First version: *April 23, 2009* Last update: *August 18, 2010*

To weight or not to weight

We recently showed that information theory (IT)-based weighted methods for transcription factor-binding site search generally perform worse than non-weighted ones (Erill and O'Neill 2009). Therefore, the question is: should we use weighted methods or not. And why?

Weighted and non-weighted methods

IT-based methods are derived from an information theory-based analysis of transcription factor binding sites. Given a collection of known binding sites for a transcription factor, these methods are used to score a target sequence positions using a predetermined window size. By applying a threshold, these methods can act as classifiers for putative binding sites in the sequence of interest. Weighted methods are those that weight up/down position scores depending on the "importance" of each position (as derived from the mutual information profile; see this for an introduction to the field), while non-weighted methods simply assume a uniform scoring.

Although there are several variants (Erill and O'Neill 2009), the most widespread IT-based methods are the R_i index (Schneider 1997) and its weighted counterpart, $R_{sequence} \cdot R_i$ (O'Neill 1989; Erill and O'Neill 2009).

$$R_i(l) = \left[-\sum_{S \in \Omega} \left[f(S) \cdot \log_2 \left(f(S) \right) \right] \right] - \left[-\log_2 \left(p(S_{i,l}) \right) \right] = H_{before} - \left[-\log_2 \left(p(S_{i,l}) \right) \right] \quad (1)$$

$$R_{sequence} \cdot R_i = \sum_{l=1}^{L} R_{sequence}(l) \cdot R_i(l) \quad (2)$$

As it can be seen, the non-weighted method R_i assigns a value that is proportional to the logarithm of the frequency of the observed base in the position frequency matrix (PFM) generated by the collection. The weighted method does exactly the same, but now using the $R_{sequence}$ value at each position to weight the score assigned by R_i.

*The case **for** weighted methods*

Intuitively, weighted methods make a lot of sense. The classical argument in favor of weighted methods was clearly stated in (O'Neill 1998). The argument relies on the fact that R_i discards information on the relative importance of each position within the motif. Let's see how it works.

Suppose that for a given motif position *a* we have motif frequencies $p_a(A)=0.6$, $p_a(C)=0.4$, $p_a(T)=0.0$ and $p_a(G)=0.0$. This is a *good* position, in the sense that it is quite decently conserved in the motif. If we observe a *C* in our query sequence, then $R_i(a)=H_{before}-\log_2(0.4)$. Now, suppose position *b* of the motif has motif frequencies $p_b(B)=0.2$, $p_b(C)=0.4$, $p_b(T)=0.2$ and $p_b(G)=0.2$. This a *bad* position, not very well conserved. However, if we again observe a *C* in position *b* of the query sequence, $R_i(b)=H_{before}-\log_2(0.4)$. So R_i assigns the same score to a *C* observed in a relatively well conserved position (*a*) and to a *C* observed in a nearly random one (*b*). This is counterintuitive in the sense that we would expect that a match in a conserved position be more significant than a match in a poorly conserved one.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
A	0.4	0.5	0.3	0.1	0.1	0.1	0.1	0.8	0.2	0.3	0.2	0.3	0.2	0.4	0.1	0.1	0.8	0.2	0.6	0.3	0.4	0.4
C	0.1	0.0	0.1	0.1	0.0	0.1	0.0	0.1	0.2	0.3	0.2	0.2	0.2	0.2	0.1	0.7	0.0	0.6	0.1	0.2	0.1	0.1
T	0.4	0.4	0.4	0.7	0.2	0.8	0.1	0.0	0.4	0.2	0.4	0.2	0.2	0.3	0.8	0.1	0.1	0.1	0.2	0.4	0.5	0.4
G	0.1	0.1	0.1	0.1	0.7	0.0	0.8	0.1	0.2	0.2	0.2	0.2	0.3	0.2	0.1	0.0	0.1	0.0	0.2	0.1	0.0	0.1

Table 1 – Position Specific Frequency Matrix for transcription factor CRP.

