

# Marina – Using several statistical metrics to identify over-represented transcription factor binding sites.

Marina version 1.01 – Documentation & Tutorial

Parsa Hosseini

March 19, 2013

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	What Marina is and is not	2
1.2	Marina prerequisites	2
1.2.1	Abbreviations	2
<b>2</b>	<b>TFBS Quantification and Derivation of Over-Representation</b>	<b>2</b>
2.1	Contingency Matrix	2
2.1.1	Iterative Proportional Fitting (IPF) algorithm	3
2.2	Metrics for evaluating TFBS over-representation	3
2.2.1	Support (SP) & Confidence (CF)	4
2.2.2	Laplace Correction (LP)	4
2.2.3	Lift (LI)	4
2.2.4	Jaccard (JAC)	4
2.2.5	Phi coefficient (PHI)	5
2.2.6	Kappa Coefficient (K)	5
2.2.7	Cosine (CO)	5
<b>3</b>	<b>Data Representation</b>	<b>5</b>
3.1	Accepted TFBS Models	5
3.2	TFBS counts and the hypergeometric distribution (optional)	5
<b>4</b>	<b>Providing custom PWMs or DNA motifs</b>	<b>6</b>
4.1	Schema for DNA Motifs	6
4.2	Schema for PWMs	6
<b>5</b>	<b>Tutorial</b>	<b>7</b>
5.1	Runtime Options	7
5.2	Running Sample Data	9
<b>6</b>	<b>Questions and Comments</b>	<b>10</b>

# 1 Introduction

## 1.1 What Marina is and is not

**Marina** is a GUI software tool for identifying over-represented transcription factor binding sites (TFBS) through the use of several knowledge-discovery / data-mining metrics. These metrics quantify and normalize TFBS abundance, ultimately producing a standardized rank of each TFBS and its magnitude of over-representation. Since multiple metrics are involved, a normalization algorithm known as Iterative Proportional Fitting (IPF) is used to yield “agreement” across all metrics as to which TFBSs are truly the most over-represented.

**Marina** is not a centralized database of TFBSs much like that of TRANSFAC, AthaMap, AGRIS, amongst others. Rather, **Marina** is built for an investigator to efficiently mine promoter sequences and find statistically-sound and over-represented TFBSs across multiple groups of promoter-sequence sets.

## 1.2 Marina prerequisites

Prior to version 1.00, **Marina** was built using the Python programming language. Since then, this software has been re-built from the ground up using the Java programming language. Several major fixes are present in this latest build and we encourage usage of this over prior Python builds.

- Java (version 7+ recommended) <http://www.java.com>
- x86 or x64 system with Linux, Mac or Windows OS.

### 1.2.1 Abbreviations

Abbreviation	Full name
CF	Confidence (metric)
CM	Contingency Matrix
CO	Cosine (metric)
IPF	Iterative Proportional Fitting
JAC	Jaccard Index (metric)
JM	J-Measure (metric)
K	Kappa Coefficient (metric)
LI	Lift (metric)
PHI	Phi Coefficient (metric)
PWM	Position Weight Matrix
TFBS	Transcription Factor Binding Site

Table 1: List of abbreviations for commonly used terms

# 2 TFBS Quantification and Derivation of Over-Representation

## 2.1 Contingency Matrix

Central to deriving magnitude of TFBS over-representation to utilization of a contingency matrix (CM). Such a structure is used to model multivariate frequencies, in our case, TFBS abundance between a baseline and control set of promoter sequences. Since this data-structure contains discretized counts, these matrices are frequently used to model magnitude of relationships between a given set of categorical variables.

As illustrated in table 2, frequency counts can be used to derive relationships given both a variable of interest ( $x$ ) and a specific categorical variable ( $C$ ). The cumulative sum per row as well as each column therefore equals that of the entire matrix. Indeed a contingency matrix could be extended to have  $i * j$  rows and columns respectively, however **Marina** utilizes a  $2 * 2$  contingency matrix.

