DFA7, a New Method to Distinguish between Intron-Containing and Intronless Genes

Chenglong Yu[1]◑, Mo Deng[2]◑, Lu Zheng[3], Rong Lucy He[4]*, Jie Yang[5]*, Stephen S.-T. Yau[6]*

1 Mind-Brain Theme, South Australian Health and Medical Research Institute, Adelaide, South Australia, Australia, **2** Quant Investment Department, Huashang Fund Management Co., Ltd., Beijing, China, **3** Electrical and Computer Engineering and CyLab Mobility Research Center, Carnegie Mellon University, Moffett Field, California, United States of America, **4** Department of Biological Sciences, Chicago State University, Chicago, Illinois, United States of America, **5** Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, Chicago, Illinois,United States of America, **6** Department of Mathematical Sciences, Tsinghua University, Beijing, China

Abstract

Intron-containing and intronless genes have different biological properties and statistical characteristics. Here we propose a new computational method to distinguish between intron-containing and intronless gene sequences. Seven feature parameters α, β, γ, λ, θ, ϕ, and σ based on detrended fluctuation analysis (DFA) are fully used, and thus we can compute a 7-dimensional feature vector for any given gene sequence to be discriminated. Furthermore, support vector machine (SVM) classifier with Gaussian radial basis kernel function is performed on this feature space to classify the genes into intron-containing and intronless. We investigate the performance of the proposed method in comparison with other state-of-the-art algorithms on biological datasets. The experimental results show that our new method significantly improves the accuracy over those existing techniques.

Citation: Yu C, Deng M, Zheng L, He RL, Yang J, et al. (2014) DFA7, a New Method to Distinguish between Intron-Containing and Intronless Genes. PLoS ONE 9(7): e101363. doi:10.1371/journal.pone.0101363

Editor: Ramin Homayouni, University of Memphis, United States of America

Received February 2, 2014; **Accepted** June 5, 2014; **Published** July 18, 2014

Funding: This research is supported by the U.S. National Science Foundation grant DMS-1120824, China National Science Foundation grant 31271408. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* Email: rhe@csu.edu (RLH); jyang06@uic.edu (JY); yau@uic.edu (SSTY)

◑ These authors contributed equally to this work.

Introduction

An important problem for geneticists as well as computer scientists involves classifying particular items into common groups. Here we focus on classifying gene sequences as either intron-containing or intronless. Intron-containing and intronless genes have different biological properties and statistical characteristics. For example, congruent with the Spearmans rank correlation, the comparison of intron-containing and intronless genes shows significantly reduced expression for intronless genes when compared to intron-containing genes [1]. Furthermore, intron-containing and intronless genes usually play important roles in evolution of proteins [2–4]. These observations raise interesting questions about the classification of intron-containing and intronless genes.

Peng et al. [5] have discovered that long-range correlation exists in the intron-containing genes but does not exist in the intronless genes. This work was based on a simple random-walk model of gene sequences, in which a pyrimidine led to a step up and a purine a step down. Consequently, the walk resulted in a definite landscape for a given sequence and only one parameter was calculated based on the landscape. This parameter was proposed to distinguish between the intron-containing and intronless genes. However, further study showed that this finding can not be used as a general method to identify intronless genes [6,7]. Zhang et al. [7,8] introduced a Z-curve consisting of three parameters. As an application, they used the Z-curve method to classify a dataset consisting of 100 intron-containing and 100 intronless genes. The discriminant accuracy as high as 89.0% can be obtained by using Fisher's linear discriminant algorithm based on Z-curve. However, although the distributions of three different biological types were displayed in Z-curve, it did not reveal the cross-correlations of distances between the nucleic bases, which are also important parameters to classify genes into intron-containing and intronless. In a similar way, Ma [9] created a model based on position weight function to describe genes by transforming them into quaternary numbers. Especially, this method indicates that E.coli K12s genome and the eukaryote yeasts genome have different strengths of single nucleotide periodicities. Yau et al. [10] firstly developed two-dimensional DNA graphical representation without degeneracy. Since then Yau and his collaborators have been studying efficient methods to cluster and classify DNA and proteins [11–18].

Some successful programs for exon/intron parsing are also proposed. For example, GENSCAN [19] was shown to be dramatically more accurate than the previous state-of-the-art prediction algorithms. It is based on a generalized hidden Markov model (GHMM) framework, and remains a popular bioinformatics tool. More recent de novo gene predictors have also been created, including N-SCAN [20] and EXONSCAN [21]. De novo gene predictors additionally made use of aligned gene sequence from other genomes [22]. Alignments can increase predictive accuracy since protein-coding genes exhibit distinctive patterns of conservation. These modern gene-finding or gene-parsing systems

Table 1. The seven feature parameters of 12 sample genes.

Genbank Acce. No.	α	β	γ	λ	θ	ϕ	σ
A00033	0.386	0.500	0.491	0.9933	0.9702	0.9415	0.2839
A17677	0.508	0.544	0.565	1.0757	1.0307	0.8789	0.2588
A11542	0.422	0.470	0.497	1.0431	1.0104	0.8742	0.2876
A22239	0.465	0.495	0.480	1.0673	1.0353	0.9351	0.2951
A24782	0.513	0.572	0.634	1.1773	1.1755	1.1214	0.3233
Z31371	0.500	0.526	0.558	1.0233	1.0336	1.0200	0.2732
M28289	0.576	0.635	0.632	1.3088	1.2991	1.0684	0.3462
V01510	0.575	0.617	0.716	1.3347	1.2171	1.0992	0.3429
U25810	0.560	0.657	0.623	1.2284	1.2740	1.1454	0.3340
M13580	0.480	0.622	0.624	1.2339	1.2114	1.0537	0.3337
U06674	0.507	0.498	0.555	1.1821	1.1609	0.9518	0.3274
J02989	0.518	0.577	0.537	1.2192	1.2463	1.0363	0.3495

doi:10.1371/journal.pone.0101363.t001

provide a prediction of precise (predicted) splice sites of the exons/introns in genes, while also producing the intron-bearing status of genes.

