Mirsynergy: detect synergistic miRNA regulatory modules by overlapping neighbourhood expansion

Yue Li
yueli@cs.toronto.edu

July 15, 2014

1 Introduction

MicroRNAs (miRNAs) are $\sim$22 nucleotide small noncoding RNA that base-pair with mRNA primarily at the $3'$ untranslated region (UTR) to cause mRNA degradation or translational repression [1]. Aberrant miRNA expression is implicated in tumorigenesis [4]. Construction of microRNA regulatory modules (MiRM) will aid deciphering aberrant transcriptional regulatory network in cancer but is computationally challenging. Existing methods are stochastic or require a fixed number of regulatory modules. We propose *Mirsynergy*, a deterministic overlapping clustering algorithm adapted from a recently developed framework. Briefly, Mirsynergy operates in two stages that first forms MiRM based on co-occurring miRNAs and then expand the MiRM by greedily including (excluding) mRNA into (from) the MiRM to maximize the synergy score, which is a function of miRNA-mRNA and gene-gene interactions (manuscript in prep).

2 Demonstration

In the following example, we first simulate 20 mRNA and 20 mRNA and the interactions among them, and then apply `mirsynergy` to the simulated data to produce module assignments. We then visualize the module assignments in Fig.1

```
> library(Mirsynergy)
> load(system.file("extdata/toy_modules.RData", package="Mirsynergy"))
> # run mirsynergy clustering
> V <- mirsynergy(W, H, verbose=FALSE)
> summary_modules(V)

$moduleSummaryInfo
  miRNA mRNA total   synergy    density
1     4    4    12 0.1680051 0.04426190
2     2    2     6 0.1654560 0.09630038
3     6   10    22 0.1870070 0.02471431
```

```
4      8     7    23 0.1821842 0.02318249
5      2     3     7 0.1640842 0.08457176
6      3     4    10 0.1602223 0.04856618

$miRNA.internal
  modules miRNA
1       2     2
2       1     3
3       1     4
4       1     6
5       1     8

$mRNA.internal
  modules mRNA
1       1    2
2       1    3
3       2    4
4       1    7
5       1   10
```

Additionally, we can also export the module assignments in a Cytoscape-friendly format as two separate files containing the edges and nodes using the function `tabular_module` (see function manual for details).

3 Real test

In this section, we demonstrate the real utility of *Mirsynergy* in construct miRNA regulatory modules from real breast cancer tumor samples. Specifically, we downloaded the test data in the units of RPKM (read per kilobase of exon per million mapped reads) and RPM (reads per million miRNA mapped) of 13306 mRNA and 710 miRNA for the 15 individuals from TCGA (The Cancer Genome Atlas). We furhter log2-transformed and mean-centred the data. For demonstration purpose, we used 20% of the expression data containing 2661 mRNA and 142 miRNA expression. Moreover, the corresponding sequence-based miRNA-target site matrix $\mathbf{W}$ was downloaded from TargetScanHuman 6.2 database [3] and the gene-gene interaction (GGI) data matrix $\mathbf{H}$ including transcription factor binding sites (TFBS) and protein-protein interaction (PPI) data were processed from TRANSFAC [6] and BioGrid [5], respectively.

```
> load(system.file("extdata/tcga_brca_testdata.RData", package="Mirsynergy"
```

Given as input the 2661×15 mRNA and 142×15 miRNA expression matrix along with the 2661×142 target site matrix, we first construct an expression-based miRNA-mRNA interaction score (MMIS) matrix using LASSO from *glmnet* by treating mRNA as response and miRNA as input variables [2].

```
> load(system.file("extdata/toy_modules.RData", package="Mirsynergy"))
> plot_modules(V,W,H)
```