	$C$	$\neg C$	
$x$	$n(x, C)$	$n(x, \neg C)$	$n(x)$
$\neg x$	$n(\neg x, C)$	$n(\neg x, \neg C)$	$n(\neg x)$
	$n(C)$	$n(\neg C)$	$N$

Table 2: A contingency matrix given a variable of interest,  $x$ , and a categorical variable,  $C$

### 2.1.1 Iterative Proportional Fitting (IPF) algorithm

An option in `Marina` is to normalize counts in a contingency matrix so as to better extract underlying patterns and trends. One algorithm for such a purpose is Iterative Proportional Fitting (IPF).<sup>1</sup> The purpose behind IPF standardization is to adjust cell frequencies in such a way that both row and column counts are equal to one another (see table 3).

	$C$	$\neg C$	
$x$	$c(0, 0)$	$c(1, 0)$	$N/2$
$\neg x$	$c(0, 1)$	$c(1, 1)$	$N/2$
	$N/2$	$N/2$	$N$

Table 3: IPF adjusts counts in a matrix,  $c$ , to aide in identifying inherent patterns and associations.

Upon successful frequency adjustments, a contingency matrix,  $c$ , would exhibit counts satisfying the following patterns:  $c_{1,1} = c_{0,0}$ , and  $c_{0,1} = c_{1,0}$ . This equality pattern is represented in table 4. Note the monotonic nature of this matrix given  $x$  and  $\neg x$ .

	$C$	$\neg C$	
$x$	$a$	$N/2 - a$	$N/2$
$\neg x$	$N/2 - a$	$a$	$N/2$
	$N/2$	$N/2$	$N$

Table 4: Adjusting a contingency matrix using IPF yields frequency counts to aide inherent pattern identification

As shown in table 4 and supported by the two earlier equality patterns, two equations are needed to populate this matrix<sup>2</sup>:

$$c_{1,0} = c_{0,0} = a = \frac{N\sqrt{c_{1,1}c_{0,0}}}{2(\sqrt{c_{1,1}c_{0,0}} + \sqrt{c_{1,0}c_{0,1}})} \quad (1)$$

$$c_{0,1} = c_{1,0} = N/2 - a \quad (2)$$

Equations 1 and 2 present solutions for populating a matrix all-while satisfying not only previous equality patterns but also the monotonic nature of the matrix.

## 2.2 Metrics for evaluating TFBS over-representation

An association rule models dependency between a set of variables,  $X$  and  $Y$ , defined as  $X \rightarrow Y$ . We assert that both  $X$  and  $Y$  occur beyond a user-defined or default threshold. Association rules can oftentimes model weak dependencies and as a result, provide very little novel insight. Strong dependencies on the other hand deem themselves worthy of detailed attention and may warrant domain expertise. In other words, a strong association rule implies occurrence of a variable,  $Y$ , if  $X$  is present. Contingency matrices can model these very relationships assuming  $X$  and  $Y$  are discretized variables.

A total of 7 statistical metrics are implemented in `Marina` to help infer what TFBSs are over-represented and what are not. Each of the seven metrics are discussed below, however for an in-depth review of such metrics, please see [GH06].

<sup>1</sup>Developed by W. E. Demming. On a Least Squares Adjustment of a Sampled Frequency Table When the Expected Marginal Totals are Known. *Annals of Mathematical Statistics*, 1940

<sup>2</sup>As presented by T., P-N. et. al., 'Selecting the right interestingness measure for association patterns', SIGKDD 2002.

### 2.2.1 Support (SP) & Confidence (CF)

- **Metric range:** 0 ... 1

Support and confidence are two incredibly useful knowledge-discovery metrics. Both support and confidence were developed by [AIS93] and have come to be one of the most widely-used metrics in data-mining.

$$SUPP = \frac{n(x, C)}{N} = P(x, C) \quad (3)$$

$$CONF = \frac{P(x, C)}{P(x)} = P(C|x) \quad (4)$$

Many published measures such as Jaccard and Klogsen incorporate both support and confidence. Utilization of both these measures was initially proposed by [KT]. Support and confidence can however generate a large set of association rule candidates, and unfortunately, many may not have any significant level of value.

### 2.2.2 Laplace Correction (LP)

- **Metric range:** 0 ... 1

Laplace correction aims to quantify magnitude of accuracy for a particular association rule. The  $k$  variable in the denominator represents the matrix dimension. As is the case for a 2x2 contingency matrix,  $k = 2$ . Marina sets a default Laplace correction cutoff of 0.3; adjusted via the `-1` or `--lap1` flag. Items bearing higher correction scores would certainly attract more domain-expertise than lower correction scores.

$$LP(x, C) = \frac{n(x, C) + 1}{n(x) + k} \quad (5)$$

### 2.2.3 Lift (LI)

Otherwise known as interest, lift is a metric developed by [BMUT97]. Lift computes the probability behind  $x$  and  $C$  occurring together in comparison to if they were independent of one another. Therefore lift quantifies the reliability behind  $x \rightarrow C$ . For example: a lift of 3 implies that  $x$  is three times as likely to yield  $C$  in comparison to what would be sought under a null hypothesis.

