

# *RIFA: A Differential Gene Connectivity Algorithm*

**Todd D. Allen**

This article describes the implementation of RIFA, a computational biology algorithm designed to identify the genes most directly responsible for creating differences in phenotype between two biological states, for example, tumorous liver tissue versus cirrhotic liver tissue.

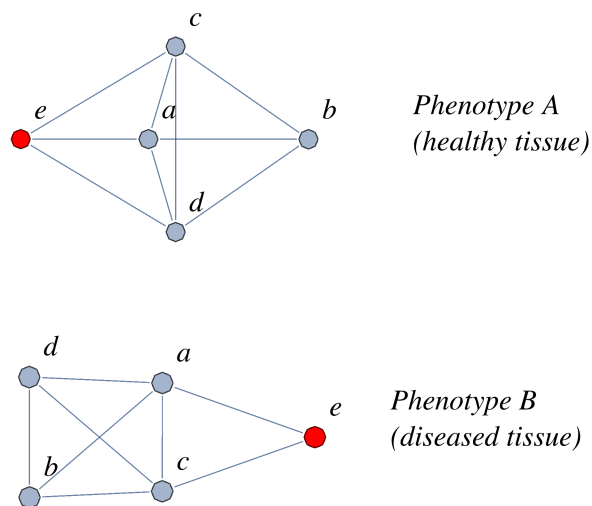
## ■ Introduction and Background

With the invention of microarray technology, scientists finally had a means to measure global changes in gene expression between two biological states [1]. This has led to thousands of scientific publications describing long lists of differentially expressed genes in each scientist's favorite experimental system. What has gradually become apparent to biologists is that having a list of differentially expressed genes, while an important first step in understanding the differences between two phenotypes (where phenotype represents the physical manifestation of one or more traits), is often not enough to identify the genes most directly responsible for driving changes in phenotype. While it is true that genes that are differentially expressed between two biological states may be important in explaining those differences, it is also possible that genes whose expression is *not* changed can also be pivotal in driving phenotypic differences.

For those unfamiliar with biology, a rough analogy may prove useful. Consider a manufacturing setting where there is a supervisor (a “boss” gene) and employees (“slave” genes) under the supervisor's responsibility charged with manufacturing widgets (a particular phenotype, such as healthy liver tissue). A change in phenotype, such as transitioning from healthy liver tissue (manufacturing blue widgets) to cancerous liver tissue (manufacturing red widgets) can be accomplished by: (1) changing the rate that employees work (such as might happen if a supervisor shouts at the employees; this is analogous to differential expression); and/or (2) changing the instructions the supervisor is giving to employees (keeping the volume of instructions constant, but changing the information contained in the instructions; this is analogous to a mutation in the “boss” gene); and/or (3) a combination of scenarios (1) and (2). In scenario (1), there is a transition in phenotype because the employees (“slave” genes) begin working faster or slower than they have previously, which produces too many or too few gene products at the wrong

time, creating a rippling effect throughout all of the manufacturing, which ends up in a different product (the red widget phenotype) being made. In this situation, the supervisor's instructions to the employees remain constant (manufacture blue widgets) but are spoken with more ("shouting") or less ("whispering") volume. Scenario (1) reflects the kind of information that can be measured in microarray studies, whose sole purpose is to identify genes whose expression changes between two biological states. In scenario (2), the rate at which employees work remains constant, but they still manufacture a different-colored widget (phenotype), because the instructions they are receiving from their supervisor have changed. Differences in phenotype due to scenario (2) are rarely discovered by producing long lists of differentially expressed genes, because the primary driving force creating a change in phenotype is a change in instruction from the supervisor (such as a mutation in the "boss" gene) to the employees ("slave" genes), not a difference in the manufacturing rate of employees.

For these reasons, computational biologists have begun to develop algorithms that are better at highlighting those genes primarily responsible for driving changes in phenotype, regardless of whether those genes are differentially expressed or not. This is the purpose of the regulatory impact factor analysis (RIFA) algorithm presented here; that is, to highlight those genes most directly responsible for driving changes in phenotype. RIFA provides a computationally tractable way to detect differences in connectivity between genes in two biological states. Figure 1 illustrates the basic premise of connectivity and differences in connectivity between two biological states.



▲ **Figure 1.** Two gene networks comprised of the same five genes (a through e) in two different biological states (phenotypes). Each vertex represents a gene, and each edge represents a connection between genes. In standard differential expression studies, each gene's expression level is compared to itself between the two biological states but ignores potential relationships between different genes. When even a casual observer compares the two networks above, it is immediately noticeable that the shape of each network is different, a difference driven by a change in connectedness between genes within each biological state.

Regulatory impact factor analysis (RIFA) is based on seminal work by Hudson, Reverter, and Dalrymple [2], which introduced three higher-order metrics all computed from basic information obtained in differential gene expression studies. The purpose of these metrics is to use the information present in gene expression studies to quantify the connectedness between differentially expressed genes (“slave” genes, using the analogy above) and a group of gene expression regulator genes, known as transcription factors (“boss” genes, using the analogy above). The three metrics are:

$$\text{Phenotype Impact Factor (PIF)} = \frac{1}{2} (E_{i,A} + E_{i,B}) dE_i \quad (1)$$

$$\text{Regulatory Impact Factor 4 (RIF4)} = \frac{1}{n_{dE}} \sum_{j=1}^{n_{dE}} (\text{PIF}_j dC_{i,j}^2) \quad (2)$$

$$\text{Regulatory Impact Factor 5 (RIF5)} = \frac{1}{n_{dE}} \sum_{j=1}^{n_{dE}} [(E_{j,A} r_A(i, j))^2 - (E_{j,B} r_B(i, j))^2] \quad (3)$$

Equation (1) (PIF) computes the average expression of the  $i^{\text{th}}$  gene between two biological states (A and B) and multiplies that result by the differential expression of the  $i^{\text{th}}$  gene between states A and B. In doing so, the magnitude of the differential expression of a gene is weighted by the overall expression level of the gene. PIF is then used to compute equation (2) (RIF4), which multiplies the PIF value for each differentially expressed gene by the differential co-expression (calculated using the Spearman correlation coefficient) between each differentially expressed gene (the “slave” genes in our analogy above) and each transcriptional regulator (the “boss” genes in our analogy above) between states A and B. By summing these calculations over each differentially expressed gene, a prioritized list of the most important regulators driving changes in phenotype between states A and B can be obtained. Equation (2) is designed to provide an answer to the question, which regulator is consistently highly differentially co-expressed with the most abundant differentially expressed genes? Equation (3) (RIF5) is an alternative metric to equation (2) (RIF4), which also seeks to produce a prioritized list of the most important regulators driving phenotypic change. By multiplying the expression of each differentially expressed gene by the correlation between itself and each transcription regulator twice, once in state A and once in state B, the difference in state values can be computed and then summed over each differentially expressed gene to yield an alternative prioritized list of the most important regulators. Equation (3) is designed to answer the question, which regulator has the most altered ability to predict the abundance of differentially expressed genes? Further details of these equations are presented in [3-4], but the basic idea behind the use of these metrics in RIFA is straightforward. When gene expression data (from a well-thought-out experiment) is presented to RIFA, the algorithm can use the “echoes of sound off structures” (differential gene expression data) to triangulate the location of the “rifle shot creating the sound” (identify the master gene(s) driving the changes in phenotype).

## ■ The Regulatory Impact Factor Analysis (RIFA) Algorithm

RIFA is template driven, meaning the algorithm expects several pieces of user-defined information to be provided in a notebook cell that is used as a template for entering information. As RIFA was designed to process output from AffyDGED [5], it will be assumed the reader is familiar with AffyDGED as well. The features of RIFA are illustrated using data from a microarray study comparing gene expression profiles of tumorous liver tissue to cirrhotic liver tissue [6]. All microarray data used in this study and presented here is publicly available at NCBI's Gene Expression Omnibus portal ([www.ncbi.nlm.nih.gov/geo](http://www.ncbi.nlm.nih.gov/geo)), using the access number GSE17548.

```
Needs["JLink`"]
ReinstallJava[JVMArguments -> "-Xmx512m"] ;

timecoursedata =
  Flatten[
    Import[
      "C:\\Users\\Wookie\\Desktop\\Mathematica
      Projects\\Mathematica Journal
      Projects\\Differential Wiring & RIF
      development\\Rif validation - Yildiz -
      liver\\liver all expression data for rifa.xls"],
    1];

conditionecol = {2, 7};
conditiontwocol = {8, 13};

rawtranscriptionreg =
  Flatten[
    Import[
      "C:\\Users\\Wookie\\Desktop\\Mathematica
      Projects\\Mathematica Journal
      Projects\\Differential Wiring & RIF
      development\\Rif validation - Yildiz -
      liver\\hgplus2 trfactors.xls"]];

rawdeggenes =
  Flatten[
    Import[
      "C:\\Users\\Wookie\\Desktop\\Mathematica
      Projects\\Mathematica Journal
      Projects\\Differential Wiring & RIF
      development\\Rif validation - Yildiz -
      liver\\liver de genes for rifa.xls"]];
```

```

affyginlocation =
  Import [
    "C:\\Users\\Wookie\\Desktop\\Mathematica
      Projects\\Mathematica Journal
      Projects\\Data\\AffyChip Description
      Files\\HG-U133_Plus_2\\LibFiles\\HG-U133_Plus_2.gin"];

savelocationroot = "C:\\Users\\Wookie\\Desktop\\";

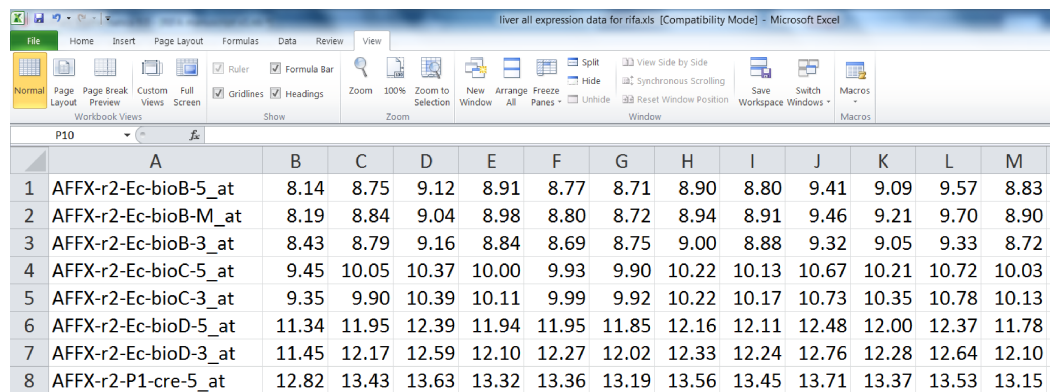
studyname = livercirrandcancer;

