Supplementary Material for: "Exploring genomic dark matter: homology search for non-coding RNA"

Eva K. Freyhult[1], Jonathan P. Bollback[2] & Paul P. Gardner[2,3]

[1] *The Linnaeus Centre for Bioinformatics, Uppsala University, Box 598,75124 Uppsala, Sweden.*
[2] *Molecular Evolution Group, The Institute of Molecular Biology and Physiology, University of Copenhagen, Universitetsparken 15, Copenhagen, Denmark.*
[3] *Correspondence should be addressed to PPGardner@bi.ku.dk*

Supplementary methods

Phylogenetic Sequence Reconstruction Ancestral sequences were stochastically sampled for each internal node of a phylogeny relating the search sequences (see Figure 1). An empirical Bayesian approach was adopted in which calculation of the posterior probabilities was conditioned on point estimates for the phylogeny, branch lengths, and model parameters[1]. MrBayes v3.1.1[2] was used to obtain a Bayesian consensus tree (τ), posterior expectation of branch lengths (v), and substitution model parameters (θ). MrBayes was run for 1 million cycles with a single Markov chain using an RNA doublet model derived from the Schöniger and von Haussler model[3] for paired sites (stems) and the general-time-reversible model[4] for unpaired sites (loops). The all-compatibility consensus tree and posterior expectations (v and θ) were obtained after excluding the first 500,000 cycles as burn-in.

Ancestral states are stochastically sampled from the posterior distribution at each internal node of the consensus tree. Specifically, the probability of the ancestral state i for site k is calculated as,

$$Pr(y_k = i \mid \mathbf{x}_k, \hat{\tau}, \hat{v}, \hat{\theta}) = \frac{l_{k,i}\pi_i I(y_k = i)}{\sum_{j \in \psi_k} l_{k,j}\pi_j I(y_k = j)} \quad (1)$$

where $l_{k,i}$ is the conditional likelihood, π_i is the frequency of i, ψ_k is the possible set of states at site k, and $\mathbf{x_k}$ is a column vector of the observations at site k. If site *k* is a doublet then ψ_k contains 16 possible states ($A{\cdot}A$, $A{\cdot}C$, $A{\cdot}G$, $\cdots U{\cdot}G$, $U{\cdot}U$), otherwise, ψ_k is the set of 4 possible nucleotides. A prior is used to constrain sampled ancestral states using the indicator function $I(y_k = i)$,

$$I(y_k = i) = \begin{cases} q & \text{site } k \text{ is a doublet, } i = \text{WC or GU} \\ 1 - q & \text{site } k \text{ is a doublet, } i \neq \text{WC or GU} \\ 1 & \text{site } k \text{ is a singleton.} \end{cases} \quad (2)$$

where, q is the observed frequency of Watson-Crick (WC) or wobble (GU) pairs at site k.

The purpose of using reconstructed sequences to enhance homology searches rests on expanding the search sequence diversity in a way that captures the diversity in unidentified orthologs. Unfortunately, when the diversity of search sequences is low ancestral reconstructions alone are unlikely to capture divergent homologs. In order to expand search sequence diversity a posterior predictive method was developed to simulate more divergent search sequences (see Supplementary Figure 1).

Using samples of the model parameters from the posterior (as described above) a predictive sequence is sampled by simulating an imaginary lineage originating from each internal node of the search sequence phylogeny. In the absence of an estimate of the evolutionary distance from the internal node to the tip of the imaginary lineage each branch is assigned an *a priori* length, λ, that has been optimized for the level of divergence among a larger subset of orthologs. The probability of observing a state at the tip of the new lineage is the product of the probability of sampling a state at an internal node and the probability of changing from one state to another along the lineage. Formally, sequences are simulated by stochastically sampling from the following distribution,

$$Pr(z_k = j \mid y_k = i, \mathbf{x}_k, \hat{\tau}, \hat{v}, \hat{\theta}) = Pr(y_k = i \mid \mathbf{x}_k, \hat{\tau}, \hat{v}, \hat{\theta}) p_{ij}(\lambda) \quad (3)$$

where the simulated observation for site k at the imaginary tip node is z_k, y_k is the state at the parent node, λ is the branch length in terms of the expected number of substitutions, and p_{ij} is the transition probability of changing from *i* to *j*. Transition probabilities are derived from the same models used in reconstructing ancestral states. It should be noted that when $\lambda = 0$ equation 3 gives the same results as equation 1.

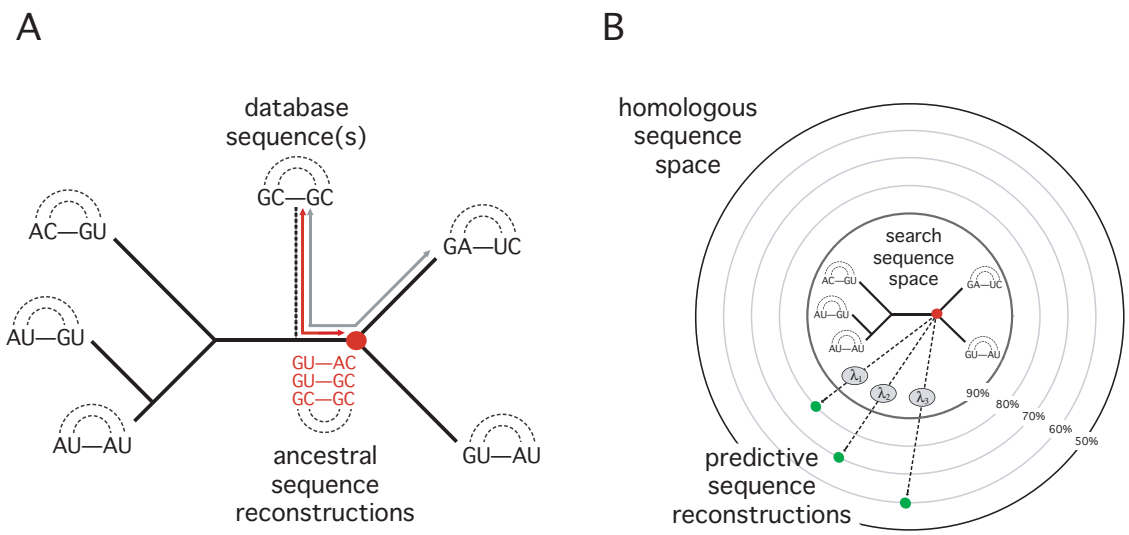


Figure 1: Schematic justification for the use of ancestral state reconstructions (A: ASR) and predictive sequence reconstructions (B: PSR) to enhance homology search. (A) Ancestral state reconstruction of a stem pair demonstrating their use to augment homology discovery. The red and grey arrowed lines represent the distance between the nearest search sequence and an unknown homolog (sampling described in the text). (B) Predictive sequence reconstructions augment homology discovery by sampling highly probable sequences at varying divergences from the back bone nodes of the phylogeny relating the search sequences; divergence, (λ_i), of the predictive lineages is optimized on training sets to balance gains in sensitivity and loss in specificity.