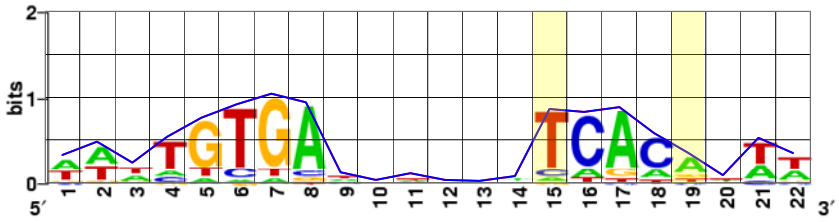

Figure 1 – Sequence logo for the binding motif CRP. The $R_{sequence}$ function is superimposed on the logo.

But is this true beyond the pathological case described above? Should we force a match to be more significant in a conserved position? There is a catch on words here, depending on how we define a *match*. The argument above exposes that a particular match in a conserved position (i.e. observing *C*) should weight more than the same match (i.e. observing *C*) on a less conserved position, *even though the frequency of C is the same in both positions*. On the other hand, if we define a match as observing

the consensus base, and consider anything else a mismatch, one can argue that a match in a conserved position will tend to be more significant than a match in a non-conserved position when using a non-weighted method. This is easy to see with a real-world example. Take the CRP binding site in Figure 1. A *T* in position 15 will get a high R_i score (`1.596`) just because the *T* frequency (consensus) is high at that position (77%), whereas an *A* in position 19 (consensus) will get a lower score (`1.184`) due to a lower frequency of the consensus *A* there (57%). What the weighting procedure does is to reinforce this trend by multiplying the R_i score times $R_{sequence}$. The argument is that conserved positions should be more important to define a motif. Thus, they should contribute more to the score of a potential site. But is this logic correct?

It seems to be. And it seems to make even more sense when considering mismatches. For CRP, a *C* in position 15 will receive a R_i score of `-1.150`. The same happens to a *C* in position 19, since both have the same *C* frequency (11%). If we apply the $R_{sequence}$ weighting, though, position 15 will now contribute `-0.987` and position 19 will generate `-0.401`. What this is saying is that having a *C* in position 15 is much worse (in fact twice as much) than having a *C* in position 19, on the grounds that position 15 is considerably more conserved (0.85 bits) than position 19 (0.35 bits). We assume it is thus more important for the protein to bind, and thereby more susceptible to mismatches. So, yes, the whole thing does seem to make sense. Or does it?

	Consensus	***C*15**	**(DIFF)**	**%**	***C*19**	**(DIFF)**	**%**
R_i	23.02	20.28	-2.75	11.926	20.69	-2.33	10.137
$R_{sequence} \cdot R_i$	13.78	11.43	-2.36	17.099	12.97	-0.81	5.912

Table 2 – Different scores (using weighted and non-weighted methods) for putative BUH sites: consensus, consensus with an *A* in position 2 and consensus with an *A* in position 10. The difference between consensus and mutated scores is shown between brackets.

The previous argument makes a subtle omission. Even though a *C* in both positions leads to a positional R_i score of `-1.150`, this does not mean that a *C* in both positions has the same effect. When we compare against the best possible score for each position (`+1.596` for a *T* in position 15 and `+1.184` for an *A* in position 19) we can easily see that the *C* score of `-1.150` is going to become more important in position 15 than in position 19, because a *C* in position 15 will not only mean having a negative `-1.150` score, but also *loosing* a larger positive putative score. Hence, using R_i *C*15 looses `2.75` with respect to consensus, but *C*19 looses only `2.33`. On the other hand, if we take into account

the full range of scores for both positions, *C*15 represents a decrease of 73% from the maximum score, while *C*19 is 100% (the worst case) decrease from the maximum score. Thus, even though a *C*15 has a larger net effect on the score, its effect is weaker than that of *C*19 in the relative terms of each position.

