GENOME SIGNAL PREDICTION USING MACHINE LEARNING ALGORITHMS.

A Minor Project Report

in partial fulfillment of the degree

Bachelor of Technology
in
Computer Science & Engineering

By

Roll.No 19K41A0563	Name : ANJU HALIYA
Roll.No 19K41A0577	Name : NAINENI ANASRI
Roll.No 19K41A0582	Name : RAMPELLI SAIMANOGNYA
Roll.No 19K41A0596	Name : DASARI SINDHU SRI

Under the Guidance of

A. Santhosh.

Submitted to

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
S.R.ENGINEERING COLLEGE(A),ANANTHASAGAR, WARANG(Affiliated to JNTUH, Accredited by NBA) May, 2022.

SR Engineering College
Innovation . Creativity . Entrepreneurship

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the Minor Project Report entitled " GENOME SIGNAL PREDICTION USING MACHINE LEARNING ALGORITHMS " is a record of bonafide work carried out by the student(s) ANJU HALIYA, NAINENI ANASRI, RAMPELLI SAI MANOGNYA, DASARI SINDHU SRI bearing RollNo(s)19K41A0563, 19K41A0577, 19K41A0582, 19K41A0596 2021-22 during the academic year 2021-22 in partial fulfillment of the award of the degree of ***Bachelor of Technology*** in **Computer Science & Engineering** by the Jawaharlal Nehru Technological University, Hyderabad.

Supervisor **Head of the Department**

External Examiner

ACKNOWLEDGEMENT

- I would like to express my gratitude to our guide, A.santhosh sir, who guided me throughout this project. I could say that this project would not have achieved this accuracy without his assistance. Sir has been our great support in helping gathering the dataset and sorting out complex errors of our code and enhancing our understanding of the project.
- I would also like to thank V. venkatraman sir, who introduced us to the magnificient python language and made us familiar with machine learning of Neural Networks.
- I would like to thank our HOD, Shashikala mam, who has ensured a required importance in slotting time for minor project and enabled us to have interaction with our guides and helped with any queries arising in between.
- I also would like to express gratitude to my Team - T5 comprising Anju Haliya, Anasri, Sai manognya, Sindhu sri who have worked as a fruitful and healthy team with all productive thoughts and effort with good co-ordination.
- I would also like to express gratitude to all the authors of the websites, research papers, publications, technitions I have referred to in order to gain knowledge for this project.
- Last but not the least, I would also like to thank my friends and family who supported me and offered deep insight into the study.

ABSTRACT

Cis - regulatory sequence, such as enhancers and promoters, control development and physiology by regulating gene expression. Mutations that affect the function of these sequences contribute to phenotypic diversity within and between species. Enhancer sequences are regulatory DNA sequences that, when bound by specific proteins called transcription factors, enhance the transcription of an associated gene. Enhancers are used in identifying genetically inherited diseases. Regulation of transcription is the most common form of gene control, and the activity of transcription factors allows genes to be specifically regulated during development and in different types of cells. The goal of transcription is to make a RNA copy of a gene's DNA sequence. For a protein-coding gene, the RNA copy, or transcript, carries the Transcription is the first step in gene expression, in which information from a gene is used to construct a functional product such as a protein. The information needed to build a polypeptide (protein or protein sub unit). Polypeptides help make up proteins by bonding numerous amino acids together. Proteins are created by the bonding of two or more polypeptides, which are then folded into a specific shape for a particular protein. Therefore, our proposed model predicts weather the given sequence is enchancer or not using ANN(Artificial Neural Networks). It account to given the accurate prediction of binary classification after performing feature extraction techniques of Kmer, RCKmer, ENAC, CKSNAP. This area of research could be so useful because the genome transcriptions of specific kind are significantly responsible for diseases like cancer.This days Bio analysts face problem in detecting whether the diseases are genetically inherited or not. So in order to help bio analysts in detecting the genetic specific diseases we have came across this project. Our model predicts whether the given genome sequence is enhancer or not with 99% accuracy using ANN model. Finally our model helps in predicting the sequences of enhancers.

CONTENTS

1.INTRODUCTION

Enhancers are non-coding fragments in DNA sequences, which play an important role in gene transcription and translation. Enhancers are a small area of DNA that can link with protein, located upstream or downstream of the gene, and gene transcription will be enhanced after they bind with protein. The central dogma of molecular biology states that DNA makes RNA makes proteins.The process by which DNA is copied to RNA is called transcription, and that by which RNA is used to produce proteins is called translation.

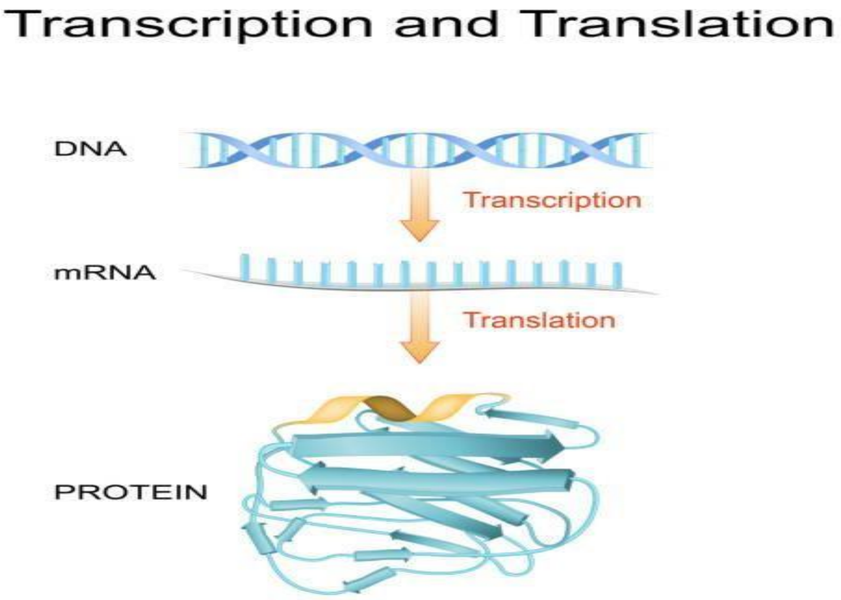


Fig : 1 Transcription and Translation

Our aim is to build a system which will predict the genome signal is enhancer or not by using learning ANN(Artificial Neural Networks).

