α

The University of Queensland

School of Molecular and Microbial Sciences
Department of Biochemistry and Molecular Biology

BIOC6103

Research Project in Biochemistry

Topic:

“Prediction of H/ACA-box small nucleolar RNAs in *Drosophila melanogaster*

Prof John Mattick

Principle Supervisor

Dr Larry Croft

Associate Supervisor

Sylvia Tippmann

Honours Student

Acknowledgements

I want to acknowledge….

Larry for keeping me as relaxed as possible during this year, Evgenij for explaining Alzheimer's to me, Mike for 'This is NOT a table!', Ryan for controlling a bioinformatician while running my first PCR, cutting my first gel and spreading bacteria and John Mattick for constant support during the whole work.

And of course….

… all the other members of the Mattick group for a lot of interesting and distracting discussions.

Abstract

Apart from mRNAs and infrastructural RNAs such as tRNAs and rRNAs there are other small non-coding RNAs (ncRNAs) that are able to regulate gene expression in a tissue-specific and developmental manner. There are two main classes of these small regulatory ncRNAs that have been identified: microRNAs (miRNAs) and small nucleolar RNAs (snoRNAs), of which there are two types - C/D-box and H/ACA-box snoRNAs. H/ACA-box snoRNAs guide pseudouridinylation of their target RNA, which can be either rRNA, tRNA, snRNA or even mRNA leading to conformational changes. However, unlike the related methylation guide, C/D-box snoRNAs, only a few H/ACA-box snoRNAs have been identified to date. In this study bioinformatic methods were developed to predict H/ACA-box like RNAs in the genome of *D. melanogaster* by analysing the structures of a set of known H/ACA-box snoRNAs and using machine learning techniques to identify significant sequence and structural characteristics. Scanning secondary structures across the whole genome for these characteristics predicted between 10,000-44,000 putative H/ACA-box snoRNAs including false positives, depending on secondary structure prediction parameters. Additionally, information about evolutionary conservation of RNA secondary structure amongst up to 6 species was used to obtain a higher confidence set of 183-624 predicted H/ACA-box snoRNAs with a sensitivity between 3-8%. The results suggest that a combination of diverse criteria is required to predict H/ACA-box snoRNAs with reasonable sensitivity and selectivity. Microarray and PCR-based experiments are underway to validate an expression of these putative small RNAs and to obtain a more accurate estimate of the actual number of H/ACA-box snoRNAs in the genome of *D. melanogaster*.

Prediction of H/ACA box snoRNAs in *Drosophila melanogaster*

Contents

1 Introduction

Traditionally, RNA has been considered only to play an infrastructural role as transfer-RNA (tRNA) and ribosomal-RNA (rRNA) or as an intermediate in gene expression in the form of messenger-RNA (mRNA). However, over the last few decades it became apparent that structured RNAs such as miRNAs and longer non-coding RNAs (ncRNAs) also regulate the process of gene expression (1,2). Furthermore, as more genomes are sequenced and more ncRNAs are being discovered it appears that the ratio of non-coding transcripts correlates with evolutionary complexity (3). The discovery in different organisms that the majority their genome is transcribed (4,5), while protein coding sequence comprises only a small fraction of it, leads to the assumption that there are much more ncRNAs than previously expected.

Two main classes of small ncRNAs are known to date: microRNAs (miRNAs) (6,7), which target mRNA, and small nucleolar RNAs (snoRNAs) (8). While miRNAs have been studied for almost a decade and are still not completely characterized, snoRNAs, which are recently revealing diverse functional importance. SnoRNAs were initially thought to be solely concerned with the modification of rRNA (9) and snRNAs, but since then they have also been found to modify other RNAs as well. Many snoRNAs have been identified experimentally by cDNA screens and northern blotting (10-12). Apart from their function in telomer maintenance and splicing control (8) snoRNAs function as modification guide to rRNAs and snRNAs. Notably, along with experimental identification of guide snoRNAs in different eukaryotes, an increasing number of orphan snoRNAs, lacking known targets, have been discovered. One of these orphan snoRNAs has been assigned to target the mRNA of a serotonin receptor (13) and many of the remaining orphan snoRNAs are expressed specifically in brain. This suggests that there might be more snoRNAs, which fulfil diverse functions and are await discovery. At present there are two modifications known to be guided by snoRNAs: 2'-*O*-methylation of the target sequence and pseudouridinylation (Ψ). The latter is directed by H/ACA-box snoRNAs named according to its short conserved consensus sequence motifs. Pseudouridinylation guides have been studied in exhaustively in yeast (14-16) and later also in vertebrates (17,18). They range between 120 and 170

nucleotides in length and have a characteristic secondary structure (19,20) (see **Figure 1**).

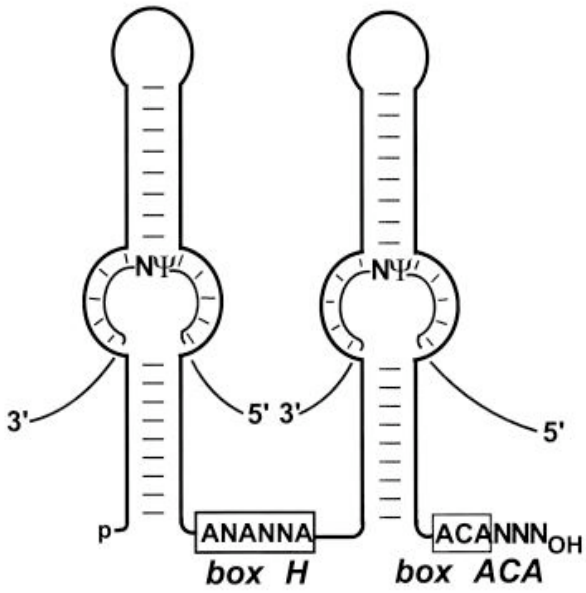

Figure 1. Idealized Secondary Structure of a H/ACA box snoRNA. H/ACA box snoRNAs are characterized by their hairpin-hinge-hairpin-tail structure. The variable 'H' motif (ANNANA, with N={A,C,G,U}) is located in between the two hairpins and the 'ACA' motif can be found 3 nucleotides downstream of the 3' end of the structure. The 2 fold-back-hairpins usually have bulges, at least one of which contains the antisense sequence complementary to the pseudo-uridinylation site of the target RNA.

In contrast to coding transcripts, non coding transcripts such as H/ACA-box snoRNAs are more difficult to detect with biochemical methods because they are relatively short (< 170 nt). They are often not polyadenylated and might be expressed only in certain tissues and under specific conditions. Although experimental methods discovered some non-coding RNAs (10,11), it became evident that no single screen is able to discover all ncRNAs of an organism completely. As shown in previous studies (21-26) it might be more effective to apply bioinformatic approaches first to detect ncRNA candidates and subsequently verify them by biochemical methods such as northern blots or microarrays. The growing number of sequenced genomes (27) and corresponding annotation databases provides sufficient initial data available to apply this procedure.

Earlier prediction approaches in mammals required a Ψ-site in rRNAs or snRNAs (18), restriction of the search to orthologs in introns (28) or to regions of high sequence conservation (18,29). Since restriction to a certain genomic locations or targets should be avoided in this study *D. melanogaster* (*D. mel*) is used as a model or-

ganism (30) because it has a manageable genome size and is therefore ideal for initial computational approaches regarding the whole genome. Drosophila species are one of the most comprehensively sequenced group of organisms. In total nine insect species, including *D. mel* are annotated (31) and aligned to each other (32,33). The most complete set of known H/ACA box snoRNAs in *D. mel* comprises 113 partly experimentally validated H/ACA-box snoRNAs (34), all but nine are located in introns and most guide rRNA and snRNA Ψ. However, assuming there are many orphan snoRNAs remaining undiscovered, it is expected that the total number is much larger – possibly of the order of 10^3–10^4. (Box 1) In this study it will be investigated if this estimation is realistic and therefore predict novel H/ACA-box snoRNAs in *D.mel*.

In a bioinformatic context, a prediction is done through observational studies of known facts. To test the predictive ability of our algorithm, we measure its sensitivity and specificity in regard to the set of known H/ACA-box snoRNAs.

For each prediction experiment, 4 characteristic values were extracted: 'true positives' (TP) are known snoRNAs that are detected, where we define 'detected' with an overlap of the prediction and a known snoRNA allowing a 2nt deviation. 'False negatives' (FN) are known snoRNAs missed by the method, 'false positives' (FP) are predictions that are not known H/ACA box snoRNAs and 'true negatives' (TN) are sequences that are not known to be snoRNAs and also not predicted as such. These four values can be tabulated in a confusion matrix shown in

Table 1. It must be pointed out that the definition of TP and FP is extremely strict as there are undoubtedly undiscovered H/ACA-box snoRNAs in the genome. However, the performance measurements should not be prejudiced by these estimations and assessed with the 113 known H/ACA-box snoRNAs as the only 'true' ones.

Table 1. Confusion matrix schema showing performance of algorithms described in this work.

		Classified as	
Classes		**Y**	**N**
RNA structures which are H/ACA snoRNAs =	**Y**	TP	FN
RNA structures which are not H/ACA snoRNAs =	**N**	FP	TN

To measure the predictive ability of an approach sensitivity, specificity, selectivity (aka positive predictive value) are used. Since there are some disagreements regarding their definition I define them as follows:

$$sensitivity\ (X) = \frac{TP}{TP\ +\ FN} \qquad \text{[eq. 1]}$$

$$specificity\ (Y) = \frac{TN}{TN\ +\ FP} \qquad \text{[eq. 2]}$$

$$selectivity\ (Z) = \frac{TP}{TP\ +\ FP} \qquad \text{[eq. 3]}$$

Box 1. Order of magnitude calculation (35). for the number of expected H/ACA-box snoRNAs in *D.mel*

To define the aim in developing an algorithm with certain specificity and sensitivity more precisely, an order-of-magnitude estimation was made:

The set of known H/ACAs comprises 113 H/ACAbox snoRNAs, which is of the order of 10^2 ($\sim 10^2$). One H/ACA snoRNA is about 200nt in length, consequently, if there were $\sim 10^5$ H/ACA snoRNAs in the genome of *D. mel*, this would add up to $\sim 10^7$ nucleotides of H/ACA snoRNAs, which is 10% of the whole genome. This number is highly unlikely, so the expected number of H/ACA snoRNAs in the genome ranges from $\sim 10^2$ to $\sim 10^4$.

