

Correlation of Amino Acid Preference and Mammalian Viral Genome Type

John R. Rose^{1,*}, William H. Turkett, Jr.¹, Iulia C. Oroian¹, William W. Laegreid², John Keele²

¹Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208; and ²U.S. Meat Animal Research Center, U.S. Department of Agriculture/Agricultural Research Service, Clay Center, Nebraska 68933

Abstract

Motivation: In the event of an outbreak of a disease caused by an initially unknown pathogen, the ability to characterize anonymous sequences prior to isolation and culturing of the pathogen will be helpful. We show that it is possible to classify viral sequences by genome type (*dsDNA*, *ssDNA*, *ssRNA positive strand*, *ssRNA negative strand*, *retroid*) using amino acid distribution.

Results: In this paper we describe the results of analysis of amino acid preference in mammalian viruses. The study was carried out at the genome level as well as two shorter sequence levels: short (300 amino acids) and medium length (660 amino acids). The analysis indicates a correlation between the viral genome types *dsDNA*, *ssDNA*, *ssRNA positive strand*, *ssRNA negative strand*, and *retroid* and amino acid preference. We investigated three different models of amino acid preference. The simplest amino acid preference model, 1-AAP, is a normalized description of the frequency of amino acids in genomes of a viral genome type. A slightly more complex model is the ordered pair amino acid preference model (2-AAP), which characterizes genomes of different viral genome types by the frequency of ordered pairs of amino acids. The most complex and accurate model is the ordered triple amino acid preference model (3-AAP), which is based on ordered triples of amino acids. The results demonstrate that mammalian viral genome types differ in their amino acid preference.

Availability: The tools used to format and analyze data and supplementary material are available at <http://www.cse.sc.edu/~rose/aminoPreference/index.html>

Contact: rose@cse.sc.edu

Introduction

While searching for features that might distinguish coding from non-coding sequence, we observed that in-frame amino acid distributions of the mammalian viral genomes were strongly correlated with viral genome type. These results are the subject of this paper.

We describe the three amino acid models that we have investigated and present results that demonstrate the correlation between amino acid preference in mammalian viruses and viral genome type. The three models are single amino acid preference (1-AAP), ordered pair amino acid preference (2-AAP), and ordered triple amino acid preference (3-AAP). By pair amino acid preference we mean the preference for certain ordered amino acid pairs. Likewise, triple amino acid preference is used to describe the preference for certain ordered amino acid triples.

Three types of analysis are conducted. At the genome level, cross-validation is used to demonstrate the predictive capability of amino acid preference of viral genome type. Next at the sequence level, two types of resampling analysis are used to investigate the correlation of amino acid preference and viral genome

* To whom correspondence should be addressed.

type. Bootstrapping of short sequences is done to examine whether the amino acid preference observed at the genome level appears throughout at the shorter sequence level. This type of analysis explores the local features of the genome. Direct bootstrapping of amino acid preference distributions is done to establish a type of upper boundary on the correlation that one would expect if all subsequences of a genome were generated by the genome amino acid preference distribution. This later study calibrates the analysis of the sequence bootstrapping. Taken together, these resampling approaches examine the difference between the local distributions of subsequences and the aggregate distribution of the genome. Finally, decimated models focusing on subsets of amino acids that are statistically significantly correlated with viral genome type are examined.

In the following methodology section the construction of data sets, models, and model testing and cross-validation is presented. The subsequent results section describes the analyses in which amino acid preference models are used to classify short mammalian viral sequence fragments (corresponding to 300 amino acids and 660 amino acids) according to the viral genome types *dsDNA*, *ssDNA*, *ssRNA positive strand*, *ssRNA negative strand*, and *retroid*. The results section concludes with a comparison of full 3-AAP models and reduced feature set (decimated) 3-AAP models.

Methodology

Data

The sequence data used to build and test the AAP models was obtained from NCBI. We gathered a total of 236 complete mammalian viral genomes that were available in the summer of 2003. For those viruses for which multiple examples of a given virus are available, only one example was included to avoid biasing the dataset towards overrepresented genomes. The breakdown of viruses by genome type is 60 *dsDNA*, 42 *retroid*, 16 *ssDNA*, 42 *ssRNA negative strand*, and 76 *ssRNA positive strand*. A list of the particular viruses included in the dataset can be found at <http://www.cse.sc.edu/~rose/aminoPreference/classificationFilesUsed.htm>.

Building Models

Each genome file was processed individually to extract coding sequences. The coding sequences were then used to produce amino acid preference distributions on a per genome basis for each of the three types of amino acid preference models that we investigated. In particular, each file was parsed to retrieve for each gene the start location, skip information (when present), stop location, strand information, and coding nucleotide sequence (FASTA format). The coding nucleotide sequences were translated into the corresponding amino acid representation.

Amino Acid Distribution Data

Having extracted the coding sections of each genome, amino acid distributions were then tabulated. The tabulation was carried out for the three model types: 1) amino acid preference (1-AAP), 2) ordered pair amino acid preference (2-AAP), and 3) ordered triple amino acid preference (3-AAP).

In the case of 1-AAP data, a histogram of the amino acids present in a genome was tabulated over all the amino acid sequences found in the genome. The histogram was then normalized to produce an amino acid distribution summing to 1.

In the case of 2-AAP and 3-AAP models, overlapping pairs and triples were extracted from the coding sections of genomes. If $\langle a_1 a_2 a_3 \dots a_n \rangle$ is a contiguous sequence of n amino acids, there are $n-1$ pairs in the sequence, *i.e.*, $\langle a_1 a_2 \rangle$, $\langle a_2 a_3 \rangle$, ..., $\langle a_{n-1} a_n \rangle$. For 2-AAP data, the number of occurrences of each of the 400 (20^2) possible ordered pairs for a genome was tabulated. The histogram is then normalized to sum to 1. Similarly, in each contiguous sequence of n amino acids, there are $n-2$ ordered triples, *i.e.*, $\langle a_1 a_2 a_3 \rangle$,

$\langle a_2 a_3 a_4 \rangle, \dots, \langle a_{n-2} a_{n-1} a_n \rangle$. The 3-AAP data for a genome was computed by tabulating the number of occurrences of each of the 8000 (20^3) possible ordered triples and then normalizing the resulting distribution.

