

# Markov Model Variants for Appraisal of Coding Potential in Plant DNA

Michael E. Sparks<sup>1</sup>, Volker Brendel<sup>1,2</sup>, and Karin S. Dorman<sup>1,2</sup>

<sup>1</sup> Department of Genetics, Development and Cell Biology, Iowa State University, Ames, IA 50011, USA

<sup>2</sup> Department of Statistics, Iowa State University, Ames, IA 50011, USA

**Abstract.** Markov chain models are commonly used for content-based appraisal of coding potential in genomic DNA. The ability of these models to distinguish coding from non-coding sequences depends on the method of parameter estimation, the validity of the estimated parameters for the species of interest, and the extent to which oligomer usage characterizes coding potential. We assessed performances of Markov chain models in two model plant species, *Arabidopsis* and rice, comparing canonical fixed-order,  $\chi^2$ -interpolated, and top-down and bottom-up deleted interpolated Markov models. All methods achieved comparable identification accuracies, with differences usually within statistical error. Because classification performance is related to G+C composition, we also considered a strategy where training and test data are first partitioned by G+C content. All methods demonstrated considerable gains in accuracy under this approach, especially in rice. The methods studied were implemented in the C programming language and organized into a library, `IMMpractical`, distributed under the GNU LGPL.

## 1 Introduction

Markov chain models, as applied to problems concerning gene recognition in DNA sequences, make the fundamental assumption that sequences of different functional roles exhibit distinct and reproducible dependencies among adjacent nucleotides, such that sequences can be distinguished by oligomer usage. In practice, Markov models appear to be a suitable proxy to the (unknown) generative models that have produced biologically relevant nucleic acid sequences, and they have enjoyed widespread use in popular gene prediction applications, including `GENSCAN` [1], `GlimmerM` [2], and `GeneMark.HMM` [3]. The Markov models used in these applications tend to be complex, and in most cases, only heuristic procedures exist for estimating Markov transition probabilities. Because both the validity of the Markov model assumption and the accuracy of the estimation procedures are unknown, it remains important to assess classification performance in novel applications. As this study is primarily motivated by the need to annotate plant gene structures, we used sequences from the model plant species *Arabidopsis thaliana* and *Oryza sativa* (rice).

There are a number of distinct methods for estimating Markov chain transition probabilities and selecting among models of varying complexity. Azad and

Borodovsky [4] undertook an empirical survey of fixed-order,  $\chi^2$ -interpolated [5,6], and top-down deleted interpolated Markov models [7] in prokaryotic taxa, and found considerable differences in the relative performances of each method as a function of genomic sequence characteristics, particularly G+C composition. The present study extends this work by comparing a greater breadth of training methods and by considering method performances in the context of two eukaryotic taxa. We show that, for the task of binary classification of coding and intron sequences from *A.thaliana* and rice, all the Markov model variants surveyed here (canonical fixed-order,  $\chi^2$ -interpolated, top-down and bottom-up deleted interpolated [7]) performed approximately equally. All Markov model variants were implemented in the C programming language and organized into a library, called `IMMpractical`, which is distributed under the GNU lesser/library general public license and is available for download at [8].

We also compared a *standard approach* that trains and tests without concern for the G+C composition of sequences, and a *quartiled approach* in which sequences are first partitioned into quartiles on the basis of overall G+C content, and quartile-specific transition probability estimates are used to classify. The latter strategy resulted in substantial improvements in classification accuracy relative to the standard approach, particularly in rice.

## 2 Materials and Methods

### 2.1 Data Accumulation

The success of any gene-finding algorithm to accurately classify sequences is largely dependent on how well the training data represent true coding and non-coding DNA. To obtain a reliable set of nuclear protein coding and intron sequences for training and testing purposes, we started with the current genome annotations for *A.thaliana* and rice available from the TAIR (version 6.0, [9]) and TIGR (version 4.0, [10]) resources, respectively.

As we were primarily interested in distinguishing coding sequences from introns in split genes, single exon genes were excluded. This exclusion also eliminated many processed pseudogenes, which are often intronless and share similar features with functional genes [11,12]. We ignored all loci with multiple gene models because these may be alternatively spliced [13], making coding/intron classification much more difficult [14].