So, what is going on here? On the one hand, we can conclude that it is not absolutely true that non-weighted methods do not take into account the importance of each position, as they do so implicitly by using the position frequency matrix: *a mismatch in a highly conserved position represents a larger net loss in score than the same mismatch (same frequency) in a less conserved position*. The question thus becomes: should we reinforce this effect by weighting with the $R_{sequence}$ value? Again, we can turn to the numbers to try to get an answer. Using the weighted $R_{sequence} \cdot R_i$ method, the *C*15 score decreases 17% from consensus, whereas *C*19 decreases only 6% (the figures are 12% and 10% for the R_i method). Even though the numbers may look a bit disproportionate, the 17% score reduction does still appeal to our intuition when we look at the CRP logo in Figure 1. Not having a *T* in the 15th position is relatively rare (23% of sites) and therefore missing it should be penalized quite strongly (barring unanticipated interposition dependencies that compensate for the loss), whereas the *A* at position 19th is missing in 43% of cases and thus should be expected to be missing relatively often. In this sense, the difference in scores for the R_i method (even though significant, 2%) seems too small to reflect the importance of the error incurred in by missing the correct base. (This argument is more obvious with artificial motifs; see Appendix).

To conclude, weighting seems to make sense for relatively large motifs, in which the individual contributions of positional R_i values become diluted. Going back to our example, the difference between *C*15 and *C*19 in R_i absolute score loss may be quite significant if taken alone (`2.75` vs. `2.33`; 15% difference), but becomes relatively meaningless when added to the other 21 positional scores (2%). Weighting with $R_{sequence}$ exacerbates the positional score loss (`-2.36` vs. `-0.81`; 65% difference) leading to a difference in total score that still reflects the difference in importance between the positions (13%).

So? So, yes, apparently, weighting makes intuitive sense, emphasizing the importance of conserved positions when scoring sites. Why, then, did we initiate this whole discussion?

*The case **against** weighted methods*

As stated at the beginning, we recently showed that weighted methods perform worse than non-weighted ones when conducting genomic searches for TF-binding sites. This is clear-cut when looking at results for CRP:

