Identification of Active Oligonucleotide Sequences Using Artificial Neural Network

Alex Luke[1], Sarah Fergione[1], Riley Wilson[1], Brady Gunn[1],
Missouri Western State University,
Department of Chemistry
4525 Downs Drive
St. Joseph, MO 64507
aluke2, sfergione, rwilson26, bgunn1,
@missouriwestern.edu

Stan Svojanovsky[2]
Missouri Western State University
Department of Chemistry
4525 Downs Drive
St. Joseph, MO 64507
ssvojano@missouriwestern.edu

ABSTRACT

In this project we designed an Artificial Neural Network (ANN) computational model to predict the activity of short oligonucleotide sequences (octamers) with important biological role as exonic splicing enhancers (ESE) motifs recognized by human SR protein SC35. Since only active sequences were available from the literature as our initial data set, we generated an additional set of complementary sequences to the original set. We used back-propagation neural network (BPNN) with MATLAB® Neural Network Toolbox™ on our research designated computer. In Stage I of our project we trained, validated and tested the BPNN prototype. We started with 20 samples in the training and 8 samples in the validation sets. Trained and validated BPNN prototype was then used to test the unique set of 10 octamer sequences with 5 active samples and their 5 complementary sequences. The test showed 2 classification errors, one false positive and the other false negative. We used the test data and moved into Stage II of the project. First, we analyzed the initial DNA numerical representation (DNR) and changed the scheme to achieve higher difference between the subsets of active and complementary sequences. We compared the BPNN results with different numbers of nodes in the second hidden layer to optimize model accuracy. To estimate future model performance we needed to test the classifier on newly collected data from another paper. This practical application included the testing of 41 published, non-repeating SC35 ESE motif octamers, together with 41 complementary sequences. The test showed high BPNN accuracy in the predictive power for both (active and inactive) categories.

This study shows the potential for using a BPNN to screen SC35 ESE motif candidates.

Categories and Subject Descriptors

J.3 [Life and Medical Science]: Biology and genetics.

DOI: https://doi.org/10.22369/issn.2153-4136/9/2/4

Keywords

Artificial neural network (ANN), back-propagation neural network (BPNN), nucleotide sequences, exonic splicing enhancers (ESE), DNA numerical representation (DNR).

1. INTRODUCTION

1.1 Artificial Neural Networks (ANN)

Over the past few decades, machine learning processes have become more sophisticated and useful in many different fields of theoretical and applied science, such as applied biology, biomedical research, medicine, and drug discoveries. These methods are based on pattern recognition capabilities [1, 2].

The new and more advanced applications of these models now achieved a major growing momentum.

They are now incorporated in text (spam filtering) and voice recognition (Alexa, Siri and Cortana), virtual video games, self-driving cars, economic forecasting, health related scans and images to reveal any abnormal patterns related to different symptoms and many other fields.

Among other computer-assisted approaches such as machine-learning Decision Trees and Nearest Neighbors algorithms, the ANN–based schemes have gained probably the most attention and are now widely applied.

The initial information (signal) is entering the network of 'neurons' called nodes that is programmed to react to this initial signal and passed the transformed signal to other cluster of nodes so that other signal transformation could be performed. Part of the ANN design is to assign a finite number of these clusters (layers) together with the number of nodes in each layer. The general process of turning the initial input into the output information is the result of ANN program and model design. So, the computer is actually allowed to 'learn' specific information by repeating the very same process, and adjusting the connections intensity between the nodes till the required output is reached. ANNs are then used to solve the problems that are too difficult for both: people and our digital computers. Since these models work on pattern recognition they do not need any underlying data distribution function that is usually required prior to any statistical data analysis and the requirement of data normality before hypotheses testing.

[1] Undergraduate Student

[2] Corresponding Author

1.2 Biological Aspects

Literature [3, 4] and personal communication are the sources of active oligonucleotide sequences (class=1) used in this project.

The authors used the SELEX [5] method to generate a set of sequences with 8 nucleotides (octamers) that were originally evaluated by calculated scores.

Only unique octamers, each with the non-repeating sequence pattern were used in our project. Nucleotide frequencies of a single position of each individual active sequence were then combined into score matrix resulting in an assembly of more general, biologically active SC35 motif [GGCCCCTG] called consensus sequence that we also incorporated into our BPNN model.

