Pacific Symposium on Biocomputing 12:355-366(2007)

ABSENT SEQUENCES: NULLOMERS AND PRIMES

GREG HAMPIKIAN

Biology, Boise State University, 1910 N University Drive
Boise, Idaho 83725, USA

TIM ANDERSEN

Computer Science, Boise State University, 1910 N University Drive
Boise, Idaho 83725, USA

We describe a new publicly available algorithm for identifying absent sequences, and demonstrate its use by listing the smallest oligomers not found in the human genome (human "nullomers"), and those not found in any reported genome or GenBank sequence ("primes"). These absent sequences define the maximum set of potentially lethal oligomers. They also provide a rational basis for choosing artificial DNA sequences for molecular barcodes, show promise for species identification and environmental characterization based on absence, and identify potential targets for therapeutic intervention and suicide markers.

1. Introduction

As large scale DNA sequencing becomes routine, the universal questions that can be addressed become more interesting. Our work focuses on identifying and characterizing absent sequences in publicly available databases. Through this we are attempting to discover the constraints on natural DNA and protein sequences, and to develop new tools for identification and analysis of populations. We term the short sequences that do not occur in a particular species "nullomers," and those that have not been found in nature at all "primes." The primes are the smallest members of the potential artificial DNA lexicon. This paper reports the results of our initial efforts to determine and map sets of nullomer and prime sequences in order to demonstrate the algorithm, and explore the utility of absent sequence analysis.

It is well known that the number of possible DNA sequences is an exponentially increasing function of sequence length, and is equal to 4^n, where n is the sequence length. This means that any attempt to assemble the complete set of unused sequences is hopeless. We have developed an approach that examines the minimum length sequences that are absent. These absent oligomers (nullomers and primes) occur at the boundary between the sets of natural and potentially unused sequences, and in part can be utilized to delineate the two sets[15]. By identifying the boundary nullomers surrounding the various branches of the phylogenetic tree of life, we hope to produce a map of the negative sequence space around each group. While the nullomer and prime sets will shrink as more sequences are reported, the mechanisms of mutation allow for rational predictions to be made about sequence evolution based on the accumulated nullomer data. The excluded sequences can be used for a number of purposes including:

1. Molecular bar codes
2. Species identification
3. Sequence specification for: RNAi, PCR primers, gene chips
4. Database verification and harmonization
5. Drug target identification
6. Suicide targets for recalling or eliminating genetically engineered organisms
7. Pesticide/antibiotic development
8. Environmental monitoring
9. Evolution studies

Our ultimate goal in studying nullomers, is to model and predict which bio-sequences (DNA, RNA and amino acid) are unlikely to be found in the biosphere. If "forbidden" sequences can be identified and confirmed through bioassays, this information will be foundational to understanding the basic rules governing sequence evolution. The insights gained could also greatly improve the theoretical foundation for comparative genomics, and provide an important conceptual framework for genetic engineering using artificial sequences.

2. Background

A naïve assumption of early genomic analysis was that sequence distribution over large genomes would approximate randomness. That is, a 6 base sequence would be found on average every 4^6 or 4096 bases. These types of assumptions were used for such calculations as the number of expected restriction enzyme recognition sites in a genome. But even early studies of genome organization using thermal melting and gradient centrifugation[8,13] showed that there is great non-uniformity in genomic sequences, particularly in warm-blooded vertebrates. What has emerged from many subsequent genome studies is a striking non-random distribution of certain large and short sequence motifs. Many of the described irregularities concern functional units of sequences.

For example, AGA codons are rare in bacterial genes, and when artificially substituted for synonymous codons they often have lethal consequences. This is believed to be due to ribosome stalling and the consequent early termination of protein synthesis. The reason for this effect is that while the codon chart tells us that AGA is one of the codons for the amino acid arginine, most bacteria preferentially use CGA to code for arginine. Even though the bacteria have the requisite tRNAs to use an AGA codon, these tRNAs are in such low concentration that the ribosome complex is destabilized while waiting for the t-RNA to load an arginine[6]. Examples of such "codon biases" have been seen in all species sequenced to date[20], and are a good example of the constraints on sequence evolution based on progenitor biases.