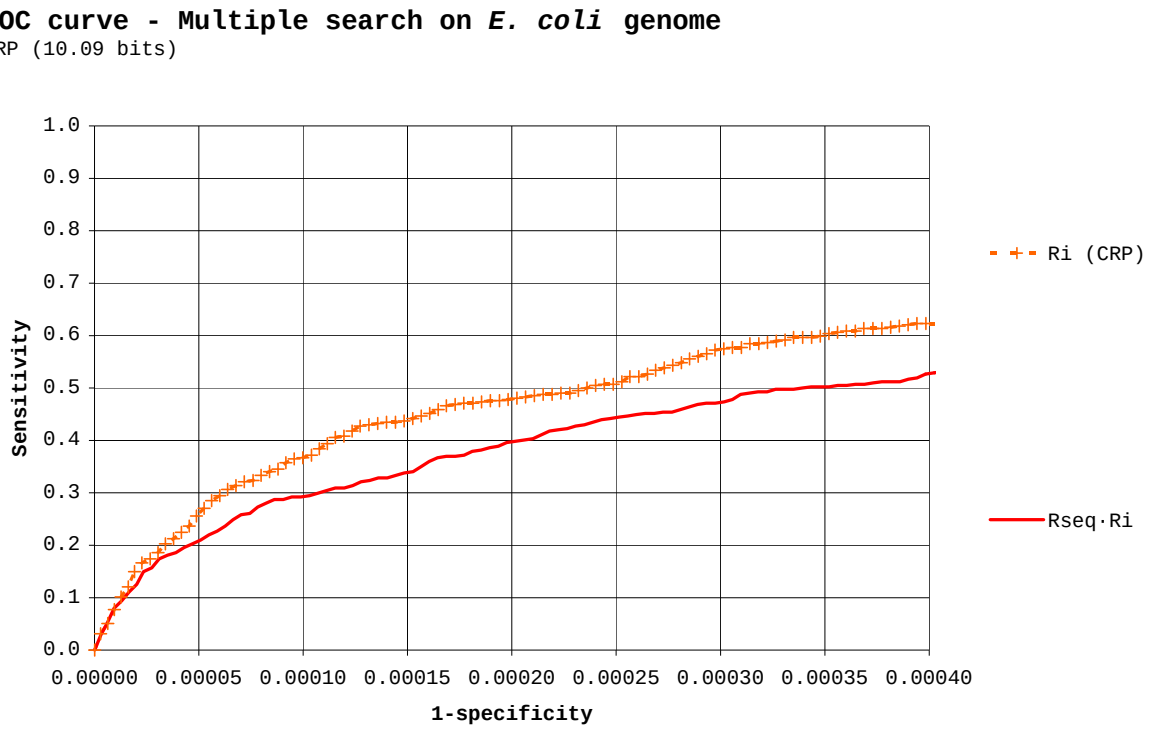

Figure 2 – ROC curve of search for CRP sites on the *E. coli* genome.

The plot above is a ROC curve, showing the decrease in specificity (x-axis) as one tries to increase the sensitivity (y-axis) by raising the method's threshold. On average, the weighted method does 16% worse (more false positives for the same sensitivity) than the non-weighted one, which is clearly not what we were expecting from the previous argument. So, why does this happen? The answer is tantalizingly simple, yet very informative. By concentrating on a number of key conserved positions (and thus downplaying non-conserved ones), weighted methods are more prone to report false positives, simply because the chances that 8 random sequence positions match a profile are far greater than the chances that any 22 random sequence positions match a profile. We can see this clearly with an example. The site GGGG**GTGA**GGGGGG**TCAC**GGGG is clearly not a CRP site. Even though it retains the main motif N4-GTGA-N6-TCAC-N4, any experienced researcher working with CRP will quickly

tell you that the heavy presence of *G*'s in non-conserved positions makes it an unlikely candidate for a CRP site.

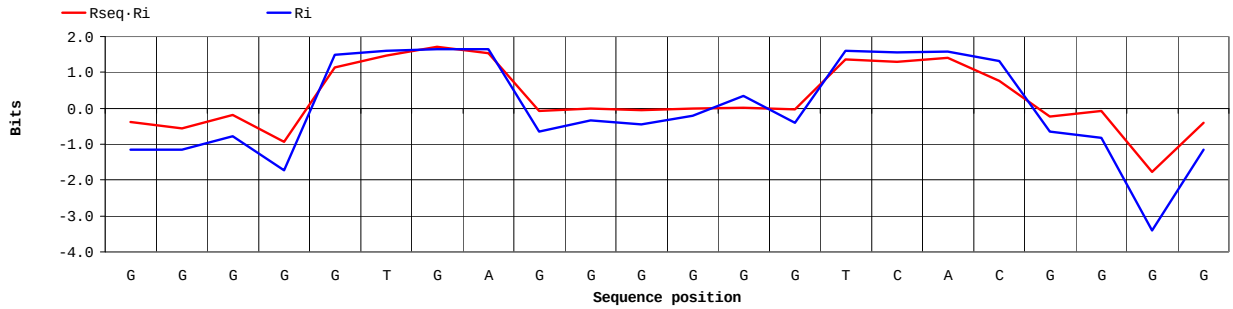

Figure 3 – Scoring profile for sequence GGGGGTGAGGGGGGTCACGGGG using the $R_{sequence}$·R_i and the R_i methods.