Supplementary Figure 1 - Phylogenetic sequence reconstruction

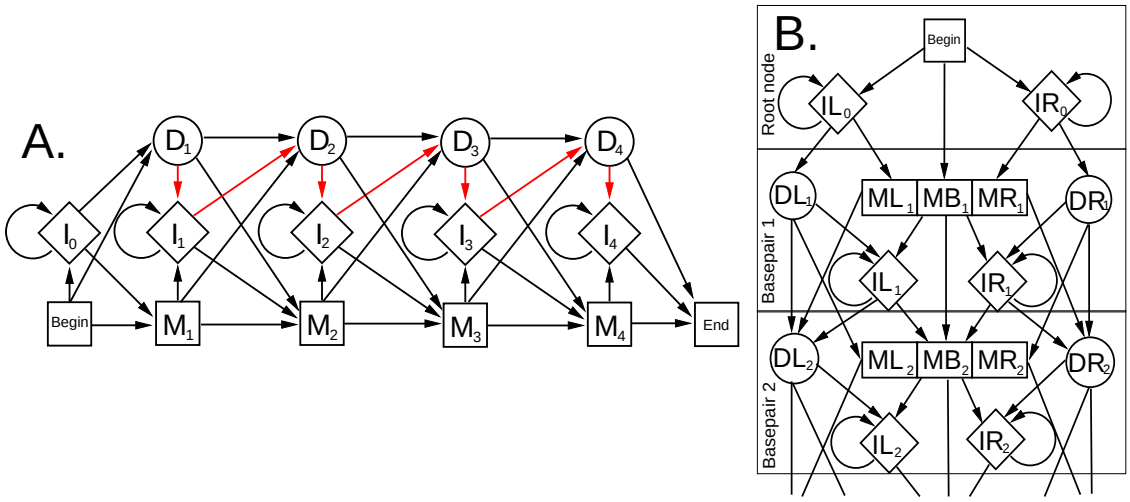


Figure 2: **A.** An illustration of the profile hidden Markov model (HMM) architecture proposed by Krogh *et al.* (1994). States labelled "M" capture matches between database sequences and the query profile, states labelled "I" allow for insertions in the database sequences relative to the profile and states labelled "D" allow for deletions in the database sequences relative to the profile. The insertion to deletion and vice versa states (highlighted in red) are absent from the HMMer2 'Plan 7' architecture, which results in improved speed at a slight cost to accuracy. **B.** An illustration of 2 basepairs of a covariance model (CM) as proposed by Eddy & Durbin[5]. A CM is a generalization of a profile HMM that allows basepairs to be modeled. In the above representation of a CM: "MB" states match both the 5' & 3' bases in a basepair, "ML" and "MR" states match only the 5' or 3' bases respectively. "DL" and "DR" states model deletions of either a 5' or a 3' base respectively. "ML" states are tied to "DR" states and "MR" states are tied to "DL" states. "IL" and "IR" states allow for insertions between bases in either the 5' or 3' side of a basepair. Loops in the secondary structure degenerate to a profile HMM.

Supplementary Figure 2 - Examples of Profile HMM and covariance model architectures

Supplementary results

Tables

Program	Sensitivity	Specificity	MCC	Ave. MCC Rank	
				Median	Mean
Sequence based methods					
NCBI-BLAST	(0.23,**0.59,**0.88)	(0.995,**0.997**,0.999)	(0.44,**0.73**,0.90)	31.0	29.37
NCBI-BLAST (W7)	(0.41,**0.70,**0.93)	(0.981,**0.987**,0.994)	(0.57,**0.74**,0.86)	30.0	27.75
NCBI-BLAST (W7,65%)	(0.40,**0.88,**0.96)	(0.998,**0.999**,1.000)	(0.61,**0.93**,0.96)	17.0	15.84
WU-BLAST	(0.30,**0.63,**0.89)	(0.996,**0.998**,0.999)	(0.52,**0.76**,0.91)	28.0	26.77
WU-BLAST (W7)	(0.37,**0.92,**0.97)	(0.999,**0.999**,1.000)	(0.59,**0.95**,0.97)	15.0	14.51
WU-BLAST (W3)	(0.36,**0.93,**0.96)	(0.999,**1.000**,1.000)	(0.59,**0.95**,0.97)	11.0	13.46
WU-BLAST (W7,65%)	(0.39,**0.88,**0.96)	(0.999,**1.000**,1.000)	(0.60,**0.93**,0.96)	17.0	15.68
WU-BLAST (W7,PUPY)	(0.41,**0.84,**0.97)	(0.994,**0.995**,0.999)	(0.62,**0.88**,0.94)	23.0	21.36
FASTA	(0.43,**0.90,**0.96)	(0.998,**0.999**,1.000)	(0.64,**0.94**,0.97)	14.0	13.04
FASTA [ASR]	(0.52,**0.94,**0.98)	(0.994,**0.997**,0.999)	(0.70,**0.92**,0.95)	12.0	14.86
FASTA (ktup6,65%)	(0.40,**0.89,**0.96)	(0.999,**0.999**,1.000)	(0.61,**0.93**,0.97)	16.0	14.45
FASTA (U)	(0.31,**0.89,**0.97)	(0.997,**0.998**,1.000)	(0.54,**0.92**,0.96)	19.0	19.05
FASTA (RIBOSUM)	(0.06,**0.25,**0.58)	(0.993,**0.999**,1.000)	(0.15,**0.47**,0.64)	34.0	34.24
FASTA (FOLDALIGN)	(0.11,**0.74,**0.97)	(0.989,**1.000**,1.000)	(0.32,**0.84**,0.91)	27.0	27.59
ParAlign	(0.27,**0.64,**0.77)	(0.996,**0.998**,0.999)	(0.49,**0.76**,0.84)	30.0	29.36
ParAlign (65%)	(0.38,**0.82,**0.89)	(0.999,**1.000**,1.000)	(0.60,**0.89**,0.93)	21.0	19.65
SSEARCH	(0.38,**0.90,**0.95)	(0.999,**1.000**,1.000)	(0.60,**0.94**,0.97)	12.0	13.49
SSEARCH [ASR]	(0.46,**0.93,**0.97)	(0.997,**0.999**,1.000)	(0.66,**0.94**,0.96)	13.0	12.77
SSEARCH (65%)	(0.38,**0.88,**0.95)	(0.999,**1.000**,1.000)	(0.60,**0.93**,0.97)	14.0	14.30
SSEARCH (U)	(0.27,**0.89,**0.96)	(0.995,**0.997**,1.000)	(0.50,**0.91**,0.95)	22.0	21.79
Profile HMM methods					
HMMer (1.8.4,local)	(0.50,**0.86,**0.95)	(1.000,**1.000**,1.000)	(0.69,**0.92**,0.97)	13.0	13.77
HMMer (1.8.4,global)	(0.70,**0.85,**0.95)	(0.992,**0.997**,1.000)	(0.76,**0.90**,0.95)	19.0	16.14
HMMer (2.3.2,local)	(0.59,**0.89,**0.96)	(0.998,**0.999**,0.999)	(0.75,**0.93**,0.96)	11.0	12.40
HMMer (2.3.2,global)	(0.74,**0.88,**0.97)	(0.991,**0.996**,0.999)	(0.77,**0.92**,0.96)	12.0	13.38
SAM (local)	(0.48,**0.91,**0.96)	(1.000,**1.000**,1.000)	(0.67,**0.95**,0.97)	**10.0**	10.63
SAM (global)	(0.40,**0.88,**0.94)	(0.999,**1.000**,1.000)	(0.61,**0.93**,0.96)	14.0	15.73
SAM (model) + HMMer (2.3.2,search)	(0.68,**0.90,**0.97)	(0.982,**0.995**,0.999)	(0.67,**0.91**,0.95)	17.0	17.14
Structure enhanced methods					
ERPIN (4)	(0.01,**0.01,**0.03)	(1.000,**1.000**,1.000)	(0.07,**0.10**,0.15)	37.0	36.93
ERPIN (5)	(0.05,**0.15,**0.27)	(1.000,**1.000**,1.000)	(0.21,**0.37**,0.50)	35.0	34.69
ERPIN [PSR]	(0.09,**0.17,**0.29)	(1.000,**1.000**,1.000)	(0.28,**0.39**,0.51)	35.0	33.48
Infernal (0.7)	(0.94,**0.98,**1.00)	(0.996,**1.000**,1.000)	(0.95,**0.97**,1.00)	**2.0**	5.79
Infernal (0.7,local)	(0.91,**0.98,**1.00)	(0.999,**1.000**,1.000)	(0.95,**0.98**,0.99)	**3.0**	5.19
Infernal (0.55)	(0.32,**0.38,**0.52)	(1.000,**1.000**,1.000)	(0.55,**0.60**,0.71)	32.0	30.70
RAVENNA (ML)	(0.61,**0.82,**0.94)	(1.000,**1.000**,1.000)	(0.77,**0.90**,0.97)	**8.0**	11.74
RAVENNA (ML,local)	(0.64,**0.89,**0.96)	(0.997,**0.998**,0.999)	(0.78,**0.92**,0.96)	13.0	13.38
RSEARCH	(0.93,**0.99,**1.00)	(0.999,**1.000**,1.000)	(0.96,**0.99**,1.00)	**2.0**	4.57
RSmatch	(0.31,**0.42,**0.51)	(0.997,**0.999**,1.000)	(0.53,**0.62**,0.68)	32.0	26.70