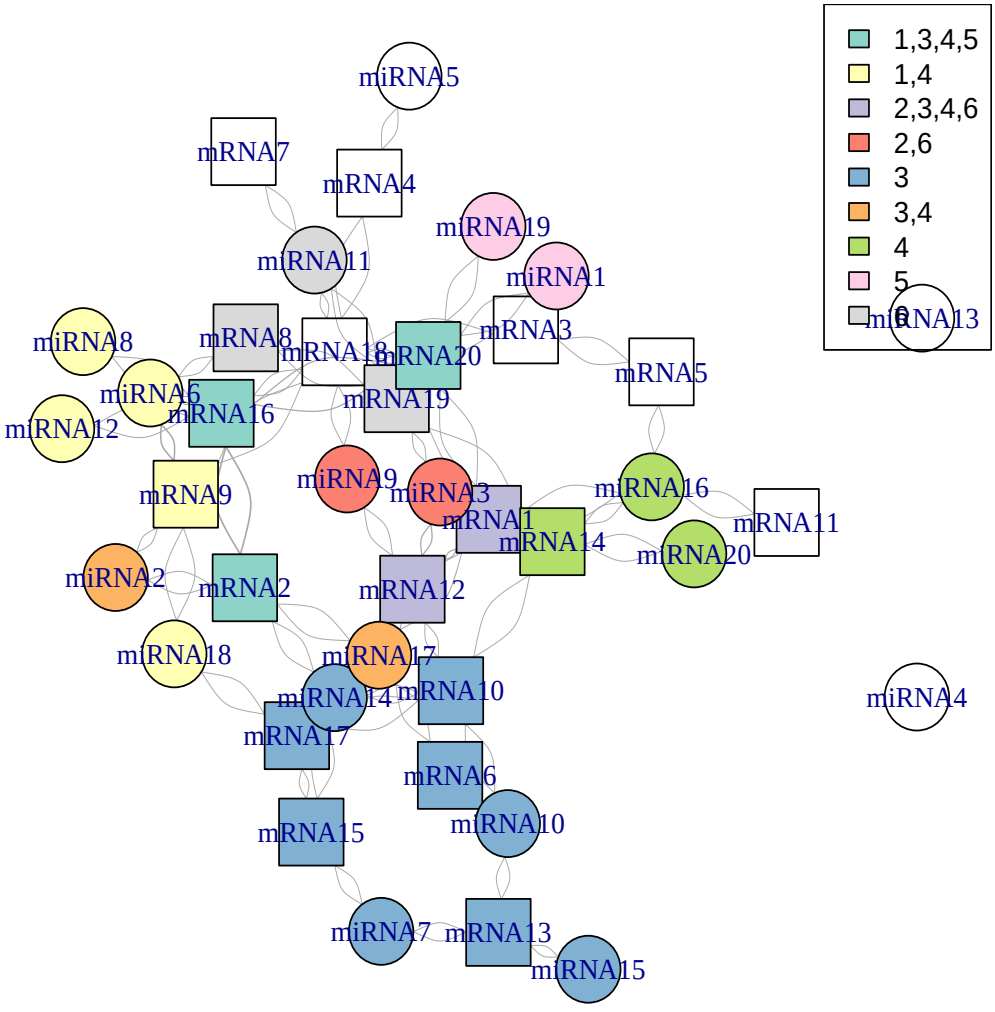

Figure 1: Module assignment on a toy example.

```
> library(glmnet)
> ptm <- proc.time()
> # lasso across all samples
> # X: N x T (input variables)
> #
> obs <- t(Z)  # T x M
> # run LASSO to construct W
> W <- lapply(1:nrow(X), function(i) {
+
+         pred <- matrix(rep(0, nrow(Z)), nrow=1,
+                 dimnames=list(rownames(X)[i], rownames(Z)))
+
+         c_i <- t(matrix(rep(C[i,,drop=FALSE], nrow(obs)), ncol=nrow(obs))
+
+         c_i <- (c_i > 0) + 0 # convert to binary matrix
+
+         inp <- obs * c_i
+
+         # use only miRNA with at least one non-zero entry across T sample
+         inp <- inp[, apply(abs(inp), 2, max)>0, drop=FALSE]
+
+         if(ncol(inp) >= 2) {
+
+                 # NOTE: negative coef means potential parget (remove inte
+                 x <- coef(cv.glmnet(inp, X[i,], nfolds=3), s="lambda.min"
+
+                 pred[, match(colnames(inp), colnames(pred))] <- x
+         }
+         pred[pred>0] <- 0
+
+         pred <- abs(pred)
+
+         pred[pred>1] <- 1
+
+         pred
+ })
> W <- do.call("rbind", W)
> dimnames(W) <- dimnames(C)
> print(sprintf("Time elapsed for LASSO: %.3f (min)",
+         (proc.time() - ptm)[3]/60))

[1] "Time elapsed for LASSO: 1.005 (min)"
```

