

METHOD FOR PREDICTION OF SINGLE NUCLEOTIDE POLYMORPHISMS

Field of the Invention

[0001] The present invention relates to computational methods for predicting single nucleotide polymorphisms (SNP) and more particularly to the methods of using learning machines including support vector machine (SVM) and artificial neural network (ANN) for predicting SNPs.

Background of the Invention

[0002] Genomic mutation is seen as an integrated part of life. The mutation in a genome can be caused by many factors either alone or in combination and be effected in many forms including single nucleotide polymorphisms (SNPs), deletion, insertion, and recombination. It is known that certain mutations are associated with diseases. Moreover, the nature, frequency and location of gene mutations causing human genetic diseases are non-random and determined in part by local DNA sequence environment. Thus, an understanding of the association between genomic mutations and diseases is beneficial in improving diagnosis, treatment and prevention of the diseases.

[0003] It is generally agreed that SNPs are one kind of genomic mutations, while there is no consensus among scientific community about a specific definition of SNPs. In some occasions, SNPs are defined as germ line point mutations in the global human population. Applicants, however, prefer that SNPs be defined as the variation of a single base pair in the human genomes. The variation can be a deletion, an insertion, or a substitution, and occurs every 100-1000 base pairs. Straight sequencing is the historical method of discovering the SNPs. Lately, upon the completion of a draft human genome sequence, numerous SNP-hunting projects have emerged to link single base variation to disease using DNA sequencing, such as Celera's SNP database, the SNP Consortium, and work done by the National Human Genome Research Institute. It is noted that not all SNPs have harmfully phenotypic effects. For instance, the SNPs in genes encoding blood types effect no harmful phenotypes. Nonetheless, SNPs have been linked to diseases. For example, a single adenine to thymine

point mutation in the sixth amino acid of the beta globin causes sickle-cell anemia. Discovered mutations may relate to disease susceptibility either through direct association, where the allele has a deleterious effect on fitness and will be found at a higher frequency in a disease population versus an unaffected population, or indirect association, where the variant is a member of a set of alleles in linkage disequilibrium with another allele known to be causative of disease. As discussed above, identification of the association of SNPs with diseases will enable us to improve diagnosis, seek better treatment and take preventive measures.

[0004] There are more reasons why the search for SNPs has become so important within the last few years. For instance, as previously mentioned, SNPs occur very frequently within the genome. The more common the genetic marker is, the higher the likelihood of finding significant differences among individuals. In addition, the mutation rate of SNPs is low from generation to generation. This allows scientists to conduct more accurate population studies where the goal is to map gene location.

[0005] Identification of SNPs has been hampered for lack of economic and efficient methods. As discussed above, straight sequencing is the historical method of identifying SNPs. While being highly accurate, it is time-consuming and expensive for large-scale identification of SNPs in the human genome. Computational techniques have been used to aid the identification of SNPs. For example, Polybayes has been applied to EST database to discover new SNPs. This method is based on pure sequence alignment. (Liu, et al., *In-Silico* Identification and Classification of SNPs in Sorghum EST Sequences, Poster presentation for 2003 Genome Informatics Meeting at Cold Spring Harbor). Another SNP prediction program is also based on EST sequence alignment. (SEAN SNP Prediction and Display Programs, <http://zebrafish.doc.ic.ac.uk/Sean/sean.htm>) Obviously, the sequence alignment programs are not adequate for predicting SNPs in the human genome because no database contains multiple copies of genomic sequences. Furthermore, US Patent Application Publication 2003/0138778 discloses an apparatus, system and method for predicting SNPs. The method disclosed in USPA '778 includes steps for obtaining a variation predictiveness matrix and predicting one or more single nucleotide variations of a nucleic acid sequence based on the variation predictiveness matrix. The invention disclosed is based on the statistical analysis of DNA variation patterns and uses that statistical analysis to identify disease-causing mutations from sequence context.

[0006] In recent years, machine-learning approaches for data analysis have been widely explored for recognizing patterns which, in turn, allow extraction of significant information contained within a large data set that may also include data consists of nothing more than irrelevant details. Learning machines comprise algorithms that may be trained to generalize using data with known outcomes. Trained learning machine algorithms may then be applied to predict the outcome in cases of unknown outcome, i.e., to classify the data according to learned patterns. The common learning machines include SVM and ANN. Machine-learning approaches including kernel-based classifiers such as support vector machines are ideally suited for domains characterized by the existence of large amounts of data, noisy patterns and the absence of general theories. Support vector machines are disclosed, for example, in U.S. Patents Nos. 6,128,608 and 6,157,921, which are incorporated herein in their entireties.

[0007] SVM is an effective method for general purpose supervised pattern recognition (Vapnik 1995, 1998). SVM has been applied successfully to biological data recently, including the identification of unknown genes using the gene expression data from DNA microarray hybridization experiments (Brown et al., Knowledge-based analysis of microarray gene expression data by using support vector machines, PNAS, 97: 262-7, 2000), and prediction of protein structural class (Cai et al., Support Vector Machine for predicting protein structural class, BMC Bioinformatics, 2:3, <http://www.biomecentral.com/1471-2105/2/3>, 2001). In addition, SVM was also used for searching translation initiation sites (Zien et al., Engineering Support Vector Machine kernels that recognize translation initiation sites, Bioinformatics, 2000 Sep; 16(9): 799-807) and for splice site recognition (Sonnenburg et al., New methods for splice site recognition, Proceedings of the International on Artificial Neural Networks, 2002).

[0008] An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well. ANN has been applied to certain bioinformatics problems including protein configuration (Dietmann and

Frommel, Prediction of 3D neighbours of molecular surface patches in proteins by artificial neural networks, *Bioinformatics* 18: 167-174, 2002), and prediction of translation initiation sites (Pedersen and Ni, Neural network prediction of translation initiation sites in eukaryotes: perspectives for EST and genome analysis, *ISMB*, 5:226-33, 1997).

[0009] Therefore, there is an existing need for developing computer programs that are capable of predicting SNPs in genomic sequences by learning from the known SNPs and non-SNPs. The SNP prediction can be used to reduce the time and cost for SNP discovery by predicting sequences that are likely to contain SNPs. The present invention satisfies all of the needs.

Summary of the Invention

[0010] The present invention relates to computational methods for predicting of single nucleotide polymorphism (SNP). More specifically, the methods utilize learning machine algorithms for predicting of SNPs in polynucleotide sequences. The methods comprise steps of generating training sets of positive and negative SNPs; calculating one or more parameters of each member of the training sets of positive and negative SNPs; training a learning machine algorithm with the one or more parameters of members of the training sets of positive and negative SNPs; and testing the trained learning machine algorithm with test sets of positive and negative SNPs; thereby the best prediction results are recorded by using any of the one or more parameters. The parameters of positive and negative SNPs used for the learning machine algorithms include free energy change (ΔG), entropy (ΔS), enthalpy (ΔH), melting temperature, GC content, SNP distribution score information, sequence information, gene information, intron information and exon information; wherein each parameter is calculated by its own rules.