Table 1: Tabulated results of searches with **five** sequences. From left to right column one contains program names and settings, column two sensitivity, column three specificity, column four Matthew's correlation coefficient (MCC) and column five contains both a median and mean ranking determined by the MCC (see the Methods for definitions). Sensitivity, specificity and MCC are summarised as 3-tuples displaying the lower quartile, median and upper quartile. Median MCC rankings below ten (one being the best) are indicated with bold font. See the Methods section and Supplementary Tables 5&6 for algorithm settings.

Program	Sensitivity	Specificity	MCC	Ave. MCC Rank	
				Median	Mean
Sequence based methods					
NCBI-BLAST	(0.53,**0.71,**0.94)	(0.978,**0.987**,0.993)	(0.67,**0.76**,0.85)	29.0	29.51
NCBI-BLAST (W7)	(0.69,**0.83,**1.00)	(0.924,**0.951**,0.974)	(0.67,**0.70**,0.72)	31.0	30.89
NCBI-BLAST (W7,65%)	(0.83,**0.96,**1.00)	(0.991,**0.997**,0.998)	(0.89,**0.94**,0.96)	16.0	15.74
WU-BLAST	(0.58,**0.76,**0.98)	(0.977,**0.991**,0.994)	(0.72,**0.80**,0.88)	27.0	26.83
WU-BLAST (W7)	(0.88,**0.97,**0.98)	(0.989,**0.997**,0.998)	(0.90,**0.93**,0.97)	20.0	16.47
WU-BLAST (W3)	(0.89,**0.97,**0.98)	(0.990,**0.997**,0.999)	(0.91,**0.93**,0.97)	19.0	15.92
WU-BLAST (W7,65%)	(0.83,**0.96,**1.00)	(0.991,**0.998**,0.999)	(0.90,**0.94**,0.96)	17.0	15.86
WU-BLAST (W7,PUPY)	(0.77,**0.97,**1.00)	(0.973,**0.981**,0.985)	(0.81,**0.87**,0.89)	26.0	23.38
FASTA	(0.85,**0.97,**1.00)	(0.988,**0.996**,0.998)	(0.88,**0.93**,0.96)	15.0	15.92
FASTA (ktup6,65%)	(0.83,**0.96,**1.00)	(0.991,**0.997**,0.998)	(0.89,**0.94**,0.96)	16.0	14.69
FASTA (U)	(0.84,**0.97,**0.98)	(0.989,**0.993**,0.996)	(0.89,**0.93**,0.94)	21.0	20.46
FASTA (RIBOSUM)	(0.25,**0.57,**0.79)	(0.936,**0.993**,0.998)	(0.46,**0.54**,0.69)	33.0	32.85
FASTA (FOLDALIGN)	(0.69,**0.85,**0.98)	(0.988,**1.000**,1.000)	(0.82,**0.89**,0.93)	23.0	21.74
ParAlign	(0.57,**0.76,**0.93)	(0.979,**0.988**,0.992)	(0.70,**0.78**,0.85)	28.0	28.42
ParAlign (65%)	(0.81,**0.93,**0.96)	(0.994,**0.999**,1.000)	(0.89,**0.93**,0.96)	16.0	15.43
SSEARCH	(0.83,**0.96,**0.98)	(0.994,**0.999**,0.999)	(0.90,**0.95**,0.97)	12.0	12.68
SSEARCH (65%)	(0.81,**0.96,**0.97)	(0.994,**0.999**,0.999)	(0.89,**0.95**,0.97)	12.0	13.36
SSEARCH (U)	(0.84,**0.96,**0.98)	(0.984,**0.989**,0.995)	(0.86,**0.91**,0.93)	25.0	23.68
Profile HMM methods					
HMMer (1.8.4,local)	(0.71,**0.90,**0.95)	(0.999,**1.000**,1.000)	(0.83,**0.95**,0.97)	11.0	12.08
HMMer (1.8.4,global)	(0.74,**0.84,**0.90)	(1.000,**1.000**,1.000)	(0.85,**0.91**,0.94)	16.0	15.19
HMMer (2.3.2,local)	(0.74,**0.94,**0.97)	(0.996,**0.997**,0.998)	(0.84,**0.93**,0.97)	12.0	14.40
HMMer (2.3.2,global)	(0.74,**0.86,**0.94)	(1.000,**1.000**,1.000)	(0.85,**0.92**,0.96)	**9.5**	12.89
SAM (local)	(0.74,**0.94,**0.97)	(0.999,**1.000**,1.000)	(0.85,**0.96**,0.98)	**10.0**	10.44
SAM (global)	(0.84,**0.94,**0.98)	(0.997,**0.998**,0.999)	(0.88,**0.95**,0.97)	11.0	11.24
SAM (model) + HMMer (2.3.2,search)	(0.78,**0.90,**0.95)	(0.998,**1.000**,1.000)	(0.83,**0.94**,0.97)	12.0	13.20
Structure enhanced methods					
ERPIN (4)	(0.10,**0.23,**0.42)	(1.000,**1.000**,1.000)	(0.30,**0.47**,0.63)	34.0	33.51
ERPIN (5)	(0.40,**0.52,**0.76)	(1.000,**1.000**,1.000)	(0.62,**0.70**,0.86)	31.0	27.48
Infernal (0.7)	(0.97,**0.99,**1.00)	(0.997,**1.000**,1.000)	(0.97,**0.98**,1.00)	**2.0**	4.29
Infernal (0.7,local)	(0.96,**0.99,**1.00)	(0.999,**1.000**,1.000)	(0.97,**0.99**,1.00)	**2.0**	3.12
Infernal (0.55)	(0.88,**0.93,**0.98)	(1.000,**1.000**,1.000)	(0.93,**0.96**,0.99)	**4.0**	8.38
RAVENNA (ML)	(0.79,**0.94,**0.97)	(1.000,**1.000**,1.000)	(0.88,**0.96**,0.98)	**7.0**	7.90
RAVENNA (ML,local)	(0.80,**0.94,**0.97)	(0.995,**0.998**,0.999)	(0.88,**0.94**,0.96)	14.0	13.29
RSEARCH	(0.84,**0.94,**0.99)	(1.000,**1.000**,1.000)	(0.91,**0.97**,0.99)	**4.0**	6.63
RSmatch	(0.57,**0.62,**0.91)	(0.989,**0.997**,0.998)	(0.72,**0.76**,0.89)	27.0	26.17

Table 2: Tabulated results of searches with **twenty** sequences. From left to right column one contains program names and settings, column two sensitivity, column three specificity, column four Matthew's correlation coefficient (MCC) and column five contains both a median and mean ranking determined by the MCC (see the Methods for definitions). Sensitivity, specificity and MCC are summarised as 3-tuples displaying the lower quartile, median and upper quartile. Median MCC rankings below ten (one being the best) are indicated with bold font. See the Methods section and Supplementary Tables 5&6 for algorithm settings.