```

The template cell begins with a command to purposefully reinstall Java, for the express purpose of expanding the memory available to import large datasets into *Mathematica*.

The remainder of the template cell defines several variables requiring user input.

1. **timecoursedata**: This variable points to the directory containing the microarray gene expression data, in spreadsheet format, to be processed by RIFA. While this variable uses the term "timecourse" as part of its name, it is not necessary for the microarray data to be part of a time course experiment. The spreadsheet format of the data is non-negotiable and requires strict organization. To aid in instruction, a screen shot of the **timecoursedata** that will be described throughout this paper is included here (Figure 2).



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	AFFX-r2-Ec-bioB-5_at	8.14	8.75	9.12	8.91	8.77	8.71	8.90	8.80	9.41	9.09	9.57	8.83
2	AFFX-r2-Ec-bioB-M_at	8.19	8.84	9.04	8.98	8.80	8.72	8.94	8.91	9.46	9.21	9.70	8.90
3	AFFX-r2-Ec-bioB-3_at	8.43	8.79	9.16	8.84	8.69	8.75	9.00	8.88	9.32	9.05	9.33	8.72
4	AFFX-r2-Ec-bioC-5_at	9.45	10.05	10.37	10.00	9.93	9.90	10.22	10.13	10.67	10.21	10.72	10.03
5	AFFX-r2-Ec-bioC-3_at	9.35	9.90	10.39	10.11	9.99	9.92	10.22	10.17	10.73	10.35	10.78	10.13
6	AFFX-r2-Ec-bioD-5_at	11.34	11.95	12.39	11.94	11.95	11.85	12.16	12.11	12.48	12.00	12.37	11.78
7	AFFX-r2-Ec-bioD-3_at	11.45	12.17	12.59	12.10	12.27	12.02	12.33	12.24	12.76	12.28	12.64	12.10
8	AFFX-r2-P1-cre-5_at	12.82	13.43	13.63	13.32	13.36	13.19	13.56	13.45	13.71	13.37	13.53	13.15

▲ **Figure 2.** Formatting of gene expression data for variable <timecoursedata>.

Column A contains unique transcript identification information from the microarray chip used in the study. Columns B through X contain gene expression measurement from samples (or time points) under the *same experimental condition* of the study. The columns after X contain gene expression measurements from samples (or time points) under the *same control condition* of the study. For example, in the liver study referenced above, tumor samples from multiple patients were randomly placed into six groups and compared to six groups of cirrhotic liver tissue by AffyDGED. Column B contains the gene expression measurements (transcript abundance, not differential expression) for the first group of tumor samples processed with AffyDGED, column C contains the gene

expression measurements from the second group of tumor samples, and so on. In this example, the last column containing tumor (experimental condition) data is column G. Column H is the first column containing gene expression measurements from the first group of cirrhotic (control condition) tissue, column I from the second group of cirrhotic tissue, and so on. Notice how columns B and H contain output from AffyDGED's processing of the first groups of tumor and cirrhotic tissues, respectively.

2. `conditionecol`: This contains a short list defining the first and last column positions within `timecoursedata` to contain experimental condition data.
3. `conditionwocol`: This contains a short list defining the first and last column positions within `timecoursedata` to contain control condition data.
4. `rawtranscriptionreg`: This variable points to the location of a spreadsheet file containing a list (organized into a single column) of known or suspected transcription factor genes present on the microarray chip being used. The file used here was created by parsing the biological process column of the annotation file for Affymetrix's Human Genome U133 Plus 2.0 chip (available at [www.affymetrix.com](http://www.affymetrix.com)) for genes linked to the transcription process. The probeset identifiers referring to this group of genes were used to build a list of transcription factor genes.
5. `rawdegenes`: This link points to the spreadsheet file containing lists of differentially expressed genes (referenced by their probeset IDs, organized into columns) created by processing the experimental and control groups referenced in `timecoursedata` (above) with AffyDGED. In the liver example here, there are six columns of differentially expressed genes created by using AffyDGED to compare the six groups of tumorous livers with the six groups of cirrhotic livers.
6. `affyginlocation`: This variable holds the directory location for finding the Affymetrix `.gin` (gene information) file that provides the necessary information to annotate output from RIFA.
7. `savelocationroot`: This variable holds the location where the user would like the final results of the analysis to be saved.
8. `studyname`: This variable allows the user to name the output files generated by RIFA with study-specific information.

The first tasks completed by RIFA include the loading, parsing, and organization of raw data to facilitate downstream computation.

```

starttime = AbsoluteTime[];

allmicroarraytranscripts = timecoursedata[[All, 1]];

conditionetimepts =
  timecoursedata[[All,
    conditionecol[[1]] ;; conditionecol[[2]]]]];
conditiontwotimepts =
  timecoursedata[[All,
    conditionwocol[[1]] ;; conditionwocol[[2]]]]];

```

```

empties =
  Flatten[Map[Position[rawtranscriptionreg, #] &,
    Complement[rawtranscriptionreg,
      allmicroarraytranscripts]], {1, 2}];

transcriptionreg = Delete[rawtranscriptionreg, empties];

tregpositions =
  Flatten[Map[Position[allmicroarraytranscripts, #1] &,
    transcriptionreg]];

tregconditiononetimepts =
  conditiononetimepts[[tregpositions]];
tregconditiontwotimepts =
  conditiontwotimepts[[tregpositions]];

delist = Drop[Union[rawdegens], 1];

delistpositions =
  Flatten[Map[Position[allmicroarraytranscripts, #1] &,
    delist]];

delistconditiononetimepts =
  conditiononetimepts[[delistpositions]];
delistconditiontwotimepts =
  conditiontwotimepts[[delistpositions]];

condition1liststocorrelate =
  Flatten[Outer[List, tregconditiononetimepts,
    delistconditiononetimepts, 1], {1, 2}];
condition2liststocorrelate =
  Flatten[Outer[List, tregconditiontwotimepts,
    delistconditiontwotimepts, 1], {1, 2}];

```

Upon completion of this first section of code, the transcription factor genes (the “boss” genes from the analogy above) are grouped with the differentially expressed genes (the “slave” genes from above) to facilitate calculation of each pairings’ Spearman rank correlation coefficient.

RIFA proceeds by calculating the Spearman rank correlation coefficients, which requires that each vector of gene expression measurements be tested for the presence of duplicate entries, which requires special handling to calculate Spearman rho. This is the purpose of the `tieCheck` module below. Based on the results of `tieCheck`, the code calls the `spearmanControl` module to optimize calculation of Spearman rho, taking advantage of function listability and the use of compilable expressions, where appropriate, to maximize speed.

```

tieCheck[origdata_] :=
Module[{tiecheck1, tiecheck2, tiepos, tiecheckresult,
  datawithnomultiples, tiedata, notiepos},

  tiecheck1 = Length[origdata[[1, 1]]];
  tiecheck2 = Map[Length[Union[#]] &, origdata, {2}];

  tiepos = Position[tiecheck2,
    {x_, y_} /; (x < tiecheck1  $\vee$  y < tiecheck1), {1}];
  notiepos = Position[tiecheck2,
    {x_, y_} /; (x == tiecheck1 && y == tiecheck1), {1}];

  If[Length[Flatten[tiepos]] == 0,
    (tiecheckresult = False),
    (datawithnomultiples = Delete[origdata, tiepos];
     tiedata = Extract[origdata, tiepos];
     tiecheckresult = {datawithnomultiples, tiedata,
       tiepos, notiepos})]]

spearmanControl[tieresult_, condliststocorrelate_] :=
Module[{sprho, tierho, tierhotopos, notierhotopos,
  joinrho, rhosorted, finalrho},

  If[tieresult === False,
    sprho = fastSpearNoTie[condliststocorrelate[[All, 1]],
      condliststocorrelate[[All, 2]]],

    sprho = fastSpearNoTie[tieresult[[1]][[All, 1]],
      tieresult[[1]][[All, 2]]];
    tierho =
      Table[SpearmanRho[Apply[Sequence, tieresult[[2, i]]]],
        {i, 1, Length[tieresult[[2]]}];

    tierhotopos =
      Partition[
        Flatten[MapThread[List, {tierho, tieresult[[3]]}],
          2];
    notierhotopos =
      Partition[
        Flatten[MapThread[List, {sprho, tieresult[[4]]}], 2];
    joinrho = Join[tierhotopos, notierhotopos];
    rhosorted = Sort[joinrho, #1[[2]] < #2[[2]] &];
    finalrho = rhosorted[[All, 1]]]

fastSpearNoTie[vector1_, vector2_] :=
Module[{vector1sort, vector2sort, rankvec1, rankvec2},

