
Vladimir Bajić
is Director of the Centre for
Engineering Research at
Technikon Natal, South Africa.
His research interest is in
domain AI and complex
systems.

Comparing the success of different prediction software in sequence analysis: A review

Vladimir B. Bajić

Date received (in revised form): 11th May 2000

Abstract

The abundance of computer software for different types of prediction in DNA and protein sequence analyses raises the problem of adequate ranking of prediction program quality. A single measure of success of predictor software, which adequately ranks the predictors, does not exist. A typical example of such an incomplete measure is the so-called correlation coefficient. This paper provides an overview and short analysis of several different measures of prediction quality. Frequently, some of these measures give results contradictory to each other even when they relate to the same prediction scores. This may lead to confusion. In order to overcome some of the problems, a few new measures are proposed including some variants of a 'generalised distance from the ideal predictor score'; these are based on topological properties, rather than on statistics. In order to provide a sort of a balanced ranking, the *averaged score measure (ASM)* is introduced. The ASM provides a possibility for the selection of the predictor that probably has the best overall performance. The method presented in the paper applies to the ranking problem of any prediction software whose results can be properly represented in a true positive–false positive framework, thus providing a natural set-up for linear biological sequence analysis.

Keywords: comparison of
prediction software, measure
of prediction success,
generalised distance, linear
sequence analysis

INTRODUCTION

In the last decade much computer software has been made available to biologists for the analysis of DNA and protein sequences (see references in Baldi and Brunak!). Some of that software aims at predicting the existence and location of specific regions in DNA, eg promoters; or it aims at finding and locating some short motifs in DNA sequences such as transcription start sites, poly-A signals, translation start sites and intron–exon boundaries; or it relates to protein structure and protein function prediction, and so on. The aim of this paper is to make users aware of the fact that improper selection of the measures of prediction quality can bias their conclusion about the overall efficiency of predictor programs in a particular task. To the best knowledge of the author, no adequate ranking system of predictor programs is available. This

may produce a very skewed perception of the relative efficiency of prediction programs and raises a need for a balanced approach in assessing the quality of different prediction software. The method presented here is applicable to any prediction problem, the results of which can be summarized in a 2×2 contingency table and thus expressed in terms of true positive and false positive recognitions. This makes it directly applicable to many linear sequence analysis problems. For the sake of clarity, in what follows the term 'score' is used to denote the actual true positives and false positives obtained in a particular experiment by a program. The term program 'rank' is used to denote the ranking position of a program relative to the ranking positions of other programs in a test.

A reasonable approach to compare the performance quality that predictor programs may achieve is to test several

Vladimir B. Bajić,
Centre for Engineering Research,
Technikon Natal, PO Box 953,
Durban 4000, South Africa

Tel: +2731 204 2560
Fax: +2731 204 2560
E-mail: bajić.v@cer.co.za
[http://www.cer.co.za/staff/
bajić.htm](http://www.cer.co.za/staff/bajić.htm)

averaged score measure of them on the same set of sample sequences, and then, somehow, to compare their scores. Typical examples of such evaluations can be found in Fickett and Hatzigeorgiou² for the prediction of transcription start site of eukaryotic promoters, or in Burset and Guigo³ and Claverie⁴ for gene structure prediction and gene identification.

The predictor programs are based on different methods and algorithms. Thus, one may expect that they will produce different results on the same sample set. The key point for a user having an interest in selecting the best predictor program for a particular task would be to find out which predictor program performs the best. As will be seen, this is not a simple task. Different statistical measures of success are in common use by bioinformaticians (for example, see Burset and Guigo³ and Fickett and Tung⁵). These relate to the correlation coefficient, approximate correlation coefficient, etc. A number of such potentially useful measures can be found in references 3, 6–14. However, one frequently faces a strong discrepancy in ranking prediction programs by these measures, even when the measures relate to the same prediction scores. Moreover, some of these success measures produce results contradictory to each other. This happens because many of these measures emphasise only a few, or even only one, of the several aspects of the quality of predictor performance. Hence, the need exists for an adequate integral measure of success of such programs which can be expressed by a single number.^{3,6–14}

transcription start site

Several possible candidates for such integral assessments of the prediction quality are discussed here and some new ones are introduced. Some of these are variants of the generalised distance from the ideal predictor score. The distance measures are topological in nature, rather than statistical. In addition, it is shown by a typical example that, as a rule, many of the presented measures produce different ranking of predictor performances.

ideal predictor score

As a way out, an averaged score measure (ASM) is proposed, which is a balanced combination of different ranking results. An ASM provides a reasonable overall ranking of predictor performance that will not overemphasise a few specific aspects of the prediction quality. This measure enables a user to have a relatively reliable way to select the most suitable predictor for a particular purpose.

The application of the rank comparison technique of different prediction quality measures will be illustrated through the achieved scores of eukaryotic promoter prediction programs on the test set used in Fickett and Hatzigeorgiou.²

ENVIRONMENT FOR PREDICTOR COMPARISON

To make this paper simple, an analysis is made using examples of the transcription start site (TSS) prediction. This problem is one of the most important in the promoter prediction tasks.^{2,15} However, the results and conclusions that relate to methodology are of a general nature and can easily be modified to cover other problems of predictor ranking, as long as the prediction results can be represented in a true positives–false positives framework.

Let us consider a set S of k DNA sequences s_i , $i = 1, 2, \dots, k$, each of length n_i nucleotides. The task is to attempt to locate all TSSs that may exist in these sequences. Let us assume that for the set S all TSSs are known. We aim to find out how well the different programs that can be used for the TSS prediction perform on this set. After obtaining their scores, we want to rank them, from the best to the worst, so as to be able to select the most suitable one for the TSS prediction task.

We will assume that if the predictor program signals the existence of a TSS, then this prediction will be counted as correct if it falls within certain bounds (data window) around the actual TSS

location. For example, the predicted TSS will be counted as correct if it is up to and including position $-k_1$ (upstream) of the actual TSS and up to and including position k_2 (downstream) of the actual TSS. Here, $k_1, k_2 \geq 0$. In this text we will consider the position of the TSS to be between the nucleotides with positions -1 and $+1$. Then the data window width W is $W = k_1 + k_2$.

