Supplementary Material for the Manuscript

mGene: Accurate SVM-Based Gene Finding with an Application to Nematode Genomes

Gabriele Schweikert,[1,2,3] Alexander Zien,[1,4,†] Georg Zeller,[1,3,†] Jonas Behr,[1] Christoph Dieterich,[3,‡] Cheng Soon Ong,[1,2,§] Petra Philips,[1] Fabio De Bona,[1] Lisa Hartmann,[1] Anja Bohlen,[1] Nina Krüger,[1] Sören Sonnenburg,[4,1] Gunnar Rätsch[1,*]

[1] Friedrich Miescher Laboratory, Max Planck Society, Tübingen, Germany
[2] Max Planck Institute for Biological Cybernetics, Tübingen, Germany
[3] Max Planck Institute for Developmental Biology, Tübingen, Germany
[4] Fraunhofer Institute FIRST.IDA, Berlin, Germany

Contents

A Additional Results: the nGASP competition **2**
A.1 Original Evaluation Results by the Competition Organizers 2
A.2 Performance of the Individual **mGene** Signal Detectors 2

B Additional Results: Genome-wide Predictions for C. elegans **4**
B.1 Summary Statistics 4
B.2 Analysis of Novel Genes 5

C Additional Results: Genome-wide Predictions for Five Nematodes **6**
C.1 Summary Statistics 6
C.2 Comparative Analysis 7

D Availability of the Source Code **8**
D.1 **mGene**: Original Code for mGene.init, mGene.multi and mGene.seq . . . 8
D.2 **mGene**: Standalone Tools for Training and Prediction 8

E Methods: mGene **9**
E.1 Layer 1: Signal Detectors 9
E.2 Layer 1: Content Detectors 11
E.3 Layer 1: Incorporating Additional Information 11
E.4 Layer 2: State Model for Gene Structure Prediction 12
E.5 Layer 2: Gene Structure Scoring Function 13
E.6 Layer 2: Training 14
E.7 nGASP: Experimental Protocol 16
E.8 Methodological Differences to the Developmental Versions 16
E.9 Genome-wide Gene Predictions for Five Nematodes 17

F Methods: Verification of Predictions by RT-PCR **17**
F.1 Experimental Protocol and Evaluation 17
F.2 Tables 19

*Correspondence should be addressed to: Gunnar.Raetsch@tuebingen.mpg.de.
†Authors contributed equally.
‡Current address: Max Delbrück Center for Molecular Medicine, Berlin.
§Current address: Department of Computer Science, ETH, Zürich, Switzerland.

A Additional Results: the nGASP competition

A.1 Original Evaluation Results by the Competition Organizers

As an indication that our own numerical evaluation of gene finder accuracies in the nGASP setting is appropriate, we reproduce an excerpt from the results table of the nGASP evaluation (Coghlan et al., 2008) in Table S1. Shown are values for the same gene finders as in Table 1 of the main text, except for the improved versions of mGene, which did not participate in nGASP. While there are several numerical differences to our own evaluation on the transcript and gene levels, they are small, and the ranking of the methods by their accuracy remains unchanged except for two cases, where Augustus and Fgenesh perform very similar. To better compare the results, we have added the average of sensitivity and specificity for each evaluation criteria.

		Base			Exon			Transcript			Gene		
Cat.	Method	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$
1	mGene.init (dev) (v3)	96.9	91.6	**94.2**	84.2	78.6	**81.4**	43.5	40.5	**42.0**	53.4	44.8	49.1
1	Craig	95.6	90.9	93.2	80.2	78.2	79.2	35.7	36.3	36.0	43.8	37.8	40.8
1	Eugene	94.0	89.5	91.7	80.3	73.0	76.6	49.1	28.8	39.0	60.2	30.2	45.2
1	Fgenesh	98.2	87.1	92.7	86.4	73.6	80.0	47.1	34.6	40.9*	57.8	35.4	46.6
1	Augustus (v1)	97.0	89.0	93.0	86.1	72.6	79.3	50.1	28.7	39.4*	61.1	38.4	**49.7**
2	mGene.multi (dev) (v2)	97.7	90.9	**94.3**	85.8	78.3	**82.1**	51.2	40.9	**46.1**	62.7	43.8	**53.3**
2	N-Scan	97.4	88.1	92.7	83.5	70.8	77.2	39.2	27.7	33.4	48.1	28.4	38.2
2	Eugene	96.2	87.5	91.9	82.8	72.8	77.8	50.3	30.2	40.2	61.7	31.4	46.5
3	mGene.seq (dev) (v3)	98.7	91.9	95.3	91.0	80.6	85.8	57.7	48.0	52.9	70.6	51.1	60.9
3	Gramene (v1)	98.2	95.4	**96.8**	88.5	71.8	80.1	41.7	19.6	30.6	48.7	37.2	42.9
3	Fgenesh++	97.6	89.7	93.6	90.4	80.9	85.7	65.5	53.4	**59.5**	78.3	54.2	**66.3***
3	Augustus (v1)	99.0	90.5	94.7	92.5	80.2	**86.3**	68.3	47.1	57.7	80.1	51.8	66.0*
4	Evigan	99.3	89.6	94.4	91.1	82.3	86.7	64.2	52.4	58.3	80.7	52.7	66.7
4	Jigsaw (v1)	98.9	93.2	**96.1**	90.5	87.4	**88.9**	63.6	60.2	**61.9**	79.9	61.0	**70.5**

Table S1: nGASP evaluation of selected gene predictors. Sensitivities (Sn) and specificities (Sp) are taken from (Coghlan et al., 2008); to facilitate comparisons, we further report their averages. The numbers slightly deviate from our own evaluation on the transcript and gene level due to minor differences in the evaluation criteria. These differences, however, do not change the ranking except in two cases, where the performances are very close together (relevant results are marked with an asterisk). For the sake of completeness this table also includes performance estimates for Category 4 submissions, including Jigsaw and Evigan (Liu et al., 2008). Note, however, that these performance measures are not directly comparable to the other categories, because more training data and different test data were used.

A.2 Performance of the Individual mGene Signal Detectors

In order to verify the validity of the signal detectors devised for mGene, we assess their accuracies for the corresponding signal detection tasks on the nGASP training regions. Positive examples are deduced from the nGASP training annotation; negative examples are sampled as described in Section E.1. In a five-fold cross validation scheme, signals are trained on three fifths of the nGASP training data only, and hyper-parameters are tuned on one fifth of the data, such that predictions can be evaluated on the remaining fifth. This procedure is repeated five times in order to obtain predictions for the whole genome.

The accuracy of each signal detector is quantified by comparing it to the respective annotation. In Table S2 we report the area under the precision-recall-curve (auPRC) (Metz, 1978) and the area under the receiver operating characteristic curve (auROC) (Davis and Goadrich, 2006). Note, that an optimal classifier achieves a value of 1 for both measures, while random guessing leads to a value of 0.5 for auROC and a value depending on the relative sizes of the two classes for auPRC. Due to the sub-sampling of the negative examples, positive examples may be over-represented in both training and evaluation as compared to the real-life situation of predicting on genomic sequences (in particular for transcription start and cleavage sites). We therefore also report the fraction of positives in the generated sets of examples.

	TSS	Trans	TIS	Stop	Acc	Don	PolyA	Cleave
Ratio of positives	0.01	0.03	0.02	0.01	0.02	0.01	0.01	0.01
auPRC (nGASP)	0.57	0.33	0.31	0.30	0.88	0.82	0.14	0.24
auROC (nGASP)	0.95	0.89	0.88	0.93	0.99	0.99	0.87	0.92

Table S2: Evaluation of individual signal detectors: transcription start sites (TSS), trans-splice sites (Trans), translation start sites (TIS), translation termination sites (Stop), acceptor splice sites (Acc), donor splice sites (Don), poly-adenylation sites (PolyA) and cleavage sites (Cleave). Given is the ratio of positive examples among the labeled examples and the areas under the precision-recall-curve (auPRC) and under the receiver operating characteristic curve (auROC) for predictions on the nGASP training regions.

While Table S2 indicates that our signal detectors work reasonably well when applied individually, the accuracies further increase when information from surrounding (different) signals is also taken into account. This is exactly what a full gene finder like mGene does. In Table S3 we evaluate the most accurate *ab initio* gene finders of nGASP when used to detect individual signals. mGene yields the most accurate signal recognition for five out of six signals. The only exception is the cleavage site where it was second to Augustus, suggesting that mGene could be further improved by adopting a cleavage site model similar to that of Augustus.

	TSS			TIS			Acc			Don			Stop			Cleave		
Method	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$
mGene.init	32.8	11.5	**22.2**	71.1	65.4	**68.2**	88.9	85.54	**87.2**	90.0	85.7	**87.9**	80.3	76.4	**78.4**	29.5	15.7	22.6
mGene.init (dev)	28.4	10.7	19.5	66.6	67.5	67.1	88.7	85.02	86.8	89.8	84.9	87.3	76.8	79.2	78.0	15.5	10.1	12.8
Craig	16.7	7.0	11.9*	57.1	65.2	61.2	85.8	84.78	85.3	86.9	85.2	86.0	70.7	77.5	74.1	3.4	2.1	2.8*
Eugene	17.0	4.3	10.7	70.7	46.3	58.5	84.0	78.65	81.3	86.7	81.0	83.8	86.0	59.5	72.8	16.9	7.3	12.1
Fgenesh	21.4	6.5	14.0*	73.1	57.6	65.4	90.7	79.20	85.0	91.3	79.5	85.4	81.5	66.7	74.1	3.9	1.6	2.8*
Augustus	15.8	4.3	10.1	72.4	50.6	61.5	89.7	78.92	84.3	90.8	79.8	85.3	86.2	65.6	75.9	38.7	17.11	**27.9**

Table S3: Sensitivity (Sn), specificity (Sp) and their average for various signal predictions by ab initio gene finding systems (Category 1): transcription start sites (TSS), translation initiation sites (TIS), acceptor (Acc) and donor (Don) splice sites in coding regions, translation termination sites (Stop), cleavage sites (Cleave). The evaluation was performed on the two gold standard gene sets of the nGASP test regions (referred to as **ref1** and **ref2** in (Coghlan et al., 2008)). For TSS and Cleave the evaluation is similar to the promoter evaluation at the EGASP competition (Guigó et al., 2006): Within 20 nucleotides of an annotation a prediction is counted as correct. However, for each annotation only one positive is counted. Alternative annotations are counted more than once if a prediction is within 20 nucleotides of both annotations. Note that Craig and Fgenesh are not able to predict untranslated regions. We therefore used the translation start and stop as transcription start and cleavage site predictions (affected results are marked with $*$).

Because start and end sites of transcription are particularly difficult to predict correctly, even as parts of entire predicted genes, we investigate the deviation of the predictions from the annotation in more detail. Figure S1 depicts the distributions of predicted gene starts and ends relative to the closest corresponding site of a high-quality gene annotation (referred to as **ref2** in Coghlan et al., 2008). For this analysis we have excluded annotated genes with a UTR length smaller than 10 nucleotides, because we assume that these annotated UTRs are very likely to be incomplete. The resulting set consists of 430 and 587 annotated transcription start and cleavage sites, respectively.

For the gene start predictions, mGene is extremely accurate, with low variance and essentially no bias. This can be attributed in part to the trans-splicing model of mGene rather than to the TSS signal detector alone. Eugene also produces a favorable distribution of TSS predictions, while Augustus is biased towards too long 3' UTRs.

For the cleavage sites, Eugene again does not seem to consistently prefer too short or too long UTRs, but often misses the true boundaries (high variance). While mGene does reasonably well, with medium variance and a tendency to predict too long 3' UTRs, Augustus can be seen to place a higher fraction of predictions close to the annotated site.

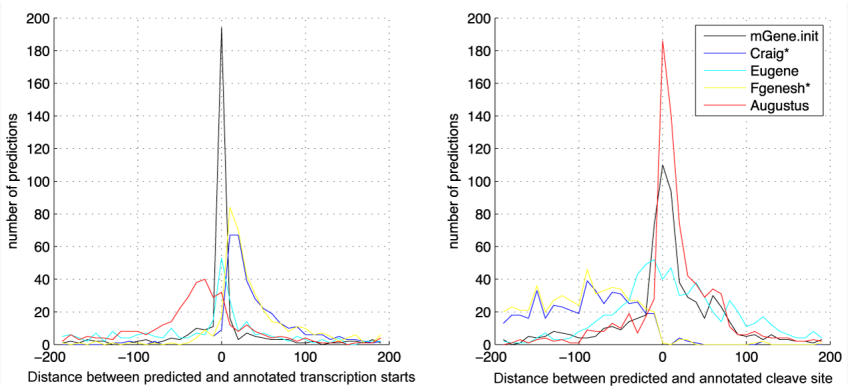


Figure S1: Distributions of predicted gene starts (left: TSS) and ends (right: Cleave) relative to the closest corresponding site (according to the annotation) on the nGASP test regions. As Craig and Fgenesh do not predict UTRs, the translation boundaries are used as transcription start and cleavage site, which results in consistent under-prediction of the UTRs. Note that the total numbers of predictions can be different for each gene finder.

B Additional Results: Genome-wide Predictions for C. elegans

B.1 Summary Statistics

The genome-wide gene predictions by mGene are available at `http://mgene.org/predictions` and on the Wormbase genome browser. Additionally, we provide the predictions of individual signal sites in suitable formats for two popular genome browsers (Wormbase and UCSC). These predictions may help human curators to annotate genes in the future.

In this section we provide a more detailed comparison of the genome-wide predictions for C. elegans. In Table S4 we show a summary of the main differences between the C. elegans annotation WS180 and the gene predictions of the considered methods mGene.seq, Fgenesh++, Augustus, and Jigsaw. Additionally, we show in Figure S2, distributions of

Method	No. of genes	No. exons/gene		Exon length [bp]		Intron length [bp]		ORF length [bp]		No. of corr. genes	
		mean	median	mean	median	mean	median	mean	median	All	Conf.
mGene.seq	21,489	6.3	5.0	200.3	146.0	327.3	65.0	1248.9	990.0	11,392	5343
Fgenesh++	23,368	5.7	5.0	202.5	146.0	305.8	71.0	1150.4	939.0	16,603	5413
Augustus	21,525	6.1	5.0	202.1	146.0	294.7	64.0	1174.2	945.0	12,136	5639
Jigsaw	20,423	6.1	5.0	207.6	148.0	297.7	64.0	1252.4	996.0	13,851	5621
WS180	20,133	6.5	5.0	203.5	147.0	292.7	63.0	1229.0	996.0	-	-

Table S4: Summary statistics of the predictions on the C. elegans genome of Fgenesh++, Augustus, Jigsaw, and mGene.seq. Reported are the predicted number of genes, average and median number of exons per gene, exon length, intron length, and ORF length. Additionally, we report the number of all and all confirmed annotated genes, for which at least one transcript is predicted completely correct by a given gene finder.

exon, intron, number of exons per transcript, and coding region length for genes in the C. elegans annotation WS180 and mGene's predictions as well as for newly discovered and missing genes. We can observe that exon and intron length distributions agree well among all gene sets. However, for missing and newly predicted genes, we find a considerably smaller number of exons per gene and shorter coding regions than for an average gene. From Table S4, we can observe that all gene finders predict a larger number of genes, but fewer exons per gene. It seems possible that this is due to the suboptimal detection of the transcript boundaries and the erroneous splitting of a transcript into two or more pieces. Another possibility is supported by the above observation in Figure S2: gene predictors find new genes that have a considerably smaller number of exons than average genes and, hence, the average number of exons per genes is reduced.