However, by concentrating only on conserved positions, $R_{sequence}$·R_i gives this sequence a score of `6.005`, while R_i scores it as `-0.535`. The global range (best to worst possible scores) for $R_{sequence}$·R_i is `+13.78` to `-27.78` (`41.56`). The range for R_i is `23.32` to `-45.98` (`69.3`). Seen in this light, $R_{sequence}$·R_i drops 18.71% with respect to consensus for the query sequence, while R_i drops much more (34.42%). So, basically, R_i is paying more attention to the G's in non-conserved positions than $R_{sequence}$·R_i is. This is obvious in the middle positions, where the $R_{sequence}$·R_i score is negative but almost zero, while R_i scores mainly negative values on the 0.4-0.7 range. It is still more obvious at position 21, which is not very strongly conserved (0.51 bits) but has an extremely low frequency of G's (2%). $R_{sequence}$·R_i assigns a negative score (-1.78) based on the low G frequency, but markedly attenuated by the low information content, whereas R_i strongly punishes the presence of *G*21 (-3.41).

This may sound all like numerology, but it has important effects for the efficiency of both methods. This becomes clearer when we examine the scores for the 210 known CRP sites used to derive the CRP motif. As it can be seen in Table 3, the average score for a CRP site is `7.15` using the weighted method, and the standard deviation is `3.32`. Therefore, the `6.005` score obtained by the GGGG**GTGA**GGGGGG**TCAC**GGGG sequence lies well within what is considered a "normal" CRP site under the scope of the $R_{sequence}$·R_i method when it is obviously not a site. The same does not hold true for the R_i method. Here, the query sequence gets `-0.535`, while the mean and standard deviation are `10.08` and `4.6`. Therefore, the query sequence falls way below the radar for the R_i method. In fact, applying a 95% cutoff under the assumption of a normal distribution, the query would be accepted by $R_{sequence}$·R_i

and rejected by R_i, which is precisely the reason why weighted methods fare worse than non-weighted ones in genomic searches.

	Mean	Std dev	Min	Max	95% cutoff	Worst	Best
$\boldsymbol{R_i}$	10.08	4.6	-19.27	19.02	4.18	-45.99	23.32
$\boldsymbol{R_{sequence} \cdot R_i}$	7.15	3.32	-12.99	13.17	2.89	-27.79	13.78

Table 3 – Mean, standard deviation, maximum and minimum scores (using weighted and non-weighted methods) for CRP sites, as well as the logical cut-off to retain 95% of the 210 sites if assuming a normal distribution. Worst and best possible scores for "ideal" CRP sites are also shown.

Information invisible

So, much to our intuition's chagrin, non-weighted methods perform better in site search because they score fairly non-conserved positions. By definition, however, non-conserved positions are positions with very little positional information content. This means that, apparently, the protein does not care much what base occupies those particular positions. Yet the R_i method is performing a very simple calculation and performing much better than the weighted methods that discard these positions. How is it doing so? What information is it relying on?

R_i itself should be able to provide us with the answer. The method is doing a linear sum of log frequencies. It follows, therefore, that *the sum of these log frequencies* contains information that is useful to discriminate CRP sites. That is, there is no positional information per se in the non-conserved regions, but there is nonetheless global frequency information in these stretches. This means, namely, that R_i is using multi-position information on base frequencies to help it discriminate CRP sites. A well known instance of multi-position frequency information is the GC-content of a sequence segment. Indeed, if one uses the GC-content of the non-conserved stretches (weighted by positional information content, so that log frequency information is provided) to re-sort the predictions of the weighted index, the results improve considerably, but are still not as good as those of R_i.