Here we propose a new approach, DFA7, to classify genes as to their intron-bearing status. We investigate three new parameters which are based on the cross-correlations between the distributions of distances of nucleic bases in gene sequences. Those new parameters together with Zhang et al. 's original three parameters [7] and the value of their total standard deviation can be used to significantly improve the accuracy of classification on intron-bearing status of genes. We perform our DFA7 method on three large gene datasets. The experimental results show that our method significantly improves the discriminant accuracy over those existing techniques. In addition, we examine our 7-dimensional feature vector by one-by-one feature deletion, and compare the SVM's efficiency with other machine learning approaches.

Materials and Methods

Background

The Z-curve theory of DNA sequences was firstly developed by Zhang et al. [7,8]. Consider a DNA sequence with N bases. Let the number of steps be denoted by n ($n=1,2,\ldots,N$). We count the cumulative numbers of base A, C, G, T which occur in the subsequence from the first to the nth base in the DNA sequence. The cumulative numbers are denoted by A_n,C_n,G_n and T_n, respectively. The Z-curve is a three-dimensional curve which consists of a series of nodes P_n ($n=1,2,\ldots,N$), whose coordinates are denoted by x_n, y_n and z_n. It is shown that

$$\begin{cases} x_n = 2(A_n+G_n)-n \\ y_n = 2(A_n+C_n)-n \\ z_n = 2(A_n+T_n)-n \end{cases}$$

where $n=1,2,\ldots,N$ and $A_0=C_0=G_0=T_0=0$. The connection of the nodes $P_0=0$, $P_1,\ldots,P_N$ one by one by straight lines is defined as the Z curve of the DNA sequence.

Detrended fluctuation analysis (DFA), firstly introduced by Peng et al. [5], is a scaling analysis method used to estimate long-range power law correlation parameters in noisy signals. By using this technique, Zhang et al. [7] calculated three exponents α, β, and γ for a given sequence based on its Z-curve. A 3-dimensional space is spanned by the three exponents. Each DNA sequence may be represented by a point in this space. For any query gene sequence, calculate its 3 exponents α, β, and γ, corresponding to a point in the 3-dimensional space. If the point is situated at the upper region of the separating plane, the gene is discriminated as an intronless one; otherwise, the gene is an intron-containing one.

For pursuing higher classification accuracy, more intrinsic parameters are needed. Here we propose a novel method, DFA7 method. In this approach, we introduce 4 new feature parameters λ, θ, ϕ, and σ for each DNA sequence based on DFA. Combining with 3 known parameters α, β, and γ from Z-curve we can generate a 7-dimensional feature space, which can be used to classify gene sequences into intron-containing and intronless with much higher discriminant accuracy.

DFA7 method

In a DNA sequence, D_j^m represents the cumulative distance of all nucleotides of nucleic base j ($j=A,\ C,\ G,\ T$) to the first nucleotide (regarded as origin) in m steps. Let t_i^j be the distance from the first nucleotide to the ith nucleotide if the ith nucleotide is

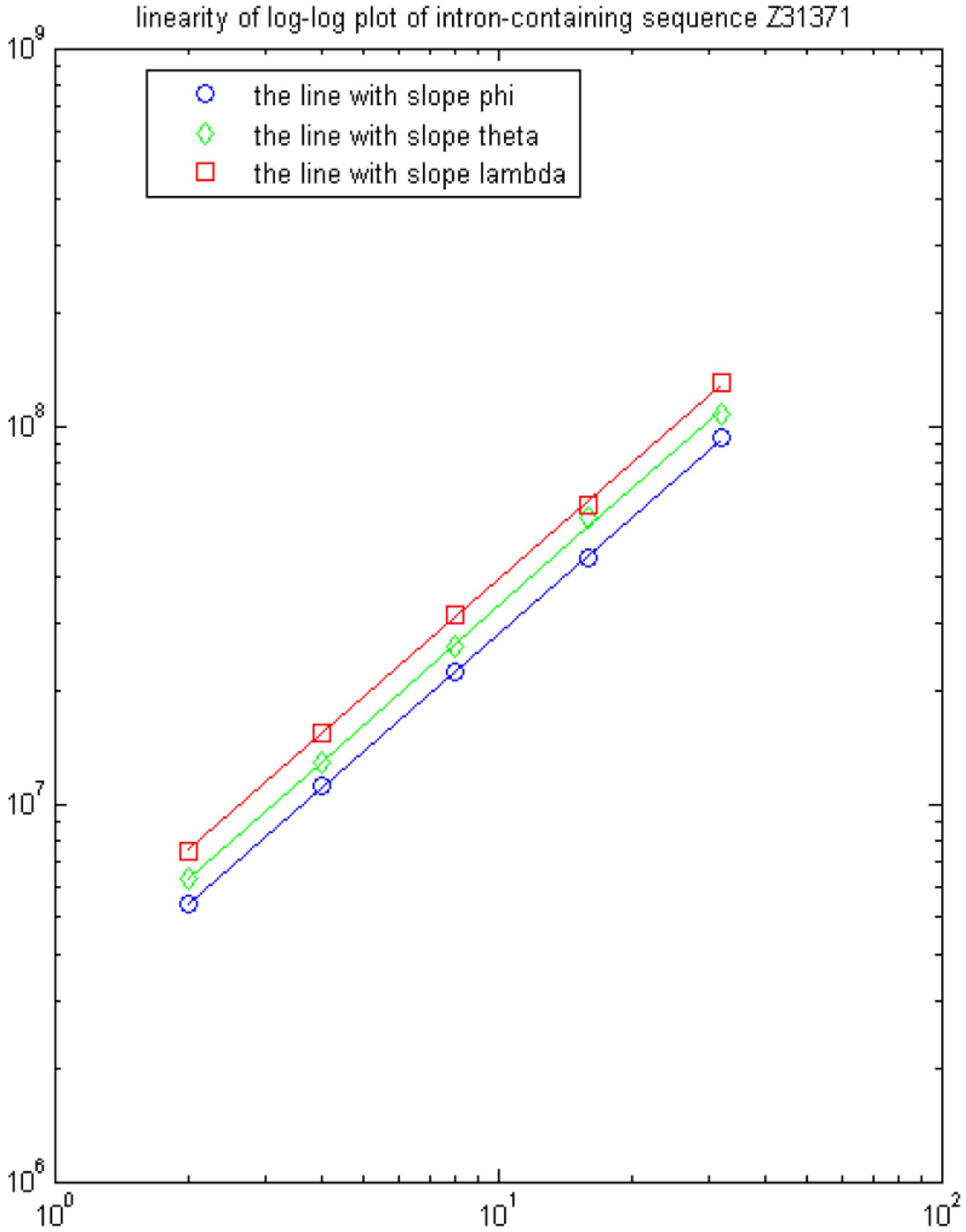

Figure 1. Linearity of log-log plots of three feature parameters λ, θ, and ϕ based on gene Z31371.
doi:10.1371/journal.pone.0101363.g001