Since the number of known H/ACA snoRNAs is very small in comparison to what we consider to be in the genome, its features might not even be representative for the functional group H/ACA snoRNA. Consequently we will not care too much about a high sensitivity and instead focus on narrowing down the vast number of candidate structures. Assuming a possible structure starting at each position of the genome on both strands, there are $\sim 10^8$ possible candidate structures in *D. mel*. To arrive at the upper bound of reasonable predictions the specificity must be at least 99.99%, which is almost 1.

From the known H/ACAs, we must identify at least one to be sure the filter is working at all. Consequently, the minimum bound on the sensitivity is 1%, which is $\sim 10^{-2}$. Ideally the sensitivity is greater but given this small set of known H/ACA box snoRNAs we have to deal with this uncertainty! The sensitivity can therefore be at least 100 times smaller than the specificity and has to be weighted 100 times less than the specificity. We take this ratio for weighting into account later on when we compare the different approaches.

1ST approach: Prediction of H/ACA-box snoRNAs using five criteria

1.1 Introduction

At least some con-coding RNAs, including miRNAs and snoRNAs, should undergo a selective pressure in the same way as coding transcripts if they fulfil essential functions. Indeed, there are quite recognizable examples of ncRNAs, which can be silhouetted against their flanking regions just by primary sequence conservation (36) (**Figure 2**). It seems therefore a promising idea just to work with a conserved subset of the genome giving more confidence in the function of predictions. Conservation of nucleotide sequence will be used as the first criteria defining H/ACA box snoRNAs. The other 4 criteria will be secondary structure derived from literature (37) (see **Figure 1**). Hence the folding structure of each candidate sequence needs to be obtained using secondary structure prediction tools.

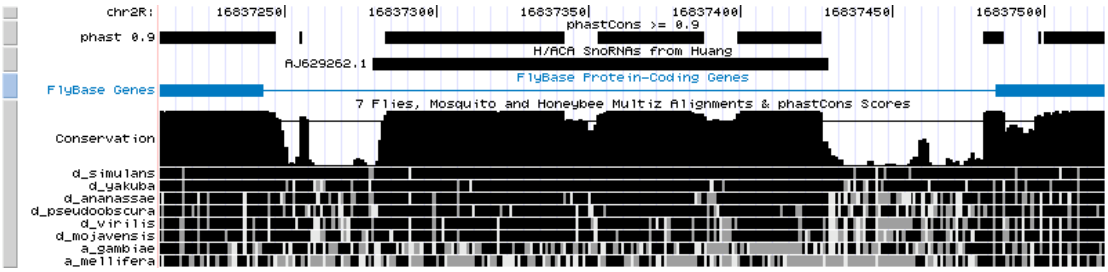

Figure 2. Example for a highly conserved ncRNA. H/ACA box snoRNA in *D. mel* chromosome 2R. The first row shows position within the chromosome. The second row shows blocks that are conserved over a threshold of 0.9. The location of the known H/ACA box snoRNA is given in row 3 together with its ID 'AJ629262.1'. The blue line indicates annotated flyBaseGenes (31), the thin line indicates intron, thick line protein coding sequence. Below the nucleotide conservation with respect to *D. mel* and 8 more insect species is shown in a conservation profile.

1.2 Method

First a highly conserved subset of the *D. mel* genome with was extracted based on the whole genome alignment of 9 insect species, (multiz9way, UCSC). The conservation threshold was derived by using the program phastCons with a conservation score over 0.9 [Siepel, 2005 #283]. PhastCons is based on a two-state phylogenetic hidden

Markov model and predicting conserved elements based on this model. This conserved subset comprises ~26% of the *D. mel* genome (1,591,344 fragments). For the purpose of this study however, just the fragments in this subset, which are at least 120nt in length to fit the minimal size of an H/ACA-box snoRNA are regarded. Surprisingly, this leaves less than 1% (~15,500 fragments) of the total genome. Reducing the input more then 100 times decreases the time and space requirement for further computational processing for this initial analysis substantially.

To apply a filter based on secondary structure criteria it is necessary to obtain the folding of each candidate RNA using a prediction program such as RNAfold (38). Since it is not reasonable to fold sequences that are much longer than the longest H/ACA-box snoRNA, these long blocks were pre-processed by tiling. A window of 180nt in length, sliding nucleotide by nucleotide, yielded a set of structures comprising 49,944 candidate sequences in total. The minimum free energy (MFE) secondary structure of these candidates was predicted using RNAfold with the parameters '–noLP' (allows no lonely pairs) and '–d2' (dangling energies are added for the bases adjacent to a helix on both sides in any case) (Vienna Package version_1.5). The file containing ID, sequence and secondary structure in bracket notation for each candidate serves as input to the following H/ACA-box snoRNA prediction algorithm.

I developed a filter algorithm for H/ACAbox snoRNA features derived from published data (37). It scans for the characteristic H/ACA box snoRNA features (see **Figure 1**). More the selection criteria used in the filter algorithm can be written as follows:

- Appearance of exactly 2 hairpins
- Appearance of both H-motif and ACA-motif
- Hairpins sizes should be symmetric
- Each hairpin stem should have >15 paired bases

Executable source code of the filter program 'SnoSearch' containing the exact specification of each criterion and additional test input files can be found in Appendix IV.I.

1.3 Results

The filter program was applied to the input set of folded candidate sequences from conserved blocks. Since only the conserved subset of the *D. mel* genome was scanned the computation time was ~30min on a linux cluster with 8 processors. More than 99% of the 49,944 candidates were filtered out by one of the listed criteria. Only 76 were classified as H/ACA box snoRNA. Amongst these 76 positives none of the known H/ACA box snoRNAs appeared.

1.4 Summary

The filter criteria appear to be very stringent on the candidate set since it filters out ~99.84% of total input. Surprisingly, however none of the Huang H/ACA-box snoRNAs appears in the set of positively classified H/ACA box snoRNAs. It appears the assumptions (a) snoRNAs are highly conserved elements and (b) the perceived characteristic features are not good predictors.

This could be due to several reasons: (a) the criteria in the filter algorithm do not model features of known H/ACA box snoRNAs very well and/or (b) the known H/ACA box snoRNAs do not completely appear in the set of conserved regions; or (c) the secondary structure prediction does not identify H/ACAbox snoRNA structures. In the following chapter these problems will be investigated to improve the prediction approach later.

2 Possible difficulties

2.1 Algorithm performance

The most obvious reason why none of the known H/ACA-box snoRNAs was identified is that the sequence and structural characteristics are not sensitive enough. To investigate this idea, the filter algorithm SnoSearch was applied to the known H/ACA-box snoRNAs from Huang previously folded with RNAfold using the same parameters as in 2.2. (see **Table 2**). To determine the impact of each filter criteria, the numbers of remaining known H/ACA-box snoRNAs is listed for each step.

Table 2. Number of snoRNAs from Huang that passed each filter step.

Filter	Remaining candidates after filtering	
Input Set	113	100%
1) Appearance of two hairpins	109	96%
2) Appearance of H and ACA motif	73	65%
3) Symmetry of hairpins	61	54%
4) Length of single hairpins	53	47%

Since H/ACA-box snoRNA features were extracted out of the literature, it assumed they hold true for the majority of the known H/ACA-box snoRNAs. A sensitivity of only 47% suggests that more detailed analysis of the known set will be required for choosing filter parameters. However, since at least a subset of the known H/ACA-box snoRNAs is picked up by the algorithm, the poor sensitivity is likely to originate also in low sequence conservation of the known H/ACA-box snoRNAs.

2.2 Sequence conservation of known H/ACA box snoRNAs

Another possible problem for the TP-rate could be that the sequence conservation of the known H/ACA-box snoRNAs is not as high as expected. The average length of fragments in the set of conserved sequences is 22nt (see

Table **1**), suggesting that most of the known H/ACA-box snoRNAs might not be entirely covered by one fragment in this set. To investigate this assumption the known H/ACA-box snoRNAs were analysed with respect to their conservation among all 9 aligned insect species. To obtain the conserved sequences and intersect their location with the set of known H/ACA-box snoRNAs, the UCSC tool 'featureBits' was used. **Table 3** displays the number of nucleotides and run-on sequences without gaps (blocks) in the respective file (column 1), 'Huang H/ACA snoRNAs' refers to the collection of the 113 known H/ACA-box snoRNAs. 'phastCons pieces' are all sequences of the *D. mel* genome where the conservation is over the threshold of 0.9. This comprises the input file for prediction. Due to sharp conservation peaks in the sequence the number of nucleotides is just ~20 times larger then the number of blocks (34,656,886 / 1,591,344 = 21.8), implying that one block is on average 20nt long. Therefore it is not surprising no single known H/ACA-box snoRNA is completely overlapping with a run-on conserved block. Therefore none of the 47% of known H/ACA-box snoRNAs, which would have been returned by the filter algorithm appeared as a result because they weren't in the input set of conserved long fragments. However, about 50% of the nucleotides in known H/ACA-box snoRNAs appear to be conserved. Examining some conservation plots of known H/ACA-box snoRNAs different 'patterns' of conservation were observed spanning from zero to five conservation peaks, they never occurred in a big completely conserved block (for examples of conservation profiles see Appendix I). Another file was therefore created, where closely located conservation blocks (closer than 120nt) are joined together ('phastCons pieces, joined) and indeed, 73 out of 113 Huang H/ACA snoRNAs are now covered by these conserved blocks suggesting that this set might have been an input set to achieve higher sensitivity in the prediction.

However, since the file containing 'joined phasCons pieces' covers already half of the *D. mel* genome the whole genome can be used as input as well. However, primarily the applicability of RNAfold in terms of computation time to this large input set was examined before re-folding (see **Table 4** next section).

Table 3. Primary sequence conservation analysis of the set of known H/ACA-box snoRNAs.