In eukaryotes too, many genomic features have been identified which skew the distribution of very short sequence motifs. For example, one of the authors (GH) was involved in research that examined the role of GG sequences in oxidative damage to DNA. It was found that when oxidizing agents captured electrons from DNA, the electron holes were transferred along DNA until they reached a GG sequence where they induced strand breakage[12]. Subsequent studies have borne out our hypothesis that GGG stretches are rare in coding regions, and other researchers have shown that "sentinel GGG" motifs found in non-coding introns serve as sacrificial sinks for oxidative damage[11]. Statistical studies using the autocorrelation function of Bernaola-Galván (2002) have shown that the human genome contains areas with GC-rich isochors displaying long-range correlations and scale invariance. Other studies have shown long range correlations between sequence motifs and regularly spaced structural features of the genome such as nucleosome binding sites[2,21].

All of these studies demonstrate what we would expect for a highly ordered information processing system: it is highly organized, non-random, and constrained by many factors, including the architecture of its storage and processing systems. Thus, even though DNA is passed on through dynamic evolving systems, there are still limits on its content, and some of these limits exist within large species groups. For example, any limits imposed by nucleosomal organization are applicable to all eukaryotic organisms; while bacteria which lack nucleosomal structure are immune to these constraints. This suggests one obvious use for our nullomer approach: the identification of molecular therapeutic targets that are present in the pathogen and absent in the host, or vise versa. Other constraints may be universal, since all organisms share a presumed origin, and many components of DNA function are highly conserved. By examining universally absent sequences (primes), we hope to discover insights into the most conserved mechanisms of molecular biology: inviolable rules which preclude these prime sequences.

Interestingly, the vast majority of bio-sequence analysis has ignored the exploration of absent sequences, instead focusing entirely on sequences that are either very rare, or very common. Some work has been done to characterize the expected number of missing words in a random text[19], however the primary focus of this research was the application of the result to the construction of pseudo-random number generators. One group has discussed the "absence versus presence" of short DNA sequences for the sake of identifying species[10], and another group has examined absent protein sequences[18]; but our approach is unique in that we are studying the set of smallest absent sequences (nullomers and primes) in order to discover basic rules of sequence evolution, and then apply this understanding for practical purposes such as drug development and the development of a DNA tagging system.

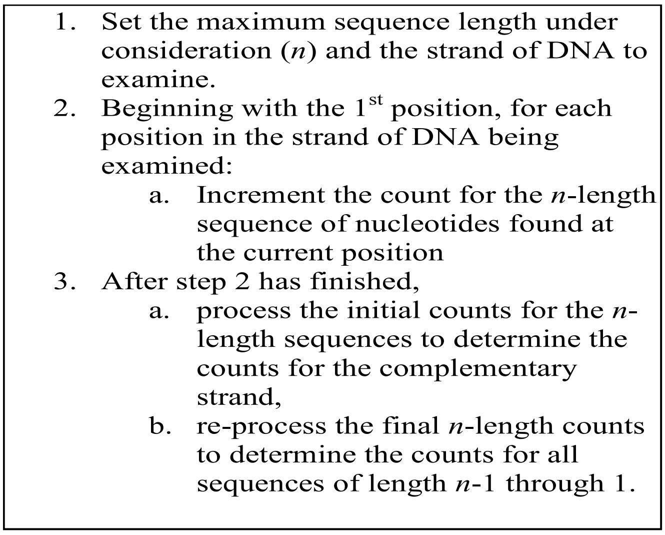
Our research stems from one of the primary assumptions of genomic analysis, that over and under-represented sequences are more likely to be interesting. While our work focuses on the novel area of absent oligomers, the general determination of over and under-represented sequences has received a great deal of attention[3,4,5,14,16,17,22]. For example, Nicodeme[16] developed a fast statistical approximation method for determining motif probabilities and demonstrated that over and under representation of protein motifs can be a good indicator of functional importance[17]. Stefanov[22] introduced a computationally tractable approach for determining the expected inter-site distance between pattern occurrences in strings generated by a Markov model. Bourdon and Vallee[5] and Flajolet[7] extended techniques to determine the likelihood and frequency of sequence motifs to generalized patterns, in particular patterns where the gap lengths between elements of the pattern in a random text are both bounded and unbounded. Amir et al.[1] generalize the notion of string matching further, developing statistical analysis techniques for a string matching approach they term structural matching. With this approach, the exact text of the strings is not important, rather, two strings are considered to match if some generalized relation between the two strings is satisfied.