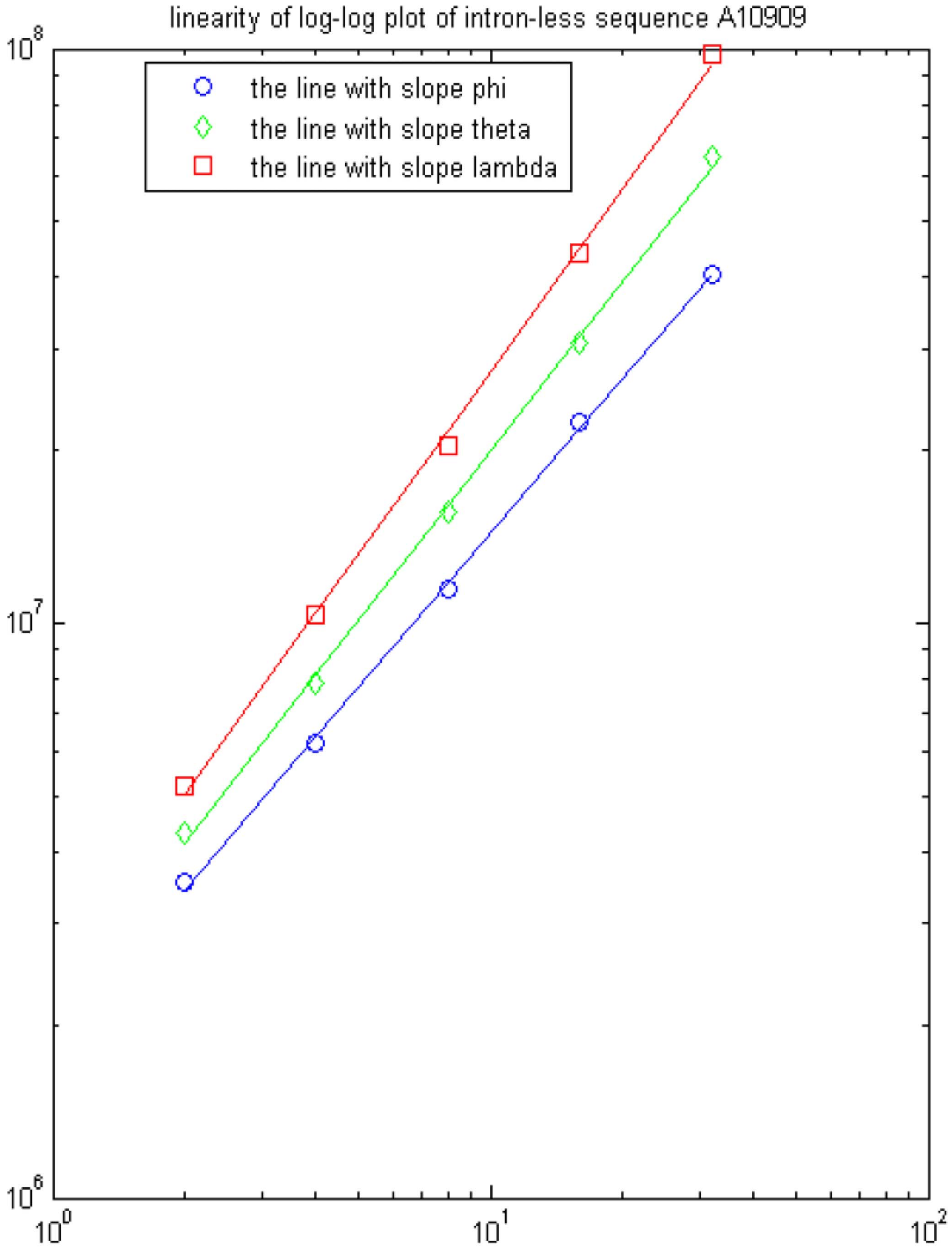

Figure 2. Linearity of log-log plots of three feature parameters λ, θ, and ϕ based on gene A10909.
doi:10.1371/journal.pone.0101363.g002

