

#ahora creamos la carpeta para el punto 3 prokka

- cd ../
- cd ../
- mkdir Punto3
- cd Punto3

#copiamos nuevamente el genoma ancestral en la carpeta para el punto3

- cp ../../ensambles/ancestral\_assembly/contigs.fasta .

#instalamos prokka en caso de no tenerlo

- mkdir prokka\_results
- cd prokka\_results
- conda activate bioinfo
- conda install -c conda-forge -c bioconda -c defaults prokka
- prokka --version
- prokka --outdir anotacion\_ancestral --prefix ecoli\_ancestral punto\_3/contigs.fasta

#luego creamos el Directorio del punto 4 y enlazamos los archivos vcf necesarios

- cd ../
- mkdir Punto4
- cd Punto4
- mkdir evol\_1
- mkdir evol\_2

#aqui se escribe el path absoluto con el único motivo de mover archivos de carpeta a carpeta para seguir trabajando siempre con el path relativo

- cd evol\_1
- ln -s ../Punto2/evol1/evol1\_variantes\_filtradas.vcf .
- ln -s ../Punto2/evol1/evol1\_variantes.vcf .

- cd ../

- cd evol\_2
- ln -s ../Punto2/evol2/evol2\_variantes\_filtradas.vcf .
- ln -s ../Punto2/evol2/evol2\_filtradas.vcf .

#ahora generamos los snpEff para ambos evol

# instalamos el snpeff

- Conda activate bioinfo
- conda install -c bioconda snpeff
- mkdir SnpEff\_resultados

# según un proyecto en github es importante para trabajar con SnpEff trabajar con su base de datos local, por lo que se hizo lo siguiente .

#cuando SnpEff esta instalado en conda tenemos que ir a su ubicación en el directorio, en nuestro caso

- cd ~/micromamba/envs/bioinfo/share/snpeff-5.0-1
- mkdir data
- mkdir data/ecoli\_ancestral

#ahora traeremos los archivos de ensamblaje y la anotación de prokka con cp y los ubicamos en data

- cp ~/entrega\_1/assembly/contigs.fasta  
data/ecoli\_ancestral/sequences.fa
- cp ~/punto\_3/prokka\_results/ecoli\_ancestral.gff  
data/ecoli\_ancestral/genes.gff

#ejecutamos lo siguiente

- nano snpEff.config #ahí está toda la base de datos de snpEff

#y en el archivo nano pegamos (ecoli\_ancestral.genome : Escherichia\_coli\_ancestral) guárdamos con ctrl o y  
volvemos con ctrl x

#después debemos construir la base de datos de la siguiente manera :

- java -jar snpEff.jar build -gff3 -v ecoli\_ancestral #donde se lean los resultados gff de prokka y las  
secuencias de ensamblaje y con eso se creará la base de datos interna que SnpEff usará para mapear  
variantes.

#predicciones funcionales

- snpEff -v ecoli\_ancestral ~/punto\_4/evol\_1/evol1\_variantes\_filtradas.vcf > evol1\_annotadas.vcf
- snpEff -v ecoli\_ancestral ~/punto\_4/evol\_2/evol2\_variantes\_filtradas.vcf > evol2\_annotadas.vcf

# por ultimo con mv movemos estos 2 archivos a SnpEff\_resultados

# hay muchas variants y solo debemos utilizar las que nos den más información o nuestras variantes de interés, por  
lo que hicimos lo siguiente

- cd punto\_4
- mkdir Extra
- cd Extra

#los archivos vcf son difíciles de visualizar y por ende filtrar. Entonces vamos a extraer su información (con detalles  
importantes como CHROM (cromosoma) POS (posición variante) REF (nucleótido referencia) ALT (variante) QUAL  
(calidad) INFO (profundidad) INFO/ANN (anotaciones sobre genes)).

Para después convertir vcf a csv, separado por comas.

#Para evol 1

- bcftools query -f '%CHROM\t%POS\t%REF\t%ALT\t%QUAL\t%INFO/DP\t%INFO/ANN\n' /  
punto\_4/evol\_1/evol1\_annotadas.vcf | sed 's/\t/,/g' > evol1\_ann\_table.csv1

#filtrar solo high moderate

- bcftools view -i 'INFO/ANN ~ "HIGH|MODERATE"' /punto\_4/evol\_1/evol1\_annotadas.vcf -Ov -o  
punto\_4/evol\_1/evol1\_high\_mod.vcf

#para evol 2

- bcftools query -f '%CHROM\t%POS\t%REF\t%ALT\t%QUAL\t%INFO/DP\t%INFO/ANN\n' /  
punto\_4/evol\_2/evol2\_annotadas.vcf | sed 's/\t/,/g' > /punto\_4/evol\_2/evol2\_ann\_table.csv1

#filtrar solo high moderate

- bcftools view -i 'INFO/ANN ~ "HIGH|MODERATE"' /punto\_4/evol2\_annotadas.vcf -Ov -o /  
punto\_4/evol\_2/evol2\_high\_mod.vcf

Luego de tener estos archivos csv lo que hicimos para seleccionar las variantes de interés fue cargar estos archivos

a jupyter y empezar a filtrar manualmente la tabla resultante hasta tener solo las variantes más representativas

Luego convertimos la tabla resultante de Python nuevamente en csv para hacer el análisis en el trabajo

Todo esto está adjunto en la carpeta del punto 4 Extra