These active octamers (also named SC35 ESE motifs) play a major biological role during the process of exon splicing process as exonic splicing enhancers (ESE) that are recognized by human SR protein SC35. This protein is responsible for splicing of another enzyme called pyruvate dehydrogenase (PDH). Any significant deficiency in the process of producing PDH complex is a major cause of lactic acidosis and mental retardation in childhood. SR proteins are involved in proper RNA splicing. They are named SR since this family of proteins is rather conserved and contains many repeats of serine (S) and arginine (R) amino acids [6].

1.3 Goals

The major objective of our project was to apply ANN concept and design the back-propagation neural network (BPNN) on available SC35 ESE motifs. DNA numerical representation (DNR) scheme was then applied to encode the nucleotide bases into numerical values representing each sequence. The set of signals was normalized and partitioned into two major subgroups:

1. training and validation (train+val) subsets
2. testing (test) subset

Both of these subsets contained only unique signals, i.e. none of the test sequences were included in train+val subsets and vice versa.

If the ANN prototype shows high accuracy in sequence classification into active (1) and non-active (0) groups then it might be potentially used as the screening tool for SC35 ESE motifs.

2. METHODS

The very first step was to extract active unique sequences in their letter description format as shown in Tables 1 and 2. It means the sequences of 8 letters combination of A, C, G, and T described as SC35 ECE motif. The letter format represents different types of nucleotides based on their chemical structure and biochemical properties:

A = adenine
C = cytosine
G = guanine
T = thymine

Computer-assisted BPNN is usually considered at least 2-class pattern recognition system with one class representing active (1) feature vectors and the other class holding the non-active (0) feature vectors. In order to satisfy these criteria and make balanced model we generated the matrices with complementary sequences representing non-active output. It is based on the general biological rule that complementary sequences would not fit as SC35 ESE motifs. This process was a part of our BPNN script, so the complementary matrix was computationally generated according to the basic biology principles, where G is the complementary (or antisense) base of C and A is a complement to T.
The conversion could be expressed by G ↔ C and A ↔ T.

We started with 20 active sequences in training and 8 active octamers in the validation set with generated complementary non-active sequences as shown in Tables 1 and 2.

Table 1. Training set of 20 unique sequences

ID	Active (1)	Non-active (0)
1.	GATCCCCG	CTAGGGGC
2.	GGCTCGTG	CCGAGCAC
3.	GGCCGCAG	CCGGCGTC
4.	GGCCCACA	CCGGGTGT
5.	GGTTGGCG	CCAACCGC
6.	GTCCTCCG	CAGGAGGC
7.	GTCCCCTG	CAGGGGAC
8.	GTTCTGTA	CAAGACAT
9.	GAATACCG	CTTATGGC
10.	GGACCGTA	CCTGGCAT
11.	GTCTAACG	CAGATTGC
12.	AGCCTCAG	TCGGAGTC
13.	GGATGGAG	CCTACCTC
14.	GGACTGTA	CCTGACAT
15.	GGTTGTTG	CCAACAAC
16.	GAGCACTG	CTCGTGAC
17.	TGTTACTA	ACAATGAT
18.	GGCTCCAA	CCGAGGTT
19.	GGATCCGG	CCTAGGCC
20.	GACCTGCT	CTGGACGA

Table 2. Validation set of 8 unique sequences

ID	Active (1)	Non-active (0)
1.	GTTTCGAG	CAAAGCTC
2.	GGTCGCCG	CCAGCGGC
3.	GGTCAGTG	CCAGTCAC
4.	GGCTGATG	CCGACTAC

5.	CGCCCTTG	GCGGGAAC
6.	AGCTCCCA	TCGAGGGT
7.	GACCGGTG	CTGGCCAC
8.	GACTAGAA	CTGATCTT

In any machine learning process, DNA sequence are converted to numerical values for data representation and feature learning related to specific biological or biochemical application. The distinct nature of the DNA sequence being discrete in the 'amplitude' and 'time' offers multiple DNA numerical representation (DNR) techniques in the form of single or multidimensional array. Current DNR techniques could be divided into three main categories: single-value mapping, multidimensional sequence mapping, and cumulative sequence mapping [7].