Full-length coding sequences were parsed from assembled pseudomolecules based on reference coordinates, and if any ambiguous nucleotide symbols were encountered, the gene was discarded. Start and stop codons along with 5'- and 3'-UTRs were removed from the coding sequences, and only genes encoding translation products of 150 or more amino acids were retained. The selected sequences were compared to the TIGR plant repetitive element database [10] using BLASTN [15], and all coding sequences with significant matches (E-value  $< 10^{-15}$ ) were removed. We also used BLAST to limit redundancy in the coding data by randomly retaining only one member of each pair of sequences having at least

80% nucleotide identity over at least 80% of the length of both sequences. Reduction in dataset size during this refinement process is indicated in Table 1. Introns from the remaining gene structures were parsed from the pseudomolecules, leaving the concatenated exons as the coding data set. Introns that exceeded 50 nucleotides in length and contained no ambiguous characters were retained, and the resulting collection was made non-redundant using BLAST, as described above, to form the intron dataset. In total, we retained 15,538 coding sequences (mean length 1,467nt) and 87,477 intron sequences (mean length 159nt) from *A.thaliana*. In rice, 24,349 coding sequences (mean length 1,502nt) and 104,737 intron sequences (mean length 396nt) were retained. For the quartiled approach, coding and intron sequences were separated into quartiles according to their overall percent G+C composition (see Fig. 1).

**Table 1.** Number of genes excluded in the *A.thaliana* and *O.sativa* data sets at each refinement stage

Type Removed	<i>A.thaliana</i>	<i>O.sativa</i>
Annotated pseudogenes	3,818	0
Intronless genes	5,793	12,780
Alternatively spliced genes	2,887	4,280
Genes with ambiguous nucleotides	4	20
Genes with protein length < 150	2,009	7,433
Repetitive elements	60	7,379
Redundant genes	250	322
Total remaining	15,538	24,349

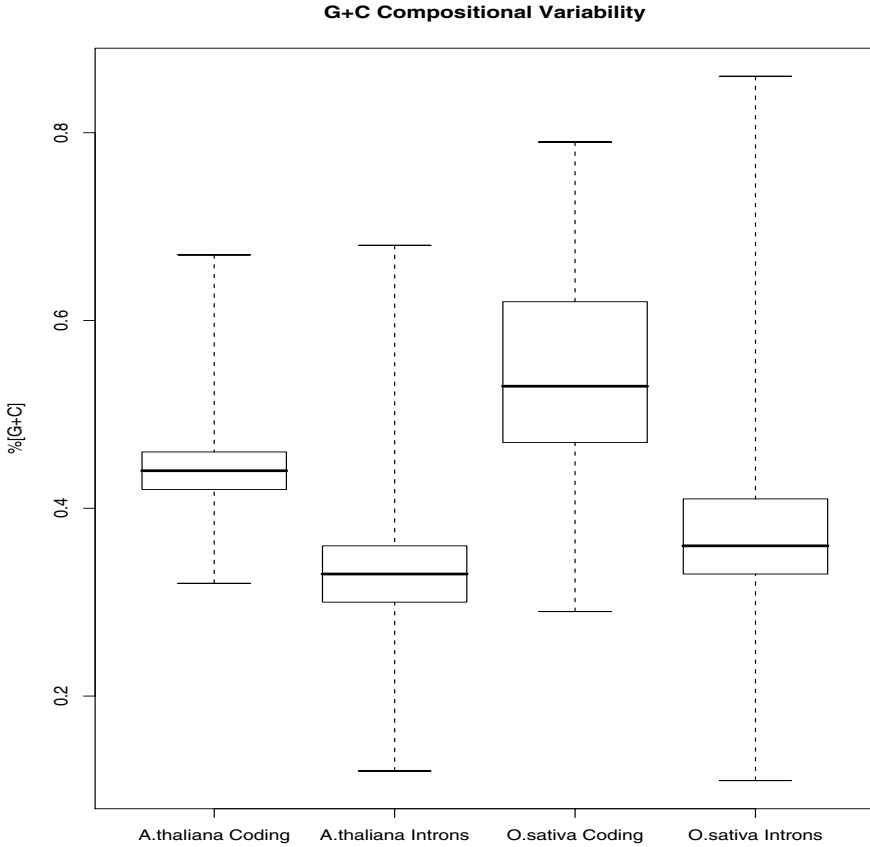
## 2.2 Fixed-Order Markov Models (FO)