SVM Training Datasets and Model Generation

Motivated by the observation in preliminary studies that amino acid distribution correlates with viral genome type we constructed and evaluated classifiers derived from AAP distributions. In the analysis, we used a supervised learning method based on the support vector machine (SVM), which belongs to the class of kernel-based learning methods (Vapnik, 1998). As the focus of this study is to test the hypothesis that amino acid preference is correlated with viral genome type rather than to determine which classification method is optimal¹ for this class of problem, we did not investigate alternative classification methods.

The SVM package that was used, SVM^{light}, was developed by Thorsten Joachims (Joachims, 1999; Joachims, 2002). This package is available for download at <http://svmlight.joachims.org/>. The format of the training data for this package is the class membership, which can be 1 or -1 for positive or negative example, followed by the feature vector.

Multiple two-class problems were formulated such that each class is contrasted with the remaining classes. For example, the training file for an SVM model that is used to distinguish *ssRNA positive strand* 1-AAP from other viral genome types based on the single amino acid preference contains the positive examples, *i.e.*, 1-AAP data from the *ssRNA positive strand* genome distributions and the negative examples, *i.e.*, the 1-AAP distributions of all other viral genome type genomes. Each positive example in the training dataset is represented by a 1, denoting a positive example, followed by the 20 features (percentages of each amino acid). Likewise, each negative example in the training dataset is represented by a -1, denoting a negative example, followed by the 20 features.

The initial analysis was performed using the 2-AAP model and sequences of 300 and 660 amino acids (Smith *et al.*, 2001)². The results produced by the 2-AAP model prompted us to also investigate 1-AAP and 3-AAP models. The motivation for investigating 1-AAP models was to uncouple the effects of individual amino acids from that of ordered pairs on classification. Finally, in spite of our concern that short query sequences of 300 to 660 amino acids might not contain enough information to support 3-AAP models, whose feature space has a dimension of 20^3 , we also created 3-AAP models for each of the five viral genome types.

Test Data and Cross-Validation

The performance and robustness of the models was evaluated by cross-validation (Kohavi, 1995; Stone, 1974). Cross-validation was carried out by taking the total available set of genome datasets and partitioning it into 10 approximately equal-sized sets. The genomes in each partition are randomly selected. The contents of each partition, as well as the size of each partition in terms of the number of genomes, amino acids, pairs of amino acids and triples of amino acids of each genome type is listed in Table 1. (Note significant differences in genome length that exist between different viral genome types are illustrated in Table 1. For example, single stranded DNA viruses can be roughly two orders of magnitude shorter than double stranded DNA viruses.)

¹ SVM training is a quadratic programming problem. Empirically, training for triple preference models using 236 samples (our database of full genomes) to train models for each of the 5 different genome types took 4 minutes 50 seconds wall clock time on a single 1.3GHz Itanium processor with 1GB of local memory. Classifying 1000 examples (equivalent to our sequence or distribution bootstrapping experiments) took 15 minutes wall clock time.

² A 300 amino acid sequence equates to 900 base pairs of nucleic acid sequence, an approachable read length for modern sequencing technology. If sequenced from both ends, 600 ~ 660 residues could be obtained.

Each partition contained approximately one tenth of the available genomes of each viral genome type. We then used 9 partitions to train the models and tested with the remaining partition. This was repeated nine times, leaving in turn a different partition of the data out of the training set and using it to validate the resulting models.

Table1: Partitions used in 1-AAP, 2-AAP, and 3-AAP model cross-validation											
		Partitions									
		1	2	3	4	5	6	7	8	9	10
dsDNA	genomes	6	6	6	6	6	6	6	6	6	6
	singles	180424	147320	230772	144057	120428	94641	113584	111139	189290	137450
	pairs	179814	146882	230072	143655	120130	94329	113194	110885	188747	137118
	triples	179204	146444	229372	143253	119832	94017	112804	110631	188204	136786
Retroid	genomes	5	5	4	4	4	4	4	4	4	4
	singles	14199	17894	11131	8059	7731	8407	11513	11398	8910	9300
	pairs	14175	17860	11111	8044	7712	8391	11495	11377	8892	9279
	triples	14151	17826	11091	8029	7693	8375	11477	11356	8874	9258
ssDNA	genomes	2	2	2	2	2	2	1	1	1	1
	singles	2671	5018	3639	2729	1769	1939	2006	2148	1359	2174
	pairs	2665	5005	3632	2722	1755	1935	2003	2144	1357	2170
	triples	2659	4992	3625	2715	1741	1931	2000	2140	1355	2166
ssRNA negative strand	genomes	5	5	4	4	4	4	4	4	4	4
	singles	22118	20569	17701	17116	16059	18489	21652	19315	15312	14534
	pairs	22084	20537	17669	17092	16040	18461	21624	19289	15289	14521
	triples	22050	20505	17637	17068	16021	18433	21596	19263	15266	14508
ssRNA positive strand	genomes	8	8	8	8	8	8	7	7	7	7
	singles	20910	30664	23500	38402	42292	37384	26139	32531	22615	36982
	pairs	20897	30645	23490	38376	42267	37359	26125	32514	22604	36949
	triples	20884	30626	23480	38350	42242	37334	26111	32497	22593	36916

Table 1: The 236 mammalian viral genomes representing five genome types were partitioned into ten sets, each containing approximately one tenth of the viral genomes of each genome type. The number of genomes, singles, pairs, and triples of amino acids is indicated by genome type for each of the ten sets.

Bootstrapping

The bootstrap was used to evaluate the degree to which amino acid preference distributions of short sequences conform to the model. The models were derived from all 236 genomes in our data set. Two levels of granularity of bootstrapping were then carried out. One type of bootstrap involved randomly selecting query sequences from the same dataset, computing normalized 1-AAP, 2-AAP, and 3-AAP distributions from the query sequences and then using the models to classify them. In the second type of bootstrap, 1-AAP, 2-AAP, and 3-AAP distributions were directly sampled from randomly selected genome distributions and classified.³

Bootstrapping Sequences

In the case of sequence bootstrapping, the models were evaluated with respect to their accuracy in classifying 1000 short bootstrapped sequences. This was carried out twice: for sequences 300 amino acids long and for sequences 660 amino acids long. Each sample was generated by randomly selecting a viral genome (with replacement) from our database and then randomly sampling the genes within it. An analysis of the distribution of gene lengths in our data set revealed that 53.3% of the annotated genes are shorter than 300 amino acids and 84.9% are shorter than 660 amino acids. In order to avoid having a bias that would exclude short genes, each bootstrapped sample is made of an ensemble of randomly sampled gene fragments. These gene fragments are 60 amino acids long for aggregate samples of 300 amino acids

³ The built-in Java Random class (java.util.Random) as implemented in Java version 1.4 was used. We downloaded the NIST RNG Statistical Test Suite (<http://csrc.nist.gov/rng/>) from the U.S. National Institute of Standards and Technology and verified that the random numbers generated are uniformly random.

and aggregate samples of 660 amino acids⁴. The analysis of gene lengths indicated that only 1.9% of annotated genes are shorter than 60 amino acids.

