Tutorial: How to convert VCF file to a binary matrix for somatic mutation data

Prerequisites

1. **VCF file**: A VCF (Variant Call Format) file is a standard file format developed by the 1000 Genomes project that describes variant calls in a genome. These files are used in bioinformatics to encode DNA sequence variations including single nucleotide polymorphisms (SNPs), insertions, deletions, and other variations.
2. **MAF file**: A MAF (Mutation Annotation Format) file is a tab-delimited text file with aggregated mutation information from VCF files. It's a standard format used by The Cancer Genome Atlas (TCGA) and other projects, and it includes information about the type of mutation and its location in the genome.
3. **VEP (Variant Effect Predictor)**: This is a tool developed by Ensembl for annotating variants. It provides information about the effect of variants (e.g., SNPs, insertions, deletions) on genes, transcripts, and protein sequence, as well as regulatory regions.
4. **vcf2maf**: This is a tool that converts a VCF into a MAF. You can find it here.
5. **SMDIC package**: This is an R package that can convert a MAF file into a binary matrix. You can find it here.
6. **R and RStudio**: You will need R and RStudio installed on your computer. You can download R here and RStudio here.
7. **Unix-like operating system** (Linux, macOS) and enough storage space to store the VCF, MAF, and binary matrix files

Step 1: Install VEP on Linux

1. **Install conda**: If you don't already have conda, install it into **$HOME/miniconda3** as follows:

```
curl -sL https://repo.anaconda.com/miniconda/Miniconda3-py37_4.9.2-Linux-x86_64.sh -o /tmp/miniconda.sh
sh /tmp/miniconda.sh -bfp $HOME/miniconda3
```

2. **Add the conda bin folder into your $PATH**: You can also add this to your **~/.bashrc** or **~/.profile** for this to persist across logins:

```
export PATH=$HOME/miniconda3/bin:$PATH
```

3. **Download and install VEP, its dependencies, and also samtools/bcftools/liftOver**:

```
conda install -qy -c conda-forge -c bioconda -c defaults ensembl-vep==102.0 htslib==1.10.2 bcftools==1.10.2 samtools==1.10 ucsc-liftover==377
```

4. **Download VEP's offline cache for GRCh38, and the reference FASTA**:

```
mkdir -p $HOME/.vep/homo_sapiens/102_GRCh38/
rsync -avr --progress rsync://ftp.ensembl.org/ensembl/pub/release-102/variation/indexed_vep_cache/homo_sapiens_vep_102_GRCh38.tar.gz $HOME/.vep/
tar -zxf $HOME/.vep/homo_sapiens_vep_102_GRCh38.tar.gz -C $HOME/.vep/
```

```
rsync -avr --progress rsync://ftp.ensembl.org/ensembl/pub/release-102/fasta/homo_sapiens/dna_index/ $HOME/.vep/homo_sapiens/102_GRCh38/
```

Step 2: Install vcf2maf on Linux

vcf2maf can be installed by downloading the latest release from GitHub or using Docker. Here are the steps for both methods:

1. **Download the latest release**: You can download the latest release from GitHub using the following commands in your terminal:

```
export VCF2MAF_URL=`curl -sL https://api.github.com/repos/mskcc/vcf2maf/releases | grep -m1 tarball_url | cut -d\" -f4`
curl -L -o mskcc-vcf2maf.tar.gz $VCF2MAF_URL
```

2. **Extract the downloaded file and navigate to the vcf2maf directory**:

```
tar -zxf mskcc-vcf2maf.tar.gz
cd mskcc-vcf2maf-*
```

3. **Check the installation**:

```
perl vcf2maf.pl --man
perl maf2maf.pl -man
```

Step 3: Convert VCF to MAF using vcf2maf

Once you have vcf2maf installed, you can convert your VCF files into a combined MAF file. Here's a basic command to do this:

```
perl vcf2maf.pl --input-vcf /path/to/input.vcf --output-maf /path/to/output.maf
```

Replace **/path/to/input.vcf** with the path to your VCF file and **/path/to/output.maf** with the path where you want the MAF file to be saved.

Step 4: Install the SMDIC Package in R

1. **Open RStudio**: Start by opening RStudio on your computer.
2. **Install the SMDIC package**: Use the **install.packages** function to install the SMDIC package:

```
install.packages("SMDIC")
```

3. **Load the SMDIC package**: Once the package is installed, you can load it into your R environment using the **library** function:

```
library(SMDIC)
```

Step 5: Load the MAF File into R

You can load the MAF file into R with the following command:

```
maf <- read.maf("/path/to/output.maf")
```

Replace **/path/to/output.maf** with the path to your MAF file.

Step 6: Convert the MAF File into a Binary Matrix

You can convert the MAF file into a binary matrix with the following command:

```
binary_matrix <- maf2binary(maf)
```