File	Blocks	Nucleotides
H/ACA snoRNAs	113	16,151
PhasCons pieces > 0.9	1,591,344	34,656,886
H/ACA pieces in conserved blocks	388	7,747
Complete H/ACA in conserved blocks	0	-
PhasCons pieces > 0.9, joined	154,357	69,746,740
H/ACA pieces in conserved blocks	106	14,313
Complete H/ACA in conserved blocks	73	-

2.3 Secondary structure prediction

Predicting the secondary structure of a large sequence is challenging since it is not forming one big structure, but various little sub-structures. Taking into account that both DNA strands can be transcribed into RNA, thus doubling the search space, the genome sequence of *D. mel* gives a total of 260 million bases (twice the genome size) to predict secondary structure from.

2.3.1 Runtime for RNA secondary structure prediction

The RNAfold algorithm is a global secondary structure prediction based on minimum free energy (MFE) (39); it uses a 2D-matrix describing all possible nucleotide pairings of the input sequence of length n and therefore has a time complexity of $O(n^2)$. Since possible structures are searched within this long sequence rather than folding the genome into one big structure, local alignments are favourable rather than a global one. In order to predict possible 'foldings' of genome sub sequences, a brute force approach would be to split the whole genome into sub sequences of a desired size and predict the secondary structure of each using RNAfold (38). However, not to miss out on any possible structure, the sub sequences need to overlap, the more they do the

more likely all the foldings are picked up. In an ideal case this means, we predict secondary structure using a window sliding over the whole genome nucleotide by nucleotide. Consequently 260 million sub sequences would be faced. Evidently, this is a process of enormous time complexity: $O(n^2)*x$, with x number of sub-sequences. To address this problem the Vienna Package provides a program for whole genome secondary structure prediction, RNALfold (40). It does not search for possible base pairings of far distant nucleotides throughout the genome but restricts its search space by a maximal folding size L. It implements the RNAfold algorithm and runs in the same time complexity. However, this implementation avoids calling the RNAfold program 260 million times, resulting in substantial improvement in actual runtime, In fact, benchmark tests on a small random sequence of 1 Mb revealed that RNAfold would run in an infeasible time (see Table 4). Given this benchmark test RNALfold was chosen for whole genome secondary structure prediction.

Table 4. Time requirement of RNAfold contrasting RNALfold.

	RNAfold	RNALfold
Time for sequence length $= 1 * 10^6$	630m	8m47s
Time for sequence length $= 265 * 10^6$	167072m ~ 2329h*	38m ~ 0.6h

*indicates extrapolated value

2.3.2 Problem of minimum free energy: RNALfold

RNALfold predicts secondary structure of a given RNA sequence based on thermodynamic stability using the minimum free energy algorithm (39). The program was to the whole *D. mel* genome and returned all energetically favourable structures up to a given maximal length. These possible structures, predicted on both strands, were

overlapping and covered the genome almost completely (see **Figure 3**). In fact, the number of nucleotides involved in theses structures together covers the genome size more than four times! This obviously introduces a lot of false-positives, resulting in a poor specificity and selectivity. However, to get a better idea of the actual false-

$$\frac{\text{1,066 Mb in Lfold structures}}{\text{265 Mb in genome sequence}} \approx 4$$

positive rate a random sequence set was created, representing a negative set, with length of the *D.mel* chromosome 3R (largest chromosome) and secondary structures was predicted in both, the sequence of chromosome 3R and in the random sequence. Surprisingly, a similar number of structures (chr3R: 824,196 / random sequence: 945,454) was obtained, suggesting that MFE secondary structure prediction is not a good method to distinguish functional (coding or non-coding) RNA from random sequence. This observation is consistent with a statistical analysis by Rivas et al., stating that secondary structure alone is generally not significant enough for the detection of ncRNAs (41).

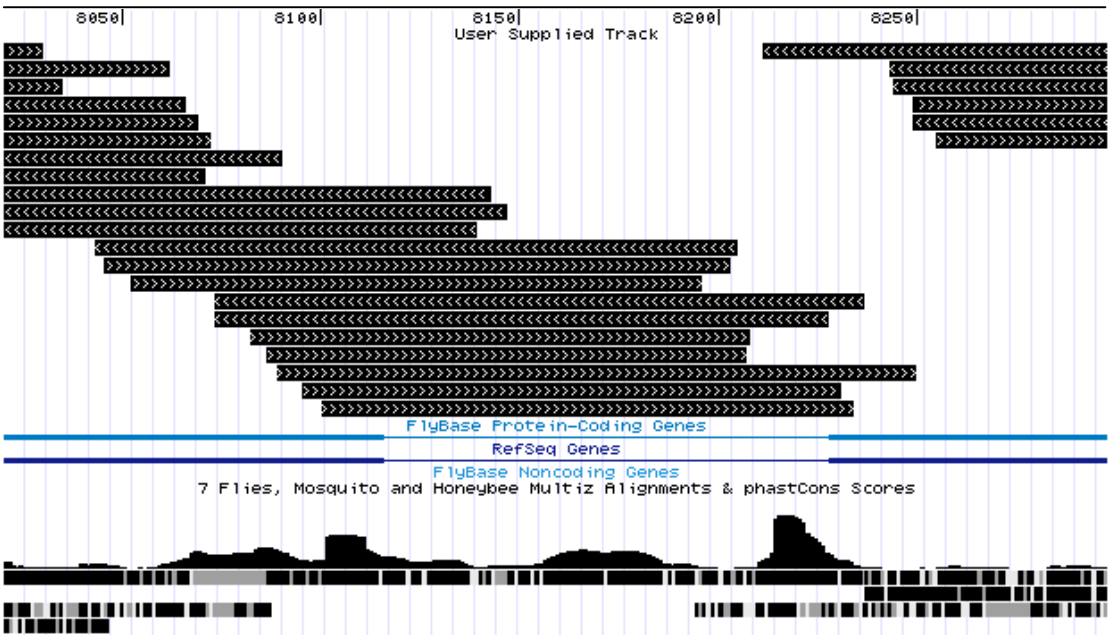

Figure 3. Illustration of predicted secondary structures in genome sequence. Cut-out window of UCSC genome browser showing predicted secondary structures with RNALfold in an intron of a coding gene. Secondary structures, shown as black bars, cover the region multiple times. Arrows indicate reading direction.

However, at this time no other tool, applicable on a whole genome scale, was available to predict secondary structures using another method. Optimisation of the parameters for RNALfold was required to fit the purpose of this study. RNALfold allows the user to choose between several options, which have different effects on the predicted structure of the RNA. Since there is no actual experimental evidence for how the structure forms *in vivo*, the authors left it to the user to adjust the parameters. One of the options is how to treat 'dangling end' energies for bases adjacent to helices in free ends and multi-loops: With (-d1) only unpaired bases can participate in at most one dangling end. With -d2 this check is ignored, dangling energies will be added for the bases adjacent to a helix on both sides in any case. -d0 ignores dangling ends altogether. 113 sequences of known H/ACA-box snoRNAs were folded using –d0, -d1 and –d2 option respectively. The main features of the three different secondary structures of each known H/ACA-box snoRNA were derived to get an idea of the 'average' shape for each structure set (-d0/-d1/-d2) shown in Table 5. The frequency of each feature is shown according to the dangling option used for folding. An example is given in **Figure 4**. Using each option to fold the sequences of the known H/ACA-box snoRNAs, it was found that this seemingly minor change has a large impact on the prediction of the most preferable structure and therefore changes the composition of structures in the folded set substantially. The possibility of optimising the secondary structure prediction step will be investigated by changing the default value (–d2) when folding the whole *D. mel* geno

Table 5. Main features of predicted secondary structures. *multiloops are loops with more than 2 protruding stems.

Foling optiont	Frequency of structures with number of hairpins			Frequency of structures with multiloop
	1 hairpin	2 hairpins	>2 hairpins	
D0	1	98	13	7
D1	4	101	8	30
D2	8	91	14	37

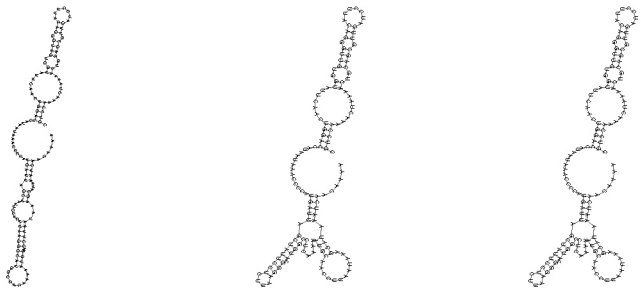

Figure 4. Illustration of secondary structure prediction. Folded is the same sequence with each of the three different dangling end options (-d0, -d1, -d2 f.l.t.r). The example sequence is a known H/ACA box snoRNA.

This study will be continued using all three sets of secondary structures in parallel, contrasting the prediction performance of them respectively throughout the whole process to obtain the optimal result by either choosing one of these sets or combining all of them together. Primarily, RNALfold was applied the whole *D. mel* genome using all three options. To calculate the sensitivity respectively, the number of known H/ACA-box snoRNAs that appear in each secondary structure prediction set was determined. To be classified as TP the secondary structure of a known H/ACA-box snoRNA must fulfil the requirement to appear in the prediction set with a maximal deviation of two nucleotides. Surprisingly, although the set of predicted structures covers the genome multiple times, not all of the known H/ACA-box snoRNA structures have been predicted. Only 30 out of 113 known H/ACA-box snoRNAs were identified with the option –d0 (see **Table 6**). Thus the trade-off between sensitivity and selectivity in the case of RNALfold is particularly unfavourable. To analyse why the sensitivity is so low, folding energies of the known H/ACA-box snoRNAs were analysed revealing a relatively high energy (data not shown), which correlates with the finding that H/ACA-box snoRNAs *in vivo* never occur alone but incorporated in a complex of four proteins. These proteins are able to keep the H/ACA box 'in shape' explaining why no strong folding energy needs to be achieved. Considering these circumstances we confirm again that MFE is not the optimal approach to predict the secondary structure of H/ACA box snoRNAs as they might occur *in vivo*.

Table 6. Secondary structure prediction with RNALfold.