[0011] Accordingly, one object of the present invention is to provide a learning machine algorithm-based method to predict SNPs, so that the method reduces the time and cost for SNP discovery by predicting sequences that are likely to contain SNPs.

[0012] The objectives and advantages of the invention will become apparent from the following detailed description of preferred embodiments thereof.

Detailed Description of the Invention

[0013] A "polymorphism" is a variation in the DNA sequence of some members of a species. The genomes of animals and plants naturally undergo spontaneous mutation in the course of their continuing evolution (Gusella, J. F., *Ann. Rev. Biochem.* 55:831-854 (1986)). In some instances, the mutated sequence and the initial sequence co-exist in stable or quasi-stable equilibrium. In other instances, the mutation confers a survival or evolutionary advantage to the species, and accordingly, it may eventually be incorporated into the genome of every member of that species.

[0014] A polymorphism is thus said to be "allelic," in that, due to the existence of the polymorphism, some members of a species may have the unmutated sequence (i.e. the original "allele") whereas other members may have a mutated sequence (i.e. the variant or mutant "allele"). In the simplest case, only one mutated sequence may exist, and the polymorphism is said to be diallelic. Diallelic polymorphisms are the most common and the preferred polymorphisms of the present invention. The occurrence of alternative mutations can give rise to triallelic, etc. polymorphisms. An allele may be referred to by the nucleotide(s) that comprise the mutation.

[0015] The present invention is directed to a particular class of allelic polymorphisms, and to their use in genotyping a plant or animal. Such allelic polymorphisms are referred to herein as "single nucleotide polymorphisms," or "SNPs." "Single nucleotide polymorphisms" are defined by the following attributes. A central attribute of such a polymorphism is that it contains a polymorphic site, "X," most preferably occupied by a single nucleotide, which is the site of variation between allelic sequences. A second characteristic of an SNP is that its polymorphic site "X" is preferably preceded by and followed by "invariant" sequences of the allele. The polymorphic site of the SNP is thus said to lie "immediately" 3' to a "5'-proximal" invariant sequence, and "immediately" 5' to a "3'-distal" invariant sequence. Such sequences flank the polymorphic site. As used herein, a sequence is said to be an "invariant" sequence of an allele if the sequence does not vary in the population of the species.

[0016] Since genomic DNA is double-stranded, each SNP can be defined in terms of either strand. Thus, for every SNP, one strand will contain an immediately 5'-proximal invariant sequence and the other will contain an immediately 3'-distal invariant sequence. In the preferred embodiment, wherein a SNP's polymorphic site, "X," is a single nucleotide, each strand of the double-stranded DNA of the SNP will contain both an

immediately 5'-proximal invariant sequence and an immediately 3'-distal invariant sequence.

[0017] Although the preferred SNPs of the present invention involve a substitution of one nucleotide for another at the SNP's polymorphic site, SNPs can also be more complex, and may comprise a deletion of a nucleotide from, or an insertion of a nucleotide into, one of two corresponding sequences. For example, a particular gene sequence may contain an A in a particular polymorphic site in some animals, whereas in other animals a single or multiple base deletions might be present at that site. Although the preferred SNPs of the present invention have both an invariant proximal sequence and invariant distal sequence, SNPs may have only an invariant proximal or only an invariant distal sequence.

[0018] The SNPs of the present invention can be used to analyze the DNA of any plant or animal including humans, non-human primates, domestic animals (such as dogs, cats, etc.), farm animals (such as cattle, sheep, etc.) and other economically important animals.

[0019] SNPs occur at high frequency and uniformity, thus, making SNPs extremely valuable. For example, if a particular trait (e.g. predisposition to cancer) reflects a mutation at a particular locus, then any polymorphism that is linked to the particular locus can be used to predict the probability that an individual will be exhibiting that trait. The value of such a prediction is determined in part by the distance between the polymorphism and the locus. Since the SNPs of the present invention are present approximately once every 300 bases in the mammalian genome, and exhibit uniformity of distribution, a SNP can, statistically, be found within 150 bases of any particular genetic lesion or mutation.

[0020] SNPs are more stable than other classes of polymorphisms. Their spontaneous mutation rate is approximately 10^{-9} . SNPs permit a high degree of genetic resolution of identity, paternity exclusion, and analysis of an animal's predisposition for a particular genetic trait. SNPs reflect the highest possible definition of genetic information-nucleotide position and base identity.

[0021] Statistical analysis of SNPs can be used for any of a variety of purposes. For instance, the SNPs predicted by the present invention can be used to predict the statistical probability that two individuals will have the same alleles for any given polymorphisms. Where a particular animal has been previously tested, such testing can be

used as a "fingerprint" with which to determine if a certain animal is, or is not that particular animal. Moreover, the SNPs can be used to exclude paternity of a male for a particular individual (such as a stallion's paternity of a particular foal), or to assess the probability that a particular individual is the progeny of a selected female (such as a particular foal and a selected mare). In addition, the SNPs can be used for the construction of a genetic map of a target species. Thus, the particular array of polymorphisms identified by the methods of the present invention can be correlated with a particular trait, in order to predict the predisposition of a particular animal (or plant) to such genetic disease, condition, or trait. As used herein, the term "trait" is intended to encompass "genetic disease," "condition," or "characteristics." The term, "genetic disease" denotes a pathological state caused by a mutation, regardless of whether that state can be detected or is asymptomatic. A "condition" denotes a predisposition to a characteristic (such as asthma, weak bones, blindness, ulcers, cancers, heart or cardiovascular illnesses, skeletal-muscular defects, etc.). A "characteristic" is an attribute that imparts economic value to a plant or animal. Examples of characteristics include longevity, speed, endurance, rate of aging, fertility, etc. The methods for identification and parentage verification and genetic trait analysis are known to those skilled in the art. For example, U.S. Pat. App. 2003/0170624 is incorporated in its entirety.

[0022] SNPs can be characterized using any of a variety of experimental methods. Such methods include direct or indirect sequencing of an interested target, use of restriction enzymes where the respective alleles of the site create or destroy a restriction site, use of allele-specific hybridization probes, use of antibodies that are specific for the proteins encoded by the different alleles of the polymorphism, or by other biochemical interpretation. While all of the experimental methods provide accurate identification of SNPs, they are expensive and time-consuming, thus unsuitable for large scale of identification of SNPs. Therefore, an accurate prediction of the existence of SNPs in an interested sequence will save money and time.

