REVEALER

(/cancer/software/genepattern/modules/docs)

REVEALER (*Repeated Evaluation of VariablEs conditionAL Entropy and Redundancy*), is an analysis method specifically suited to identifying groups of genomic alterations that together correlate with an independently acquired functional activation, gene dependency, or drug response profile.

Author: Pablo Tamayo

Contact: Pablo Tamayo (ptamayo@ucsd.edu)

Algorithm Version: REVEALER 1.0.0

Summary

REVEALER is an analysis method that enables the discovery of an ensemble of mutually exclusive genomic alterations correlated with “functional” phenotypes, e.g., a gene or pathway expression profile, a drug sensitivity or gene dependency profile. REVEALER can be used to identify complementary genomic alterations that together account for a fraction of the ”activated,” “sensitive” or “dependent” samples with respect to a functional target.

For more details on the method and specific examples of its use please see the original REVEALER article: *Kim and Botvinnik et al. 2016* in the References section below.

Background

The extensive characterization of the cancer genome provided by large-scale sequencing projects, such as the International Cancer Genome Consortium (ICGC) and the Cancer Genome Atlas (TCGA) have generated rich and diverse catalogs of somatic and epigenetic alterations contained in human tumors. However, despite the comprehensive and detailed nature of this molecular information, there is still a formidable challenge in identifying the functional role of genetic lesions in cancer. The identification of such relevant driver lesions is made more difficult by genomic instability that increases the number of mutations and copy number alterations and the abundance of low penetrant events with hard-to-elucidate functional roles.

REVEALER (Repeated Evaluation of VariablEs conditionAL Entropy and Redundancy) is an analysis method specifically suited to identify groups of genomic alterations that together correlate with an independently acquired functional activation, including also gene dependency, or drug response profiles. REVEALER can be applied to a wide

variety of problems as an exploratory tool that takes advantage of prior relevant background knowledge.

Methodology and Core Algorithm

The REVEALER article (see *Kim and Botvinnik et al. 2016* in References below) contains a very detailed technical description of the REVEALER algorithm, its examples, use cases and benchmarks. Here we will summarize the most salient characteristics of the method.

REVEALER takes as inputs a) a "target" representing activation such as gene up-regulation, pathway activation, gene-dependency or drug response for a group of individual samples across a given dataset, b) a "features" dataset containing genomic alterations, such as mutations and copy number alterations, for the same samples, and c) a "seed" feature that provides prior knowledge and a starting point for the search.

REVEALER makes use of information theoretical features of association such as the non-linear Information Coefficient (IC) and Conditional Information Coefficient (CIC) as described below. REVEALER implements a sequential conditional feature selection process where features that match the target profile but are correlated with the seed or summary feature are penalized, while features that associate with the target, and are also complementary to the seed, are scored higher. This is the way to select genomic features that effectively explain some amount of activation or sensitivity in the target profile that is not already accounted for. The algorithm can be summarized as follows:

Procedure: **REVEALER** (*t, s_0, F, max iter, n markers*)

Inputs: *t* = target, a continuous functional response of interest.

s_0 = starting seed, one or more binary features(s) representing known "causes" of activation or features associated with the target.

F = matrix of features*,* a collection of binary present/absent features, typically representing genome-wide alterations (mutations, amplifications/deletions.

max iter = maximum number of iterations to perform

n markers = top n hits to display at each iteration*.*

Step 1: Sort *t* in decreasing order (same order for s_0 and the columns of *F).*

Step 2: Pre-process F, optionally filter genomic features that are too infrequent or too frequent and/or consolidate genomic abnormalities, i.e. amplification or deletions that are identical or near identical up to a given Hamming distance threshold.

Step 3: *Set 1st* summary feature to the seed feature (s_o). Multiple seed features are reduced to one by combining (OR-ing) them.

Step 4: Iterate *max iter* times:

1. sort(*F*) by rows (x_i) in decreasing order according to the Conditional Information Coefficient $CIC(t, x_i \mid s_{k-1})$ between the target and each feature conditional to the seed or summary feature.
2. Identify top hit x_k and display the top *n markers* as a heat map.

3. Cluster and display pattern of *n markers* using Non-Negative Matrix factorization for several value of *k (*number of clusters) choose k using the cophenetic coefficient (see Supplementary Information).
4. Compute complementary *nominal p-values* and *False Discovery Rates* (FDR) using a permutation test on the target profile to generate a NULL dist.
5. Compute summary feature $s_k = OR(s_{k-1}, x_k)$.
6. Compute the *Information Coefficient IC(t, s_k)* between the new summary feature s_k and the target *t*.

Output*: n markers* at each iteration.

Final heat map summary of seed, top features and summary features at each iteration, and their Information Coefficient *IC(t, s_k)* association with the target.

At each iteration the permutation test and its associated p-values and FDR's are not used directly by REVEALER but provide a complementary assessment of significance.