	Structures		Performance measure	
Folding Option	Whole genome Count	H/ACA-box snoRNA Count	Sensitivity %	Selectivity %
D0	7,413,636	30	26.55	$4*10^{-4}$
D1	8,727,229	47	41.59	$5*10^{-4}$
D2	10,237,817	51	45.13	$5*10^{-4}$

To overcome the problem with MFE based folding, the partition function algorithm for secondary structure prediction (42) could be utilized, which has also been implemented as part of RNAfold in the Vienna Package. However, Edvardsson et al. showed that classification of H/ACA snoRNA in yeast based using these probability curves are not very successful (15). One might expect different results in vertebrates, but time complexity is a major problem for the partition function algorithm as well. Meanwhile there is an implementation of the local RNALfold, which uses the partition function making it possible to predict secondary structures on a whole genome scale without MFE. The program, RNAplfold, is also part of the Vienna Package (43). However, at the stage of whole genome folding, this program was not yet available, so RNALfolded structures were used as initial input, tolerating its poor sensitivity/selectivity trade-off.

2.4 Summary

The analysis showed that all three considerations are part of the problem of prediction H/ACA-box snoRNAs: The used filter criteria turned out to be insensitive and the assumption that H/ACA-box snoRNAs are highly conserved holds true only for part of the known H/ACA-box snoRNAs. Furthermore, RNA secondary prediction revealed to be a major problem, but it is unfeasible to address the problem of RNA folding in the given time frame. However, missing out possible H/ACA-box snoRNAs can be avoided by scanning the whole genome rather than just conserved blocks. Beyond that properties of known H/ACA-box snoRNAs can be modelled more precisely by chan-

ging the filter criteria. To do this the next section is devoted to the improvement of these filter criteria.

3 Improving filter criteria

An improvement of the filter criteria can be achieved by analysing the given data manually, which is feasible regarding the relatively small dataset of known H/ACA-box snoRNAs, or by applying automated, computational learning methods to find patterns not recognizable by eye.

3.1 Automated Method – Machine Learning

3.1.1 Introduction

Supervised machine learning algorithms try to learn decision rules from labelled input data, in this case, known H/ACA-box snoRNAs and non-H/ACA-box snoRNAs. These rules are used to classify novel data. Machine learning algorithms have been successfully applied to different biological problems as the analysis of genome-wide expression data (44), to predict genetic regulatory response (45) and also to distinguish protein-coding from non-coding RNAs (46). It might therefore a useful tool to find unrecognised criteria contained within the known H/ACA-box snoRNAs.

3.1.2 Method

Sequences and secondary structure predictions of the known H/ACA-box snoRNAs were used to extract sequence and structure information, which could possibly be interesting, of each candidate. In total there are 45 attributes (features) obtained per instance, where an instance is an H/ACA-box snoRNA (for list of attributes see Appendix II.I).

This data serves as input to the machine-learning algorithm. Applied was the Weka program package (47), which has been used for bioinformatic tasks such as automated protein annotation (48,49) ,probe selection for gene expression arrays (50) and many more. Many of the algorithms implemented in Weka are described in (51). However, no algorithm can be favoured in general, it needs to match the data mining problem specifically to yield useful information and realistic models. Therefore Weka provides several types of model algorithms: decision trees, rules functions and Bayesian classifiers. To determine which algorithm is most suitable for classifying H/ACA-box

snoRNAs they have been analysed on a training set. Training with 4 fold cross validation was performed and the percentages of correctly and incorrectly classified instances were obtained. The training set consisted of known H/ACA-box snoRNAs (positive) and randomly selected genomic sequence as negative controls. Negative controls were selected from different genomic locations in one chromosome: introns, untranslated regions (UTRs) and coding sequence (CDS). H/ACA-box snoRNAs are not expected to occur in UTRs nor in CDS these controls are supposed to contrast the sequence properties (nucleotide distribution, GC content etc.) to real H/ACA-box snoRNAs. The intronic negative control might have similar sequence properties but a different secondary structure, providing the learning algorithm with non-H/ACA shapes. Furthermore different amounts of shuffled sequences of the positive H/ACA-box snoRNAs were added to provide another contrast with exactly the same nucleotides but a totally different structure. The number of shuffled sequence to original snoRNA was varied, so the algorithm builds a model involving more secondary structure attributes rather than sequence properties and patterns.

3.1.3 Results

Detailed results of this algorithm comparison are shown in Appendix II.II. I choose to use the J48 algorithm (52) for training since it had the best percentage of correctly classified instances after 4 times cross validation on the training set. Furthermore, in contrast to some similar performing algorithms, J48 returned a decision tree, which revealed the classification rules. Out of the 45 attributes only about 10 were selected by the learning algorithm to build each model. However, the chosen attributes were different depending on the input set, resulting in a variety of decision trees. An example of a possible decision tree is shown in Appendix II.III.

3.1.4 Summary

From the 45 given attributes the J48 model algorithm selected subsets, which seem to be sufficient to build a model for classification (for decision trees build for each training set see Appendix II.III). Since the choice of these attributes appeared to be highly dependent on the training set they need to be accessed statistically for their significance towards the known H/ACA-box snoRNAs. If analysis confirms a continuous appearance of these features amongst the known H/ACA-box snoRNAs they can be included in the list of filter criteria.

3.1.5 One step further: ML as classification tool

I wanted to determine if the machine-learning algorithm performs better predicting H/ACA-box snoRNAs than the first approach on the same dataset. Since the set of conserved regions turned out to be useless, both machine learning and the first filter algorithm, were applied to the complete *D. mel* genome previously folded with RNALfold. Input is a set of 7,864,903 candidate structures, for performance assessment the 113 known Huang H/ACA-box snoRNAs were used. Sensitivity, specificity and selectivity are calculated according to EQ. 1, 2 and 3. All three values vary largely depending on the training set, however it was not obvious why this great variance occurs. Results of the prediction are shown in **Table 7** and contrasted against the first approach in **Table 15**.

Table 7. Results of machine learning experiments 1-7 .

Experiment		ML predictions		Performance measure		
#	Details	Total	H/ACA	Sensitivity	Specificity	Selectivity
				%	%	%
1	standard shuffle 10	41,613	21	18.58	99.47	0.05
2	standard shuffle 2	16,657	5	4.42	99.79	0.03
3	standard shuffle 5	46,984	23	20.35	99.40	0.05
4	standard no shuffled	172,330	31	27.43	97.81	0.02
5	standard shuffle 1	23,931	2	1.77	99.70	0.01
6	no shuffle, +1 UTR	11,482	2	1.77	99.85	0.02
7	no shuffle, +1 CDS	440	0	0	99.99	0

3.1.6 Conclusion

I wanted to investigate if improvement in the prediction can be achieved by machine learning. Therefore, I applied the first filter program (2.2) to the whole genome. Some

machine learning experiments show an improvement in one or the other value, however, an improvement in sensitivity always correlates with a decrease in specificity and selectivity (see Table 15). The ROC-like plot in Figure **8** illustrates the relation between sensitivity and selectivity. Since I didn't see any possibility to improve the machine learning based prediction method, I decided to improve the filter algorithm from the first approach. However, using machine learning revealed some interesting criteria, which will be verified manually and eventually included in the filter algorithm later on.

3.2 Statistical analysis of known H/ACA-box snoRNAs

3.2.1 Introduction

From the decision trees built during modelling in machine learning I obtained attributes, which appear to be a good classifier for H/ACA-box snoRNAs. The attributes

- Size of each hairpin (Bases)
- Size of biggest internal loops in hairpins (longestShortLR)
- GC-content (GCseqPaired, -Unpaired, -Hairpin, -Symm, -Asymm, -Bulge)
- MFE (energy)
- Number of missing dinucleotides (dinuc_num_missing)

appeared in the several decision trees, suggesting these might be meaningful. Before including these features as filter steps statistical analysis needs to be done. Furthermore I am going to assess features that have been used as filter criteria in the first approach already and subject them to closer assessment.

- Sequence length
- Number of hairpins
- H-motif
- ACA-motif

3.2.2 Method

To compare the significance of an attribute for classifying RNAs into H/ACA-box snoRNA and non-H/ACA-box snoRNA, I did the statistical analysis on the set of known H/ACA-box snoRNAs and contrasted it to negative dataset. The negative dataset was obtained by randomly relocating the coordinates of each known H/ACA-box snoRNA within the same chromosome. I obtained the sequence at this new position and predicted MFE secondary structure of this sequence using RNAfold. This yielded 113 random genomic sequences and structures with the same lengths as the known H/ACA-box snoRNAs. Features of each sequence and corresponding secondary structure in the two datasets respectively were extracted using perl scripts.

3.2.3 Results

Features of known H/ACA-box snoRNAs, including total length, number of hairpins, the length of single hairpins, internal loop sizes, the number of missing dinucleotide combinations, minimum free energy and GC content, were contrasted against a dataset of random genomic sequences. Obtained data is visualized in diagrams; tables are not shown. Diagrams for each analysed criterion and detailed description are shown in Appendix III. The sequence length of known H/ACA-box snoRNAs range between 124 and 267 nt, however, apart from two outliers, the majority falls in an interval 124 to 165 nt. The number of hairpins in known H/ACA-box snoRNAs has been reported to be exactly two, but according to my analysis, this holds true for less than 90%, the rest are structures with mainly 3 hairpins. In the random set, three-hairpin structures occur in the same frequency as in the known set. However the distribution of numbers of hairpins can still be clearly contrasted against random structures, where an equal frequency of structures with one and with two hairpins were found. The H-motif is present in only 9% and the ACA-motif in 43% of the random set. Although H/ACA-motifs are not as ubiquitously apparent in known H/ACA-box snoRNAs, their frequency is significantly higher (8x higher for H-motif, 2x higher for ACA-motif). Since the individual hairpin size appeared in all models build during machine learning appeared in all decision trees (attribute ‘bases’ in decision trees Appendix II.III), this feature was analysed for its significance in contrast to the random set as well (Appendix III.III). A clear difference in size distribution was observed: The sizes of the two longest hairpins in H/ACA-box snoRNAs are, clustered in an interval between 55-90 nt for the longest and 50-70 nt for the second longest hairpin.