Program	Sensitivity	Specificity	MCC	Ave. MCC Rank	
				Median	Mean
Five input sequences					
NCBI-BLAST (W7,65%)	(0.40,**0.88,**0.96)	(0.998,**0.999**,1.000)	(0.61,**0.93**,0.96)	3.0	2.85
WU-BLAST (W7,65%)	(0.39,**0.88,**0.96)	(0.999,**1.000**,1.000)	(0.60,**0.93**,0.96)	3.0	3.00
FASTA (ktup6,65%)	(0.40,**0.89,**0.96)	(0.999,**0.999**,1.000)	(0.61,**0.93**,0.97)	2.0	2.31
ParAlign (65%)	(0.38,**0.82,**0.89)	(0.999,**1.000**,1.000)	(0.60,**0.89**,0.93)	4.0	3.47
SSEARCH (65%)	(0.38,**0.88,**0.95)	(0.999,**1.000**,1.000)	(0.60,**0.93**,0.97)	3.0	2.57
Twenty input sequences					
NCBI-BLAST (W7,65%)	(0.83,**0.96,**1.00)	(0.991,**0.997**,0.998)	(0.89,**0.94**,0.96)	2.00	3.15
WU-BLAST (W7,65%)	(0.83,**0.96,**1.00)	(0.991,**0.998**,0.999)	(0.90,**0.94**,0.96)	3.00	3.38
FASTA (ktup6,65%)	(0.83,**0.96,**1.00)	(0.991,**0.997**,0.998)	(0.89,**0.94**,0.96)	1.00	2.64
ParAlign (65%)	(0.81,**0.93,**0.96)	(0.994,**0.999**,1.000)	(0.89,**0.93**,0.96)	4.00	3.00
SSEARCH (65%)	(0.81,**0.96,**0.97)	(0.994,**0.999**,0.999)	(0.89,**0.95**,0.97)	4.00	2.26

Table 3: Tabulated results of the sequence based methods with identical scoring parameters. From left to right column one contains program names, column two sensitivity, column three specificity, column four Matthew's correlation coefficient (MCC) and column five a median and mean ranking determined by the MCC. Sensitivity, specificity and MCC are summarised as 3-tuples displaying the lower quartile, median and upper quartile.

Program	Sensitivity	Specificity	MCC	Ave. MCC Rank	
				Median	Mean
Five input sequences					
WU-BLAST (W7,65%)	(0.39,**0.88,**0.96)	(0.999,**1.000**,1.000)	(0.60,**0.93**,0.96)	2.0	3.13
WU-BLAST (W7,PUPY)	(0.41,**0.84,**0.97)	(0.994,**0.995**,0.999)	(0.62,**0.88**,0.94)	4.0	4.80
FASTA	(0.43,**0.90,**0.96)	(0.998,**0.999**,1.000)	(0.64,**0.94**,0.97)	1.0	2.17
FASTA (U)	(0.31,**0.89,**0.97)	(0.997,**0.998**,1.000)	(0.54,**0.92**,0.96)	5.0	3.94
FASTA (RIBOSUM)	(0.06,**0.25,**0.58)	(0.993,**0.999**,1.000)	(0.15,**0.47**,0.64)	8.0	7.98
FASTA (FOLDALIGN)	(0.11,**0.74,**0.97)	(0.989,**1.000**,1.000)	(0.32,**0.84**,0.91)	7.0	6.57
SSEARCH	(0.38,**0.90,**0.95)	(0.999,**1.000**,1.000)	(0.60,**0.94**,0.97)	3.0	2.52
SSEARCH (U)	(0.27,**0.89,**0.96)	(0.995,**0.997**,1.000)	(0.50,**0.91**,0.95)	6.0	4.88
Twenty input sequences					
WU-BLAST	(0.58,**0.76,**0.98)	(0.977,**0.991**,0.994)	(0.72,**0.80**,0.88)	5.5	6.02
WU-BLAST (W7,PUPY)	(0.77,**0.97,**1.00)	(0.973,**0.981**,0.985)	(0.81,**0.87**,0.89)	2.0	5.21
FASTA	(0.85,**0.97,**1.00)	(0.988,**0.996**,0.998)	(0.88,**0.93**,0.96)	1.0	2.31
FASTA (U)	(0.84,**0.97,**0.98)	(0.989,**0.993**,0.996)	(0.89,**0.93**,0.94)	4.0	3.42
FASTA (RIBOSUM)	(0.25,**0.57,**0.79)	(0.936,**0.993**,0.998)	(0.46,**0.54**,0.69)	8.0	8.00
FASTA (FOLDALIGN)	(0.69,**0.85,**0.98)	(0.988,**1.000**,1.000)	(0.82,**0.89**,0.93)	7.0	4.62
SSEARCH	(0.83,**0.96,**0.98)	(0.994,**0.999**,0.999)	(0.90,**0.95**,0.97)	**3.0**	1.71
SSEARCH (U)	(0.84,**0.96,**0.98)	(0.984,**0.989**,0.995)	(0.86,**0.91**,0.93)	5.5	4.81

Table 4: Tabulated results comparing sequence based methods with and without RNA specific scoring parameters. From left to right column one contains program names and settings, column two sensitivity, column three specificity, column four Matthew's correlation coefficient (MCC) and column five a median and mean ranking determined by the MCC. Sensitivity, specificity and MCC are summarised as 3-tuples displaying the lower quartile, median and upper quartile.