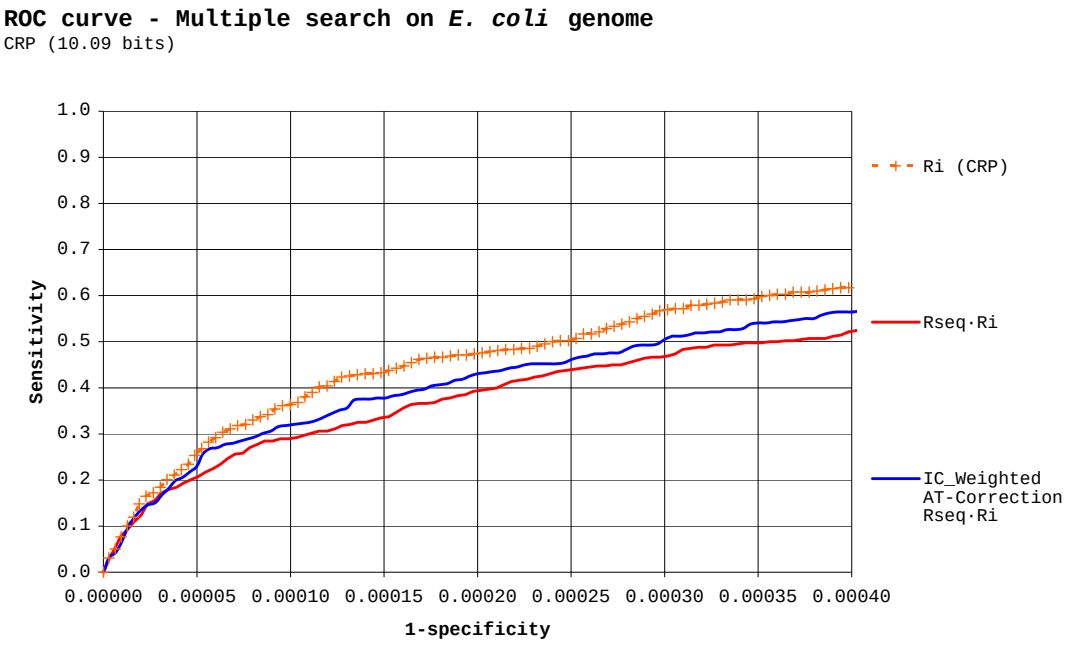

Figure 4 – ROC curve of search for CRP sites on the *E. coli* genome.

Invisible information and protein function

It seems clear thus that the exact scoring of R_i is not simply accounting for %GC, but it must amount to a slightly more complex approximation of it. This explains how a method based on positional information content can extract information from positions with very low information content. But how does this translate into a protein locating its binding sites? We know that proteins bind mostly their targets using specific contacts between particular amino acids and DNA bases (von Hippel and Berg 1989). This again suggests a weighted approach, since specific contacts are the main players. And this seems to be the case when a protein has to differentiate among several of its binding sites, yielding different affinities. This is known as the *ranking* problem and it has been assumed traditionally to be akin to the *search* problem. The results above, however, suggest that this is not so. Apparently, a protein searching for its binding sites is substantially driven by non-specific contacts, which allow it to infer global motif or sub-motif properties, such as AT-richness or curvature, and use them to reject false positive sites more efficiently.

The last word: weighted methods strike back

After such a lengthy treatise, one would presume that all is well and said about weighted and non-weighted methods: weighted methods relate better to the *ranking* problem, non-weighted methods

relate better to the *search* problem. Or so it would seem. One of the questions the above discussion raises is what to use as a binding site alignment. In particular, since weighted methods prove that non-conserved positions can be important for search purposes, one might wonder where a binding site starts/ends. Does it end at the last conserved position as it has been traditionally assumed? Or should we use additional positions, as done above with the 22 bp (instead of conventional 20 bp) CRP collection? And, if so, how many? As it turns out, the question is far from trivial and yields some unexpected results (Bhargava and Erill 2010).