1.1 EXISTING SYSTEM

The enhancer is a short regulatory element that plays a major role in gene expression. To identify enhancers, an experimental process takes a long time and high cost. Therefore, an accurate computational tool is a much-needed work in this area. Existing techniques were developed by the use of handcrafted features followed by machine learning techniques, Enhancers are cis-regulatory elements that activate promoter transcription over large distances and play an important role in upregulating eukaryotic gene expression and the production of RNA and proteins . The development of all living organisms depends on differential gene expression, which is controlled by enhancers. They usually function independently of orientation and at any distance location (up to 1 Mbp) from their target genes, or even in a different chromosome . Their presence in distinct genomic regions, or the dynamic nature of enhancers, makes the identification challenging task. Despite these challenges, some techniques have been developed in the past for identifying enhancers on a genomic scale. Initially, in early development, enhancers were found in different vertebrates and mammalian species . Genetic variations in enhancers regions are responsible for some human diseases such as cancer, inflammatory bowel disease etc. Even though experimental methods are time-consuming and expensive but some works have been done successfully to detect many regulatory enhancers, such as chromatin immunoprecipitation followed by high- throughput sequencing (ChIP-Seq) , that determines the chromatin accessibility in different organism and tissues.

The experimental process takes a long time and high cost, on the other hand, enhancer's identification by computational approach is a popular area of research, especially in biomedical research and computational biology. In order to speed up enhancer identification in genomics, many computational prediction methods have been established. Many developed computational tools for identify enhancers are, including EnhancerPred ,EnhancerPred2.0 , ChromiaGenSvm , DELTA,GKM-SVM , DEEP-ENCODE , BiRen and DeepEnhancer . Most recently, two predictors, iEnhancer-2L and iEnhancer-EL , were proposed not only for identifying enhancers but also their strength as well. In iEnhancer-2L, pseudo-k-tuple nucleotide composition was utilized with a support vector machine (SVM) as a predictive model. Meanwhile, iEnhancer-EL was formed by fusing an array of six key individual classifiers and ten key individual classifiers for both stages of identification. iEnhancer-EL was followed by SVM based on k-mer, subsequence profile and PseKNC (PseKNC K-tuple Nucleotide Composition). Even though the above-mentioned machine learning-based methods can provide good results, the deep learning-based model performs better without the need for manual feature extraction . Furthermore, the above machine learning models need domain knowledge to hand-design the input features from the given DNA/RNA/Protein sequences, on the other hand, the important features from multiple stages of abstraction can be automatically learned effectively with CNN.

1.2 PROPOSED SYSTEM

Our proposed system is genome signal predictor using machine learning techniques. In our model we aim to predict the given gene expression is enhancer or not. It comes under binary networks)classification type of machine learning model because, we are predicting whether the given gene expression is enhancer or not. For this, we used machine learning algorithm ANN(Artificial neural networks) to find the best accuracy. Machine learning algorithms cannot directly perform annotations on continuous nucleotide sequences, so it is necessary to convert nucleotide sequences represented by strings into feature vectors. We have used these three feature extraction techniques: CKSNAP ,ENAC ,PSTKNC_3MERP. CKSNAP converts a DNA sequence into a numerical feature vector by computing the occurrence frequency of all possible *k*-spaced nucleotide pairs (KNP) along the sequence. For instance, in the sequence ‘AXXTXXXG’, ‘AT’ and ‘TG’, respectively, represent two-spaced and three-spaced nucleotide pairs. ENAC (Enhanced nucleic acid composition) is the frequency of each nucleotide occurring within a fixed sequence window length, which slides continuously from the 5 end to the 3 end of each nucleotide sequence and usually used to encode nucleotide sequences of the same length. Ipro2L-PSTKNC This is an 2 layer prediction of features of ieee article. Where analyzing happens mapping from 2 layer sequences.

1. LITERATURE SURVEY

2.1 RELATED STUDY

Existing models uses one feature extraction technique to predict whether the given genome sequence is enhancer or not. Our model uses multiple feature extraction techniques to predict whether the sequences enhancer or not. The below listed are the related studies to our project:

[1] Model for predicting and analyzing transcription factor binding sites using a tool called CiiDER, as this is done using programming language called Java,this model finds out the gene signal pattern whether belongs to enhancer or promotor.

[2] classification of sequences as enhancers or non-enhancers. In this paper, They derived statistical and nonlinear dynamic features along with k-mer features from experimentally validated sequences taken from Vista Enhancer Browser through random walk model and applied different machine learning based methods to predict whether an input test sequence is enhancer or not.

[3] The enhancer classification model was built by word2vec and attention-based Bi -LSTM. Finally, the accuracies of our enhancer identification and classification models were 77.25% and 73.50%, respectively, and the Matthews' correlation coefficients (MCCs) were 0.5470 and 0.4881, respectively, which were better than the performance of most predictors.

2.2 SYSTEM STUDY

We have used ANN - Artificial Neural Networks accounting for its precision and accuracy. We have worked on 1484 Enhancer ACGT sequences of benchmark dataset of enhancers and non enhancers, a total of 2968 samples. We have used the feature extracting techniques of CKSNAP,ENAC,PSTKNC_3MERP and concatenated all of them to bring out a dataset of 1484 * 1078 dimension. We have deployed our model using ANN and optimizer adam, loss function of binary entropy, activation function of RELU and soft max output function and predicts the result using binary classification. Our system accounts a whopping 99% accuracy and the sensitivity, specificity and mathews correlation coorelation matrix accounting to 99%,99%, 98% significance. Enhancer file containing only benchmark dataset of enhancers.

2. DESIGN

3.1 REQUIREMENT SPECIFICATION

3.1.1 FUNTIONAL REQUIREMENTS

- ENHANCER DATA OF THE GENOME SIGNAL
- USER - BIO INFORMATION ANALYST

3.1.2 NON FUNCTIONAL REQUIREMENTS

- A SYSTEM TO RUN ANN OF PYTHON
- DATASET OF ACGT ENHANCER SEQUENCES
- PIP LIBRARIES OF KERAS,TENSORFLOW,NUMPY, SKLEARN,SGBOOST.