Given the $\mathbf{W}$ and $\mathbf{H}$, we can now apply `mirsynergy` to obtain MiRM assignments.

```
> V <- mirsynergy(W, H, verbose=FALSE)
> print_modules2(V)

M1 (density=2.53e-02; synergy=2.3e-01):
hsa-miR-302a hsa-miR-3183 hsa-miR-495 hsa-miR-519d hsa-miR-302e hsa-miR-431
RASD2 CLP1 ZC3HAV1L NSF AIF1L TSEN34 GFOD2 FBXO41 MYCN SLC2A4 ZBTB41 TRPV6 I
M2 (density=2.58e-02; synergy=1.49e-01):
hsa-miR-548n hsa-miR-541 hsa-miR-1229 hsa-miR-921 hsa-miR-33a hsa-miR-3689b
ZNF423 EBF1 EMILIN3 PCDH7 EPHA8 ZNF746
M3 (density=4.15e-02; synergy=1.85e-01):
hsa-miR-4328 hsa-miR-605 hsa-miR-935 hsa-miR-517b
RAI14 POLD3 LMO4 ITSN1 PAPD7 CHMP4A ISL1 CHMP4C DEPDC1 F2R NUP210
M4 (density=4.18e-02; synergy=1.97e-01):
hsa-miR-4311 hsa-miR-424 hsa-miR-1193 hsa-miR-759 hsa-miR-601
WDR43 SEH1L NUAK1 FAM60A PCDHA7 TAF7L PCDHA6
M5 (density=2.98e-02; synergy=1.97e-01):
hsa-miR-626 hsa-miR-621 hsa-miR-122 hsa-miR-3658 hsa-miR-137 hsa-miR-762
PCNT FREM2 FBXO31 FAM84A CTPS EPHB4 KCNN4 CCDC25 SEMA3B MDGA2
M6 (density=3.89e-02; synergy=1.88e-01):
hsa-miR-98 hsa-miR-4284 hsa-miR-1227 hsa-miR-1261
TBX5 FOXM1 NID2 ATP7B TGIF2 YIPF1 DUSP4 SLC2A12 C5orf62
M7 (density=4.06e-02; synergy=1.83e-01):
hsa-miR-4308 hsa-miR-340 hsa-miR-552 hsa-miR-3161
GIPC2 VPS37B ACADSB C18orf1 PALLD FGF1 SYNM SYT1
M8 (density=3.49e-02; synergy=2.03e-01):
hsa-miR-216a hsa-miR-4262 hsa-miR-181d hsa-miR-3941 hsa-miR-147
RAB27B WDFY3 RAB35 L1CAM TBPL1 ATG16L1 HYOU1 CELSR3 PCDHA7 CNTN2 ABTB2 PI4K2
M9 (density=8.23e-02; synergy=1.51e-01):
hsa-miR-891b hsa-miR-1322
CBFB ZNF644 CSDE1 RUNX1
M10 (density=9.1e-02; synergy=1.84e-01):
hsa-miR-586 hsa-miR-595
ZFP1 ADAT2 BNIP2
M11 (density=5.51e-02; synergy=2.05e-01):
hsa-miR-3183 hsa-miR-495 hsa-miR-519d hsa-miR-4316
RASD2 ZC3HAV1L AIF1L GFOD2 LPAR3 GABBR2 RFX4
M12 (density=5.66e-02; synergy=1.91e-01):
hsa-miR-1912 hsa-miR-764 hsa-miR-555
XPO5 ERC2 IPO9 SASS6
M13 (density=1.07e-01; synergy=2.31e-01):
hsa-miR-519e hsa-miR-494
GDAP1 GJC1 PNOC
M14 (density=4.01e-02; synergy=1.51e-01):
hsa-miR-1915 hsa-miR-1254 hsa-let-7d
ETNK2 RNF170 TRHDE GABRA6 UBFD1 EGR3 KIAA1467
```

```
M15 (density=3.45e-02; synergy=2.18e-01):
hsa-miR-513b hsa-miR-181c hsa-miR-1297 hsa-miR-143 hsa-miR-544b
C6orf170 GPR126 PTGS2 AGPAT5 BOLL CD163 PLEK HSPH1 CMTM7 ABCA13 KCNJ10 NUPL
M16 (density=2.74e-02; synergy=1.65e-01):
hsa-miR-548y hsa-miR-181c hsa-miR-891b hsa-miR-1322 hsa-miR-3135 hsa-miR-14
DOCK2 CBFB ZNF644 CD163 GALK2 PLEK CSDE1 KCNJ10
M17 (density=7.24e-02; synergy=1.39e-01):
hsa-miR-185 hsa-miR-4276
HAUS5 SYNGAP1
M18 (density=8.4e-02; synergy=1.45e-01):
hsa-miR-608 hsa-miR-4293
ARL4D NDRG1 PKM2
M19 (density=8.07e-02; synergy=2.04e-01):
hsa-miR-377 hsa-miR-448
PPM1L YEATS2 RNGTT EPHA8 ZNF746 MAP3K7
M20 (density=2.37e-02; synergy=1.84e-01):
hsa-miR-320e hsa-miR-216a hsa-miR-18b hsa-miR-4262 hsa-miR-181d hsa-miR-130
RAB27B WDFY3 RAB35 L1CAM TBPL1 ATG16L1 HYOU1 CELSR3 PCDHA7 CNTN2 ABTB2 PI4K2
M21 (density=7.82e-02; synergy=1.41e-01):
hsa-miR-665 hsa-miR-661
CHMP4A CHMP4C F2R
```

```
> print(sprintf("Time elapsed (LASSO+Mirsynergy): %.3f (min)",
+   (proc.time() - ptm)[3]/60))

[1] "Time elapsed (LASSO+Mirsynergy): 1.068 (min)"
```