Remark. A predictor program may be trained on the data using a window of width W_1 , obtained by considering p_1 nucleotides upstream of the TSS and p_2 nucleotides downstream of the TSS. In order to be able to recognise a TSS, the program then requires a presentation of the data window of length $W_1 = p_1 + p_2$, where the TSS is located just after the nucleotide at the position p_1 counted from the starting nucleotide of the window. If p_{TSS} is the position of the nucleotide just before the actual TSS location in, say, the sequence s_j , then the program that requires a data window of width W_1 will not be able to recognise this TSS if $p_{\text{TSS}} < p_1$ or if $p_{\text{TSS}} > n_j - p_2$. In other words, all TSSs in the set S should be such that their positions relative to the starting nucleotides of the sequences, as well as the ending nucleotides of the sequences along the examined strand, are sufficiently distanced so as to allow the correct application of all predictor programs that are to be compared. If that is not the case, then the comparison will not be fair for all programs. For this reason, it is assumed throughout the text that if a comparison of predictor programs is made, it is done in such a way that each of the programs gets an equal chance of success, ie that the TSSs in the set S are sufficiently distanced from the ends of the sequences, or that all tested programs cannot recognise the same TSSs for the reasons mentioned.

true positives

false negatives

false positives

It is useful to analyse the set S with regards to the window W . Let us assume that S contains N_{TSS} actual TSS positions, of which N_{TSS}^s are on the sense strands and N_{TSS}^c on the complement strands. Consider the sequence s_j and assume that it contains a TSS on the sense strand. Then, taking into account the original assumption on when we count recognition as correct, the number of data windows that may produce correct prediction of this TSS will be equal to W . However, only one of all such possible predictions will count as correct while the other $W - 1$ will be ignored. To calculate the number of non-TSS positions, the discarded $W - 1$ positions should be subtracted from the total number of possible window positions on the sense strand of the sequence s_j . Consequently, the total number of non-TSS positions on the sense strands, N_n^s , to be considered is

$$N_n^s = \left[\sum_{j=1}^k (n_j - W + 1) \right] - N_{\text{TSS}}^s W$$

If both DNA strands are used for the search, then the total number of non-TSS positions N_n is

$$N_n = 2 \left[\sum_{j=1}^k (n_j - W + 1) \right] - (N_{\text{TSS}}^s + N_{\text{TSS}}^c) W \quad (1)$$

A predictor program will normally make some correct predictions. These are called true positives (TP). Obviously, $0 \leq \text{TP} \leq N_{\text{TSS}}$. The remaining part of the TSS locations that are not predicted by the program designates false negatives (FN). It follows that $\text{FN} = N_{\text{TSS}} - \text{TP}$, $0 \leq \text{FN} \leq N_{\text{TSS}}$. It is also common that predictor programs make false predictions. Let the predictor program make predictions of the TSS locations that cannot be considered as correct (according to our accepted criterion for the distance of the predicted TSS location from the actual TSS within the window of length W). Such predictions are called false positives (FP). We have $0 \leq \text{FP} \leq N_n$.

true negatives

Also, all other non-TSS locations are essentially correctly treated by the predictor – it does not predict them falsely as the TSS locations. Thus, they are called true negatives (TN). As a consequence we have $0 \leq TN \leq N_n$, and $TN = N_n - FP$.

relative measures of success

A typical situation that a user will have to face is that, on the set S of sequences used for testing, each of the predictor programs will produce, in general, different values of these four weak indicators of prediction success. Which program is then the best predictor? Or, maybe, such a question makes no sense? As will be seen later, the question as to the best predictor cannot be answered. All ranking will express only *relative* measures of success.

contingency table

CONTINGENCY TABLE

Consider two discrete variables, X and Y , defined as follows. If the data window of width W is located so that the actual TSS is between nucleotides at the positions k_1 and $k_1 + 1$ and within the window, counting the position of the first nucleotide in the window as 1, then X will indicate the correct location of the TSS. Otherwise, X indicates a non-TSS location. We can think of X as having values that belong to one of the two possible categories: correct indication of the TSS location and incorrect indication of the TSS location (or indication of the non-TSS location). In other words, the variable X expresses the real TSS presence. The other variable Y relates to the prediction of the TSS presence. If the analysis of data in the window suggests the presence of the TSS, then Y counts as correct, otherwise it counts as incorrect.

strength of association

Similarly, as for X the variable Y has values that can be in one of the two categories: the result predicts the presence of the TSS, or the result does not predict the presence of the TSS (ie the score indicates the presence of the non-TSS). This allows us to set up a contingency table (Table 1) for each predictor in the form of frequencies of occurrence of each category of variables X and Y . In Table 1, P is the total number of scores indicating the presence of TSS, N is the total number of scores indicating non-TSS, while N_{tot} is the total number of tested positions. The contingency table can serve as a background for many measures that reflect some aspects of predictor quality. Note that in the contingency table there is no correction for the program-specific data windows width for different prediction programs. This is a consequence of the comments in the Remark above and of the assumptions introduced there. If there are TSSs that are located too close to the beginning or to the end of a sequence, which makes it impossible for some of the compared programs to recognise them, but at the same time makes this possible for the other programs, then the recognition of such TSSs should be excluded from the comparison analysis. However, a specific abilities of programs to recognise such TSSs may be important in the overall context of program quality assessment.