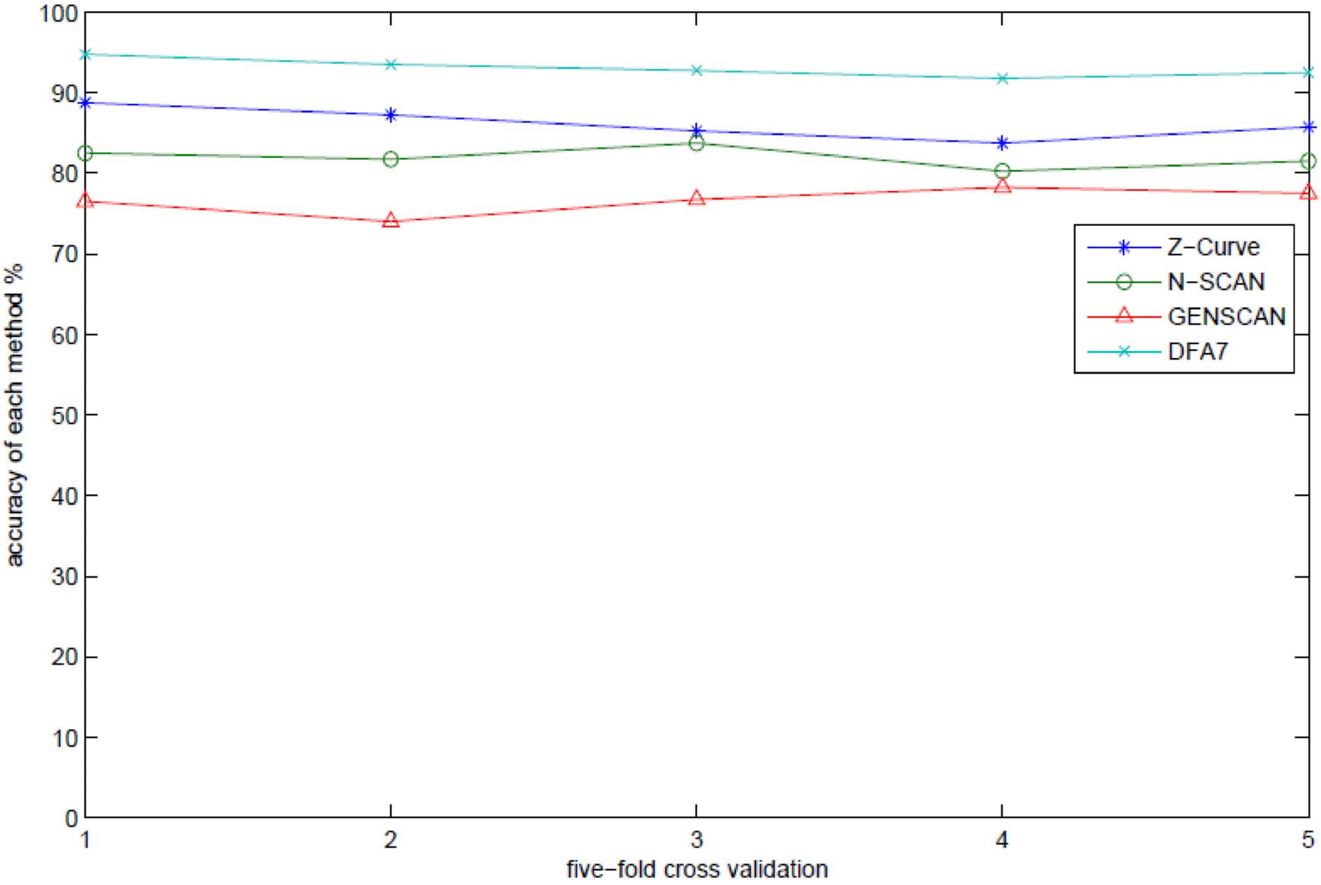

Figure 3. The accuracy comparison of DFA7 and other three methods on 2000 mixed prokaryotic and eukaryotic genes.
doi:10.1371/journal.pone.0101363.g003

j, otherwise; $t_i^j=0$. Thus $D_j^m=\sum_{i=1}^m t_i^j$. For example, $(AGCCTCGACT)$ is a DNA sequence. For nucleic base C, $t_1^c=0$, $t_2^c=0$, $t_3^c=2$, $t_4^c=3$, $t_5^c=0$, $t_6^c=5$, $t_7^c=0$, $t_8^c=0$, $t_9^c=8$, $t_{10}^c=0$, so $D_C^{10}=2+3+5+8=18$. Similarly, we get $D_A^{10}=7$, $D_G^{10}=1+6=7$ and $D_T^{10}=4+9=13$. Thus three types of cumulative distances can be defined as follows:

$$\begin{cases} D_n = (D_A^n+D_G^n)-(D_C^n+D_T^n) \\ E_n = (D_A^n+D_C^n)-(D_G^n+D_T^n) \\ H_n = (D_A^n+D_T^n)-(D_C^n+D_G^n) \end{cases}$$

where $n=1,2,\ldots,N$ and N is the length of the DNA sequence. D_n, E_n, and H_n reveal the cross-correlation of the "position" of each nucleic base in a DNA sequence. These cumulative distances are natural objects from the original DNA sequence, and embody more sequence information which Z-curve fails to provide.

Now we construct a 3×3 cumulative distance matrix based on D_n,E_n and H_n, and then use this matrix to compute 3 new feature parameters λ, θ, and ϕ. The algorithmic steps of setting the new parameters λ, θ, and ϕ are provided as follows:

(1) Set a window with width l, $l=2^n$, $n=1,2,3,4,5$, and move the window from the site l_0.

(2) Calculate the variation of each distribution at the two ends of the window,

$$\begin{aligned} \Delta D_l &= D_{l_0+l}-D_{l_0} \\ \Delta E_l &= E_{l_0+l}-E_{l_0} \\ \Delta H_l &= H_{l_0+l}-H_{l_0} \end{aligned}$$

(3) Shift the window sequentially from the beginning site $l_0=1$ to $l_0=2$ and so on, up to $l_0=N-l$, where N is the length of the sequence. For each value of l_0 starting from 1 to $N-l$, calculate each corresponding ΔD_l, ΔE_l, ΔH_l.

(4) Define the fluctuation functions

$$\begin{aligned} \rho_{DD}(l) &= |\overline{(\Delta D_l \Delta D_l)}-(\overline{\Delta D_l})(\overline{\Delta D_l})| \\ \rho_{EE}(l) &= |\overline{(\Delta E_l \Delta E_l)}-(\overline{\Delta E_l})(\overline{\Delta E_l})| \\ \rho_{HH}(l) &= |\overline{(\Delta H_l \Delta H_l)}-(\overline{\Delta H_l})(\overline{\Delta H_l})| \end{aligned}$$

$$\begin{aligned} \rho_{DE}(l)=\rho_{ED}(l) &= |\overline{(\Delta D_l \Delta E_l)}-(\overline{\Delta D_l})(\overline{\Delta E_l})| \\ \rho_{DH}(l)=\rho_{HD}(l) &= |\overline{(\Delta D_l \Delta H_l)}-(\overline{\Delta D_l})(\overline{\Delta H_l})| \\ \rho_{EH}(l)=\rho_{HE}(l) &= |\overline{(\Delta E_l \Delta H_l)}-(\overline{\Delta H_l})(\overline{\Delta E_l})| \end{aligned}$$

Table 2. Prediction results of different methods on 2000 mixed prokaryotic and eukaryotic genes (%).