3. Counting Sequences

We have developed a set of software utilities for counting sequences in a variety of sequence data. The main software package that we have created is SeqCount. This program has two primary functions. First, it counts the frequency of occurrence of all possible short sequences up to a user given maximum length in a set of sequence data and then writes this frequency count information to a file. Second, SeqCount determines the set of sequences that do not occur (nullomers) and writes these sequences to an additional set of files, one file for each sequence length being examined.

The algorithm used for counting sequences is shown in figure 1. The computational complexity of the algorithm is O(*mn*), where *m* is the maximum sequence length and *n* is the amount of DNA being processed. The algorithm can calculate the frequency of DNA sequences up to length 13 for the human genome (3 billion bases) in approximately 25 minutes on a single processor machine. The parallel version of the algorithm can process the human genome in less than 1 minute. A single pass through the entire set of DNA data downloaded from the NCBI web site takes approximately 12 hours.

In addition to SeqCount, we have created a number of secondary support tools for manipulating and understanding the data output by SeqCount. These support tools are available in both C and Java versions. Also, we have created a web-based interface to some of the data that we have generated with SeqCount. In particular, one can access the sequence counts and nullomer sets for several species for sequences up to length 13. Following is the full list of software packages and support tools that are available:



```
1. Set the maximum sequence length under
   consideration ($n$) and the strand of DNA to
   examine.
2. Beginning with the 1st position, for each
   position in the strand of DNA being
   examined:
      a. Increment the count for the $n$-length
         sequence of nucleotides found at
         the current position
3. After step 2 has finished,
      a. process the initial counts for the $n$-
         length sequences to determine the
         counts for the complementary
         strand,
      b. re-process the final $n$-length counts
         to determine the counts for all
         sequences of length $n$-1 through 1.
```

Figure 1. Algorithm for counting sequences.

- **SeqCount**: Given a set of genomic data in binary format, counts the total number of all sequences up to a user deter-mined length. The counts are saved in a single file. Additionally, if any sequences within the length given are not found, these sequences are output to a set of nullomer files (1 file for each nullomer length).
- **GBK2Bin**: Given a set of files in Genbank format, this pro-gram converts the files to a binary format wherein each DNA nucleotide is encoded as a 2-bit value. A single file is created for each contiguous sequence of DNA found in the genbank files, with the file name encoding the location of the sequence.

- **CountNulls**: Counts the number of nullomers in a nullomer file and prints the result.
- **Char2Null**: Converts any set of carriage return delimited sequences encoded in ascii format to the nullomer file format. This utility is typically used to take the piped output from either DiffNulls, IntNulls, UnionNulls, or ViewNulls and convert the ascii-based output of these files to binary format.
- **DiffNulls**: Takes as input 2 to many nullomer files and prints to the screen the set difference of the 1st nullomer file minus the union of the rest of the nullomer files.
- **IntNulls**: Takes as input 2 to many nullomer files and prints to the screen the set intersection of the nullomer files.
- **UnionNulls**: Takes as input 2 to many nullomer files and prints to the screen the set union of the nullomer files.
- **ViewNulls**: Takes as input 1 nullomer file and prints to the screen in ascii format the nullomers contained in the file.