MEASURES FOR EXPRESSING PREDICTOR QUALITY

There are several measures that can be used to express the strength of the association of variables X and Y . Some of

Table 1: Contingency table

Real	Predicted		Total
	Score indicates TSS	Score indicates non-TSS	
Window points to TSS	TP	FN	$N_{TSS} = TP + FN$
Window points to non-TSS	FP	TN	$N_n = FP + TN$
Total	$P = TP + FP$	$N = FN + TN$	$N_{tot} = N_{TSS} + N_n$

χ^2 statistic

these measures are related, one way or another, to the χ^2 statistic, which is frequently used for the analysis of contingency table data.^{6,9,11,14} Note that for the 2×2 contingency table data from Table 1, χ^2 can be calculated according to

$$\chi^2 = N_{\text{tot}} \frac{(TP \times TN - FP \times FN)^2}{(TP + FN)(TN + FP)(TP + FP)(FN + TN)}$$

ideal predictor score

Also, several connections with the quality of the predictor scoring can be made based on the contingency table, and consequently we can use some of these measures of association of X and Y to express the quality of predictor results.

worst predictor score

Interpretation with respect to the prediction quality

All association measures of X and Y that are obtained on the basis of χ^2 statistic provide a partial answer to the following question: are the observed frequencies in the rows of the contingency table (Table 1) independent of the frequencies in the columns of the table? The null hypothesis H_0 requires that the row frequencies are independent of the column frequencies. In other words, H_0 states that the predictor score is independent of whether the predictor scoring window points to the TSS or not. The alternative hypothesis H_A is the opposite of H_0 and claims that the predictor score depends on the situation when the predictor scoring window points to the TSS and when it points to the non-TSS. In the case of a

good prediction quality, the alternative hypothesis has to hold. The inherent problem with the association measures of X and Y used on the basis of the contingency table stems from the fact that there are two completely opposite situations that characterise maximally dependent frequencies of occurrence in rows and columns. One is when $FN = FP = 0$; this one corresponds to what one may call the ‘ideal predictor score’. There are no false predictions and the program correctly guesses all positive targets (all TSSs) and all negative targets (all non-TSSs). The other is when $TP = TN = 0$; this corresponds to the ‘worst possible predictor score’. In this case the program did not make any correct prediction. Thus, the necessary conditions for candidates for good measures of association of X and Y that reflect the prediction quality should: (a) distinguish between these two opposite situations, and (b) show a gradual, but strictly monotonic, change in the association measure value when the score changes from the worst to the best case scenario. Table 2 lists the measures commented on in this paper and their relation to conditions (a) and (b). It is interesting to note that two popular measures, sensitivity SN and specificity SP, do not satisfy condition (b). This, however, does not mean that they are not useful, but only that they are not the most appropriate as single measures of prediction success.

The connection with the quality of prediction stems from the following observation: a good predictor

Table 2: The relation of different measures to the conditions (a) and (b). Note that the popular measures of sensitivity SN and specificity SP do not satisfy condition (b) on gradual strict monotonic change of their values when the scores change from the worst possible case to the best possible case.

Conditions	K_1	K_2	CTG	ϕ_1	ϕ_2 (CC)	Q	m	SMC	S_p	SP	SN	ACP	ACC	GDIP ₁	GDIP ₂	GDIP ₃	ASM
(a)	+	+	-	-	+	+	+	+	+	+	+	+	+	+	+	+	+
(b)	+	+	-	-	+	+	+	+	-	-	-	+	+	+	+	+	+

See text for abbreviations.

prediction quality coefficients

should, in practice, make the ratio TP/FN considerably larger than FP/TN. This seems logical, as one would expect to have a better score in the correct recognition of the real TSS locations on the data windows that point to the TSS, rather than on windows that point to the non-TSS. Hence, one may establish a relation

$$\frac{TP}{FN} = K \frac{FP}{TN} \tag{2}$$

with the interpretation that the higher the value of K , the better is the predictor quality. From equation (2) one obtains

$$K = \frac{TP \times TN}{FN \times FP}, \quad K \geq 0 \tag{3}$$

The product in the numerator is the product of correct recognition. The value of $TP \times TN$ increases as the number of correct recognitions increases, and it decreases as the number of correct recognitions decreases. The opposite observation holds for $FN \times FP$. Thus, K is higher if the correct recognitions are higher, and it decreases with the increased number of false recognitions. One problem here is that if either FN or FP or both are zero, then K is undefined.

contingency coefficient

The first and second prediction quality coefficients (K_1 , K_2)

Since K becomes undefined when either FP or FN or both are zero, one can overcome this problem by a simple alteration of the formula in (3) to

$$K_1 = \frac{TP \times TN}{FN \times FP + \frac{1}{N_{tot}}}, \quad 0 \leq K_1 \leq N_{tot} \times TN \times TP$$

As an indicator of prediction quality K_1 has good properties: it is 0 when there are no correct positive predictions or no correct negative predictions ($TP = 0 \vee TN = 0$) It gradually increases up to the value $N_{tot} \times TN \times TP$, as long as there are correct recognitions of both positive and negative samples ($TP > 0 \wedge TN > 0$).

Cramer's ϕ_1 coefficient

We will call K_1 the first prediction quality coefficient.

If one considers a ratio of all correct recognitions and all false recognitions, then such a ratio can relate directly to the quality of predictor performance. In order to be able to include the performance of the ideal predictor into such a measure, the second prediction quality coefficient K_2 is defined as

$$K_2 = \frac{TP + TN}{FN + FP + \frac{1}{N_{tot}}}, \quad 0 \leq K_2 \leq N_{tot}(TP + TN)$$

The measure by K_2 is a good one. It is zero when there are no correct predictions ($TP = TN = 0$) and it reaches its maximal value $N_{tot}(TP + TN)$ when there are no false predictions ($FP = FN = 0$). As the number of correct recognitions increases and the number of false recognitions decreases, the value of K_2 gradually increases.

Contingency coefficient CTG

One other measure of association between X and Y is the contingency coefficient^{7,8,10,14} denoted by CTG. For the case of Table 1 it is given by

$$CTG = \sqrt{\frac{\chi^2}{\chi^2 + N_{tot}}}, \quad 0 \leq CTG \leq \frac{\sqrt{2}}{2}$$

This coefficient will produce highest values when the score relates to the ideal predictor score, but also in the case when it relates to the worst predictor score. It will produce 0 when there is no association between X and Y ($\chi^2 = 0$). CTG does not satisfy conditions (a) and (b). Therefore it will not be able to distinguish between the ideal prediction score and the worst prediction score and thus it is not a good measure to be used for predictor quality ranking.