REVEALER Information Coefficients. REVEALER's use of mutual and conditional mutual information has several advantages over the more traditional correlation coefficient and other feature selection methods (see Online methods in the Kim and Botvinnik et al. 2016 article for details). The differential Mutual Information, *I(t, s)*, is a function of the ratio of joint and marginal probabilities densities for target (*t*) and seed or summary feature (*s*),

$$I(t,s) = \iint P(t,s)\, log \frac{P(t,s)}{P(t)P(s)} dtds.$$

REVEALER rescales the mutual information making it easier to interpret and lie in the range [-1, 1], in analogy with the correlation coefficient, using the relationship between mutual information and correlation for Gaussian variables: $I(t, s) = -0.5 \log(1 - \rho^2(t, s))$. The Information Coefficient (*IC*) is defined as,

$$IC(t,s) = sign(\rho(t,s))\sqrt{(1 - \exp(-2I(t,s))}.$$

Where the sign of the correlation coefficient $\rho(t,s)$ adds directionality to the association metric. The relevant Information Coefficient in REVEALER is that between the target *t* and each summary feature *IC(t, s_k)*.

In order to perform an iteration REVEALER also requires the use of the conditional mutual information of the target (*t*), and a feature (*x*), conditional to the seed or summary feature (*s*),

$$I(t,x|s)) = \iiint P(t,x,s) log \frac{P(t,x|s))}{P(t|s)P(x|s))} dtdxds.$$

The corresponding Conditional Information Coefficient (CIC) is defined as follows,

$$CIC(t,x|s) = sign(\rho(t,x))\sqrt{1-\exp\left(-2(I(t,x|s)\right)}.$$

Estimating these quantities involves the computation of empirical probability density distributions for the relevant variables using kernel density estimation.

Running Considerations

Because REVEALER is somewhat different than other modules in GenePattern there are a number of running considerations that deserve some discussion because they are important to make optimal use of this method. Here we will briefly review those.

Target profile. The use of an effective target profile is a key requirement in order to make a REVEALER run successful. The main recommendation is to choose as target a functional readout that defines an "active" biological state of interest as it was done in the examples featured in the REVEALER article (*Kim and Botvinnik et al 2016*). In an ideal situation the target profile is a direct experimentally generated activation assay. In general, the target profile can be any measure of activation, dependency or sensitivity, such as a single gene transcriptional or protein expression profile, gene set enrichment scores, molecular signatures, drug sensitivity (IC50/AUC) or RNAi/CRISPR cell viability profiles.

Genomic features: Typically the features used in REVEALER are genomic abnormalities using the format: <gene_symbol>_MUT, <gene_symbol>_AMP and <gene_symbol>_DEL. However, other types of genomic features can also be used in principle. This includes finer- or coarser-grained binary indicators, such as specific locus or alleles of mutations or larger chromosomal regions such as GISTIC regions. The features can also be other types of structural genome-wide alterations such as gene fusions, promoter or enhancer alterations, methylation marks, epigenetic features, etc.

Seed choice. REVEALER implements a sequential conditional feature selection process and its results are influenced by the choice of seed. The seed allows the use of prior knowledge and therefore it is preferable if it represents an already established association with the target. The seeds are typically mutations, deletions or amplifications but other types of features can also be used as we described above. REVEALER can also run without a seed ("NULLSEED") e.g. if none is known or if one wants to perform the analysis without the influence of any prior knowledge.

Directionality of match: REVEALER can match features using a positive or negative direction. Positive means that the presence of the features, e.g. the mutations and copy number alterations, are matched against the higher values of the target. This is the case where the target is an activation profile, such as a transcriptional signature or gene or protein expression profile where higher values delineate the "active" biological state. Negative is useful when the matching is against the lower values of the target. This is the case, e.g. when the target is a drug sensitivity (IC50/AUC), gene dependency/RNAi/CRISPR or cell viability profile where lower values i.e. increased sensitivity or lower cell viability, delineate the biological state of interest.

Feature frequency filtering thresholds: REVEALER allows the filtering of features that are either too infrequent or too frequent using the parameters *count thres low* and *count thres high.* Filtering very infrequent alterations will speed up the code; however, one can also unintentionally eliminate low frequency biological meaningful features. Filtering very frequent alterations will also speed up the code; however one can also eliminate high penetrant biological meaningful features.

Alternative feature hits and clustering. At each iteration REVEALER chooses and displays the best matching features based on the value of their CIC scores. However, in many cases it may be useful to analyze other potential hits, besides the top one, because of their biological importance. This can easily be done by looking at the top *n markers* hits for each iteration in the output PDF file. Sometimes the top scoring features display interesting patterns. To allow a better analysis of those patterns REVEALER also shows each iteration results where the *n markers* hits have been clustered using Non-Negative Matrix Factorization. REVEALER determines the optimal value of groups to display using the cophenetic coefficient.