Integer, real number, and measurement representations are still frequently used encoding schemes. In many scenarios of single-value mapping A, C, G, and T are assigned to a single indicator such as 1, 2, 3, and 4. This scheme (also called Galois field) is also feasible for a complementary encoding because it provides symmetric deviations between both groups. Also, it was used in the past for DNA barcode in large-scale screening of multiple genomic core databases. Other direct encoding schemes include Atomic representation, where each nucleotide is assigned its atomic number (i.e. number of protons) [C=58, T=66, A=70, and G=70]. Calculated electron energies for each nucleotide [C=0.1340, T=0.1335, A=0.1260 and G=0.0806] are the core of Electron-Ion Interaction Pseudopotential (EIIP) single-value scheme, while the Molecular Mass encoding is applied in mapping DNA sequences based on molecular mass of different nucleobases [C=110, T=125, A=134, and G=150] with atomic mass units.

Multidimensional sequence mapping include binary sequence indicators such as A=[00], C=[11], G=[10], and T=[01]; 4-bit representation with A=[1000], C=[0100], G=[0010], and T=[0001].

Cumulative representation include Z-curve, DNA walk and other more complex DNA encoding schemes.

Currently, no DNR is considered to be the 'gold standard' and the choice is usually driven by the applicable biological aspects and the specific goals of the machine learning project.

We selected direct, single-mapping Galois field encoding method because it provides uniform distance between active and non-active (complementary) sequences with symmetric deviations. Other advantage is to use simple barcode method to label each sequence for automated sequence screening. It also supports our biological goals of the project to separate the signals for active and non-active octamers. However, this structure might imply that pyrimidines (C and T) are in some respect 'greater than' purines (A and G), which is a disadvantage of this encoding method.

Table 3 represents 10 octamers that we used to test BPNN model. This is a data set of unique sequences with known activities. Five of them are active and five of them are from the non-active group. None of these sequences were previously used in training and validation subset. Active samples (1, 4, 5 and 10) are from the published article [3], sample 9 was added based on the private communication [Luke, personal communication].

Table 3. Testing subset of 10 unique sequences

ID	Designation	Class
1.	GATCGCTG	1
2.	AGTCGGAT	0
3.	CTCATTGC	0
4.	GGCCCCTG	1
5.	GGACGCTG	1
6.	CCGGGGAC	0
7.	CTAGCGAC	0
8.	CCTGCGAC	0
9.	TCAGCCTA	1
10.	GAGTAACG	1

Figure 1 shows the general ANN based on one-layer hidden units, where all nodes have the same number of weights (synapses) and all receive the input signal simultaneously.

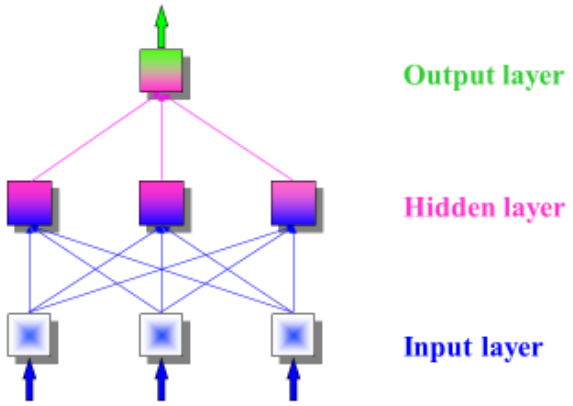
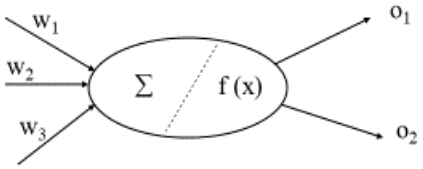


Figure 1. General assembly of neural network processing

Action of formal neuron (node) consists in summing all weighted inputs (w_i) transformed via activation function into output signals (o_j). BPNN default is the sigmoid function.



$$f(x) = 1/[1+\exp(-x)]$$

Figure 2. ANN node action with sigmoid transformation function

3. RESULTS AND DISCUSSION

3.1 Project Stage I

The initial step in BPNN design was to generate Galois field numerical encoding for A = 1, T = 2, C = 3, and G = 4.
Active sequences were added into BPNN MATLAB script with activity equal to 1. The next part of the script generated the complementary, non-active sequences that were used to balance the BPNN model. All data went through the normalization into [0, 1] interval across each feature matrix.