```



```

vector1sort = fastSort[vector1];
vector2sort = fastSort[vector2];

rankvec1 = fastRank[vector1sort, vector1];
rankvec2 = fastRank[vector2sort, vector2];

listableSpear[rankvec1, rankvec2]]

fastSort = Compile[{{vector, _Real, 2}},

Module[{vectorsort},

vectorsort = Map[Sort[#] &, vector]],

CompilationTarget -> "WVM", Parallelization -> True];

fastRank =
Compile[{{vectorsort, _Real, 2}, {vector, _Real, 2}},

Module[{rankvec},

rankvec = Map[Flatten,
Table[Position[vectorsort[[i]], vector[[i, j]]],
{i, 1, Length[vectorsort]},
{j, 1, Length[vectorsort[[1]]}], {1}]],

CompilationTarget -> "WVM", Parallelization -> True];

listableSpear =
Compile[{{rankvec1, _Integer, 2},
{rankvec2, _Integer, 2}},

Module[{bottom, final},

top = 6 * (Total[(rankvec1 - rankvec2)^2, {2}]);

bottom = Length[rankvec1[[1]]] *
(Length[rankvec1[[1]]]^2 - 1);
final = 1 - (top / bottom) // N],

{{top, _Integer, 1}}, CompilationTarget -> "WVM",
Parallelization -> True, RuntimeAttributes -> {Listable}];

```

**Caution:** Due to the sheer volume of computation that needs to be completed using the data described in this paper, the next segment of code will likely take 20–40 minutes to complete (depending on the speed of your computer) and consume roughly 28 Gb of RAM. Computations on machines with less RAM will finish but will require significant use of the hard drive, slowing computation considerably.

```

tierresultscond1 = tieCheck[condition1liststocorrelate];
tierresultscond2 = tieCheck[condition2liststocorrelate];

condition1corr = spearmanControl[tierresultscond1,
  condition1liststocorrelate];
Clear[tierresultscond1, condition1liststocorrelate];

condition2corr = spearmanControl[tierresultscond2,
  condition2liststocorrelate];
Clear[tierresultscond2, condition2liststocorrelate];

```

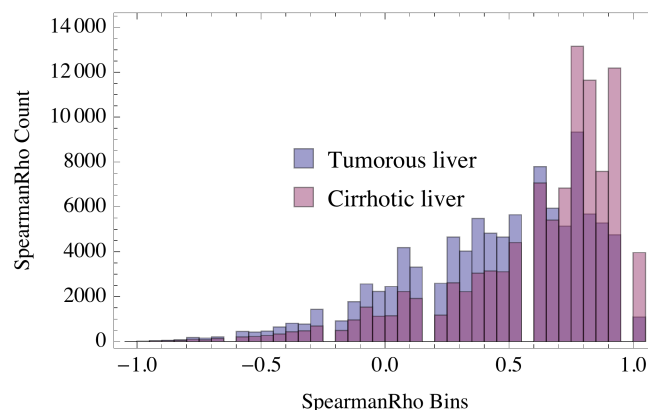
By pairing (and taking a sampling of) the correlations of the “boss” and “slave” genes between the tumor and cirrhotic livers, a satisfying (but not biologically surprising) pattern reveals itself (Figure 3). It is clear that there are many more strongly positive correlations between the “boss” and “slave” genes in the cirrhotic liver than the tumorous liver. This supports what biologists have known for a long time; that is, gene regulation in tumorous tissue is significantly uncoordinated.

```

temp1 = MapThread[List, {condition1corr, condition2corr}];
temp2 = RandomSample[temp1, 100 000];

Histogram[{temp2[[All, 1]], temp2[[All, 2]]},
  ChartLegends → Placed[{"Tumorous liver", "Cirrhotic liver"},
  Center], PerformanceGoal → "Speed", Frame → True,
  FrameLabel → {"SpearmanRho Count", ""},
  {"SpearmanRho Bins", ""}]

```



▲ **Figure 3.** A histogram of the Spearman rank correlation coefficients between regulator and differentially expressed genes in tumorous and cirrhotic liver biopsies.

Following the correlation calculations, RIFA calculates several important metrics, including PIF, RIF4, and RIF5 of equations (1), (2), and (3) described above.

```

diffwiring = condition1corr - condition2corr;

diffexp = Map[Mean, delistconditiononetimepts] -
  Map[Mean, delistconditiontwotimepts];

timeptscombinedmean =
  Map[Mean, Map[Flatten[#, &,
    MapThread[List, {delistconditiononetimepts,
      delistconditiontwotimepts}]]]];

pif = timeptscombinedmean * diffexp;

pifcopyforrif =
  Flatten[Table[pif, {Length[transcriptionreg]}]];

rif4 =
  (Map[Total, Partition[pifcopyforrif * (diffwiring^2),
    Length[delist]]]) / Length[diffexp];

rif4standard = Standardize[rif4];

conditiononemeans = Map[Mean, delistconditiononetimepts];
conditiontwomeans = Map[Mean, delistconditiontwotimepts];

cond1meancopyforrif5 =
  Flatten[Table[conditiononemeans,
    {Length[transcriptionreg]}]];
cond2meancopyforrif5 =
  Flatten[Table[conditiontwomeans,
    {Length[transcriptionreg]}]];

rif5 =
  (Map[Total,
    Partition[((cond1meancopyforrif5 * condition1corr)^2) -
      ((cond2meancopyforrif5 * condition2corr)^2),
    Length[delist]]) / Length[diffexp];

rif5standard = Standardize[rif5];

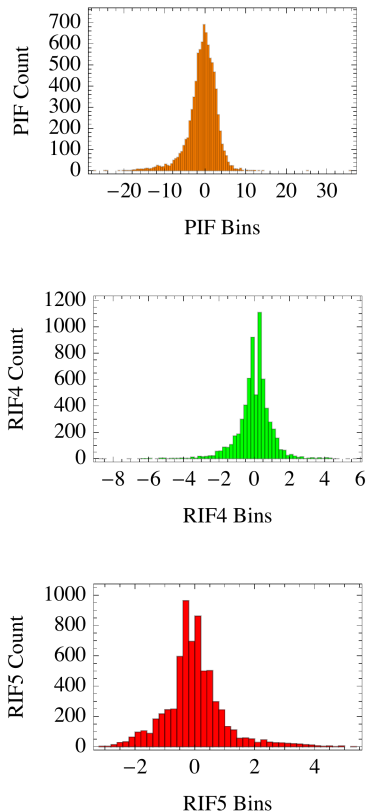
avgrif =
  Map[Mean, MapThread[List, {rif4standard, rif5standard}]];
avgrresult = MapThread[List, {transcriptionreg, avgrif}];

rif4result = MapThread[List,
  {transcriptionreg, rif4standard}];
rif5result = MapThread[List,
  {transcriptionreg, rif5standard}];

```

The resulting plots (Figure 4) of PIF (equation (1)), RIF4 (equation (2)), and RIF5 (equation (3)) reveal the bidirectional nature of each of the three metrics. In other words, regardless of the metric used,  $-8$  and  $+8$  are equally influential to the underlying biology. This makes sense when one remembers that gene expression measurements, used in the calculations of the metrics above, are represented on a  $\log_2$  scale.

```
temp3 = Histogram[pif, ChartStyle → Orange, Frame → True,
  FrameLabel → {"PIF Count", ""}, {"PIF Bins", ""}],
  PerformanceGoal → "Speed"];
temp4 = Histogram[rif4standard, ChartStyle → Green,
  Frame → True,
  FrameLabel → {"RIF4 Count", ""}, {"RIF4 Bins", ""}],
  PerformanceGoal → "Speed"];
temp5 = Histogram[rif5standard, ChartStyle → Red,
  Frame → True,
  FrameLabel → {"RIF5 Count", ""}, {"RIF5 Bins", ""}],
  PerformanceGoal → "Speed"];
GraphicsColumn[{temp3, temp4, temp5}]
```



▲ **Figure 4.** Histograms for each of the three primary metrics used in the RIFA algorithm. Positive and negative values should be interpreted as equally important (i.e., a gene that is fourfold down in expression is equally as likely to be important as a gene that is fourfold up in expression.)