- **Metric range:** 0 ...  $\infty$

$$LI(x, C) = \frac{P(x, C)}{P(x)P(C)} \quad (6)$$

### 2.2.4 Jaccard (JAC)

- **Metric range:** 0 ... 1

The Jaccard measure [TKS02] is quite appealing as it incorporates both support and confidence. When two variables are compared against one another, if they exhibit similar patterns, the Jaccard metric returns a value close-to or equal to 1. On the contrary, if the variables are different from one another, the Jaccard metric yields a much lesser value in the vicinity of 0. The definition for this metric is shown below:

$$JAC = \frac{P(x, C)}{P(x) + P(C) - P(x, C)} \quad (7)$$

### 2.2.5 Phi coefficient (PHI)

- **Metric range:** -1 ... 1

The phi ( $\phi$ )–coefficient is a metric to quantitate magnitude of association given two variables. It is important to note the difference between “association” and “correlation”. The former implies dependency while the latter implies a linearly–bound relationship binding two variables. The equation for computing  $\phi$ -coefficient given a contingency matrix like that in table 2 is defined below.

Since the range of this metric is  $\pm 1$ , results at these polar–boundaries represent high association between two variables. If however a coefficient of zero was obtained, this would represent no inherent relationship.

$$\phi(x) = \frac{n(x, C)n(\neg x, \neg C) - n(x, \neg C)n(\neg x, C)}{\sqrt{n(x)n(C)n(\neg x)n(\neg C)}} \quad (8)$$

### 2.2.6 Kappa Coefficient (K)

- **Metric range:** -1 ... 1

$$\frac{P(x, C) + P(\bar{x}, \bar{C}) - P(x)P(C) - P(\bar{x})P(\bar{C})}{1 - P(x)P(C) - P(\bar{x})P(\bar{C})} \quad (9)$$

### 2.2.7 Cosine (CO)

- **Metric range:** 0 ... 1

Cosine [KT], also known as Interest–Support (IS), is an interesting metric for discerning variability given two variables against a null hypothesis.

$$CO = \frac{P(x, C)}{\sqrt{P(x)P(C)}} \quad (10)$$

## 3 Data Representation

### 3.1 Accepted TFBS Models

As mentioned earlier, a TFBS can be modeled in one of two ways (or both, if TFBS models are available for each). The first representation is in the form of fixed–length DNA sequences, hence the name “DNA motif”. The second model, and most preferred, is known as Position Weight Matrices (PWMs). Skeleton templates (schemas) for supplying your own TFBS models are described in section 4.

### 3.2 TFBS counts and the hypergeometric distribution (optional)

Given the hypergeometric distribution and a TFBS–specific contingency matrix, a p-value can be evaluated given TFBS counts across group<sub>a</sub> and group<sub>a+1</sub>. The hypergeometric probability distribution function is as follows:

$$P(X = k) = \frac{\binom{M}{k} \binom{N-M}{n-k}}{\binom{N}{n}} \quad (11)$$

As defined in equation 11,  $x$  represents the random variable,  $N$  is the total population size,  $M$  represents the number of successes given the total population, and  $n$  represents the sample size drawn. As a result, this distribution aims to quantify the probability of  $x$  successes from amongst  $n$  samples, from population,  $N$ . Since we have a populated contingency matrix like that in table 2, we can work with this matrix to yield a hypergeometric distribution probability (table 5).

Our random variable,  $k$ , represents the frequency at which  $x$  is in group  $G$ . On the contrary,  $n(-x, G)$  represents the difference between the number of trials and the number of successes obtained. Modelling

$n(x, -G)$  is however more complicated since this represents the differences between the number of successes in the population and number of successes obtained. Marginals can now be deduced which can aide in filling in the rest of this matrix.