For fixed-order methods, an order,  $k$ , is selected for the Markov chain based on empirical or statistical considerations—in practice, this is often set to five [1,3,11], and we also used this value. Let  $h_k$  represent some pretext, or history, of length  $k$  that precedes a nucleotide  $i$ . The fixed-order Markov chain has  $4^{k+1}$  transition probabilities, whose maximum likelihood estimates are

$$\hat{P}(i | h_k) = \frac{\text{Cnt}(h_k, i)}{\sum_{j \in \{A, C, G, T\}} \text{Cnt}(h_k, j)}, \quad (1)$$

where  $\text{Cnt}(h_k, x)$  is the count of oligomer  $h_k$  succeeded by some nucleotide  $x$  in the training data. Given a test sequence  $S = s_1 s_2 \cdots s_n$  of length  $n$ , the likelihood, assuming that the sequence belongs to some functional class  $t$  with maximum likelihood estimates  $\hat{\Omega}_t = \{\hat{P}(i | h_k)\}$ , is

$$P(S | t, \hat{\Omega}_t) = \prod_{j=1}^{n-k} \hat{P}(s_{j+k} | s_j s_{j+1} \cdots s_{j+k-1}).$$



**Fig. 1.** Box-and-whiskers plot showing variation in G+C percent composition for coding and intron sequences in *A.thaliana* and *O.sativa*. Each data set was partitioned on the quartiles, such that partitions contained roughly 3,884 coding and 21,869 intron sequences for *A.thaliana* and roughly 6,087 coding and 26,184 intron sequences for *O.sativa*, respectively.

For coding sequences, one recognizes the distinct properties of the three codon positions by computing one set of transition probabilities  $P^{(f)}(i | h_k)$  for each of the three reading frames,  $f = 1, 2, 3$ . There are now  $3 \times 4^{k+1}$  parameters to estimate for this inhomogeneous Markov chain model, and each is estimated by Eq. (1) with oligomer counts from the appropriate codon position. When simultaneously modeling a coding sequence shadow, parameters are also estimated for the three codon positions in the reverse complement [16].

Certain rare oligomers may not appear in the training data, resulting in null transition probabilities using Eq. (1). Any test sequence containing an unobserved oligomer is then impossible (has zero likelihood) under the estimated model. To avoid this problem, we use parameter smoothing, where for all

$h_k$  and  $i$ ,  $\text{Cnt}(h_k, i)$  is incremented by a fixed integer (in practice, five), ensuring at least a basal representation of all possible oligomers in the training data.

### 2.3 Interpolated Markov Models (IMMs)

The general paradigm of IMMs is that each transition probability is determined by taking linear, weighted sums of relevant fixed-order transition probabilities. For the transition probability with context  $h_k$ , fixed order transition probabilities for the pretext of length  $k$  and all shorter pretexts are used to produce the smoothed transition probability

$$P_{\text{imm}}(i | h_k) = \sum_{x=-1}^k \mu_x(h_k) \widehat{P}(i | h_x).$$

Here,  $\widehat{P}(i | h_{-1})$  is taken to be one over the cardinality of the nucleotide alphabet, i.e., 0.25.  $P_{\text{imm}}(i | h_k)$  is a probability when the weights  $\mu_x(h_k)$  satisfy  $0 \leq \mu_x(h_k) \leq 1$  for all  $x$  and  $\sum_{x=-1}^k \mu_x(h_k) = 1$ . To account for data sparsity, these models assign weights in terms of oligomer frequencies, preferentially giving more weight to oligomers with longer histories, unless they occur rarely enough in training data that more weight should be given to one of their 5'-truncated variants. Final, smoothed transition probabilities of oligomers whose histories do not occur in the training data are defined as  $P_{\text{imm}}(i | h_k) = P_{\text{imm}}(i | h_z)$ , where  $z = \max z' \in [1, k] : \text{Cnt}(h_{z'}) > 0$  and  $k$  is the maximum Markov chain order. We consider three distinct methods for estimating the smoothed transition probabilities as described in [4,5,6,7].