Methods	1	2	3	4	5	average
GENSCAN	76.50	74.00	76.75	78.25	77.50	76.60 ± 1.61
N-SCAN	82.50	81.75	83.75	80.25	81.50	81.95 ± 1.29
Z-Curve	88.75	87.25	85.25	83.75	85.75	86.15 ± 1.90
DFA7	94.75	93.50	92.75	91.75	92.50	93.05 ± 1.14

doi:10.1371/journal.pone.0101363.t002

where the bars indicate an average over all site l_0 in the sequence. Then the matrix of fluctuation functions is defined as follows:

$$F=\begin{pmatrix} \rho_{DD}(l) & \rho_{DE}(l) & \rho_{DH}(l) \\ \rho_{ED}(l) & \rho_{EE}(l) & \rho_{EH}(l) \\ \rho_{HD}(l) & \rho_{HE}(l) & \rho_{HH}(l) \end{pmatrix}$$

Obviously, F is a real and symmetric matrix. Denote the three eigenvalues of F by ϵ_1,ϵ_2 and ϵ_3, such that $\epsilon_1 \geq \epsilon_2 \geq \epsilon_3$. Based on fluctuation analysis, we can get that

$$\epsilon_1 \propto l^{\lambda}, \quad \epsilon_2 \propto l^{\theta}, \quad \epsilon_3 \propto l^{\phi}$$

where λ, θ, and ϕ are three parameters determined by the slopes in the log-log plots. In other words, ϵ_i $(i=1,2,3)$ is a proportional function of l^j $(j=\lambda,\theta,\phi)$, i.e., $\epsilon_i = c \times l^j$, $c \neq 0$. Because of the nonlinear scaling of the axes, a function of the form $y=a \times x^b$ will appear as a straight line on a log-log graph, in which b is the slope of the line. Therefore, the parameters λ, θ, and ϕ can be computed by estimating each slope of log-log graph corresponding to ϵ_1,ϵ_2 and ϵ_3 from the numerical data.

(5) Estimate the slopes λ, θ, and ϕ of each log-log graph corresponding to ϵ_1, ϵ_2, and ϵ_3 computed in step (4).

Thus, for any given DNA sequence, we can calculate three parameters λ, θ, and ϕ by using the above five algorithmic steps. In step (1), in order to reduce the error for determining the slope ϕ and improve the computational efficiency, the values of $l=2^n$ $(n=1,2,3,4,5)$ are adopted. The line fitted by those l's is perfect. Even if the linearity is not so perfect in several cases, the squared error with respect to the slope and intercept parameters is minimized and the unique straight line can also be obtained by performing a least-squares fit of the data.

After determining λ, θ, and ϕ, we have a 7-dimensional feature vector consisting of parameters α, β, γ, λ, θ, ϕ, and σ, where σ is the sample standard deviation of the first 6 features. Then a machine learning method based on a support vector machine (SVM) equipped with a Gaussian radial basis kernel function (RBF) is used for prediction of intronless and intron-containing genes based only on the primary sequences.

We pick up 12 gene sequences as an illustration. The corresponding 7 feature parameters are calculated and listed in Table 1. The first 6 genes are intronless and the last 6 are intron-containing. Then an optimal hyperplane for separating intronless and intron-containing genes can be obtained by implementing SVM classifier based on this 7-dimensional feature space. Figure 1 and Figure 2 show the linearity of log-log plots of one intron-containing gene (Z31371) and one intronless gene (A10909) on the value λ, θ, and ϕ. We can see that eigenvalues ϵ_1, ϵ_2, and ϵ_3 are perfectly fitted by the lines with slope λ, θ and ϕ when $l=2^n$, $n=1,2,3,4,5$.

SVM parameter optimization

The kernel function $K(\cdot,\cdot)$ dominates the learning capability of the SVM [23]. We use radial basis kernel function $K(x_i,x_j)=exp(-\gamma||x_i-x_j||^2)$ to predict the intronless from intron-containing genes. As in many multivariate applications, the performance of the SVM for classification depends on the combination of several parameters. In general, the SVM involves two classes of parameters: the penalty parameter C and kernel type K. C is a regularization parameter that controls the tradeoff between maximizing the margin and minimizing the training error. The kernel type K is another important parameter. In the radial basis function used in this study, γ is an important parameter to dominate the generalization ability of SVM by regulating the amplitude of the kernel function. Accordingly, two parameters C and γ should be optimized. The parameter optimization is performed by using a grid search approach within a limited range. Prediction accuracy associated with mean-square-error (MSE) is used to select the parameters:

$$\text{Prediction accuracy} = 1 - \text{MSE}/(1-(-1))^2$$

Table 3. Prediction results of different methods on 1000 eukaryotic genes.

	DFA7 Method	Z-Curve Method
Average error counts on 800 training genes	148.4	217.0
Average error counts on 200 testing genes	50.4	58.8
Average error counts on total dataset	198.8	275.8
Average accuracy rate	(1000−198.8)/1000 = 80.12%	(1000−275.8)/1000 = 72.42%

doi:10.1371/journal.pone.0101363.t003

Table 4. Prediction results of GENSCAN on 1000 eukaryotic genes.

	GENSCAN-Vertebrate	GENSCAN-Maize
Partition 1	31	80
Partition 2	38	72
Partition 3	31	78
Partition 4	33	57
Partition 5	42	87
Total error counts	175	374
Average accuracy rate	(1000−175)/1000 = 82.50%	(1000−374)/1000 = 62.60%

doi:10.1371/journal.pone.0101363.t004

In the SVM classification, each data point represents a pair (geneID, y); if the gene is experimentally intronless, y is assigned to 1, otherwise, y is -1.