There are several convenience functions implemented in the package to generate summary information such as Fig.2. In particular, the plot depicts the m/miRNA distribution across modules (upper panels) as well as the synergy distribution by itself and as a function of the number of miRNA (bottom panels).

For more details, please refer to our paper (manuscript in prep.).

4 Session Info

```
> sessionInfo()

R version 3.1.1 (2014-07-10)
Platform: x86_64-apple-darwin13.1.0 (64-bit)

locale:
[1] C/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base
```

```
> plot_module_summary(V)
```

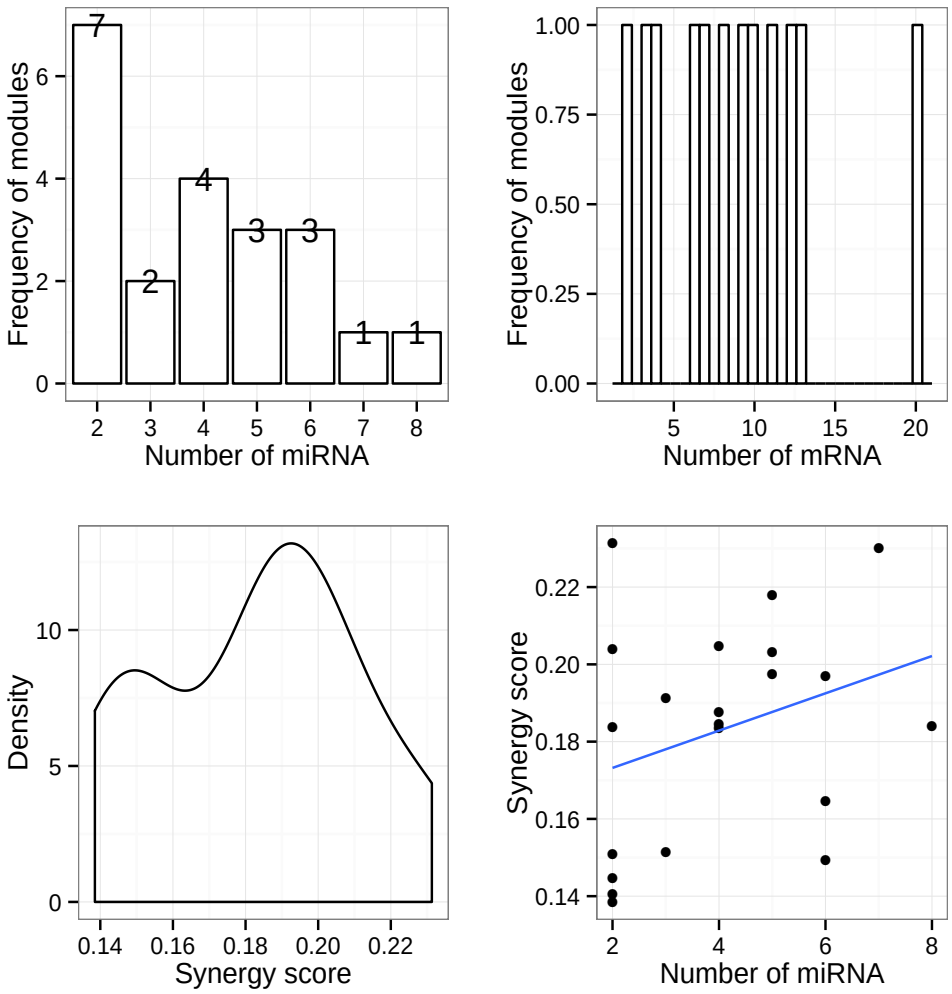

Figure 2: Summary information on MiRM using test data from TCGA-BRCA. Top panels: m/miRNA distribution across modulesas; Bottom panels: the synergy distribution by itself and as a function of the number of miRNA.

```
other attached packages:
[1] glmnet_1.9-8     Matrix_1.1-4     Mirsynergy_1.0.1 ggplot2_1.0.0
[5] igraph_0.7.1

loaded via a namespace (and not attached):
 [1] MASS_7.3-33        RColorBrewer_1.0-5 Rcpp_0.11.2        colorspace_1.2
 [5] digest_0.6.4       evaluate_0.5.5     formatR_0.10       grid_3.1.1
 [9] gridExtra_0.9.1    gtable_0.1.2       knitr_1.6          labeling_0.2
[13] lattice_0.20-29    munsell_0.4.2      parallel_3.1.1     plyr_1.8.1