**$\chi^2$ -Interpolated Markov Models ( $\chi^2$ ).** The  $\chi^2$ -IMM defines transition probabilities iteratively as

$$P_{\text{chi}}(i | h_k) = \lambda(h_k) \widehat{P}(i | h_k) + [1 - \lambda(h_k)] P_{\text{chi}}(i | h_{k-1}), \quad (2)$$

with boundary condition  $P_{\text{chi}}(i | h_{-1}) = \widehat{P}(i | h_{-1})$ . The history weights for  $x = 0, \dots, k$  are

$$\lambda(h_x) = \begin{cases} 1 & \text{if } \text{Cnt}(h_x) \geq T; \\ 0 & \text{if } \text{Cnt}(h_x) < T \text{ and } q < 0.5; \\ \frac{q \times \text{Cnt}(h_x)}{T} & \text{otherwise.} \end{cases}$$

$T$  is some minimally-reliant count threshold for pretexts, e.g., 400; and  $q$  is the confidence (one minus the p-value) that the distribution of  $i | h_x$  differs from that of  $i | h_{x-1}$ ,  $i \in \{A, C, G, T\}$ , obtained by a  $\chi^2$  statistical test [5,6].

One possible scenario that is not addressed in any literature we encountered describing  $\chi^2$ -IMMs [5,6,17,4] is the condition where some pretext  $h_y$  occurs more than  $T$  times in the training data, but  $i | h_y$  does not occur for some nucleotide  $i$ . Then recursion (2) for computing  $P_{\text{chi}}(i | h_x)$  can generate problematic null transition probabilities, precisely the complication interpolated models were developed to avoid. For such cases, we used an approach similar to that described in [18] for correcting weight array matrices in splice site modeling:

$$P_{\text{fix}} = \frac{1}{\text{Cnt}(h_y)}$$

$$P_{\text{new}} = P_{\text{old}}(1 - 4 \times P_{\text{fix}}) + P_{\text{fix}},$$

where any null transition probability is re-assigned the value  $P_{\text{fix}}$ , and all remaining non-null probabilities in the distribution are adjusted to  $P_{\text{new}}$  as a function of  $P_{\text{fix}}$  and their previous values,  $P_{\text{old}}$ . Alternative solutions are described in [7].

**Top-down Deleted IMM (TDDI).** The basic idea of deleted IMM is to divide the training data into a large *development* set ( $D$ ) and a small *heldout* set ( $H$ )—the development set generates initial, unrefined transition probability estimates according to Eq. (1), which are generalized to the heldout set by cross-set maximization [7]. To prevent over-fitting to the heldout set, pretexts in the development set are partitioned into groups based on their frequencies, and all pretexts in a group are tied to the same weight. The pretext partitions are called buckets  $B_{x,m} = \{ h_x : \text{bound}_{x,m-1} \leq \text{Cnt}_D(h_x) < \text{bound}_{x,m} \}$ , where  $x$  indexes pretext length,  $m$  indexes the bucket, and  $\text{Cnt}_D$  indicates counts in  $D$  only. Bucket width is specified using a real-valued constant (e.g., 1.2, which is used in our implementation) dictating ratios of adjacent bucket boundaries.

Top-down deleted IMM-smoothed probabilities are computed by recursively solving, for  $x = 0, 1, \dots, k$ ,

$$P_{\text{TD}}(i | h_x) = \lambda_m(h_x)\widehat{P}(i | h_x) + [1 - \lambda_m(h_x)]P_{\text{TD}}(i | h_{x-1}), \tag{3}$$

with  $\lambda_m(h_x)$  values computed as

$$\operatorname{argmax}_{0 < \lambda < 1} \left\{ \sum_{\substack{i \in \{A,C,G,T\} \\ h_x \in B_{x,m}}} \text{Cnt}_H(h_x, i) \log \left[ \lambda \widehat{P}(i | h_x) + (1 - \lambda)P_{\text{TD}}(i | h_{x-1}) \right] \right\}, \tag{4}$$

and  $P_{\text{TD}}(i | h_{-1}) = \widehat{P}(i | h_{-1})$  again initializes the recursion.

