oncoKB注释流程

> OncoKB：由**Memorial Sloan Kettering癌症中心（MSK）**开发并维护的精准肿瘤学知识库。该知识库以**体细胞突变**为核心，收录突变对应靶向药的精准使用、突变在生物学与肿瘤学方面的影响以及突变在人群中的分布频率特征等信息。
>
> 该知识库收录信息的来源非常多样化，包括FDA、NCCN、ASCO或ESMO会议论文、不同癌种专家共识以及科学文献等。虽然知识库的信息来源多样化，但是每条信息都会经过临床基因组学注释委员会的定期审阅与修订，保证信息的准确性与严谨性。
>
> 与其他收录体细胞突变的数据库（如COSMIC）相比，OncoKB的主要内容与**肿瘤精准用药**相关。

1. oncoKB注释流程由xcw已经搭建完成，具体内容如下：

```
#workflow for oncokb-annotator
workflow oncokb_annotator {
    File outdir
    File input_vcf
    String sample
    File ref="vol1@xtao:rendong/project_info/ref_file/hg19/hg19.fasta"
    File VEP_data="vol1@xtao:test/xcw/DB/.vep"
    String API="fbedbe37-0e48-4d0b-8f3b-386e306c11e8"
    call annotator {
        input:
            ref=ref,
            outdir=outdir,
            sample=sample,
            input_vcf=input_vcf,
            VEP_data=VEP_data,
            API=API
    }
}
task annotator {
    File ref
    File outdir
    File input_vcf
    String sample
    File VEP_data
    String API
    command {
        export PATH=/root/miniconda3/bin:$PATH
        perl /home/software/vcf2maf-1.6.21/vcf2maf.pl --input-vcf ${input_vcf} --output-maf ${outdir}/${sample}.maf --vep-path /root/miniconda3/bin --vep-data ${VEP_data} --ref-fasta ${ref} --species homo_sapiens --vep-overwrite && python2 /home/software/oncokb-annotator-3.0.0/MafAnnotator.py -i ${outdir}/${sample}.maf -o ${outdir}/${sample}.hgsp_short.oncokb.xls -b ${API} -q hgvsp_short
    }
    runtime {
        ldapauth:true
        docker: "oncokb_annotator:v10"
        memory: 4000 + "MB"
        cpu: 1
    }
    output {
        File results="${outdir}/${sample}.hgsp_short.oncokb.xls"
```

```
    }
}
```

其中**必须在oncoKB网站用教育邮箱注册账号得到API**才可调用oncokb-annotator。另外VCF文件不能直接注释，必须转换成MAF文件。故第一步先用`vcf2maf`转换格式，将combine.vcf转换成maf，其中会产生中间文件combine.vep.vcf。

2. 配置json文件，内容如下：

```
{
    "InputDataSet": {
        "WorkflowInput": {
            "oncokb_annotator.outdir": "vol1@xtao:test/wbzhao/oncokb",

 "oncokb_annotator.input_vcf":"vol1@xtao:test/wbzhao/combine_vcf/id.combine.vcf"
,
            "oncokb_annotator.sample":"id"
        }
    },
    "Name": "ONCOKB_test",
    "Pipeline": "ONCOKB",
    "Priority": 8
}
```

注意！pipeline无法访问到vol1_root或商检目录的原始combine.vcf文件，所以**必须把vcf文件cp到当前工作目录**，注意增加样本编号前缀以区分文件，且，vcf文件**必须增加列名header**。

```
#somatic 体系文件为combine.vcf
sed -i '1i\#CHROM\tPOS\tID\tREF\tALT\tQUAL\tFILTER\tINFO\tFORMAT\tTUMOR\tNORMAL'
combine.vcf
#germline 胚系没有combine.vcf，选择genotype_split.vcf
sed -i '1i\#CHROM\tPOS\tID\tREF\tALT\tQUAL\tFILTER\tINFO\tFORMAT\tNORMAL'
genotype_split.vcf
```

如果不增加header就无法输出count。

```
#批量生成json文件，其中id.list为样本编号列表
for i in `cat id.list`;do cp oncokb.json ${i}.json&&sed -i s/id/${i}/g
${i}.json;done
```

注意！json文件投递有数量限制，一次只能投递100个，需根据资源消耗度分批投递。

```
biocli job submit ${i}.json
```

3. 当遇到一些老的项目可能不存在combine.vcf，需要从mutect.vcf生成combine.vcf，请运行以下流程：

```
####如果是somatic则运行两步，germline运行一步，到genotype_split.vcf即可
python
/mnt/vol1_root/rendong/project_info/pipeLine/scripts/docker_contain/wdl_python/BioBin/vcf_genotype_split.py
/mnt/vol1_root/rendong/project/$panel/$id/3.somatic_VC/mutect.vcf
/mnt/vol2_ws/test/wbzhao/combine_vcf/$id.genotype_split.vcf&&python
/mnt/vol1_root/rendong/project_info/pipeLine/scripts/docker_contain/wdl_python/BioBin/complex_variant.py -i
/mnt/vol2_ws/test/wbzhao/combine_vcf/$id.genotype_split.vcf -f
/mnt/vol1_root/rendong/project_info/ref_file/hg19/hg19.fasta -o
/mnt/vol2_ws/test/wbzhao/combine_vcf/$id.combine.vcf
```

4.当遇到vcf文件很大时，process会断掉，报错如下：

```
------------------------------------------------------------
| TaskID: paladin-task.969de460-e089-45ee-8a1b-d9e6c963f3ee.cmd_err
------------------------------------------------------------
STATUS: Running VEP and writing to: /mnt/vol2_ws/test/wbzhao/combine_vcf/2011885.combine.vep.vcf
Possible precedence issue with control flow operator at /root/miniconda3/lib/site_perl/5.26.2/Bio/DB/IndexedBase.pm line 805.
```

遇到这种情况需要①切分文件，请使用`split`命令切分vcf文件后②增加header然后③分开注释，最后④合并注释文件。