The distribution of ORF lengths in mGene predictions shows a strong peak at a

length of 1000 bp. It turns out that this peak is likewise observable in all other nGASP prediction sets (Fgenesh, Augustus, Jigsaw, data not shown) and preserves quite well the same feature in the WS180 annotation (and also in earlier annotations and in the nGASP training genes). However, when analyzing the ORF lengths of the annotation WS180 in terms of confirmed, partially-confirmed and predicted genes, this peak is only present in the predicted set of genes and the partially confirmed set (data not shown). We speculate, that this peak could be an artifact of the original genefinder predictions that were used for generating the annotation (personal communication with L. Stein, 2008). However, it is interesting to see how the subsequently applied gene finders all mimic the same distribution. This stresses the importance of training set selection: To avoid the propagation of systematic errors, gene finders should only be trained on a set of highly trusted genes.

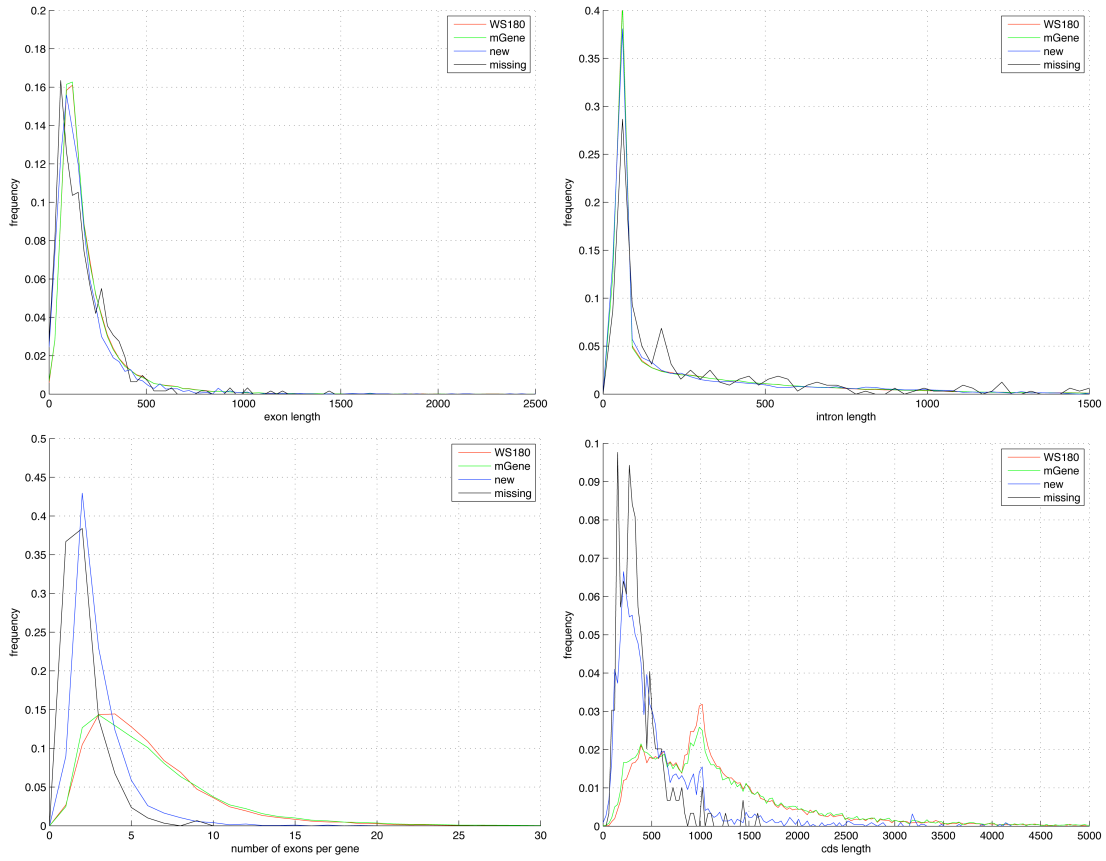


Figure S2: Distributions of exon, intron and coding region lengths, as well as number of exons per gene.

B.2 Analysis of Novel Genes

We annotated the protein sequences of all 2,197 novel genes with Interpro domains (see Methods). Figure S3 depicts the ten most abundant domain categories expressed in terms of matching novel genes. The seven-transmembrane chemo-receptors (103 genes) constitute the largest addition to the current C. elegans annotation. For most chemoreceptor families, C. elegans has the largest number of genes when compared to C. briggsae, suggesting changes in the importance of chemoperception between the species (Thomas and Robertson, 2008). Another interesting group are the F-box proteins. It has been hypothesized, based on patterns of molecular evolution, that most members of the F-box superfamily are adapters that target foreign proteins for proteolysis. Thomas (Thomas, 2006) speculates that this system functions to combat viral pathogens or bacterial protein toxins.

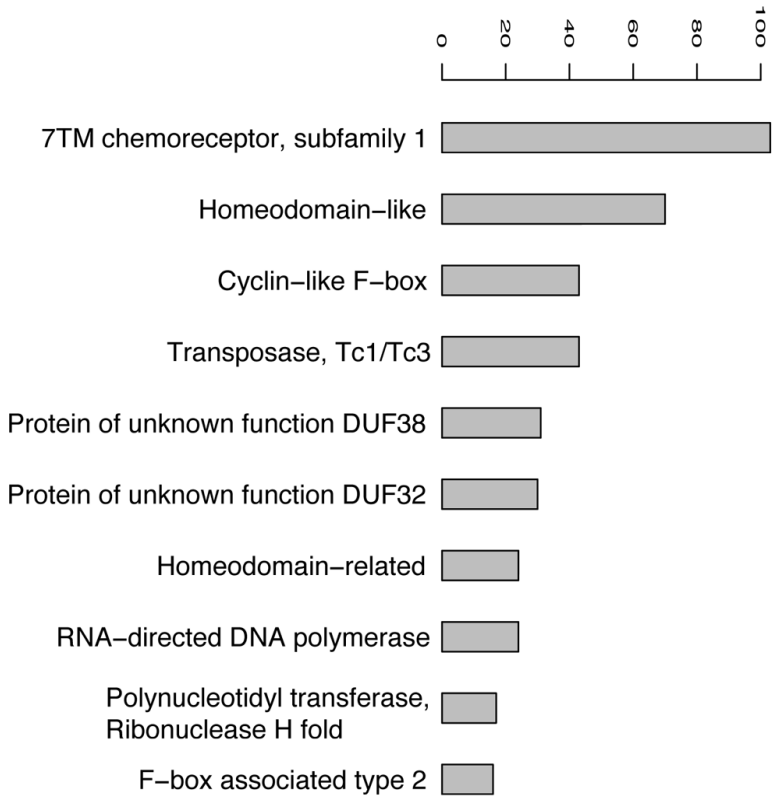


Figure S3: The ten most abundant protein domain annotations for the group of novel mGene genes.

C Additional Results: Genome-wide Predictions for Five Nematodes

C.1 Summary Statistics

We investigate to what extent mGene is able to generalize from the species it has been trained for – here C. elegans – to related, but different species. Here, we focus on other nematodes, namely, C. remanei, C. japonica, C. brenneri, and C. briggsae. Table S5 provides a comparison of genome-wide mGene.seq predictions on all five nematode genomes. The predictions were produced with mGene.seq trained on C. elegans, but applied in a Category 3 setting, i.e. we utilized alignments of protein and EST sequences available at prediction time (see Section E.3). Here, we used the confirmed protein sequences (8,134 fully EST/cDNA confirmed translated coding regions) from the C. elegans annotation WS180 for alignment against the five nematodes' genomes. For C. elegans we used the C. briggsae (19,334) and Saccharomyces cerevisiae (11,081) annotated peptide sequences. Furthermore, we obtained mRNA sequences from dbEST (Boguski et al., 1993) and the NCBI Nucleotide database for C. remanei (20,292 EST sequences), C. briggsae (2,424 EST and 108 other sequences), C. brenneri (2,474 EST and 6 other sequences), and C. elegans (346,064 sequences).

As Table S5 shows, not only the genome sizes vary considerably, but also the numbers and the lengths of predicted introns show strong variability. Note that the true numbers are not yet known for all five species; in some cases the huge numbers of contigs indicate that even the genome assembly is far from being finished.

Of course, the question is whether the different characteristics of the predictions reflect mGene's inability to adapt to evolutionary distant genomes, or whether, on the contrary, they are real and faithfully uncovered by mGene. Furthermore, we are also interested in seeing how the performance of mGene in this respect compares to that of the best competing gene finders. To answer both questions, we downloaded genome-wide

Genome	No. of contigs	Genome size [Mbp]	No. of genes	No. exons/gene		Exon length [bp]		Intron length [bp]		ORF length [bp]	
				mean	median	mean	median	mean	median	mean	median
C. elegans	6	100.27	21489	6.3	5.0	200.3	146.0	327.1	65.0	1249.5	990.0
C. remanei	3670	235.94	31503	5.7	4.0	212.6	148.0	284.3	52.0	1186.1	951.0
C. japonica	4657	266.90	20121	5.3	4.0	228.2	147.0	598.6	71.0	1180.0	879.0
C. brenneri	10292	453.09	41129	5.4	4.0	226.1	152.0	288.8	53.0	1194.6	942.0
C. briggsae	12	108.48	22542	6.0	5.0	206.9	148.0	348.7	54.0	1231.9	957.0

Table S5: Summary of gene characteristics for genome-wide predictions by mGene.seq on five nematode genomes. Shown are basic properties of the genomes, the number of predicted genes as well as the average and median exon, intron and ORF lengths.

predictions for all considered nematodes with the Category 3 versions of Fgenesh, Augustus, and Jigsaw (Category 4) from the nGASP ftp site (`ftp://ftp.wormbase.org/pub/wormbase/nGASP/final_gene_predictions/predictions/*/processed/GFF3`). We assessed the predictions by comparing them to EST alignments which contain information about introns and internal exons. We therefore restricted the evaluation to the sensitivity and specificity of identifying internal exons. For this evaluation, we obtained the most recent sets of EST and mRNA sequences from the NCBI nucleotide databases leading to 33,050, 29,929, and 16,537 new EST sequences for C. japonica, C. brenneri, and C. briggsae, as well as 143 new mRNA sequences for C. remanei. The results in Table S6 suggest that mGene retains a very high prediction accuracy on the other nematodes except for C. briggsae, where a non-negligible drop is observed. However, the other gene finders are hit substantially harder. Note that mGene consistently outperforms the other three gene finders with respect to sensitivity and specificity, while never being substantially worse by any measure.

	No. of exons		Fgenesh			Augustus			Jigsaw			mGene.seq		
	total	new	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$	Sn	Sp	$\frac{Sn+Sp}{2}$
C. elegans	56970	0%	85.43	94.61	90.02	90.11	96.06	93.09	88.12	99.13	93.62	90.06	99.00	**94.53**
C. remanei	11500	1%	81.21	93.52	87.36	91.70	95.93	93.81	83.74	95.26	89.50	94.33	98.79	**96.56**
C. japonica	8385	100%	72.30	94.82	83.56	81.35	96.07	88.71	82.60	95.76	89.18	88.85	97.65	**93.25**
C. brenneri	12832	84%	73.56	93.12	83.34	82.50	93.15	87.83	75.16	97.05	86.11	89.16	96.9	**93.05**
C. briggsae	64968	95%	65.70	84.69	75.20	76.59	87.48	82.04	74.57	85.41	79.99	85.13	88.81	**86.97**

Table S6: Consensus of internal exons inferred from aligned EST sequences with exons predicted by mGene, Fgenesh++, Augustus, and Jigsaw. We counted a predicted exon as correct if both boundaries were correct, and as a false prediction if it overlapped a region covered by an EST alignment, but did not exactly match an EST-confirmed exon. Shown are sensitivity (Sn), specificity (Sp), and their average. Note that we observe higher values for C. elegans than in Table 1 (main text), as we here only evaluate regions covered by an EST alignment. Also provided are numbers of considered exons found by EST alignments and fractions thereof that have not been utilized for prediction by the gene finding systems.

The values in Table S6 might not be fully conclusive, since some of the ESTs which have been used for evaluating prediction accuracy had been utilized in computing the predictions before. The second and third columns indicate the total number of internal EST-confirmed exons and what fraction of that is new to the evaluation, i.e. has not previously been utilized for prediction. Such ESTs exist due to the time gap between prediction and evaluation and the intermittent growth of dbEST. This effect varies strongly over the considered species, which may make it difficult to interpret the results. Consequently, we restricted the evaluation to exons that had not been covered by any EST utilized at prediction time. The corresponding results can be found in Table S7. Here, one difficulty is that the values for C. remanei may not be fully reliable due to the low number of (newly) EST-confirmed exons. Nevertheless, the results essentially confirm the conclusions drawn from Table S6.

C.2 Comparative Analysis

We carried out a comparative analysis with Multiparanoid (Alexeyenko et al., 2006) on the newly predicted proteomes of C. elegans (21,489 predictions), C. briggsae (22,542

	N	Fgenesh			Augustus			Jigsaw			mGene		
		Sp	Sn	$\frac{Sn+Sp}{2}$	Sp	Sn	$\frac{Sn+Sp}{2}$	Sp	Sn	$\frac{Sn+Sp}{2}$	Sp	Sn	$\frac{Sn+Sp}{2}$
C. elegans	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
C. remanei	82	101.33*	92.68	**97.01**	97.53	96.34	96.94	95.77	82.93	89.35	100.00	93.90	96.95
C. japonica	8385	94.82	72.30	83.56	96.07	81.35	88.71	95.76	82.60	89.18	97.65	88.85	**93.25**
C. brenneri	10754	93.19	74.02	83.60	93.41	83.17	88.29	96.94	75.53	86.23	96.58	89.19	**92.89**
C. briggsae	62210	84.72	65.60	75.16	87.44	76.40	81.92	85.39	74.43	79.91	88.64	85.02	**86.83**

Table S7: Summary of genome-wide predictions on five nematode genomes (see description in Table S6). Here, the evaluation was done only on exons that were not confirmed by an EST alignment used for gene predictions. *The submission by Fgenesh includes a small number of overlapping genes. Correctly predicted exons might therefore be counted more than once.

predictions), C. brenneri (41,129), C. remanei (31,503), and C. japonica (20,121). This analysis served two purposes: first to determine gene homology relations, and second to estimate the number of species-specific gene inventions. We could identify orthology relations for a proportion of $63-84\%$ of a specie's gene set, with C. brenneri being at the bottom and C. elegans at the top of the list (Table S8).