[0023] Learning machines are ideal for SNP prediction. While several learning machines exist and advancements are expected in this field, certain embodiments of the present invention focus on kernel-based learning machines and more particularly on the support vector machine (SVM). SVM has many applications including Text Classification (Tong & Koller, Support vector machine active learning with applications to text

classification, *Journal of Machine Learning Research*, 2: 45-6, 2001), and Face Detection (Osuna et al., Training support vector machines: an application to face detection, *IEEE Proceeding of Computer Vision and Pattern Recognition*, 97:130-136, 1997).

[0024] SVM is a learning algorithm that was developed by Vladimir N. Vapnik. The concepts underlying the SVM are explained in detail in his book, entitled *Statistical Learning Theory* (John Wiley & Sons, Inc. 1998), which is herein incorporated by reference in its entirety. Accordingly, a familiarity with SVMs and the terminology used therewith are presumed throughout this specification. The basic idea behind SVM is the formation of a linear classifier to separate the positive and negative training data. This linear classifier is then used on the test data. Depending on the position of the point falling on the positive or negative side of the linear classifier, the prediction result is calculated. In real world situation, a linear classifier may not be able to separate the positive and negative data sets. To overcome this limitation, kernels are used to transform the data into a feature space where the linear classifier can be used for the separation. Choosing the right kernel to use is therefore essential for the separation in the feature space.

[0025] An SVM implements specialized algorithms for providing generalization when estimating a multidimensional function from a limited collection of data. An SVM may be particularly useful in solving dependency estimation problems. More specifically, an SVM may be used accurately in estimating indicator functions (e.g., pattern recognition problems) and real-valued functions (e.g., function approximation problems, regression estimation problems, density estimation problems, and solving inverse problems). See, U.S. Pat. No. 5,950,146, titled "Support Vector Method For Function Estimation", whose inventor is Vladimir Vapnik, and is hereby incorporated by reference in its entirety as though fully and completely set forth herein. U.S. Pat. No. 5,649,068, titled "Pattern Recognition System Using Support Vectors", whose inventors are Bernard Boser, Isabelle Guyon, and Vladimir Vapnik, and is hereby incorporated by reference in its entirety as though fully and completely set forth herein

[0026] A full coverage of the use of SVM for pattern recognition is given by Vert (Vert, 2001), but some basic concepts are introduced here. It is assumed that there are a series of examples (or input vectors) $\bar{x}_i \in R^m$ ($i = 1, 2, \dots, N$) with corresponding labels $y_i \in \{+1, -1\}$ ($i = 1, 2, \dots, N$). In the case of SNP, the input corresponds to the feature parameters and y_i (+1 or -1) represents SNP/non-SNP. This task is carried out by (i)

mapping of the input vectors into a high dimensional feature space and (ii) construction of an optimal separating hyperplane (OSH) in the new feature space. The OSH is the hyperplane with the maximum distance to the nearest data points of each class in the feature space H . One of the most central points in using SVM is the choice of mapping ϕ , which is defined by a kernel function $K(\bar{x}_i \bar{x}_j)$. The decision function used by SVM can be written:

$$f(x) = \text{sgn}\left(\sum_{i=1}^N y_i \alpha_i \cdot K(\bar{x}_i \bar{x}_j) + b\right) \quad (1)$$

The coefficients α_i are given by the solution of the quadratic programming task: Maximize

$$\sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j \cdot y_i y_j \cdot K(\bar{x}_i \bar{x}_j)$$

subject to

$$0 \leq \alpha_i \leq C \quad (2)$$

$$\sum_{i=1}^N \alpha_i y_i = 0, i = 1, 2, \dots, N$$

C in equation (2) is a parameter controlling the trade off between the margin and the training error. It is the kernel function that determines the dimension of the feature space, meaning that different kernels will represent the input vectors in different ways. The aim of the SVM is then to find an OSH without losing the ability of generalization, often referred to as over-training. The kernels tested for SNP predictions are linear, polynomial and radial basis function. Equation (3) is an example of the radial basis function and equation (4) shows the polynomial kernel function.

$$K(\bar{x}_i \bar{x}_j) = \exp(-\gamma \|\bar{x}_i - \bar{x}_j\|^2) \quad (3)$$

$$K(\bar{x}_i \bar{x}_j) = \left(\bar{x}_i \bullet \bar{x}_j + 1\right)^d \quad (4)$$

[0027] Since its introduction in the mid-nineties the SVM method has quickly become one of the most popular learning algorithms and is currently applied to more and more real-life classification problems. The main reason for its popularity is that it is very efficient compared to other methods in many real-world applications (Hua, 2001).

[0028] Being efficient means several things.

[0029] *Good generalization performance:* once SVM is presented with a training set, it is able to learn a rule which can correctly classify any new object quite often.

[0030] *Computational efficiency:* the algorithm is efficient in terms of speed and complexity, it is equivalent to looking for the minimum of a convex functional i.e. 1 with no local minima (unlike neural network).

[0031] *Robust in high dimensions:* dealing with large dimensional objects is usually difficult for learning algorithm, because of the over-fitting issue. SVM seem to be more robust than other methods in such cases.

[0032] Certain embodiments of the present invention employ the ANN for SNP prediction. ANN has remarkable ability to derive meaning from complicated or imprecise data and to extract patterns and trends. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyse. This expert can then be used to provide projections given new situations of interest and answer "what if" questions. Other advantages include: 1. Adaptive learning: An ability to learn how to do tasks based on the data given for training or initial experience; 2. Self-Organisation: An ANN can create its own organisation or representation of the information it receives during learning time; 3. Real Time Operation: ANN computations may be carried out in parallel, and special hardware devices are being designed and manufactured which take advantage of this capability; and Fault Tolerance via Redundant Information Coding: Partial destruction of a network leads to the corresponding degradation of performance. However, some network capabilities may be retained even with major network damage.

[0033] While ANN is known to those skilled in the art, a brief description will be helpful. In order to train a neural network to perform some task, it is to adjust the weights of each unit in such a way that the error between the desired output and the actual output is reduced. This process requires that the neural network compute the error derivative of the weights (EW). In other words, it must be calculated how the error changes as each weight

is increased or decreased slightly. A back propagation algorithm is the most widely used method for determining the **EW**.

[0034] The back-propagation algorithm is easiest to understand if all the units in a network are linear. The algorithm computes each **EW** by first computing **EA**, the rate at which the error changes as the activity level of a unit is changed. For output units, the **EA** is simply the difference between the actual and the desired output. To compute the **EA** for a hidden unit in a layer just before the output layer, it is first to identify all the weights between the hidden unit and the output units to which it is connected. It is then to multiply those weights by the **EAs** of those output units and add the products. This sum equals the **EA** for the chosen hidden unit. After calculating all the **EAs** in the hidden layer just before the output layer, the **EAs** for other layers can be calculated similarly, moving from layer to layer in a direction opposite to the way activities propagate through the network. This is what gives back propagation its name. Once the **EA** has been computed for a unit, it is straight forward to compute the **EW** for each incoming connection of the unit. The **EW** is the product of the **EA** and the activity through the incoming connection.