Thresholds

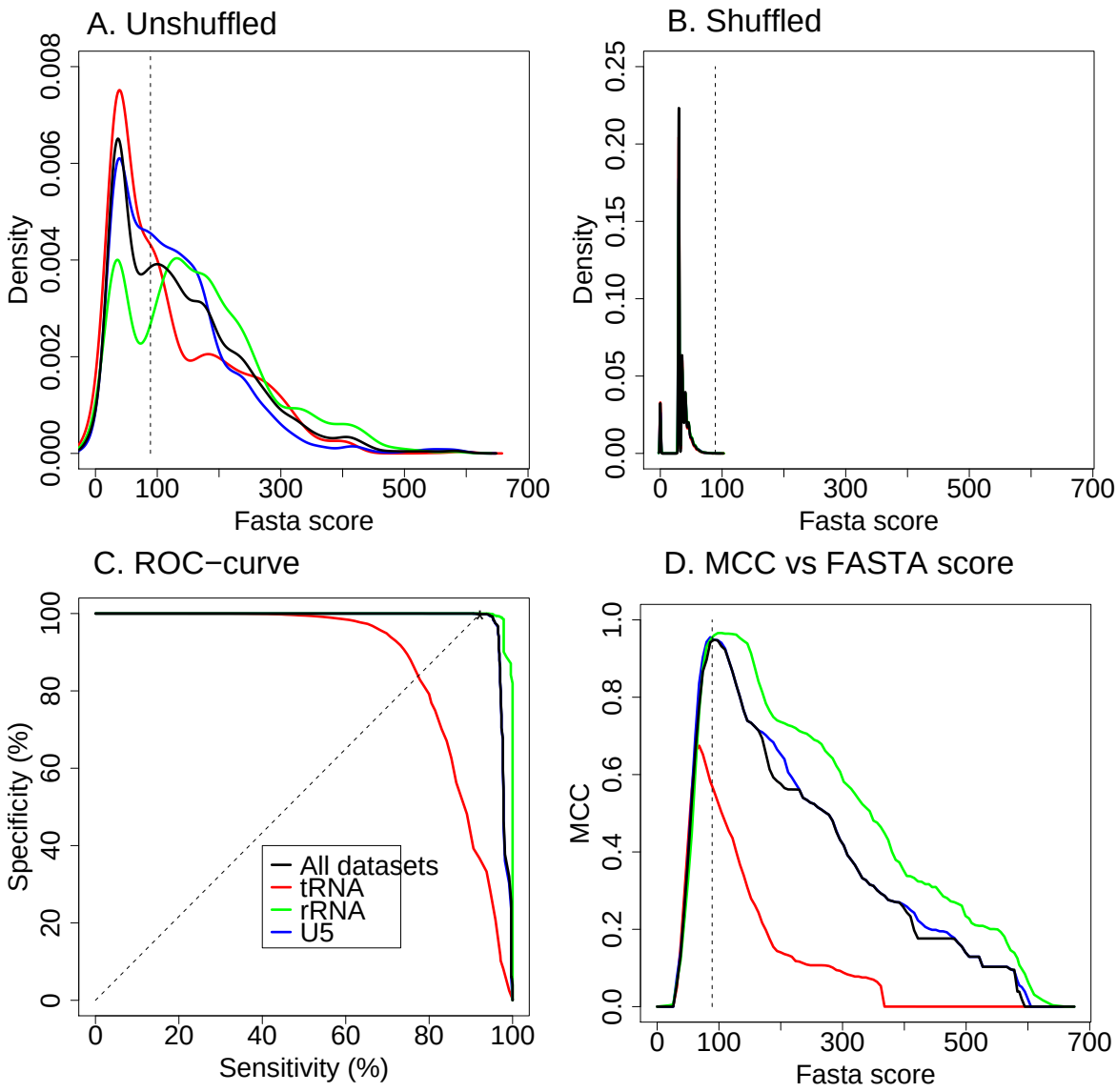
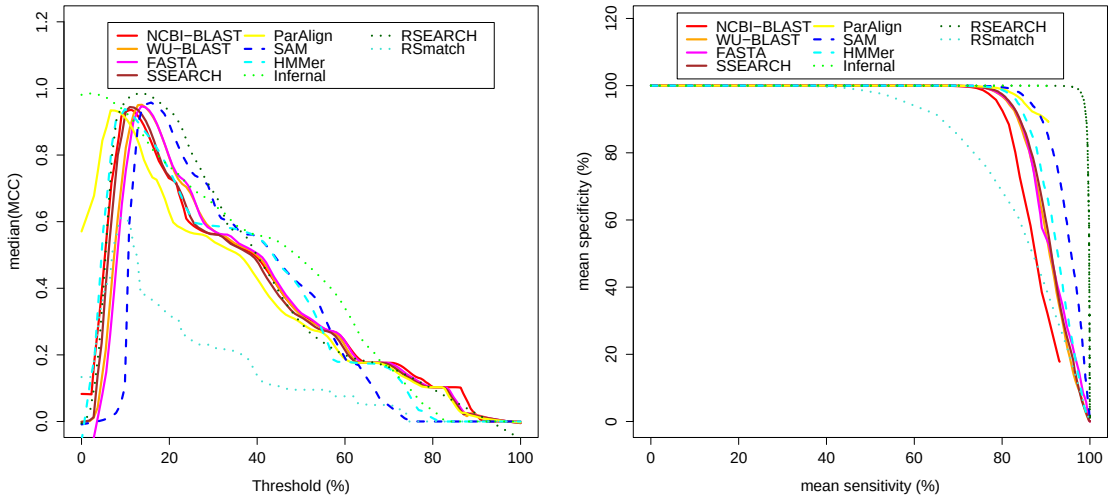


Figure 3:
A FASTA case study for optimal threshold determination. Subplot **A** displays density plots of FASTA score distributions for a selection of inputs from each RNA family (rRNA,tRNA,U5) and identity range (40-60%, 50-70%, 60-80%, 70-85% and 80-95%) when screening the relevant unshuffled database. Subplot **B** similar to subplot **A** except the relevant shuffled database is screened instead. Subplot **C** displays a ROC (receiver operator characteristic) plot showing sensitivity and specificity values over the entire range of possible thresholds. Subplot **D** displays curves of the Matthews correlation coefficient (MCC) as a function of score threshold. The red curves indicate rRNA data, green is tRNA and blue is U5. Black curves show data for all 3 families combined. The dashed black vertical line in each subplot indicates the MCC optimal score threshold (89) over all datasets.



(a) MCC distributions

(b) ROC plots

Figure 4: For each algorithm a score threshold is required. We have optimized these for small structured ncRNAs. For the sake of brevity only the highest ranking parameter settings for each algorithm is shown. (a) Distributions of median MCC scores as a function of algorithm score ranging from 0% filtered to 100% filtered fraction. (b) Receiver Operator Characteristic (ROC) plots for each algorithm.