More precisely, the following sampling approach was adopted. The 300 amino acid long sequences were derived by randomly selecting five genes (with replacement), and then randomly selecting 60 contiguous amino acids from each gene⁵. Thus each sample is made up of independently sampled subsequences from the selected genome. Each sample was then processed to produce normalized 1-AAP, 2-AAP, or 3-AAP distributions. In order to avoid artifacts as a result of concatenating two sequences, pairs and triples were only extracted from the contiguous 60 amino acid subsequences. In other words, no sampled pairs or triples spanned two subsequences. Thus from the five 60 subsequence 295 amino acid pairs were sampled for 2-AAP models and 290 amino acid triples were sampled for 3-AAP models.

In a similar manner, the 660 amino acid long samples were derived by randomly selecting 11 genes with replacement and selecting 60 contiguous amino acids from each gene. Again, no pairs or triples of amino acids spanning two sequences were sampled. Thus 649 amino acid pairs were sampled for 2-AAP models and 638 amino acid triples were sampled for 3-AAP models. These samples were processed to produce normalized 1-AAP, 2-AAP, and 3-AAP distributions.

Bootstrapping Distributions

In addition to bootstrapping sequences, we also directly bootstrapped distributions to test the models induced from the training data. In sequence bootstrapping, the distribution of the randomly selected sequence does not necessarily match the aggregate distribution of its genome. The intent in bootstrapping a distribution is to create a sample distribution reflecting the sparseness of the derived sample sets from short bootstrapped sequences in the previous section but that conforms to the genome distribution. (We refer to a distribution derived from a sparse sample set as a “sparse distribution” in subsequent discussions.) The implicit hypothesis is that the individual viral genome distributions reflect the distribution of their viral genome type. If this is true, then there is the expectation that sparse distributions randomly sampled from randomly selected genome distributions should also be correctly classified by the corresponding model. Moreover, classification performance should be higher than for bootstrapped sequences since the amino acid preference of actual sequences within a genome may vary from that of the genome as a whole.

Each of the models, 1-AAP, 2-AAP, and 3-AAP, was evaluated with 1000 bootstrap samples. Each bootstrap sample is created by first selecting a random virus from the database. Next a normalized distribution is tabulated from the coding sequences in the selected viral genome, according to the description in the section *Amino Acid Distribution Data*. Finally, the required number of singles, pairs, or triples is randomly sampled from the distribution. To illustrate random sampling from a distribution, consider the normalized distribution of pairs of amino acids in a genome. The sum of the probabilities of the pairs of amino acids is 1. Denoting the distribution of the 400 possible amino acid pairs as P_1, \dots, P_{400} , we can randomly sample this distribution by generating a random number $R \sim Uniform(0, 1)$, such that $0 \leq R \leq 1$ and selecting the least j such that $R \leq \sum_{i=1}^j P_i$. The count corresponding to the j^{th} amino acid pair is incremented. This process is then repeated for the n required samples resulting in a sample

⁴ By using ensembles of randomly sampled gene fragments, we essentially treat individual genes within a genome as independent; consequently, our analysis overlooks any potential positional correlation between successive genes along the genome.

⁵ We do not mean to imply that one would sample sequences in a laboratory by pooling subsequences. Since our interest in this study focuses on general amino acid preference over all genes, we did not wish to exclude shorter genes from our analysis. Consequently, we have adopted a pooling method which allows us to include contributions from all but the shortest (< 60 aa) of annotated genes in our data set.

distribution. The resulting distribution is then normalized. If the numbers R are uniformly random, then the expected distribution of samples will be proportional to the distribution that was sampled.

In order to facilitate comparisons with the results of bootstrapping sequences 300 and 660 amino acids long, comparable distributions were directly bootstrapped from genome distributions. In the bootstrapping sequence tests, sequences that were 300 amino acids long produced 1-AAP distributions of 300 amino acids, 2-AAP distributions of 295 pairs and 3-AAP distributions of 290 triples. Corresponding randomly sampled distributions with the same numbers were created. Specifically, we bootstrapped sampled 1-AAP distributions containing 300 amino acids, 2-AAP distributions containing 295 pairs, and 3-AAP distributions containing 290 triples. Likewise, we created bootstrapped 1-AAP, 2-AAP, and 3-AAP distributions of the same numbers as that derived from the 660 amino acid sequences. In this case the 1-AAP distributions contained 660 amino acids, the 2-AAP distributions contained 649 pairs, and the 3-AAP distributions contained 638 triples.

Results and Discussion

The performance of the models under each of the three types of evaluation methods described in the methodology section is presented in this section. The cross-validation results demonstrate the degree to which the different models are able to classify viral genomes on the basis of aggregate genome amino acid preference distributions. The bootstrapped sequence results demonstrate the performance of these models on much sparser distributions. Finally, the bootstrapped distribution results show the degree to which sparse distributions that are representative of the distribution space around that of the individual genome distributions are correctly classified. This tests the hypothesis that individual viral genome distributions reflect the distribution of their viral genome type.

Cross-Validation

The results of the cross-validation tests are shown in Table 2, Table 3, and Table 4 for 1-AAP, 2-AAP and 3-AAP models, respectively. The most remarkable aspect of Table 1 is how well the 1-AAP model performs in the cases of *retroid*, *ssRNA negative strand*, and *ssRNA positive strand* viral types. Recall that the cross-validation tests entail the classification of genome distributions. It is therefore a surprising result that individual amino acid preferences at the genome level with no structural information are enough to achieve this degree of accuracy in classification. A nonparametric analysis of variants (Kruskal and Wallis, 1952; Zar, 1998) reveals that four of the amino acids (L, Q, V, and W) have a p -value of less than 10^{-15} and therefore their distribution is strongly correlated with viral genome type. For example, in our study set of viral genomes Valine consistently occurs with greater frequency in the *ssRNA positive strand* genomes than in the other genome classes.