After the metric calculations are completed, four files are exported containing all the results in file formats directly usable by *Mathematica* and Microsoft Excel. One set of files is appended with the phrase “RifSortByAvg” and contains the following information in table form, sorted by the average of RIF4 and RIF5 values.

- Column 1: unique transcript (gene) IDs
- Column 2: the average of RIF4 and RIF5 values
- Column 3: RIF4 values
- Column 4: RIF5 values
- Column 5: genbank accession numbers
- Column 6: gene names
- Column 7: gene product information

A second set of files is appended with the phrase “sortedPIF” and contains the following information in table form, sorted by PIF values.

- Column 1: unique transcript (gene) IDs
- Column 2: PIF values
- Column 3: genbank accession numbers
- Column 4: gene names
- Column 5: gene product information

As described above, the RIF4 and RIF5 results are most useful for identifying the “boss” genes and the PIF results are most useful for identifying the “slave” genes. Both the “boss” and “slave” genes can be influential in creating differences in phenotypes between two states.

```

resultginpositions =
  Flatten[Map[Position[affyginlocation[[All, 4]], #] &,
    avgresult[[All, 1]]]];

ginannotationdata = affyginlocation[[resultginpositions]][[
  All, 8 ;; 10]];

combinationresult =
  MapThread[List, {avgresult[[All, 1]], avgresult[[All, 2]],
    rif4result[[All, 2]], rif5result[[All, 2]],
    ginannotationdata}];

combinationresult = Table[Flatten[combinationresult[[i]],
  {i, 1, Length[combinationresult]}];

combosortbyavg = Sort[combinationresult, #1[[2]] > #2[[2]] &];
combosortbyrif4 = Sort[combinationresult,
  #1[[3]] > #2[[3]] &];

```

```

combosortbyrif5 = Sort[combinationresult,
  #1[[4]] > #2[[4]] &];

pifid = MapThread[List, {delist, pif}];

pifginpositions =
  Flatten[Map[Position[affyginlocation[[All, 4]], #] &
    pifid[[All, 1]]];]

pifannotationdata = affyginlocation[[pifginpositions]][[
  All, 8 ;; 10]];

pifresults =
  Sort[MapThread[List, {delist, pif, pifannotationdata}],
  #1[[2]] > #2[[2]] &];

pifresults = Table[Flatten[pifresults[[i]],
  {i, 1, Length[pifresults]}];]

date = DateString[];
date = DateString[date,
  {"Month", "Day", "Year", "Hour", "Minute", "Second"}];]

foldername = StringJoin[ToString[studyname], " - ", date];

SetDirectory[savelocationroot];
savelocationfinal = CreateDirectory[foldername];
SetDirectory[savelocationfinal];

Put[combosortbyavg, StringJoin[ToString[studyname],
  " - RifSortbyAvg"]];]
Export[StringJoin[ToString[studyname],
  " - RifSortbyAvg.csv"], combosortbyavg];

Put[pifresults, StringJoin[ToString[studyname],
  " - sortedPIF"]];]
Export[StringJoin[ToString[studyname], " - sortedPIF.csv"],
  pifresults];]

Print[
  Style["RIFA calculations complete. All data saved to: ",
  Bold], Style[ToString[savelocationroot], Bold]];

endtime = AbsoluteTime[] - starttime;
Print["Computational Time: ", endtime, " seconds"];

```

**RIFA calculations complete. All data saved to:  
C:\Users\Wookie\Desktop\**

Computational Time: 1541.1711362 seconds

The final output of RIFA is a network graph that associates the most strongly correlated, highest impact PIF scores with the highest impact RIF scores. In this graph, the top 10 most positive and negative average RIF entries, the top 10 most positive and negative RIF4 entries, and the top 10 most positive and negative RIF5 entries are linked to the phenotype of interest with red edges. In other words, the red edges highlight the “boss” genes most responsible for driving changes in phenotype. The nodes of the graph use tooltips to identify the gene represented by the node. The phenotype of interest node is abbreviated “POI.” Blue edges are used to highlight the “slave” genes most responsible and most correlated to the “boss” genes for driving changes in phenotype. In this way, the graph highlights the “slave” genes responding to the “boss” genes’ orders to change phenotype. Here, the highest 0.5% of positive and the lowest 0.5% of negative PIF scores are connected to transcription regulators (represented by RIF nodes) if they share a Spearman rho value of  $\pm 0.9$  with the transcription regulator.

```

graphRIFA :=
  Module[{graphregdata, pifsize, pifforgrph,
    delistposofpifids, topregpos, cond1corrpart,
    cond2corrpart, pifbyregcond1, pifbyregcond2,
    pifcorrstringency1, pifcorrstringency2,
    pifforgrphidsonly, regtopifposition1, regtopifposition2,
    cond1regtopif, cond2regtopif, regpifgrphdata,
    finalgrphdata, grph1},

  graphregdata =
    Union[Join[combosortbyavg[[1 ;; 10]],
      combosortbyavg[[-10 ;; -1]],
      combosortbyrif4[[1 ;; 10]],
      combosortbyrif4[[-10 ;; -1]],
      combosortbyrif5[[1 ;; 10]],
      combosortbyrif5[[-10 ;; -1]]][[All, 1 ;; 2]]];

  grph1 = Thread[graphregdata[[All, 1]] → "POI"];

  pifsize = Ceiling[(Length[pif] * 0.01) / 2];

  pifforgrph = Join[pifresults[[1 ;; pifsize]],
    pifresults[[-pifsize ;; -1]]];

  delistposofpifids =
    Flatten[Map[Position[delist, #] &,
      pifforgrph[[All, 1]]]];

```

```

topregpos =
  Flatten[Map[Position[transcriptionreg, #] &,
    graphregdata[[All, 1]]]];

cond1corrpart = Partition[condition1corr, Length[delist]][[
  topregpos]];

cond2corrpart = Partition[condition2corr, Length[delist]][[
  topregpos]];

pifbyregcond1 =
  Table[cond1corrpart[[i]][[delistposofpifids]],
    {i, 1, Length[cond1corrpart]}];

pifbyregcond2 =
  Table[cond2corrpart[[i]][[delistposofpifids]],
    {i, 1, Length[cond1corrpart]}];

pifcorrstringency1 =
  Table[Flatten[Position[pifbyregcond1[[i]],
    x_ /; (x ≤ -0.90 ∨ x ≥ 0.90)]],
    {i, 1, Length[pifbyregcond1]}];

pifcorrstringency2 =
  Table[Flatten[Position[pifbyregcond2[[i]],
    x_ /; (x ≤ -0.90 ∨ x ≥ 0.90)]],
    {i, 1, Length[pifbyregcond2]}];

pifforgrphidsonly = pifforgrph[[All, 1]];

regtopifposition1 =
  Table[pifforgrphidsonly[[pifcorrstringency1[[i]]]],
    {i, 1, Length[pifcorrstringency1]}];

regtopifposition2 =
  Table[pifforgrphidsonly[[pifcorrstringency2[[i]]]],
    {i, 1, Length[pifcorrstringency2]}];

cond1regtopif =
  Table[Thread[graphregdata[[All, 1]][[i]] →
    regtopifposition1[[i]],
    {i, 1, Length[graphregdata[[All, 1]]]}];
cond2regtopif =
  Table[Thread[graphregdata[[All, 1]][[i]] →
    regtopifposition2[[i]],
    {i, 1, Length[graphregdata[[All, 1]]]}];

```



```

regpifgrphdata =
  Flatten[Delete[Union[cond1regtopif, cond2regtopif], 1]];

finalgrphdata = Union[Flatten[Join[regpifgrphdata, grph1]]]

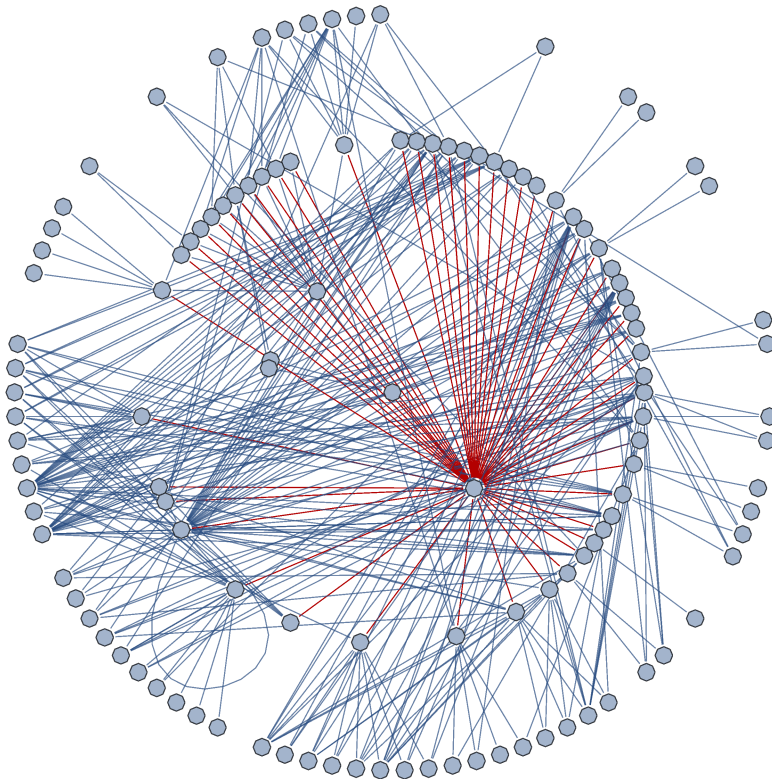
rifagrresults = graphRIFA;

Print[
  Style[
    "Transcription factor and correlated PIF gene network:",
    Bold]];

Graph[MapThread[Tooltip,
  {Join[rifagrresults[[All, 1]], rifagrresults[[All, 2]],
    {"POI"}], Join[rifagrresults[[All, 1]],
    rifagrresults[[All, 2]], {"POI"}]], rifagrresults,
  DirectedEdges → False,
  GraphHighlight → Select[rifagrresults, #[[2]] == "POI" &,
  GraphLayout → "RadialDrawing"]