REVEALER results. REVEALER generates a PDF containing multiple pages of results including: the top *n marker* hits at each iteration, the clustering of those results and at the end the final results in two modalities: including the summary feature after each iteration and omitting it. REVEALER runs and performs the number of iterations specified by the *max n iter* parameter.

REVEALER exploration of parameters and number of iterations. REVEALER is an exploratory tool and therefore the user may need to make a few preliminary runs, e.g. with different choices of frequency filters or candidate targets or seeds in order to assess their performance and identify preliminary findings. It is also important to inspect the REVEALER output and determine how many iterations actually made the IC between the target and the summary feature increase. The corresponding features to those iterations are the ones that should be considered as the final output results (see e.g. the 4 examples in the *Kim and Botvinnik et al. 2016* article). After making a number of exploratory runs the user can make a final run to produce this final result set and corresponding heatmaps with only the desired parameters and showing only the relevant features (i.e. the ones that made the IC increase).

Parameters

Name	Description
ds1	Target profile dataset in GCT format. This file should contain the "target name" as one of the row entries.
target name	Target profile name. This should be one of the rows contained in the ds1 input file.
target match	The direction to match features: Positive or negative.
ds2	Features dataset in GCT format.
seed names	Single seed name, multiple seeds as an R vector e.g. c("seed1", seed2"...) or "NULLSEED" to run without seed.
max n iter	Maximum number of iterations. This number can be initially e.g. in the range 2-5 and be increased if the IC of the summary feature

	and the target keeps increasing up to the last iteration.
n markers	Number of top hits to show in heatmap at every iteration. Default is 30.
locs table file	An (optional) annotation table with chromosomal locations for each gene symbol or NULL
count thres low	Filter out features with frequency counts less than this threshold (default is 3). Filtering very infrequent alterations will speed up the code; however be careful as you can also eliminate a low frequency meaningful feature.
count thres high	Filter out features with frequency counts above this threshold (default is 50). Filtering very frequent alterations will speed up the code; however be careful as you can also eliminate high penetrant meaningful features.
pdf output file	Output PDF file with results (see output files below).

Input Files

1. ds1: input file GCT-formatted file containing the target profile of interest.
2. ds2: input file GCT-formatted file containing the genomic alteration features.
3. locs table file: (optional) tab-separated ASCII file with chromosomal location for each gene symbol or NULL

Output Files

1. Pdf output file. This file displays the results at each REVEALER iteration including the direct *n marker* hits, and the results of clustering them, and also the final results at the end.

Example Data

The default parameters for REVEALER in its GenePattern page are set to run the first example from the article (Activation of beta-catenin) loading the corresponding input files from specially prepared URLs.

In the supplementary material of *Kim et al. 2016* (see References above) there are input datasets that correspond to the examples shown in the paper. These datasets are:

- CTNBB1_transcriptional_reporter.gct (Example 1 target dataset, the target name is "BETA-CATENIN-REPORTER").
- NRF2_activation_profile.gct (Example 2 target dataset, the target names is "NFE2L2.V2").
- MEK_inhibitor_profile.gct (Example 3 target dataset, the target name is "PD-0325901").
- KRAS_essentiality_profile.gct (Example 4 target dataset, the target name is

"KRAS_essentiality_profile").

- hgnc_complete_set.Feb_20_2014.v2.txt (text file with chromosomal location for each gene symbol). This file is optional and it can be set to NULL.

The features dataset for the 4 examples used in the article is available in the Browse/Data section of the CCLE web site www.broadinstitute.org/ccle, as "binary calls for copy number and mutation data" (file: CCLE_MUT_CNA_AMP_DEL_0.70_2fold.MC.gct).

Details on how these datasets were prepared and other considerations when running the algorithm are available in the "*Datasets and Pre-processing"* and "*Running REVEALER: relevant factors to consider*" sections on the Online Methods of Kim and Botvinnik et al. 2016.

The user can provide other datasets as input to REVEALER if they have the right format (GCT files).

References

1. Kim et al. *Characterizing genomic alterations in cancer by complementary functional associations*. Nature Biotechnology April 18, 2016.
2. Linfoot, E.H. An informational measure of correlation. Information and Control 1, 85-89 (1957).
3. Joe, H. Relative Entropy Measures of Multivariate Dependence. Journal of the American Statistical Association 84, 157-164 (1989).
4. Kraskov, A., Stogbauer, H. & Grassberger, P. Estimating mutual information. Phys Rev E Stat Nonlin Soft Matter Phys 69, 066138 (2004).
5. Cover, T.M. & Thomas, J.A. Elements of information theory, Edn. 2. (John Wiley & Sons, 2012).

Requirements

REVEALER requires R3.1.0. The GenePattern team has confirmed test data reproducibility for this module using R3.1.0.

Platform Dependencies

Task Type:

Feature Selection (sequential conditional feature selection)

CPU Type:

any

Operating System:

any

Language:

R3.1.0

Version Comments