[0035] Now, a brief description of the procedure to perform the back propagation algorithm is given below as an illustration.

[0036] 1. Initialize the weights to small random values.

[0037] 2. Randomly choose an input pattern from the training sets (positive & negative).

[0038] 3. Propagate the signal forward through the network.

[0039] 4. Compute the error signal δ_k in the output layer

$$\delta_k = (t_k - o_k) f'(net_k)$$

where o_k is the actual activation value of an output unit k,

t_k is the target value for unit k,

f' is the derivative of the activation function f , and

net_k is the net input to neuron k.

[0040] 5. Compute the deltas for the preceding layers by propagating the errors backwards.

$$\delta_j = f'(net_j) \sum_k \delta_k w_{kj}$$

[0041] 6. Update weights using

$$w_{jk} \leftarrow w_{jk} + \eta \delta_k o_j$$

where η is some relatively small positive constant called the learning rate, and w_{jk} is the weight between neuron k and neuron j.

[0042] 7. Go to step 2 and repeat for the next pattern until the error in the output layer is below a pre-specified threshold or the maximum number of iterations is reached.

[0043] 8. After the training is completed, the samples from the test set are inputted into the neural network to obtain the prediction result.

[0044] The parameters for the back propagation include number of layers, number of nodes in each layer and learning rate η .

[0045] In practice, a set of parameters (number of layers, number of nodes per layer, learning rate and the error threshold of the number of iterations to stop the learning) are first decided. The weights in the ANN are then randomly set to small values. A dataset from the positive or negative training set is then chosen to train the ANN. The dataset includes part or all of the parameters (Entropy, Enthalpy, Free-energy, GC content, melting temperature, SNP distribution score info, sequence info, gene info, intron and exon info) and the output vector value is designated as 1 if it is from positive training set and the output vector value as 0 if from negative training set. The dataset is fed into the ANN and the error signal at the output layer is calculated. Next, the errors in the preceding layers are calculated and the errors are used to change the weights. Another dataset is then chosen to train the ANN. This process is repeated until the error threshold is reached or when the maximum number of iteration is reached. The trained ANN is then applied onto the test data to obtain the prediction result. The whole process is then repeated for other set of parameters. The set of parameters that yield the best prediction result are used for prediction on unknown sequences.

[0046] As with the SVM, a trial and error method is used to determine the most effective parameters. Different combination of parameters can be used to train the neural network, and the trained neural work can be used to predict the result on a test set. The parameters that give the highest prediction result are considered the best parameters.

[0047] Now, one embodiment of the present invention will be described in its application of SVM in the prediction of SNPs.

[0048] Usually, for analyzing data using learning machines, a specific problem is formalized for application of analysis through machine learning. Particularly important is

a proper formulation of the desired output of the learning machine. For instance, in predicting future performance of an individual equity instrument, a learning machine is likely to achieve better performance when predicting the expected future change rather than predicting the future price level. Here, the general problem can be formulated as prediction of SNP existence in any polynucleotide sequences. The polynucleotide sequences, however, may be from any resources including human, animal, plants and microorganisms. The SNPs identified in different species provide a good tool in studying evolution and genetic divergence. In addition, the polynucleotide sequences may be ones clustered by their homology or functionality. For instance, the genes in a signal transduction pathway or a metabolic pathway may be clustered together. Then, the SNPs identified in these clustered genes will provide insight information of the relationship between the sequence and function. Especially, the SNPs may provide insight into the causal relationships between SNPs and diseases. Thus, specific problems for predicting of SNPs can be further formulated to reflect any particular interests.

[0049] After a specific problem is formulated, training data are collected. Training data comprises a set of data points having known characteristics. Here, the data comprises positive and negative SNPs. The positive SNPs are defined as polynucleotide sequences including DNA and RNA that contain an SNP nucleotide residue. The SNP nucleotide residue has been identified or verified by any traditional experimental methods or computer programs. In the present invention, an SNP can have a length of sequence with 1 to 1000 nucleotide residues. Preferably, the SNP has 3—500, 3—400, 3—300, 3—200, 3—100, 3—50, 3—20, 3—10 or 3 nucleotide residues. The SNP site may be located in the middle of the SNP sequence with equal nucleotide residues at the upstream and downstream of the SNP. Alternatively, the SNP site may be located at any residue along the SNP sequence. The length of the SNP sequence and the location of the SNP site within the SNP sequence are not critical in applying the present invention to predict SNPs. The negative SNPs are defined as polynucleotide sequences that contain no SNP residues within a specified length. It does not mean that there is absolutely no SNP in a negative SNP sequence. It means that, under current available techniques, there is no identified or verified SNP residue within the negative SNP sequences. The suitable length of negative SNP sequences can be any number of nucleotide residues from 1 to 1000. The length is not critical in applying the present invention to predict SNPs.

[0050] The training set of positive and negative SNPs can be from a single species including human, horse, pig, mouse, rat, chicken and other animals. The SNPs can be selected randomly from any genes of the genomes of each species. Alternatively, the SNPs may be chosen from a well-defined cluster of genes. For example, the genes may share same functions or locations such as apoptosis genes and intracellular genes. In these circumstances, the positive and negative SNPs can be selected from the same clusters. The size of the training sets depends upon the characteristics of the positive and negative SNPs. Preferably, the training sets contain 100—100,000 SNPs. More preferably, the training sets contain 500—10,000 SNPs. Even more preferably, the training sets contain 1,000—5,000 SNPs.

[0051] The positive and negative SNPs can be extracted from any database, experiments and published journals. The source of the data and the type of the data provided are not crucial to the methods. Training data may be collected from one or more local and/or remote sources. The data may be provided through any means such as via the internet, server linkages or discs, CD-ROMs, DVDs or other storage means. The collection of training data may be accomplished manually or by way of an automated process, such as known electronic data transfer methods.

[0052] Next, the collected training data is optionally pre-processed in order to allow the learning machine to be applied most advantageously toward extraction of the knowledge inherent to the training data. In some cases, pre-processing involves selecting a method for reducing the dimensionality of the feature space, i.e., selecting the features which best represent the data. Methods which may be used for this purpose include recursive feature elimination (RFE) and elimination of data points based on expert knowledge, e.g., knowledge that measurement at ends of the measurement range tend to be predominantly noise. The features remaining after feature selection are then used to train a learning machine for purposes of pattern classification, regression, clustering and/or novelty detection. For the SNP training set, the SNPs may be further selected based on some additional criteria. For example, the SNPs must be from certain regions of the related genes or sequences including promoters, enhancers, silencers, introns, exons, and other specific regions. Or the SNPs must be from genes that express at the same developmental stages or physiological conditions. This procedure will add similarity or uniformity into the training set, therefore, enhancing prediction outcome.