**Bottom-up Deleted IMM (BUDI).** In the bottom-up deleted IMM approach, development pretexts of length  $k$  are partitioned into buckets  $B_{k,m}$  in similar fashion to the top-down variant. Each BUDI transition probability  $P_{\text{BU}}(i | h_k)$  is produced through a series of iterations initialized with

$$P^{(k)}(i | h_k) = \xi \widehat{P}(i | h_{-1}) + (1 - \xi)\widehat{P}(i | h_k), \tag{5}$$

for  $\xi = 10^{-5}$ . The recursion formula for the smoothing procedure is

$$P^{(l-1)}(i | h_k) = \lambda_{l,m}(h_k)P^{(l)}(i | h_k) + [1 - \lambda_{l,m}(h_k)]\widehat{P}(i | h_{l-1}), \tag{6}$$

starting at  $l = k$  and producing  $P_{\text{BU}}(i | h_k) := P^{(-1)}(i | h_k)$  upon termination when  $l = 0$ . Weighting factors  $\lambda_{l,m}(h_k)$  for the recursion are computed as

$$\operatorname{argmax}_{0 < \lambda < 1} \left\{ \sum_{\substack{i \in \{A,C,G,T\} \\ h_k \in B_{k,m}}} \text{Cnt}_H(h_k, i) \log \left[ \lambda P^{(l)}(i | h_k) + (1 - \lambda)\widehat{P}(i | h_{l-1}) \right] \right\}. \tag{7}$$

## 2.4 Accounting for G+C Content

We compared two approaches for fitting and using Markov chains with our data sets. The default method—the *standard approach*—involved producing a single set of transition probability estimates by training with all available data from each cross validation replicate; the same estimates were used to assay all test fragments. We also considered a *quartiled approach*, in which all sequences available for a given species were classified into quartiles on the basis of overall G+C composition (See Fig. 1). Coding and intron training sequences were quartiled separately, and quartile-specific Markov chains were estimated. Each test sequence was assigned a quartile based on its G+C content and likelihoods were computed using the appropriate Markov chains.

## 2.5 Test Design

The estimation methods were assessed on their abilities to correctly identify the—a *priori* known—functional class of a test sequence using the familiar Genmark framework [16]. Only binary classification of sequences as either coding or intron was tested. Likelihoods of test data were computed under  $N = 7$  Markov models: six coding Markov models for each frame of the forward  $f_1, f_2, f_3$  or shadow  $w_1, w_2, w_3$  strand; and a homogeneous Markov chain *itr* for the intron hypothesis. Prior probabilities are specified as follows. Let  $z$  be the (hypothesized) sequence type; then the prior is

$$P(z) = \begin{cases} 1/2 & \text{if } z = \textit{itr}; \\ \frac{1}{(N-1) \times 2} & \text{otherwise.} \end{cases} \quad (8)$$

Bayes rule provides the classifier:

$$P(z | S) = \frac{P(S | z) \times P(z)}{P(S)}, \quad (9)$$

where  $S$  is a test sequence.  $P(\textit{intron} | S) = P(\textit{itr} | S)$  is obtained directly from Bayes rule, and  $P(\textit{coding} | S) = \sum_{z \in \{f, w\}} \sum_{i=1}^3 P(z_i | S)$ . A sequence was classified as coding if  $P(\textit{coding} | S)$  exceeded 0.5—otherwise it was labeled as an intron.

We used a five-fold cross validation approach where, for each cyclic permutation, transition probabilities for each of the coding and intron classes were estimated using four of the data partitions, and methods were assayed against the remaining test partition. (Note that for the deleted IMM variants, three of the five data partitions were used for the development set, and one for the held-out.) Results from all five cross-validation replicates were pooled and averaged for final reporting.

To establish uniformity in training and testing sample sizes, we reduced the sizes of the five initial data partitions by randomly sampling a subset from each partition. For each species and sequence type, 3,000 random sequences were retained for the standard approach, and 750 from each bin in the quartiled

data. For test samples used under the standard approach, 2,500 fragments were randomly sampled from each test partition, for each of the coding and intron data sets, independently for both species; similarly, we randomly sampled 750 fragments from each bin in the quartiled method.

Normalizing for test sequence length is crucial for comparing performances of the methods at classifying sequences—longer test sequences would increase the odds of detecting a signal characteristic of the underlying generative model, and would tend to increase classification accuracy relative to shorter fragments. A fixed length of 96 nucleotides was used for assaying the methods under both the standard and quartiled approaches. A single test fragment was randomly parsed from each sequence among the test data partitions.

### 3 Results

Table 2 presents the average classification success of the Markov model training variants under the standard approach, for both species. Although the  $\chi^2$ -IMM achieved the maximum accuracy in all but one category, this advantage was not statistically significant. The only statistically significant differences (p-values < 0.01) were the poorer performance of FO compared to all three IMM variants in *A.thaliana* and the poorer performance of BUDI relative to the other IMM variants in rice. Notably, all methods were significantly less successful at the classification task in rice relative to *A.thaliana*.