In this project we used a supervised training where both the input signal and the output activity are provided. The network transforms the inputs with connection weights through the nodes and layers and calculate the errors between the resulting and desired outputs. Errors are then propagated back through the network to adjust the weights which control the network assembly. During this learning process the training data set is processed many times as the connection weights are continually adjusted and finally refined.

Validation process that is parallel to training enables to validate the final model specification with the validation data set. The model is trained on the training set and the error is calculated on the validation set multiple times while adjusting the weights. It is used to analyze the value of parameters in the model which usually results in less error on validation set.

Testing provides then an unbiased evaluation of a final model fit on the training dataset.

For our BPNN model we used the seed for the random number generator applied for the initial weights to be equal 1.

BPNN component was applied with multiple variables:

- Convergence error (SSE): usually about 0.0001
- Number of iterations: 100

Samples not previously included in training process were used for the validation.
Finally, we tested the BPNN classifier with test data set (i.e. 10 unique octamers with known output 0 or 1) in specific model conditions with 8 nodes in the first layer and 6 nodes in the second hidden layer.

Table 4. BPNN classification of test sequences

ID	Letter Designation	Known classification	BPNN classification
1.	GATCGCTG	1	0.9935
2.	AGTCGGAT	**0**	**0.8843**
3.	CTCATTGC	0	0.0046
4.	GGCCCCTG	1	0.9934
5.	GGACGCTG	1	0.9949
6.	CCGGGGAC	0	0.0028
7.	CTAGCGAC	0	0.0043
8.	CCTGCGAC	0	0.0044
9.	TCAGCCTA	**1**	**0.0027**
10.	GAGTAACG	1	0.9958
Training error			0.0144
Validation1 error			0.9769
Validation0 error			0.9821
Testing error			2

Classification with the BPNN model under the specific conditions revealed 2 errors. Non-active sample 2 was predicted to be active (false positive), while the active sequence 9 was misplaced by BPNN model into the cluster of non-active sequences (false negative). We were not satisfied with model performance and moved into Stage II of the project.

3.2 Project Stage II

We started this stage with graphical interpretation of active and non-active feature vectors that provided the partial key to the problem. Our computer script generated the matrix of complementary (non-active) sequences based on the given instructions with the application of the existing biology rules. Our complementary sequences were generated with the absolute difference of 1 between nucleobases A and T and C and G.

$$| A - T | = | 1 - 2 | = | C - G | = | 3 - 4 | = 1$$

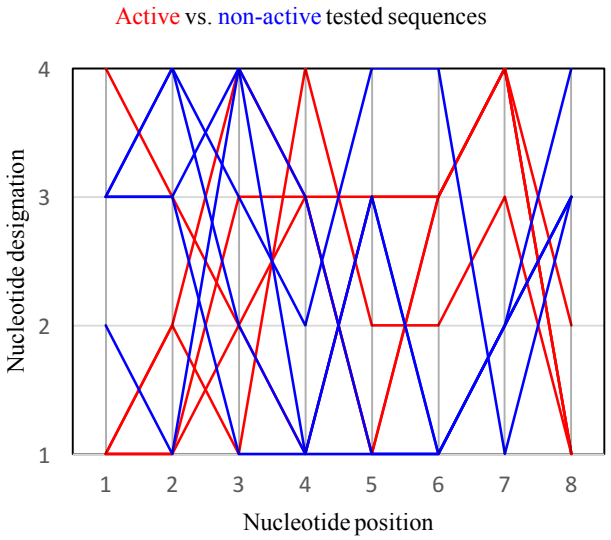
The data analysis of the initial train+val sequence subsets showed that the majority of the active sequences (86%) started with the first nucleotide G (C for complementary sequences). All tested sequences starting with G or C were then correctly classified by BPNN model. However, in the initial train+val subsets we also had total of 3 sequences starting with A nucleotide (2 active and 1 non-active). The active sequences started with AG and non-active AC dinucleotide. Size limitation of the training set could be the potential reason of the lower performance of our BPNN model resulting in misclassification of 2 tested sequences.

In attempt to reduce misclassification error we applied higher resolution between the active and non-active categories to further separate both of these subsets in their space. We went back and changed the initial single-value DNR scheme to achieve higher and constant difference between both groups.