Algorithm accuracy and computational efficiency

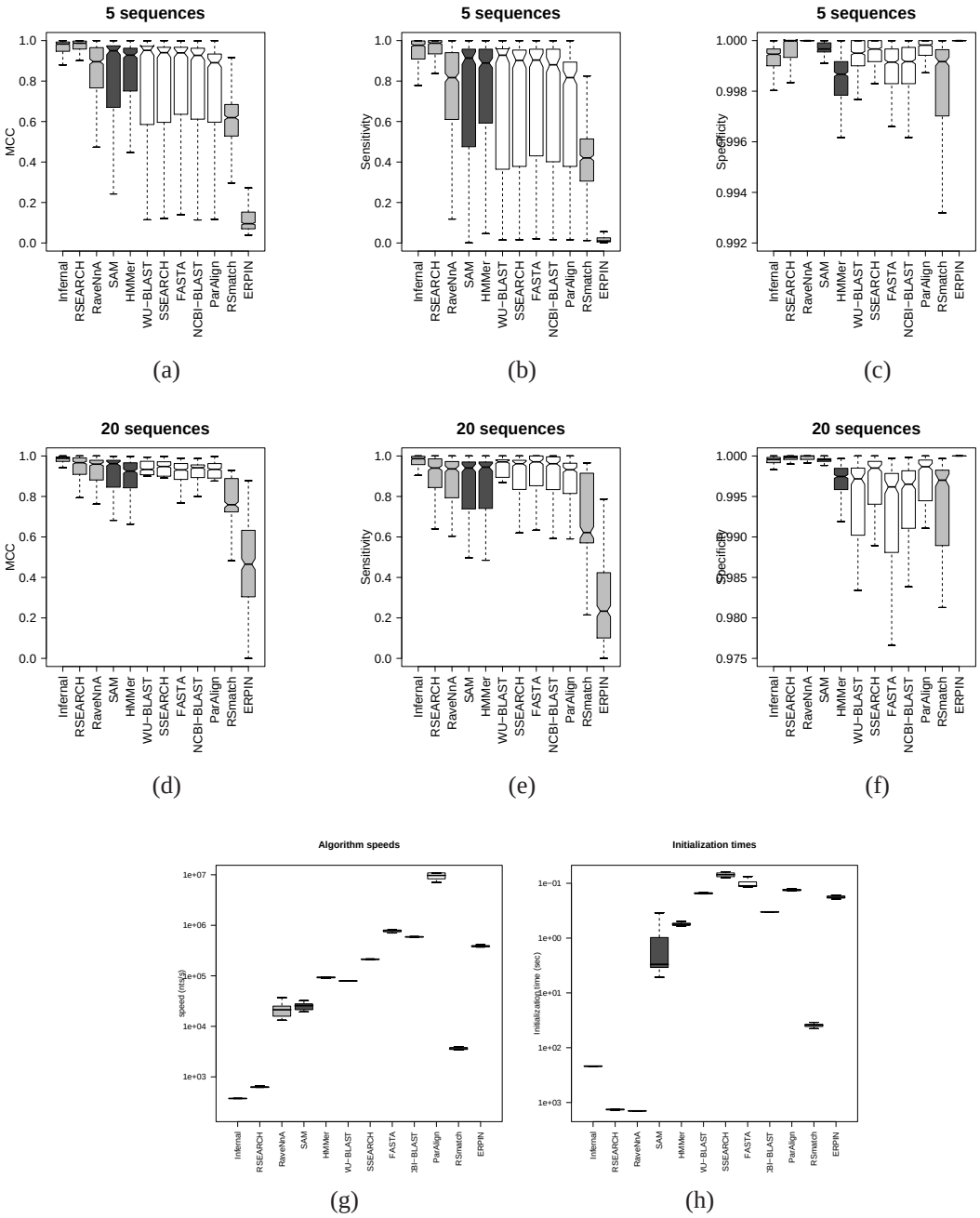


Figure 5: A comparison of the accuracy and efficiencies of homology search methods where only the highest ranking parameter settings for each algorithm from Supplementary Table 1 are shown. These were NCBI-BLAST (W7,65%), WU-BLAST (W3), FASTA, ParAlign (65%), SSEARCH, HMMer (2.3.2,local), SAM (3.5,local), ERPIN, Infernal (0.7,local), RAVENNA, RSEARCH, and RSmatch. (a-c) Boxplots of algorithm accuracies (MCC, sensitivity and specificity respectively) for the 5 sequence datasets. (d-f) Boxplots of algorithm accuracies (MCC, sensitivity and specificity respectively) for the 20 sequence datasets. (g) Boxplots of algorithm speeds in nucleotides per second. (h) Boxplots of initialization times for each algorithm, in seconds.

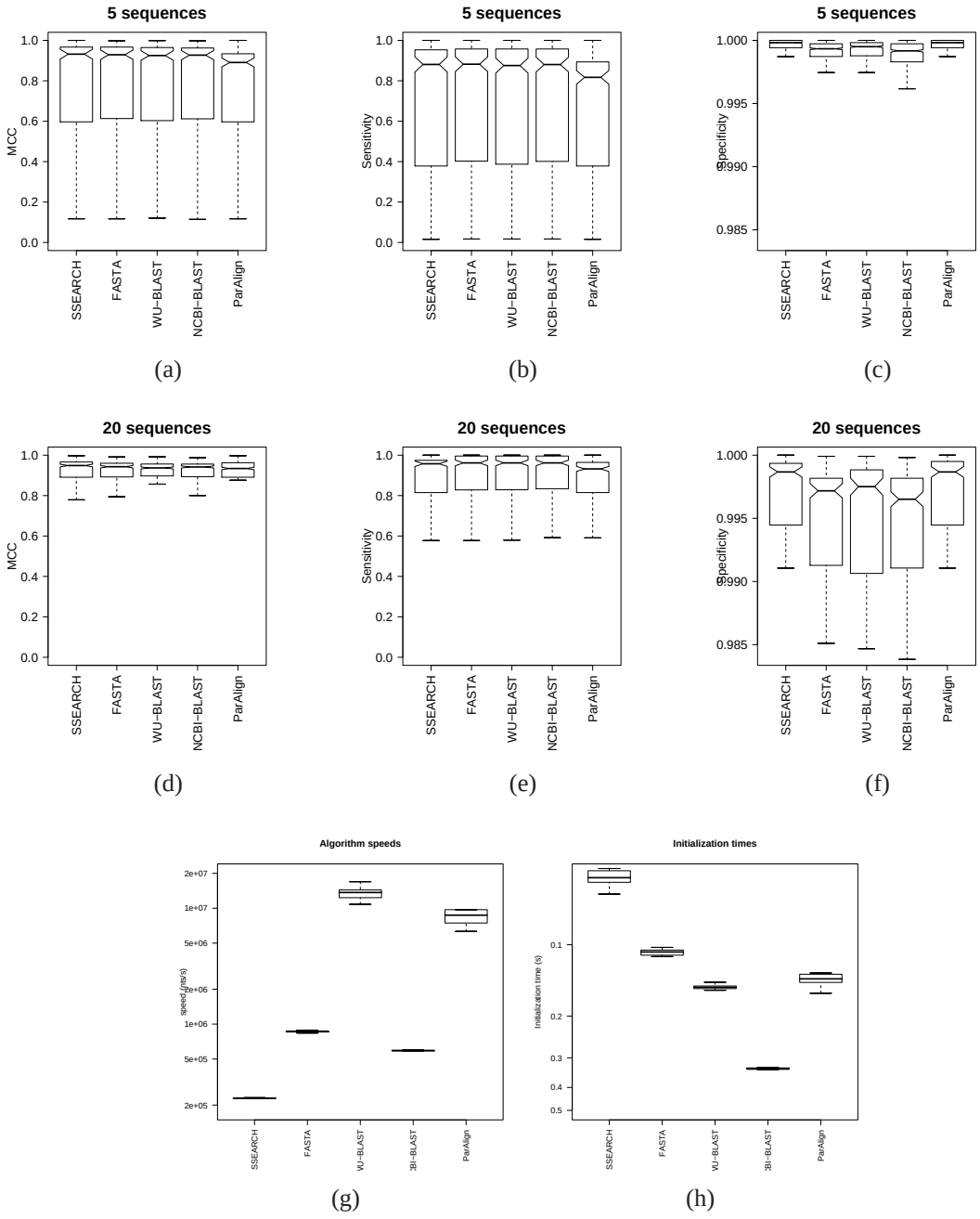
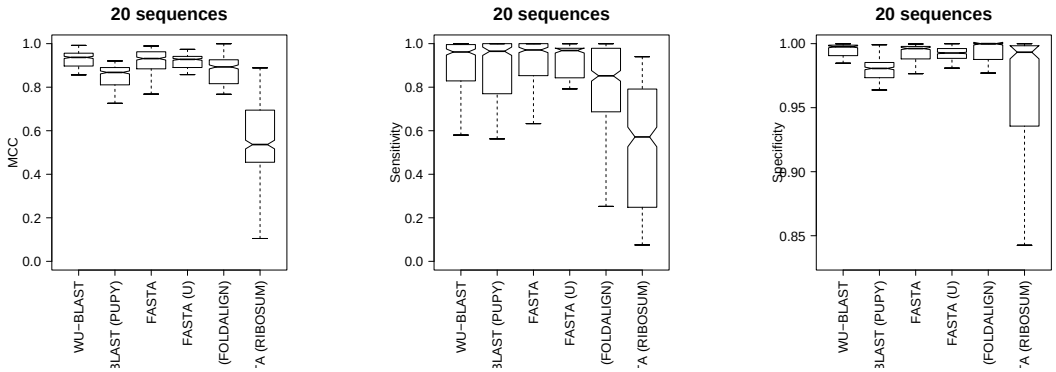
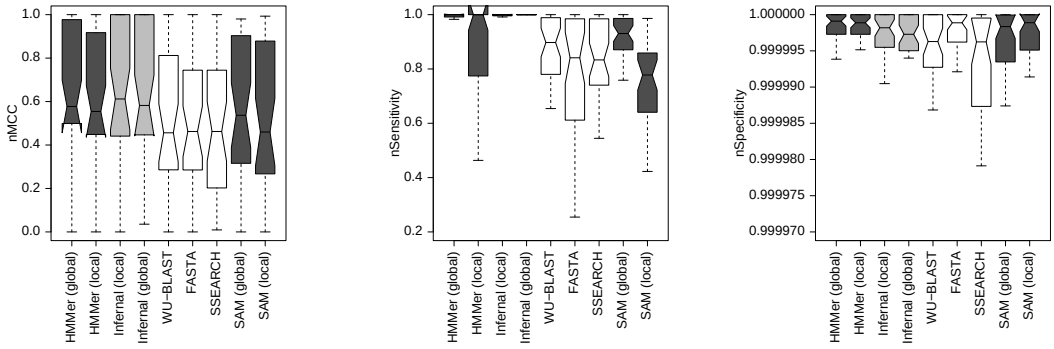


Figure 6: A comparison of the accuracy of sequence based methods with identical scoring parameters. These boxplots show the distributions of the MCC (a&d), sensitivity (b&e) and specificity (c&f) for each of the homology search methods when using a scoring scheme optimized for nucleotide sequences with 65% identity (match=+5, mismatch=-4, gapopen=10,gapextension=10). (g&h) Display algorithm speeds and initialization times respectively.