**Table 2.** Mean success rates, averaged over five cross-validation replicates, for *A.thaliana* and *O.sativa* coding and intron sequences, under the standard approach. Values are given as percentages and standard deviations are shown in parentheses.

	<i>A.thaliana</i>			<i>O.sativa</i>			Overall
	Coding	Intron	Averaged	Coding	Intron	Averaged	
FO	96.78 (0.16)	94.96 (0.29)	95.87 (0.22)	87.15 (1.12)	86.89 (0.90)	87.02 (1.01)	91.44 (0.62)
TDDI	97.16 (0.24)	95.28 (0.45)	96.22 (0.34)	87.20 (0.61)	87.30 (0.65)	87.25 (0.63)	91.73 (0.49)
BUDI	96.95 (0.34)	94.99 (0.63)	95.97 (0.48)	86.28 (0.93)	86.45 (1.09)	86.37 (1.01)	91.17 (0.75)
$\chi^2$	97.20 (0.22)	95.31 (0.41)	96.25 (0.32)	87.42 (0.59)	87.29 (0.74)	87.36 (0.67)	91.81 (0.49)

We noticed that the success of classification varied considerably depending on the G+C content of the test sequence (data not shown). To address this problem, we partitioned the data into quartiles based on G+C content and trained quartile-specific Markov chains (the quartiled approach). Under this approach, classification performance still depended on G+C content as shown in Table 3 (only  $\chi^2$ -based results are shown, though all methods exhibit similar patterns). For coding sequences, method performance generally increased slightly with G+C composition, except in the fourth quartile, where a slight tapering in prediction accuracy was seen. In contrast, performance generally decreased as G+C composition increased for intron sequences, with a marked drop in performance in the fourth quartile.

**Table 3.** Quartile-specific mean coding and intron fragment identification success rates for the  $\chi^2$ -interpolated method, averaged over all five cross-validation replicates. Values are given as percentages and standard deviations are shown in parentheses.

Species	Class	1st	2nd	3rd	4th
<i>A.thaliana</i>	coding	98.11 (0.71)	99.47 (0.25)	99.41 (0.18)	99.12 (0.28)
	intron	99.89 (0.11)	99.55 (0.20)	98.61 (0.20)	92.85 (0.66)
<i>O.sativa</i>	coding	94.93 (0.88)	97.84 (0.53)	99.44 (0.15)	98.46 (0.58)
	intron	99.71 (0.15)	99.47 (0.14)	99.33 (0.16)	88.69 (1.65)

Despite the continued G+C content-dependent performance differences, the quartiled approach achieved a clear performance gain over the standard approach for all training methods (Table 4). The performance boost was moderate—though significant—for *A.thaliana* (roughly 2 – 3%) and even more dramatic for rice (roughly 10%). Importantly, all measures of classification performance improved under the quartiled approach.

**Table 4.** Comparison of classifiers under standard (std) and quartiled (qrt) approaches. Predictions across cross-validation replicates were pooled for a total of 30,000 distinct test cases. Classification measures, per [19], are Accuracy =  $\frac{TP+TN}{TP+TN+FP+FN}$ ; Sn (Sensitivity) =  $\frac{TP}{TP+FN}$ ; Sp (Specificity) =  $\frac{TN}{TN+FP}$ ; Corr. Co. (Correlation Coefficient) =  $\frac{TP \times TN - FP \times FN}{\sqrt{(TP+FN)(TP+FP)(TN+FP)(TN+FN)}}$ ; ROC AUC (Area under receiver operator characteristic curve, calculated using [20]); where *TP* are true positives; *FP*, false positives; *TN*, true negatives; *FN*, false negatives.