```

Transcription factor and correlated PIF gene network:



## ■ Gaining Confidence in RIFA

Three primary lines of evidence show that RIFA is performing to design specifications. Evidence line 1: RIFA was created to provide a *Mathematica* implementation of the regulatory impact factor algorithm originally described by Hudson et al. to highlight those genes most directly responsible for driving changes in phenotype. During development, RIFA was vetted with the original data used by these authors, to highlight the genes most responsible for driving phenotypic differences between Wagyu and Piedmontese cattle. The most prominent (and well-characterized) phenotypic difference between these breeds of cattle is the increased musculature of Piedmontese animals, which is known to be due to a mutation in the breed's myostatin (GDF8) gene [7]. Using the author's own data, RIFA correctly identifies GDF8 at the bottom (most negative value of  $-3.02$ ) of its RIF5 output.

```
combosortbyrif5[[-3 ;; -1]] (* do not execute,  
output from another dataset *)  
  
{ {FOXQ1, -4.06869, -5.24315, -2.89423},  
  {GATA3, -3.97711, -5.05679, -2.89744},  
  {GDF8, -2.77124, -2.51563, -3.02685} }
```

Evidence line 2: The Piedmontese/Wagyu data represents the only dataset that is completely and publicly available to validate RIFA. For this reason, evidence line 1 represents the strongest line of evidence that RIFA is functioning properly, as RIFA is able to duplicate the results of Hudson et al. Even so, other gene expression datasets have been analyzed and discussed in the literature that allow for comparison to RIFA output. Please keep in mind that RIFA's results cannot be identical to these other examples, as the full list of normalized gene expression data and the full list of transcription regulators, both necessary input to RIFA, are not publicly available. Reverter and colleagues discuss their analysis of porcine gene expression data from [8] and attempt to explain why their results do not prioritize SRY, a gene that is arguably one of the most important sex-determining genes known to science [9]. Reanalysis of this same data using normalized gene expression from AffyDGED and an alternative list of gene regulators shows that RIFA does highlight SRY as the fourth most negatively prioritized gene. While being a satisfying result, it also serves to highlight the fact that all algorithms (RIFA included) are sensitive to the quality of the input data provided to them.

Evidence line 3: Keeping in mind the discussion above, a similar reanalysis of data referenced in [10–11] shows that RIFA highlights CDK8 [12] as the 33rd most negatively prioritized gene. Hudson's analysis of this data highlights CDK8 as the fourth most positive regulator of colon cancer. Why does RIFA rank CDK8 at position 33, while Hudson's analysis ranks it at 4? The most reasonable explanation is that RIFA processed a transcription regulator list that included 6,685 regulators, while Hudson's work employed a smaller regulator list of 1,292 entries. Fourth out of 1292 (4 divided by 1292) is 0.0031, while 33rd out of 6685 is 0.0049. On a percentage basis of the transcription regulator list size, RIFA's output is nearly identical to that obtained by the original authors who developed the algorithm.

## ■ Interpreting RIFA Results

From the evidence presented, we know that RIFA is performing as expected and can begin to ask if results from other studies make biological sense. If RIFA is working correctly, it should highlight genes that have been linked to cirrhosis and/or cancer in the scientific literature. Keep in mind that not every gene likely to be linked to cirrhosis or cancer has been discovered or characterized yet—which is the value in using a program like RIFA, that is, to find new connections between genes and phenotype. A small sample of output will be reprinted for easier referencing here. Additionally, the output will be reformatted to fit within printable margins by forcing the data in each entry to occupy two or more rows of the table below.

```
optempl = Join[combosortbyavg[[1 ;; 10]],
  combosortbyavg[[-10 ;; -1]], combosortbyrif4[[1 ;; 10]],
  combosortbyrif4[[-10 ;; -1]], combosortbyrif5[[1 ;; 10]],
  combosortbyrif5[[-10 ;; -1]]];
optempl[[6, 7]] =
  "Homo sapiens mRNA;\ncDNA DKFZp667F2113\n(from
  clone DKFZp667F2113)";
optempl[[8, 7]] =
  "Homo sapiens cDNA FLJ12380 fis,\nclone MAMMA1002556";
optempl[[10, 7]] =
  "peroxisome proliferative activated receptor\ngamma,
  coactivator 1";
optempl[[13, 7]] =
  "TATA box binding protein (TBP)-associatedfactor,\nRNA
  polymerase I, A, 48kd";
optempl[[15, 7]] =
  "Homo sapiens cDNA: FLJ22281 fis,\nclone
  HRC03849,\nhighly similar to S69002 human mRNA
  for AML1-EVI-1";
optempl[[19, 7]] =
  "Homo sapiens cDNA FLJ11655 fis,\nclone HEMBA1004554";
optempl[[20, 7]] =
  "ESTs, weakly similar to S47072 finger protein
  HZF10,\nKrueppel-related [H.sapiens]";
optempl[[33, 7]] =
  "signal transducer and activator of\ntranscription
  3 (acute-phase response factor)";
optempl[[36, 7]] =
  "Homo sapiens cDNA FLJ11655 fis,\nclone HEMBA1004554";
optempl[[40, 7]] =
  "ESTs, weakly similar to S47072 finger protein
  HZF10,\nKrueppel-related [H.sapiens]";
optempl[[41, 7]] =
  "Homo sapiens mRNA;\ncDNA DKFZp434P228 (from clone
  DKFZp434P228)";
optempl[[42, 7]] =
  "nuclear receptor subfamily 4,\ngroup A, member 3";
optempl[[47, 7]] =
```

```

"nuclear receptor subfamily 4,\ngroup A, member 2";
optempl[[50, 7]] =
  "v-maf musculoaponeurotic fibrosarcoma\n(avian)
  oncogene family, protein F";
optempl[[52, 7]] =
  "ESTs, highly similar to B45036 Pur beta [H.sapiens]";
optempl[[53, 7]] =
  "ESTs, weakly similar to A32891 finger protein
  1,\nplacental [H.sapiens]";
optempl[[54, 7]] =
  "Homo sapiens mRNA;\ncDNA DKFZp566P1124 (from
  clone DKFZp566P1124)";
optempl[[55, 7]] =
  "Homo sapiens cDNA FLJ11344 fis,\nclone
  PLACE1010870,\nmoderately similar to zinc
  finger protein 91";
optemp2 = optempl[[All, 1 ;; 4]];
optemp3 = optempl[[All, 5 ;; 7]];
optemp4 =
  Text[Grid[Riffle[optemp2, optemp3],
    Dividers -> {False, {{True, False}}}]

```

221427_s_at	2.14857	5.78666	-1.48953
gb:NM_030937.1	HCLA-ISO	hypothetical protein hCLA-iso	
242297_at	2.06738	5.37004	-1.23529
gb:BF904033	None	ESTs	
222999_s_at	2.03408	3.84355	0.224612
gb:AF251294.1	None	hCLA-iso	
200935_at	1.9719	0.927797	3.016
gb:NM_004343.2	CALR	calreticulin precursor	
203752_s_at	1.94041	4.16704	-0.28622
gb:NM_005354.2	JUND	jun D proto-oncogene	
1566901_at	1.87135	3.80376	-0.0610645
gb:AL832409.1	None	Homo sapiens mRNA; cDNA DKFZp667F2113 (from clone DKFZp667F2113)	
204753_s_at	1.85264	2.12814	1.57713
gb:AI810712	HLF	hepatic leukemia factor	
215032_at	1.75512	3.79062	-0.280379
gb:AK022442.1	None	Homo sapiens cDNA FLJ12380 fis, clone MAMMA1002556	
204937_s_at	1.72693	3.88217	-0.428303
gb:NM_016325.1	ZNF274	KRAB zinc finger protein HFB101L	
219195_at	1.69302	-0.676152	4.06218
gb:NM_013261.1	PPARGC1	peroxisome proliferative activated receptor gamma, coactivator 1	
225935_at	-2.88534	-3.58043	-2.19025
gb:AI350995	None	ESTs	
205522_at	-3.01448	-4.23534	-1.79362
gb:NM_014621.1	HOXD4	homeo box D4	
206613_s_at	-3.03219	-4.6906	-1.37378
gb:NM_005681.1	TAF1A	TATA box binding protein (TBP)-associatedfactor, RNA polymerase I, A, 48kD	