	$G$	$\neg G$	
$x$	$k$	$M - k$	$M$
$\neg x$	$n - k$	$N + k - n - M$	$N - M$
	$k$	$N - n$	$N$

Table 5: Transforming a contingency matrix to be modeled by the hypergeometric distribution [URL]

## 4 Providing custom PWMs or DNA motifs

Marina does not come pre-packaged with TFBS models, be it in the form of PWMs or DNA motifs. Thankfully, many resources exist containing both these models. In plants, for instance, several online resources contain useful models which can be imported into Marina: AthaMap [URL], JASPAR [URL], TRANSFAC [URL], AGRIS [URL], and PLACE [URL]. Investigators can therefore create their own custom TFBS models given these resources, assuming licensing and registration requirements are met.

### 4.1 Schema for DNA Motifs

As discussed earlier, there are two models for representing a TFBS. The first being in the form of a linear string of nucleotide characters, and the latter being as a PWM. In this section, we discuss the schema of the the former model, DNA motifs.

Listing 1: Three-column DNA-motif schema.

bHLH	OsIRO2	CACGTGG
WHIRLY	StWhy1	GTCAAAA
ARF	ARF1	TGTCTC
TRIHHELIX	GT1-BOX	GTGTGGTTAATATG
TRIHHELIX	GT2-BOX	GCGTAATTAA
TRIHHELIX	GT3-BOX	GAGGTAAATCCGCGA

As shown in listing 1, DNA motifs are represented in a three-column tab-delimited file. The first column must be a TF family and the second must represent the TF gene-name. Lastly, the third column represents the actual TF gene DNA motif (binding site). A collection of literature-derived DNA motifs are available in the /demo/ folder.

### 4.2 Schema for PWMs

The second way to model a TFBS is through PWMs. Below is an arbitrary example of a PWM whereby frequency counts are masked as “x”, cushioned by flanking braces. These flanking braces are purely optional and are present in the example below to separate counts in the actual matrix from its respective nucleotide. PWM data-points are separated from one another by either a space or a tab. All elements must also have a corresponding value otherwise an exception will be thrown at runtime. Similar to sample-DNA motifs, several sample-PWMs are also available in the /demo/ folder.

```
>PWM_1
A [ x x x x x x x x x ]
C [ x x x x x x x x x ]
G [ x x x x x x x x x ]
T [ x x x x x x x x x ]
...
>PWM_N
```

```

A [ x x x x x x x ]
C [ x x x x x x x ]
G [ x x x x x x x ]
T [ x x x x x x x ]

```

## 5 Tutorial

Assuming **Marina** was successfully downloaded and all prerequisites have been met, **Marina** can be executed simply by double-clicking on the “Marina.jar” icon or manual execution by issuing the command “java -jar Marina.jar”. The main GUI will then appear, ready for analysis (Figure 1).

To begin analysis, two input conditions must be met:

- 2x FASTA files are required. One of the two will serve as the baseline while the other serves as a query, however both files must contain promoter sequences. In both files, promoter sequences do not have to be the same length. Similarly, input FASTA files do not have to have the same number of FASTA entries.
- Either DNA motifs or PWMs (or both). These TFBS models are to be mapped onto the 2x user-provided FASTA files.

Sample FASTA files are provided. These files represent promoter sequences of the most-induced and most-suppressed genes during a Soybean–Soybean Rust RNA–Seq study (Tremblay et. al, 2012). Sample DNA motifs and PWMs are also provided. If you wish to supply your own motifs, please follow section 4 which outlines the required schema for providing custom TFBS models. Both input FASTA files and TFBS models can be supplied through the **File** → **Load FASTA** and **File** → **Load TFBSs** respectively.