	FO		TDDI		BUDI		$\chi^2$		
	std	qrt	std	qrt	std	qrt	std	qrt	
<i>A.thaliana</i>	Accuracy (%)	95.83	98.01	96.29	98.41	96.11	98.42	96.27	98.38
	Sn (%)	96.84	98.53	97.29	99.04	97.21	98.99	97.33	99.03
	Sp (%)	94.93	97.52	95.39	97.81	95.12	97.88	95.31	97.76
	Corr. Co.	0.92	0.96	0.93	0.97	0.92	0.97	0.93	0.97
	ROC AUC (%)	99.02	99.67	99.11	99.73	99.08	99.74	99.11	99.73
<i>O.sativa</i>	Accuracy (%)	86.84	96.89	87.11	97.22	86.37	97.26	87.25	97.24
	Sn (%)	86.80	97.35	86.96	97.61	86.01	97.77	87.17	97.67
	Sp (%)	86.87	96.45	87.23	96.86	86.63	96.77	87.31	96.83
	Corr. Co.	0.74	0.94	0.74	0.94	0.73	0.95	0.75	0.94
	ROC AUC (%)	93.91	99.03	94.06	99.12	93.52	99.08	94.11	99.14

## 4 Discussion

While the gene structure prediction community has increasingly turned to gene annotation approaches dependent on homology information [21,22,23], the continued development of single-genome *ab initio* gene prediction tools remains worthwhile. Multi-genome gene prediction requires the presence of syntenic regions from two or more moderately divergent genomes. Genomic sequences from

related taxa do not always exist, and the optimal level of evolutionary divergence between such genomes remains unknown [22]. Indeed, even if requisite genomic data were abundant for all such gene annotation tasks, and the models worked perfectly, these methods would restrict attention to shared, homologous gene structures. Arguably, the complement of unique, species-specific genes, e.g., novel antifreeze glycoproteins in Arctic fish [24] and sex pheromones in moths [25], would be considerably more interesting for further experimental characterization by biologists. Thus, demand for highly sensitive single-genome gene prediction methods persists.

We have assayed the relative performances of a number of transition probability estimation methods for Markov chain models on coding and intron sequences of varying G+C composition in the model plant species *Arabidopsis thaliana* and *Oryza sativa*. Computational gene finders produced most gene annotations used to form our dataset [9,10], which could have biased the data to favor one model over another. Because prediction methods are not recorded [13], we were unable to test or correct for such bias. However, Fisher's exact tests on the accuracies we have computed show that most methods perform equivalently in the standard approach. Only FO is significantly worse in *A. thaliana*, and BUDI is significantly worse than the other IMM variants in rice. The fixed order model becomes statistically less accurate in both plant species under the quartiled approach, but the order  $k = 5$  may not be appropriate for the reduced size of quartiled data sets.

It is well known that classification success depends on G+C content (e.g., [1,4]). We observed that misclassified coding fragments are generally G+C-poorer than usual, while misclassified intron fragments are generally G+C-richer. In fact, the G+C profile of misclassified fragments loosely mimics that of correctly classified fragments in the competing functional category (data not shown). Markov models perform well when there is little overlap in oligomer usage between competing functional classes, but fail if overlap is considerable. Apparently, similar G+C content, a very simple indicator of monomer usage, also indicates substantial overlap in higher order oligomer usage. The G+C dependent performance motivated our quartiled approach, where training data are partitioned by G+C content and a Markov chain is trained for all partitions. Test sequences were first assigned a partition and then classified using partition-specific Markov chains. The deployment of all the models studied under a quartiled framework yielded considerable performance gains in both taxa, but most dramatically in rice (see Table 4).

The quartiled approach involves the estimation of more Markov chains, and presumably far more parameters; the fixed order Markov chain requires four times as many parameters under the quartile methodology, but the interpolated variants automatically adjust the parameter space according to data complexity. However, all methods, including the fixed order Markov chain, classified substantially better under quartile training. The results suggest that enhancing model complexity through chain order may not be the most efficient way to distinguish sequence class. Instead, the assumption that there is a single Markov chain generating each class is suspect. We are currently investigating mixture

models where multiple Markov chains generate each class. IMM variants can be seen as mixture models across chain orders, but the resulting parameterization may be overly restrictive.

Our results suggest that use of essentially any of the interpolated estimation methods, coupled with a G+C composition-specific (quartiled) framework, should improve gene annotation in plant genomic sequences, particularly in monocot species, including those with mature (rice) and emerging (maize and sorghum) genomic resources. The availability of our software to efficiently train Markov chains from species-specific and stratified data sets can facilitate incorporation of tailored parameter sets into general *ab initio* gene prediction programs.

**Acknowledgements.** We would like to thank two anonymous reviewers for helpful suggestions that improved this report. This work was supported in part by NSF Grant DBI-0606909 to V.B. M.E.S was also supported in part by the USDA with an IFAFS Multidisciplinary Graduate Education Training Grant (2001-52100-11506).

