AMARETTO: Multi-omics data fusion for cancer module discovery.

Olivier Gevaert, Magali Champion and Katie Planey

October 27, 2016

Stanford Center for Biomedical Informatics Research (BMIR)
Department of Medicine Department of Biomedical Data Science, Stanford University
1265 Welch Road
Stanford CA, 94305-5479

Contents

1 Introduction

The package **AMARETTO** contains functions to use the statistical algorithm AMARETTO, an algorithm to identify cancer drivers by integrating a variety of omics data from cancer and normal tissue. Due to the increasing availability of multi-omics data sets, there is a need for computational methods that integrate multi-omics data set and create knowledge. Especially in the field of cancer research, large international projects such as The Cancer Genome Atlas (TCGA) and the International Cancer Genome Consortium (ICGC) are producing large quantities of multi-omics data for each cancer site. However it remains unknown which profile is the most meaningful and how to efficiently integrate different omics profiles.

AMARETTO is an algorithm to unravel cancer drivers by reducing the data dimensionality into cancer modules. AMARETTO first models the effects of genomic/epigenomic data on disease specific gene expression. AMARETTO's second step involves constructing co-expressed modules to connect the cancer drivers with their downstream targets. We applied AMARETTO to several cancer sites of the TCGA project allowing to identify several cancer driver genes of interest, including novel genes in addition to known drivers of cancer. This package also includes functionality to access TCGA data directly so the user can immediately run AMARETTO on the most recent version of the data.

2 Getting started

Installing the package. To install the *AMARETTO* package, the easiest way is through bioconductor:

```
> source("http://bioconductor.org/biocLite.R")
> biocLite("AMARETTO")
```

Another way to install AMARETTO is to first download the appropriate file for your platform from the Bioconductor website http://www.bioconductor.org/. 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. For Linux/Unix, use the usual command `R CMD INSTALL` or install from bioconductor.

Loading the package. To load the *AMARETTO* package in your R session, type `library(AMARETTO)`.

Help files. Detailed information on *AMARETTO* package functions can be obtained in the help files. For example, to view the help file for the function `AMARETTO` in an R session, use `?AMARETTO`.

3 Data input

AMARETTO combines gene expression, DNA copy number and DNA methylation data into co-expressed gene expression models. Ideally, we recommend a cohort of at least 100 samples for each of these three technologies, where for most patients all data modalities have to be present.AMARETTO can be run with your own data but when interested in TCGA data, AMARETTO can also download TCGA data for you, see the next section.

3.1 Data access

The data in this vignette is accessible at The Cancer Genome Atlas (TCGA) portal. A programmatic way of downloading data is through the firehose_get tool developed by the broad institute ("https://confluence.broadinstitute.org/display/GDAC/Download"). Firehose_get provides a unified way to download data for all cancer sites and all platforms.

We have provided R functions that directly link with firehose get to download mRNA expression data and copy number data processed by GISTIC. Downloading TCGA data has been tested for twelve cancer sites:

Cancer code	Cancer site
BLCA	bladder urothelial carcinoma
BRCA	breast invasive carcinoma
CESC	cervical squamous cell carcinoma and endocervical adenocarcinoma
CHOL	cholangiocarcinoma
COAD	colon adenocarcinoma
ESCA	esophageal carcinoma
GBM	glioblastoma multiforme
HNSC	head and neck squamous cell carcinoma
KIRC	kidney renal clear cell carcinoma
KIRP	kidney renal papillary cell carcinoma
LAML	acute myeloid leukemia
LGG	brain lower grade glioma
LIHC	liver hepatocellular carcinoma
LUAD	lung adenocarcinoma
LUSC	lung squamous cell carcinoma
OV	ovarian serous cystadenocarcinoma
PAAD	pancreatic adenocarcinoma
PCPG	pheochromocytoma and paraganglioma
READ	rectum adenocarcinoma
SARC	sarcoma
STAD	stomach adenocarcinoma
THCA	thyroid carcinoma
THYM	thymoma
UCEC	endometrial carcinoma
COADREAD	colon cancer + rectal cancer