[0053] Now, the learning machine is trained with the training data. It does not matter whether the training data has been pre-processed or not. As is known in the art, a learning machine is trained by adjusting its operating parameters until a desirable training output is achieved. The determination of whether a training output is desirable may be accomplished either manually or automatically by comparing the training output to the known characteristics of the training data. A learning machine is considered to be trained when its training output is within a predetermined error threshold from the known characteristics of the training data.

[0054] The learning machine is a kernel machine in which the kernels are selected in consideration of certain characteristics of the data. The SVM is one of the kernel machines. As is known in the art, different kernels will cause a SVM to produce varying degrees of quality in the output for a given set of input data. Therefore, the selection of an appropriate kernel may be essential to the desired quality of the output of the SVM. As is known in the art, exemplary kernels include polynomial kernels, radical basis classifier kernels, linear kernels, etc. In an alternate embodiment, a customized kernel may be created that is specific to a particular problem or type of data set. In yet another embodiment, multiple SVMs may be trained and tested simultaneously, each using a different kernel.

[0055] Those skilled in the art should appreciate that SVMs are capable of processing input data having extremely large dimensionality. However, according to the present invention, certain features have been selected to reduce the dimensionality of feature space. It is understood that there are unlimited features of SNPs that can be applicable in SVMs. By way of illustration, features used in the SVM in the present invention include Entropy, Enthalpy, Free-energy, GC content, melting temperature, SNP distribution score information, sequence information, gene information, intron information and exon information. These features can be used together or separately. Many features can be extracted from the DNA sequences directly, such as the GC content and sequence information. Others require that additional transformations be applied to the DNA sequences to create information of other dimensions for SNP prediction, such as the melting temperature, energies and structure information. The method and calculations are known to those skilled in the art. For example, calculation of DNA energies was disclosed

in a peer-reviewed journal. (Owczarzy et al, Predicting sequence-dependent melting stability of short duplex DNA oligomers, *Biopolymers*, 44:217, 1998).

[0056] After the SVMs are trained with the training sets, the trained SVMs are tested with test sets of positive and negative SNPs. In one embodiment, the test sets are the training sets themselves. Alternatively, test sets may be produced according to certain criteria designed by any user. For instance, if the SVM is trained with SNPs from a set of genes that are overexpressed in colon cancer, the test set may contain a set of genes that are overexpressed in colon cancer, but not used in the training set. In addition, test data may be collected from one or more local and/or remote sources. In practice, test data and training data may be collected from the same source(s) at the same time. Thus, test data and training data sets can be divided out of a common data set and stored in a local storage medium for use as different input data sets for a learning machine. Regardless of how the test data is collected, any test data used may be pre-processed in the same manner as was the training data. As should be apparent to those skilled in the art, a proper test of the learning may only be accomplished by using testing data of the same format as the training data.

[0057] The SVMs may be optionally retrained and retested with the same or different sets of positive and negative SNPs. The retraining and retesting can be done in the same manner as the training and testing. The output of training and testing is recorded, and the kernels and features that give out best prediction results are also recorded, and will be used for further testing.

[0058] Those skilled in the art should appreciate that a SVM is operable to ascertain an output having a global minimum error. However, as mentioned above, output results of a SVM for a given data set will typically vary with kernel selection. Therefore, there are in fact multiple global minimums that may be ascertained by a SVM for a given set of data. As used herein, the term "optimal minimum" or "optimal solution" refers to a selected global minimum that is considered to be optimal (e.g. the optimal solution for a given set of problem specific, pre-established criteria) when compared to other global minimums ascertained by a SVM. Accordingly, determining whether the optimal minimum has been ascertained may involve comparing the output of a SVM with a historical or predetermined value. Such a predetermined value may be dependant on the test data set. For example, in the context of a pattern recognition problem where data

points are classified by a SVM as either having a certain characteristic or not having the characteristic, a global minimum error of 50% would not be optimal. In this example, a global minimum of 50% is no better than the result that would be achieved by flipping a coin to determine whether the data point had that characteristic. As another example, in the case where multiple SVMs are trained and tested simultaneously with varying kernels, the outputs for each SVM may be compared with output of other SVM to determine the practical optimal solution for that particular set of kernels. The determination of whether an optimal solution has been ascertained may be performed manually or through an automated comparison process.

[0059] If it is determined that the optimal minimum has not been achieved by the trained SVM, kernel selection is adjusted. Adjustment of the kernel selection may comprise selecting one or more new kernels or adjusting kernel parameters. Furthermore, in the case where multiple SVMs were trained and tested simultaneously, selected kernels may be replaced or modified while other kernels may be re-used for control purposes.

[0060] The above disclosure generally describes the present invention. The following Examples will be of assistance to the understanding of the present invention. These Examples are described solely for purposes of illustration and are not intended to limit the scope of the invention. Changes in form and substitution of equivalents are contemplated as circumstances may suggest or render expedient. Although specific terms have been employed herein, such terms are intended in a descriptive sense and not for purposes of limitations.

[0061] Examples

[0062] 1. Random extraction of a set of positive SNPs from an SNP database.

[0063] Every positive SNP suitable for the present invention satisfied at minimum three conditions: validated by wet-lab; allele frequency higher than 0.2 where allele frequency is a measure of how common an allele is in a population; and sample size higher than 5 where the sample size refers to the number of chromosome being used in wet-lab experiment. 5,000 positive SNPs were randomly extracted from the Japan Single Nucleotide Polymorphisms (JSNP) database (<http://snp.ims.u-tokyo.ac.jp>). The JSNP database provides complete information required by our invention and is organized in

well-structured manner. Most of the SNPs in the JSNP database have been validated in the wet-lab. In the present invention, a set of 5000 positive SNPs was used as it is the size that the SVM machine could return result within acceptable time. It is noted that a set of more or less positive SNPs can be used to train the SVM to achieve reasonable prediction result.

[0064] In addition, most of the Japan SNPs come with a standard flanking sequence length, *i.e.*, 60bp upstream and 60bp downstream of the SNP position. Thus, the total sequence length with one positive SNP is 121bp. This entire sequence could be used in the present invention to extract features (discussed below) for SNP prediction. For the sake of simplicity of testing the SVM program, a positive SNP data used in the testing refer to a triplet of three nucleotides that consist of an SNP nucleotide and the adjacent upstream and downstream nucleotide of the SNP nucleotide.