SeqCount processes sequence data in a single pass, and has been optimized for speed of processing. SeqCount can be executed in either parallel mode on a Beowulf cluster or in sequential mode on a single workstation. In sequential mode the program is limited to counting sequences up to length 13. When the program is executed in parallel mode and the user requests the program to count sequences of length greater than 13, the program evenly divides the sequence space up amongst the available processors and then each process is responsible for counting sequences that occur within its assigned sequence space. At the end of processing the counts from each process are collected and written to a file as in the sequential version. The software packages, documentation, and web-based interface can be freely accessed at:
http://trac.boisestate.edu/bioinformatics/nullomers.

4. Results

We have downloaded the entire sequence database from the NCBI web site and used our algorithms to determine the nullomer sequences for several fully sequenced organisms: chimpanzee, human, etc. These results are given in section 4.1. We have also processed all of the data in the entire DNA sequence database and determined the “prime” DNA sequences (sequences that do not occur in any of the data), and these results are given in section 4.2. In addition, we have processed the entire protein database and also give these results in section 4.2

4.1. *Nullomers – fully sequenced organisms*

Table 1 gives the number of DNA nullomers found at lengths 8 through 13 for several different organisms. The results for bacteria, fungi, and yeast are across all sequenced organisms.

Table 1. Number of DNA nullomers at sequence length 8 through 13.

	8	9	10	11	12	13
arabid			107	23646	1167012	20237388
bacteria					541	562870
c_elegans			2	7686	1152038	23339534
chicken			2	590	131515	4722702
chimp				136	45938	2426474
cow				96	45060	2432554
dog				40	25217	1868964
fruitfly				206	221616	12399300
human				**80**	**39852**	**2232448**
mouse				178	54383	2625646
rat				50	30708	1933220
zebrafish				2	15561	2469558

Table 2 shows how the nullomer sets of each of the organisms given in table 1 intersect with each other. The names of the organisms are listed in the first column. The 2nd through 4th column show the actual size of each intersection for lengths 11 through 13. The 5th through 7th column show the expected size (with the assumption that each set was independently and randomly generated), and the 8th through 10th column give the ratio of the actual/expected. For the ratio, numbers greater than 1 indicate the degree to which the intersection is larger than expected. The results are sorted in descending order on the ratio value at length 12.

Table 2. Intersection of human nullomers with the nullomers of other organisms.

	actual size			expected size			ratio		
	11	12	13	11	12	13	11	12	13
chimp	28	19581	1521778	0.002594	109.1195	80719.25	10794.16	179.4455	18.85273
dog	0	4963	731372	0.000763	59.89956	62173.08	0	82.85536	11.76348
rat	8	5975	734566	0.000954	72.94269	64310.63	8388.608	81.91363	11.42216
cow	0	7314	886544	0.001831	107.0339	80921.51	0	68.33348	10.9556
mouse	2	8765	927076	0.003395	129.1794	87344.92	589.0876	67.85136	10.61397
chicken	4	10946	1162632	0.011253	312.396	157105.7	355.4495	35.03886	7.400316
zebrafish	0	1080	504532	3.81E-05	36.96304	82152.48	0	29.21837	6.141409
fruitfly	0	2122	761094	0.003929	526.4187	412476	0	4.031012	1.845184
arabid	0	9521	1325550	0.451012	2772.079	673218.3	0	3.434607	1.968975
c_elegans	0	8378	1273344	0.146599	2736.51	776414.5	0	3.061564	1.640031
bacteria	0	0	24242	0	1.285072	18724.47	0	0	1.294669

Human and chimp have the greatest intersection between their absent sequences, and mammals in general show a much stronger intersection with human than the other listed organisms. While this is intuitively satisfying, further studies will be required to demonstrate if nullomer sets can be used to corroborate phylogenetic relationships among species.

4.2. *Human Genome nullomers*

Other researchers have reported absent sequences as a part of large scale analysis[9], however, as far as we know this is the first publication of an actual list of human nullomers. Our results also differ from earlier reports of 44 absent 11-mers, in that we have found 43 sequences and their compliments which are not found in the two published human genomes (Table 3). Of these sequences, 4 11-mers and their complements currently have no sequence match in any reported human sequence in GenBank as determined by BLAST.