Out of all the species-specific gene predictions, we found 541 (50%) for C. elegans, 676 (27%) for C. briggsae, and only a very small fraction for the other nematodes matching entries in the Uniref90 database. This is not surprising as the protein sequences of these two well-annotated genomes have already been deposited in the Uniref database, while those of the other nematodes have not yet been entered.

	C. briggsae	C. remanei	C. brenneri	C. elegans	C. japonica
One:One orthologs	5,339 (23.7%)	5,339 (17.0%)	5,339 (13.0%)	5,339 (27.3%)	5,339 (26.5%)
Other orthologs	11,946 (53.0%)	16,588 (52.6%)	20,459 (49.7%)	11,002 (56.2%)	8,387 (41.7%)
Protein similarity	2,709 (12.0%)	4,819 (15.3%)	8,324 (20.2%)	2,150 (11.0%)	2,736 (13.6%)
No similarity	2,548 (11.3%)	4,757 (15.1%)	7,007 (17.0%)	1,092 (5.6%)	3,659 (18.2%)

Table S8: Comparison of the gene sets predicted by mGene.seq for different nematodes: Shown are the number of protein coding genes predicted for each organism with one-to-one orthologs, other orthologs, with weak, and with no significant protein sequence similarity.

D Availability of the Source Code

D.1 mGene: Original Code for mGene.init, mGene.multi and mGene.seq

The source code of the versions mGene.init, mGene.multi and mGene.seq that have been used for generating the results presented in this work will be provided on our website. The intention for releasing this source code—together with initial training data, the trained model, and the test regions—is to enable the reproducibility of our results. These mGene versions will be provided under an academic license and will only be supported to reproduce the published results.

D.2 mGene: Standalone Tools for Training and Prediction

Additionally, we provide the source code of a mature version of mGene.init licensed under GPL version 3. This is a further developed version that uses a simpler and thus more general state model than the original version (i.e. no trans-splicing and no operon prediction). It is based on C/C++ and Octave and will be fully supported. The same version is also available in the mGene.web webservice (Schweikert et al., 2009).

The simplest way to use mGene is to use it via the following three commands:

`mgene_train_workflow` uses a FASTA file with genomic sequences and a (partial) annotation of these sequences in GFF3 format in order to train a complete mGene model.

`mgene_predict_workflow` uses a trained mGene Predictor (in a format called, TmGP) as well as a FASTA file with genomic sequences and performs genome-wide gene predictions.

`mgene_eval_workflow` compares two annotations in GFF3 format, presumably a gold standard and a prediction gene set and reports accuracy values.

Additionally, we provide command-line tools for individual sub-tasks needed for gene finding: These tools can be sorted into three groups:

Data preparation including the `mgene_genometool` for creating an efficient internal data structure that contains the genome sequence, and `mgene_gff2anno` for generating an internal data structure storing a genome annotation.

Signal training and prediction including the tools `mgene_anno2signallabel`, to generate labeled genomic signal locations from a genome annotation, `mgene_signal_train`, `mgene_signal_predict`, and `mgene_signal_eval` for training, genome-wide prediction, and performance evaluation of signal predictors.

Content training and prediction including the tools `mgene_anno2contentlabel`, to generate labeled segments from a genome annotation, `mgene_content_train`, `mgene_content_predict`, and `mgene_content_eval` for training, genome-wide prediction, and performance evaluation of content predictors.

Gene prediction including the tools `mgene_gene_train` and `mgene_gene_predict` for training the integrative layer of mGene and for genome-wide gene predictions, given the signal and content outputs.

For more details on the individual tools are included in the mGene software release available at `http://mgene.org/download`.

E Methods: mGene

In the following we provide details about the SVMs, including the chosen kernels, which were used as signal and content detectors in Layer 1, and about technical concepts used in Layer 2 of mGene.

E.1 Layer 1: Signal Detectors

E.1.1 Selecting positive and negative training examples

For signals, except the transcription start and the cleavage sites, we use consensus strings that have to appear at potential signal positions. These consensus strings are used to find candidate sites for each signal. They include `AG` for acceptor splice sites, `GT` and `GC` for donor splice sites, `ATG` for translation initiation sites, `TAA`, `TAG`, as well as `TGA` for translation termination sites, and `AATAAA`, `AATGAA`, `TATAAA`, `CATAAA`, `ATTAAA`, `AGTAAA`, `TATATA`, `AATAAG`, and `GAAAAA` for poly-adenylation sites. These sites are used to generate positive and negative examples for training.

E.1.2 Kernel Choice and Consensus Motifs

We take advantage of the fact that the sum of several kernels (possibly multiplied by non-negative weights) is again a valid kernel (Schölkopf and Smola, 2002). Thus we can combine several kernels aimed at modeling diverse types of evidence in order to achieve the best accuracy. For the signal detectors, we generally follow a common scheme: we use a position-dependent (WD or WDS) kernel around the site of interest, flanked by two spectrum kernels, one upstream and one downstream. While the central kernel

is supposed to model a core motif, for instance a binding site of the spliceosome, the other two kernels are intended to model local sequence compositional features, like over-representation of exonic splicing enhancers. For poly-adenylation signals, we go beyond this scheme and use additional spectrum kernels.

For each signal, this leaves a number of parameters to be specified. For WD kernel and spectrum kernel, the order needs to be determined, i.e. the length of the considered oligomers; the WDS kernel additionally requires setting the maximal allowed shift. For each kernel, we must also define the exact location (relative to the site of interest) of the sequence window to which the kernel is applied.

In order to set all these parameters, we first fix a broad set of values, supposed to cover the entire reasonable range. Then we use cross-validation (CV) to determine a suitable combination of parameter settings (for more details, see Sonnenburg et al., 2006, 2007b). This can be done separately for each signal detector, because the mutual dependence of the signals has not yet been taken into account at this stage (Layer 1). The values obtained by this procedure are reported in Table S9.

	left kernel		center kernel		right kernel		conservation kernel	
	win	type	win	type	win	type	win	type
TSS	[-400 200]	Sp (d=4)	[-50 70]	WDS (d=14,s=28)	[-100 800]	SP (d=4)	[-50 70]	WD (d=16)
Cleave	[-400 100]	Sp (d=5)	[-60 60]	WDS (d=14,s=28)	[0 800]	SP (d=3)	[-60 60]	WD (d=16)
TIS	[-200 0]	Sp (d=3)	[-30 110]	WDS (d=24,s=0)	[0 180]	SP (d=5)	[-30 110]	WD (d=16)
Stop	[-200 0]	Sp (d=3)	[-110 10]	WDS (d=20,s=0)	[0 180]	SP (d=4)	[-110 10]	WD (d=16)
Acc	—	—	[-60 80]	WDS (d=22)	—	—		
Don	—	—	[-80 60]	WDS (d=22)	—	—		
Trans	—	—	[-190 80]	WDS (d=30)	—	—		
Polya	{[-300 -150], [-149 0]}	SP (d=4)	[-300 306]	WD (d=18)	{[-300 306], [1 150], [151 306]}	SP(d=4)		

Table S9: Features and kernels used for signal detectors. Reported are the sequence windows (win) around the candidate sites on which the kernels are applied, the type of applied kernels (Sp: spectrum kernel, WD:Weighted Degree Kernel, and WDS: Weighted Degree Kernel with shift), and the order d of the kernels. the

E.1.3 Genome-wide Signal Predictions and Post-Processing of SVM Outputs

Genome-wide predictions are computed using five-fold cross-validation: the signals are trained on only three fifths of the nGASP training data and hyper-parameters are tuned on one fifth of the data, thus allowing predictions to be evaluated on the remaining fifth. This procedure is repeated five times, cycling through the five chunks of the partitioned data in order to obtain predictions for the whole genome, ensuring that for each predicted position we used an SVM predictor that was trained only on other positions in the genome (a similar strategy has been used in (Clark et al., 2007)). For the other nematode genomes, we randomly choose one of the three available SVM predictors to perform signal predictions for a candidate site. These predictions are done independently for each signal.

One potential problem with SVMs is the fact that their outputs are, in general, not normalized, i.e. it is unclear in advance which range they will cover, in particular for different hyper-parameter settings (like SVMs' regularization and the kernel parameters). This can cause a problem since we use five different SVMs for a given signal, and the output range of these signals can deviate. In order to achieve a robust scoring system, we post-process the outputs of each SVM: we estimate the posterior probability of observing a true site, thereby obtaining values from the range $[0, 1]$. This normalization is performed by fitting a strictly monotonic function to the distribution of outputs using the validation set following the strategy described in (Sonnenburg et al., 2007b). The so determined transformation functions are applied to all the corresponding genome-wide predictions.

In the following, when SVM outputs f from layer 1 are mentioned, we refer to the thus post-processed outputs.

E.2 Layer 1: Content Detectors

The content detectors are also implemented by SVMs with sequence kernels. In order to model the k-mer compositions of segment types, we use a spectrum kernel of order k. As oligomers of several lengths may be indicative of the segment type, we in fact use a sum of several spectrum kernels of corresponding orders, as detailed in Table S10.

	seq kernel 1		seq kernel 2		cons kernel	
	step	orders	*step*	orders	*step*	orders
intergenic	1	3-6			3	3,6
inter-cistronic	1	3-6			3	3,6
UTR	1	3-6			3	3,6
coding exon	1	3-6	3	3,6	3	3,6
intron	1	3-6			3	3,6

Table S10: Kernels used for content detectors.

For coding sequences we additionally implemented a modified version of the spectrum kernel which is especially suitable for picking up periodic characteristics like coding potential. It takes into account only part of the k-mers that occur in the segment: those whose starting position relative to the segment start is a multiple of a number to be specified, called *step*. Obviously, the standard spectrum kernel is contained as a special case for $step = 1$. For the coding sequence detectors, we use $step = 3$.

In contrast to the signal detectors, the outputs of which are pre-computed in order to save substantial amounts of computation time, it is possible to efficiently evaluate the content detectors "on the fly" while determining the segmentation (in the dynamic program).[3] Thus, while they conceptually belong to Layer 1, we implemented the evaluation of content detectors rather as a part of Layer 2.

E.3 Layer 1: Incorporating Additional Information

As mentioned in the main text, the two-layered architecture of mGene also provides a convenient means of incorporating additional information. Specifically, we have used such information to improve the signal and content detector outputs which serve as features of Layer 2. Corresponding to the categories of the nGASP competition, we consider two types of additional information: conservation (Category 2; based on genome alignments) and transcription evidence (Category 3; based on proteins and EST alignments).

E.3.1 Using Conservation Information

To utilize conservation information, we first pre-process the multiple genome alignment by deriving a conservation score for each position, which is essentially a count of pairwise nucleotide matches. This score is then discretized into four bins of values, which enables the application of the nucleotide sequence kernels to these data. The rightmost columns of Table S9 show the settings of the additional conservation kernels for signal detectors. Specifically, they are applied to the same sequence windows as are the central kernels for each signal, and the degree is determined separately by cross-validation. The conservation information is used in a similar way for content detectors. We use two spectrum kernels with $step = 3$, one of order 3 and one of order 6, as indicated in the rightmost columns of Table S10.

[3] It is in fact difficult to efficiently pre-compute the content detectors, as they depend on the start and the end of the segment, which is only known during computation of the segmentation.

E.3.2 Using EST and Protein Alignments

In category 3 we exploit known protein and EST sequences in the following way: Using blat (Kent, 2002), we aligned the available EST sequences against the genomic DNA. We refined the alignment by correcting typical sequencing errors, for instance by removing minor insertions and deletions. If an intron did not exhibit the splice site consensus `GT/AG` or `GC/AG` at its 5' and 3' ends, then we tried to achieve this by shifting the boundaries by up to 2 base pairs. If this still did not lead to a consensus or the resulting intron was less than $30nt$ long, then we split the sequence into two parts and considered each sub-sequence separately. In the next step, we merged alignments if they did not disagree and shared at least one complete exon or intron. The alignments were used to confirm exons and introns. We repeated the above procedure with known cDNAs and protein sequences from Wormbase. These sequences only contain the coding part of the mRNA (this is also true for Wormbase cDNAs). We therefore additionally use their ends as the annotation for start and stop codons. All introns inferred from alignments having the relatively stringent properties described above were used as evidence for splice sites. At these positions, we transform predicted signal probability f by f^3 in order to enhance signal recognition at the corresponding sites. Regions inside exons and introns were considered free of splice sites. All regions covered by EST, cDNA, or protein sequence alignments were considered as improbable for the appearance of transcription start and cleavage sites (we transformed f by $sqrt[3](f)$). Regions within introns were assumed to be free of translation initiation and termination sites. Note, that by exploiting EST and protein alignments in this way, we can use the ab-initio model, and don't need to retrain.

E.4 Layer 2: State Model for Gene Structure Prediction

So-far we have only generated several unrelated and independently computed lists of predictions for different signals. However, the task of computational gene finding is to compute a complete segmentation of the underlying DNA sequence, that is most likely given the sequence, but also obeys certain biological rules. Examples of such rules are: 'A gene starts with a transcription start site', 'Any donor has to be followed by an acceptor' or 'The length of the open reading frame has to be a multiple of 3'. Selecting the actual signal sites based on a simple thresholding approach on the prediction scores would ignore these rules and lead to biological nonsense. We therefore employ a strategy that is based on a state model to reconcile the individual predictions according to a biological model for gene structures. The employed model is depicted in Figure S4. It allows UTR splicing, the recognition of trans-splicing, and poly-adenylation sites. The states `Don0-Don2` and `Acc0-Acc2` are necessary for carrying the frame information across introns to the next exon. Additionally, the out-of-frame states `Don1`, `Don2` and `Acc1`, `Acc2` are further divided to avoid the creation of an in-frame stop codon through splicing: For example a transition from a frame 1 donor (Don1b) that is preceded by a `T` to a `Acc1a` that is followed by `AA, AG, GA` is prohibited, as this would generate an in-frame stop codon `TAA, TAG, TGA`.

We have also added some mixed states, to accommodate special cases like combinations of acceptor splice sites with translation initiation sites (i.e. `AG|ATG`, where `|` denotes the intron-exon boundary) or of acceptor splice sites with the stop codon (e.g. `AG|TAA`). Finally, we introduced an additional state between the sequence start and the states for the 5' end of the transcript, between the 3' and 5' transcript ends, as well as between the 3' transcript end and the end of the sequence, called *intergenic-long*, in order to permit long intergenic regions ($>$ 20knt). This state, however is not associated with a signal detector.