Cramer's ϕ_1 coefficient

Another association coefficient is the so-called ϕ_1 coefficient:¹⁶

$$\phi_1 = \sqrt{\frac{(TP \times TN - FP \times FN)^2}{(TP + FN)(TN + FP)(TP + FP)(FN + TN)}}, \quad 0 \leq \phi_1 \leq 1$$

It ranges from 0 to 1 and does not satisfy conditions (a) and (b) mentioned above. As the CTG coefficient, it is not suitable for the predictor score ranking.

ϕ_2 coefficient

correlation coefficient

A better association measure of X and Y is the ϕ_2 coefficient (sometimes called the correlation coefficient CC) defined by

$$\phi_2 = \text{(CC)} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TP} + \text{FP})(\text{FN} + \text{TN})}}, \quad -1 \leq \phi_2 \leq 1 \quad (4)$$

correlation coefficient of Ives and Gibbons

Note that contrary to ϕ_1 , the coefficient ϕ_2 satisfies both conditions (a) and (b). The nearer the value of ϕ_2 is to 1, the better is the overall predictor performance. However, the ϕ_2 coefficient may not always be defined. This happens when any of the factors in the denominator of equation (4) becomes zero, which happens frequently in bioinformatics research.³

simple matching coefficient

This measure is very popular in bioinformatics, but, as will be seen later by way of an example, drawing conclusions on the quality of prediction based predominantly on this measure is not advisable. The ranking of prediction software performance obtained in this way need not necessarily rank programs adequately.

Yule's association coefficient

Yule's association coefficient Q

A good statistical measure of the association of X and Y , the coefficient Q is provided by:^{12,13}

$$Q = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\text{TP} \times \text{TN} + \text{FP} \times \text{FN}}, \quad -1 \leq Q \leq 1$$

In two pathological cases Q is not defined. One is when $\text{TP} = N_{\text{TSS}}$, $\text{TN} = 0$, $\text{FN} = 0$, $\text{FP} = N_{\text{p}}$, ie when the predictor in all tested cases signals the presence of positive targets; the other one is when the predictor does not recognise any positive targets but recognises all negative targets correctly ($\text{TP} = 0$, $\text{TN} = N_{\text{p}}$, $\text{FP} = 0$, $\text{FN} = N_{\text{TSS}}$.

specificity

These two situations are very rare in practice. The coefficient Q satisfies conditions (a) and (b). In a majority of practical situations we have N_{tot} of the order of 10^3 to 10^4 . Then, if there are both $\text{FN} > 0$ and $\text{FP} > 0$, one can neglect the value of $1/N_{\text{tot}}$ with regards to $\text{FN} \times \text{FP}$. In such a situation, one can expect that K_1 and Q will produce the same ranking of prediction scores.

Correlation coefficient m of Ives and Gibbons

Another correlation coefficient¹⁷ for the association of X and Y is defined by

$$m = \frac{\text{TP} + \text{TN} - \text{FP} - \text{FN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \quad -1 \leq m \leq 1$$

This measure also has good characteristics: it is always defined and it satisfies conditions (a) and (b). This coefficient always produces the same ranking as the so-called simple matching coefficient and as K_2 .

Simple matching coefficient SMC

The probability of correct prediction is called simple matching coefficient SMC^{3,18} and it is defined as

$$\text{SMC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \quad 0 \leq \text{SMC} \leq 1$$

This and many other simple matching coefficients can be found in reference 6. The SMC satisfies both conditions (a) and (b) and it is simpler to calculate than m . The important thing to note here is that the ranking of prediction scores by K_2 , m and by SMC will *always be the same*. This can be shown by simple algebraic manipulations. For this reason we need to use only one of these three measures.

Specificity S_p

One of the frequently used measures of predictor quality is that of specificity. It is the probability that the negative target will be recognised correctly by the predictor, when the predictor program

sensitivity

points to the negative target. In our case of the TSS prediction, this will be the probability that the predictor program will indicate the presence of the non-TSS when the program window points to the non-TSS location. Specificity is defined by

$$S_p = \frac{TN}{TN + FP}, \quad 0 \leq S_p \leq 1$$

If the specificity is small, then the predictor does not recognise negative targets correctly and the level of FP will be unacceptably high. If its value is close to 1, it makes a good recognition of the negative targets (non-TSSs). But even if S_p is close to 1, indicating that almost all negative targets are correctly identified, it is still not a good enough indication of prediction success as the predictor may not recognise positive targets at all, ie $TP = 0$ is possible. Reference 3 comments that since in practice TN is normally much higher than FP, then S_p is generally insufficiently informative. This, however, is only partially true and depends on the context in which S_p is used. Even in the cases when TN is several orders of magnitude larger than FP the coefficient S_p can be good for ranking the prediction efficiency.

second specificity coefficient

Second specificity coefficient SP

In bioinformatics another specificity measure is also in use,^{3,19-21} denoted as SP. It is defined as

$$SP = \frac{TP}{TP + FP}, \quad 0 \leq SP \leq 1$$

This measure is the probability that the predictor program has indicated a positive target when it predicts the target as positive. In other words, in relation to the TSS prediction, SP represents the probability that the TSS is correctly recognised when the predictor program indicates the presence of the TSS. If $SP = 1$, it is still possible to have a very low level of TP so as to make the prediction quality unacceptable.

averaged conditional probability

Sensitivity SN

Another popular measure is sensitivity. It is the probability that the predictor will correctly recognise positive targets when positive targets are presented to it. In other words, relating to our TSS prediction, sensitivity is the probability that the predictor program will indicate the presence of the TSS when the program window points to the TSS. Thus, the sensitivity is defined by

$$SN = \frac{TP}{TP + FN}, \quad 0 \leq SN \leq 1$$

Values of sensitivity close to 1 indicate that the predictor is able to recognise most of the positive targets. However, even if the predictor score implies $SN = 1$, this does not mean that it is close to the ideal predictor, and the number of false positive predictions may be so high that the overall predictor performance may be regarded as useless. Similarly, as with specificity coefficients, we frequently experience this deficiency in practice. In other words, each of the measures, S_p , SP and SN, overemphasises only one of the aspects of the predictor quality and they are not good as single measures of predictor success. Note that the measures S_p , SP and SN do not satisfy condition (b).