21990_at	-3.08847		-4.4771	-1.69983
gb:NM_024680.1	FLJ23311		hypothetical protein FLJ23311	
221884_at	-3.29527		-4.32838	-2.26216
gb:BE466525	None		Homo sapiens cDNA: FLJ22281 fis, clone HRC03849, highly similar to S69002 human mRNA for AML1-EVI-1	
235355_at	-3.38656		-5.92359	-0.849537
gb:AL037998	None		ESTs	
201292_at	-3.55369		-5.36767	-1.73972
gb:AL561834	TOP2A		topoisomerase (DNA) II alpha (170kD)	
209153_s_at	-3.5649		-4.67937	-2.45043
gb:M31523.1	TCF3		None	
233446_at	-3.72081		-6.24142	-1.20021
gb:AU145336	None		Homo sapiens cDNA FLJ11655 fis, clone HEMBA1004554	
239043_at	-3.97311		-8.63574	0.689522
gb:AA084273	None		ESTs, weakly similar to S47072 finger protein HZF10, Krueppel-related [H.sapiens]	
221427_s_at	2.14857		5.78666	-1.48953
gb:NM_030937.1	HCLA-ISO		hypothetical protein hCLA-iso	
242297_at	2.06738		5.37004	-1.23529
gb:BF904033	None		ESTs	
242438_at	1.56291		4.67727	-1.55145
gb:AI819150	None		ESTs	
231806_s_at	1.35598		4.57285	-1.86088
gb:AL133630.1	DKFZp434N0223		hypothetical protein	
208012_x_at	1.60648		4.38909	-1.17613
gb:NM_004509.1	IFI41		interferon-induced protein 41, 30kD	
239193_at	1.68248		4.25423	-0.889276
gb:BF060981	None		ESTs	
213743_at	1.68248		4.25423	-0.889276
gb:BE674119	CCNT2		cyclin T2	
232652_x_at	1.43289		4.20575	-1.33997
gb:AF207829.1	RAZ1		SCAN-related protein RAZ1	
226166_x_at	1.43289		4.20575	-1.33997
gb:AU149216	KIAA1278		KIAA1278 protein	
239738_at	0.638527		4.18173	-2.90467
gb:AW780006	None		ESTs	
224787_s_at	-1.14794		-5.98942	3.69354
gb:AI333232	RAB18		RAB18, member RAS oncogene family	
224377_s_at	-1.14794		-5.98942	3.69354
gb:AF274957.1	None		PNAS-32	
208991_at	-1.12408		-6.18497	3.93681
gb:AA634272	STAT3		signal transducer and activator of transcription 3 (acute-phase response factor)	
210541_s_at	-1.37037		-6.18751	3.44677
gb:AF230394.1	None		tripartite motif protein TRIM27 beta	
215223_s_at	-1.32836		-6.20014	3.54342
gb:W46388	SOD2		superoxide dismutase 2, mitochondrial	
233446_at	-3.72081		-6.24142	-1.20021
gb:AU145336	None		Homo sapiens cDNA FLJ11655 fis, clone HEMBA1004554	
207001_x_at	-0.846111		-6.42241	4.73019
gb:NM_004089.1	DSIP1		delta sleep inducing peptide, immunoreactor	

1554375_a_at	-1.59955		-7.18522	3.98611
gb:AF478446.1	NR1H4		farnesoid-X-receptor beta splice variant 2	
234361_at	-2.01425		-8.52923	4.50073
gb:AC005620	None		Homo sapiens chromosome 19, cosmid R33590	
239043_at	-3.97311		-8.63574	0.689522
gb:AA084273	None		ESTs, weakly similar to S47072 finger protein HZF10, Krueppel-related [H.sapiens]	
227340_s_at	0.362377		-4.66577	5.39053
gb:AL117590.1	None		Homo sapiens mRNA; cDNA DKFZp434P228 (from clone DKFZp434P228)	
207978_s_at	0.949308		-3.01468	4.9133
gb:NM_006981.1	NR4A3		nuclear receptor subfamily 4, group A, member 3	
223650_s_at	-0.261895		-5.38941	4.86562
gb:AF267866.1	None		hNRBF-2	
201130_s_at	1.21496		-2.41524	4.84516
gb:L08599.1	UVO		uvomorulin	
200776_s_at	0.254236		-4.32905	4.83752
gb:AL518328	KIAA0005		KIAA0005 gene product	
216248_s_at	1.16571		-2.48064	4.81205
gb:S77154.1	TINUR		None	
204622_x_at	1.16571		-2.48064	4.81205
gb:NM_006186.1	NR4A2		nuclear receptor subfamily 4, group A, member 2	
207001_x_at	-0.846111		-6.42241	4.73019
gb:NM_004089.1	DSIPI		delta sleep inducing peptide, immunoreactor	
211922_s_at	1.35814		-1.92754	4.64381
gb:AY028632.1	CAT		catalase	
205193_at	0.610199		-3.3539	4.5743
gb:NM_012323.1	MAFF		v-maf musculoaponeurotic fibrosarcoma (avian) oncogene family, protein F	
244462_at	0.380642		3.48294	-2.72165
gb:AA811983	None		ESTs	
227718_at	-1.40892		-0.0703075	-2.74754
gb:BF337790	None		ESTs, highly similar to B45036 Pur beta [H.sapiens]	
242463_x_at	-0.952103		0.862884	-2.76709
gb:AI620827	None		ESTs, weakly similar to A32891 finger protein 1, placental [H.sapiens]	
225594_at	-0.952103		0.862884	-2.76709
gb:AL038866	None		Homo sapiens mRNA; cDNA DKFZp566P1124 (from clone DKFZp566P1124)	
227796_at	-1.86823		-0.910755	-2.8257
gb:AW157773	None		Homo sapiens cDNA FLJ11344 fis, clone PLACE1010870, moderately similar to zinc finger protein 91	
211721_s_at	-2.55532		-2.27466	-2.83599
gb:BC005868.1	None		Similar to zinc finger protein 304	
239738_at	0.638527		4.18173	-2.90467
gb:AW780006	None		ESTs	
226037_s_at	-2.18038		-1.29873	-3.06202
gb:AL049589	LOC51616		neuronal cell death-related protein	
238631_at	0.552748		4.17006	-3.06457
gb:AA490928	None		ESTs	
1568865_at	-0.462844		2.26554	-3.19123
gb:BC035148.1	None		Homo sapiens, clone IMAGE:5264828, mRNA	

What information is revealed in this list? First, let us obtain a list of the unique entries present in this list (a handful of entries may be prioritized by RIF4 and RIF5 metrics simultaneously, thus showing up more than once).

```
optemp5 = Union[optemp1];
```

Do any of the results contain entries that have been linked to cirrhotic or tumorous livers in the scientific literature? Any entry that has a gene name associated with it may have information that can be investigated further.

```
optemp6 = Select[optemp5, #[[6]] != None &][[All, 6]]
```

```
{NR1H4, KIAA0005, CALR, UVO, TOP2A, JUND, NR4A2, HLF,
  ZNF274, MAFF, HOXD4, TAF1A, DSIPI, NR4A3, IFI41, STAT3,
  TCF3, CAT, CCNT2, SOD2, TINUR, PPARGC1, FLJ23311,
  HCLA-ISO, RAB18, LOC51616, KIAA1278, DKFZp434N0223, RAZ1}
```

```
Length[optemp6]
```

29

RIFA produces a list of 29 unique gene names that can be searched for in PubMed ([www.ncbi.nlm.nih.gov/pubmed](http://www.ncbi.nlm.nih.gov/pubmed)). Performing a literature search for these genes in association with liver disease search terms produces the results described in Table 1.

NR1H4 [13, 14]	CALR [15]	UVO [16]	TOP2A [17, 18, 19]
JUND [20, 21]	NR4A2 [22]	HLF [23]	ZNF274 [24]
MAFF [25]	HOXD4 [26]	DSIPI [27]	NR4A3 [28]
STAT3 [29]	TCF3 [30]	CAT [31]	CCNT2 [32]
SOD2 [33]	TINUR [34]	PPARGC1 [35]	HCLA-ISO [36]
RAB18 [37]			

▲ **Table 1.** Results of a PubMed literature search using the names of the genes above in combination with one or more of the following search terms: “liver cancer,” liver cirrhosis,” “cancer.” Citations listed represent a small sampling of the total hits typically discovered.

Twenty-one of the 29 RIFA output entries with a gene name associated with them yield compelling connections between each gene and the disease phenotype search terms “liver cirrhosis,” “liver cancer,” and “cancer,” suggesting that RIFA is enriching for genes driving the phenotypic changes observed between cirrhotic and tumorous liver tissue.

The remaining eight out of 29 genes do not show evidence in the scientific literature linking them to these disease phenotypes. Explanations for this abound, but it is impossible to rule out the possibility that these genes are, in fact, linked to the disease phenotypes but have not yet been characterized by the scientific community. It is simply impossible to conclude if those eight genes are or are not linked to the disease phenotypes at this time. The same conclusion must also be admitted for the other 25 RIFA output entries that have no gene name associated with them. In other words, RIFA has identified 25 potential new “boss” genes associated with the cirrhotic to tumor transition in liver tissue. These may represent valuable new avenues of research.