E.5 Layer 2: Gene Structure Scoring Function

Recall that we use the scoring function $G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$ to determine which segmentation

$$\mathbf{y} = \{(j_i, j_i', S_i, S_i', C_i) \mid i = 1, \ldots, |\mathbf{y}|\}$$

best fits a given genomic sequence $\mathbf{x}$. In this function $G_{\boldsymbol{\theta}}$, features of the sequence are first individually processed by transformation functions $t_{\boldsymbol{\theta}} : \mathbb{R} \to \mathbb{R}$ which are parametrized with $\boldsymbol{\theta}$, and then the transformed values are linearly combined:

$$G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y}) = \sum_{(j,j',S,S',C) \in \mathbf{y}} t_{\boldsymbol{\theta}}^{\text{sig},S}\left(f_j^S\right) + t_{\boldsymbol{\theta}}^{\text{cont},C}\left(f_{[j,j']}^C\right) + t_{\boldsymbol{\theta}}^{\text{len},S,S'}(j' - j + 1) + t_{\boldsymbol{\theta}}^{\text{trans},S,S'} \quad (3)$$

The linearity and a small number of parameters θ are essential features of $G_{\boldsymbol{\theta}}$ that permit efficient training. However, the function $G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$ also needs to be expressive enough to allow separation of good segmentations from bad ones. For instance, it is clearly inappropriate to assume that the quality of a segmentation depends linearly on the lengths of its segments. To resolve this apparent conflict, we utilize piecewise linear functions (PLiFs; Rätsch and Sonnenburg, 2007) for the transformation t.

E.5.1 Transformation of Genomic Features

The applied PLiFs for the transformation t can be formulated as follows:

$$t_{\mathbf{s},\mathbf{v}}(f) = \begin{cases} v_1 & f < s_1 \\ \frac{v_k(s_{k+1} - f) + v_{k+1}(f - s_k)}{s_{k+1} - s_k} & s_k \le f < s_{k+1} \\ v_K & s_K \le f \end{cases} \quad (4)$$

with $K = 20$ supporting points $\mathbf{s} = (s_k)_{k=1,\ldots,K}$ (satisfying $s_k < s_{k+1}$) and values $\mathbf{v} = (v_k)_{k=1,\ldots,K}$. For given supporting points $\mathbf{s}$, application of a PLiF to the segment lengths (and SVM outputs) is equivalent to expanding each single length (output) feature into a sparse vector $\phi_{\mathbf{s}}(f)$ of length K, which is then linearly combined with the values of $\mathbf{v}$:

$$t_{\mathbf{s},\mathbf{v}}(f) = \mathbf{v}^\top \phi_{\mathbf{s}}(f) \ , \quad (5)$$

where $\phi_{\mathbf{s}}(f) \in \mathbb{R}^K$ is a vector with $\frac{s_{k+1} - f}{s_{k+1} - s_k}$ at the k-th component, $\frac{f - s_k}{s_{k+1} - s_k}$ at the $(k+1)$-th component, and 0 elsewhere for $s_k \le f < s_{k+1}$. While t is thus linear – in the values $\mathbf{v}$ as parameters – for given support points $\mathbf{s}$, it effectively models a learnable non-linear transformation of the segment lengths (SVM outputs).

For each genomic feature that we model, a PLiFs is learnt to transform the feature value to a score contribution. For example, the output of a TIS detector SVM is subjected to the PLiF $t_{\boldsymbol{\theta}}^{sig,TIS}$. For all signal and content transformations, we chose 20 support points $\mathbf{s} \in \mathbb{R}^{20}$, equally spaced on the range [0,1] of post-processed SVM outputs. For each length feature, the support points are set to be the 20 quantiles of the empirical distribution of corresponding lengths.

E.5.2 Scoring Function

Now we show how plugging in PLiFs for the transformations t results in a linear scoring function $G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$. First, we construct a comprehensive feature representation $\Phi(\mathbf{x}, \mathbf{y})$ of

$(\mathbf{x}, \mathbf{y})$ by stacking the expanded representations $\phi_{\mathbf{s}}(f)$ of all individual features

$$\Phi(\mathbf{x}, \mathbf{y}) \quad = \quad \left(\begin{array}{c} \left(\sum_{(j,j',S,S',C) \in \mathbf{y}} \mathbb{I}\{S = s\}\, \phi^{\mathrm{sig},s}\left(f_j^s\right) \right)_{s \in \mathcal{S}} \\ \left(\sum_{(j,j',S,S',C) \in \mathbf{y}} \mathbb{I}\{C = c\}\, \phi^{\mathrm{cont},c}\left(f_{[j,j']}^c\right) \right)_{c \in \mathcal{C}} \\ \left(\sum_{(j,j',S,S',C) \in \mathbf{y}} \mathbb{I}\{S = s\}\, \mathbb{I}\left\{S' = s'\right\}\, \phi^{\mathrm{len},s,s'}\left(j' - j + 1\right) \right)_{s,s' \in \mathcal{S}} \\ \left(\sum_{(j,j',S,S',C) \in \mathbf{y}} \mathbb{I}\{S = s\}\, \mathbb{I}\left\{S' = s'\right\} \right)_{s,s' \in \mathcal{S}} \end{array} \right) \tag{6}$$

where $\mathbb{I}\{\cdot\}$ is the indicator function, which yields 1 if its argument is true and 0 otherwise, $\mathcal{S}$ is the set of states, and $\mathcal{C}$ is the set of transitions as described in the previous sections. This feature representation depends on the supporting values of the PLiFs, which, however, are fixed prior to training as described above. Analogous to the situation for a single PLiF, $G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$ can now be expressed as a linear function of this feature representation:

$$G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y}) \quad = \quad \boldsymbol{\theta}^\top \Phi(\mathbf{x}, \mathbf{y}) \ , \tag{7}$$

where $\boldsymbol{\theta}$ is the stacked vector of the values $\mathbf{v}$ of all the PLiFs t^{sig}, t^{cont}, and t^{len}, with the entries of the transition score table t^{trans} appended to it.

E.6 Layer 2: Training

Recall that for training we solve the following optimization problem (eq. (2) in the main text):

$$\begin{array}{ll} \min\limits_{\boldsymbol{\xi} \in \mathbb{R}^N, \boldsymbol{\theta}} & \sum\limits_{n=1}^{N} \xi_n + \boldsymbol{P}(\boldsymbol{\theta}) \\ \text{s.t.} & G_{\boldsymbol{\theta}}(\mathbf{x}^n, \mathbf{y}^n) - G_{\boldsymbol{\theta}}(\mathbf{x}^n, \mathbf{y}) \geq \ell(\mathbf{y}^n, \mathbf{y}) - \xi^n \\ & \xi^n \geq 0 \quad \forall\, n = 1, \ldots, N,\ \mathbf{y} \in \overline{\mathbf{Y}}_n, \end{array} \tag{8}$$

that is, for a set of labeled training examples $\left\{(\mathbf{x}^n, \mathbf{y}^n)\right\}, n = 1, \ldots, N$, the parameters $\boldsymbol{\theta}$ are tuned such that each true labeling $\mathbf{y}^n$ scores higher than all other possible labelings $\mathbf{y} \in \overline{\mathbf{Y}}_n$ by a large margin, i.e. $G_{\boldsymbol{\theta}}(\mathbf{x}^n, \mathbf{y}^n) \gg G_{\boldsymbol{\theta}}(\mathbf{x}^n, \mathbf{y})$.

Here $\boldsymbol{P}$ is a suitable regularizer, the ξ's are slack variables and $\ell(\cdot, \cdot)$ is a loss function measuring the difference between label and prediction.

We will now first describe the regularizer and subsequently the employed loss function in more detail.

E.6.1 Regularizer

For the regularizer $\boldsymbol{P}(\boldsymbol{\theta})$ we use a convex quadratic function of $\boldsymbol{\theta}$, which allows us to efficiently solve the training optimization problem (8). This implies that $\boldsymbol{P}(\boldsymbol{\theta}) = \boldsymbol{\theta}^\top \boldsymbol{Q} \boldsymbol{\theta} + \boldsymbol{c}^\top \boldsymbol{\theta}$ for some positive-semidefinite symmetric matrix $\boldsymbol{Q}$ and some vector $\boldsymbol{c}$. However, instead of numerically providing $\boldsymbol{Q}$ and $\boldsymbol{c}$ it is much more enlightening to describe their structure, which we will do in the following.

The purpose of the regularizer is to stabilize the solutions (i.e. to make them insensitive to minor variations in the data) and to do so by drawing the solution towards a parametrization which respects our prior beliefs. For mGene we implement the beliefs that (i) PLiFs should tend to be smooth functions, i.e. have a rather low amplitude; (ii) PLiFs responsible for signal and content detectors should be monotonic functions; (iii) the average function value of any PLiF should be close to zero; and (iv) transitions should be neither extremely profitable nor extremely costly. It can be shown that regularizing parameters towards zero is mathematically equivalent to asking for a parametrization which maximizes the score gap between the correct gene prediction and the best wrong prediction.

The four prior beliefs about good solutions are implemented by adding terms to the objective to be minimized or by augmenting the set of constraints. Let $\mathbf{v} \in \mathbb{R}^K$ be the values of any PLiF at its K supporting points, and let $\mathbf{t} \in \mathbb{R}^{\mathcal{S}\times\mathcal{S}}$ be the table of transition scores, the prior beliefs can be formulated as:

$$(i) \qquad C_1^{smooth} \sum_{k=1}^{K-1} |v_k - v_{k+1}| + \frac{1}{2} C_2^{smooth} \sum_{k=1}^{K-1} (v_k - v_{k+1})^2 \tag{9}$$

$$(ii) \qquad \forall k = 1, \ldots, K-1: \quad v_k \leq v_{k+1} \quad (\text{or } v_k \geq v_{k+1}, \text{ as appropriate}) \tag{10}$$

$$(iii) \qquad C_1^{small} \sum_{k=1}^{K} |v_k| + \frac{1}{2} C_2^{small} \sum_{k=1}^{K} v_k^2 \tag{11}$$

$$(iv) \qquad C_1^{trans} \sum_{s,s' \in \mathcal{S}} \left|t_{s,s'}\right| + \frac{1}{2} C_2^{trans} \sum_{s,s' \in \mathcal{S}} t_{s,s'}^2. \tag{12}$$

For training mGene, we found the following settings to be suitable: $C_1^{smooth} = 0.010$, $C_2^{smooth} = 0.010$, $C_1^{small} = 0$, $C_2^{small} = 0.001$, $C_1^{trans} = 0.010$, $C_2^{trans} = 0.010$. We further constrain all signal detector PLiFs to be monotonically increasing when applied to the output or their associated signal detector and to be monotonically decreasing otherwise; likewise for content detectors.

As a technical note, we mention that while absolute values are not linear functions, they can be emulated by linear operations when real values are represented by differences of non-negative values.

E.6.2 Loss Function

During training, we measure the distance of our prediction $\mathbf{y}$ to the given label $\mathbf{y}_n$ by $\ell(\mathbf{y}_n, \mathbf{y})$, which is the Hamming loss on the sequence of segment labels. This loss is the weighted sum of the number of mismatching segments (see Figure S5). Here the joint segmentation is considered, which respects all segment boundaries from both compared segmentations, as indicated in the figure. We do not penalize mismatches between: intergenic and intercistronic regions; 3' UTR regions and "poly A region"; 5' UTR regions and "trans-exons".

In the version submitted to nGASP, all mismatched segments (with the above exceptions) had the same weight, and therefore the Hamming loss was essentially the count of mismatched segments. In our latest version, we have increased the penalty for mismatches between intergenic/ intercistronic regions and other regions, as described in Table S11.

We have further introduced a tolerance for the transcript boundaries. More specifically, if segments disagree by no more than 10 nucleotides at a TSS, polyA, or cleavage site, they are not considered to mismatch. This is done to account for the evident uncertainty about the exact location of these signals, be it due to insufficient knowledge or even due to inherent biological variability.

segment1	segment2	loss
intergenic	inter-cistronic	0
intergenic	coding exon	3
intergenic	intron	3
intergenic	5' UTR exon	2
intergenic	transexon	2
intergenic	3' UTR exon	2
intergenic	polya tail	2
utr3exon	polya tail	0
intron	coding exon	3
utr5exon	transexon	0

Table S11: Mismatch penalty for mGene Hamming loss. All other mismatches incur penalty 1.

E.7 nGASP: Experimental Protocol

For the nGASP competition (Coghlan et al., 2008), we first processed the provided annotation of the 10 training regions by inferring missing UTRs and poly-adenylation sites as well as trans-splicing sites in a semi-automatic manner (the resulting annotation can be obtained from the project website). We further corrected minor inconsistencies and excluded genes with non-consensus splice sites or missing reading frames. Finally, we grouped genes with a distance less than 250nt from each other into operons. Based on the augmented annotation we sub-divided the training regions into 747 consecutive sub-regions with at least two genes each, in order to retain complete intergenic regions.

From the sub-regions we generated labeled data for training and model selection of signal and content detectors (Layer 1). For instance, for every annotated splice site we generated a positive example and for every other consensus position a negative example. For signals without consensus we considered all other positions, but sub-sampled in order to speed up training. Using a two-level nested cross-validation scheme, we were able to generate unbiased predictions for the entire training and test set (see Supplemental Section E.1 for details).

For Layer 2 we used the signal predictions of Layer 1 on the sub-regions and the processed annotation to learn to predict whole gene structures for a given DNA sequence. This was done by first converting the annotation into the "true" label sequence $\mathbf{y}$. We then ran the HSM-SVM algorithm on 500 sub-regions to obtain optimized parameters $\boldsymbol{\theta}$. The remaining 247 sub-regions were used for tuning hyper-parameters that are part of the regularizer $\boldsymbol{P}(\boldsymbol{\theta})$. During this phase, we noticed that we could improve the prediction accuracy significantly for the remaining 247 sub-regions by separately tuning a single parameter, namely, the transition scores from the state recognizing the cleave signal ($t_4^{cleave,*}$), which strongly influences the recognition of gene ends. This parameter was tuned to achieve the best performance on the 247 sub-regions. The resulting final model was used to predict genes in the 10 test regions. These predictions were submitted to the organizers of the nGASP competition (referred to as mGene Category 1 version 2 on the nGASP web page and as mGene.init (dev) in this work).[4]

E.8 Methodological Differences to the Developmental Versions

To achieve the reported improvements we have not consciously used any information that was not available during the actual competition. Also, we have not taken advantage of experiences gained by successful competitors after the end of the deadline. Before the

[4]We had three submissions in the ab initio category: (a) an initial submission mGene Category 1 version 1 with suboptimal parameters, (b) mGene Category 1 version 2 with improved parameters, and (c) mGene Category 1 version 3 as in (b) but including predictions of alternative splicing (description goes beyond the scope of this work).

deadline we only managed to train the system a single time and did not have the time for sufficient tuning using the available data. After the deadline we performed steps that lead to the improved performance and which we could have done before the deadline. In fact most of the changes included bug fixes or a cleaner implementation. We made several improvements to the loss function, further optimized the tuning of the recognition of gene boundaries and used *all* 747 regions for Layer 2 training. This led to a second set of predictions, which we called mGene.init/mGene.seq, generated in accordance with the rules of the competition.