Averaged conditional probability ACP

The so-called averaged conditional probability ACP^{3,6} is defined by

$$ACP = \frac{1}{4} \left(SN + SP + S_p + \frac{TN}{TN + FN} \right), \quad 0 \leq ACP \leq 1$$

if all members within the brackets are defined. Since these members represent the appropriate probabilities³ the name of ACP is used. If some of the members in the brackets are not defined, then ACP is calculated as the averaged sum of the defined members. Note that, in any case, at least two members in the brackets are defined, implying that ACP is

always defined. This is a good measure and one that satisfies conditions (a) and (b).

Approximate correlation coefficient ACC

approximate correlation coefficient

A derived coefficient from ACP is the so-called approximate correlation coefficient ACC defined by

$$ACC = 2ACP - 1$$

ACP and ACC will always produce the same ranking of predictor scores and thus it is advisable to use only one of them.

generalised distances from the ideal predictor

Generalised distances from the ideal predictor (GDIP₁, GDIP₂ and GDIP₃)

It was mentioned previously that the measures of prediction success based on the contingency table and the association of variables X and Y , involve an inherent problem given by the fact that the ideal predictor score and the worst predictor score are characterised by the maximally dependent frequencies of occurrence in rows and columns of the contingency table. This causes a problem in interpreting the prediction quality by using the strength of association of X and Y . For this reason three measures are introduced that express the quality of prediction score in relation to the distance from the ideal predictor score. These measures are topological measures. In the FP–FN plane, the score of the ideal predictor will be positioned at the origin. The score of any other predictor will be represented by a point in the first quadrant, including the axes. Then the Euclidean distance of a predictor score from the origin is given by $\sqrt{FP^2 + FN^2}$.

averaged score measure

However, in cases when one predictor score is characterised by $TP = 0$ and $FP \ll TN$, it may happen that the other predictors have high TP values and still be located at larger distances from the origin than the first one. This is obviously unacceptable. To eliminate this anomaly, one can introduce correction factors to obtain the generalised distance formulas as

$$GDIP_1 = \frac{\sqrt{FP^2 + FN^2}}{TP + TN + \frac{1}{N_{tot}}},$$

$$0 \leq GDIP_1 \leq N_{tot} \sqrt{N_n^2 + N_{TSS}^2}$$

$$GDIP_2 = \frac{\sqrt{FP^2 + FN^2}}{TP + \frac{1}{N_{tot}}},$$

$$0 \leq GDIP_2 \leq N_{tot} \sqrt{N_n^2 + N_{TSS}^2}$$

$$GDIP_3 = \frac{\sqrt{FP^2 + FN^2}}{TN + \frac{1}{N_{tot}}},$$

$$0 \leq GDIP_3 \leq N_{tot} \sqrt{N_n^2 + N_{TSS}^2}$$

The motivation for using three different correction factors is as follows. For $GDIP_1$ the correction factor penalises all false recognitions. The larger the number of overall false recognitions $FP + FN$, the smaller $TP + TN$ will be, and the distance from the ideal predictor score will be larger. Similarly, $GDIP_2$ penalises FN recognitions, while $GDIP_3$ penalises FP recognitions. These measures express the quality of prediction in the most direct way, not indirectly as in the case of association measures that follow from the contingency table. The predictor quality is thus better as its scores place it closer to the origin in the $FP - FN$ plane. Thus, the smallest value of $GDIP_1$ or $GDIP_2$ or $GDIP_3$ correspond to the best ranked predictor. These measures satisfy both conditions (a) and (b) and are very good for expressing prediction quality.

Averaged score measure (ASM)

As will be seen from the example that follows in the next section, different measures of predictor success imply different ranking positions for the same prediction score. The question is, how in such a situation can one find the most reasonable ranking? To find a reasonable compromise between the different measures and the ranking based on them, one of several possible

averaged score measures is presented. Assume that we have used z measures to rank predictor scores and that we are interested in comparing the relative performances of p programs. Let the ranking of the scores of the i th program obtained by using different measures be given by the row vector $r_i = [P_1^i, \dots, P_z^i]$. This vector has elements that are positive integers P_j^i , representing the ranking position of the program score obtained by a particular measure m_j . Here we assume that position 1 is for the best program and that the ordering is in ascending order. Now we define the averaged score measure for the i th program as

$$\text{ASM}_i = \frac{1}{z} \sum_{j=1}^z P_j^i$$

The use of the ASM requires a special consideration. Note that every averaged measure of ranking results produces ranking relative to scores of programs used in a comparison. Thus, it may happen that the addition of some of the programs to be used in the comparison, or removal of some of them, influences the mutual relative ranking of other programs. To eliminate this problem we need to make invariant the mutual ranking positions of all of the compared programs obtained using z measures. To do this we can adopt two approaches. One is to make the final rank list based on the comparison of ASM values of *all possible scores* for a given experiment. Frequently this requirement is computationally expensive. In order to be more practical and to reduce the computational problems significantly we can adopt the following consideration. Consider the sequence s_j . Make the prediction of the TSS locations at positions $1, 1 + W, 1 + 2W$, etc., where the position denotes the position of the nucleotide just before the guessed TSS, and where counting of nucleotides begins with the first

nucleotide in a sequence. Then, to guess all TSSs contained in a sequence s_j in a two-strand search, one needs $2I_j$ guesses, where I_j is the integer part of n_j/W , ie where we ignore the remainder. Note that here we rely on the adopted criterion (see section on 'Environment for predictor comparison' above) for what counts as a correct guess of the TSS.