[17] proto_0.3-10       reshape_0.8.5      reshape2_1.4       scales_0.2.4
[21] stringr_0.6.2      tools_3.1.1
```

References

[1] David P Bartel. MicroRNAs: Target Recognition and Regulatory Functions. *Cell*, 136(2):215–233, January 2009.

[2] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of statistical software*, 33(1):1–22, 2010.

[3] Robin C Friedman, Kyle Kai-How Farh, Christopher B Burge, and David P Bartel. Most mammalian mRNAs are conserved targets of microRNAs. *Genome Research*, 19(1):92–105, January 2009.

[4] Riccardo Spizzo, Milena S Nicoloso, Carlo M Croce, and George A Calin. SnapShot: MicroRNAs in Cancer. *Cell*, 137(3):586–586.e1, May 2009.

[5] Chris Stark, Bobby-Joe Breitkreutz, Andrew Chatr-Aryamontri, Lorrie Boucher, Rose Oughtred, Michael S Livstone, Julie Nixon, Kimberly Van Auken, Xiaodong Wang, Xiaoqi Shi, Teresa Reguly, Jennifer M Rust, Andrew Winter, Kara Dolinski, and Mike Tyers. The BioGRID Interaction Database: 2011 update. *Nucleic acids research*, 39(Database issue):D698–704, January 2011.

[6] E Wingender, X Chen, R Hehl, H Karas, I Liebich, V Matys, T Meinhardt, M Prüss, I Reuter, and F Schacherer. TRANSFAC: an integrated system for gene expression regulation. *Nucleic acids research*, 28(1):316–319, January 2000.