E.9 Genome-wide Gene Predictions for Five Nematodes

For the genome-wide predictions on C. elegans and the four additional nematodes, C. briggsae, C. brenneri, C. remanei, and C. japonica, we used the model mGene.init, which was trained on the nGASP training regions. To improve accuracy we used available EST, cDNA, and protein data in the way described above.

Genome-wide predictions for all five nematodes were provided to the Wormbase consortium in the form of GFF3 files and can also be downloaded from the project web site. Starting with the Wormbase WS190 release, mGene predictions will be standard features in future Wormbase genome browsers. Additionally, we provide genome-wide signal predictions, which can be accessed from the project web site and be displayed in genome browsers as custom tracks.

F Methods: Verification of Predictions by RT-PCR

F.1 Experimental Protocol and Evaluation

Genes targeted for verification were selected based on a set of predictions produced earlier that used the genomic sequence of C. elegans version WS170, which was transferred in a subsequent step to genome version WS180. This prediction set will be called mGene170. We used the Wormbase genome annotation version WS180 (GFF3 file downloaded from `ftp://ftp.wormbase.org/pub/wormbase/genomes/c_elegans/genome_feature_tables/GFF3/elegansWS180.gff3`) and parsed all entries with the source (second column) specified as `Coding_transcript` and the type (third column) specified as `gene`, `mRNA`, `exon`, `CDS`, `intron`, `five_prime_UTR`, or `three_prime_UTR`. We call the reconstructed gene set Anno180. Additionally, we have parsed all pseudo-genes (source: `Pseudogene` or `history`; type: `pseudogene`), and all non-coding RNAs (source: `miRNA`, `ncRNA`, `rRNA`, `scRNA`, `snRNA`, `snlRNA`, `snoRNA` or `Non_coding_transcript`; type: `ncRNA`) from the WS180 annotation. These sets will be denoted as pseudo180 and ncRNA180. To assess the quality of our predictions, we have examined three different sets of predicted genes:

mGene170-pseudo — mGene170 genes that had no overlap with any segment from Anno180, yet overlapped with segments from pseudo180.

mGene170-nc — mGene170 genes that had no overlap with any segment from Anno180, yet overlapped with segments from ncRNA180.

mGene170-novel — mGene170 genes that had no overlap with any segment from Anno180, pseudo180, or ncRNA180.

Additionally, we have examined a fourth group of unconfirmed annotated genes that had no overlap with the predictions:

missed-unconf180 — Anno180 genes that were unconfirmed and did not overlap with any gene from mGene170.

From each of these four groups we selected genes with at least two coding exons and designed primer pairs in the first and last coding exon using primer3 (cf. (Rozen and Skaletsky, 2000) and `http://primer3.sourceforge.net`) with default parameters. We checked that the primer sequences had only a single match in the C. elegans genome and chose a subset at random if the number of resulting genes exceeded 50. A summary of the experiments is given in Table S12, and the primers are provided in Tables S13, S14, S15, and S16.

	total number	number tested	sequence results	hit to target	unambiguously	correct structure
mGene170-pseudo	425	46	35	29	25	25
mGene170-nc	17	13	6	6	5	5
mGene170-novel	635	52	35	31	31	26
missed-unconf180	347	46	14	13	11	7
DNA control	-	20	20	18	17	-

Table S12: Summary of validation results by RT-PCR and sequencing of mGene170 predictions in the three different categories mGene170-pseudo (overlap with pseudo-genes), mGene170-nc (overlap with non-coding RNAs), and mGene170-novel (new predictions), as well as unconfirmed genes in the WS180 annotation which mGene did not predict.

We isolated mRNA from a mixed population of wild type C. elegans (TRIzol, Invitrogen; PolyATtract mRNA Isolation System III, Promega), which was then reverse-transcribed using SuperScript II & III reverse transcriptase (Invitrogen) with random primers (Invitrogen). Subsequently, touchdown PCR was performed using TAQ DNA-Polymerase S (Genaxxon). We screened for sufficient PCR amplification at final annealing temperatures of 55 and 52°C. The PCR products were then analyzed by agarose gel electrophoresis. DNA fragments were excised from the gel and purified using Wizard SV Gel and PCR Clean-Up System (Promega). The purified DNA fragments were sequenced from both ends by Big Dye Terminator sequencing (Applied Biosystems, Foster City, CA). Obtained sequences were aligned against the genome using blat (Kent, 2002). Subsequently, we checked whether the alignment had any overlap ($>$ 1nt) with the corresponding targeted gene. If so, and if, in addition, the alignment hit was unique or the score of the hit to the targeted gene was highest and the difference to the second best hit was sufficiently large ($>$ 5%), then we counted the prediction as correct. Furthermore, we counted the number of genes that had at least one splice site confirmed by the aligned sequence. All results are summarized in Table S12. Additionally, the detailed results for each experiment can be found in Tables S17, S18, S19, and S20. The sequences are provided as additional FASTA files, and the alignments as PSL files.

For the sake of completeness, we also compared the sequencing results with the new set of mGene predictions on the WS180 genome, denoted as mGene.seq in the main text. However, not all previously analyzed cases could be assigned to a corresponding prediction: For the results shown in Table 3 we only considered the cases where both primer pairs could be found in correct orientation in a predicted transcript in mGene.seq's predictions. The number of cases usable for the evaluation of the mGene.seq predictions and missing WS180 annotations is, therefore, considerably reduced: 57 and 24 instead of 111 and 46 cases, respectively (see Tables 3 and S12. (While this may look like a big change in the prediction sets, please note that the targeted genes are among the ones hardest to detect: they are also missing in the annotation, possibly due to low expression or weak signals. The genome-wide predictions mGene170 and mGene.seq, however, are very similar for a large proportion of the genes.) The success rates of the validation experiments for the reduced sets of genes (mGene.seq) agree well with the success rates determined with the earlier predictions (mGene170).

F.2 Tables

Table S13: Primer List for targeted genes from group mGene170-pseudo.

	Name	left primer	right primer	product length
1	mgAA_1	TCCAGATTCAGTTCTTCAACCA	TTATTTGGGGCATTTGAAACA	471
2	2gAB_1	GTGCCACTCTTTTCAAACCAC	AAAAATGGAGCAAGAAGCCAT	235
3	mgAC_1	AAATTTTTCGGAGGCATGAAG	ATCCATGTCATTATGAACCGC	351
4	mgAD_1	GGCCAGCATGCTATATCTTCA	CAGAAATACCACAATGGTCGC	203
5	mgAE_1	CTGATCGAGTCAACTGAAGCC	TTTAATCTTCAGAGCTCTCCGC	235
6	mgAG_1	TGGAGTGTGGTCCAACTATGG	GCAGTTTCAAATGCGGTTTTA	433
7	mgAH_1	GGATTCGCAGCTTCAAAAATA	TATGGGTCAGCACAAACATGA	837
8	mgAI_1	TGGTTTGATCTCACTGGTGCT	AAGTGTCTGGTGTCTTCGCAT	300
9	mgAJ_1	AGTGTTCACGTTGGGATCCAT	CAAGTGTACGTTTTGAAAATTCG	260
10	2gAK_1	GGATACTTGCTTCTTCCCGTT	TCTCCGTAAAACTTCTCGCAA	234
11	2gAL_1	CCCGAAGAAAGTGCTCAGAAG	AAAAATGAACGCGAAACCC	499
12	2gAM_1	GCAAGCGATTGTTGTTGACTT	CTTGTTGAAGAACCCATGAGC	212
13	mgAN_1	TGTAGAAAAGTGGGGAAAGACG	TCATATGCATCGTCGAACAAA	836
14	mgAO_1	GGTGCTCTTCCATCCGATT	TGACGAGCTGATCTTGGATTT	391
15	mgAP_1	GGGAACTTGCGTTGGTCTT	AGCGGCGGTAAGTAGGTAGTC	366
16	mgAQ_1	GAATTGCGATTGCTGACTGTA	TTCAAGACGAATGGACAAAGAA	835
17	2gAS_1	TGCGATTTGTGATTTTTCTCA	TCAGATTTCCATCCAATAGCG	657
18	mgAT_1	GCTGGTTGCTCAGTGGCTA	GTGGTCGGGACTGGACTG	397
19	mgAU_1	AAATCCAGAAATCTGAATAGACGA	GATTTGCTACAATTTTCAGCCC	235
20	2gAV_1	GCACAAATTCATTTCTGATATTCC	TTGTGATTCCATGTGTTCGAG	451
21	mgAW_1	TCCACATCACTGTTCATCCAA	GAGGGACTTCATGATCTTCACC	826
22	mgAX_1	CTCAACTAACAATGGGCCAGA	AATTTATTGAAACGGACGCAA	178
23	mgAY_1	TGGGAACAAGAGTGGACAGTT	TGTGAAGTGAGCATCATCGAG	236
24	mgAZ_1	ATGCTCGAGGATTGGATTTTT	TTGGAATACACAACACGATGG	818
25	mgBA_1	ATCGTCGTCTCTGTGGACACT	CTAGTGACGGTCCAAAGCAAG	246
26	mgBB_1	TTGCGGGGTTAACTTTTCAAT	GATGCTCCGAAGCAATATTTTC	489
27	2gBC_1	CTAAAAAGGAATTCGTGCTGG	AATGAAGAGCGACGGATTTTT	793
28	mgBD_1	CGCTGTGCTCGTAACATCA	GGAATTTAGGAAGAACATAATTGG	497
29	2gBE_1	TGTGTTGTGTATTGAGCGAGC	CCCACAAGTTTTGTCGCAGTA	297
30	mgBF_1	AAATCATGAACGAAATCCGTG	ACGACGAAGATTGTCCTCTCC	548
31	mgBG_1	GCCCACTTTCACGAATGTATC	CTGAAAAGTTGTCCGATGGAG	669
32	2gBI_1	ATGCCTGGCTGCCAAATA	CCACTTTGAAATTGTCATCGAA	397
33	2gBJ_1	GATGTCAAATGAGCGGATTGT	GTTGTTGAGGAAACGGTTTGA	232
34	2gBL_1	GGTGTTGCAGGTGAAATTGTT	AAGCGATAATAGCAAGTGGGG	173
35	2gBM_1	TGTCAAACCAAGAAAGTATTGAAC	TTCTTCTGGCCATTTCCAA	605
36	2gBN_1	TCAAGGATCTGGAATGTTGTTTT	AATTGTTTGCTTAGGGACCGT	541
37	mgBO_1	ATGTTCAGGATGGCTTCATCT	GCCGTTCTAGTGGCAGGTA	846
38	mgBP_1	AAAGATGATCAACGCTCTTCG	CTTCTTGATCAAATGCCGAAA	227
39	mgBQ_1	CGTACTCGCCATGGGCT	AACACCATTGTGACGGAGC	694
40	mgBR_1	CACCGATGGAGCAGCAAT	CAACCACAACGTCCATTGATA	248
41	2gBS_1	GAAGCCACATGTGCTCAACTT	GCTGACATCTCTCAGCAAGTACA	353
42	mgBT_1	ATACGTTGGTTGCATTTGGTC	TGGTTTCGATAAGTCCTGTGAA	483
43	mgBU_1	GGCTATGGCGGGCTATG	ATGAATTGCCAAGGCAAAAA	499
44	mgBV_1	GTGGAGCTCATGGAAGAGGTT	CTTTTGATCATGCTACCGAGC	373
45	2gBW_1	TCCTGTACATTGTGGAGTTGGA	CTGAAACTTTTTGGTTCTGCG	236
46	mgBX_1	ATCCGCAAGCATCTTGTTAGA	TTTTGCCATGATAGAACGGAG	228

Table S14: Primer List for targeted genes from group mGene170-nc

	Name	left primer	right primer	product length
1	mgAA_2	GCAACCGCTTTATTTCTCTTG	ACAATTGTGGAGAAGACGGTG	250
2	mgAB_2	GTTCACGGATTCCCGTCTC	ATCGCAGATTGACAGCAAAAT	152
3	mgAC_2	GGGCATTTTCAAGTTCACAGT	GCAATATTCCAAGATTTCGTTTG	831
4	mgAD_2	ACGGTAGTGTTGACGCAATTC	GAAGCATCCAATAGGAGGGAG	798
5	mgAE_2	TCAAAATTTGGCAACTATGGG	TCAAAGGAGCTCGAATTGAAA	578
6	mgAF_2	TGTCCAGCAATCAGAAAATTG	CAGCCATTTCTTTGTGCATTT	229
7	mgAG_2	TCCTTCCTCACTCTTCACACAA	GGTGTTTGTAGAAACTCGGCA	489
8	2gAH_2	CACCAATTTTGCGACTATGCT	ACGTCAGGTGCAATGAAAAAC	495
9	mgAI_2	TGCATGCATTCAACATTCATT	AGATATCTCATCACAACGTCGAA	646

cont'd on next page

	Name	left primer	right primer	product length
10	2gAJ_2	CCGCAATAGTTCGTCGGT	CTCATGATGAAGCTGGGGTT	196
11	mgAK_2	GGCGCCTAATCTCGTGTAAAG	TCCCACGTGAAACAGAAAGTT	479
12	2gAL_2	TGTCCAACTCAACCATCACAA	TTCCATTTCCTTAGGCGTTTT	271
13	mgAM_2	TGGGCATTAGAAAGTGGCA	GGTGGCAGTTGTGAGACCA	500

Table S15: Primer List for targeted genes from group mGene170-novel.