The second interesting result is the misclassification of *dsDNA* and *ssDNA* viruses. The major trend is the misclassification of these viruses as *ssRNA positive strand*. Out of 60 *dsDNA* viruses, 29 are misclassified as *ssRNA positive strand*. Similarly, 10 out of 16 *ssDNA* viruses are misclassified as *ssRNA positive strand*.

Consideration of amino acids pairs, as shown in Table 3, results in a significant overall improvement of classification from 69% to 91.5%. Out of 236 viral genomes only 163 are correctly classified by 1-AAP models as shown in Table 2. In contrast, 216 out of 236 are correctly classified by 2-AAP models as shown in Table 3. In particular, the classification rate for *dsDNA* viruses increases from 20% for the 1-AAP model to 88% for the 2-AAP model. Less dramatic are the improvements in classification rates of *ssRNA negative strand* from 73% to 90% and *ssDNA* from 18% to 50%. These results suggest that in addition to correlation between single amino acid distribution and viral genome type, there is also correlation with structure in the distribution, e.g., amino acid pairs.

The 3-AAP models increased the overall classification performance to 97.46% correct, i.e., 230 out of 236 viral genomes were correctly classified. This result is quite surprising. While one might expect that

adding further structural constraints would improve classification, the feature space for triples is very large: the number of ordered triples is 8000 (20^3), a number larger than the number of ordered triples in many of the *retroid* and *ssDNA* genomes in the data set. Consequently, we were initially doubtful that smaller genomes would contain enough information to provide representative distributions for classification based on the 3-AAP model. The cross-validation results suggest that a subset of the 8000 ordered triples are adequate for accurate classification of the viruses in the data set.

	dsDNA	Retroid	ssDNA	ssRNA Negative	ssRNA Positive	Total	% Correct
dsDNA	12	9	0	10	29	60	20.00
Retroid	0	41	0	0	1	42	97.62
ssDNA	0	3	3	0	10	16	18.75
ssRNA Negative	0	1	0	31	10	42	73.81
ssRNA Positive	0	0	0	0	76	76	100.00

	ssDNA	Retroid	ssDNA	ssRNA Negative	ssRNA Positive	Total	% Correct
dsDNA	53	0	0	0	7	60	88.33
Retroid	0	42	0	0	0	42	100.00
ssDNA	1	2	8	0	5	16	50.00
ssRNA Negative	0	1	0	38	3	42	90.48
ssRNA Positive	1	0	0	0	75	76	98.68

	dsDNA	Retroid	ssDNA	ssRNA Negative	ssRNA Positive	Total	% Correct
dsDNA	58	0	0	0	2	60	96.67
Retroid	0	42	0	0	0	42	100.00
ssDNA	1	0	14	0	1	16	87.50
ssRNA Negative	0	0	0	41	1	42	97.62
ssRNA Positive	1	0	0	0	75	76	98.68

Tables 2, 3, and 4: Performance of cross-validation testing over 10 runs for the 1-AAP, 2-AAP, and 3-AAP models, respectively.

Bootstrapping of Sequences

The results of the bootstrapping of sequences 300 amino acids long for 1-AAP, 2-AAP, and 3-AAP models are shown in Figure 1. (A detailed breakdown of the classification results similar to Tables 2, 3, and 4 for these models can be found in the supplementary material.) In contrast to cross-validation tests, which employ distributions generated from entire genomes, these tests evaluate the performance of the models by using distributions generated from short bootstrapped sequences. It is not surprising that the performance of the 1-AAP model is lower for this testing set than it is for that of genome distributions. Out of 1000 bootstrapped sequences, 564 are correctly classified, *i.e.*, 56.4%. In comparison, 69% of the genome distributions are correctly classified in the 1-AAP cross-validation tests. Nonetheless, in the case of the *ssRNA positive strand* viruses in this test set, amino acid distributions based on sequences 300 amino acids long provide enough information to correctly classify 86.75% of them.

The same bootstrapped 300 amino acid sequences were processed to generate 2-AAP test distributions for evaluating the 2-AAP models. The results of these tests are shown in Figure 1. The overall number of correctly classified sequences compared with 1-AAP improved from 56.4% to 77.7%. While this is a significant improvement over the 1-AAP models, it does not approach the performance of the 2-AAP models on genome distributions as shown in Table 3, *i.e.*, 77.7% versus 91.5%. The obvious reason is the sparseness of the sample set relative to the number of categories in the distribution. In this case, the sample distribution is derived from a sequence that contains 295 amino acid pairs. The total number of possible distinct pairs for 2-AAP models is 400 (20^2). With a sample set of this size at least 25% of the possible pairs are guaranteed not to appear in any given sequence.

Finally, the bootstrapped 300 amino acid sequences were processed to generate 3-AAP distributions for evaluating the 3-AAP models. The sample distributions in this case are derived from 290 ordered amino

acid triples. This is a very small number compared to the total possible number of distinct triples, *i.e.*, 20^3 . (We observed all 8000 possible triples in our data set). In comparison to the distributions for 1-AAP and 2-AAP models, these distributions are derived from sample sets that are extremely sparse. For a given sequence, no more than 290 out of 8000 of the possible triples will appear. Nonetheless, for the 3-AAP models, 941 out of 1000 bootstrapped sequences are correctly classified. This result is additional evidence that a small subset of the 8000 possible triples is responsible for the high classification accuracy of viral genomes in our data set. This is explored in greater detail at the end of this section.