K-fold cross-validation

After all the seven parameters are determined, we can perform the K-fold cross-validation to estimate the accuracy of our predictive model. In a K-fold cross-validation, the original sample is randomly partitioned into K subsamples. Of the K subsamples, a single subsample is retained as the validation data for testing the model, and the remaining $K-1$ subsamples are used as training data. The cross-validation process is then repeated K times (the folds), with each of the K subsamples used exactly once as the validation data. Then the K results from the folds are averaged (or otherwise combined) to produce a single estimation. Five-fold cross-validation is performed on this work. Using a grid search method, the model with best (C,γ) is obtained, which yields a minimum misclassification rate. The program implementing SVM comes from the R package "e1071" which is based on the libsvm 2.8 package [24]. The discriminant accuracy is defined as follows:

$$\mathrm{p}=\frac{\text{The number of all correct discriminations}}{\text{The number of sequences in the testing dataset}}$$

GENESCAN, N-SCAN, Z-curve method, and our DFA7 method are implemented on the dataset in order to compare the results. Since the output of these gene parsing and finding systems provides us with the (predicted) beginning and ending coordinates of exons/introns in these sequences, it is easy for us to determine whether or not the gene is intron-bearing based on the prediction. For the intronless genes, the prediction is regarded as "false" if it predicts the splice sites between exons and introns. This confirms the existence of introns. For the intron-containing genes, the prediction is regarded as "false" if prediction shows "single exon". For the intron-containing or intronless genes which do contain exons, the prediction is regarded as "false" if the predicted answer is "no exon". By this way, the programs of GENESCAN and N-SCAN are performed directly on the testing set.

Results and Discussion

On 2000 mixed prokaryotic and eukaryotic genes

We test our DFA7 method on a large dataset which contains 1000 intronless genes (from prokaryotic genomes completely) selected randomly from UniProtKB/Swiss-Prot (release 15.1) and 1000 intron-containing genes selected randomly from Genbank database (release 170). These genes come from human, thale cress, mus musculus, and other eukaryotes in order to avoid similarity. The classical gene parsing systems GENSCAN, N-SCAN, Z-curve method, and DFA7 method are implemented. To avoid the bias of the discriminant accuracy defined in [7] and the similarity of testing dataset, five-fold cross-validation is used. In SVM classification, the parameter ranges are given as follows: $C\in(2^{-1},\ldots,2^{8})$, $\gamma\in(2^{-8},\ldots,2^{8})$. The prediction error profile has a minimum value at $(C,\gamma)=(16,2^{-6})$, indicating that the optimal values of C and γ to construct the SVM model are 16 and 0.015625, respectively.

Using the optimal values of C and γ, the prediction model is constructed based on the training set by using the SVM learning algorithm with RBF. To minimize data dependence on the prediction model, five-fold cross-validation sampling method is prepared. Each training set consists of 1600 sequences; half of them are randomly selected from data of intronless sequences, and the other half are randomly selected from data of intron-containing sequences. Each testing set is constructed using the left 400 sequences. The prediction results are listed in Table 2 and Figure 3. In Table 2, one can see that our DFA7 approach has higher accuracy than Z-Curve, N-SCAN, and GENSCAN methods.

Table 5. Prediction results of different methods on 1200 eukaryotic genes.

	DFA7 Method	Z-Curve Method
Average error counts on 960 training genes	204	266.2
Average error counts on 240 testing genes	62.6	74.4
Average error counts on total dataset	266.6	340.6
Average accuracy rate	(1200−266.6)/1200 = 77.78%	(1200−340.6)/1200 = 71.62%

doi:10.1371/journal.pone.0101363.t005

Table 6. Prediction results of 1000 eukaryotic genes based on DFA7 method by one-by-one feature deletion testing.

	All 7 parameters	deleting α	deleting β	deleting γ	deleting λ	deleting θ	deleting ϕ	deleting σ
Average error counts on 500 training genes	72.2	89.4	99.4	87.6	76.0	68.2	88.6	73.4
Average error counts on 500 testing genes	130.0	131.4	126.0	141.2	129.4	132.8	128.4	129.2
Average error counts on total dataset	202.2	220.8	225.4	228.8	205.4	201.0	217.0	202.6
Average accuracy rate	79.78%	77.92%	77.46%	77.12%	79.46%	79.90%	78.30%	79.74%

doi:10.1371/journal.pone.0101363.t006

On 1000 eukaryotic genes

In order to further illustrate the efficiency of our approach we test our method on another dataset. This carefully-selected dataset contains 1000 genes: 500 of them are human intronless genes which are randomly chosen from Intronless Gene Database [25], and the other 500 are intron-containing genes which are randomly chosen from Genbank database (release 170). Here we must emphasize that all the 1000 genes are from eukaryotic organisms. We partition the 1000 sequences into 5 parts; each part contains 200 sequences (100 intronless genes and 100 intron-containing genes). We treat each part as testing dataset, and the corresponding leftover (800 sequences) as training dataset.

We test two models: (1) using all 7 parameters (our DFA7 method), (2) using the first 3 parameters only (Z-Curve method). For each model, we firstly run 10-fold cross-validation on training dataset to get the best SVM tuning parameters C and γ, then fit the SVM model with the chosen parameters C and γ, and finally use the fitted SVM model to predict the labels of training dataset (get training errors) and the labels of testing dataset (get testing errors). We show the test results in Table 3. We can see that, for the gene dataset from eukaryotic organisms, our DFA7 method still has higher accuracy than Zhang et al. 's method.

We also compare our DFA7 method with GENSCAN, which can predict the locations and exon-intron structures of genes in genomic sequences from a variety of organisms. We use the same dataset partitions as before (the 1000 sequences into 5 parts; each part contains 200 sequences: 100 intronless genes and 100 intron-containing genes) for this program. In Table 4, we can see that, GENSCAN seems to have higher accuracy (82.50%) than ours (80.12%). However, when we are using GENSCAN, some parameters (for example, organism) are needed to be specified. There are only three organisms to choose from: Vertebrate, Arabidopsis, and Maize. Since our datasets are from eukaryotes (actually, most sequences are from human), we choose organism parameter as "Vertebrate". In this case, in order to get high-accuracy result for GENSCAN, we must know the prior information for the sequences, at least the hosts of these genes. Otherwise, for example, if we choose "Maize" as the organism parameter, GENSCAN got much lower accuracy (62.60%) as shown in Table 4. Thus it is a disadvantage for GENSCAN method. On the contrary, our DFA7 method does not need any prior information for the gene sequences. The 7 parameters are 7 natural quantities from the original DNA sequence, not set by any artificial intervention.