[0065] 2. Random extraction of a set of DNA sequences as negative SNPs from the human genome draft

[0066] A negative SNP is defined as a contiguous stretch of nucleotide sequence that contains no reported SNP. While it is possible that an SNP may have not been detected within the sequence, the probability is quite low in the random extraction of the present invention. Especially, the sequence is extracted from an SNP poor region of the human genome draft, where no SNP has been reported in that region. In the present invention, 5000 DNA sequences with a length of 121bps were randomly extracted from human genome draft NCBI Build 33. The negative SNPs used for the testing were the DNA sequences that contained no SNP within 60bps upstream and downstream of the chosen nucleotide position. In addition, the negative SNPs were randomly chosen over the 24 chromosomes. For the sake of simplicity of testing the SVM program, a negative SNP data used for testing refer to a triplet of three nucleotides that consist of a chosen nucleotide and the adjacent upstream and downstream nucleotide of the chosen nucleotide.

[0067] 3. Features used in the SVM

[0068] Features used in the SVM in the present invention include Entropy, Enthalpy, Free-energy, GC content, melting temperature, SNP distribution score information, sequence information, gene information, intron information and exon information. These features can be used together or separately. Many features can be extracted from the DNA sequences directly, such as the GC content and sequence information. Others require that additional transformations be applied to the DNA

sequences to create information of other dimensions for SNP prediction, such as the melting temperature, energies and structure information. The method and calculations are known to those skilled in the art. For example, calculation of DNA energies was disclosed in a peer-reviewed journal. (Owczarzy et al, Predicting sequence-dependent melting stability of short duplex DNA oligomers, *Biopolymers*, 44:217, 1998). Nevertheless, a brief description of calculations of different features is helpful.

[0069] a. Free energy change (ΔG)

[0070] Free energy change (ΔG) of a reaction is a measure of its spontaneity, and of how much energy is released or required to drive the reaction. Free energy change is made up of two components, enthalpy change (ΔH) and entropy change (ΔS). Their relationship is represented by the equation: $\Delta G = \Delta H - T \Delta S$. The enthalpy change (ΔH) and entropy change (ΔS) will be discussed below. For a sequence with n nucleotides, the free energy change is calculated: $\Delta G = \Delta G_{1,2} + \Delta G_{2,3} \dots + \Delta G_{n-1,n}$. For example, free energy change of AAT = free energy of AA + free energy of AT = $1.02 + 1.02 = 2.04$. Table 1 shows the free energy changes for different pairs of nucleotides predicted from their structures. (SantaLucia, et al., Improved nearest-neighbour parameters for predicting DNA duplex stability, *Biochemistry*, 35:3555-62, 1996).

[0071] Table 1. Free Energy Change Values for Different Nucleotide Pairs

AA	1.02
TT	1.02
CC	1.77
GG	1.77
AT	0.73
TA	0.6
AC	1.38
CA	2.38
AG	1.46
GA	1.46
CT	1.16
TC	1.16
TG	1.43
GT	1.43
CG	2.09
GC	2.28

[0072] b. Entropy (ΔS)

[0073] Entropy (ΔS) is a measure of the randomness or disorder in either the system or the surroundings. For a sequence with n nucleotides, the entropy is calculated: $\Delta S = \Delta S_{1,2} + \Delta S_{2,3} \dots + \Delta S_{n-1,n}$. For example, the entropy of AAT = entropy of AA + entropy of AT = $23.6 + 18.8 = 42.4$. Table 2 shows the Entropy (ΔS) for different pairs of nucleotides predicted from their structures. (SantaLucia, et al., 1996).

[0074] Table 2: Entropy Values for Different Nucleotide Pairs

AA	23.6
TT	23.6
CC	15.6
GG	15.6
AT	18.8
TA	18.5
AC	19.3
CA	19.3
AG	20.3
GA	20.3
CT	16.1
TC	16.1
TG	23
GT	23
CG	25.5
GC	28.4

[0075] c. Enthalpy (ΔH)

[0076] Enthalpy (ΔH) is the heat/energy released or consumed during a chemical reaction due to differences between the chemical bond energies of the products and reactants of the reaction. Whenever $\Delta H < 0$, heat/energy is released by the reaction; $\Delta H > 0$, heat/energy is consumed by the reaction; and $\Delta H = 0$, heat/energy is neither released nor consumed by the reaction. For a sequence with n nucleotides, the enthalpy is calculated: $\Delta H = \Delta H_{1,2} + \Delta H_{2,3} \dots + \Delta H_{n-1,n}$. For example, the enthalpy of AAT = enthalpy of AA + enthalpy of AT = $8.4 + 6.5 = 14.9$. Table 3 shows the Enthalpy (ΔH) for different pairs of nucleotides predicted from their structures. (SantaLucia, et al., 1996).

[0077] Table 3: Enthalpy Values for Different Nucleotide Pairs

AA	8.4
TT	8.4
CC	6.7

GG	6.7
AT	6.5
TA	6.3
AC	7.4
CA	7.4
AG	7.7
GA	7.7
CT	6.1
TC	6.1
TG	8.6
GT	8.6
CG	10.1
GC	11.1

[0078] d. Melting temperature

[0079] Melting temperature is defined as the temperature at which 50% of the oligonucleotide and its perfect complement are in duplex. The melting temperature is calculated using Wallace rule (Wallace, et al., Nucleic Acids Res., 6:3543, 1979): $MT = 4 \times (\#C + \#G) + 2 \times (\#A + \#T) ^\circ C$. For example, if the sequence is ATG, then, $\#C=0$, $\#G=1$, $\#A=1$, $\#T=1$, the Melting temperature = $4 \times (0+1) + 2 \times (1+1) = 8$.

[0080] e. GC content

[0081] The GC content of genomic DNA is defined as the mean percentage of guanine (G) and cytosine (C). For example, a sequence of 10 bases, ATGTACCCCG, will have GC content of 60%. GC rich regions melt at higher temperatures than regions that are AT rich. For example, GC content for ATG is: $(\text{No of G \& C})/(\text{No of A \& T}) = (0+1)/(1+1) = 50\%$.

[0082] f. SNP distribution score information

[0083] The SNP distribution score is calculated as follows: dividing the human genome into smaller regions of predetermined length (100000 nucleotides length is used in this project); recording the number of SNPs located in each region for all the regions in the entire human genome; using the position of the SNP data to locate the region that it belongs to; and recording the number of SNPs found in the region as the SNP distribution score.

[0084] g. Sequence information

[0085] The sequence information contains the numerical (binary) representation of the actual nucleotide sequence. The nucleotides are converted to numeric codes based on

Table 4 below. For Example, ACT can be represented by 0001 0010 0100 with the presumption that three nucleotides are used.

[0086] Table 4: Numeric Code for Nucleotide Information

A	0 0 0 1
C	0 0 1 0
T	0 1 0 0
G	1 0 0 0

[0087] h. Gene information

[0088] Gene is a unit of DNA that encodes information for production of a single polypeptide chain. In the present invention, the gene information was given a value of 1 if the chosen nucleotide position is found in a gene and 0 if otherwise. The gene information from the EMBL database was used as reference.