Random Structures show no cluster at all, the sizes are equally distributed from 1-145 nt for both hairpins. The internal loop sizes do not appear in each decision tree nor do they have a clearly different distribution from random sequence. However, H/ACA-box snoRNA internal loops are a little more pronounced in an interval between 6-24

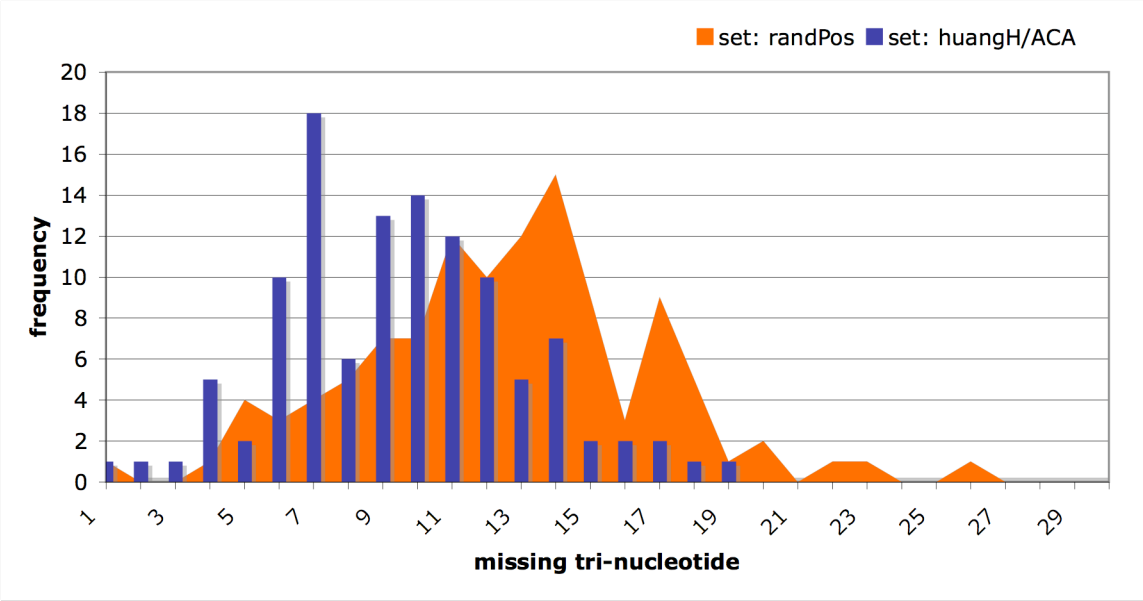

Figure 5. Histogram of number of missing tri-nucleotides. Known H/ACA-box snoRNAs (blue bars) are contrasted against a background of randomly relocated sequence

nt, while random sequence loops are distributed up to 38 nt. Analysis of minimum free energy and GC content shows that the known H/ACA-box snoRNAs don't have significantly better MFE, in fact the MFE appears to be slightly worse. However, as we expected GC content and MFE appear to be connected. The number of missing di-nucleotide compositions appears often in decision trees, however no substantial difference in comparison to the random set could be determined. In order to address possibly higher sequence complexity of H/ACA-box snoRNAs, the tri-nucleotide composition was analysed and the known set indeed lacks less than expected of the 64 possible tri-nucleotides.

By investigating the extrema of each attribute derived from known H/ACA-box snoRNAs two were identified (psi28s-2566 snoRNA, psi28s-291 snoRNA) as outliers in length. The overall length of the outliers is correlated to hairpin length and to MFE, which is itself correlated to GC content. By excluding the outliers a tighter distribu-

tion of MFE, GC content, hairpin sizes and internal loop sizes is observed (see Table 8).

3.2.4 Summary

With the data obtained in this analysis boundaries for the improved filter algorithm, can now be determined, which should lead to an increased sensitivity and specificity. Using tighter cut-offs for features by disregarding the two outliers I expect a much higher specificity over the whole genome.

Furthermore, a decision whether to include the H and ACA motif filter had to be made. Since just 72 of the 113 known H/ACA-box snoRNAs actually have both motifs, this cut-off would reduce the TP-rate quite substantially and decrease the sensitivity of our filter. However, the constraint of both motifs were kept as a filter since we are looking for very strict criteria to narrow down our large set of input structures.

4 2ND approach: prediction of H/ACA-box snoRNAs within the whole genome

The results of the first approach revealed that primary sequence conservation is not a good predictor or filter for finding H/ACA-box snoRNAs. I found that the nucleotide sequence of a large fraction of known H/ACA-box snoRNAs is not conserved throughout insects. Consequently I decided to turn to a whole genome prediction approach. The haploid *D. mel* genome has a size of about 130 Mb, about 13,767 protein coding genes and 808 non-coding genes have been found to date (31).

4.1 SnoStorm: An improved filter algorithm for H/ACA-box snoRNA features

4.1.1 Method

From Machine Learning (3.1) and the following manual analysis (3.2) new parameter cut-offs were derived, to filter the input dataset for H/ACA-box snoRNA-like properties, which are summarized in Table 8 and Table 9.

Table 8. Numerical cut-offs of filter criteria used in the algorithm.

Feature		MIN cut-off	MAX cut-off
Overall size	nt	124	165
Minimum free energy	kJ	-56.42	-18.2
Dinucleotide missing	#	0	2
Trinucleotide missing	#	2	19
GC content	%	24	55
Number of hairpins	#	1	4
1st hairpin size	nt	55	109
2nd hairpin size	nt	0	72

sum of two hairpins	nt	106	148
loop size of 1st hairpin	nt	0	22
loop size of 2nd hairpins	nt	0	21
max loop of 2 haipins	nt	4	22

Table 9. Appearance of H and ACA motif in known H/ACA-box snoRNAs.

	Frequency of motif appearance		Requirement for motif
Motif	absent	present	
ACA-motif	24	87	yes
H-motif	35	76	yes

With these given parameters I am able to classify all but the two outliers given the known H/ACA-box snoRNAs as input dataset. However, the supposably characteristic motifs "ACA" and "H" are only present in a smaller subset of the known H/ACA-box snoRNAs. After performing some tests regarding specificity, I decided to keep this criterion anyway in order to massively reduce the candidate space since specificity is the most important factor for our purpose. The filter program requires input data in fasta-format with a third line containing secondary structure. It outputs the candidates classified as H/ACA-box snoRNA-like in the same format and two additional files containing scanned features of each candidate and details about filtering (see

Figure 6).

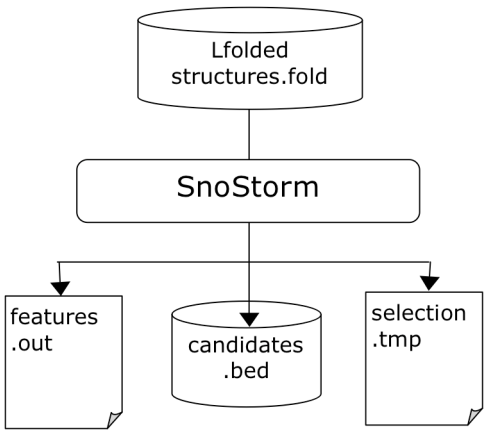

Figure 6. Flowchart of filter program 'SnoStorm'.

SnoStorm was implemented in perl; it iterates through all given input candidates scanning primary sequence and secondary structure for a fixed number of features. Since it operates largely with regular expression on the given strings (nucleotide sequence, secondary structure in Vienna notation) it is independent from the length of each candidate. Moreover, the size of each candidate sequence was restricted in the initial secondary prediction step by RNALfold (-L, maximal window size) already. Therefore the time complexity O(n), with n being the number of candidates.

4.1.2 Results

Executable source code of the filter program 'SnoStorm' and additional test input files can be found in Appendix IV.II. SnoStorm was used on each input set of different secondary structure predictions (-d0, -d1, -d2), the filter substantially reduced the number of possible candidates down to a two orders of magnitude smaller set comprising H/ACA-box snoRNA-like predictions. In all three cases we cut down the input set by more than 99%/. Analysis of due to which criteria the input set was Sensitivity, specificity and selectivity are shown for each applied dangling option.

Table 10. Results of improved filter algorithm 'SnoStorm'.

	Input set		SnoStorm predictions		Performance measure		
Folding option	Total	H/ACA	Total	H/ACA	Sensitivity	Specificity	Selectivity
					%	%	%
D0	7,413,636	30	10,839	4	3.54	99.85	0.04
D1	8,727,229	47	31,646	12	10.62	99.64	0.04
D2	10,237,817	51	44,942	14	12.39	99.56	0.03

4.1.3 Summary

The presented algorithm, SnoStorm, was designed to be highly specific resulting in a substantially reduced candidate set of predicted H/ACA-box snoRNAs.

Being aware of the low sensitivity but according to the approximation from the introduction Box 1 this sensitivity is bearable assuming that the set of known H/ACA-box snoRNAs might not even be representative for the class of largely undiscovered H/ACA-box snoRNAs. Performance of SnoStorm is contrasted to the first approach for prediction H/ACA-box snoRNAs and to the attempt of using machine learning in Table 15.

4.2 Using conservation of secondary structure

4.2.1 Method

The prediction so far resulted in a very low sensitivity, largely due to the poor applicability of MFE folding for H/ACA snoRNAs. In this step I try to overcome this problem by re-folding. Henceforth, the prediction set comprises now just ~1% of the original input, so it was possible to use a more complex algorithm to predict the secondary structure, assuming to obtain a structure which is closer to the 'real' structure. Single sequence methods based on MFE have some intrinsic limits: many bases of structural RNAs are modified by sugar methylation, pseudo-uridinylation etc. Additionally some functional RNAs have bistable structures, so the thermodynamic model is just an estimate of the real folding. Therefore, it is generally believed that comparative methods return more reliable results (53). There are basically three approaches to obtain aligned structures from RNA sequences: (1) Simultaneous folding and alignment based on the Sankoff algorithm is extremely time intense ($O(n^{3m})$, where n sequence length and m number of sequences) and unfeasible for this whole genome approach. Another approach is to align homologous secondary structures (2), however, this requires highly reliable individual secondary structures (derived by crystallography/NMR), which is not available at this stage. The third possibility (3) is to perform multiple sequence alignment of the homologous RNA sequences and use this alignment as input for secondary structure prediction. Since 9 insect species are sequenced and aligned there is a dataset of high homology sequences available, so I choose this approach. Several programs have been published for the task of folding

an alignment, the most recent of which is RNAz (54). Although it decreases time requirements benchmark estimate the effective computation time for the whole *D.mel* genome to be infeasible (see Table 11). 100 alignments of length 160nt respectively were used to execute benchmark test and approximated the time requirement to run RNAz on our set of SnoStorm-predictions (44,000 candidates) and on the total tiled *D. mel* genome (265,192,960 candidates). Regarding these approximations, I restricted the method to re-fold the three SnoStorm-prediction sets

Table 11. Time requirement for RNAz.