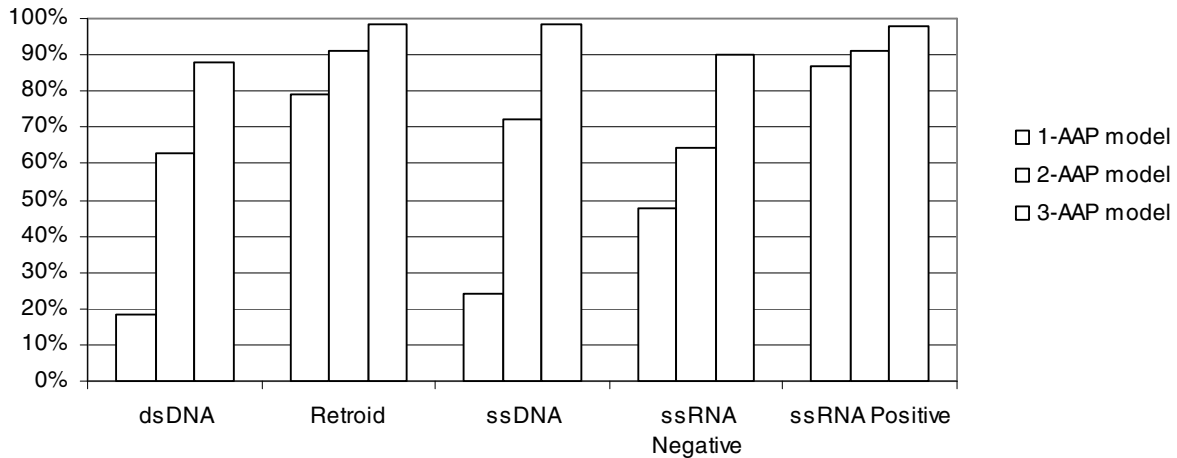


Figure 1: Performance of the 1-AAP, 2-AAP, and 3-AAP models on distributions derived from bootstrapped sequences 300 amino acids long.

In addition to bootstrapping sequences 300 amino acids long, we also bootstrapped sequences 660 amino acids long. The results of the bootstrapping of sequences 660 amino acids long for 1-AAP, 2-AAP, and 3-AAP models are shown in Figure 2. These results together with those for the shorter 300 amino acid sequences indicate of the sensitivity of our models to short sequences within the range of length that can be expected prior to sequence assembly. As in the case of shorter 300 amino acid sequences, the 3-AAP models perform better than 2-AAP, which in turn are an improvement over 1-AAP models. In comparison with the results of the shorter 300 amino acid sequences, the results for 660 amino acid sequences show consistent improvement. The overall performance of the 1-AAP models improve from 56.4% to 61.1%.

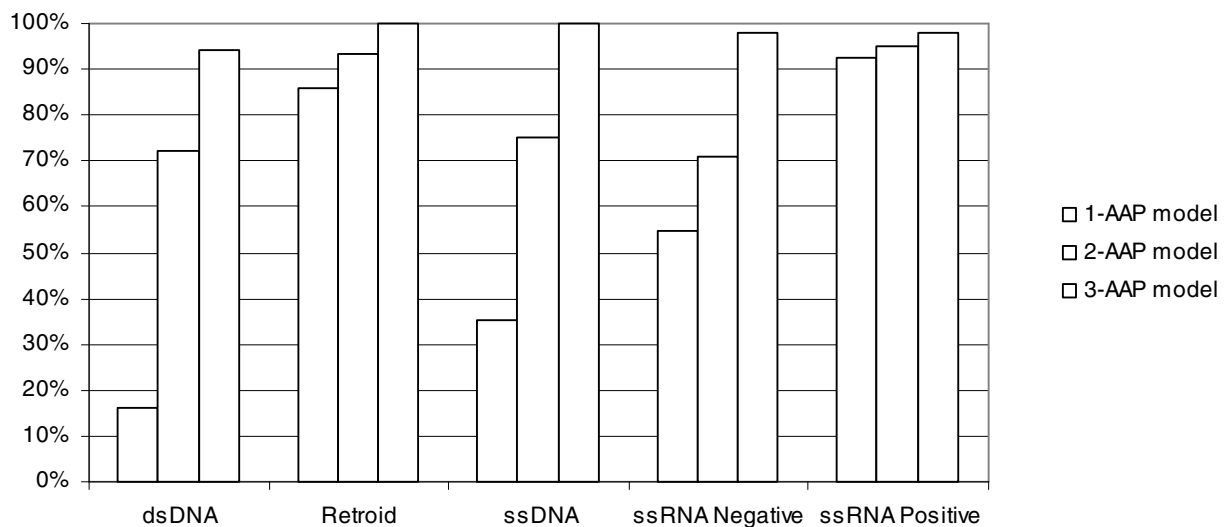


Figure 2: Performance of 1-AAP, 2-AAP, and 3-AAP models on distributions derived from bootstrapped sequences 660 amino acids long.

The 2-AAP models are able to correctly classify 83.1% of the distributions compared to 77.7% for the shorter sequences. For the 2-AAP model, each 660 amino acid sequence is sampled as eleven 60 amino acids subsequences resulting in a total of 649 amino acid pairs. We note that while the result is a sparse distribution given that there are 400 possible amino acid pairs, the 2-AAP models perform reasonably well. The best classification results are produced by the 3-AAP models. The overall performance improves from 94.1% for sequences 300 amino acids long to 97.5% for sequences 660 amino acids long.

Bootstrapping of Distributions

The motivation for bootstrap sampling of genome distributions is to examine the degree to which sparse distributions representative of individual genome distributions are correctly classified. This tests the hypothesis that individual viral genome distributions reflect the distribution of their viral genome type as learned by the SVM models. The results of the bootstrapping of very small distributions comparable to that derived from sequences 300 amino acids long are shown for 1-AAP, 2-AAP, and 3-AAP models in Figure 3. In comparison to the results for bootstrapped sequences, the classification performance for bootstrapped distributions displays a consistent improvement for all models. Specifically, the 1-AAP results improve from 56.4% to 62%, the 2-AAP results improve from 77.7% to 86.5%, and the 3-AAP results improve from 94.1% to 98.4%.

The results of the bootstrapping of small distributions comparable to that derived from sequences 660 amino acids long are shown for 1-AAP, 2-AAP, and 3-AAP models in Figure 4. In comparison to the results for bootstrapped sequences shown in Figure 2, the classification performance for bootstrapped distributions displays a consistent improvement for all models. The 1-AAP results improve from 61.1% to 66.7%. The 2-AAP results improve from 83.1% to 91.5%. Finally, the 3-AAP results improve from 97.5% to 99.4%.