	Name	left primer	right primer	product length
1	mgAA_3	GCTCTCTTTCCTGCTCCTAGC	GGAGGAATCATCAATAGCTGC	184
2	mgAB_3	TCCTCTGTATTCGGACCCC	ATAATTTCTGTGATGCGCGAC	838
3	mgAC_3	TTCGGAATATCGGTCACACAT	TTAGATGTCATCGCAGTGGTT	191
4	mgAD_3	ACTGCTGTTATTGTTGGAGCA	TCACTTTTCGTTTCGGGAGTT	250
5	mgAE_3	CTGTTCTTCTTTTTCACCGCC	AGTGCTCTCAGTTGCACCAGT	204
6	mgAF_3	CAAGTACAATGGATCCTGAGCA	GAAGCACCAAGATATGGAACA	295
7	mgAG_3	GGGGAACCTCCACCAATAAT	TGCAATACACTGCTCCTTTGA	218
8	mgAH_3	TGGACACATCAGTCAACATGAA	TCCTGTGAGCTTCTCCAGTGT	198
9	mgAI_3	AAAATGGGCCAGTGGTTGT	AGCACATAATCCAGCGTTATCA	150
10	mgAJ_3	GGACTTCACAGGACCTCCG	CAGCAGAATACCAGATTTTCCA	128
11	2gAK_3	ATGGTGCCAACTCATTTCCTC	CCTTCCATTTGGTCACTTCAA	184
12	2gAL_3	GGAGTTAGCATCACAATTGGC	ATCGGATGAGCTGAAAGGATT	198
13	2gAM_3	CCATCGATCCGTAGAAATCCT	TAATTCCTTCGTTTGTCGTGG	182
14	mgAN_3	TGAAGAGAGAAGCCAAGAGCA	TCCATCACTCTTTCGTTCGAT	534
15	mgAO_3	ACTCCAGTGTCTTTGCCAGAA	GATTTCCACGACGTTTGATTG	232
16	2gAP_3	TGAGTGAGAAGGAGTTTTTCAATG	CTGGTGGCACTACTGGATCAT	187
17	mgAQ_3	CAAGCTTGCCTTGGTCTTTG	TCTCGCTGAAACTGACTCCAT	212
18	mgAR_3	CTTGGTCAAAGCCAACAGTCT	GAACTGAGCTAGTAACATGGCG	245
19	mgAS_3	TGGTGGTTATGATGAGCAAAG	GCTCGAATAGAGACGTTGTCG	231
20	mgAT_3	TGGGATCTTCTAGTGAAACACG	AGGCAGAAGAGGAGAATCAGG	248
21	mgAU_3	GGGGTGCTTATAGTTATGCGA	CCAATTGGAAAAGGACACAGA	395
22	mgAV_3	CTGACAACTCGGTGGAAAAAG	TCGGAATCAAGTAATCATGGC	369
23	mgAW_3	CGCCCCTCTTATTTAGTTTCG	TTATCGGCATGTTCCCTTGTA	220
24	mgAX_3	CAGGCGGGAAACATAGAAAAT	TCTTGAAGTTTGGGGATTTCA	248
25	mgAY_3	AAACTCGGCATTTTTCATTCC	TTTGGGCATTTAATTTGAGCA	498
26	mgAZ_3	TGAAACTGCTCGTTTTGCTAT	TGATTTCAAGTTGTGAAGCGA	693
27	mgBA_3	TGTGACATTTCAACTGGACGA	GATCGATGTAGGTGGCACG	113
28	mgBB_3	AGCCACTGGTATTTGGTGTTG	ACAATAACACTCGATGCTCCG	233
29	mgBC_3	TGATTGAACTGAACCAGGAGG	TGTATAATCAGTTGGCGGAGC	464
30	mgBD_3	GCCACTGATAAAGCATGTGTG	TAAGCGTTTGGTAATTGCAGC	245
31	mgBE_3	CGGAGTTTTTGTTCTAATCTGC	AACGTTACCAATTCCTCAAAGC	794
32	mgBF_3	GTTGTGGAGCGGTATTGGAG	CAACCCTTGGTTGCACTTTG	200
33	mgBG_3	TGGTCAGAAATCCAAAAACAA	TCTGGTAGAATTGATCGGACG	386
34	2gBH_3	ACGCGATCTCAGTGAGGATTA	ATCTTTCAATTGCCTTCGGTG	498
35	mgBI_3	GGCCAACAACTCAAAAACAAA	CTCAATCCTTCATTTGTGCAAT	233
36	mgBJ_3	TGGGCAGTTGCTACAGATTTT	AATTTCCGGCTTCTGAACTGT	243
37	2gBK_3	TGGAAAGCACGAGTCTGAAGT	GAGGGTCCTCTCACGAAGTTT	244
38	2gBL_3	ATGTCTGGCGAAGATGGTTG	TTGCATTTGGCACATTTATTG	174
39	mgBM_3	CTCGAGATTGTCGATGAGGAC	GGGAAGGGTGGATCATTAGAA	492
40	2gBO_3	ATGGTGAAGGAGGTGGATTCT	TTCGAATATGACGAGGTTTGC	248
41	mgBP_3	GATTGCTTAATTGCAGCCTCA	GAGTTGGAAGCTAAGCGGTCT	207
42	mgBQ_3	CCTCATCCTCGATTATCGTCA	ACACAGATCCCATCAAGAACG	175
43	2gBR_3	CAGCCGTTGAGAAGACAGTTC	CTTTTCAGTCGATGGATACGG	174
44	mgBS_3	GAATTGGAAACCTAACTACGCC	ATCTCCAATCGTGGCTTTTCT	214
45	mgBT_3	GAGTTGCCAAAGTGTGAGGAC	TCATTCTTCTGGATGACTGCC	187
46	mgBU_3	TCTACTGATTGGAATGGGCAC	TCTTAGTTCTCCTCTGCCCAAC	238
47	mgBV_3	AAATGGATTCTTCACTGATTGTCA	TTTCACATCGTGCCATACAAG	447
48	mgBW_3	TGGCTGCAAAGAGAAGAAAAA	TCCAATTGAACAAATCCAGTG	337
49	mgBX_3	GACATCGAAAAAGGCCCAG	CAATGTTCGGGGATCCTG	250
50	mgBY_3	TCAAGTTCAAAGTCTGCCGAT	GGGATCCACTCGGTACCTAAA	205
51	mgBZ_3	ATTCCAGGAGAGACGGATGAT	ATTCCTGGGATTGCCTTTCT	161
52	mgCA_3	CTATGGATTTACGTTGGAGCG	CACATTCCACATGAAGAAGCA	231

Table S16: Primer List for targeted genes from group missed-unconf180.

	Name	left primer	right primer	product length
1	2gAA_4	CGGTACGTTGGTTGTGGG	TTTATATCGGGGATTGAATCAG	398
2	2gAB_4	GGTGTCTTCTTTTGTTGACCG	GCTCCTCCTTTCTCACTTTTCA	227
3	2gAC_4	CTGCTCATTGCTGGAGTTTTG	GTTTCGTTTTCCGAAGGATTC	245
4	2gAD_4	GGTTCTCAAGACAGTTTGCGT	GTTCACAAACACGATGAGCAA	390
5	mgAE_4	CATTCTTTGTGCTCTTGGCTC	ATGATTTCCTTATCGCCCAGT	173
6	2gAF_4	GCACCGTCCAGGGTACAT	TCTTGGGAATGTGTTATTGGC	241
7	2gAG_4	CGTTTTGCGCTGATATTACAC	GCTGGTCGAAGAGAGAAGAATG	841
8	mgAI_4	TGTGGATTCAACAAGTTACGAA	TTCTTCAAGCTCAAATTCCCC	850
9	2gAJ_4	GGATCCAGTCCAAATCGAGA	ACTGTCAGATCCAGAGCCAGA	206
10	2gAL_4	TGAACTCAGGTCAGAGGGATG	TCCAAATCCATTTCTCGATTG	496
11	mgAM_4	TGTTGAAATCGAATTTCGGG	TACATGTTTGCAAGGTCACCA	277
12	2gAN_4	TGATCCACATTGGAGTTCACA	GTTTTATGTGTTTTTGTGCCG	399
13	2gAO_4	TCGTAGCTCCAAAAGTTCTCG	TGAATGCTTCCATTGCAGA	248
14	mgAP_4	GAACACTTTTCATCCAGCCAG	GCACCTTTTCTGGGTTTCTTT	298
15	mgAQ_4	GTGGCGTGAAAGAGCAAAA	CATCGTTTCTTGTACGCGTTT	236
16	mgAR_4	CTCCGTAGCATCGTCCTTCTT	ACACTGGAAGCAGAGAGCAAA	236
17	2gAS_4	CATGTCTATTTGAGCTGGCCT	TGACTGTGTTGGTAGGGTTCC	571
18	mgAT_4	CCAAAATGAACTGAGGATATGGA	GTGGACATCACGGCTTGAC	500
19	mgAU_4	GAGAAAGGAGAATTTGTGCGG	GCCAAGTTGCTCTGTTTGAAG	211
20	2gAV_4	GTTTTATGCACCGTTGTCGTT	CTTCGACCATCAGACCAAGAG	230
21	2gAW_4	ATGTATTCTGAAAGCGTCCCA	CGCAACGCACCTTAAATCTAC	456
22	mgAY_4	GGATGAGGGGCTTACAAAAA	TCATCATTCTCCCCATCTCTG	433
23	2gAZ_4	TGGGGTTACATGAGTGGGATA	TCAATGTAATTTGTTGAAAAGCA	298
24	2gBA_4	GACACCGAGAACAACTGAAGG	GAAACTGATGAACCATTTCCG	270
25	2gBB_4	CGATTCAAAGGGAAAGTTCGT	GGTGACGGCTGTGGTTGT	179
26	2gBC_4	GAAAACCTACTCATTTTTCGTGC	TTTATTCGGGACAGCAAATGT	231
27	2gBD_4	GCGTTGCTTTTTCTTTTCTGTT	TATTCGGGAATTGTTTGAGCA	695
28	mgBG_4	TTTTTATGGTGGAGACGCC	ACTGCTCTTGCCAAATACGTG	160
29	mgBH_4	GCCGTTTTCCGTCCTTGT	AAACGAAATGGCGTGATTTTT	399
30	2gBJ_4	ATGAGCATCTGGAGAATTGC	GAGGGAGATGCAACAACAAGT	163
31	2gBK_4	TTGTTTGCGCTGACAACTCTA	GGAACCGTTTGACCAGAGATA	661
32	mgBL_4	CTTTTGGTGCTGGTCATCCT	GGGAAAATACACGATGTGAACC	291
33	2gBM_4	GAGCTCGAAGGCTGTTCAGAT	CCGGCAATTATTCTCAAAACA	189
34	2gBN_4	CAAACGTTAAAGTGGGTTTCAA	AAATATCCTCGGTTTCTACCTCA	972
35	mgBO_4	AAGATTGGACCTGCAAGGAAT	AATTTCCACAGACAAGGACCA	366
36	2gBP_4	GATTAAGAACCAAAGCAGCCC	TTCGGTGCGAAAATAAACATT	159
37	2gBQ_4	GCGGAGAATGGAAAAAGACA	TTTTTCCTCAAACCCCGAAT	129
38	mgBR_4	TGGCAAAATATTCCGATTCTG	TTGAAAATATATTCCAGCAGGATG	825
39	mgBS_4	GTTCTGCCTTCTCGAGGATTT	TGAACATGAAGCCTTGAATTTG	480
40	2gBT_4	GGATAGAGGTGGAGGGTCGAT	CACTGTGCTTAGTTTGATGGC	191
41	2gBU_4	GACGAAATTCCATGCTCAAAA	TGTCAACTTGGTTGCTTTTACG	651
42	2gBV_4	TCAATGCGAGCACTGACTATC	GGTCCGTCGATATTGTGAAGA	843
43	mgBW_4	AGCGACATCACTCTGGGATT	CATCCCAATCCTAGTGCTTCC	204
44	2gBY_4	AAACGCCAAAATTCTGGTATAAA	TTTAATGAGCACCGCTGATG	234
45	2gBZ_4	CCCACTGGATATTGGAAACCT	AACTCCTCAATTGCTGAACCA	240
46	2gCA_4	CAGGAGCATGTCCAAAGAAAG	AACTTGGGTATTGAGATGATGG	234

Table S17: Detailed results for group mGene170-pseudo

Name	Sequence name	num blat hits	best score	second best score	hit to target	unambiguous	structure agreement
mgAA_1	mgAA_1_ba2_PL_rep0	1	374	0	1	1	1
	mgAA_1_ba2_PR_rep0	1	424	0	1	1	1
mgAB_1					0	0	0
mgAC_1	mgAC_1_ba1_PL_rep0	3	274	196	1	1	1
	mgAC_1_ba1_PR_rep0	3	297	259	1	1	1
	mgAC_1_ba2_PL_rep0	3	307	269	1	1	1
	mgAC_1_ba2_PR_rep0	3	312	276	1	1	1
mgAD_1	mgAD_1_ba1_PL_rep0	8	159	146	1	1	1
	mgAD_1_ba1_PR_rep0	9	142	128	1	1	0
mgAE_1	mgAE_1_ba1_PL_rep0	1	336	0	0	0	0

cont'd on next page

Name	Sequence name	num blat hits	best score	second best score	hit to target	unambiguous	structure agreement
	mgAE_1_ba1_PR_rep0	1	275	0	0	0	0
mgAG_1	mgAG_1_ba1_PL_rep0	1	758	0	1	1	1
	mgAG_1_ba1_PR_rep0	1	758	0	1	1	1
mgAH_1	mgAH_1_ba1_PL_rep0	2	512	420	1	1	1
	mgAH_1_ba1_PR_rep0	2	528	370	1	1	1
mgAI_1	mgAI_1_ba1_PL_rep0	6	232	228	0	0	0
	mgAI_1_ba1_PL_rep1	1	104	0	0	0	0
	mgAI_1_ba1_PR_rep0	1	118	0	0	0	0
	mgAI_1_ba1_PR_rep1	1	82	0	0	0	0
mgAJ_1					0	0	0
mgAK_1	mgAK_1_ba1_PL_rep0	1	384	0	0	0	0
	mgAK_1_ba1_PR_rep0	1	379	0	0	0	0
mgAL_1	2gAL_1_ba1_PL_rep0	6	492	290	1	1	1
	2gAL_1_ba1_PR_rep0	10	505	378	1	1	1
	mgAL_1_ba1_PL_rep0	3	363	90	1	1	0
	mgAL_1_ba1_PR_rep0	3	349	84	1	1	1
mgAM_1					0	0	0
mgAN_1	mgAN_1_ba1_PL_rep0	2	624	36	1	1	1
	mgAN_1_ba1_PR_rep0	2	751	37	1	1	1
mgAO_1	mgAO_1_ba1_PL_rep0	16	271	249	1	1	1
	mgAO_1_ba1_PR_rep0	12	72	53	1	0	0
	mgAO_1_ba2_PL_rep0	17	338	182	1	0	0
	mgAO_1_ba2_PR_rep0	17	338	169	1	0	0
	mgAO_1_ba3_PL_rep0	14	276	253	1	0	0
	mgAO_1_ba3_PR_rep0	17	280	262	1	0	0
mgAP_1	mgAP_1_ba1_PL_rep0	1	524	0	0	0	0
	mgAP_1_ba1_PL_rep1	1	300	0	0	0	0
	mgAP_1_ba1_PR_rep0	1	597	0	0	0	0
	mgAP_1_ba1_PR_rep1	1	559	0	0	0	0
mgAQ_1					0	0	0
mgAS_1					0	0	0
mgAT_1	mgAT_1_ba1_PL_rep0	2	353	353	1	0	0
	mgAT_1_ba1_PR_rep0	2	351	351	1	0	0
mgAU_1					0	0	0
mgAV_1					0	0	0
mgAW_1					0	0	0
mgAX_1	mgAX_1_ba1_PL_rep0	1	374	0	1	1	1
	mgAX_1_ba1_PR_rep0	1	384	0	1	1	1
mgAY_1	mgAY_1_ba1_PL_rep0	1	186	0	1	1	1
	mgAY_1_ba1_PR_rep0	1	197	0	1	1	1
mgAZ_1					0	0	0
mgBA_1	mgBA_1_ba1_PL_rep0	2	179	167	1	0	0
	mgBA_1_ba2_PL_rep0	2	201	189	1	0	0
	mgBA_1_ba2_PR_rep0	2	205	187	1	0	0
mgBB_1					0	0	0
mgBC_1	2gBC_1_ba1_PL_rep0	12	661	196	1	1	1
	2gBC_1_ba1_PR_rep0	14	493	225	1	1	1
	mgBC_1_ba1_PL_rep0	14	850	144	1	1	1
	mgBC_1_ba1_PR_rep0	3	545	37	1	1	1
mgBD_1	mgBD_1_ba1_PR_rep0	190	443	439	1	0	0
	mgBD_1_ba1_PR_rep1	190	259	228	1	0	0
mgBE_1	2gBE_1_ba1_PL_rep0	34	257	139	1	1	1
	2gBE_1_ba1_PR_rep0	6	218	187	1	1	1
	mgBE_1_ba1_PL_rep0	5	57	57	1	0	0
	mgBE_1_ba1_PL_rep1	8	324	222	1	1	0
	mgBE_1_ba1_PR_rep0	4	64	63	1	0	0
mgBF_1	mgBF_1_ba1_PL_rep0	5	429	338	1	1	1
	mgBF_1_ba1_PR_rep0	6	504	373	1	1	1
mgBG_1	mgBG_1_ba1_PL_rep0	11	681	145	1	1	1
	mgBG_1_ba1_PR_rep0	9	688	127	1	1	1
mgBI_1	2gBI_1_ba1_PL_rep0	2	357	244	1	1	1
	2gBI_1_ba1_PR_rep0	2	360	192	1	1	1
mgBJ_1	mgBJ_1_ba1_PL_rep0	1	158	0	1	1	0
	mgBJ_1_ba1_PR_rep0	1	165	0	1	1	1

