(a) From config.yaml input file into VIPER

```
#Turn on the following if you want a genome wide snp scan in addition to chr6 scan
# possible value are [true,false]
snp_scan_genome: false
```

(b) From SNP Module of VIPER

```
#-------------------------------------------------------------------------------
# snp calling GENOME wide (hidden config.yaml flag- 'snp_scan_genome:True'
#-------------------------------------------------------------------------------

rule call_snps_genome:
    input:
        bam="analysis/STAR/{sample}/{sample}.sorted.bam",
        ref_fa=config["ref_fasta"],
    output:
        protected("analysis/snp/{sample}/{sample}.snp.genome.vcf")
    params:
        varscan_path=config["varscan_path"]
    message: "Running varscan for snp analysis genome wide"
    shell:
        "samtools mpileup -f {input.ref_fa} {input.bam} | awk \'$4 != 0\' | "
        "{params.varscan_path} mpileup2snp - --min-coverage 20 --min-reads2 4 --output-vcf > {output}"


rule snpEff_annot:
    input:
        vcf="analysis/snp/{sample}/{sample}.snp.genome.vcf"
    output:
        vcf_annot = protected("analysis/snp/{sample}/{sample}.snpEff.annot.vcf"),
        vcf_stats = protected("analysis/snp/{sample}/{sample}.snpEff_summary.html")
    params:
        snpEff_conf=config["snpEff_conf"],
        snpEff_db=config['snpEff_db']
    message: "Running varscan for snpEff annotation analysis"
    shell:
        "snpEff -Xmx2G -stats {output.vcf_stats} -c {params.snpEff_conf} {params.snpEff_db} {input.vcf} > {output.vcf_annot}"
```

Supplementary Figure 4