We also added COADREAD as a combination of colon and rectal cancer, as reports have shown that both can be seen as a single disease. The cancer code is needed to download data from TCGA and one needs to decide on a target location to save the data locally in the TargetDirectory, e.g. the /Downloads/ folder on a mac.

```
> TargetDirectory='./Downloads/'
> CancerCode='READ'
> DataSetDirectories=Download_CancerSite(CancerSite,TargetDirectory)
```

We recommend to use one TargetDirectory for all cancer sites, as this will save all data in one hierarchy is convenient when revisting results later on. The directory structure that is created will also include the data version history, so it is easy to report what version of the data is used. Download_CancerSite downloads the data, extracts archives and provides the paths to the downloaded folder for preprocessing. Download_CancerSite can also be run without actually downloading the data as follows:

```
> TargetDirectory='./Downloads/'
> CancerCode='READ'
> DataSetDirectories=Download_CancerSite(CancerSite,TargetDirectory,FALSE)
```

This is convenient when revisiting a data set because you want to redo-downstream analysis, but not the actual downloading. Running this way, will only set the data paths. The next step is preprocessing.

3.2   Data preprocessing

The data preprocessing step will take care of preprocessing the gene expression and DNA copy number data, see the next section for the DNA methylation data processing. Data preprocessing is done by Preprocess_CancerSite which takes the

CancerSite and the data set directories as parameters:

```
> CancerCode='READ'
> ProcessedData=Preprocess_CancerSite(CancerSite,DataSetDirectories)
```

This function preprocessed the gene expression data and the DNA copy number data. For the gene expression data, different preprocessing is done for microarray and RNA sequencing data. This involves missing value estimation using K-nearest neighbors. Also genes or patients that have more than 10% missing values are removed. Next, batch correction is done using the Combat method. For certain cancer sites, the gene expression data is split up in separate sub-data sets. This function first uses the preprocessing pipeline on each sub-data set separately and combines the data afterwards. For the DNA copy number data, the GISTIC algorithm output data is used. All genes that are in amplifications or deletions based on GISTIC output are extracted and the segmented DNA copy number data is stored. The segmented DNA copy number data is also batch corrected using Combat.

3.3   DNA methylation data

The DNA methylation data is not actually downloaded from TCGA. Due to the data set size of the cancer sites that have 450k data, downloading takes a long time. Similarly, DNA methylation data has to be run by **MethylMix** which is also computationally intensive and therefore we have chosen to provide add the MethylMix output to the AMARETTO package instead of processing the raw DNA methylation data. This functionality is available in the MethylMix package. The MethylStates data object has the methylation states from MethylMix for the 24 sites and the COADREAD combination.

```
> data(MethylStates)
> head(MethylStates$BLCA)
```

4   Running AMARETTO

In the case that you run AMARETTO with your own data, three data matrices are needed with preprocessed gene expression, DNA copy number and DNA methylation data, where genes are in the rows and patients are in the columns. Once you have your own data in this format or using a previously downloaded TCGA data set, you can start doing analysis with AMARETTO. First, we need to initialize the algorithm by clustering the gene expression data and creating the regulator data object. This is done by the AMARETTO_Initialize function and the TCGA LAML data set:

```
> data(ProcessedDataLAML)
> NrModules=10
> VarPercentage=5
> AMARETTOinit=AMARETTO_Initialize(ProcessedDataLAML$MA_TCGA,ProcessedDataLAML$CNV_TCGA,
+          ProcessedDataLAML$MET_TCGA,NrModules,VarPercentage)
```

Besides the three data sets, you need to decide how many modules to build and how much of the gene expression data is going to be used. For a first run we recommend 100 modules and to use the top 25% most varying genes. The AMARETTOinit object now contains cluster information to initialize an AMARETTO run and also stores the parameters that are required for AMARETTO.
Now we can run AMARETTO as follows:

```
> AMARETTOresults=AMARETTO_Run(AMARETTOinit)
```

```
Running AMARETTO on 50 genes and 173 samples.
        Stopping if less then 0.5 genes reassigned.
        Autoregulation is turned ON.
Elapsed time is 5.780000 seconds
Average nr of regulators per module: 6.400000
Elapsed time is 0.070000 seconds
Nr of reassignments is: 10
```

```
Elapsed time is 1.490000 seconds
Average nr of regulators per module: 7.666667
Elapsed time is 0.060000 seconds
Nr of reassignments is: 50
Elapsed time is 1.500000 seconds
Average nr of regulators per module: 7.777778
Elapsed time is 0.040000 seconds
Nr of reassignments is: 0
Elapsed time is 8.960000 seconds
```