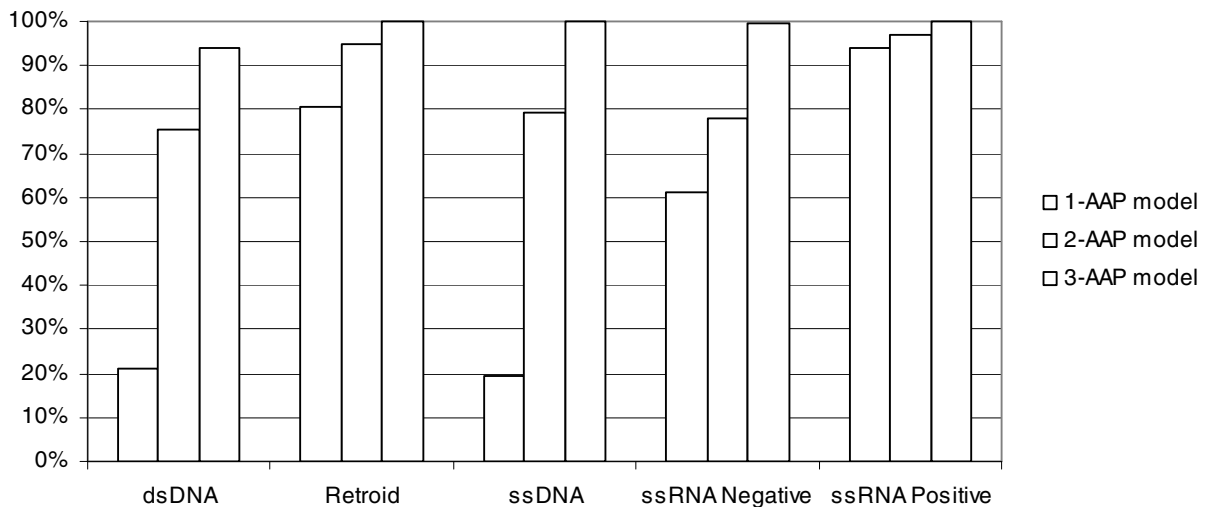


Figure 3: Performance of the 1-AAP, 2-AAP, and 3-AAP models on sparse distributions sampled from 1000 bootstrapped genome distributions equivalent to 300 amino acids.

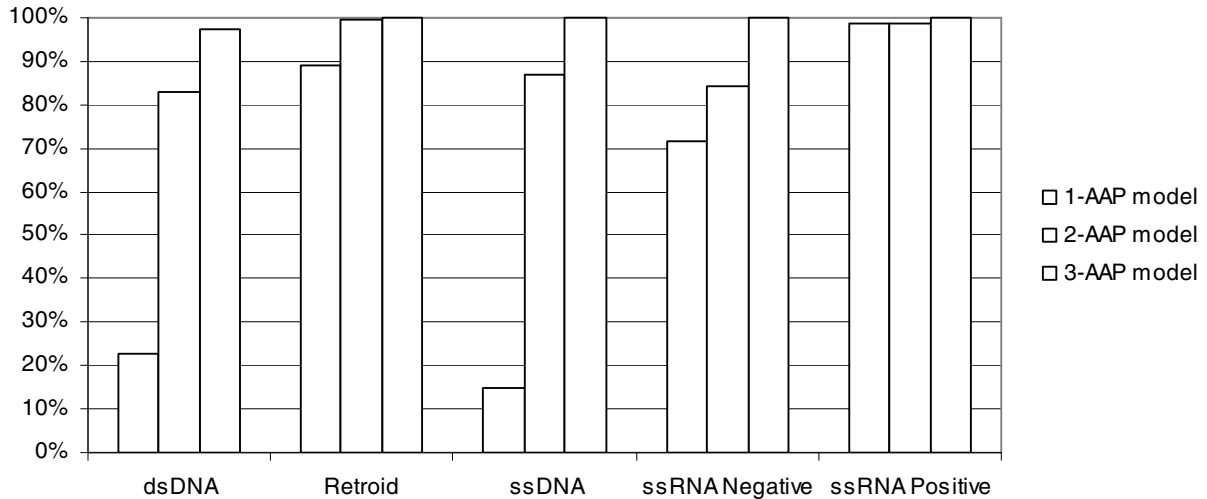


Figure 4: Performance of the 1-AAP, 2-AAP, and 3-AAP models on sparse distributions samples from 1000 bootstrapped genome distributions equivalent to 660 amino acids.

From these results one may draw two conclusions. First, the sparse bootstrapped distributions are closer to the viral genome type distribution learned by the SVM models than the distributions derived from bootstrapping actual viral sequences from our data set. Second, the 3-AAP model is able to achieve good performance with extremely sparse distributions. The 3-AAP models achieve 98.4% correct classification as shown in Figure 3 with sparse distributions in which fewer than 3.7% of the 8000 possible triples are present in any sample. Likewise, the 99.4% correct classification results shown in Figure 4 is achieved with sparse distributions in which fewer than 7.9% of the 8000 possible triples are present in any sample. This suggests that only a small subset of the 8000 possible triples may actually be required for good classification performance.

Decimated 3-AAP Models

In order to estimate the number of triples that are significant for classifying genomes according to viral genome type, an analysis of variance was performed. A preliminary analysis of the data using the R statistics package (Ihaka and Gentleman, 1996; Dalgaard, 2002) indicated that the triple amino acid distributions do not satisfy normality assumptions. We then analyzed distributions for normality with the Kolmogorov-Smirnov test. The results confirmed the lack of normality. Of the 8000 triple amino acid distributions, only 8 are normal in all 5 viral genome types. Hence the Kruskal-Wallis test, a nonparametric analysis of variance (Neter *et al.*, 1996; Kruskal and Wallis, 1952; Zar, 1998) rather than ANOVA was performed. The Kruskal-Wallis test was used to establish which amino acids triples exhibit a statistical difference between viral genome types. Next, we used a multiple comparison procedure (Conover, 1998) based on the Kruskal-Wallis rank sums to find where the differences occur. Table 5 shows the number of significant triples for p -values from 10^{-5} down to 10^{-9} as determined by the multiple comparison procedure.