Consequently, the number of required guesses for the prediction of all TSS locations in the two-strand search is equal to $b = 2 \sum_{j=1}^k I_j$. Then, instead of all possible scores, we need use only all combinations of scores obtained by varying TP from 0 to N_{TSS} , and by varying FP from 0 to b . Thus, the total number of scores to be used to derive the relative ranking of the considered p programs is $G = (N_{\text{TSS}} + 1) \times (b + 1)$. The reason for using this reduced number of scores is that we can exclude from consideration, as inefficient, any program whose score produces $TP < N_{\text{TSS}}$ while having $FP \geq b$, or when $TP = N_{\text{TSS}}$ but has $FP > b$ since the simple equidistant guesses of possible TSSs produces a better score. Once the ranking of all G scores is obtained, the relative positions in terms of the ASM of the considered p programs define their ranking position for the conducted experiment. The price we have to pay for making the ranking positions of the considered p programs mutually invariant is in having to rank a larger list of possible scores (G scores in total).

The rank position degrades as the ASM increases. The worst score is given by the highest value of the ASM. This measure performs a sort of averaging of results of other ranking measures and it is quite suitable in representing the overall score. One can interpret the ranking results obtained by it as the probable overall performance *relative to the measures used*. This measure can include all other measures available for the evaluation of predictor performance. It will provide a sort of balanced overall ranking, not overemphasising any specific aspects of

relative ranking

pool of measures

predictor performance that other individual measures may introduce.

Comment. The use of ASM poses some problems that require comments. The crucial question related to the ASM is: which measures should be included in the pool of measures used to derive the ASM? This is a very sensitive and complicated issue, and its proper solution requires a separate study. We will highlight some of the problems related to such a selection. Ideally, the measures used should be statistically unrelated as much as possible. The standard approach to measure this type of independence is by the correlation coefficient. Let c_{ij} denote the correlation coefficient between the rankings of G scores produced by any two measures m_i and m_j . We would expect that the closer the c_{ij} is to 1, the more likely it is that measures m_i and m_j will produce a similar ranking for the compared p programs. Also, if c_{ij} is very close to zero, one would expect that the measures m_i and m_j will produce different rankings of our considered p programs. Both of these reasonings appear to be wrong. We can get the same ranking for our considered p programs based on measures m_i and m_j even when c_{ij} is close to zero; but we can also get very disparate ranking when c_{ij} is relatively large. The example that follows will illustrate such behaviour. Yet another aspect of the problem of selecting criteria for the inclusion of different measures in the pool of measures is that a general practice in bioinformatics is to consider the behaviour of predictor programs with regards to sensitivity SN and specificity SP. If more measures that behave like, say, SN (or SP) are included in the pool, then the ASM may become biased. However, the resolution of this problem is intimately tied to the solution of the previous one. These are the problems that motivate a comprehensive study on the criteria for inclusion of different measures in the pool of measures to derive the optimal ASM for a specific group of problems.

Unless such a study is made, a hint would be to include in the pool of measures as many measures as possible that theoretically do not produce the same ranking results; or to use only those measures whose mutual correlation coefficient c_{ij} obtained on ranking all G scores is less than a predetermined threshold, eg that including all measures for which $|c_{ij}| < 0.9$.

EXAMPLE

As a convincing example of different resultant ranking and an illustration of finding a balanced overall ranking, we will use the results obtained in an evaluation study of programs for the prediction of eukaryotic promoters presented in Fickett and Hatzigeorgiou.² The data set used is small and not properly representative of the diversity of eukaryotic promoters. Thus, the ranking results presented in this section serve only to illustrate the application of the method for software prediction comparison, and *cannot* be used to draw conclusions on the actual quality of promoter prediction programs included in the comparison.

Table 3: Programs and their scores, to be compared as an illustration of the procedure

	TP	FP
Audic ²⁶	5	33
Autogene ²⁷	7	51
Score of Promoter 1.0 is replaced by score of Promoter 2.0	10	43
NNPP	13	72
Promoter Find ²⁸	7	29
PromoterScan	3	6
TATA ²⁹	6	47
TSSG ³⁰	7	25
TSSW ³⁰	10	42
HMM	12	39
SPANN1	12	44
SPANN2	8	16

The correct conclusion in this regard would be to evaluate promoter prediction programs on a sufficiently rich and statistically properly structured set of sequences. Unfortunately, at this moment such a representative set is not publicly available (A. Hatzigeorgiou, private communication). In Fickett and Hatzigeorgiou² only the ability of programs to detect the presence of the TSS was tested. First, nine programs listed in Table 3 have been evaluated. In the meantime, several other results that use strand-specific searches were reported on the same data set.^{22–25} These are indicated in Table 3 as HMM,²⁵ SPANN1²² and SPANN2.²³ The original result achieved by Promoter1.0 is replaced by the new result,²⁴ as it performs better. For details on other programs see references in Table 3.

The data set from Fickett and Hatzigeorgiou² contains 18 sequences of a total length of 33,120 base pairs (bp). It also contains 24 TSS locations, hence $N_{\text{TSS}} = 24$. The score of prediction was counted as correct if it was within -200 nucleotides upstream of the real TSS location and $+100$ nucleotides downstream of the TSS location. Thus, $W = 300$. Details on the length of individual sequences are given in Fickett and Hatzigeorgiou². The total number of non-TSS positions was thus, according to equation (1), $N_n = 48,276$. The total number of scores that have to be used in computing ASM is determined as $(N_{\text{TSS}} + 1)(b + 1) = 5,075$, where $b = 202$ for the two-strand search. The criterion for selection of the measures for inclusion in the pool of measures is that $|c_{ij}| < 0.9$ for any two measures m_i and m_j in the pool.

Whatever is our perception of the quality of prediction results of the programs from Table 3, Table 4 shows how they rank using ten different measures. These results are obtained by making the rank lists with different measures, using all G combinations of scores for this particular experiment.