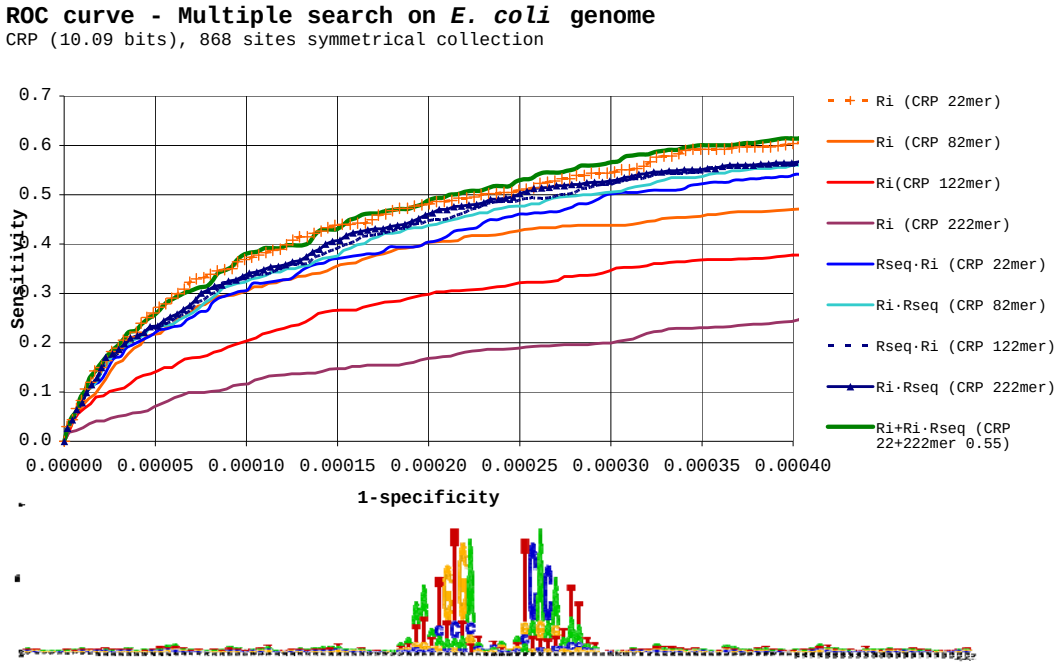

Figure 5 – ROC curve of search for CRP sites on the *E. coli* genome and extended CRP profile (122 bp).

Adding 30 bp on each side to the 22 bp CRP motif dramatically reduces the efficiency of the R_i method, making it worse than the weighted $R_{sequence} \cdot R_i$ method using 22 bp. To the unwary eye this would suggest that there is no additional non-positional information beyond the 22 bp used before for CRP sites. However, if we repeat the experiment with the weighted $R_{sequence} \cdot R_i$ method, we get completely different results: search efficiency *improves* with 82 bp sites. And it continues to improve with 122 and 222 bp sites, albeit very slightly between these last two.

These results tell us that there *is* additional search information beyond the 22/20 bp of conventional sites. They also tell us that the non-weighted approach has built-in limitations. The non-weighted approach works in 22 bp sites because it does not downplay non-conserved positions, thus not allowing conserved positions to dominate and fall prey to random false positives. By the same token, though, the R_i method performs miserably when the site is extended, as the positional information in *conserved* sites (which, after all, is the main source of information for these methods) is progressively diluted by a deluge of non-conserved positions. By weighting non-conserved positions down, weighted methods are able to take in additional positions without much trouble, and to extract some useful information out of them. These results thereby suggest that each set of methods is picking different subsets of non-positional information: inner spacer information for R_i, extended neighborhood information for $R_{sequence}$·R_i. It follows that by combining them we should be able to improve R_i results, and this is precisely what happens, even though mildly, when both scores are combined.

References

Bhargava, N. and I. Erill (2010). "xFITOM: a generic GUI tool to search for transcription factor binding sites." Bioinformation **5**(2): 49-50.

Erill, I. and M. C. O'Neill (2009). "A reexamination of information theory-based methods for DNA-binding site identification." BMC Bioinformatics **10**(1): 57.

O'Neill, M. C. (1989). "Consensus methods for finding and ranking DNA binding sites. Application to *Escherichia coli* promoters." J Mol Biol **207**(2): 301-10.

O'Neill, M. C. (1998). "A general procedure for locating and analyzing protein-binding sequence motifs in nucleic acids." Proc Natl Acad Sci U S A **95**(18): 10710-5.