3.2 UML DIAGRAMS

3.2.1 USE CASE DIAGRAM

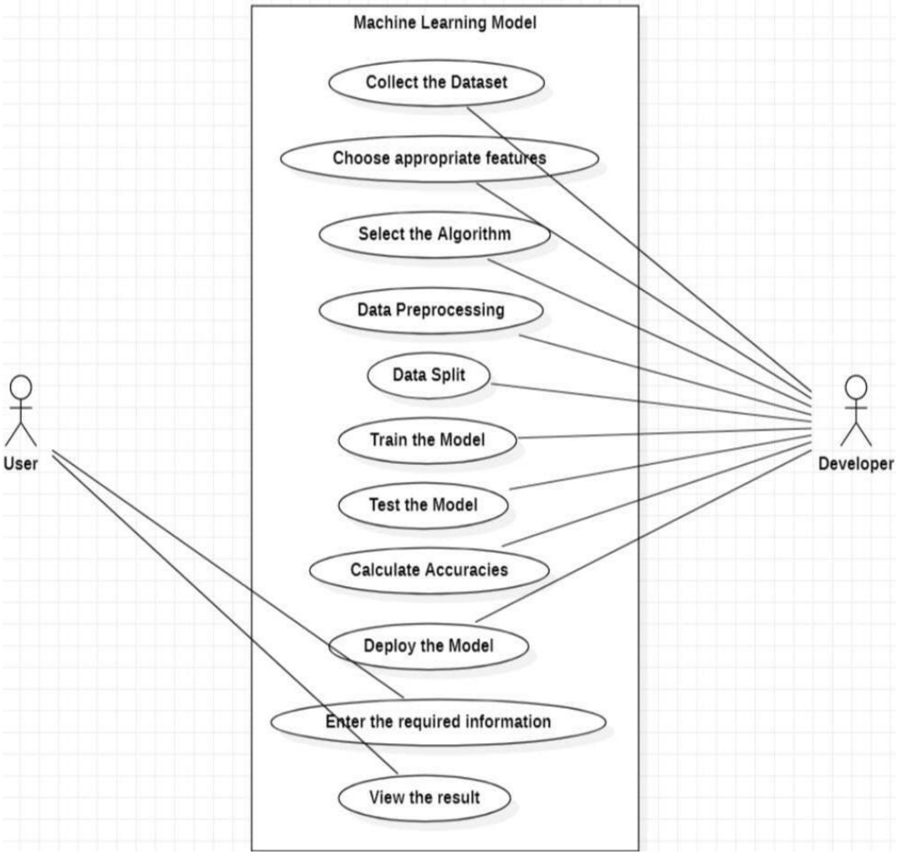


Fig 2 : UML Diagram

4.IMPLEMENTATION

4.1 MODULES

4.1.1 FEATURE EXTRACTION TECHNIQUES

4.1.1.1 CKSNAP

CKSNAP converts a DNA sequence into a numerical feature vector by computing the occurrence frequency of all possible *k*-spaced nucleotide pairs (KNP) along the sequence. For instance, in the sequence ‘AXXTXXXG’, ‘AT’ and ‘TG’, respectively, represent two-spaced and three-spaced nucleotide pairs. The frequency of KNP can be defined as:

$$f(KNP)=m(KNP)N-k-1,k\in[0,kmax],f(KNP)=m(KNP)N-k-1,k\in[0,kmax],(2)$$

where m(KNP) represents the number of KNP along the sequence, and (N-k-l) represents the number of KNP along a sequence with length N. We kept k_{max}=5 that generated 96D feature vector.

4.1.1.2 ENAC

Enhanced nucleic acid composition is the frequency of each nucleotide occurring within a fixed sequence window length, which slides continuously from the 5 end to the 3 end of each nucleotide sequence and usually used to encode nucleotide sequences of the same length.

4.1.1.3 Ipro2L-PSTKNC

This is an 2 layer prediction of features of ieee article. Where analyzing happens mapping from 2 layer sequences

4.1.2 ACTIVATION FUNCTION - RELU

ReLU stands for Rectified Linear Unit. Although it gives an impression of a linear function, ReLU has a derivative function and allows for backpropagation while simultaneously making it computationally efficient. The main catch here is that the ReLU function does not activate all the neurons at the same time. The neurons will only be deactivated if the output of the linear transformation is less than 0.
Mathematically it can be represented as

$$ReLU$$

$$f(x) = max\ (0, x)$$

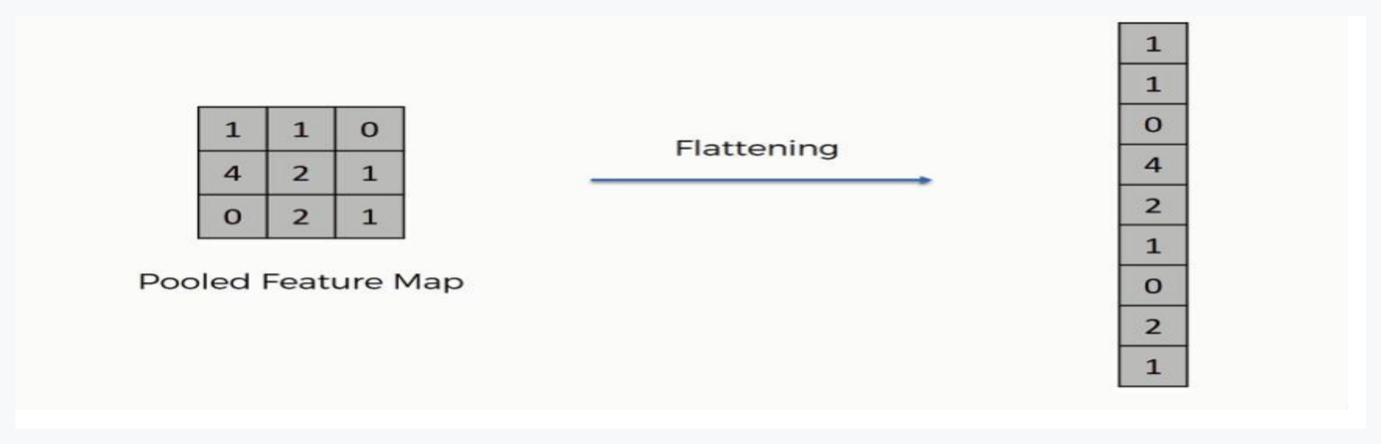
The advantages of using ReLU as an activation function are as follows:

- Since only a certain number of neurons are activated, the ReLU function is far more computationally efficient when compared to the sigmoid and tanh functions.

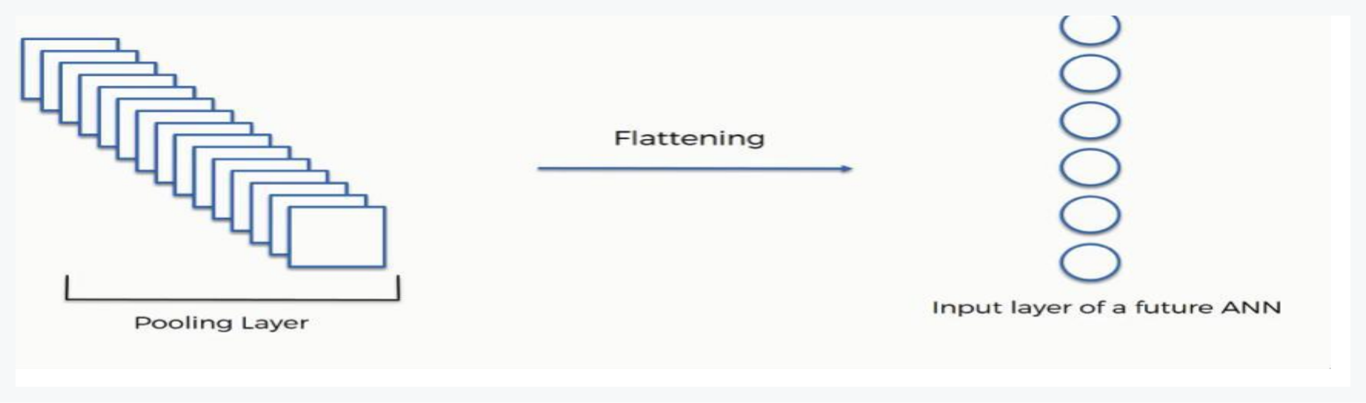
- ReLU accelerates the convergence of gradient descent towards the global minimum of the loss function due to its linear, non-saturating property.