Number of alignments	Time in Seconds
100	39
44,000	17,199
265,192,960	103,661,276
	(= 1,200 days)

4.2.2 Results

For all candidates predicted by SnoStorm multiple alignments were obtained from UCSC, and subsequently selected for the six most related species (because RNAz can currently just handle up to six sequences in the alignment). Given these alignments I applied RNAz and obtained the number of consensus structures.

Table 12 Results and performance measure using SnoStorm and RNAz.

Folding option	Input		SnoStorm+RNAz Predictions		Performance measure		
	Total	H/ACA	Total	H/ACA	Sensitivity	Specificity	Selectivity
					%	%	%
D0	7,413,636	30	138	3	2.65	>99.99	1.64
D1	8,727,229	47	428	6	5.31	>99.99	1.40
D2	10,237,817	51	624	9	7.96	99.99	1.44

Table 13. Filtering of consensus structures obtained by RNAz

Folding option	Input	Sequential filters							
		Non linear		Length <165		2 hairpins		hairpin sum	
		#	%	#	%	#	%	#	%
D0	10,839	6,751	62	1,476	14	355	3	183	1
D1	31,646	18,924	60	5,401	17	892	3	428	1
D2	44,942	26,693	60	8,188	18	624	3	624	1

Consensus structures might not have the characteristic H/ACA box snoRNA features anymore; they can even be linear due to sequence alignments with many insertions and deletions. Therefore I implemented some loose filters to discard structures that vary too much from the individual structure in *D. mel* (see Table 13). (Executable source code of the sequential filter and additional test input files can be found in Appendix IV.III.) The first filter discards all cases where no consensus structure was found, i.e. where the consensus structure is linear. The consensus structure can also be quite long due to insertions in the alignments, so secondly, I just want to keep structures that do not exceed the maximal size of known H/ACA-box snoRNAs (165nt). Furthermore, two hairpins are required in a third filter step, which should (forth) have a cumulative size of 106nt–148nt (which is also a cut-off value in 'SnoStorm'). The remaining consensus secondary structures after filtering adapt an H/ACA-box snoRNA-like shape. Only about 2% of the consensus structures obtained by RNAz survive the filter, however, a much larger fraction of the known H/ACA-box snoRNAs make it trough the filter (50-75%). This suggests that the RNAz consensus structure in combination with the applied rough filter criteria is a good predictor for H/ACA-box snoRNAs.

Since the characteristic motifs H and ACA could be shifted in respect to the structure during consensus structure prediction we investigated their presence in our final pre-

diction sets of 183 (-d0), 428 (-d1) and 624 (-d2) possible H/ACA snoRNAs. The results of this analysis are shown in Table 14. However, since the consensus structure as such does not exist *in vivo*, I wont use the relative position of the motif in the consensus structure as additional filter step.

Table 14. Analysis of H/ACA motif in consensus secondary structures.

Folding option	Input	Appearance of sequence motif in consensus structure					
		ACA motif		H motif		H/ACA motifs	
		#	%	#	%	#	%
D0	183	156	85	25	18	22	12
D1	428	393	92	100	23	92	22
D2	624	580	93	163	26	149	24

4.2.3 Summary

Comparing the three different structure sets obtained by different folding options at the end of the prediction process (Table 12), reveals that –d0 has the highest selectivity, however the worst sensitivity. Folding with -d2 gives us the most sensitive prediction and –d1 ranges in the middle of these two. The trade-offs between sensitivity and selectivity, however, is almost none between the three dangling options. The set of predicted H/ACA-box snoRNAs overlap only in a small fraction (see **Figure 7**), suggesting that the structures predicted by RNALfold have been quite different in the first place. No set can be favoured due to any performance measure, so, the decision, which one to put the highest confidence in, is subjective.

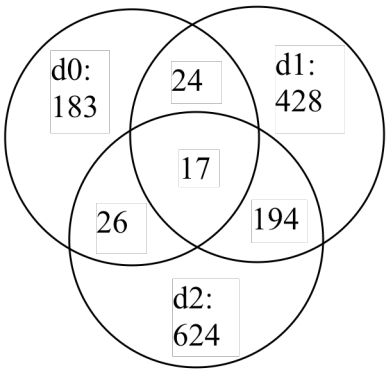

Figure 7. Intersection of H/ACA box snoRNA classified sets. Predicted possible H/ACA snoRNAs using different dangling options for initial folding step. 17 candidates appear to be classified as H/ACA snoRNA by all three options, however, there are still 1058 unique putative H/ACA snoRNAs in total.

Finally, all three dangling options predict structures, which might occur *in vivo*. Any RNA fragment can adapt a great variety of shapes under different environmental conditions concluding that none of the assumed dangling behaviours is in fact 'wrong'. Choosing one or the other option rather changes the rank of a structure in a list of possible foldings each time giving us another energetically most favourable structure. Under this assumption it appears most reasonable to consider putative H/ACA-box snoRNAs obtained by all of the three sets together. Analysing and validating predictions from all sets in parallel, might give additional insight in which assumed dangling behaviour models *in vivo* folding best.

5 Summary: Comparison of all approaches predicting H/ACA-box snoRNAs

Finally, all prediction approaches explored in this work were compared (**Table 15**).

Table 15. Comparison of all prediction approaches.

Method	Classified structures				Perfomance measure		
	TP	FP	TN	FN	Sensitivity	Specificity	Selectivity
ML exp 1	21	41,592	7,823,198	92	18.58	99.47	0.05
ML exp 2	5	16,652	7,848,138	108	4.42	99.79	0.03
ML exp 3	23	46,961	7,817,829	90	20.35	99.40	0.05
ML exp 4	31	172,299	7,692,491	82	27.43	97.81	0.02
ML exp 5	2	23,929	7,840,861	111	1.77	99.70	0.01
ML exp 6	2	11,480	7,853,310	111	1.77	99.85	0.02
ML exp 7	0	440	7,864,350	113	0	99.99	0
SnoStorm d0	4	10,835	7,402,688	109	3.54	99.85	0.04
SnoStorm d1	12	31,634	8,695,482	101	10.62	99.64	0.04
SnoStorm d2	14	44,928	10,192,776	99	12.39	99.56	0.03
RNAz d0	3	180	7,413,343	110	2.65	>99.99	1.64
RNAz d1	6	422	8,726,694	107	5.31	>99.99	1.40
RNAz d2	9	615	10,237,089	104	7.96	99.99	1.44

Applying the first filter algorithm to the whole genome resulted in a sensitivity of 10%, however yielding a low selectivity, i.e. a lot of potential false-negatives in the

prediction set. This rough filter however, was just supposed to provide an initial idea of how many snoRNAs could be expected and which problems could occur.

The machine learning approach to predict H/ACA-box snoRNAs was unsteady in sensitivity, specificity and selectivity depending on the training set. Machine learning experiments 1,3 and 4 were extremely sensitive, however yielded the most potential false-positives as well. On the other hand experiments 5, 6 and 7 reduced false-positives substantially but yielded a very low sensitivity. I designed a filter algorithm SnoStorm to filter the given input set by a carefully analysed selection of criteria. This improved selectivity and specificity steadily but lost some percentage of sensitivity instead. The obtained sets of predictions have a high specificity with numbers of putative H/ACA-box snoRNAs ranging from 10,000 (-d0)– 44,000 (-d2). By predicting the consensus structure of these candidates, and subsequent filtering, only (25%-50%) in sensitivity was lost but two orders of magnitude more selective prediction sets were obtained comprising numbers of putative H/ACA-box snoRNAs ranging from 183 (-d0) – 624 (-d2). The ROC-like plot in Figure **8** illustrates the relation between sensitivity and selectivity. The green spot indicates the first naïve prediction algorithm. It is close to some of the machine learning approaches (blue). However, sensitivity and selectivity of machine learning is distributed without any pattern in the ROC space, illustrating how variable the results where depending on our training set. The predictions obtained from SnoStorm (orange) are very selective without loosing too much in sensitivity and by applying RNAz we can even achieve a substantially better selectivity.

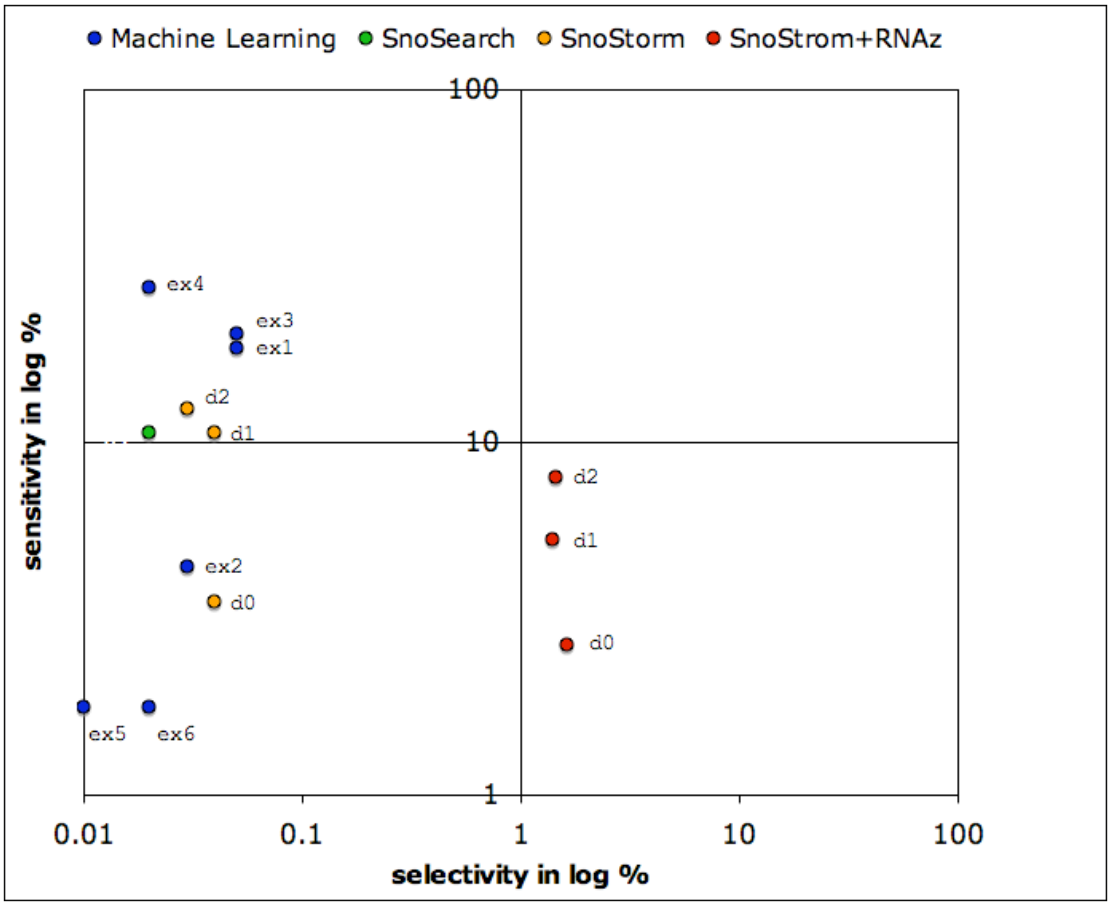

Figure 8. ROC (receiver operating characteristic)-like plot. Graphical representation of the trade-off between the selectivity and sensitivity of all prediction approaches. The more a data point falls into the upper-right corner of the ROC space the more accurate the prediction.