## ■ RIFA Performance

To gauge the performance of RIFA, several publicly available datasets of different sizes and complexity were analyzed. The first column of Table 2 shows the series accession number for each dataset available at NCBI’s Gene Expression Omnibus. Timings were acquired running *Mathematica* 9.0.1 under Windows 7 (64 bit) using an Intel Core i5-2500K processor overclocked to 4.48Ghz. Total system memory is 32GB. All reported timings use a fresh kernel.

Table 2 reveals that small datasets can easily be processed in under one minute, while very large datasets, involving thousands of transcription regulators, differentially expressed genes, and multiple time points can take upwards of 30 minutes. RIFA’s code base utilizes functions with the `Listable` attribute whenever possible to increase speed, which places demands on the computer’s memory infrastructure, as evidenced by the sizeable memory consumption measured with large datasets.

Series accession number	Time (sec)	Number of transcription regulators	Number of differentially expressed genes	Number of time points per condition	Max memory used (bytes)
GSE14739	19.4	354	2411	4	456 369 816
GSE7032	21.5	1595	1113	2	522 299 688
GSE8536	94.4	766	6685	6	3 561 336 832
GSE17548	1391	6685	8830	6	40 899 150 240
GSE4183	1883	6685	7777	8	47 534 803 296

▲ **Table 2.** Performance timings of RIFA using five different, publicly available datasets.



## ■ Conclusion

Changes in gene expression are at the core of what distinguishes healthy tissue from diseased tissue. Part of unraveling the mystery behind disease centers on identifying those genes most directly responsible for controlling the differences in gene expression that link those differences to disease traits. RIFA's implementation brings to the *Mathematica* user community a compelling algorithm used by biomedical researchers to intelligently prioritize the thousands of genes present in an organism and tie their behavior to specific traits of interest.

## ■ References

- [1] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown, "Quantitative Monitoring of Gene Expression Patterns with a Complementary DNA Microarray," *Science*, **270**(5235), 1995 pp. 467–470. doi:10.1126/science.270.5235.467.
- [2] N. J. Hudson, A. Reverter, and B. P. Dalrymple, "A Differential Wiring Analysis of Expression Data Correctly Identifies the Gene Containing the Causal Mutation," *PLOS Computational Biology*, **5**(5): e1000382, 2009. doi:10.1371/journal.pcbi.1000382.
- [3] A. Reverter, N. J. Hudson, S. H. Nagaraj, M. Pérez-Enciso, and B. P. Dalrymple, "Regulatory Impact Factors: Unraveling the Transcriptional Regulation of Complex Traits from Expression Data," *Bioinformatics*, **26**(7), 2010 pp. 896–904. doi:10.1093/bioinformatics/btq051.
- [4] N. J. Hudson, B. P. Dalrymple, and A. Reverter, "Beyond Differential Expression: The Quest for Causal Mutations and Effector Molecules," *BMC Genomics*, **13**(356), 2012. doi:10.1186/1471-2164-13-356.
- [5] T. Allen, "Detecting Differential Gene Expression Using Affymetrix Microarrays," *The Mathematica Journal*, **15**, 2013. doi:10.3888/tmj.15-11.
- [6] G. Yildiz, A. Arslan-Ergul, S. Bagislar, O. Konu, H. Yuzugullu, O. Gursoy-Yuzugullu, N. Ozturk, C. Ozen, H. Ozdag, E. Erdal, S. Karademir, O. Sagol, D. Mizrak, H. Bozkaya, H. Gokhan Ilk, O. Ilk, B. Bilen, R. Cetin-Atalay, N. Akar, and M. Ozturk, "Genome-Wide Transcriptional Reorganization Associated with Senescence-to-Immortality Switch during Human Hepatocellular Carcinogenesis," *PLOS One*, May 15, 2013. doi:10.1371/journal.pone.0064016.
- [7] C. Berry, M. Thomas, B. Langley, M. Sharma, and R. Kambadur, "Single Cysteine to Tyrosine Transition Inactivates the Growth Inhibitory Function of Piedmontese Myostatin," *American Journal of Physiology: Cell Physiology*, **283**(1), 2002. doi:10.1152/ajpcell.00458.2001.
- [8] M. Pérez-Enciso, A. L. J. Ferraz, A. Ojeda, and M. López-Béjar, "Impact of Breed and Sex on Porcine Endocrine Transcriptome: A Bayesian Biometrical Analysis," *BMC Genomics*, **10**(89), 2009. doi:10.1186/1471-2164-10-89.
- [9] D. Wilhelm, S. Palmer, and P. Koopman, "Sex Determination and Gonadal Development in Mammals," *Physiological Reviews*, **87**(1), 2007 pp. 1–28. doi:10.1152/physrev.00009.2006.
- [10] S. H. Nagaraj and A. Reverter, "A Boolean-Based Systems Biology Approach to Predict Novel Genes Associated with Cancer: Application to Colorectal Cancer," *BMC Systems Biology*, **5**(35), 2011. doi:10.1186/1752-0509-5-35.
- [11] O. Galamb, S. Spisák, F. Sipos, K. Tóth, N. Solymosi, B. Wichmann, T. Krenács, G. Valcz, Z. Tulassay, and B. Molnár, "Reversal of Gene Expression Changes in the Colorectal Normal-Adenoma Pathway by NS398 Selective COX2 Inhibitor," *British Journal of Cancer*, **102**, 2010 pp. 765–773. doi:10.1038/sj.bjc.6605515.

- [12] A. S. Adler, M. L. McClelland, T. Truong, S. Lau, Z. Modrusan, T. M. Soukup, M. Roose-Girma, E. M. Blackwood, and R. Firestein, "CDK8 Maintains Tumor Dedifferentiation and Embryonic Stem Cell Pluripotency," *Cancer Research*, **72**(8), 2012 pp. 2129–2139. doi:10.1158/0008-5472.CAN-11-3886.
- [13] G. Li, Y. Zhu, O. Tawfik, B. Kong, J. A. Williams, L. Zhan, K. M. Kassel, J. P. Luyendyk, L. Wang, and G. L. Guo, "Mechanisms of STAT3 Activation in the Liver of FXR Knockout Mice," *American Journal of Physiology: Gastrointestinal and Liver Physiology*, **305**(11), 2013 pp. G829–837. doi:10.1152/ajpgi.00155.2013.
- [14] G. Li, B. Kong, Y. Zhu, L. Zhan, J. A. Williams, O. Tawfik, K. M. Kassel, J. P. Luyendyk, L. Wang, and G. L. Guo, "Small Heterodimer Partner Overexpression Partially Protects against Liver Tumor Development in Farnesoid X Receptor Knockout Mice," *Toxicology and Applied Pharmacology*, **272**(2), 2013 pp. 299–305. doi:10.1016/j.taap.2013.06.016.
- [15] K. J. Archer, V. R. Mas, K. David, D. G. Maluf, K. Bornstein, and R. A. Fisher, "Identifying Genes for Establishing a Multigenic Test for Hepatocellular Carcinoma Surveillance in Hepatitis C Virus-Positive Cirrhotic Patients," *Cancer Epidemiology, Biomarkers & Prevention*, **18**(11), 2009 pp. 2929–2932. doi:10.1158/1055-9965.EPI-09-0767.
- [16] M. Mareel, M. Bracke, and F. Van Roy, "Cancer Metastasis: Negative Regulation by an Invasion-Suppressor Complex," *Cancer Detection and Prevention*, **19**(5), 1995 pp. 451–464. www.ncbi.nlm.nih.gov/pubmed/7585733.
- [17] N. Wong, W. Yeo, W.-L. Wong, N. L.-Y. Wong, K. Y.-Y. Chan, F. K.-F. Mo, J. Koh, S. L. Chan, A. T.-C. Chan, P. B.-S. Lai, A. K.-K. Ching, J. H.-M. Tong, H.-K. Ng, P. J. Johnson, and K.-F. To, "TOP2A Overexpression in Hepatocellular Carcinoma Correlates with Early Age Onset, Shorter Patients Survival and Chemoresistance," *International Journal of Cancer*, **124**(3), 2009 pp. 644–652. doi:10.1002/ijc.23968.
- [18] N. B. Dawany, W. N. Dampier, and A. Tozeren, "Large-Scale Integration of Microarray Data Reveals Genes and Pathways Common to Multiple Cancer Types," *International Journal of Cancer*, **128**(12), 2011 pp. 2881–2891. doi:10.1002/ijc.25854.
- [19] J. M. Llovet, Y. Chen, E. Wurmbach, S. Roayaie, M. I. Fiel, M. Schwartz, S. N. Thung, G. Khitrov, W. Zhang, A. Villanueva, C. Battiston, V. Mazzaferro, J. Bruix, S. Waxman, and S. L. Friedman, "A Molecular Signature to Discriminate Dysplastic Nodules from Early Hepatocellular Carcinoma in HCV Cirrhosis," *Gastroenterology*, **131**(6), 2006 pp. 1758–1767. doi:10.1053/j.gastro.2006.09.014.
- [20] S. C. Hasenfuss, L. Bakiri, M. K. Thomsen, E. G. Williams, J. Auwerx, and E. F. Wagner, "Regulation of Steatohepatitis and PPAR $\gamma$  Signaling by Distinct AP-1 Dimers," *Cell Metabolism*, **19**(1), 2014 pp. 84–95. doi:10.1016/j.cmet.2013.11.018.
- [21] M. R. Ebrahimbhani, F. Oakley, L. B. Murphy, J. Mann, A. Moles, M. J. Perugorria, E. Ellis, A. F. Lakey, A. D. Burt, A. Douglass, M. C. Wright, S. A. White, F. Jaffré, L. Maroteaux, and D. A. Mann, "Stimulating Healthy Tissue Regeneration by Targeting the 5-HT $2B$  Receptor in Chronic Liver Disease," *Nature Medicine*, **17**(12), 2011 pp. 1668–1673. doi:10.1038/nm.2490.
- [22] Z.-Q. Pan, Z.-Q. Fang, and W.-L. Lu, "Characteristics of Gene Expression of Adrenal Cortical Steroid Synthetase and Its Regulatory Factor in Mice with H22 Liver Cancer of Different Patterns," *Zhongguo Zhong Xi Yi Jie He Za Zhi*, **31**(1), 2011 pp. 85–89. www.unboundmedicine.com/medline/citation/21434351.
- [23] J. Dzieran, J. Fabian, T. Feng, C. Coulouarn, I. Ilkavets, A. Kyselova, K. Breuhahn, S. Doolley, and N. M. Meindl-Beinker, "Comparative Analysis of TGF- $\beta$ /Smad Signaling Dependent Cytostasis in Human Hepatocellular Carcinoma Cell Lines," *PLOS One*, Aug 22, 2013. doi:10.1371/journal.pone.0072252.
- [24] J. W. Prokop, F. J. Rauscher 3rd, H. Peng, Y. Liu, F. C. Araujo, I. Watanabe, F. M. Reis, and A. Milsted, "MAS Promoter Regulation: A Role for Sry and Tyrosine Nitration of the KRAB Domain of ZNF274 as a Feedback Mechanism," *Clinical Science*, **126**(10), 2014 pp. 727–738. doi:10.1042/CS20130385.