If A = 2, T = 4, C = 3, and G = 1, then

$$| A - T | = | 2 - 4 | = \mathbf{2} \text{ and } | C - G | = | 3 - 1 | = \mathbf{2}$$

We also created a 2-D distribution chart to differentiate between active and non-active categories. Graph 1 displayed a complete overlap of both groups at position 3 and some partial overlaps at positions 2, 4, and 5, respectively.



Graph 1. Distribution of 10 tested sequences

In the following step of Stage II we tried different numbers of nodes in the second hidden layer in order to find an optimal estimate.
The results are summarized in Table 5 together with training, validation, and test errors.

Table 5. BPNN outputs for tested sequences with variable number of nodes in the second hidden layer

ID	Class	# nodes 1	# nodes 2	# nodes 3	# nodes 4
1.	1	0.9946	0.9964	0.9963	0.9970
2.	0	0.0589	0.1048	0.0371	0.0291
3.	0	0.0091	0.0064	0.0014	0.0033
4.	1	0.9948	0.9966	0.9970	0.9967
5.	1	0.9940	0.9963	0.9967	0.9964
6.	0	0.0075	0.0043	0.0013	0.0023
7.	0	0.0077	0.0050	0.0013	0.0028
8.	0	0.0084	0.0045	0.0013	0.0035
9.	1	**0.0094**	**0.0051**	**0.0014**	**0.0058**
10.	1	0.9930	0.9971	0.9967	0.9969
Training error		0.0292	0.0104	0.0125	0.0113
Validation1 error		0.0086	0.0054	0.0063	0.0048
Validation0 error		0.0163	0.0114	0.0020	0.0082
Testing error		1	1	1	1
ID	**Class**	**# nodes 5**	**# nodes 6**	**# nodes 7**	**# nodes 8**
1.	1	0.9962	0.9969	0.9891	0.9951
2.	0	0.3789	0.0129	0.4544	0.0792
3.	0	0.0070	0.0042	0.0048	0.0051
4.	1	0.9972	0.9968	0.9894	0.9955
5.	1	0.9966	0.9971	0.9894	0.9951
6.	0	0.0040	0.0043	0.0044	0.0053
7.	0	0.0042	0.0041	0.0045	0.0048
8.	0	0.0049	0.0044	0.0055	0.0059
9.	1	**0.0043**	**0.0047**	**0.0057**	**0.0070**
10.	1	0.9974	0.9963	0.9896	0.9948
Training error		0.0141	0.0084	0.0170	0.0216
Validation1 error		0.0062	0.0046	0.0125	0.0082
Validation0 error		0.0087	0.0078	0.0121	0.0141
Testing error		1	1	1	1

Based on calculated training, validation, testing errors and the BPNN overall performance, the optimal estimate is represented by 6 nodes in the second hidden layer.

Variables of the optimal BPNN prototype:

- Convergence error (SSE): **0.0001**
- Number of iterations: **100**
- Number of nodes in the first layer: **8**
- Number of nodes in second (hidden) layer: **6**

3.3 Testing larger database

We used Stage I test data to initiate Stage II and to optimize the number of nodes in the second hidden layer, so the test performance is likely an optimal estimate. To evaluate future performance, we needed to test the classifier on newly collected data from another paper [6]. The authors provided the list of 128 active SC35 ESE motif sequences specifically arranged by different tissues, genes, and selected organs. They proposed highly conserved SC35 motif between tissues, among different genes, and within the same chromosome. They showed a slight variation in the SC35 ESE sequence motif among human chromosomes, with the conserved G nucleotide at the very first position of all active sequences.

The set included multiple sequence duplicates as they occurred in several tissues and various genes, and chromosomes. Prior to the test we removed all duplicates (87 sequences) and used the total of 41 unique active sequences together with 41 complementary non-active sequences with our optimal BPNN classifier. Again, none of these tested sequences were included in our BPNN train+val sets.

Model classification, together with training, validation and test errors are summarized in Table 6.