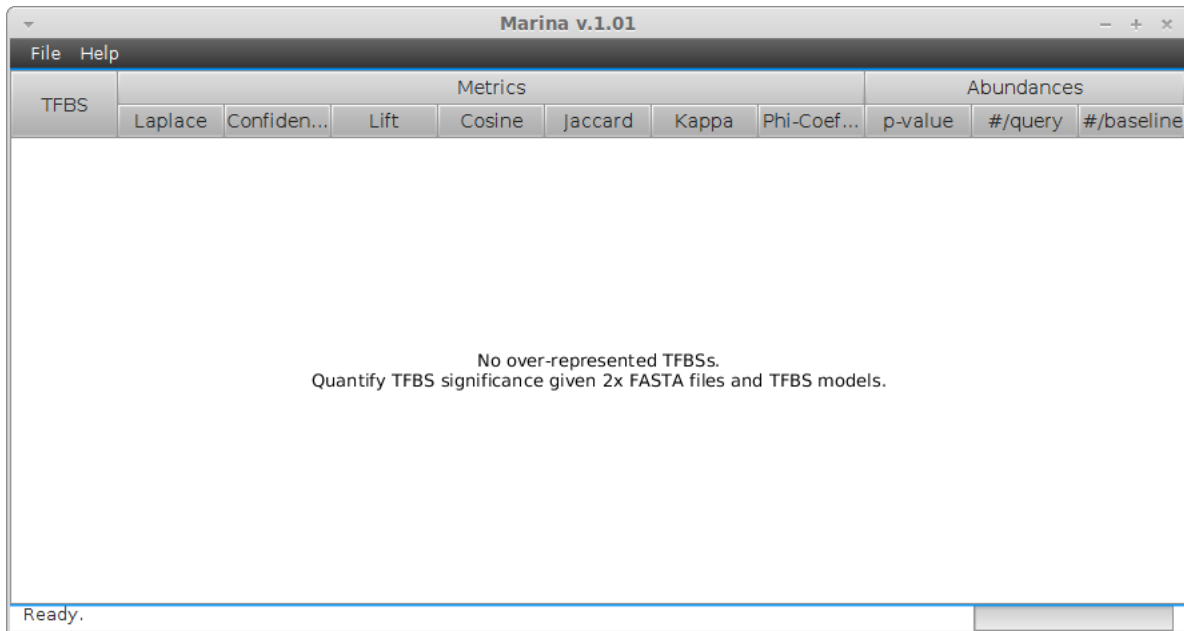


Figure 1: GUI upon launching **Marina**

### 5.1 Runtime Options

The investigator can tweak options by going to **File** → **Options** (Figure 2). Definition of these parameters are provided in Table 6. For this tutorial, we will use default arguments.

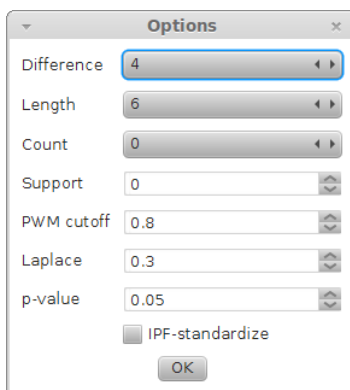


Figure 2: Options can be modified to provide a more strict or lenient mode of analysis

Name	Default	Range	Description
Difference	4	0 ... 100	Represents the difference when a graph node's count is compared to its equivalent node in another group. Graph nodes with differences less than this are removed.
Length	6	0 ... 100	Remove TFBSs that are less than this length.
Count	0	0 ... 100	Each TFBS is modeled as a graph node which is part of a group-specific acyclic graph. Each node has "count" property which is simply the #/TFBSs mapping to this very node. Graph nodes with counts less than this are removed.
Support	0	0 ... 100	Represents probability a TFBS, $t_i$ , is in a group, $G_a$ . This is otherwise written as $P(t_i, G_a)$ . All graph nodes having supports less than this are removed.
PWM cutoff	0.80	0 ... 0.99	Pertains only to when PWMs are supplied. When PWMs are mapped onto TFBSs, a probabilistic score is produced; akin to sequence identity. Fragments with scores $\geq$ this cutoff are kept.
Laplace	0.30	0 ... 0.99	Several statistical metrics are used for evaluating TFBS over-representation. One such metric is Laplace correction (see section 2.2 for details). This parameter filters TFBSs based on their Laplace probability. Lowering this flag may yield many TFBSs that may not be over-represented.
p-value	0.05	0 ... 0.99	For each over-represented TFBS, an accompanying p-value from the hypergeometric distribution is derived. Please refer to section 3.2 for detailed workings of this distribution. All TFBSs with p-values over this p-value cutoff are filtered-out.
IPF--standardize	False	N/A	Contingency matrices are used to model raw-abundance per TFBS given $G_a$ and $G_{a+1}$ . Standardizing counts within this matrix could shed light on inherent associations which TFBS abundance between two groups. Such normalization is performed via the Iterative Proportional Fitting (IPF) algorithm (see section 2.1.1).