[0089] i. Intron information

[0090] Introns are portions of genomic DNA that are transcribed (and thus present in the primary transcript) but later spliced out. In the present invention, the intron information was given a value of 1 if the chosen nucleotide position was found in an intron and 0 if otherwise. The gene information from the EMBL database was used as reference.

[0091] j. Exon information

[0092] Exons are the portions of a genomic DNA sequence that will be represented in the final, mature mRNA. In the present invention, the exon information was given a value of 1 if the chosen nucleotide position was found in an exon and 0 if otherwise. The gene information from the EMBL database was used as reference.

[0093] 4. Initial training and testing

[0094] The values of different features as discussed in item (3) can be computed from the original DNA sequences. In the present invention, the DNA sequences were the sets of positive and negative SNPs. The training and prediction can be performed on all or a sub set of the features, depending the needs. One was to divide the positive SNPs into test and train groups evenly. In the present invention, for a set of 5000 DNA sequences, 2500 SNPs were used to train the SVM, while the other 2500 SNPs were used to test the prediction model. The process is repeated for the negative SNPs dataset.

[0095] A selected machine-learning algorithm (e.g. SVM) can be trained with the positive and negative training sets using different kernels and parameters. In the present invention, SVMLight was used to implement the SVM-based SNP prediction (Joachims, T., Making large-scale SVM learning practical, In: Schölkopf, B., Burges, C., Smola, A. (Eds.), Advances in Kernel Methods-Support Vector Learning. MIT Press, Cambridge, MA, 1999; Joachims, T., Proceedings of the International Conference on Machine Learning, 1999). The training sets of 2,500 positive and 2,500 negative SNPs were used to train the SVMLight with different parameters for the linear kernel and the radial basis function (RBF) kernel. The best prediction results were recorded under different kernels and different parameters. The radial basis function kernel gave the best results when compared to linear and polynomial kernel. Then, the trained machine-learning algorithm SVMLight was tested with the positive and negative test sets, and the best prediction results were recorded under different kernels and different parameters.

[0096] 5. Retraining and retesting

[0097] The training sets and the test sets were reversed and the SVM was retrained and tested to get a second set of prediction results. The average prediction result was recorded and the best performing kernel and parameters were chosen.

[0098] Once the best set of kernel and parameters were selected, the algorithm was retrained with the entire set of 5,000 positive and 5,000 negative SNPs using the selected kernel and parameters.

[0099] The testing results from the present invention are partially shown in Table 5. The energy feature provided the best prediction rates with an 85% accuracy. The energy feature consists of the Enthalpy, Entropy, and Free Energy.

[0100] Table 5. Prediction results for different features and different parameters

Features	Gamma	Cost	j	Correct	Correct
				Positive	Negative
				Prediction	Prediction
All	0.001	1	1	0.7684	0.696
Energy	0.1	0.05	1	0.8504	0.868

[0101] Conclusion

[0102] SNP is not a random process. SVM can predict SNPs with an 85% accuracy. Without wish to be bound by any specific theories, inventors hypothesize that when there is environmental pressure on the sequence, it should be expected that certain nucleotide position with weaker bonds would be easier to break and replace than one with stronger bonds. The proposed prediction system can be used to help scientists to search for SNPs of interest by predicting regions that are likely to contain SNPs and thus cutting down on the time and cost needed to discover SNPs.

[0103] While the present invention has been described with reference to particular embodiments, it will be understood that the embodiments are illustrated and that the invention scope is not so limited. Alternative embodiments of the present invention will become apparent to those having ordinary skill in the art to which the present invention pertains. Such alternate embodiments are considered to be encompassed within the spirit and scope of the present invention. Accordingly, the scope of the present invention is described by the appended claims and is supported by the foregoing description.

CLAIMS

What is claimed is:

1. A method of utilizing a learning machine for predicting of single nucleotide polymorphism (SNP) in a polynucleotide sequence, comprising steps of:
 - generating training sets of positive and negative SNPs;
 - calculating one or more parameters of each member of the training sets of positive and negative SNPs, wherein the one or more parameters of positive and negative SNPs are selected from the group consisting of free energy change (ΔG), entropy (ΔS), enthalpy (ΔH), melting temperature, GC content, SNP distribution score information, sequence information, gene information, intron information and exon information;
 - training a learning machine algorithm with the one or more parameters of members of the training sets of positive and negative SNPs; and
 - testing the trained learning machine algorithm with test sets of positive and negative SNPs; thereby the best prediction results are recorded.
2. The method of claim 1, further comprising steps of:
 - running a DNA sequence through the trained and tested learning machine algorithm to predict of whether an SNP exists in the DNA sequence.
3. The method of any of claims 1, wherein the test sets may be the same of the training sets.
4. The method of any of claims 1, wherein the training and test sets of positive and negative SNPs are randomly chosen.
5. The method of claim 1, wherein the free energy change is calculated by the following: for a positive or negative SNP having a sequence of $x_1x_2x_3\dots x_{n-1}x_n$, the free energy change

(ΔG) of the sequence of $x_1x_2x_3\dots x_{n-1}x_n$ equals to: $\Delta G_{x_1x_2} + \Delta G_{x_2x_3} + \dots + \Delta G_{x_{n-1}x_n}$; wherein the x represents any of A, T, G, and C.

6. The method of claim 1, wherein the entropy is calculated by the following: for a positive or negative SNP having a sequence of $x_1x_2x_3\dots x_{n-1}x_n$, the entropy (ΔS) of the sequence of $x_1x_2x_3\dots x_{n-1}x_n$ equals to: $\Delta S_{x_1x_2} + \Delta S_{x_2x_3} + \dots + \Delta S_{x_{n-1}x_n}$; wherein the x represents any of A, T, G, and C.

7. The method of claim 1, wherein the enthalpy is calculated by the following: for a positive or negative SNP having a sequence of $x_1x_2x_3\dots x_{n-1}x_n$, the enthalpy (ΔH) of the sequence of $x_1x_2x_3\dots x_{n-1}x_n$ equals to: $\Delta H_{x_1x_2} + \Delta H_{x_2x_3} + \dots + \Delta H_{x_{n-1}x_n}$; wherein the x represents any of A, T, G, and C.

8. The method of claim 1, wherein the melting temperature is calculated by $MT = 4x(\#C + \#G) + 2x(\#A + \#T)^\circ C$.

9. The method of claim 1, wherein the GC content is calculated as the mean percentage of G + C.

10. The method of claim 1, wherein the SNP distribution score is calculated as follows: dividing the human genome into smaller regions of predetermined length (e.g., 100,000 nucleotides); recording the number of SNPs located in each regions for all the regions in the entire human genome; and locating the region that an SNP belongs to; and recording the location and the number of SNPs found in the region.