4.1.3 LOSS FUNCTION - binary_crossentropy

As the name of this step implies, we are literally going to flatten our pooled feature map into a column like in the image below.



The reason we do this is that we're going to need to insert this data into an artificial neural network later on.



As you see in the image above, we have multiple pooled feature maps from the previous step.

What happens after the flattening step is that you end up with a long vector of input data that you then pass through the artificial neural network to have it processed further.

What is Binary Cross Entropy Or Logs Loss

Binary cross entropy compares each of the predicted probabilities to actual class output which can be either 0 or 1. It then calculates the score that penalizes the probabilities based on the distance from the expected value. That means how close or far from the actual value.

A formal definition of binary cross-entropy

Binary Cross Entropy is the negative average of the log of corrected predicted probabilities.

4.1.4 OPTIMZERS - ADAM

Adam can be looked at as a combination of RMSprop and Stochastic Gradient Descent with momentum. It uses the squared gradients to scale the learning rate like RMSprop and it takes advantage of momentum by using moving average of the gradient instead of gradient itself like SGD with momentum. Let's take a closer look at how it works.

Adam is an adaptive learning rate method, which means, it computes individual learning rates for different parameters. Its name is derived from adaptive moment estimation, and the reason it's called that is because Adam uses estimations of first and second moments of gradient to adapt the learning rate for each weight of the neural network. Now, what is moment ? N-th moment of a random variable is defined as the expected value of that variable to the power ofn.

More formally:

$$M=E(X^n)$$

M — moment, X -random variable.

4.1.5 OUTPUT FUNCTION - SOFTMAX

Before exploring the ins and outs of the Softmax activation function, we should focus on its building block—the sigmoid/logistic activation function that works on calculating probability values.

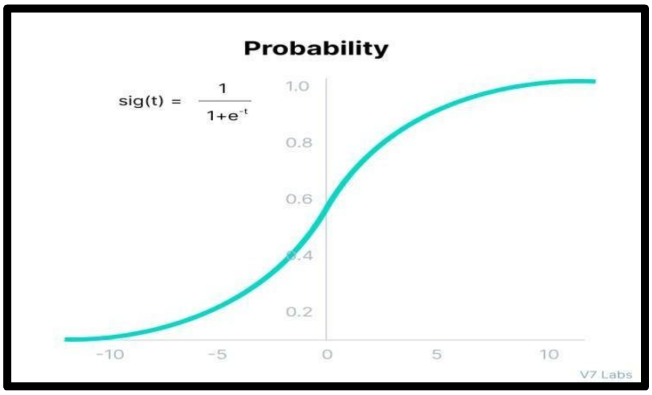


Fig : 3 Probability using softmax

The output of the sigmoid function was in the range of 0 to 1, which can be thought of as probability. You see, the Softmax function is described as a combination of multiple sigmoids. It calculates the relative probabilities. Similar to the sigmoid/logistic activation function, the SoftMax function returns the probability of each class.

It is most commonly used as an activation function for the last layer of the neural network in the case of multi-class classification.

Mathematically it can be represented as:

$$soft\ \max(z) = \exp(z) / \sum \exp(k)$$

4.2 OVERVIEW TECHNOLOGY - ANN

At earlier times, the conventional computers incorporated algorithmic approach that is the computer used to follow a set of instructions to solve a problem unless those specific steps need that the computer need to follow are known the computer cannot solve a problem. So, obviously, a person is needed in order to solve the problems or someone who can provide instructions to the computer so as to how to solve that particular problem. It actually restricted the problem-solving capacity of conventional computers to problems that we already understand and know how to solve.

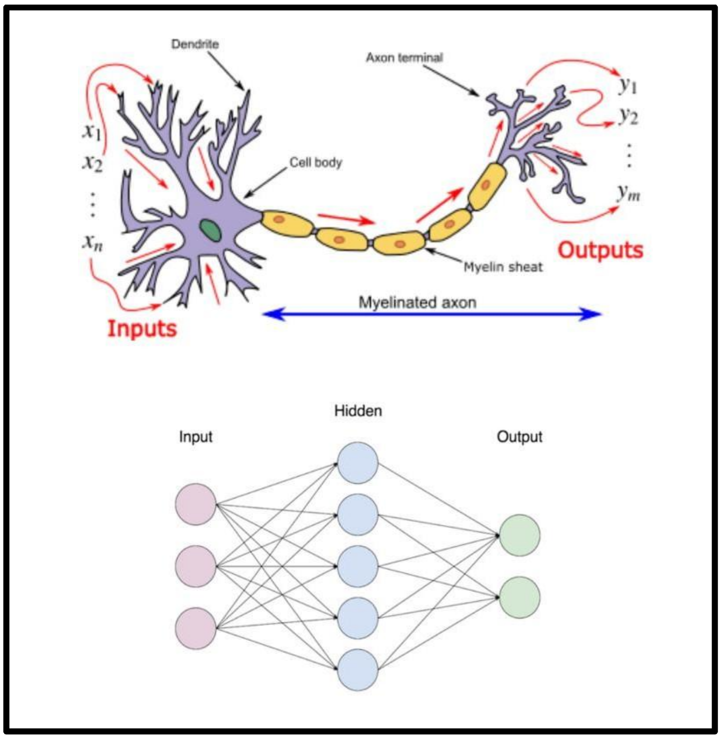
But what about those problems whose answers are not clear, so that is where our traditional approach face failure and so Neural Networks came into existence. Neural Networks processes information in a similar way the human brain does, and these networks actually learn from examples, you cannot program them to perform a specific task. They will learn only from past experiences as well as examples, which is why you don't need to provide all the information regarding any specific task. So, that was the main reason why neural networks came into existence.

"Artificial Neural Network is biologically inspired by the neural network, which constitutes after the human brain".

Neural networks are modeled in accordance with the human brain so as to imitate their functionality. The human brain can be defined as a neural network that is made up of several neurons, so is the Artificial Neural Network is made of numerous perceptron.

A neural network comprises of three main layers, which are as follows:

- **Input layer:** The input layer accepts all the inputs that are provided by the programmer.
- **Hidden layer:** In between the input and output layer, there is a set of hidden layers on which computations are performed that further results in the output.
- **Output layer:** After the input layer undergoes a series of transformations while passing through the hidden layer, it results in output that is delivered by the output layer.