cont'd on next page

Name	Sequence name	num blat hits	best score	second best score	hit to target	unambiguous	structure agreement
mgBL_1	2gBL_1_ba1_PL_rep0	3	124	106	1	1	1
	2gBL_1_ba1_PR_rep0	3	130	73	1	1	1
	mgBL_1_ba1_PL_rep0	3	194	121	1	1	0
	mgBL_1_ba1_PL_rep1	3	207	134	1	1	0
	mgBL_1_ba1_PR_rep0	3	178	75	1	1	0
	mgBL_1_ba1_PR_rep1	3	182	68	1	1	0
mgBM_1	2gBM_1_ba1_PL_rep0	3	458	33	1	1	1
	2gBM_1_ba1_PR_rep0	1	173	0	1	1	0
	2gBM_1_ba2_PL_rep0	5	554	117	1	1	1
	2gBM_1_ba2_PR_rep0	4	549	32	1	1	1
mgBN_1	mgBN_1_ba1_PL_rep0	1	226	0	0	0	0
	mgBN_1_ba1_PL_rep1	1	221	0	0	0	0
	mgBN_1_ba1_PR_rep0	1	232	0	0	0	0
	mgBN_1_ba1_PR_rep1	1	228	0	0	0	0
mgBO_1	mgBO_1_ba1_PL_rep0	1	623	0	1	1	1
	mgBO_1_ba1_PR_rep0	1	447	0	1	1	1
mgBP_1	mgBP_1_ba1_PL_rep0	4	205	179	1	1	1
	mgBP_1_ba1_PR_rep0	3	212	163	1	1	0
mgBQ_1	mgBQ_1_ba1_PR_rep0	5	121	121	1	0	0
	mgBQ_1_ba1_PR_rep1	1	16	0	0	0	0
mgBR_1	mgBR_1_ba1_PL_rep0	3	192	84	1	1	1
	mgBR_1_ba1_PR_rep0	3	207	81	1	1	1
	mgBR_1_ba2_PL_rep0	3	197	84	1	1	1
	mgBR_1_ba2_PR_rep0	3	212	80	1	1	1
mgBS_1					0	0	0
mgBT_1	mgBT_1_ba2_PL_rep0	1	243	0	0	0	0
	mgBT_1_ba2_PL_rep1	1	225	0	0	0	0
	mgBT_1_ba2_PR_rep0	1	266	0	0	0	0
	mgBT_1_ba2_PR_rep1	1	264	0	0	0	0
mgBU_1	mgBU_1_ba1_PR_rep0	1	215	0	1	1	1
	mgBU_1_ba2_PL_rep0	2	461	34	1	1	1
	mgBU_1_ba2_PR_rep0	2	301	34	1	1	1
mgBV_1	mgBV_1_ba1_PL_rep0	10	282	257	1	1	1
	mgBV_1_ba1_PL_rep1	6	281	247	1	1	1
	mgBV_1_ba1_PR_rep0	9	213	211	1	0	0
	mgBV_1_ba1_PR_rep1	4	89	85	1	0	0
	mgBV_1_ba2_PL_rep0	8	206	202	1	0	0
	mgBV_1_ba2_PL_rep1	4	228	224	1	0	0
	mgBV_1_ba2_PR_rep0	9	326	326	1	0	0
	mgBV_1_ba2_PR_rep1	4	88	82	1	0	0
	mgBV_1_ba3_PL_rep0	9	285	281	1	0	0
	mgBV_1_ba3_PL_rep1	4	280	276	1	0	0
	mgBV_1_ba3_PR_rep0	13	378	374	1	0	0
	mgBV_1_ba3_PR_rep1	7	368	368	1	0	0
	mgBV_1_ba4_PL_rep0	5	375	371	1	0	0
	mgBV_1_ba4_PL_rep1	5	390	386	1	0	0
	mgBV_1_ba4_PR_rep0	6	380	376	1	0	0
	mgBV_1_ba4_PR_rep1	7	379	375	1	0	0
mgBW_1	2gBW_1_ba1_PL_rep0	1	189	0	1	1	1
	2gBW_1_ba1_PR_rep0	1	192	0	1	1	1
mgBX_1	mgBX_1_ba1_PL_rep0	4	184	78	1	1	1
	mgBX_1_ba1_PR_rep0	4	186	110	1	1	1

Table S18: Detailed results for group mGene170-nc.

Name	Sequence name	num blat hits	best score	second best score	hit to target	unambiguous	structure agreement
mgAA_2	mgAA_2_ba2_PL_rep0	3	102	68	1	1	0
	mgAA_2_ba2_PR_rep0	3	119	101	1	1	1
mgAB_2	mgAB_2_ba1_PL_rep0	4	104	82	1	1	0
	mgAB_2_ba1_PL_rep1	4	107	93	1	1	0
	mgAB_2_ba1_PR_rep1	4	112	84	1	1	1
mgAC_2					0	0	0
mgAD_2					0	0	0
mgAE_2					0	0	0
mgAF_2	mgAF_2_ba1_PL_rep0	1	156	0	1	1	1
	mgAF_2_ba1_PL_rep1	1	158	0	1	1	1
	mgAF_2_ba1_PR_rep0	1	169	0	1	1	1
	mgAF_2_ba1_PR_rep1	1	191	0	1	1	1
mgAG_2					0	0	0
mgAH_2					0	0	0
mgAI_2					0	0	0
mgAJ_2	2gAJ_2_ba2_PR_rep0	5	171	171	1	0	0
mgAK_2	mgAK_2_ba1_PR_rep0	1	438	0	1	1	1
	mgAK_2_ba2_PL_rep0	1	428	0	1	1	1
mgAL_2	2gAL_2_ba1_PL_rep0	1	223	0	1	1	1
	2gAL_2_ba1_PR_rep0	1	237	0	1	1	1
mgAM_2					0	0	0

Table S19: Detailed results for group mGene170-novel.

Name	Sequence name	num blat hits	best score	second best score	hit to target	unambiguous	structure agreement
mgAA_3					0	0	0
mgAB_3					0	0	0
mgAC_3	mgAC_3_ba1_PL_rep0	1	281	0	1	1	0
	mgAC_3_ba1_PR_rep0	1	132	0	1	1	0
mgAD_3					0	0	0
mgAE_3					0	0	0
mgAF_3	mgAF_3_ba1_PL_rep0	7	229	188	1	0	0
	mgAF_3_ba1_PR_rep0	9	189	174	1	1	1
mgAG_3	mgAG_3_ba1_PL_rep0	4	159	144	1	1	1
	mgAG_3_ba1_PR_rep0	5	185	161	1	1	1
mgAH_3					0	0	0
mgAI_3	mgAI_3_ba1_PL_rep0	1	83	0	1	1	0
	mgAI_3_ba1_PR_rep0	1	98	0	1	1	1
mgAJ_3	mgAJ_3_ba1_PL_rep0	1	54	0	1	1	0
	mgAJ_3_ba1_PL_rep1	1	62	0	1	1	0
	mgAJ_3_ba1_PR_rep1	1	66	0	1	1	0
	mgAJ_3_ba2_PL_rep0	1	118	0	1	1	0
	mgAJ_3_ba2_PL_rep1	1	137	0	1	1	1
	mgAJ_3_ba2_PR_rep0	1	133	0	1	1	0
	mgAJ_3_ba2_PR_rep1	1	118	0	1	1	0
mgAK_3	2gAK_3_ba1_PL_rep0	2	143	124	1	1	1
	2gAK_3_ba1_PR_rep0	2	146	139	1	0	0
	mgAK_3_ba2_PL_rep0	3	163	103	1	1	0
	mgAK_3_ba2_PR_rep0	4	168	102	1	1	0
mgAL_3	2gAL_3_ba1_PL_rep0	128	66	61	0	0	0
	2gAL_3_ba1_PR_rep0	5	323	305	1	1	1
	2gAL_3_ba2_PL_rep0	4	9	-5	0	0	0
	2gAL_3_ba2_PR_rep0	2	321	201	1	1	1
	2gAL_3_ba3_PL_rep0	1	35	0	0	0	0
	2gAL_3_ba3_PR_rep0	2	236	113	1	1	1
	mgAL_3_ba1_PL_rep0	1	204	0	1	1	0
	mgAL_3_ba1_PR_rep0	1	211	0	1	1	0
mgAM_3	2gAM_3_ba1_PL_rep0	1	252	0	1	1	0
	2gAM_3_ba1_PR_rep0	1	249	0	1	1	0

cont'd on next page

Name	Sequence name	num blat hits	best score	second best score	hit to target	unambiguous	structure agreement
	mgAM_3_ba1_PL_rep0	1	299	0	1	1	0
	mgAM_3_ba1_PR_rep0	1	293	0	1	1	0
mgAN_3	mgAN_3_ba1_PL_rep0	1	111	0	1	1	1
	mgAN_3_ba1_PR_rep0	1	123	0	1	1	1
mgAO_3	mgAO_3_ba1_PL_rep0	1	176	0	1	1	1
	mgAO_3_ba1_PR_rep0	1	187	0	1	1	0
mgAP_3	mgAP_3_ba1_PL_rep0	1	349	0	0	0	0
	mgAP_3_ba1_PR_rep0	1	333	0	0	0	0
mgAQ_3					0	0	0
mgAR_3	mgAR_3_ba1_PL_rep0	1	180	0	1	1	1
	mgAR_3_ba1_PR_rep0	1	194	0	1	1	1
	mgAR_3_ba2_PL_rep0	1	73	0	0	0	0
	mgAR_3_ba2_PR_rep0	1	90	0	0	0	0
mgAS_3	mgAS_3_ba1_PL_rep0	1	140	0	1	1	0
	mgAS_3_ba1_PR_rep0	1	171	0	1	1	1
	mgAS_3_ba2_PL_rep0	1	183	0	1	1	0
	mgAS_3_ba2_PR_rep0	1	194	0	1	1	1
mgAT_3	mgAT_3_ba1_PL_rep0	1	208	0	1	1	1
	mgAT_3_ba1_PR_rep0	1	143	0	1	1	0
mgAU_3	mgAU_3_ba1_PL_rep0	1	783	0	1	1	0
	mgAU_3_ba1_PR_rep0	1	975	0	1	1	0
mgAV_3					0	0	0
mgAW_3					0	0	0
mgAX_3	mgAX_3_ba1_PL_rep0	1	170	0	1	1	1
	mgAX_3_ba1_PR_rep0	1	133	0	1	1	0
mgAY_3	mgAY_3_ba1_PL_rep0	1	124	0	1	1	1
	mgAY_3_ba1_PR_rep0	1	459	0	1	1	1
	mgAY_3_ba2_PL_rep0	1	454	0	1	1	1
	mgAY_3_ba2_PR_rep0	1	463	0	1	1	1
mgAZ_3					0	0	0
mgBA_3					0	0	0
mgBB_3	mgBB_3_ba1_PL_rep0	1	157	0	0	0	0
	mgBB_3_ba1_PR_rep0	1	386	0	0	0	0
	mgBB_3_ba2_PL_rep0	1	56	0	0	0	0
	mgBB_3_ba2_PR_rep0	1	173	0	0	0	0
	mgBB_3_ba3_PL_rep0	1	115	0	1	1	0
	mgBB_3_ba3_PR_rep0	1	126	0	1	1	0
mgBC_3					0	0	0
mgBD_3	mgBD_3_ba1_PL_rep0	1	207	0	1	1	1
	mgBD_3_ba1_PR_rep0	1	199	0	1	1	1
mgBE_3	mgBE_3_ba1_PL_rep0	2	415	55	1	1	1
	mgBE_3_ba1_PR_rep0	4	418	43	1	1	1
mgBF_3	mgBF_3_ba1_PL_rep0	1	57	0	0	0	0
	mgBF_3_ba1_PR_rep0	1	93	0	0	0	0
mgBG_3					0	0	0
mgBH_3	2gBH_3_ba2_PL_rep0	1	456	0	1	1	1
	2gBH_3_ba2_PR_rep0	1	452	0	1	1	1
	mgBH_3_ba1_PL_rep0	1	129	0	1	1	0
	mgBH_3_ba1_PR_rep2	1	97	0	1	1	0
mgBI_3					0	0	0
mgBJ_3	mgBJ_3_ba1_PL_rep0	1	195	0	1	1	1
	mgBJ_3_ba1_PR_rep0	1	200	0	1	1	1
mgBK_3	2gBK_3_ba1_PL_rep0	7	295	283	1	0	0
	2gBK_3_ba1_PR_rep0	6	293	281	1	0	0
	2gBK_3_ba2_PL_rep0	6	265	261	1	0	0
	2gBK_3_ba2_PR_rep0	5	273	269	1	0	0
	mgBK_3_ba1_PL_rep0	10	194	180	1	1	0
	mgBK_3_ba1_PR_rep0	4	45	45	1	0	0
	mgBK_3_ba2_PL_rep0	12	199	185	1	1	0
	mgBK_3_ba2_PR_rep0	7	185	162	1	1	1
	mgBK_3_ba3_PL_rep0	4	139	129	1	0	0
	mgBK_3_ba3_PR_rep0	5	153	151	1	0	0
mgBL_3	mgBL_3_ba1_PL_rep0	22	379	359	0	0	0
	mgBL_3_ba1_PR_rep0	28	377	369	0	0	0