Table 6. Prediction for 41 active and complementary sequences with the optimal BPNN model

ID	Class (1)	BPNN	Class (0)	BPNN
1.	GACCCCTG	0.9917	CTGGGGAC	0.0039
2.	GACCTCTG	0.9916	CTGGAGAC	0.0034
3.	GACCACTG	0.9917	CTGGTGAC	0.0027
4.	GATCACTG	0.9920	CTAGTGAC	0.0033
5.	GATCCCTG	0.9922	CTAGGGAC	0.0050
6.	GGCCCCTG	0.9922	CCGGGGAC	0.0053
7.	GGCTCCTG	0.9920	CCGAGGAC	0.0122
8.	GACTCCTG	0.9920	CTGAGGAC	0.0058
9.	GACTCCCG	0.9917	CTGAGGGC	0.0048
10.	GACCCCCG	0.9917	CTGGGGGC	0.0035

11.	GACCACCG	0.9922	CTGGTGGC	0.0025
12.	GGCCCCCG	0.9913	CCGGGGGC	0.0046
13.	GGCCTCTA	0.9921	CCGGAGAT	0.0032
14.	GGCCTCTG	0.9913	CCGGAGAC	0.0047
15.	GGCCTCCA	0.9921	CCGGAGGT	0.0029
16.	GGCCTCCG	0.9915	CCGGAGGC	0.0041
17.	GGCCCCTA	0.9907	CCGGGGAT	0.0036
18.	GTCTCCTG	0.9888	CAGAGGAC	0.0433
19.	GTCCCCTA	0.9923	CAGGGGAT	0.0090
20.	GGCTCCAG	0.9922	CCGAGGTC	0.0205
21.	GGCCCCAG	0.9915	CCGGGGTC	0.0068
22.	GGCCCCCA	0.9924	CCGGGGGT	0.0032
23.	GGCTACTG	0.9920	CCGATGAC	0.0121
24.	GGCTTCTG	0.9925	CCGAAGAC	0.0119
25.	GGCTGCTG	0.9922	CCGACGAC	0.0118
26.	GGCCACTG	0.9922	CCGGTGAC	0.0039
27.	GGCCGCTG	0.9922	CCGGCGAC	0.0042
28.	GGCTCCTA	0.9916	CCGAGGAT	0.0057
29.	GGCTCCCG	0.9923	CCGAGGGC	0.0093
30.	GGCTCCCA	0.9917	CCGAGGGT	0.0047
31.	GACTCCCA	0.9912	CTGAGGGT	0.0032
32.	GATTTCCG	0.9921	CTAAAGGC	0.0059
33.	GATTCCCG	0.9923	CTAAGGGC	0.0065
34.	GACTTCCG	0.9917	CTGAAGGC	0.0044
35.	GACCTCCG	0.9916	CTGGAGGC	0.0031
36.	GACCTCCA	0.9904	CTGGAGGT	0.0024
37.	GACCTCTA	0.9904	CTGGAGAT	0.0026
38.	GACCCCCA	0.9906	CTGGGGGT	0.0027
39.	GACCCCTA	0.9907	CTGGGGAT	0.0029
40.	GACTTCTG	0.9916	CTGAAGAC	0.0052
41.	GGCCTCAG	0.9920	CCGGAGTC	0.0063
Training error			0.0205	
Validation1 error			0.0107	
Validation0 error			0.8008	
Testing error			0	

The test confirmed that the BPNN prototype satisfactory distinguishes between all 41 proposed SC35 ESE active motifs and their compliments with high accuracy in BPNN classification performance.

4. CONCLUSION

In our research project we used ANN script to construct a functional back-propagation neural network (BPNN) model. We designed this model in order to classify the short oligonucleotide sequences with 8 nucleotide elements (octamers) into two categories: active (1) and non-active (0) clusters. The visual interpretation of the data (Graph 1) shows some partial overlaps of both groups on multiple feature vector elements, which supports our decision to apply neural network concept. Statistical data analysis requires a prior knowledge of data distribution, which could be very complex in case of any overlap. Also, all elements of the feature vector are discrete values in relatively small data set which will most likely require non-parametric statistical analysis.

We used single-value scheme to encode sequence letter description into numerical designation. The model was trained with 20 active sequences and validated with the set of 8 active sequences. In order to keep the model balanced the complementary, non-active sequences were generated. The initial virtual screen included 10 unique sequences from the testing data set (5 active and 5 non-active sequences) used to assess the model accuracy and overall performance. After the BPNN model update we tried different number of nodes (1-8) in the second hidden layer to determine the optimal model.