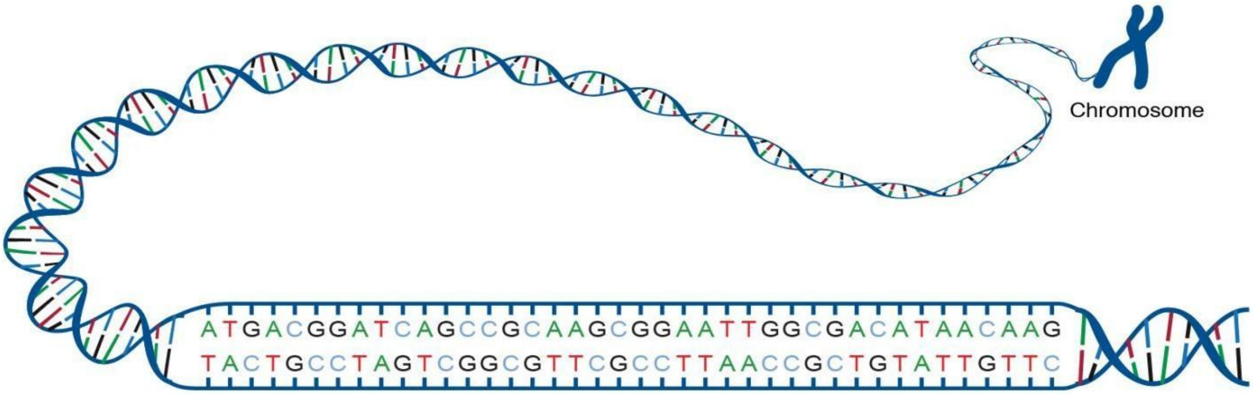
Artificial Neural Networks.

3. TESTING

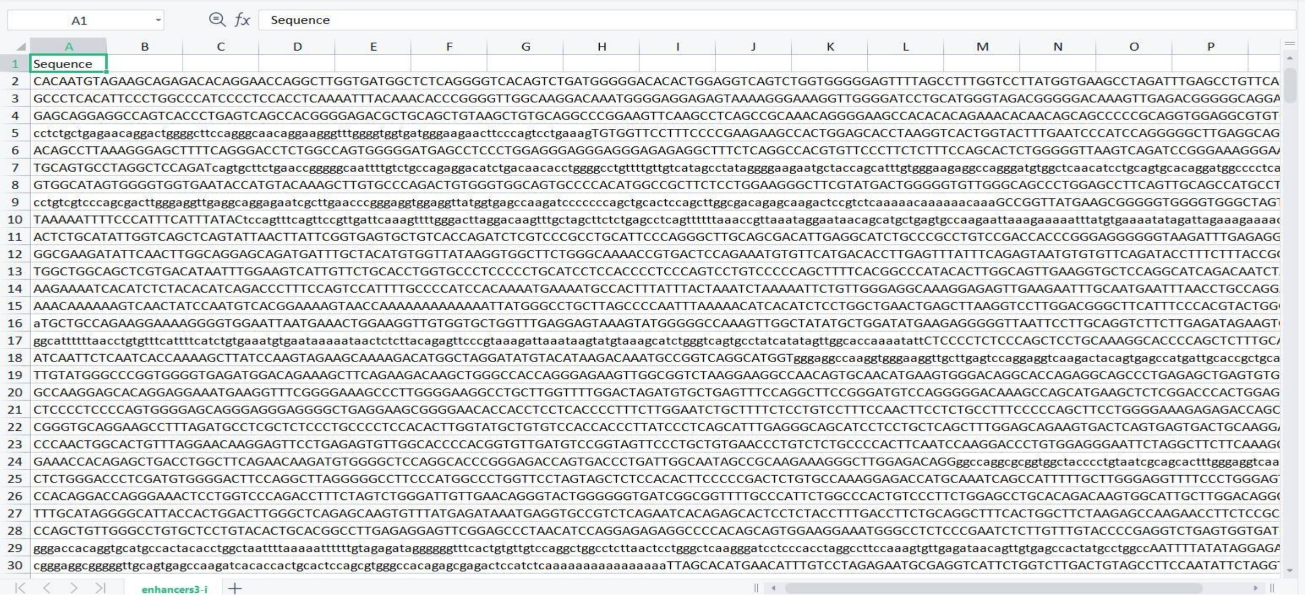
5.1 TEST CASES

ACGT sequences of enhancers or ACGT sequences of DNA belonging to cis - regulatory site : Enhancer.

ACGT is an acronym for the four types of bases found in a DNA molecule: adenine (A), cytosine (C), guanine (G), and thymine (T). A DNA molecule consists of two strands wound around each other, with each strand held together by bonds between the bases. Adenine pairs with thymine, and cytosine pairs with guanine. The sequence of bases in a portion of a DNA molecule, called a gene, carries the instructions needed to assemble a protein.