Table 6: Marina parameters and their respective arguments



## 5.2 Running Sample Data

For this tutorial, we will use files present in the `/demo/` folder. We will use both sample motifs and PWMs for this example (Table 7). You do not need both PWMs and DNA motifs to begin analysis. This example uses both models solely to illustrate the ability of *Marina* to quantify different TFBS profiles. As long as one of these two models are provided, that will suffice.

File name	entries	Represents
most_induced.fasta	556	Query
most_suppressed.fasta	585	Baseline
sample_motifs.txt	16	DNA motifs
sample_pwm.txt	3	PWMs

Table 7: Sample files used for the tutorial

For each of these four files, a success dialog will appear if parsing the respective file was a success. Exceptions are caught and the investigator is alerted if the input file was not of an acceptable format. Once these files have been provided, selecting **File** → **Run** → **Align** will initiate alignment. If only PWMs were provided, an implementation of the P-MATCH algorithm will be invoked. Similarly if only DNA motifs were provided, an implementation of the Rabin-Karp algorithm will be invoked. Since both models are provided, both alignment algorithms will be executed, back-to-back. Please note that prior *Marina* versions utilized the Boyer-Moore-Horspool algorithm for DNA motif alignment.

During analysis, alignment will take place in the background and a progress-bar will be updated to reflect degree of alignment completion (Figure 3). Eventually, alignment will reach completion and the status-bar will display “Alignments OK. Ready for quantification.”

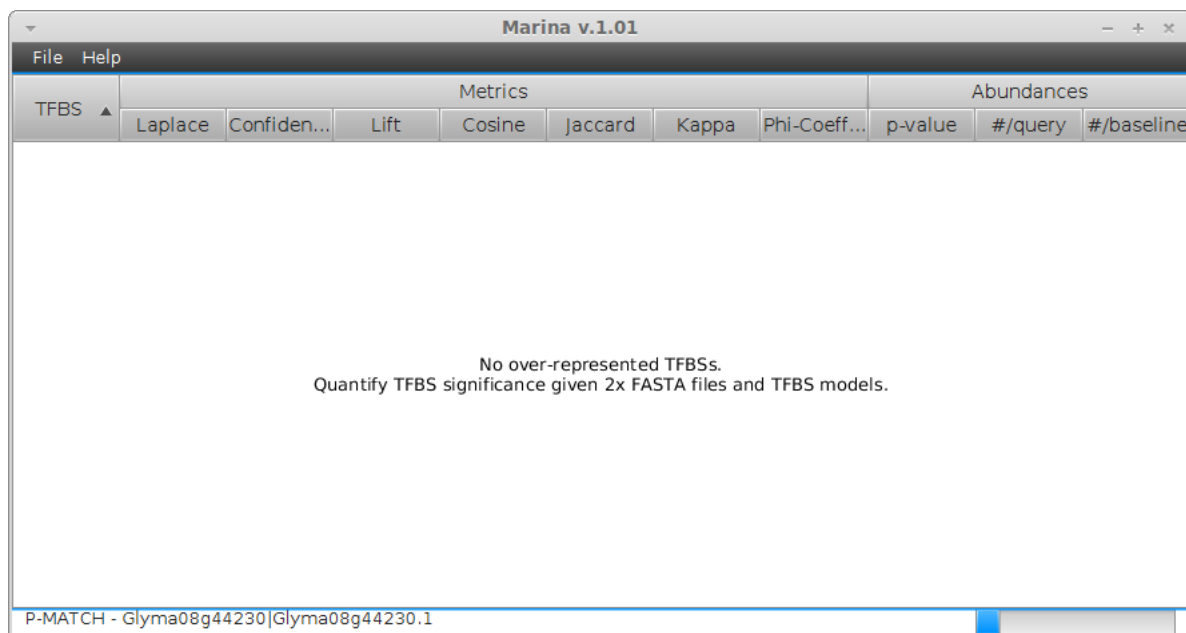


Figure 3: TFBS models being aligned to all provided promoter sequences

Upon alignment completion, we now have reference as to what TFBSs mapped to what promoter sequences. Given such mappings, we can model TFBS abundances between the two groups and quantify TFBS magnitude of over-representation. By selecting **File** → **Run** → **Quantify**, TFBSs are fed through the various options and those which pass all of them are fed into the 7 statistical metrics (Figure 4). Output generated per metric is ranked from 1.0 ...  $N$ , where  $N$  represents the total number of over-represented TFBSs.