Table 3. Human nullomers at length 11.

Human BLAST matches	Nullomer	Human BLAST matches	Nullomer
0	**cgctcgacgta**	3	cgcgcataata
0	**gtccgagcgta**	3	cgacggacgta
0	**cgacgaacggt**	3	cgaatcgcgta
0	**ccgatacgtcg**	3	cggtcgtacga
1	tacgcgcgaca	3	gcgcgtaccga
1	cgcgacgcata	3	cgcgtaatcga
1	tcggtacgcta	3	cgtcgttcgac
1	tcgcgaccgta	3	ccgtcgaacgc
1	cgatcgtgcga	3	acgcgcgatat
1	cgcgtatcggt	3	cgaacggtcgt
2	cgtcgctcgaa	3	cgcgtaacgcg
2	tcgcgcgaata	3	ccgaatacgcg
2	tcgacgcgata	3	catatcgcgcg
2	atcgtcgacga	4	cgcgacgttaa
2	ctacgcgtcga	4	gcgcgacgtta
2	cgtatacgcga	4	ccgacgatcgt
2	cgattacgcga	4	ccgttacgtcg
2	cgattcggcga	5	ccgcgcgatat
2	cgacgtaccgt	6	ccgacgatcga
2	cgacgaacgag	7	cgaccgatacg
2	cgcgtaatacg	20	cgaatcgacga
2	cgcgctatacg		

We are presently searching the available single nucleotide polymorphism (SNP) databases, to determine which if any of the nullomers are associated with known SNPs.

4.3. *Primes – all sequence data*

We have also used our algorithms to process the entire DNA sequence database available from NCBI, and found that length 15 is the shortest length at which primes (absent sequences) are found. At this length there are 60370 primes that are not found in any of the DNA sequence data. These sequences can be referenced through our web site at http://trac.boisestate.edu/bioinformatics/nullomers.

We have also processed all available protein sequences, and identified 1799 primes of length 5. It should be noted that this number is significantly less than the 12,080 “zero count pentats” that were reported by Otaki et al. in 2004[18]. In that paper, the researchers cloned 6 of their zero count pentats and showed that they were not lethal when expressed in *E.coli*. But we found (using our algorithm) that 5 of the 6 “zero-count” oligomers are actually presently listed in GenBank. This discrepancy is likely due to the addition of new protein data at NCBI since the zero count search was performed in September of 2003. This demonstrates the need for continued processing of this data, and the utility of our web-available program for conducting immediate absent sequence inventories. We believe that the approach taken by Otaki et al.[18] is a valuable first step in examining the potential lethality of absent sequences. As the number of such sequences shrinks, and large scale expression projects become more routine, the fitness effects of nullomers and primes can be studied more systematically.

The fact that the amino and nucleotide primes both presently represent a maximum of 5 amino acids (15 nucleotide bases in the DNA database, and 5 amino acids in the protein database) is coincidental. We examined all possible coding sequences for the 1799 length-5 protein primes, and did not find any intersection with the DNA primes at length 15. The nucleotide sequences include coding and non-coding DNA, while the protein database has only expressed (and hypothetically expressed) sequences. Thus it is likely that most nucleotide sequences representing codons for absent amino acid sequences are found only in non-coding regions of DNA. We are presently exploring the intersection of amino acid nullomers, and DNA nullomers in coding regions, and will report those results separately.

On average, the protein primes had about half as many possible DNA coding sequences as expected for peptides of their length, which indicates that the set of protein primes is biased towards those protein sequences that have fewer DNA coding options. We found 5 protein primes that have a single DNA coding sequence – MWMWW, MWWWW, WMMWM, WMWWW, and WWMMW. We then performed a BLAST search for short, exact matches to each of these DNA coding sequences and examined the results. Each DNA sequence yielded a number of exact matches. Most of these matches were in intron specific regions, however, several of the matches occurred in putative coding regions. We are currently working to resolve each of these database discrepancies. The identification of apparent discrepancies between protein and nucleotide primes in coding regions, demonstrates the utility of the nullomer approach as tool for harmonizing the various biomolecular databases.