- [25] A. Martínez-Hernández, H. Gutierrez-Malacatt, K. Carrillo-Sánchez, Y. Saldaña-Alvarez, A. Rojas-Ochoa, E. Crespo-Solis, A. Aguayo-González, A. Rosas-López, J. M. Ayala-Sanchez, X. Aquino-Ortega, L. Orozco, and E. J Cordova, "Small MAF Genes Variants and Chronic Myeloid Leukemia," *European Journal of Haematology*, **92**(1), 2014 pp. 35–41. doi:10.1111/ejh.12211.
- [26] J. Shen, S. Wang, Y.-J. Zhang, M. A. Kappil, H. C. Wu, M. G. Kibriya, Q. Wang, F. Jasmine, H. Ahsan, P.-H. Lee, M.-W. Yu, C.-J. Chen, and R. M. Santella, "Genome-wide Aberrant DNA Methylation of MicroRNA Host Genes in Hepatocellular Carcinoma," *Epigenetics*, **7**(11), 2012 pp. 1230–1237. doi:10.4161/epi.22140.
- [27] C. Paulin and Y. Chamay, "Demonstration of Delta Sleep Inducing Peptide in a Strain of Human Small Cell Lung Cancer by Immunocytology," *Comptes Rendus de L'Academie des Sciences. Serie III, Sciences de la vie*, **314**(6), 1992 pp. 259–262. www.ncbi.nlm.nih.gov/pubmed/1318772.
- [28] U. Flucke, B. B. J. Tops, M. A. J. Verdijk, P. J. H. van Cleef, P. H. van Zwam, P. J. Slootweg, J. V. M. G. Bovée, R. G. Riedl, D. H. Creytens, A. J. H. Suurmeijer, and T. Mentzel, "NR4A3 Rearrangement Reliably Distinguishes between the Clinicopathologically Overlapping Entities Myoepithelial Carcinoma of Soft Tissue and Cellular Extraskelatal Myxoid Chondrosarcoma," *Virchows Archiv*, **460**(6), 2012 pp. 621–628. doi:10.1007/s00428-012-1240-0.
- [29] G. Ramakrishna, A. Rastogi, N. Trehanpati, B. Sen, R. Khosla, and S. K. Sarin, "From Cirrhosis to Hepatocellular Carcinoma: New Molecular Insights on Inflammation and Cellular Senescence," *Liver Cancer*, **2**(3–4), 2013 pp. 367–383. doi:10.1159/000343852.
- [30] M. Slyper, A. Shahar, A. Bar-Ziv, R. Z. Granit, T. Hamburger, B. Maly, T. Peretz, and I. Ben-Porath, "Control of Breast Cancer Growth and Initiation by the Stem Cell-Associated Transcription Factor TCF3," *Cancer Research*, **72**(21), 2012 pp. 5613–5624. doi:10.1158/0008-5472.CAN-12-0119.
- [31] Q. Wang, W. Chen, L. Bai, W. Chen, M. T. Padilla, A. S. Lin, S. Shi, X. Wang, and Y. Lin, "Receptor-Interacting Protein 1 Increases Chemoresistance by Maintaining Inhibitor of Apoptosis Protein Levels and Reducing Reactive Oxygen Species through a MicroRNA-146a-Mediated Catalase Pathway," *Journal of Biological Chemistry*, **289**(9), 2014 pp. 5654–5663. doi:10.1074/jbc.M113.526152.
- [32] C. Simone and A. Giordano, "Abrogation of Signal-Dependent Activation of the cdk9/cyclin T2a Complex in Human RD Rhabdomyosarcoma Cells," *Cell Death & Differentiation*, **14**(1), 2007 pp. 192–195. doi:10.1038/sj.cdd.4402008.
- [33] P. Nahon, A. Sutton, P. Rufat, M. Ziol, H. Akouche, C. Laguillier, N. Charnaux, N. Ganne-Carrié, V. Grando-Lemaire, G. N'Kontchou, J.-C. Trinchet, L. Gattegno, D. Pessayre, and M. Beaugrand, "Myeloperoxidase and Superoxide Dismutase 2 Polymorphisms Comodulate the Risk of Hepatocellular Carcinoma and Death in Alcoholic Cirrhosis," *Hepatology*, **50**(5), 2009 pp. 1484–1493. doi:10.1002/hep.23187.
- [34] Y. Han, H. Cai, L. Ma, Y. Ding, X. Tan, Y. Liu, T. Su, Y. Yu, W. Chang, H. Zhang, C. Fu, and G. Cao, "Nuclear Orphan Receptor NR4A2 Confers Chemoresistance and Predicts Unfavorable Prognosis of Colorectal Carcinoma Patients Who Received Postoperative Chemotherapy," *European Journal of Cancer*, **49**(16), 2013 pp. 3420–3430. doi:10.1016/j.ejca.2013.06.001.
- [35] B. Wang, S.-H. Hsu, W. Frankel, K. Ghoshal, and S. T. Jacob, "Stat3-Mediated Activation of MicroRNA-23a Suppresses Gluconeogenesis in Hepatocellular Carcinoma by Down-Regulating Glucose-6-Phosphatase and Peroxisome Proliferator-Activated Receptor Gamma, Coactivator 1 Alpha," *Hepatology*, **56**(1), 2012 pp. 186–197. doi:10.1002/hep.25632.
- [36] H. Shimada, K. Nakashima, T. Ochiai, Y. Nabeya, M. Takiguchi, F. Nomura, and T. Hiwasa, "Serological Identification of Tumor Antigens of Esophageal Squamous Cell Carcinoma," *International Journal of Oncology*, **26**(1), 2005 pp. 77–86. doi:10.3892/ijo.26.1.77.

- [37] X. You, F. Liu, T. Zhang, Y. Li, L. Ye, and X. Zhang, "Hepatitis B Virus X Protein Upregulates Oncogene Rab18 to Result in the Dysregulation of Lipogenesis and Proliferation of Hepatoma Cells," *Carcinogenesis*, **34**(7), 2013 pp. 1644–1652. doi:10.1093/carcin/bgt089.

T. D. Allen, "RIFA: A Differential Gene Connectivity Algorithm," *The Mathematica Journal*, 2015. dx.doi.org/doi:10.3888/tmj.17-2.

### List of Additional Material

Additional electronic files:

1. Archive created by free jZip.url
2. HG-U133 Plus\_2.gin
3. hgplus2\_trfactors.xls
4. liver all expression data for rifa.xls
5. liver de genes for rifa.xls

Available at: [www.mathematica-journal.com/data/uploads/2015/02/Allen.zip](http://www.mathematica-journal.com/data/uploads/2015/02/Allen.zip)

### About the Author

Todd Allen is an associate professor of biology at HACC, Lancaster. His interest in computational biology using *Mathematica* took shape during his postdoctoral research years at the University of Maryland, where he developed a custom cDNA microarray chip to study gene expression changes in the chestnut blight pathogen, *Cryphonectria parasitica*.

#### **Todd D. Allen, Ph.D.**

*Harrisburg Area Community College (Lancaster Campus)*  
*East 206R*  
*1641 Old Philadelphia Pike*  
*Lancaster, PA 17602*  
*tdallen@hacc.edu*