(a) (b) (c)

Figure 7: A comparison of the accuracy of sequence based methods with score matrices optimized for ncRNA. These boxplots show the distributions of the MCC (a), sensitivity (b) and specificity (c) for each of the homology search methods when using one of WU-BLAST (W7), WU-BLAST (W7,PUPY), FASTA, FASTA (U), FASTA (RIBOSUM) or FASTA (FOLDALIGN). These matrices are discussed in more detail in the text.



(a) (b) (c)

Figure 8: These boxplots indicate the distributions of (a) nMCC, (b) nSensitivity and (c) nSpecificity for a subset of representative programs from each category. Each program was run on a section of human chromosome 12 (coordinates 90,000,000-130,000,000; ver NCBI35).

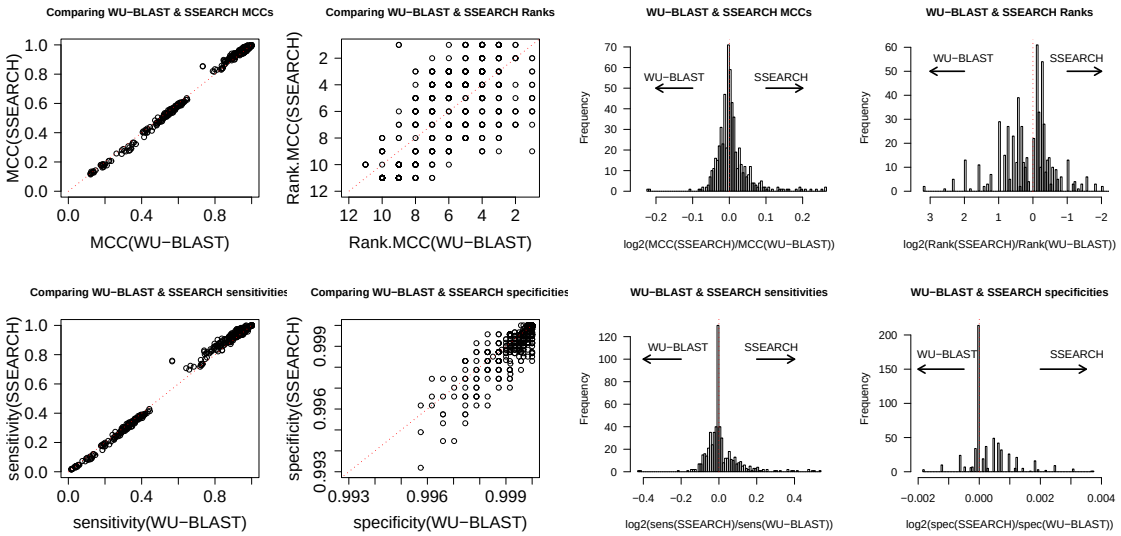


Figure 9: A detailed comparison of the performance of SSEARCH and WU-BLAST (W3). As one can see from the scatter plot of SSEARCH and WU-BLAST MCC values (top left) the performance of these two methods is strongly correlated. The histogram of the logged ratio of SSEARCH and WU-BLAST MCC values (top row, third from the left) shows a distinct skew to the right, indicating that SSEARCH generally outperforms WU-BLAST.

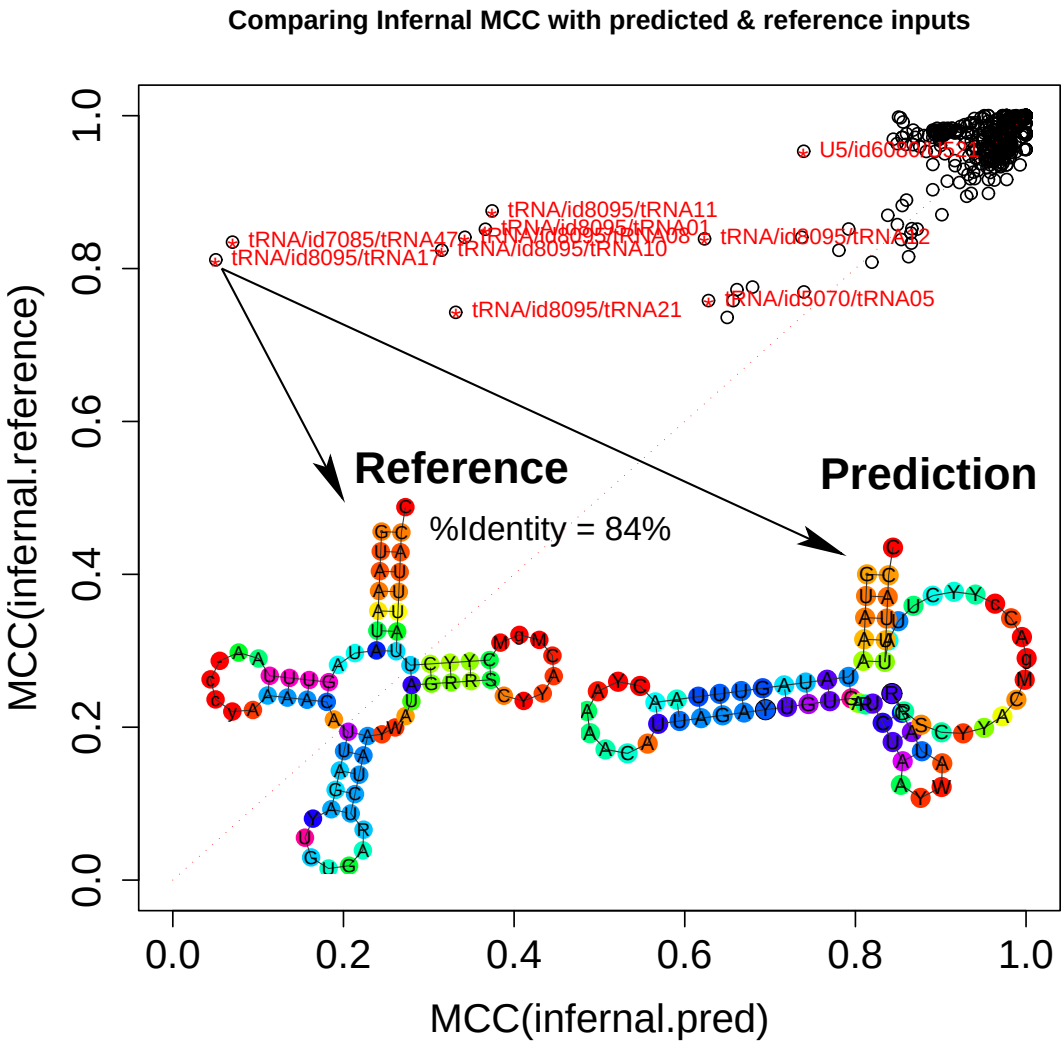


Figure 10: A comparison of Infernal (0.7) performances with inputs derived from either predicted alignments+secondary structures or reference alignments+secondary structures curated by Rfam. Points above the $y = x$ (dashed red) line indicate that Infernal performed better with the reference alignments+secondary structures. The datasets where Infernal performs poorly with the predictions are characterized by high sequence identities and poor secondary structure predictions. An example of one of the worst examples (tRNA/id8095/tRNA17) is shown inset into the figure. The base coloring follows the rainbow of reliable (red) to unreliable (violet) predictions based upon the RNAalifold prediction.