Number of Features in Decimated 3-AAP Models	
	Input Features
Full 3-AAP	8000
Decimated 3-AAP, $p < 10^{-5}$	1530
Decimated 3-AAP, $p < 10^{-6}$	927
Decimated 3-AAP, $p < 10^{-7}$	529
Decimated 3-AAP, $p < 10^{-8}$	310
Decimated 3-AAP, $p < 10^{-9}$	197

Table 5: Number of features in decimated 3-AAP models for different p -values compared to the full 3-AAP models.

Decimated 3-AAP distributions were constructed for p -values $< 10^{-5}$, 10^{-6} , 10^{-7} , 10^{-8} , and 10^{-9} by taking the original 3-AAP genome distributions and decimating them by deleting all triples except for those triples with p -values less than the selected cutoff and renormalizing. Next, decimated 3-AAP SVM models were trained from the decimated distributions. The classification performance of these decimated 3-AAP models was then evaluated by subjecting them to the same cross-validation, sequence bootstrap, and distribution bootstrap tests used for the full 3-AAP models. The performance of these decimated models is compared with that of the full 3-AAP models (8000 input features) in Figure 5. In this figure, the bootstrapped sequences are 660 amino acids long. The bootstrapped distributions are based on sampling 638 triples.

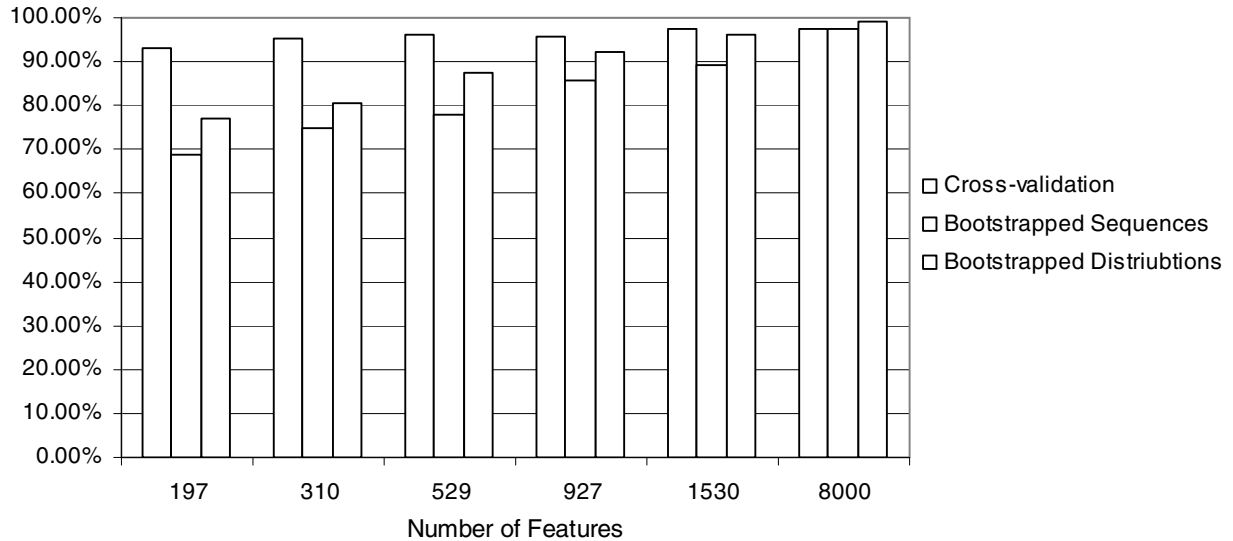


Figure 5: Summary of overall performance of the decimated 3-AAP models for different p -values compared to the full 3-AAP models.

The overall cross-validation performance of the decimated 3-AAP models shown in Figure 5 for p -values 0.01, 0.001, 0.0001, and 0.00001 compares well with the results achieved by the full 3-AAP models. In contrast there is a greater change in classification performance for bootstrapped sequences and bootstrapped distributions. This is an expected result. The cross-validation results are based on distributions derived from entire genomes and consequently are information rich. In contrast, the bootstrapped results are based on sparse distributions derived from sets containing only 630 amino acid triples.

Two inferences can be made from the results shown in Figure 5. First, the cross-validation results suggest that models derived from small subsets of the possible 8000 features (amino acid triples) are robust and able to discriminate between viral genome types provided that the query distributions are derived from large sample sets. Second, query distributions derived from a sparse sampling of a viral genome's distribution provide adequate information for determining genome type provided that the models are derived from a rich selection of distribution features.

Examples of Amino Acid Preference

Figure 6, shows the mean prevalence of five amino acid triples in the different viral genome types. These five triples were selected from the set of triples with p -value < 0.0001 . As can be seen in the figure, the mean prevalence for a given amino acid triple can differ significantly between pairs of viral genome type. For example, the triple RYF has a mean prevalence of roughly 7.29×10^{-6} for *retroid* genomes versus 2.79×10^{-4} for *double stranded DNA* genomes. Likewise, the triple SLI has a mean prevalence of roughly 1.03×10^{-3} for *single stranded RNA negative* genomes versus 2.34×10^{-4} for *single stranded DNA* genomes. Table 6 lists a subset of most significant amino acid triples selected by the multiple

comparisons test on the results of the Kruskal-Wallis test. The triples are listed by the pairs of genome type that they most strongly distinguish and *p*-value. A complete listing of amino acid triples with *p*-value < 0.01 is available with the supplementary material.

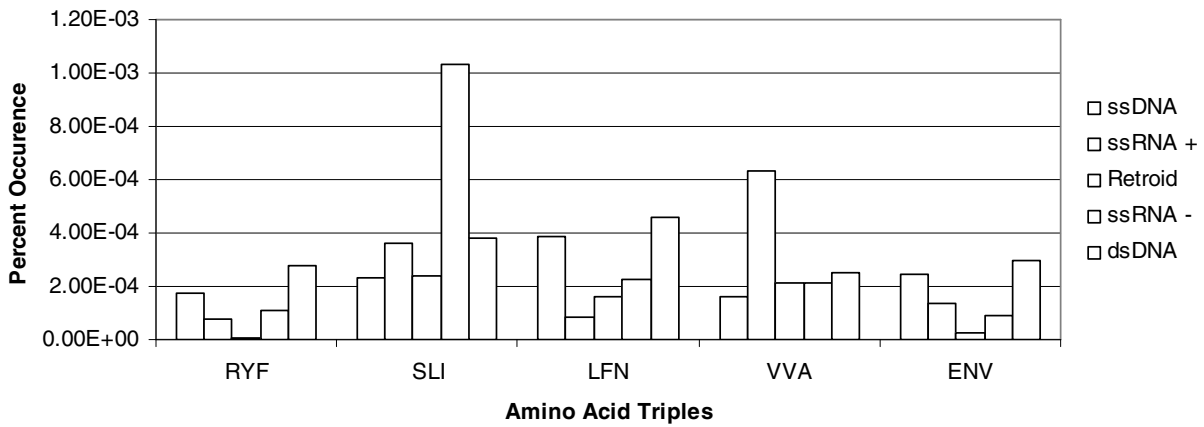


Figure 6: Example of prevalence of selected amino acid triples in the different viral genome types.