TFBS	Metrics							Abundances		
	Laplace	Confidence	Lift	Cosine	Jaccard	Kappa	Phi-Coeffici...	p-value	#/query	#/baseline
1GCC_3DTF	1.0	1.0	1.0	5.0	5.0	2.0	1.0	5.06524435...	53.0	40.0
ARF1	5.0	5.0	5.0	4.0	4.0	7.0	6.0	2.13950898...	266.0	311.0
ARF2	6.0	6.0	6.0	3.0	3.0	8.0	8.0	1.43786811...	609.0	715.0
KN1	7.0	7.0	7.0	8.0	8.0	4.0	5.0	0.00216104...	18.0	23.0
CMe-DREB1/ERF1/ERF2	8.0	8.0	8.0	7.0	7.0	6.0	7.0	0.00120759...	32.0	43.0
OsiRO2	2.0	2.0	2.0	6.0	6.0	3.0	2.0	5.63818101...	49.0	39.0
1VTO_3DTF	3.0	3.0	3.0	2.0	2.0	1.0	3.0	1.18559567...	944.0	1008.0
2QHB_3DTF	4.0	4.0	4.0	1.0	1.0	5.0	4.0	1.18774312...	1150.0	1272.0

# / TFBSs: 8; IPF-standardization: false

Figure 4: TFBSs with ranks close to 1 are most over-represented and vice-versa

Various metrics may not reach unanimous consensus as to the magnitude of over-representation. For instance, “2QHB\_3DTF” is ranked 1st by 2/7 metrics but ranked 4th by 4/7 and 5th by 1/7. Indeed this lack of consensus can make identifying the most over-represented TFBSs an analytical challenge. Had we selected “IPF-standardize” in the Options dialog (Figure 2), the rank per TFBS would be unanimously agreed-upon by each and every metric (Figure 5). Results obtained from quantification can be saved as a tab-delimited file via the **File** → **Save** option.

TFBS	Metrics							Abundances		
	Laplace	Confidence	Lift	Cosine	Jaccard	Kappa	Phi-Coeffici...	p-value	#/query	#/baseline
1GCC_3DTF	1.0	1.0	1.0	1.0	1.0	1.0	1.0	9.9795851249...	1874.12...	1547.873...
ARF1	5.0	5.0	5.0	5.0	5.0	5.0	5.0	1.1802855996...	1682.80...	1739.195...
ARF2	6.0	6.0	6.0	6.0	6.0	6.0	6.0	1.1867844441...	1676.79...	1745.209...
KN1	7.0	7.0	7.0	7.0	7.0	7.0	7.0	1.2199673005...	1646.82...	1775.172...
CMe-DREB1/ERF1/ERF2	8.0	8.0	8.0	8.0	8.0	8.0	8.0	1.2452760507...	1624.78...	1797.218...
OsiRO2	2.0	2.0	2.0	2.0	2.0	2.0	2.0	1.0176143933...	1851.15...	1570.841...
1VTO_3DTF	3.0	3.0	3.0	3.0	3.0	3.0	3.0	1.1319237448...	1729.15...	1692.847...
2QHB_3DTF	4.0	4.0	4.0	4.0	4.0	4.0	4.0	1.1538828347...	1707.75...	1714.246...

# / TFBSs: 8; IPF-standardization: true

Figure 5: IPF enables all metrics to agree as to which TFBSs are the most over-represented

## 6 Questions and Comments

Please contact Parsa Hosseini (Parsa.Hosseini@ars.usda.gov) if you have questions or comments about **Marina** or wish to report a bug.

## Useful Resources & Reading

- [AIS93] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. *ACM SIGMOD international conference on Management of data*, pages 207 – 216, 1993.
- [BMUT97] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur. Dynamic itemset counting and implication rules for market basket data. *SIGMOD*, pages 255 – 264, 1997.
- [GH06] L. Geng and H. J. Hamilton. Interestingness Measures for data mining: a survey. *ACM Computing Surveys*, 38(3), September 2006.
- [KT] V. Kumar and P. Tan. Interestingness Measures for Association Patterns: A Perspective. University of Minnesota.
- [TKS02] P. Tan, V. Kumar, and J. Srivastava. Selecting the right interestingness measure for association patterns. *SIGKDD*, 2002.