6 Conclusion

There are 113 known H/ACA-box snoRNAs in the genome of *D. mel* and possibly many more yet to be discovered. Other research groups, as discussed in the introduction, applied different approaches to identify novel members of this class experimentally and bioinformatically, with varying success. Here I present a new approach using machine learning and other methods to identify new criteria to predict H/ACA box snoRNAs.

The first surprise was that H/ACA-box snoRNAs are not as conserved as was expected. None of the known H/ACA-box snoRNAs were included in a dataset comprising the most conserved 26% of the *D. mel* genome. Secondly, the four most widely reported features of H/ACA-box snoRNAs, namely, two hairpins, H/ACA-motifs and symmetry and size of hairpins, did not prove to be very sensitive predictors. To address these problems secondary structures were predicted across the whole genome and the sequence and structural characteristics of the known H/ACA-box snoRNAs were reviewed.

The program RNALfold was used to predict structures on a genome wide scale. It ran in 1-2 days on an 8 CPU machine and detected 25%-43% of the known H/ACA-box snoRNAs depending on the folding parameters used. In contrast to the conservation approach however, this input set contained at least a subset of true-positives. However, RNALfold returned a very large number of possible secondary structures, which in fact covered the whole genome more the 4 times, i.e. including each nucleotide in the genome in four different structures on average.

Machine-learning techniques were used to explore a large number of additional sequence and structural characteristics of the known H/ACA-box snoRNAs and identified a small set of significant features including sequence complexity, internal loop sizes and hairpin sizes. These were combined with the prediction criteria from the first approach and subjected to a detailed statistical analysis verifying the significance of each feature in contrast to random sequence. A program, SnoStorm, was developed with these new criteria, finding 10,835-44,928 putative H/ACA-box snoRNAs. SnoStorm has a specificity ranging from 99.56%-99.85%, sensitivity of 3.54%-

12.39%, and selectivity of 0.03%-0.04%, all of which are improved in regard to the first algorithm SnoSearch

Using evolutionary conservation of secondary structure as evidence for functional importance a higher confidence set of H/ACA-box snoRNAs predictions can be obtained. RNAz and multiple alignments of up to 6 *Drosophilae* species were used to identify 183-615 putative H/ACA-box snoRNAs having significant conserved secondary structure. There was a small loss in sensitivity (2.65%-7.96%) due to 1-6 known H/ACA snoRNAs with low secondary structure conservation. However, the specificity was increased to over 99.99%, and the selectivity improved by almost two orders of magnitude up to 1.4%-1.64%.

In conclusion, a combination of criteria has to be used to predict H/ACA-box snoRNAs with reasonable confidence. Since there is only a very small set of known H/ACA-box snoRNAs attention has to be paid to avoid over-fitting the data by selecting too many criteria. A reasonable balance has to be found between narrowing down the large input and being too specific to the known set. A large set containing tens of thousands of putative H/ACA-box snoRNAs and a smaller and higher confidence set comprising only several hundred predictions were identified. These might represent reasonable upper and lower bounds for the total number of H/ACA-box snoRNAs in *D. mel.* genome. Ultimately, experimental validation by micro array expression analysis and subsequent sequencing of positive candidates will reveal which is a more accurate estimate of the number of H/ACA-box snoRNAs that await discovery.

These experiments are currently underway (see below), focussing on the smaller set of higher confidence predictions. Following this, a second set of arrays will be designed to assess the validity of the larger set, using a random subset of the predictions, to obtain an estimate of the true positive frequency in this set.

7 Validation of predicted H/ACA snoRNAs

7.1 Microarray

As a high throughput method of validating the predictions a microarray was designed-for the Combimatrix 12k array platform. Two probes were designed for each prediction using Combimatrix's chip design software. Probes for all known snoRNAs were added for positive controls, as well as mRNA contamination controls and other small RNAs such as 5S. All large RNA transcripts can be removed) (including mRNAs, tRNAs etc) as a source of possible cross hybridisation by hybridising the array with size fractionated RNA (less than 300nt. To estimate the quantity of degraded mRNA passing through the filter we included mRNA contamination controls. Flanking control probes were also added around known snoRNAs within exons, introns and intergenic regions. Antisense probes were added as negative controls as well.

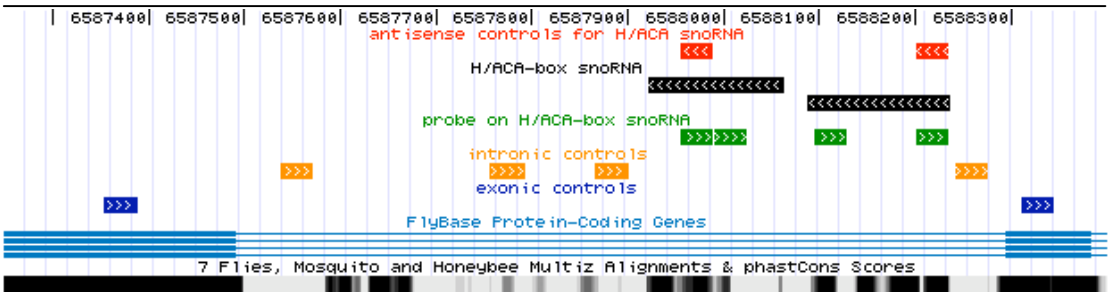

Figure 9. Choice of probes for micro array experiment. Shown is a view of the UCSC genome browser with coloured blocks representing probes we choose for validation. For a randomly chosen subset of previously known snoRNAs (black) we use 2 sense probes (green) and 1 antisense probe (red) respectively. Additionally we positioned intronic control probes around and in between clusters of snoRNAs and exonic probes in the flanking exons as close as possible to the corresponding intron.

7.2 PCR validation

To amplify RNAs as short as our predicted snoRNAs a special priming technique has to be used in order not to occupy the whole snoRNA sequence with primers. The kit was originally designed for detecting miRNAs. However, it appears to work better for snoRNAs since they are a little longer than miRNA precurosor. The protocol was tes-

tested with two known H/ACA-box snoRNAs, one of which has been also predicted as snoRNA. A clear sequence signal was obtained in both cases, indicating the transcript is apparent. This technique will be applied for validation of more H/ACA-box snoRNA predictions.

Figure 10. Validated H/ACA-box snoRNA. This known H/ACA-box snoRNA has also been predicted as such by the improved algorithm. We validated its existence by sequencing the transcript using specific primers at different positions in the snoRNA. Yellow lines indicate correctly obtained sequences. In fact, sequencing suggests that the actual length of the snoRNA transcript is several nucleotides longer than annotated before, the predicted length appears to be correct.

Protocol

Total RNA was isolated from *D. mel* larvae and pupae by Trizol extraction. Two hundred nanograms of total RNA were polyadenylated according to the NCode™ miRNA First-Strand cDNA Synthesis Kit user manual, followed by first-strand synthesis using the universal reverse transcriptase primer provided with the kit. Diverging from the NCode protocol, one microliter of total undiluted first-strand cDNA was used in each 50 microliter PCR reaction. Due to the fact that obtaining a positive signal far outweighed our concerns with high background signals, PCR conditions were used with notably low stringency (see Table 1). However, Platinum Taq polymerase was used to reduce our chances of anomalous results.

Forty microliters of each PCR reaction were transferred to 1.5mL tubes after thermal cycling, and concentrated down to ~10 microliters using a vacuumed-enabled microcentrifuge heated to 45ºC for ~30minutes. Samples were than loaded on a 4% agarose gel composed of 2% DNA Grade Agaraose (Progen Biosciences) and 2% MetaPhor low molecular weight nucleic acid resolving agarose (Cambrex Bio Science). Gels were run at 110V for ~1 hour. Bands of predicted size were excised from the

gel, and gel purified using Promega's Wizard SV Gel & PCR clean-up system. Purified PCR products were cloned into pGEM T-easy following the manufacturer's instructions, transformed into DH5α *E.coli*, and grown overnight on LB plates supplemented with 20 micrograms X-gal and 100 micrograms/microliter ampicillin. Two positive (i.e. white) colonies from each plate were screened using colony PCR and, if positive for the predicted product, were grown overnight. Plasmid DNA was isolated using Promega's Wizard Plus SV Miniprep Kit. Sequencing was performed at the local Australian Genome Research Facility laboratory.

Table 16 PCR conditions.

Thermal Cycling Parameters			PCR Reagents	
Temperature (ºC)	Time (minutes)		Reagent	Final concentration
94	2		$MgCl_2$	1.5 mM
94	30	←	dNTP	0.2 mM
45	30	35x	Plat Taq	1 U
72	30		miRNA primer	0.2 µM
72	10			
10	∞			

Figure 11. Overview of PCR bases snoRNA validation.

