DriverDetector: an R package providing multiple statistical methods for cancer driver genes detection and tools for downstream analysis

Zeyuan Wang

2024-05

Introduction

This R package is used for identify significantly mutated genes which integrates 11 statistical methods. The main goal is to find a balance among the predicted gene number, the overlap with known drivers, and the consistency of prediction. The workflow includes 4 parts, which are, preprocessing, background mutation rate, driver genes identification, and statistical analysis. The software uses a MAF file, a coverage file, and a covariate file as input, and applies a voting strategy for driver prediction. For more information, please check our research paper.

Getting started

Installing the package.

To install the *DriverDetector* package, first download the installation package from site https://github.com/FrancisWang96/DriverDetector. For Windows, start R and select the `Packages` menu, then `Install package from local zip file`. Find and highlight the location of the zip file and click on `open`.

Loading the package.

To load the `DriverDetector` package in your R session, type `library(DriverDetector)`.

Run DriverDetector

To run `DriverDetector`, 3 input data sets and a directory of chromosome files are required. The first data set is mutation MAF(Mutation Annotation Format) of a paticular cancer type, containing information of mutations. The second is coverage data, containing information of coverages. The third data set is covariate data, which contains values of covariates, and is used for background mutation rate discovery. In addition, the chromosome files directory can be either hg19 or hg38. Here is a sample on how to run the software.

```
library(DriverDetector)

laml_maf <- system.file("extdata", "tcga_laml.maf", package = "DriverDetector")
coverage <- system.file("extdata", "coverage.rda", package = "DriverDetector")
covariate <- system.file("extdata", "gene.covariates.txt", package = "DriverDetector")
bmr_result <- backgroundMutationRate(laml_maf, coverage, covariate, original_mutation_rate = 1.2e-6,
                                     max_neighbors = 50, ref_genome = NULL,
                                     output_file = TRUE, quiet = FALSE)

driverDetector(bmr_result, p_class = "all", output_file = TRUE, filter = TRUE,
               sigThreshold = 0.1, quiet = FALSE)
```