Discussion

Note that the ten measures used produce eight different overall rankings. The ASM, which is a balanced measure, gives a ranking different from all others.

Interestingly, we see that based on the ASM measure, the NNPP program,^{31,32} which scores best regarding the absolute TP score, ranks only at position 9 in the total ranking owing to a very large number of FPs. The PromoterScan program of Prestridge,³³ however, although achieving the least absolute TP score, is at the much better fifth overall position. Moreover, four measures K_2 , $GDIP_1$, $GDIP_3$ and S_p rank the PromoterScan program in the first position, while at the same time they place the NNPP program last. Contrary to this, SN places the NNPP program in the first position and the PromoterScan program last. However, ASM orders performances of these programs by combining and balancing all scores used. Also, it is interesting to note that if we exclude the programs whose results have been reported after the evaluation study of Fickett and Hatzigeorgiou,² then the TSSG program of reference 30 is the best program according to ASM, and PromoterScan program is the second best one. The NNPP program is in position 6.

However, the ranking of all G prediction scores for Q and SN has $c_{ij} = 0.865$, but there is no stronger similarity between rankings of the 12 considered programs. On the other hand, although the correlation coefficient for ranking G considered scores by $GDIP_1$ and $GDIP_3$ is $c_{ij} = 0.468$, and by $GDIP_3$ and S_p it is $c_{ij} = -0.095$, the rankings of the 12 considered programs is the same by all these three measures. Moreover, rankings of compared 12 programs based on ACP and SN was quite different although c_{ij} for these two measures on G scores is $c_{ij} = 0.928$ (result not shown).

CONCLUSIONS

This paper is aimed at highlighting some problems in using different measures of success of predictor programs and

Table 4: Ranking of different programs based on ten different measures. The upper number in a cell denotes the relative rank position of the 12 compared programs. The lower number in the cell denotes the actual numerical value that the considered measure produced for the achieved program score. For example, the TSSG program is ranked in third place (among the 12 compared programs) by the K_2 measure, where the K_2 value for the score of this program is 1,149. When more than one upper number appear in a cell, this indicates the relative ranking positions shared by other programs according to the considered measure. For example, according to sensitivity measure SN, programs Autogene, PromoterFind and TSSG share the relative rank positions 7, 8 and 9 among all 12 compared programs. The ranking positions by ASM is obtained from the relative positions of the considered programs based on all G scores

	Q	K_2	$\phi_2(CC)$	GDIP ₁	GDIP ₂	GDIP ₃	S_p	SP	SN	ASM
Audic	11 0.9948	6 927.8	12 0.1650	5 0.0007892	10 7.6160	5 0.0007893	5 0.9993	10 0.1316	11 0.2083	10 2,029
Autogene	10 0.9949	11 709.3	10 0.1870	11 0.0011150	11 7.6800	11 0.001115	11 0.9989	11 0.1207	7,8,9 0.2917	11 2,166
Promoter2.0	6 0.9975	9 846.4	6 0.2799	8 0.0009374	6 4.5220	8 0.0009376	8 0.999109	8 0.1887	4,5 0.4167	8 1,596
NNPP	8 0.99747	12 580.9	4 0.2872	12 0.0015110	8 5.6030	12 0.0015110	12 0.9985	9 0.1529	1 0.5417	9 1,906
PromoterFind	9 0.9971	4 1,049	8 0.2377	4 0.0006966	7 4.8020	4 0.0006967	4 0.9994	6 0.1944	7,8,9 0.2917	8 1,600
PromoterScan	3 0.9983	1 1,788	9 0.2039	1 0.0004524	9 7.2800	1 0.0004525	1 0.9999	1,2 0.3333	12 0.1250	5 1,488
TATA	12 0.9942	10 742.1	11 0.1676	10 0.0010430	12 8.3880	10 0.0010440	10 0.9990	12 0.1132	10 0.2500	12 2,230
TSSG	7 0.99748	3 1,149	7 0.2522	3 0.0006265	4 4.3190	3 0.0006266	3 0.995	4 0.2188	7,8,9 0.2917	4 1,486
TSSW	5 0.9976	7,8 861.5	5 0.2826	7 0.0009177	5 4.4270	7 0.0009179	7 0.99913	7 0.1923	4,5 0.4167	6 1,572
HMM	2 0.9984	5 946.1	1 0.3425	6 0.0008457	2 3.400	6 0.0008459	6 0.9992	3 0.2353	2,3 0.5000	2 1,301
SPANN1	4 0.9982	7,8 861.5	3 0.3268	9 0.0009453	3 3.8010	9 0.0009456	9 0.99908	5 0.2143	2,3 0.5000	3 1,415
SPANN2	1 0.9987	2 1,508	2 0.3330	2 0.0004688	1 2.8280	2 0.0004689	2 0.9997	1,2 0.3333	6 0.3333	1 1,088

comparison results

proposes a remedy. From the example presented, it is obvious that the question raised about the possibility of finding out which program performs the best cannot be correctly answered. Any grading of achieved program scores will directly depend on the measures of success used, and thus it has only relative significance. The practical hint, however, would be to use a greater number of measures that theoretically produce different results in ranking, or to include in the pool of measures only those measures with the mutual correlation coefficient c_{ij} obtained on ranking all G scores less than a

preselected threshold. It may happen, as in the example given, that some of the measures used produce the same ranking, which is in compliance with some of our predictions made previously. It must however be made clear that with the different selection of measures of prediction success, a different overall ranking may be achieved. Thus, the results given here cannot be taken as the absolute resolution of the ranking problem of predictor programs. However, one general conclusion follows: by using a greater number of mutually different and

reasonable measures of the prediction success, the more appropriate the ASM ranking will be. As a final conclusion, the ASM measure, by its nature, provides more reliability in the assessment of prediction score quality.