## References

1. Burge, C., Karlin, S.: Prediction of complete gene structures in human genomic DNA. *Journal of Molecular Biology* **268** (1997) 78–84
2. Majoros, W., Pertea, M., Antonescu, C., Salzberg, S.: **GlimmerM**, **Exonomy** and **Unveil**: three *ab initio* eukaryotic gene finders. *Nucleic Acids Research* **31** (2003) 3601–3604
3. Lukashin, A., Borodovsky, M.: **GeneMark.HMM**: new solutions for gene finding. *Nucleic Acids Research* **26** (1998) 1107–1115
4. Azad, R., Borodovsky, M.: Effects of choice of DNA sequence model structure on gene identification accuracy. *Bioinformatics* **20** (2004) 993–1005
5. Salzberg, S., Delchur, A., Kasif, S., White, O.: Microbial gene identification using interpolated Markov models. *Nucleic Acids Research* **26** (1998) 544–548
6. Delcher, A., Harmon, D., Kasif, S., White, O., Salzberg, S.: Improved microbial gene identification with **GLIMMER**. *Nucleic Acids Research* **27** (1999) 4636–4641
7. Potamianos, G., Jelinek, F.: A study of n-gram and decision tree letter language modeling methods. *Speech Communication* **24** (1998) 171–192
8. **IMMpractical**. <http://sourceforge.net/projects/immpractical/>
9. **TAIR**: The *Arabidopsis* Information Resource. <http://www.arabidopsis.org/>
10. **TIGR**: The Institute for Genomic Research. <http://www.tigr.org/>
11. Zhang, M.: Computational prediction of eukaryotic protein-coding genes. *Nature Reviews Genetics* **3** (2000) 698–709
12. van Baren, M., Brent, M.: Iterative gene prediction and pseudogene removal improves genome annotation. *Genome Research* **16** (2006) 678–685
13. **TIGR XML Specification**. <ftp://ftp.tigr.org/pub/data/DTDs/tigrxml.dtd>
14. Florea, L.: Bioinformatics of alternative splicing and its regulation. *Briefings in Bioinformatics* **7** (2006) 55–69
15. Altschul, S., Gish, W., Miller, W., Myers, E., Lipman, D.: Basic local alignment search tool. *Journal of Molecular Biology* **215** (1990) 403–410

16. Borodovsky, M., McIninch, J.: **GENMARK**: Parallel gene recognition for both DNA strands. *Computers in Chemistry* **17** (1993) 123–133
17. Salzberg, S., Pertea, M., Delchur, A., Gardner, M., Herve, T.: Interpolated Markov models for eukaryotic gene finding. *Genomics* **59** (1999) 24–31
18. Sparks, M., Brendel, V.: Incorporation of splice site probability models for non-canonical introns improves gene structure prediction in plants. *Bioinformatics* **21** (2005) iii20–iii30
19. Baldi, P., Brunak, S., Chauvin, Y., Andersen, C., Nielsen, H.: Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics* **16** (2000) 412–424
20. Sing, T., Sander, O., Beerenwinkel, N., Lengauer, T.: **ROCR**: visualizing classifier performance in R. *Bioinformatics* **21** (2005) 3940–3941
21. Guigó, R., Brent, M.: Recent advances in gene structure prediction. *Current Opinion in Structural Biology* **14** (2004) 264–272
22. Siepel, A., Haussler, D.: Computational identification of evolutionarily conserved exons. In: *Proceedings of the 8th Annual International Conference on Research in Computational Biology*. (2004) 177–186
23. Majoros, W., Pertea, M., Salzberg, S.: Efficient implementation of a generalized pair Hidden Markov model for comparative gene finding. *Bioinformatics* **21** (2005) 1782–1788
24. Chen, L., DeVries, A., Cheng, C.H.: Convergent evolution of antifreeze glycoproteins in Antarctic notothenioid fish and Arctic cod. *Proceedings of the National Academy of Sciences, USA* **94** (1997) 3817–3822
25. Roelofs, W., Liu, W., Hao, G., Jiao, H., Rooney, A., Linn Jr., C.: Evolution of moth sex pheromones via ancestral genes. *Proceedings of the National Academy of Sciences, USA* **99** (2002) 13621–13626