5. Conclusion and Discussion

We have developed a series of tools for the identification and study of absent sequences. Using these tools we have made publicly available the full set of amino acid and nucleotide primes (the shortest sequences not found in their respective databases.) In order to allow creative extensions of our approach, the software packages, documentation, and web-based interface can be freely accessed at: http://trac.boisestate.edu/bioinformatics/nullomers. In this paper we demonstrate some of the uses of these tools, and the elegance of the nullomer approach.

It should be noted that nullomer searches have corollaries in the natural world, most notably in the development of the human immune system. During embryonic development a large variety of antigen recognizing cells are generated by the random rearrangement of DNA cassettes coding for the "variable" segments of antibody producing cells. This DNA shuffling results in the incredible diversity of immune cells which produce molecular soldiers that each recognize a single small oligomer (peptides, lipids, sugars or nucleotide). This army is reduced by a colossal "deselection" in the embryonic thymus. Here, any immune cell which finds its target among the "self" molecules is culled from the army. In essence, what is left is a sentinel army of nullomer hunters. They recognize and destroy only absent oligomer sequences. When an adult immune cell detects its particular nullomer, it is stimulated to reproduce, and sometimes to hypermutate in order to recognize related nullomers. Thus the natural defense system of the body is based on recognizing nullomers, and anticipating oligomers that may arise from them. This type of approach would be useful in any intelligent response to novel biological threats, natural or manmade. For example, nullomer detection in environmental samples could indicate the introduction of novel natural or engineered species. The rapid response to such a potential threat should include the generation of agents to detect and possibly incapacitate related novel molecules.

The absent sequences that we report here represent the largest possible set of artificial oligomers. Within this dynamic, shrinking set will be found all lethal oligomers, if any exist. These small molecules may prove to be powerful bioactive compounds which act in a species-specific or group-specific manner. Within the set of primes, there is even the possibility of a pan-lethal agent which could function as a sterilant, or suicide gene for therapeutic and biocontrol applications.

We have also shown that nullomer searches can be used to assess the harmony of molecular databases (nucleotide and protein), and to identify potential therapeutic targets that exist in a pathogenic species but not its host. The nullomer approach may also be useful for studying genome relationships, in that the absent oligomers (nullomers) are more similar in closely related species, than in those more distantly related.

Finally, it is easy to construct artificial tags of DNA or amino acids that have not been reported in GenBank. But identifying the *smallest* oligomers that have not been found in a species or group of species, provides the first rational basis for the construction of an artificial DNA lexicon. By devising tags based on nullomers and primes, more efficient and elegant artificial sequences can be constructed. These sequences can be used to identify artificial constructs, tag them with identifying characteristics, or even code for suicide genes in order to "recall" a genetically engineered product.

Acknowledgements: The authors wish to thank the following people for their help: Dr. Amit Jain and Ben Noland for assistance with the Beowulf cluster and initial algorithms; Barry Hall, Jim Smith and Ken Cornell for comments and criticism about the nullomer approach; Jim Munger for his encouragement and support; and the anonymous reviewers who provided valuable feedback.