Chromosome
ATGACGGATCAGCCGCAAGCGGAATTGGCGACATAACAAG
TACTGCCTAGTCGGCGTTCGCCTTAACCGCTGTATTGTTC

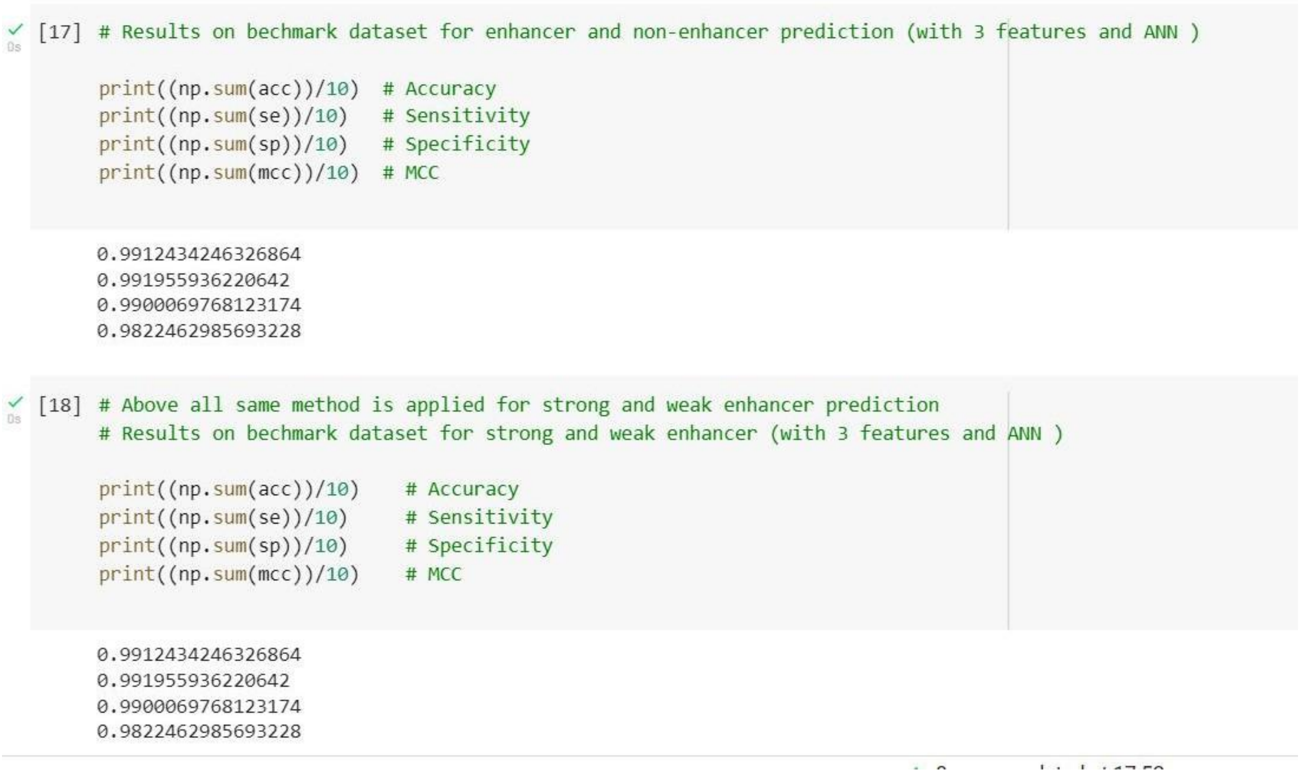
we have done our research on 1484 enhancer and non enhancer pairs.



	A
1	Sequence
2	CACAATGTAGAAGCAGAGACACAGGAACCAGGCTTGGTGATGGCTCTCAGGGGTCACAGTCTGATGGGGGACACACTGGAGGTCAGTCTGGTGGGGGAGTTTTAGCCTTTGGTCCTTATGGTGAAGCCTAGATTTGAGCCTGTTCA
3	GCCCTCACATTCCCTGGCCCATCCCCTCCACCTCAAAATTTACAAACACCCGGGGTTGGCAAGGACAAATGGGGAGGAGAGTAAAAGGGAAAGGTTGGGGATCCTGCATGGGTAGACGGGGGACAAAGTTGAGACGGGGGCAGGA
4	GAGCAGGAGGCCAGTCACCCTGAGTCAGCCACGGGGAGACGCTGCAGCTGTAAGCTGTGCAGGCCCGGAAGTTCAAGCCTCAGCCGCAAACAGGGGAAGCCACACACAGAAACACAACAGCAGCCCCCGCAGGTGGAGGCGTGT
5	cctctgctgagaacaggactggggcttccagggcaacaggaagggtttggggtggtgatgggaagaacttcccagtcctgaaagTGTGGTTCCTTTCCCCGAAGAAGCCACTGGAGCACCTAAGGTCACTGGTACTTTGAATCCCATCCAGGGGGCTTGAGGCAG
6	ACAGCCTTAAAGGGAGCTTTTCAGGGACCTCTGGCCAGTGGGGGATGAGCCTCCCTGGAGGGAGGGAGGGAGAGAGGCTTTCTCAGGCCACGTGTTCCCTTCTCTTTCCAGCACTCTGGGGGTTAAGTCAGATCCGGGAAAGGGA
7	TGCAGTGCCTAGGCTCCAGATcagtgcttctgaaccgggggcaattttgtctgccagaggacatctgacaacacctggggcctgttttgttgtcatagcctataggggaagaatgctaccagcatttgtgggaagaggccagggatgtggctcaacatcctgcagtgcacaggatggcccctca
8	GTGGCATAGTGGGGTGGTGAATACCATGTACAAAGCTTGTGCCCAGACTGTGGGTGGCAGTGCCCCACATGGCCGCTTCTCCTGGAAGGGCTTCGTATGACTGGGGGTGTTGGGCAGCCCTGGAGCCTTCAGTTGCAGCCATGCCT
9	cctgtcgtcccagcgacttgggaggttgaggcaggagaatcgcttgaacccgggaggtggaggttatggtgagccaagatccccccagctgcactccagcttggcgacagagcaagactccgtctcaaaaacaaaaaacaaaGCCGGTTATGAAGCGGGGGTGGGGTGGGCTAGT
10	TAAAAATTTTCCCATTTCATTTATACtccagtttcagttccgttgattcaaagttttgggacttaggacaagtttgctagcttctctgagcctcagtttttaaaccgttaaataggaataacagcatgctgagtgccaagaattaaagaaaaatttatgtgaaaatatagattagaaagaaaac
11	ACTCTGCATATTGGTCAGCTCAGTATTAACTTATTCGGTGAGTGCTGTCACCAGATCTCGTCCCGCCTGCATTCCCAGGGCTTGCAGCGACATTGAGGCATCTGCCCGCCTGTCCGACCACCCGGGAGGGGGGTAAGATTTGAGAGG
12	GGCGAAGATATTCAACTTGGCAGGAGCAGATGATTTGCTACATGTGGTTATAAGGTGGCTTCTGGGCAAAACCGTGACTCCAGAAATGTGTTCATGACACCTTGAGTTTATTTCAGAGTAATGTGTGTTCAGATACCTTTCTTTACCGC
13	TGGCTGGCAGCTCGTGACATAATTTGGAAGTCATTGTTCTGCACCTGGTGCCCTCCCCCTGCATCCTCCACCCCTCCCAGTCCTGTCCCCCAGCTTTTCACGGCCCATACACTTGGCAGTTGAAGGTGCTCCAGGCATCAGACAATCT
14	AAGAAAATCACATCTCTACACATCAGACCCTTTCCAGTCCATTTTGCCCCATCCACAAAATGAAAATGCCACTTTATTTACTAAATCTAAAAATTCTGTTGGGAGGCAAAGGAGAGTTGAAGAATTTGCAATGAATTTAACCTGCCAGG
15	AAACAAAAAAGTCAACTATCCAATGTCACGGAAAAGTAACCAAAAAAAAAAAAATTATGGGCCTGCTTAGCCCCAATTTAAAAACATCACATCTCCTGGCTGAACTGAGCTTAAGGTCCTTGGACGGGCTTCATTTCCCACGTACTGG