cont'd on next page

Name	Sequence name	num blat hits	best score	second best score	hit to target	unambiguous	structure agreement
	mgBL_3_ba2_PL_rep0	1	107	0	0	0	0
	mgBL_3_ba2_PR_rep0	1	96	0	0	0	0
mgBM_3	mgBM_3_ba1_PL_rep0	1	410	0	0	0	0
	mgBM_3_ba1_PR_rep0	1	451	0	0	0	0
	mgBM_3_ba2_PL_rep0	1	248	0	0	0	0
	mgBM_3_ba2_PR_rep0	1	257	0	0	0	0
mgBO_3	mgBO_3_ba1_PL_rep0	1	123	0	1	1	0
	mgBO_3_ba1_PL_rep1	1	120	0	1	1	0
	mgBO_3_ba1_PR_rep0	1	85	0	1	1	1
	mgBO_3_ba1_PR_rep1	1	111	0	1	1	1
	mgBO_3_ba2_PL_rep0	1	129	0	1	1	0
	mgBO_3_ba2_PR_rep0	1	117	0	1	1	1
mgBP_3					0	0	0
mgBQ_3	mgBQ_3_ba1_PL_rep0	1	65	0	1	1	1
	mgBQ_3_ba1_PR_rep0	1	74	0	1	1	0
	mgBQ_3_ba2_PL_rep0	1	66	0	1	1	1
	mgBQ_3_ba2_PR_rep0	1	23	0	1	1	0
	mgBQ_3_ba3_PL_rep0	1	134	0	1	1	1
	mgBQ_3_ba3_PR_rep0	1	127	0	1	1	1
mgBR_3	mgBR_3_ba1_PL_rep0	1	95	0	1	1	0
	mgBR_3_ba1_PL_rep1	1	125	0	1	1	1
	mgBR_3_ba1_PR_rep0	1	81	0	1	1	1
	mgBR_3_ba1_PR_rep1	1	81	0	1	1	1
mgBS_3	mgBS_3_ba1_PL_rep0	1	170	0	1	1	1
	mgBS_3_ba1_PR_rep0	1	166	0	1	1	1
mgBT_3	mgBT_3_ba1_PL_rep0	1	93	0	1	1	0
	mgBT_3_ba1_PR_rep0	1	142	0	1	1	1
mgBU_3	mgBU_3_ba1_PL_rep0	1	187	0	1	1	1
	mgBU_3_ba1_PR_rep0	1	193	0	1	1	1
	mgBU_3_ba2_PL_rep0	1	191	0	1	1	1
mgBV_3					0	0	0
mgBW_3	mgBW_3_ba2_PL_rep0	1	174	0	1	1	1
	mgBW_3_ba2_PL_rep1	1	156	0	1	1	1
	mgBW_3_ba2_PR_rep0	1	136	0	1	1	1
	mgBW_3_ba2_PR_rep1	1	60	0	1	1	0
	mgBW_3_ba3_PL_rep0	1	271	0	1	1	1
	mgBW_3_ba3_PL_rep1	1	269	0	1	1	1
	mgBW_3_ba3_PR_rep0	1	285	0	1	1	1
	mgBW_3_ba3_PR_rep1	1	287	0	1	1	1
mgBX_3					0	0	0
mgBY_3	mgBY_3_ba1_PL_rep0	1	170	0	1	1	1
	mgBY_3_ba1_PL_rep1	1	167	0	1	1	1
	mgBY_3_ba1_PR_rep0	1	168	0	1	1	1
	mgBY_3_ba1_PR_rep1	1	143	0	1	1	1
mgBZ_3	mgBZ_3_ba1_PL_rep0	1	122	0	1	1	0
	mgBZ_3_ba1_PR_rep0	1	113	0	1	1	0
mgCA_3					0	0	0

Table S20: Detailed results for group missed-unconf180.

Name	Sequence name	num blat hits	best score	second best score	hit to target	unambiguous	structure agreement
mgAA_4					0	0	0
mgAB_4	2gAB_4_ba1_PL_rep1	1	449	0	1	1	0
	2gAB_4_ba1_PR_rep1	1	427	0	1	1	0
mgAC_4					0	0	0
mgAD_4					0	0	0
mgAE_4	mgAE_4_ba1_PL_rep0	1	133	0	1	1	1
	mgAE_4_ba1_PR_rep0	1	124	0	1	1	1
mgAF_4					0	0	0
mgAG_4					0	0	0
mgAI_4					0	0	0
mgAJ_4	2gAJ_4_ba1_PL_rep0	1	154	0	1	1	1

cont'd on next page

Name	Sequence name	num blat hits	best score	second best score	hit to target	unambiguous	structure agreement
	2gAJ_4_ba1_PR_rep0	1	149	0	1	1	1
	mgAJ_4_ba1_PL_rep0	1	166	0	1	1	0
	mgAJ_4_ba1_PR_rep0	1	164	0	1	1	1
mgAL_4					0	0	0
mgAM_4					0	0	0
mgAN_4					0	0	0
mgAO_4					0	0	0
mgAP_4	mgAP_4_ba1_PL_rep0	1	258	0	1	1	1
	mgAP_4_ba1_PR_rep0	1	218	0	1	1	1
	mgAP_4_ba2_PL_rep0	1	119	0	0	0	0
	mgAP_4_ba2_PR_rep0	1	113	0	0	0	0
mgAQ_4					0	0	0
mgAR_4	mgAR_4_ba1_PL_rep0	1	196	0	1	1	1
	mgAR_4_ba1_PR_rep0	1	196	0	1	1	1
mgAS_4					0	0	0
mgAT_4	mgAT_4_ba1_PR_rep0	2	283	44	0	0	0
	mgAT_4_ba1_PR_rep1	1	288	0	0	0	0
	mgAT_4_ba21_PL_rep0	1	219	0	0	0	0
	mgAT_4_ba21_PR_rep0	1	197	0	0	0	0
	mgAT_4_ba21_PR_rep1	1	44	0	0	0	0
	mgAT_4_ba22_PR_rep1	1	29	0	0	0	0
	mgAT_4_ba2_PR_rep0	2	187	46	0	0	0
	mgAT_4_ba2_PR_rep1	1	89	0	0	0	0
mgAU_4					0	0	0
mgAV_4					0	0	0
mgAW_4					0	0	0
mgAY_4					0	0	0
mgAZ_4					0	0	0
mgBA_4	2gBA_4_ba1_PL_rep1	1	223	0	0	0	0
	2gBA_4_ba1_PR_rep1	1	182	0	0	0	0
	2gBA_4_ba4_PL_rep0	1	225	0	0	0	0
	mgBA_4_ba1_PL_rep0	1	203	0	1	1	0
mgBB_4	2gBB_4_ba1_PL_rep0	1	281	0	0	0	0
	2gBB_4_ba1_PR_rep0	1	278	0	0	0	0
	mgBB_4_ba1_PL_rep0	2	157	132	1	1	1
	mgBB_4_ba1_PL_rep1	1	139	0	1	1	1
	mgBB_4_ba1_PR_rep0	2	128	80	1	1	0
	mgBB_4_ba1_PR_rep1	1	92	0	1	1	0
mgBC_4					0	0	0
mgBD_4					0	0	0
mgBG_4					0	0	0
mgBH_4					0	0	0
mgBJ_4	2gBJ_4_ba1_PL_rep0	1	173	0	0	0	0
	2gBJ_4_ba1_PR_rep0	1	165	0	0	0	0
mgBK_4					0	0	0
mgBL_4					0	0	0
mgBM_4	2gBM_4_ba1_PR_rep0	3	155	49	1	1	1
mgBN_4					0	0	0
mgBO_4					0	0	0
mgBP_4					0	0	0
mgBQ_4					0	0	0
mgBR_4	mgBR_4_ba1_PR_rep0	1	129	0	1	1	0
mgBS_4					0	0	0
mgBT_4	mgBT_4_ba1_PR_rep2	1	89	0	0	0	0
mgBU_4	2gBU_4_ba1_PL_rep0	1	462	0	1	1	1
	2gBU_4_ba1_PR_rep0	1	689	0	1	1	1
mgBV_4					0	0	0
mgBW_4					0	0	0
mgBY_4	mgBY_4_ba1_PL_rep1	18	640	16	1	1	0
	mgBY_4_ba1_PR_rep1	37	623	41	1	1	0
mgBZ_4					0	0	0
mgCA_4					0	0	0

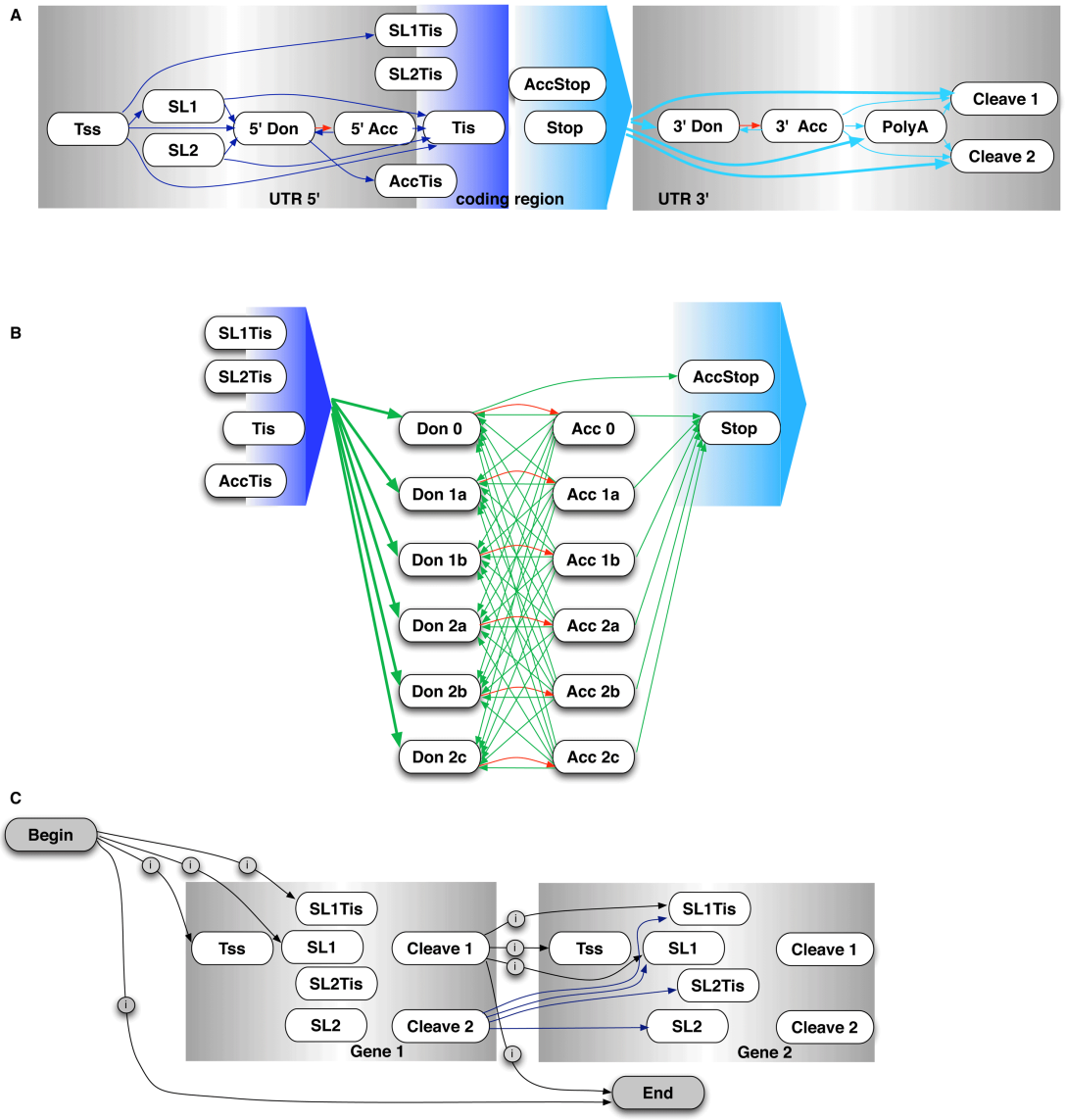


Figure S4: State model used in mGene, represented in three levels of detail. States are drawn as ovals, transitions between states as arrows. States are associated with at least one signal detector, transitions with content detectors and lengths features. For simplification, states that share all their outgoing transitions are drawn in boxes with right-pointing triangles (e.g. AccStop and Stop). The outgoing transitions are represented only once for all the grouped states. **A** Shown is the gene model, with all details for the untranslated regions. We allow splicing of both 5' and 3' UTRs. To account for the specific nematode gene biology, we model trans-splicing with SL1 and SL2 states. The model also allows to consider polyA sites in the 3' UTR, if strong corresponding predictions are present. For technical reasons we have introduced several mixed states, namely SL1Tis, SL2Tis, and AccTis, that all have the consensus AG|ATG and AccStop with a consensus GT|TAA. Blue transitions are associated with UTR content detectors, red transitions with intron content detectors. **B** shows the transitions and additional states in the coding region. Green arrows are associated with coding exon and frame content detectors, red ones with intron detectors. The splice sites in the coding regions are split to propagate the frame information across an intron to the next exon and to avoid creation of in-frame stop codons by splicing. Panel **C** shows transitions between genes. For the sake of clarity all transitions within a gene are removed and the states that mark the beginning and end of a gene are represented twice, once for gene 1 and once for gene 2. Each segmentation has to start in the Begin state and stop in the End state. The model allows for 0, 1 or more detected genes. To handle long intergenic segments we have introduced an additional dummy state 'intergenic-long' (marked as small circles with 'i'). The state cleave 2, is used to model operons. All transitions that are out-going from this state (blue arrows) are associated with the intercistronic content detector, all others (black arrows) with the intergenic content detector.

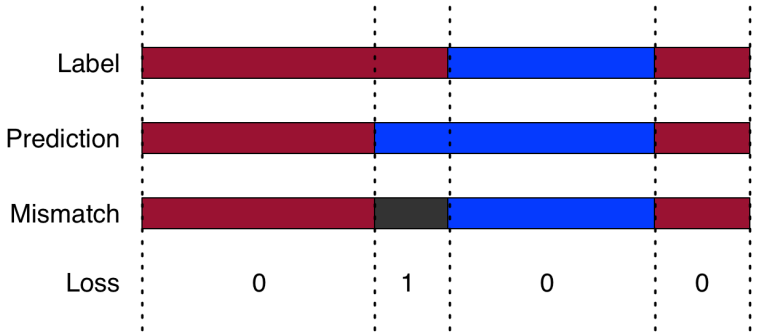


Figure S5: Schematic representation of segment loss. Consider an example with only two segment types (red and blue). The mismatching segment (gray) incurs a loss.