabile utilizzando le genotype likelihood calcolate da mpileup. Dunque, chiamiamo le varianti, aggiungendo il genotipo GT:

```
bcftools mpileup -Ou -f chr20.fa -a FORMAT/DP,FORMAT/AD \
    NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam \
| bcftools call -mv -Oz -o calls.vcf.gz
```

```
bcftools index calls.vcf.gz
```

Creiamo una piccola tabella dei genotipi

```
bcftools query -f '%CHROM\t%POS\t%REF\t%ALT\t%QUAL\t[%DP]\n'
calls.vcf.gz | head
```

Usiamo awk per contare SNPs e INDELs

```
bcftools view -H calls.vcf.gz \
| awk '{if(length($4)==1 && length($5)==1) snp++; else ind++;}
END{print "SNP:",snp,"INDEL:",ind}'
```

Usiamo bcftools per filtrare il nostro vcf

```
bcftools view -g het calls.vcf.gz | head

bcftools view -i 'FORMAT/DP>=10 && QUAL>=30' calls.vcf.gz |
head
```

Genotipizzazione multi-campione

Per ora abbiamo genotipizzato un singolo campione. Proviamo ora a fare una genotipizzazione multi-campione scaricando altri individui dai 1000 genomes:

```
wget
https://1000genomes.s3.amazonaws.com/phase3/data/NA12878/align
ment/NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.ba
m
```

```
wget
https://1000genomes.s3.amazonaws.com/phase3/data/HG00096/align
ment/HG00096.chrom20.ILLUMINA.bwa.GBR.low_coverage.20120522.ba
m
```

```
wget
https://1000genomes.s3.amazonaws.com/phase3/data/HG00110/align
ment/HG00110.chrom20.ILLUMINA.bwa.GBR.low_coverage.20130415.ba
m
```

```
samtools index
NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam
```

```
samtools index
HG00096.chrom20.ILLUMINA.bwa.GBR.low_coverage.20120522.bam
```

```
samtools index
HG00110.chrom20.ILLUMINA.bwa.GBR.low_coverage.20130415.bam
```

Esercizio supplementare (provate a implementare questi ultimi comandi in modo migliore con un for loop).

Creiamo una piccola lista dei nostri bam

```
printf "%s\n" \  
    NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam \  
    HG00096.chrom20.ILLUMINA.bwa.GBR.low_coverage.20120522.bam \  
    HG00110.chrom20.ILLUMINA.bwa.GBR.low_coverage.20130415.bam \  
> bamlist.txt
```

Genotipizziamo i nostri tre individui. Notate la sintassi. Esplorate la funzione delle varie flag con `bcftools --help`

```
bcftools mpileup -Ou -f chr20.fa \  
    -a FORMAT/DP,FORMAT/AD,FORMAT/ADF,FORMAT/ADR \  
    NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam \  
| bcftools call -m -v -Oz -o NA12878.single.recall.vcf.gz  
bcftools index NA12878.single.recall.vcf.gz
```

e rigenotipizziamo il nostro individuo singolo

```
bcftools mpileup -Ou -f chr20.fa \  
    -a FORMAT/DP,FORMAT/AD,FORMAT/ADF,FORMAT/ADR \  
    -b bamlist.txt \  
| bcftools call -m -v -Oz -o trio.recall.vcf.gz  
bcftools index trio.recall.vcf.gz
```

Usiamo il comando `isec` per ottenere una comparazione delle nostre chiamate a singolo e multi-campione:

```
bcftools isec -p isec_out NA12878.single.recall.vcf.gz  
trio.recall.vcf.gz
```

Notate che gli output (nella cartella `isec_out`) hanno questa struttura:

0000.vcf → varianti solo in file #1

0001.vcf → variant solo in file #2

0002.vcf → varianti in entrambi (record come in file #1)

0003.vcf → varianti in entrambi (record come in file #2)

Esercizi:

- 1) Esplorate le differenze tra le chiamate per i differenti individui usando awk e i comandi bash di vostra conoscenza. Quante varianti sono state perse, e quante guadagnate con la genotipizzazione multi-campione? Suggerimento: testate tutti i campioni usando un `for` loop.
- 2) Comparete le posizioni da voi trovate con i file samtools mpileup. Che tipo di varianti sono scomparse nella genotipizzazione multi-campione? Quali sono state guadagnate?
- 3) Scaricate l'intero dataset del 1000 genomes per il cromosoma 20 ed esploratelo con bcftools. Ricordate di indicizzare. Estraiete i genomi dei campioni da voi analizzati individualmente e comparatene i genotipi. Cosa notate? Avete perso o guadagnato varianti? Per scaricare i dati usate: `wget`
https://hgdownload.cse.ucsc.edu/gbdb/hg19/1000Genomes/phase3/ALL.chr20.phase3_shapeit2_mvncall_integrated_v5a.20130502.genotypes.vcf.gz

E per estrarre i vostri campioni usando la flag `-S` di bcftools

```
printf "NA12878\nNA12891\nNA12892\n" > samples.txt
bcftools view -S samples.txt -Oz \
```

```
ALL.chr20.phase3_shapeit2_mvncall_integrated_v5a.20130502
.genotypes.vcf.gz \
> trio.chr20.1000G.vcf.gz
```

Opzionalmente, per velocizzare, potete estrarre solo varianti in una data regione con

`REGION=20:10000000-10040000`

```
bcftools view -r $REGION -Oz -o trio.$REGION.vcf.gz
trio.chr20.1000G.vcf.gz
bcftools index trio.$REGION.vcf.gz
```

- 4) Chiamate i genotipi per il cromosoma 20 con bcftools per i genomi presenti in:
<http://ftp.eva.mpg.de/denisova/BAM/human/>