16	aTGCTGCCAGAAGGAAAAGGGGTGGAATTAATGAAACTGGAAGGTTGTGGTGCTGGTTTGAGGAGTAAAGTATGGGGGCCAAAGTTGGCTATATGCTGGATATGAAGAGGGGGTTAATTCCTTGCAGGTCTTCTTGAGATAGAAGT
17	ggcatttttaacctgtgtttcattttcatctgtgaaatgtgaataaaaataactctcttacagagttcccgtaaagattaaataagtatgtaaagcatctgggtcagtgcctatcatatagttggcaccaaaatattCTCCCCTCTCCCAGCTCCTGCAAAGGCACCCCAGCTCTTTGCA
18	ATCAATTCTCAATCACCAAAAGCTTATCCAAGTAGAAGCAAAAGACATGGCTAGGATATGTACATAAGACAAATGCCGGTCAGGCATGGTgggaggccaaggtgggaaggttgcttgagtccaggaggtcaagactacagtgagccatgattgcaccgctgca
19	TTGTATGGGCCCGGTGGGGTGAGATGGACAGAAAGCTTCAGAAGACAAGCTGGGCCACCAGGGAGAAGTTGGCGGTCTAAGGAAGGCCAACAGTGCAACATGAAGTGGGACAGGCACCAGAGGCAGCCCTGAGAGCTGAGTGTG
20	GCCAAGGAGCACAGGAGGAAATGAAGGTTTCGGGGAAAGCCCTTGGGGAAGGCCTGCTTGGTTTTGGACTAGATGTGCTGAGTTTCCAGGCTTCCGGGATGTCCAGGGGGACAAAGCCAGCATGAAGCTCTCGGACCCACTGGAG
21	CTCCCCTCCCCAGTGGGGAGCAGGGAGGGAGGGGCTGAGGAAGCGGGGAACACCACCTCCTCACCCCTTTCTTGGAATCTGCTTTTCTCCTGTCCTTTCCAACTTCCTCTGCCTTTCCCCCAGCTTCCTGGGGAAAGAGAGACCAGC
22	CGGGTGCAGGAAGCCTTTAGATGCCTCGCTCTCCCTGCCCCTCCACACTTGGTATGCTGTGTCCACCACCCTTATCCCTCAGCATTTGAGGGCAGCATCCTCCTGCTCAGCTTTGGAGCAGAAGTGACTCAGTGAGTGACTGCAAGGA
23	CCCAACTGGCACTGTTTAGGAACAAGGAGTTCCTGAGAGTGTTGGCACCCCACGGTGTTGATGTCCGGTAGTTCCCTGCTGTGAACCCTGTCTCTGCCCCACTTCAATCCAAGGACCCTGTGGAGGGAATTCTAGGCTTCTTCAAAGC
24	GAAACCACAGAGCTGACCTGGCTTCAGAACAAGATGTGGGGCTCCAGGCACCCGGGAGACCAGTGACCCTGATTGGCAATAGCCGCAAGAAAGGGCTTGGAGACAGGggccaggcgcggtggctacccctgtaatcgcagcactttgggaggtcaa
25	CTCTGGGACCCTCGATGTGGGGACTTCCAGGCTTAGGGGGCCTTCCCATGGCCCTGGTTCCTAGTAGCTCTCCACACTTCCCCCGACTCTGTGCCAAAGGAGACCATGCAAATCAGCCATTTTTGCTTGGGAGGTTTTCCCTGGGAGT
26	CCACAGGACCAGGGAAACTCCTGGTCCCAGACCTTTCTAGTCTGGGATTGTTGAACAGGGTACTGGGGGGTGATCGGCGGTTTTGCCCATTCTGGCCCACTGTCCCTTCTGGAGCCTGCACAGACAAGTGGCATTGCTTGGACAGGC
27	TTTGCATAGGGGCATTACCACTGGACTTGGGCTCAGAGCAAGTGTTTATGAGATAAATGAGGTGCCGTCTCAGAATCACAGAGCACTCCTCTACCTTTGACCTTCTGCAGGCTTTCACTGGCTTCTAAGAGCCAAGAACCTTCTCCGC
28	CCAGCTGTTGGGCCTGTGCTCCTGTACACTGCACGGCCTTGAGAGGAGTTCGGAGCCCTAACATCCAGGAGAGAGGCCCCACAGCAGTGGAAGGAAATGGGCCTCTCCCGAATCTCTTGTTTGTACCCCGAGGTCTGAGTGGTGAT
29	gggaccacaggtgcatgccactacacctggctaattttaaaaatttttgtagagatagggggtttcactgtgttgtccaggctggcctcttaactcctgggctcaagggatcctcccacctaggccttccaaagtgttgagataacagttgtgagccactatgcctggccAATTTTATATAGGAGA
30	cgggaggcgggggttgcagtgagccaagatcacaccactgcactccagcgtgggccacagagcgagactccatctcaaaaaaaaaaaaaaaaTTAGCACATGAACATTTGTCCTAGAGAATGCGAGGTCATTCTGGTCTTGACTGTAGCCTTCCAATATTCTAGGT