Algorithm parameters

Program	Version	Command	Reference
Sequence based methods			
NCBI-BLAST	2.2.10	`blastall -p blastn -d database.fa -i query.fa -S 1 -b 0 -v 1000000`	6
NCBI-BLAST (W7)		`blastall -p blastn -d database.fa -i query.fa -W 7 -S 1 -b 0 -v 1000000`	
NCBI-BLAST (W7,65%)		`blastall -p blastn -d database.fa -i query.fa -W 7 -r 5 -q -4 -G 10 -E 10 -S 1 -b 0 -v 1000000`	
WU-BLAST	2.0	`blastn database.fa query.fa -noseqs -top B=9999999 V=9999999`	7
WU-BLAST (W7)		`blastn database.fa query.fa -noseqs W=7 -top B=9999999 V=9999999`	
WU-BLAST (W3)		`blastn database.fa query.fa -noseqs W=3 -top B=9999999 V=9999999`	
WU-BLAST (W7,65%)		`blastn database.fa query.fa -noseqs W=7 M=5 N=-4 Q=20 R=10 -top B=9999999 V=9999999`	
WU-BLAST (W7,99%)		`blastn database.fa query.fa -noseqs W=7 M=1 N=-3 Q=5 R=2 -top B=9999999 V=9999999`	
WU-BLAST (W7,PUPY)		`blastn database.fa query.fa -noseqs -matrix=pupy -top W=7 B=9999999 V=9999999`	
FASTA	3.4	`fasta34 -H -n -d 0 -q -3 query.fa database.fa`	8
FASTA (ktup6, 65%)		`fasta34 -H -b 9999999 -n -d 0 -q -3 -f 10 -g 10 -r "+5/-4" query.fa database.fa 6`	
FASTA (-U)		`fasta34 -H -n -d 0 -q -3 -U query.fa database.fa`	
ParAlign	3.4.3	`paralign -g sencel.lic -b 1000 -s 1 -n database.fa query.fa`	9
ParAlign (65%)		`paralign -g sencel.lic -m NT2 -q 10 -r 10 -b 1000 -s 1 -n database.fa query.fa`	
SSEARCH		The same options are used as FASTA.	10, 11

Table 5: This table summarizes the parameters and provides references for the sequence based methods abbreviated in the body of this manuscript.

Program	Version	Command	Reference
Profile HMM methods			

SAM (build)
SAM (local)
SAM (global)
	3.5	`buildmodel name -a RNA -train query.fa`
		`hmmscore name -sw 2 -i name.mod -db database.fa`
		`hmmscore name -sw 0 -i name.mod -db database.fa`
			12, 13

SAM (model) +
HMMer (search)
	3.5, 2.3.2	`buildmodel name -a RNA -train query.fa`
		`sam2hmmer name -i name.mod`
		`hmmsearch name.con.hmm database.fa`	

HMMer (global)
HMMer (local)
HMMer (search)
	2.3.2	`hmmbuild -F -n name name.hmm query.aln`
		`hmmbuild -f -F -n name name.hmm query.aln`
		`hmmsearch name.hmm database.fa`
			14

HMMer HMMer (search) HMMer (search)	1.8.4	`hmmt name.hmm query.fa` `hmmls name.hmm database.fa` (global) `hmmfs name.hmm database.fa` (local)	
Structure enhanced methods			
ERPIN	4.2.5	`erpin query.epn database.fa -begin,end -nomask -fwd -sum SUM.dat -pcw 0.1`	15
Infernal Infernal (local)	0.7 0.7	`cmbuild -F query.cm query.stk; cmsearch --toponly query.cm database.fa` `cmbuild -F query.cm query.stk; cmsearch --toponly --local query.cm database.fa`	16
RAVENNA (ML) RAVENNA (ML,local)	0.2f	`ravenna.pl -filter ML -onlyForwardStrand -global -scoreThreshold 0.00001 -database database.fa -cmFileName query.cm 200` `ravenna.pl -filter ML -onlyForwardStrand -local -scoreThreshold 0.00001 -database database.fa -cmFileName query.cm 200`	17
RSEARCH	1.1	`rsearch -n 1000 -E 100000 -m RIBOSUM85-60.mat query.stk database.fa`	18
RSmatch	1.2	`RSmatch -p dsearch -D database.fa -Q query.fa -n 30000`	19

Table 6: This table summarizes the parameters and provides references for the profile HMM and structure enhanced methods abbreviated in the body of this manuscript.

1. Huelsenbeck, J. P. & Bollback, J. P. Empirical and hierarchical Bayesian estimation of ancestral states. *Syst. Biol.* **50**, 351–366 (2001).

2. Huelsenbeck, J. P. & Ronquist, F. MRBAYES: Bayesian inference of phylogenetic trees. *Bioinformatics* **17**, 754–755 (2001).

3. Schöniger, M. & von Haeseler, A. A stochastic model for the evolution of autocorrelated DNA sequences. *Mol. Phylogenet. Evol.* **3**, 240–247 (1994).

4. Lanavé, C., Preparata, G., Saccone, C. & Serio, G. A new method for calculating evolutionary substitution rates. *J. Mol. Evol.* **20**, 86–93 (1984).

5. Eddy, S. R. & Durbin, R. RNA sequence analysis using covariance models. *Nucleic Acids Res.* **22**, 2079–2088 (1994).

6. Altschul, S. F., Gish, W., Miller, W., Myers, E. W. & Lipman, D. J. Basic local alignment search tool. *J. Mol. Biol.* **215**, 403–410 (1990).

7. Gish, W. WU-BLAST 2.0 (1996-2005).

8. Pearson, W. R. & Lipman, D. J. Improved tools for biological sequence comparison. *Proc. Natl. Acad. Sci. USA* **85**, 2444–2448 (1988).

9. Saebø, P. E., Andersen, S. M., Myrseth, J., Laerdahl, J. K. & Rognes, T. PARALIGN: rapid and sensitive sequence similarity searches powered by parallel computing technology. *Nucleic Acids Res.* **33**, 535–539 (2005).

10. Pearson, W. R. Searching protein sequence libraries: comparison of the sensitivity and selectivity of the Smith-Waterman and FASTA algorithms. *Genomics* **11**, 635–650 (1991).

11. Smith, T. & Waterman, M. Identification of common molecular subsequences. *J. Mol. Biol.* **147**, 195–197 (1981).

12. Hughey, R. & Krogh, A. Hidden Markov models for sequence analysis: extension and analysis of the basic method. *Comput. Appl. Biosci.* **12**, 95–107 (1996).

13. Karplus, K., Barrett, C. & Hughey, R. Hidden Markov models for detecting remote protein homologies. *Bioinformatics* **14**, 846–856 (1998).

14. Eddy, S. R. Profile hidden Markov models. *Bioinformatics* **14**, 755–763 (1998).

15. Gautheret, D. & Lambert, A. Direct RNA definition and identification from multiple sequence alignments using secondary structure profiles. *J. Mol. Biol.* **313**, 1003–1011 (2001).

16. Eddy, S. R. A memory efficient dynamic programming algorithm for optimal structural alignment of a sequence to an RNA secondary structure. *BMC Bioinformatics* **3**, 18 (2002).

17. Weinberg, Z. & Ruzzo, W. L. Sequence-based heuristics for faster annotation of non-coding RNA families. *Bioinformatics* **22**, 445–452 (2006).

18. Klein, R. J. & Eddy, S. R. RSEARCH: finding homologs of single structured RNA sequences. *BMC Bioinformatics* **4**, 44 (2003).

19. Liu, J., Wang, J. T., Hu, J. & Tian, B. A method for aligning RNA secondary structures and its application to RNA motif detection. *BMC Bioinformatics* **6**, 89 (2005).