11. The method of any of claims 1, further comprising an optional step of retraining and retesting the learning machine algorithm by rewrapping the trainings sets and test sets.

12. The method of any of claims 1, wherein the learning machine algorithm is an artificial neural network algorithm.

13. The method of any of claims 1, wherein the learning machine algorithm is a support vector machine algorithm.
14. A method of utilizing a support vector machine for predicting of single nucleotide polymorphism (SNP) in a polynucleotide sequence, comprising steps of:
 - generating training sets of positive and negative SNPs;
 - calculating one or more parameters of each member of the training sets of positive and negative SNPs, wherein the one or more parameters of positive and negative SNPs are selected from the group consisting of free energy change (ΔG), entropy (ΔS), enthalpy (ΔH), melting temperature, GC content, SNP distribution score information, sequence information, gene information, intron information and exon information;
 - training a support vector machine algorithm with the one or more parameters of each member of the training sets of positive and negative SNPs; and
 - testing the trained support vector machine algorithm with test sets of positive and negative SNPs; thereby the best prediction results are recorded.
15. The method of claim 14, further comprising steps of:
 - running a DNA sequence through the trained and tested support vector machine algorithm to predict of whether an SNP exists in the DNA sequence.
16. The method of any of claims 14, wherein the test sets may be the same of the training sets.
17. The method of any of claims 14, wherein the training and test sets of positive and negative SNPs are randomly chosen.
18. The method of claim 14, wherein the free energy change is calculated by the following: for a positive or negative SNP having a sequence of $x_1x_2x_3\dots x_{n-1}x_n$, the free energy

change (ΔG) of the sequence of $x_1x_2x_3\dots x_{n-1}x_n$ equals to: $\Delta G_{x_1x_2} + \Delta G_{x_2x_3} + \dots + \Delta G_{x_{n-1}x_n}$; wherein the x represents any of A, T, G, and C.

19. The method of claim 14, wherein the entropy is calculated by the following: for a positive or negative SNP having a sequence of $x_1x_2x_3\dots x_{n-1}x_n$, the entropy (ΔS) of the sequence of $x_1x_2x_3\dots x_{n-1}x_n$ equals to: $\Delta S_{x_1x_2} + \Delta S_{x_2x_3} + \dots + \Delta S_{x_{n-1}x_n}$; wherein the x represents any of A, T, G, and C.

20. The method of claim 14, wherein the enthalpy is calculated by the following: for a positive or negative SNP having a sequence of $x_1x_2x_3\dots x_{n-1}x_n$, the enthalpy (ΔH) of the sequence of $x_1x_2x_3\dots x_{n-1}x_n$ equals to: $\Delta H_{x_1x_2} + \Delta H_{x_2x_3} + \dots + \Delta H_{x_{n-1}x_n}$; wherein the x represents any of A, T, G, and C.

21. The method of claim 14, wherein the melting temperature is calculated by $MT = 4x(\#C + \#G) + 2x(\#A + \#T)^{\circ}C$.

22. The method of claim 14, wherein the GC content is calculated as the mean percentage of G + C.

23. The method of claim 14, wherein the SNP distribution score is calculated as follows: dividing the human genome into smaller regions of predetermined length (e.g., 100,000 nucleotides); recording the number of SNPs located in each regions for all the regions in the entire human genome; and locating the region that an SNP belongs to; and recording the location and the number of SNPs found in the region.

24. The method of any of claims 14, further comprising an optional step of retraining and retesting the learning machine algorithm by rewrapping the trainings sets and test sets.

25. A method of utilizing an artificial neural network for predicting of single nucleotide polymorphism (SNP) in a polynucleotide sequence, comprising steps of:

generating training sets of positive and negative SNPs;

calculating one or more parameters of each member of the training sets of positive and negative SNPs, wherein the one or more parameters of positive and negative SNPs are selected from the group consisting of free energy change (ΔG), entropy (ΔS), enthalpy (ΔH), melting temperature, GC content, SNP distribution score information, sequence information, gene information, intron information and exon information;

training an artificial neural network algorithm with the one or more parameters of members of the training sets of positive and negative SNPs; and

testing the trained artificial neural network algorithm with test sets of positive and negative SNPs; thereby the best prediction results are recorded.

26. The method of claim 25, further comprising steps of:

running a DNA sequence through the trained and tested artificial neural network algorithm to predict of whether an SNP exists in the DNA sequence.

27. The method of any of claims 25, wherein the test sets may be the same of the training sets.

28. The method of any of claims 25, wherein the training and test sets of positive and negative SNPs are randomly chosen.

29. The method of claim 25, wherein the free energy change is calculated by the following: for a positive or negative SNP having a sequence of $x_1x_2x_3\dots x_{n-1}x_n$, the free energy change (ΔG) of the sequence of $x_1x_2x_3\dots x_{n-1}x_n$ equals to: $\Delta G_{x_1x_2} + \Delta G_{x_2x_3} + \dots + \Delta G_{x_{n-1}x_n}$; wherein the x represents any of A, T, G, and C.

30. The method of claim 25, wherein the entropy is calculated by the following: for a positive or negative SNP having a sequence of $x_1x_2x_3\dots x_{n-1}x_n$, the entropy (ΔS) of the sequence of $x_1x_2x_3\dots x_{n-1}x_n$ equals to: $\Delta S_{x_1x_2} + \Delta S_{x_2x_3} + \dots + \Delta S_{x_{n-1}x_n}$; wherein the x represents any of A, T, G, and C.

31. The method of claim 25, wherein the enthalpy is calculated by the following: for a positive or negative SNP having a sequence of $x_1x_2x_3...x_{n-1}x_n$, the enthalpy (ΔH) of the sequence of $x_1x_2x_3...x_{n-1}x_n$ equals to: $\Delta H_{x_1x_2} + \Delta H_{x_2x_3} + ... + \Delta H_{x_{n-1}x_n}$; wherein the x represents any of A, T, G, and C.
32. The method of claim 25, wherein the melting temperature is calculated by $MT = 4x(\#C + \#G) + 2x(\#A + \#T)^{\circ}C$.
33. The method of claim 25, wherein the GC content is calculated as the mean percentage of G + C.
34. The method of claim 25, wherein the SNP distribution score is calculated as follows: dividing the human genome into smaller regions of predetermined length (e.g., 100,000 nucleotides); recording the number of SNPs located in each regions for all the regions in the entire human genome; and locating the region that an SNP belongs to; and recording the location and the number of SNPs found in the region.
35. The method of any of claims 25, further comprising an optional step of retraining and retesting the learning machine algorithm by rewrapping the trainings sets and test sets.