Schneider, T. D. (1997). "Information Content of Individual Genetic Sequences." Journal of Theoretical Biology **189**(4): 427-441.

von Hippel, P. H. and O. G. Berg (1989). "Facilitated target location in biological systems." J Biol Chem **264**(2): 675-8.

Appendix

The pro-weighting argument using the artificial motif BUH

ATGACATCAT	ATTCGCTAAT	ATTGCGAGAT	GTGTGATCAT	ATGTTGCCAG
ATGCGACAAT	GCTAGCTCAG	ATGCTGATAT	GTACTGACAT	ATGAGATTAT
ATGCTGCCAA	TAGCTAGCAT	TTGTGATGAT	ATGCATTCAG	ATCAGACCAT
ATGCGATAGG	ATCGCGCCAT	TTAGCATGCC	ATGAATACTT	ATGACAGCAT
ATCGACGTAC	ATCGCTACAT	ATTGCATCAG	ATGGACCCCT	ATGATGACTT

Table 4 – List (or collection) of binding sites for the hypothetical protein BUH.

	1	2	3	4	5	6	7	8	9	10
A	0.76	0.04	0.08	0.28	0.12	0.44	0.24	0.12	0.80	0.04
C	0.00	0.04	0.12	0.32	0.28	0.12	0.28	0.68	0.08	0.04
T	0.12	0.92	0.16	0.16	0.28	0.12	0.40	0.08	0.08	0.68
G	0.12	0.00	0.64	0.24	0.32	0.32	0.08	0.12	0.04	0.24

Table 5 – Position Specific Frequency Matrix for transcription factor BUH.

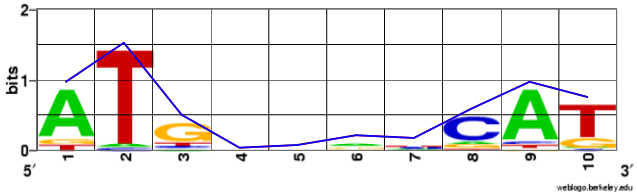

Figure 6 – Sequence logo for the binding motif BUH. The $R_{sequence}$ function is superimposed on the logo.

Here a *T* in position 2 will get a high R_i score (`1.880`) because the *T* frequency is 92%, whereas a *T* in position 10 will get a score of `1.444` due to a lower frequency (68%). Considering mismatches, an *A* in position 2 will get a R_i score of `-1.858`. The same for an *A* in position 10, since both have the same *A* frequency (4%). With $R_{sequence}$ weighting, position 2 will contribute `-2.820` and position 10 will contribute `-1.263`. Thus, having an *A* in position 2 is much worse than having an *A* in position 10.

	Consensus	**A2**	**A10**
$\boldsymbol{R_i}$	10.361	6.776 (-3.585)	7.191 (-3.170)
$\boldsymbol{R_{sequence} \cdot R_i}$	7.733	2.291 (-5.441)	5.577 (-2.156)

Table 6 – Different scores (using weighted and non-weighted methods) for putative BUH sites: consensus, consensus with an *A* in position 2 and consensus with an *A* in position 10. The difference between consensus and mutated scores is shown between brackets.

When we compare against the best possible score (a *T*) for each position (+1.727 and +1.312) we can easily see that the *A* score -1.858 is more important in position 2 than in position 10, because an *A* not only means having a negative -1.858 score, but also *loosing* a larger positive putative score. Hence, *A*2 looses -3.585 with respect to consensus, but *A*10 looses only -3.170.

Using the weighted $R_{sequence} \cdot R_i$ method, the *A*2 score decreases 70%, whereas *A*10 decreases only 28% (the figures are 35% and 31% for the R_i method). In this case, the 70% score reduction does indeed appeal to our intuition. Not having a T in the 2nd position is rare (in fact only one sequence (GCTAGCTCAG) in the collection lacks a *T* there, and its R_i score is so low that one is tempted to think of it as an outlier). In contrast, the *T* in position 10 is not that well conserved, and the penalty should not be that severe.