Acknowledgement

The author expresses his thanks to the reviewers and to Artemis G. Hatzigeorgiou for critical and constructive comments regarding the text. These have helped in clarifying some of the more difficult aspects of this paper.

References

- Baldi, P. and Brunak, S. (1998), 'Bioinformatics: The Machine Learning Approach', MIT Press, Cambridge, MA.
- Fickett, J. W. and Hatzigeorgiou, A. G. (1997), 'Eukaryotic promoter recognition', *Genome Res.*, Vol. 7(9), pp. 861–878.
- Burset, M. and Guigo, R. (1996), 'Evaluation of gene structure prediction programs', *Genomics*, Vol. 34, pp. 353–367.
- Claverie, J. M. (1997), 'Computational methods for the identification of genes in vertebrate genomic sequences', *Human Molecular Genetics*, Vol. 6(10), pp. 1735–1744.
- Fickett, J. W. and Tung, C.-S. (1992), 'Assessment of protein coding measures', *Nucleic Acids Res.*, Vol. 20, pp. 6441–6450.
- Anderberg, M. R. (1973), 'Cluster Analysis for Applications', Academic Press, New York.
- Conover, W. J. (1980), 'Practical Nonparametric Statistics' (2nd edn), John Wiley, New York.
- Everitt, B. S. (1977), 'The Analysis of Contingency Tables', Halsted Press, New York.
- Fisher, R. A. (1922), 'On the interpretation of χ^2 from the contingency tables and the calculation of P ', *J. Royal Statist. Soc.*, Vol. 85, pp. 87–94.
- Gibbons, J. D. (1976), 'Nonparametric Methods for Quantitative Analysis', Holt, Rinehart and Winston, New York.
- Sneath, P. H. A. and Sokal, R. R. (1973), 'Numerical Taxonomy', Freeman, San Francisco.
- Yule, G. U. (1912), 'On the methods of measuring the association between two attributes', *J. Royal Statist. Soc.*, Vol. 75, pp. 579–642.
- Yule, G. U. (1917), 'An Introduction to the Theory of Statistics', Griffin, London.
- Zar, J. H. (1984), 'Biostatistical Analysis', (2nd edn), Prentice Hall, Englewood Cliffs, NJ.
- Pedersen, A. G., Baldi, P., Chauvin, Y. and Brunak, S. (1999), 'The biology of eukaryotic promoter prediction – a review', *Computers Chem.*, Vol. 23, pp. 191–207.
- Cramer, H. (1946), 'Mathematical Methods of Statistics', Princeton University Press, Princeton, NJ.
- Ives, K. H. and Gibbons, J. D. (1976), 'A correlation measure for nominal data', *Amer. Statist.*, Vol. 21(5), pp. 16–17.
- Uberbacher, E. C. and Mural, R. J. (1991), 'Locating protein-coding regions in human DNA sequences by a multiple sensor-neural approach', *Proc. Natl Acad. USA*, Vol. 88, pp. 11261–11265.
- Dong, S. and Searlis, D. B. (1994), 'Gene structure prediction by linguistic methods', *Genomics*, Vol. 23, pp. 540–551.
- Guigo, R., Knudsen, S., Drake, N. and Smith, T. F. (1992), 'Prediction of gene structure', *J. Mol. Biol.*, Vol. 226, pp. 141–157.
- Snyder, E. E. and Stormo, G. D. (1993), 'Identification of coding regions in genomic DNA sequences: an application of dynamic programming and neural networks', *Nucleic Acids Res.*, Vol. 21, pp. 607–613.
- Bajić, V. B. and Bajić, I. V. (1999), 'ANN in DNA regulatory region recognitions: The case of promoters, Tutorial, CD edition', International Joint Conference on Neural Networks, Washington, DC, 10–16th July.
- Bajić, V. B. and Bajić, I. V. (2000), 'Neural network system for promoter recognition', in 'Future Directions for Intelligent Systems and Information Science', Kasabov, N., Ed., Physica-Verlag, New York.
- Knudsen, S. (1999), 'Promoter2.0: for the recognition of Pol II promoter sequences', *Bioinformatics*, Vol. 15(5), pp. 356–361.
- Ohler, U., Harbeck, S., Niemann, H. *et al* (1999), 'Interpolated Markov chains for eukaryotic promoter recognition', *Bioinformatics*, Vol. 15(5), pp. 362–369.
- Audic, S. and Claverie, J.-M. (1997), 'Detection of eukaryotic promoters using Markov transition matrices', *Computer Chem.*, Vol. 21(4), pp. 223–227.
- Kondrakhin, Y. V., Kel, A. E., Kolchanov, N. A. *et al.* (1995), 'Eukaryotic promoter recognition by binding sites for transcription factors', *Computer Applic. Biosci.*, Vol. 11, pp. 477–488.
- Hutchinson, G. B. (1996), 'The prediction of vertebrate promoter regions using differential hexamer frequency analysis', *Computer Applic. Biosci.*, Vol. 12, pp. 391–398.
- Bucher, P. (1990), 'Weight matrix descriptions of four eukaryotic RNA

- polymerase II promoter elements derived from 502 unrelated promoter sequences', *J. Mol. Biol.*, Vol. 212, pp. 563–578.
30. Solovyev, V. and Salamov, A. (1997), 'The Gene-Finder computer tools for analysis of human and model organisms genome sequences', in 'Proceedings of the Fifth International Conference on Intelligent Systems for Molecular Biology', Gaaserland, T., Karp, P., Karplus, K., Ouzounis, C., Sander, K. and Valencia, A., Eds, ISMB97, AAAI Press, Menlo Park, CA, pp. 294–302.
31. Reese, M., NNPP program Internet address (<http://www-hgc.lbl.gov/projects/promoter.html>).
32. Reese, M. G. and Eeckman, F. H. Time-delay neural networks for eukaryotic promoter prediction, submitted, 1999.
33. Prestridge, D. S. (1995), 'Predicting Pol II promoter sequences using transcription factor binding sites', *J. Mol. Biol.*, Vol. 249, pp. 923–932.