8 References

1. Mattick, J. S., and Makunin, I. V. (2005) Hum Mol Genet 14 Spec No 1, R121-132

2. Mattick, J. S. (2004) Sci Am 291(4), 60-67

3. Taft, R. J., Pheasant, M., and Mattick, J. S. (2006) Bioessays

4. Carninci, P. (2006) Trends Genet 22(9), 501-510

5. Manak, J. R., Dike, S., Sementchenko, V., Kapranov, P., Biemar, F., Long, J., Cheng, J., Bell, I., Ghosh, S., Piccolboni, A., and Gingeras, T. R. (2006) Nat Genet 38(10), 1151-1158

6. Bartel, D. P. (2004) Cell 116(2), 281-297

7. Griffiths-Jones, S., Grocock, R. J., van Dongen, S., Bateman, A., and Enright, A. J. (2006) Nucleic Acids Res 34(Database issue), D140-144

8. Bachellerie, J. P., Cavaille, J., and Huttenhofer, A. (2002) Biochimie 84(8), 775-790

9. Kiss-Laszlo, Z., Henry, Y., Bachellerie, J. P., Caizergues-Ferrer, M., and Kiss, T. (1996) Cell 85(7), 1077-1088

10. Okazaki, Y., Furuno, M., Kasukawa, T., Adachi, J., Bono, H., Kondo, S., Nikaido, I., Osato, N., Saito, R., Suzuki, H., Yamanaka, I., Kiyosawa, H., Yagi, K., Tomaru, Y., Hasegawa, Y., Nogami, A., Schonbach, C., Gojobori, T., Baldarelli, R., Hill, D. P., Bult, C., Hume, D. A., Quackenbush, J., Schriml, L. M., Kanapin, A., Matsuda, H., Batalov, S., Beisel, K. W., Blake, J. A., Bradt, D., Brusic, V., Chothia, C., Corbani, L. E., Cousins, S., Dalla, E., Dragani, T. A., Fletcher, C. F., Forrest, A., Frazer, K. S., Gaasterland, T., Gariboldi, M., Gissi, C., Godzik, A., Gough, J., Grimmond, S., Gustincich, S., Hirokawa, N., Jackson, I. J., Jarvis, E. D., Kanai, A., Kawaji, H., Kawasawa, Y., Kedzierski, R. M., King, B. L., Konagaya, A., Kurochkin, I. V., Lee, Y., Lenhard, B., Lyons, P. A., Maglott, D. R., Maltais, L., Marchionni, L., McKenzie, L., Miki, H., Nagashima, T., Numata, K., Okido, T., Pavan, W. J., Pertea, G., Pesole, G., Petrovsky, N., Pillai, R., Pontius, J. U., Qi, D., Ramachandran, S., Ravasi, T., Reed, J. C., Reed, D. J., Reid, J., Ring, B. Z., Ringwald, M., Sandelin, A., Schneider, C., Semple, C. A., Setou, M., Shimada, K., Sultana, R., Takenaka, Y., Taylor, M. S., Teasdale, R. D., Tomita, M., Verardo, R., Wagner, L., Wahlestedt, C., Wang, Y., Watanabe, Y., Wells, C., Wilming, L. G., Wynshaw-Boris, A., Yanagisawa, M., Yang, I., Yang, L., Yuan, Z., Zavolan, M., Zhu, Y., Zimmer, A., Carninci, P., Hayatsu, N., Hirozane-Kishikawa, T., Konno, H., Nakamura, M., Sakazume, N., Sato, K., Shiraki, T., Waki, K., Kawai, J., Aizawa, K., Arakawa, T., Fukuda, S., Hara, A., Hashizume, W., Imotani, K., Ishii, Y., Itoh, M., Kagawa, I., Miyazaki, A., Sakai, K., Sasaki, D., Shibata, K., Shinagawa, A., Yasunishi, A., Yoshino, M., Waterston, R., Lander, E. S., Rogers, J., Birney, E., and Hayashizaki, Y. (2002) Nature 420(6915), 563-573.

11. Huttenhofer, A., Kiefmann, M., Meier-Ewert, S., O'Brien, J., Lehrach, H., Bachellerie, J. P., and Brosius, J. (2001) EMBO J. 20(11), 2943-2953

12. Yuan, G., Klambt, C., Bachellerie, J. P., Brosius, J., and Huttenhofer, A. (2003) Nucleic Acids Res 31(10), 2495-2507

13. Kishore, S., and Stamm, S. (2006) Science 311(5758), 230-232

14. Torchet, C., Badis, G., Devaux, F., Costanzo, G., Werner, M., and Jacquier, A. (2005) Rna 11(6), 928-938

15. Edvardsson, S., Gardner, P. P., Poole, A. M., Hendy, M. D., Penny, D., and Moulton, V. (2003) Bioinformatics 19(7), 865-873

16. Schattner, P., Decatur, W. A., Davis, C. A., Ares, M., Jr., Fournier, M. J., and Lowe, T. M. (2004) Nucleic Acids Res 32(14), 4281-4296

17. Kiss, A. M., Jady, B. E., Bertrand, E., and Kiss, T. (2004) Mol Cell Biol 24(13), 5797-5807

18. Schattner, P., Barberan-Soler, S., and Lowe, T. M. (2006) Rna 12(1), 15-25

19. Balakin, A. G., Smith, L., and Fournier, M. J. (1996) Cell 86(5), 823-834

20. Ganot, P., Caizergues-Ferrer, M., and Kiss, T. (1997) Genes Dev 11(7), 941-956

21. Rivas, E., Klein, R. J., Jones, T. A., and Eddy, S. R. (2001) Curr Biol 11(17), 1369-1373

22. McCutcheon, J. P., and Eddy, S. R. (2003) Nucleic Acids Res 31(14), 4119-4128

23. Axmann, I. M., Kensche, P., Vogel, J., Kohl, S., Herzel, H., and Hess, W. R. (2005) Genome Biol 6(9), R73

24. Wassarman, K. M., Repoila, F., Rosenow, C., Storz, G., and Gottesman, S. (2001) Genes Dev 15(13), 1637-1651

25. Argaman, L., Hershberg, R., Vogel, J., Bejerano, G., Wagner, E. G., Margalit, H., and Altuvia, S. (2001) Curr Biol 11(12), 941-950

26. Washietl, S., Hofacker, I. L., Lukasser, M., Huttenhofer, A., and Stadler, P. F. (2005) Nat Biotechnol 23(11), 1383-1390

27. Liolios, K., Tavernarakis, N., Hugenholtz, P., and Kyrpides, N. C. (2006) Nucleic Acids Res 34(Database issue), D332-334

28. Fedorov, A., Stombaugh, J., Harr, M. W., Yu, S., Nasalean, L., and Shepelev, V. (2005) Nucleic Acids Res 33(14), 4578-4583

29. Eo, H. S., Jo, K. S., Lee, S. W., Kim, C. B., and Kim, W. (2005) Mol Cells 20(1), 35-42

30. Celniker, S. E., and Rubin, G. M. (2003) Annu Rev Genomics Hum Genet 4, 89-117

31. Grumbling, G., and Strelets, V. (2006) Nucleic Acids Res 34(Database issue), D484-488

32. Kent, W. J., Baertsch, R., Hinrichs, A., Miller, W., and Haussler, D. (2003) Proc Natl Acad Sci U S A 100(20), 11484-11489

33. Karolchik, D., Baertsch, R., Diekhans, M., Furey, T. S., Hinrichs, A., Lu, Y. T., Roskin, K. M., Schwartz, M., Sugnet, C. W., Thomas, D. J., Weber, R. J., Haussler, D., and Kent, W. J. (2003) Nucleic Acids Res 31(1), 51-54

34. Huang, Z. P., Zhou, H., He, H. L., Chen, C. L., Liang, D., and Qu, L. H. (2005) Rna 11(8), 1303-1316

35. Feynman, R. P. (1965) The Feynman lectures on physics, Addison-wesley, Reading, Mass.

36. Berezikov, E., Guryev, V., van de Belt, J., Wienholds, E., Plasterk, R. H., and Cuppen, E. (2005) Cell 120(1), 21-24

37. Meier, U. T. (2005) Chromosoma 114(1), 1-14

38. Hofacker, I. L. (2003) Nucleic Acids Res 31(13), 3429-3431

39. Zuker, M., and Stiegler, P. (1981) Nucleic Acids Res 9(1), 133-148

40. Hofacker, I. L., Priwitzer, B., and Stadler, P. F. (2004) Bioinformatics 20(2), 186-190

41. Rivas, E., and Eddy, S. R. (2000) Bioinformatics 16(7), 583-605

42. McCaskill, J. S. (1990) Biopolymers 29(6-7), 1105-1119

43. Bernhart, S. H., Hofacker, I. L., and Stadler, P. F. (2006) Bioinformatics 22(5), 614-615

44. Lyons-Weiler, J., Patel, S., and Bhattacharya, S. (2003) Genome Res 13(3), 503-512

45. Middendorf, M., Kundaje, A., Wiggins, C., Freund, Y., and Leslie, C. (2004) Bioinformatics 20 Suppl 1, I232-I240

46. Liu, J., Gough, J., and Rost, B. (2006) PLoS Genet 2(4), e29

47. Frank, E., Hall, M., Trigg, L., Holmes, G., and Witten, I. H. (2004) Bioinformatics 20(15), 2479-2481

48. Bazzan, A. L., Engel, P. M., Schroeder, L. F., and da Silva, S. C. (2002) Bioinformatics 18 Suppl 2, S35-43

49. Kretschmann, E., Fleischmann, W., and Apweiler, R. (2001) Bioinformatics 17(10), 920-926

50. Tobler, J. B., Molla, M. N., Nuwaysir, E. F., Green, R. D., and Shavlik, J. W. (2002) Bioinformatics 18 Suppl 1, S164-171

51. Witten, I. H., and Frank, E. (2005) Data mining : practical machine learning tools and techniques, 2nd Ed., Morgan Kaufman, San Francisco, Calif.

52. Quinlan, J. R. (1993) C4.5 : programs for machine learning, Morgan Kaufmann Publishers, San Mateo, Calif.

53. Gardner, P. P., and Giegerich, R. (2004) BMC Bioinformatics 5, 140

54. Washietl, S., Hofacker, I. L., and Stadler, P. F. (2005) Proc Natl Acad Sci U S A 102(7), 2454-2459