On 1200 eukaryotic genes

We also test our approach on another large dataset including 600 intronless genes and 600 intron-containing genes from three very different eukaryotic genomes (human, drosophila, and yeast). The 600 intronless genes include 200 human genes, 200 drosophila genes, and 200 yeast genes. Similarly, the 600 intron-containing genes also include 200 human genes, 200 drosophila genes, and 200 yeast genes. These genes are chosen from Intronless Gene Database [25], Berkeley Drosophila Genome Project [26], and Saccharomyces Genome Database [27]. We partition the 1200 sequences into 5 parts; each part contains 240 sequences (120 intronless genes and 120 intron-containing genes). We treat each part as testing dataset, and the corresponding leftover (960 sequences) as training dataset.

We test two models: (1) using all 7 parameters (our DFA7 method), (2) using the first 3 parameters only (Z-Curve method). For each model, we firstly run 10-fold cross-validation on training dataset to get the best SVM tuning parameters C and γ, then fit SVM model with the chosen parameters C and γ, and finally use

Table 7. Error counts for 800 eukaryotic genes with 5 partitions on 3 different machine learning methods.

	SVM7	SVM3	BPN3	RBFN3	RBFN7
Partition 1	170	223	298	296	246
Partition 2	178	211	319	287	279
Partition 3	129	213	306	286	235
Partition 4	151	216	315	274	232
Partition 5	114	222	306	284	237
Average error counts	148.4	217.0	308.8	285.4	245.8

doi:10.1371/journal.pone.0101363.t007

the fitted SVM model to predict the labels of training dataset (get training errors) and the labels of testing dataset (get testing errors). We show the test results in Table 5. We can see that, for the discriminant accuracy, our DFA7 method still largely outperforms Zhang et al. 's method.

Furthermore, we compare our DFA7 method with GENSCAN with the same dataset. When we are using GENSCAN, the parameter organism is needed to set. If we choose the organism parameter as "Vertebrate", the average accuracy rate for this dataset is only 40%. Actually, the genes in this dataset are from three very different eukaryotic organisms: mammalian (human), invertebrate (drosophila), and unicellular (yeast). The diversity of our dataset leads to the very low accuracy rate. Therefore, the GENSCAN can not output meaningful prediction results with the genes from very diverse hosts. However, our approach can be used on a universal dataset.

Examine DFA7 method by one-by-one feature deletion

To test whether there is any overfitting issue, we use the one-by-one feature deletion to justify our DFA7 method. Here we use the previous dataset of 1000 eukaryotic genes. We randomly divide the 500 intronless sequences into 250 and 250, and randomly divide 500 intron-containing sequences into 250 and 250, then use 250 intronless and 250 intron-containing genes as training dataset, and use the rest 250 intronless and 250 intron-containing genes as testing dataset. Thus, in this case, the training dataset and the testing dataset are totally independent.

For one-by-one feature deletion, we test 8 models: (1) using all 7 parameters, (2) using 6 parameters after deleting α, (3) using 6 parameters after deleting β, (4) using 6 parameters after deleting γ, (5) using 6 parameters after deleting λ, (6) using 6 parameters after deleting θ, (7) using 6 parameters after deleting ϕ, (8) using 6 parameters after deleting σ. For each model, we run 10-fold cross-validation on training dataset to get the best SVM tuning parameters C and "γ", fit SVM model with the chosen parameters C and "γ", then use the fitted SVM model to predict the label of training dataset (get training errors) and the label of testing dataset (get testing errors). We show the test results in Table 6.

From the results, we can see that, except "deleting θ", the model using all 7 parameters gives the highest average accuracy in Table 6. Here we must point out that θ is not an overfitting parameter. The model of "deleting θ" gives more error counts than the original DFA7 model for many partitions. However, there is only one random partition dataset in which the "deleting θ" model gives much less errors than DFA7. This big difference causes that the final average accuracy of the "deleting θ" model is slightly higher than the DFA7 model. Actually, for most random partitions of dataset, our DFA7 model gives much higher accuracies.

Comparison with other machine learning approaches

Based on our proposed new features, we also use other machine learning techniques to test the classification results. Using the same dataset of 1000 eukaryotic genes, we compare the performance of SVM, Backpropagation network (BPN), and Radial Basis Function network (RBFN) [28–30]. To train a BPN, we use the function "neuralnet" in the R package "neuralnet". To train an RBFN, we use the function "rbf" in the R package "RSNNS". Both of them are available at http://cran.r-project.org/web/packages/.

The training errors and testing errors following the same setup of partitions as in section "On 1000 eukaryotic genes" are shown in Table 7 and Table 8, respectively. Here SVM7 indicates SVM with all the 7 parameters, SVM3 indicates SVM with the first 3 parameters, and so on. Note that BPN7 is not available because the convergence of training procedure of BP network with all the 7 parameters is too slow. Based on the cross-validation results, we can see that BPN and RBFN have much more errors than SVM on the dataset.

Table 8. Error counts for 200 eukaryotic genes with 5 partitions on 3 different machine learning methods.