We tested our optimal BPNN prototype on larger data set of 82 unique (41 active and 41 non-active) sequences and the results of the data classification revealed high model accuracy for this data set.

5. FUTURE WORK

For future work we could test any proposed CS35 ESE motif candidate or use the BPNN prototype to screen any sequence database for a potential match. We might also draw random biological sequences that are not known to be SC35 ESE motif candidates and detect how many of them are classified by BPNN as active.

The initial published data were listed with their scores that were calculated using a score matrix. Another type of future work would be to incorporate this information into our model, i.e. not just to classify the data into active and non-active subsets but add some degree to the activity and answer the question: "If active, how much activity is predicted?"

Also, it would be beneficial to create and compare additional classification prototypes based on different DNA numerical representation (DNR) methods such as binary indicators and OneHot Encoder and additional classification procedures such as decision trees or k-nearest neighbor algorithm.

6. REFLECTIONS

The project described in this paper was the very first research project for all undergraduate students in my research group. They all actively participated on this project as each of them designed their own ANN model. The major attraction for all students was the introduction of artificial intelligence in the computer-assisted model and the practical application of the BPNN prototype on real SC35 ESE motif sequences.

This project provided the students with multiple opportunities to participate on each stage of the project, starting with the literature research, learning the basics of MATLAB computing together with Neural Network Toolbox, join the time consuming journey to design the proper ANN model through the training, validation, and testing procedures. They were all rather skeptical after the Stage I about the real possibility to enhance model 80% accuracy. The first run after model update in Stage II showing improved to

90% accuracy on small tested data was accepted with contagious joy and new motivation to continue and apply BPNN prototype on larger data set. I know that during this project all students learned many invaluable skills that they could apply to their future education or work. They all have a better understanding of the advantages of applied neural network models as well as the limitation of such models. Students also used this research opportunity and presented their work during all project stages in multiple forums, including poster and oral presentations at local, state and national conferences. Their poster was accepted for an oral presentation on ACS National Meeting & Exposition, as well as on ASBMB National Meeting.

7. ACKNOWLEDGMENTS

We would like to thank Dr. Swapan Chakrabarti, Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence KS 66045 for his expertise and support.

This work was supported by PORTAL, The Program of Research, Teaching, and Applied Learning at Missouri Western State University, St. Joseph, MO 64507.

8. REFERENCES

[1] Chakrabarti S., Svojanovsky S., Slavik R., Georg G. I., Wilson G. S., and Smith P. G. 2009. Artificial Neural Network Based Analysis of High-Throughput Screening Data for Improved Prediction of Active Compounds. *J. Biomol. Screen* 2009 Dec; 14 (10):1236-44

DOI: 10.1177/1087057109351312

[2] Oyedotun O.K., and Khashman A., 2017 Prototype-Incorporated Emotional Neural Network. *IEEE Trans Neural Netw Learn Syst.* 2017 Aug 15. PMID: 28816677

DOI:10.1109/TNNLS.2017.2730179

[3] Liu, H-X., Chew S. L., Cartegni L., Zhang M. Q., and Krainer A. R. 2000. Exonic Splicing Enhancer Motif Recognized by Human SC35 under Splicing Conditions *Mol. Cel. Biol.* 20 (Feb 2000), 1063-1071. PMID: 10629063

[4] Siala, O., et al., 2014. Slight variations in the SC35 ESE sequence motif among human chromosomes: a computational approach, Gene (2014), http://dx.doi.org/10.1016/j.gene.2014.04.075

[5] Kim, Soyoun, Shi, Hua, Lee, Dong-kee, and Lis, John T. 2003. Specific SR protein-dependent splicing substrates identified through genomic SELEX. *Nucleic Acids Res*. 31, 7 (Feb. 2003), 1955-61

[6] Shepard, Peter J. and Hertel, Klemens J. 2009. The SR Protein Family. Genome Biol. 242, 10 (Oct. 2009), 242.1-242.9.

[7] Gerardo Mendizabal-Ruiz, Israel Román-Godínez, Sulema Torres-Ramos, Ricardo A. Salido-Ruiz, J. Alejandro Morales, 2017, 'On DNA numerical representations for genomic similarity computation', (2017) PLOS ONE, vol. 12, no. 3, p. e0173288 https://doi.org/10.1371/journal.pone.0173288