This can take anywhere from 10 minutes up to 1 hour to build the modules for the TCGA cohorts depending on the number of genes that is modeled and the number of patients that is available. The breast cancer data set (BRCA) is the largest data set and will take the longest time to converge. AMARETTO will stop when less than 1% of the genes are reassigned to other modules. Next, one can test the AMARETTO model on the training set by calculating the Rsquare for each module using the AMARETTO_EvaluateTestSet function:

```
> AMARETTOtestReport=AMARETTO_EvaluateTestSet(AMARETTOresults,
+          AMARETTOinit$MA_TCGA_Var,AMARETTOinit$RegulatorData)
```

This function will use the training data to calculate the performance for predicting genes expression values based on the selected regulators. Of course, it is more interesting to use an independent test set. In this case only a gene expression data set is needed, for example from the GEO database. This will allow to check how well the AMARETTO modules are generalizing to new data. Take care that the an independent data set needs to be centered and scaled to unit variance. The AMARETTOtestReport will also give information of how many regulators and cluster members are actually present. The Rsquare performance has to be interpreted in this context as if many regulators are absent in the test data set due to platform limitations, the performance will be adversely affected.
Finally, modules can be visualized using the AMARETTO_VisualizeModule function:

```
> ModuleNr=1
```

```
> AMARETTO_VisualizeModule(AMARETTOinit,AMARETTOresults,ProcessedDataLAML$CNV,ProcessedDataLAML$MET,Module1
```

```
Module 1 has 6 genes and 10 regulators for 173 samples.
```

5   References

1. Gevaert O, Villalobos V, Sikic BI, Plevritis SK. Identification of ovarian cancer driver genes by using module network integration of multi-omics data. Interface Focus. 5(2) (2013)
2. Gevaert O, Tibshirani R, Plevritis SK. Pancancer analysis of DNA methylation-driven genes using MethylMix. Genome Biol. 2015 Jan 29;16(1):17.
3. Gevaert O. MethylMix: an R package for identifying DNA methylation-driven genes. Bioinformatics. 2015 Jan 20. pii: btv020.

6   Session Information

```
> toLatex(sessionInfo())
```

- R version 3.2.5 (2016-04-14), x86_64-w64-mingw32
- Locale: LC_COLLATE=French_France.1252, LC_CTYPE=French_France.1252, LC_MONETARY=French_France.1252, LC_NUMERIC=C, LC_TIME=French_France.1252
- Base packages: base, datasets, graphics, grDevices, grid, methods, parallel, stats, utils
- Other packages: AMARETTO 0.99.0, BiocStyle 1.8.0, bitops 1.0-6, circlize 0.3.9, ComplexHeatmap 1.6.0, doParallel 1.0.10, foreach 1.4.3, glmnet 2.0-5, impute 1.44.0, iterators 1.0.8, limma 3.26.9, Matrix 1.2-4, matrixStats 0.51.0, RColorBrewer 1.1-2, RCurl 1.95-4.8, stringr 1.1.0

- Loaded via a namespace (and not attached): class 7.3-14, cluster 2.0.3, codetools 0.2-15, colorspace 1.2-7, compiler 3.2.5, dendextend 1.3.0, DEoptimR 1.0-6, diptest 0.75-7, flexmix 2.3-13, fpc 2.1-10, GetoptLong 0.1.5, ggplot2 2.1.0, GlobalOptions 0.0.10, gtable 0.2.0, kernlab 0.9-25, lattice 0.20-33, magrittr 1.5, MASS 7.3-45, mclust 5.2, modeltools 0.2-21, munsell 0.4.3, mvtnorm 1.0-5, nnet 7.3-12, plyr 1.8.4, prabclus 2.2-6, Rcpp 0.12.7, rjson 0.2.15, robustbase 0.92-6, scales 0.4.0, shape 1.4.2, stats4 3.2.5, stringi 1.1.2, tools 3.2.5, trimcluster 0.1-2, whisker 0.3-2