	SVM7	SVM3	BPN3	RBFN3	RBFN7
Partition 1	53	54	80	66	60
Partition 2	46	61	75	69	56
Partition 3	50	62	70	86	60
Partition 4	44	59	78	73	62
Partition 5	59	58	73	71	65
Average error counts	50.4	58.8	75.2	73.0	60.6

doi:10.1371/journal.pone.0101363.t008

Conclusion

In this work, we propose a new computational approach to distinguish between intron-containing and intronless gene sequences. In comparison with previous literature, the predictive performance of our method has been significantly enhanced. It is anticipated that the current method can be a complementary tool for distinguishing intronless genes from intron-containing genes. Seven feature parameters α,β,γ, λ, θ, ϕ, and σ can be computed using the algorithmic steps as we described. SVM classifier with RBF function is also performed on those seven parameters to classify the genes. Our new feature parameters can be used to discover more information hidden within the genes. Our DFA7 method mainly focuses on distinguishing intron-containing and intronless gene sequences. Further studies of this method will be needed to make specific splice-site predictions available. We will also evaluate the relative importance of the feature parameters and find more valuable features, which could help to classify genes and proteins with higher accuracy based on their structures. The datasets used in this work are available at http://www.math.uic.edu/~jyang06/publications/datasets/.

Author Contributions

Conceived and designed the experiments: RLH JY SSTY. Performed the experiments: CY MD LZ JY. Analyzed the data: MD JY. Wrote the paper: CY MD. All authors participated in the paper revision: CY MD LZ RLH JY SSTY.

References

1. Lanier W, Moustafa A, Bhattacharya D, Comeron JM (2008) EST analysis of *Osterococcus lucimarinus*, the most compact Eukaryotic genome, shows an excess of introns in highly expressed genes. PLOS ONE 3(5): e2171. doi:10.1371/journal.pone.0002171.
2. Shabalina SA, Ogurtsov AY, Spiridonov AN, Novichkov PS, Spiridonov NA, Koonin EV (2010) Distinct patterns of expression and evolution of intronless and intron-containing mammalian genes. Mol Biol Evol 27(8): 1745–1749.
3. Agarwal SM (2005) Evolutionary rate variation in eukaryotic lineage specific human intronless proteins. Biochem Biophys Res Commun 337(4): 1192–1197.
4. Agarwal SM, Jyotsana G (2005) Comparative analysis of human intronless proteins. Biochem Biophys Res Commun 331(2): 512–519.
5. Peng CK, Buldyrev SV, Coldberger AL, Havlin SL, Sciortino F (1992) Long-range correlations in nucleotide sequences. Nature 356: 168–170.
6. Prabhu VV, Claverie JM (1992) Correlations in intronless DNA. Nature 359: 782.
7. Zhang CT, Lin ZS, Yan M, Zhang R (1998) A novel approach to distinguish between intron-containing and intronless genes based on the format of Z curves. J Theor Biol 192: 467–473.
8. Zhang CT, Zhang R, Ou HY (2003) The Z curve database: a graphic representation of genome sequences. Bioinformatics 19: 593–599.
9. Ma BG (2007) How to describe genes: Enlightenment from the quaternary number system. BioSyst 90: 20–27.
10. Yau SST, Wang J, Niknejad A, Lu C, Jin N, et al. (2003) DNA sequence representation without degeneracy. Nucleic Acids Res 31(12): 3078–3080.
11. Yau SST, Yu C, He R (2008) A protein map and its application. DNA and Cell Biol 27(5): 241–250.
12. Carr K, Murray E, Armah E, He RL, Yau SST (2010) A rapid method for characterization of protein relatedness using feature vectors. PLOS ONE 5(3): e9550. doi:10.1371/journal.pone.0009550.
13. Yu C, Liang Q, Yin C, He RL, Yau SST (2010) A novel construction of genome space with biological geometry. DNA Res 17(3): 155–168.
14. Yu C, Deng M, Yau SST (2011) DNA sequence comparison by a novel probabilistic method. Inf Sci 181: 1484–1492.
15. Yu C, Cheng SY, He RL, Yau SST (2011) Protein map: an alignment-free sequence comparison method based on various properties of amino acids. Gene 486: 110–118.
16. Deng M, Yu C, Liang Q, He RL, Yau SST (2011) A novel method of characterizing genetic sequences: genome space with biological distance and applications. PLOS ONE 6(3): e17293. doi:10.1371/journal.pone.0017293.
17. Yu C, Deng M, Cheng SY, He RL, Yau SST (2013) Protein space: a natural method for realizing the nature of protein universe. J Theor Biol 318: 197–204.
18. Yu C, Hernandez T, Zheng H, Yau SC, Huang HH, He RL, Yang J, Yau SST (2013) Real time classification of viruses in 12 dimensions. PLOS ONE 8(5): e64328. doi:10.1371/journal.pone.0064328.
19. Burge C, Karlin S (1997) Prediction of complete gene structures in human genomic DNA. J Mol Biol 268: 78–94.
20. Gross SS, Brent MR (2006) Using multiple alignments to improve gene prediction. J Comput Biol 13(2): 379–393.
21. Shu JH, Yun SC, Lin CY, Tang CY (2005) EXONSCAN: Exon prediction with signal detection and coding region Alignment in homologous sequences. Proceedings of the 2005 ACM symposium on Applied computing: 202–203.
22. Brent MR (2005) Genome annotation past, present, and future: How to define an ORF at each locus. Genome Res 15: 1777–1786.
23. Boser BE, Isabelle MG, Vladimir NV (1992) A training algorithm for optimal margin classifiers. Proceedings of the fifth annual workshop on Computational learning theory, ACM: 144–152.
24. Chang CC, Lin CJ (2011) LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST) 2(3): 27.
25. Louhichi A, Fourati A, Reba A (2011) IGD: a resource for intronless genes in the human genome. Gene 488(1): 35–40.
26. Hoskins RA, et al. (2011) Genome-wide analysis of promoter architecture in drosophila melanogaster. Genome Res 21(2): 182–192.
27. Cherry JM, et al. (1998) SGD: Saccharomyces genome database. Nucleic Acids Res 26(1): 73–79.
28. Caiqing Z, Ruonan Q, Zhiwen Q (2008) Comparing BP and RBF Neural Network for Forecasting the Resident Consumer Level by MATLAB. In Computer and Electrical Engineering. ICCEE 2008. International Conference on (pp. 169–172).
29. Gnther F, Fritsch S (2010) Neuralnet: Training of neural networks. The R Journal 2(1): 30–38.
30. Kak S (2002) A class of instantaneously trained neural networks. Inf Sci 148(1): 97–102.