dsDNA vs Retroid $p < 10^{-12}$	dsDNA vs ssDNA $p < 10^{-6}$	dsDNA vs ssRNA- $p < 10^{-8}$	dsDNA vs ssRNA+ $p < 10^{-12}$	Retroid vs ssDNA $p < 10^{-6}$	Retroid vs ssRNA- $p < 10^{-9}$	Retroid vs ssRNA+ $p < 10^{-10}$	ssDNA vs ssRNA- $p < 10^{-7}$	ssDNA vs ssRNA+ $p < 10^{-6}$	ssRNA- vs ssRNA+ $p < 10^{-8}$
RYP	LLL	RHY	HFF	FND	SLI	VVA	LsI	GDD	ESL
MFF	LFL	NEY	HFC	FPF	IEG	TGV	TPW	VLA	PPG
ENV	VEV	CPA	CFE	SLY	SFF	LVT	GLI	KWW	LIQ
IYF	LDI	PDP	ECM	PLP	PPP	DTT	GGG	GVV	NLN
YFY	RLD	YCH	LFN	LLA	YLI	VVV	SLS	VGL	DLN
FYF	RLF	CCA	IYY	TNK	PGP	QLL	IDL	LAV	DHS
QMF	LFC	GDM	FYH	AST	TEF	PPS	LIA	AVV	DNL
HVE	RLL	IAW	CSY	KAI	EFE	MGQ	LND	GFL	LIK
FCE	LFK	RYC	IFF	QAL	IED	LYP	LSG	GRV	SLK
RVF	LSF	MPM	RYI		RRR	DVD	WNP	DQK	ILK
FFF	AFL	TPP	CMS		AEF	VVK	IGS	ALL	KLC
YPD	ARP	SHF	YEW		GDN	SVG	WHC	AAV	SKS
FWL	RNA				LES			DLG	IEI
DED	SAD				SDD			AVL	CQK
FYL	VRD				NGV			PTG	APT
NID	PRT								
LDG									

Table 6: Most significant amino acid triples as determined by a multiple comparisons test of Kruskal-Wallis rank sums.

Conclusion

Evidence suggests that structure in amino acid preference can be used to predict viral genome type. Three types of models that examine different degrees of structure in amino acid preference have been investigated. Of these, the model imposing the greatest structural constraints, the 3-AAP model, exhibits the best classification performance. Future investigation will address the observed local inhomogeneities in amino acid preference. The small differences in the sequence and distribution bootstrap results suggest that these inhomogeneities are of limited scope. A related issue is the evaluation of the robustness of our methods for assessing amino acid preference to sequence inaccuracy. Finally, this paper reports on the observation that amino acid distribution predicts viral genome type. The mechanism that has resulted in the observed relation between amino acid preference and viral genome type has yet to be investigated.

Acknowledgements

We thank Greg Harhay, Michael Huhns, Warren Snelling, Larry Stephens, Ralph Wiedmann, Jim Wray, and Jiangying Zhou for many fruitful discussions. The U.S. Department of Agriculture, Agricultural Research Service, supported this work under Cooperative Agreement No. 58-5438-2-341.

References

- Conover, W.J. (1998) Practical Nonparametric Statistics, 3rd Ed. John Wiley & Sons.
- Dalgaard, P. (2002) Introductory Statistics with R. Springer-Verlag, New York.
- Efron, B. (1983) Estimating the error rate of a prediction rule: Improvement on cross-validation. *J. Am. Stat. Assoc.*, **78**, 316-331.
- Efron, B. and Tibshirani, R. (1993) *An introduction to the bootstrap*. Chapman & Hall/CRC, 1994.
- Ihaka, R. and Gentleman, R. (1996) R: A language for data analysis and graphics. *J. Comput. Graph. Stat.*, **5**, 299-314.
- Joachims, T. (2002) Learning to Classify Text Using Support Vector Machines. *Dissertation*, Kluwer.
- Joachims, T. (1999) Making large-scale SVM Learning Practical. In Schölkopf, B., Burges, C., and Smola, A. (eds), *Advances in Kernel Methods - Support Vector Learning*, MIT Press, pp. 169-185.
- Kohavi, R. (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In Mellish, C.S. (ed) *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, San Mateo, CA, pp. 1137-1143.
- Kruskal, W.H. and Wallis, W.A. (1952) Use of ranks in one-criterion analysis of variance. *J. Am. Stat. Assoc.*, **47**, 583-621.
- Mjolsness, E. and DeCoste, D. (2001) Machine Learning for Science: State of the Art and Future Prospects. *Science*, **293**, 2051-2055.
- Neter, J., Kutner, M.H., Nachtsheim, C.J. and Wasserman, W. (1996), *Applied Linear statistical Models*, McGraw Hill.
- Smith, T.P., Grosse, W.M., Freking, B.A., Roberts, A.J., Stone, R.T., Casas, E., Wray, J.E., White, J., Cho, J., Fahrenkrug, S.C., Bennett, G.L., Heaton, M.P., Laegreid, W.W., Rohrer, G.A., Chitko-McKown, C.G., Perte, G., Holt, I., Karamycheva, S., Liang, F., Quackenbush, J., Keele, J.W. (2001) Sequence evaluation of four pooled-tissue normalized bovine cDNA libraries and construction of a gene index for cattle. *Genome Res.*, **11**, 626-630.
- Stone, M. (1974). Cross-validation choice and assessment of statistical predictions. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, **36**, 111-147.

Vapnik, V.N. (1998) *Statistical Learning Theory*, Wiley, New York.
Zar, J.H. (1998) *Biostatistical Analysis*. Prentice Hall, New Jersey.