MultiMuC

MultiMuC is a **somatic mutation caller for multi-regional tumors**.
This method can use the stochastic modeling of the state-of-the-art mutation calling methods for single-regional tumors, e.g., Strelka2, NeuSomatic, through the probability-based outputs. Furthermore, this method can avoid a case of performance degradation ("No-TP" case defined in our paper).

How to build

This script is implemented in Julia language. We checked the result in Julia v1.0.1.
Install julia language before using our script.

Instantiate package

Before running the script, please instantiate the package and install dependencies.

```
% cd ./src
% julia
julia> ]
(v1.0) pkg>
(v1.0) pkg> activate .
(src) pkg> instantiate
```

How to run

Note. At the first execution, package builging will be conducted, and simultaneous run (running in multiple processes) should be conducted after the first time of running.

To execute, type as follows.

```
julia --project ./src/main.jl MAP \\
--seed ${seed} -n ${iter} -b ${burnin} \\
--pdata=0.02 --pevidence=0.2 --pcon=0.999 --theta=0.5 --rho=0.1 \\
-o ${output_dir} ${BF_L} ${BF_H}
```

where

\${seed}: a random seed.
\${iter}: number of sampling after burnin.
\${burnin}: number of sampling in burnin of MCMC.
\${BF_L}: a tsv file for $\log_{10}L_{i,j}$.
\${BF_H}: a tsv file for $\log_{10}H_{i,j}$.

The following shows an example setting of $L_{i,j}$, $H_{i,j}$.
$$
L_{i,j} = \begin{cases}
BF_{i,j} \cdot 10^{a} & (\mbox{Bayes factor is available}) \\
10^{-2} & (\mbox{Otherwise}) \\
\end{cases}, \\
H_{i,j} = \begin{cases}
BF_{i,j} \cdot 10^{-1.5} & (\mbox{Bayes factor is available}) \\
10^{-2} & (\mbox{Otherwise}) \\
\end{cases},
$$
where
i: the sample index
j: the genomic position
$BF_{i,j}$: the original Bayes factor from a state-of-the-art mutation calling method.
$\frac{1}{10^{a}}$: threshold of original Bayes factor to detect mutation, e.g., $a=0.0$ or $a=-0.5$.

Input format

#Comments 1
gene_1 $\log_{10}L_{1,1}$ $\log_{10}L_{2,1}$ $\log_{10}L_{3,1}$
gene_2 $\log_{10}L_{1,2}$ $\log_{10}L_{2,2}$ $\log_{10}L_{3,2}$
gene_3 $\log_{10}L_{1,3}$ $\log_{10}L_{2,3}$ $\log_{10}L_{3,3}$

#Comments 2
gene_1 $\log_{10}H_{1,1}$ $\log_{10}H_{2,1}$ $\log_{10}H_{3,1}$
gene_2 $\log_{10}H_{1,2}$ $\log_{10}H_{2,2}$ $\log_{10}H_{3,2}$
gene_3 $\log_{10}H_{1,3}$ $\log_{10}H_{2,3}$ $\log_{10}H_{3,3}$

Output format

#Comments 1 are copied
gene_1 $Y_{1,1}$ $Y_{2,1}$ $Y_{3,1}$
gene_2 $Y_{1,2}$ $Y_{2,2}$ $Y_{3,2}$

gene_3 $Y_{1,3}$ $Y_{2,3}$ $Y_{3,3}$

where $Y_{i,j}$ is the inferred state of the mutation by our method in i-th sample at j-th genomic position.

Publication

The result will appear at the proceedings of ISMCO2019 (http://ismco.net/) in Springer LNCS format.

License