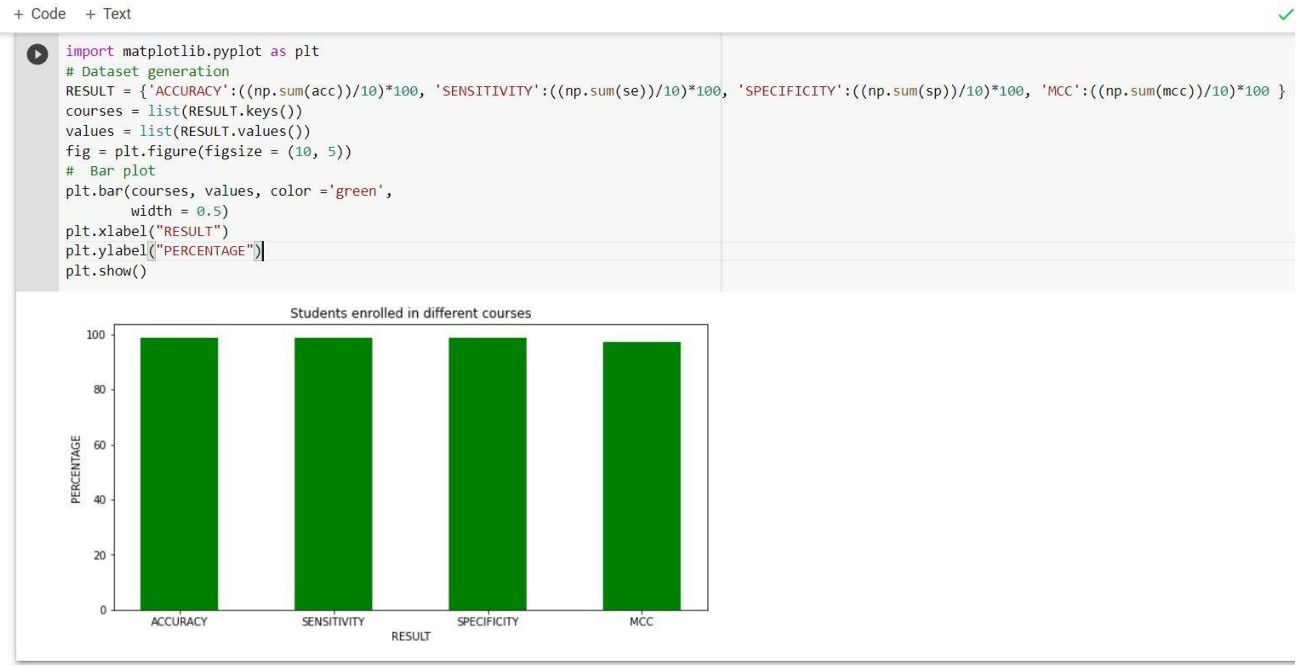
4. RESULTS

Our model accounts for a whopping accuracy on 99% when applied for Enhancer dataset file containing benchmark dataset of enhancers.



```
[17] # Results on bechmark dataset for enhancer and non-enhancer prediction (with 3 features and ANN )

     print((np.sum(acc))/10)  # Accuracy
     print((np.sum(se))/10)   # Sensitivity
     print((np.sum(sp))/10)   # Specificity
     print((np.sum(mcc))/10)  # MCC

     0.9912434246326864
     0.991955936220642
     0.9900069768123174
     0.9822462985693228
```

```
[18] # Above all same method is applied for strong and weak enhancer prediction
     # Results on bechmark dataset for strong and weak enhancer (with 3 features and ANN )

     print((np.sum(acc))/10)    # Accuracy
     print((np.sum(se))/10)     # Sensitivity
     print((np.sum(sp))/10)     # Specificity
     print((np.sum(mcc))/10)    # MCC

     0.9912434246326864
     0.991955936220642
     0.9900069768123174
     0.9822462985693228
```



```
import matplotlib.pyplot as plt
# Dataset generation
RESULT = {'ACCURACY':((np.sum(acc))/10)*100, 'SENSITIVITY':((np.sum(se))/10)*100, 'SPECIFICITY':((np.sum(sp))/10)*100, 'MCC':((np.sum(mcc))/10)*100 }
courses = list(RESULT.keys())
values = list(RESULT.values())
fig = plt.figure(figsize = (10, 5))
#  Bar plot
plt.bar(courses, values, color ='green',
        width = 0.5)
plt.xlabel("RESULT")
plt.ylabel("PERCENTAGE")
plt.show()
```

5. CONCLUSION

Therefore, genome signal prediction using machine learning is efficient and accurate to predict the genome pattern. Bio informatics gives a good scope to relate the concepts of biology to Artificial intelligence to personalize various aspects of gene sequencing predictions.

Similarly, genome signal prediction works by applying various concepts of support vector machine(SVM), Artificial Neural network (ANN), decision trees, random forest algorithm on the genome pattern to give out the transcription factor site prediction for enhancers. Our project has future scope to study how and why enhancers function to synthesize the RNA to determine various structural and functional elements of body's growth and development.

Enhancers are cis regular sites responsible for binding with binding factor and produce polypeptides to the human body.

Our thus developed model accounts for

ACCURACY : 99.12%
SENSITIVITY : 99.19%
SPECIFICITY : 99.00%
MATHEWS CO-RELATION COEFFICIENT (MCC) : 98.22%

6. FUTURE SCOPE

- Our project has future scope to study how and why enhancers function to synthesize the RNA to determine various structural and functional elements of body's growth and development.
- This can be further incorporated for promoters too.
- This area of research could be so useful to understand genome transcriptions of specific kind which are significantly responsible for diseases like cancer.
- Useful for practical laboratory synthesis of protein of specific kind and functionality.

7. BIBILIOGRAPHY

1. Y. H. Li, C. Y. Yu, X. X. Li et al., “Therapeutic target database update 2018: enriched resource for facilitating bench-to-clinic research of targeted therapeutics,” *Nucleic Acids Research*, vol. 46, no. D1, pp. D1121–D1127, 2018.View at: Publisher Site | Google Scholar

2. B. Li, J. Tang, Q. Yang et al., “NOREVA: normalization and evaluation of MS-based metabolomics data,” *Nucleic Acids Research*, vol. 45, no. W1, pp. W162–W170, 2017.View at: Publisher Site | Google Scholar

3. J. Fu, J. Tang, Y. Wang et al., “Discovery of the consistently well-performed analysis chain for SWATH-MS based pharmacoproteomic quantification,” *Frontiers in Pharmacology*, vol. 9, p. 681, 2018.View at: Publisher Site | Google Scholar

4. H. Bu, J. Hao, J. Guan, and S. Zhou, “Predicting Enhancers from Multiple Cell Lines and Tissues across Different Developmental Stages Based On SVM Method,” *Current Bioinformatics*, vol. 13, no. 6, pp. 655–660, 2018.View at: Publisher Site | Google Scholar

5. B. Liu, L. Fang, R. Long, X. Lan, and K. C. Chou, “iEnhancer-2L: a two-layer predictor for identifying enhancers and their strength by pseudo k-tuple nucleotide composition,” *Bioinformatics*, vol. 32, no. 3, pp. 362–369, 2016.View at: Publisher Site | Google Scholar

6. C. Jia and W. He, “EnhancerPred: a predictor for discovering enhancers based on the combination and selection of multiple features,” *Scientific Reports*, vol. 6, no. 1, p. 38741, 2016.View at: Publisher Site | Google Scholar

7. B. Liu, K. Li, D. S. Huang, and K. C. Chou, “iEnhancer-EL:identifying enhancers and their strength with ensemble learning approach,” *Bioinformatics*, vol. 34, no. 22, pp. 3835–3842, 2018.View at: Publisher Site | Google Scholar

8. J. Tang, J. Fu, Y. Wang et al., “Simultaneous improvement in the precision, accuracy, and robustness of label-free proteome quantification by optimizing data manipulation chains,” *Molecular & Cellular Proteomics*, vol. 18, no. 8, pp. 1683–1699, 2019.View at: Publisher Site | Google Scholar

9. Q. H. Nguyen, T. H. Nguyen-Vo, N. Q. K. le, T. T. T. Do, S. Rahardja, and B. P. Nguyen, “iEnhancer-ECNN: identifying enhancers and their strength using ensembles of convolutional neural networks,” *BMC Genomics*, vol. 20, Suppl 9, p. 951, 2019.View at: Publisher Site | Google Scholar

10. W. Xue, F. Yang, P. Wang et al., “What contributes to serotonin-norepinephrine reuptake inhibitors’ dual-targeting mechanism? The key role of transmembrane domain 6 in human serotonin and norepinephrine transporters revealed by molecular dynamics simulation,” *ACS Chemical Neuroscience*, vol. 9, no. 5, pp. 1128–1140, 2018.View at: Publisher Site | Google Scholar