References

1. Amir, A., Cole, R., Hariharan, R., Lewenstein, M., & Porat, E. (2003). Overlap Matching. Inf. Comput. 181(1), 57-74.
2. Audit B, Vaillant C, Arneodo A, d'Aubenton-Carafa Y, Thermes C. (2002) Long-range correlations between DNA bending sites: relation to the structure and dynamics of nucleosomes. J Mol Biol. 2002 Mar 1;316(4):903-18.
3. Apostolico, A., M. Bock., and S. Lonardi. (2002). Monotony of Surprise and Large-Scale Quest for Unusual Words. Proceedings of the sixth annual international conference on Computational biology, pp22-3.
4. Apostolico, A., Gong, F., and Lonardi, S. "Verbumculus and the Discovery of Unusual Words", Journal of Computer and Science Technology, vol.19, no.1, pp.22-41, 2004.
5. Bourdon, J. & Vallee, B. (2002). Generalized Pattern Matching Statistics. In Mathematics and Computer Science, II Versailles, 249-265.
6. Cruz-Vera LR, Magos-Castro MA, Zamora-Romo E, Guarneros G (2004) Ribosome stalling and peptidyl-tRNA drop-off during translational delay at AGA codons. Nucleic Acids Res. 2004 Aug 18;32(15):4462-8.
7. Flajolet, P., Guivarc'h, Y., Szpankowski, W., & Vallée, B. (2001). Hidden Pattern Statistics. ICALP 2001, 152-165.
8. Filipski, J. (1987). Correlation between molecular clock ticking, codon usage fidelity of DNA repair, chromosome banding and chromatin compactness in germline cells. FEBS Lett. 217: 184-186.
9. Fofanov Y, Luo Y, Katili C, Wang J, Y. B, Powdrill T, Fofanov V, Li T-B, Chumakov S, Pettitt BM (2003) How independent are the appearances of n-mers in different genomes? Bioinformatics, vol. 20, no. 15, pp2421-2428.
10. Fofanov V., Fofanov Y., Pettitt B. (2002). Counting array algorithms for the problem of finding appearances of all possible patterns of size n in a sequence. In The 2002 Bioinformatics Symposium, Keck/GCC

Bioinformatics Consortium, p 14. W.M. Keck Center for Computational and Structural Biology, Houston Texas.

11. Friedman K, Heller A (2001) On the Non-Uniform Distribution of Guanine in Introns of Human Genes: Possible Protection of Exons against Oxidation by Proximal Intron Poly-G Sequences. J. Phys. Chem. B, 105 (47), 11859 -11865, 2001. 10.1021/jp012043n S1089-5647(01)02043-0.
12. Henderson P.T., Jones D., Hampikian G., Kan Y., Schuster G.B. (1999) Long distance charge transport in DNA: the phonon-assisted polaron-like hopping mechanism. Proc. Natl Acad. Sci. USA. 1999;96:8353–8358.
13. Inman, R.B. (1966). A denaturation map of the l phage DNA molecule determined by electron microscopy. J. Mol. Biol. 18: 464-476.
14. Leung, M. Y., Marsh, G. M., and Speed, T. P. (1996). Over and underrepresentation of short DNA words in herpesvirus genomes. J. Comput. Bio. 3, 345-360.
15. Mitchell, T. (1997) *Machine Learning.* New York: McGraw Hill.
16. Nicodeme, P. (2001). Fast approximate motif statistics. Journal of Computational Biology, 8(3), 234-248.
17. Nicodème, P., Doerks, T., & Vingron, M. (2002). Proteome Analysis Based on Motif Statistics. Bioinformatics, vol. 18, 161—171.
18. Otaki J, Ienaka S, Gotoh T, and Yamamoto H. (2005) Availability of short amino acid sequences in proteins. Protein Science, 14:617-625.
19. Rahmann, S. & Rivals, E. (2000). Exact and Efficient Computation of the Expected Number of Missing and Common Words in Random Texts. CPM 2000, 375-387.
20. Reis, M., Savva, R. & Wernisch, L. (2004) Solving the riddle of codon usage preferences: a test for translational selection. Nucleic Acids Res. 32: 5036-5044.
21. Segal, E., Y. Fondufe-Mittendorf, L. Chen, A. Thastrom, Y. Field, I. K. Moore, J. Z. Wang, J. Widom. (2006) A Genomic Code for Nucleosome Positioning. Nature, 2006 July, 442(7104):772-8.
22. Stefanov, V. (2003). The intersite distances between pattern occurrences in strings generated by general discrete and continuous-time models: an algorithmic approach. Journal of Applied Probability. 40, no. 4, 881–892.