INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES
BULGARIAN ACADEMY OF SCIENCES
CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **25**, No 4
Sofia • 2025 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2025-0040

Inferring Biomedical Networks Using Multivariate Information Theory: Open-Source Code and Tutorial

Madhumita Das[1], *Bishwajit Das*[2], *Ishaan Majumder*[2], *Durjoy Majumder*[1*]

[1]*Department of Physiology, West Bengal State University, Berunanpukuria, Malikapur, Barasat, North 24 Parganas, Kolkata 700 126, India*
[2]*Society for Systems Biology & Translational Research, 103 Block – C, Bangur Avenue, Kolkata 700055, India*
E-mails: *mdas@ssbtr.net bishwajit@ssbtr.net majumderishaan@gmail.com*
[*]*durjoy@rocketmail.com* (*corresponding author*)

Abstract**:** *In Systems Biology, gene expression data are crucial for designing biological system circuitry. While clustering and soft computing techniques are commonly used for classification, Information Theory-based entropy functions – particularly multivariate entropy – remain underutilized for deriving biological inferences. With the advent of high-throughput data acquisition systems, more quantitative data are now available, increasing the relevance of Information Theory-based applications. Simultaneously, this creates a demand for a user-friendly, automated analytical framework. This work presents an automated computational framework for the systematic exploration of molecular data, designed to facilitate the construction of biological process-based networks. Algorithms based on multivariate Information Theory have been implemented on different platforms: one in a proprietary environment* (*MATLAB*) *and two in open-source environments* (*GNU Octave and Python*)*. All implementations are ready to use, allowing researchers to analyze their data using the platform of their choice. The algorithms have been successfully tested on published gene expression datasets.*

Keywords**:** *Gene network, Algorithm, Information, Mutual information, Multivariate delta****.***

1. Introduction

To understand the differential gene expression patterns in various cells and tissues, Functional Genomics is recommended. In that area, gene regulatory mechanisms occupy the central focus. Within a cell, many genes cooperate and interact with each other, thereby regulating the expression of others. Generally, this is represented through a gene interaction network where each gene is positioned at different nodes connected by edges, and functional relationships among them are represented through ON or OFF switches at each node. Traditionally, these are formulated with the data derived from in vitro experiments (where a particular gene is perturbed either by over-expression or knock-out of a gene, followed by the measurement of the

downstream targeted genes). The Boolean method is simple; with discrete data, it establishes relationships on static graphs. It requires noise-free data, as random graphs do not capture the high clustering coefficient property. But in clinical cases, expression data may readily be collected with different instruments [1-3].

Under disease conditions, gene interaction networks malfunction; as a result, the magnitude of gene expression at different nodal points is significantly altered. Therefore, microarray-based technologies are suggested for measuring large sets of genes. Although this technology enables the acquisition of a large amount of data in one shot, the captured data are not truly quantitative in nature. With the increasing emphasis on the analysis of human disease data, several analytical methods have been developed (Table 1).

For the analysis of gene expression, most analytical methods use data obtained from microarray-based technology, which provides a large amount of data. Having such a large amount of data, most of the available analytical methods have relied on heuristic-based methods. However, quantitative and mechanistic understanding of biological phenomena is seldom achieved through heuristic-based approaches. A major objective of the available approaches addresses the issue of the identification of disease subtypes through clustering or grouping of genes within the vast global gene network. However, with such approaches, many functional issues are kept aside, like gene–gene cooperation, feedback, and feed-forward. In such cases, a precise gene-by-gene interaction study becomes important. This would not only aid in understanding the patho-physiological state transition but also contribute to the design of a precise therapeutic protocol.

Information Theory (IT-based) analysis has been utilised for analysing a small number of genes (variables) with a precise depiction of gene-by-gene interactions. The work identifies the causation of the Human Leukocytic Antigens (HLA) surface expression down-regulation in different leukemia sub-types [19-20]. Transcription Factors (TFs) regulate two types of HLA genes – HLA class I (HLA-ABC) and II. SE is regarded as a sink in this situation, while TFs behave like sources. Thus, a numerical metric is used to represent the gene network using human leukemia data. The work identifies the differential role of different TFs in controlling the inducible expression of the immune gene HLA. Using 1st order analysis, the work identifies a TF, RFXB, in both myeloid and lymphoid origins; however, for lymphoid leukemia, CREB1, another inducible TF, may play a significant role in controlling HLA expression. Precise analysis with MaxEnt, a multivariate dependence Information Theory (IT), confirms the same finding. This analysis also reveals that another combination of two TFs (cooperative behaviour), namely CIITA and IRF1, has significance in myeloid leukemic cells only; however, no alteration is observed for other combinations of TFs in either HLA class II gene regulation or leukemia of lymphoid origins [21]. Biochemical descriptions of signalling require quantitative support to explain how complex stimuli (inputs) are encoded in distinct activities of pathways to the effectors (outputs). This can be explained by IT [22], which is suggested as a powerful inference method for different facets of biomedical areas [23].

Table 1. Progress of gene expression analytical methods

Analytical approach	Bio-medical implication	Reference
Neighbourhood analysis (Classification algorithm)	AML and ALL – both acute types of leukemia are classified through some subset of genes that have diagnostic implications. However, the performance is not consistent in multiple tests	[4, 5]
Linear Artificial Neural Network (ANN)	Based on gene expression signatures, this method is applied in the diagnosis of cancer through classification, though with a limited amount (63 tumor cell types) of training data	[6]
	ANN-based classifier with sample filtering is utilised for classifying Leukemia gene expression datasets	[7]
Support Vector Machine (SVM) (Supervised Learning Algorithm)	Work has identified and analysed a number of extracted features which are used to train the SVM and classify the unknown sample using an ovarian gene dataset. This work is a modification of the G o l u b et al. [4] (1999) method to identify feature selection from disease cells, but the accuracy rate is still not very high	[5]
	The SVM-based approach is used to classify microarray gene expression data. Colon and lymphoma cancer data are used to identify Mutual Information (MI) between the genes, and are evaluated using the Leave-One-Out Cross-Validation (LOOCV) method	[8]
Naïve Bayes' classification	With microarray data, a sequential feature extraction approach is used, utilizing naive Bayes' classification. The accuracy rate of the method is not very high	[9, 10]
Multi-objective Genetic Algorithm (MOGA) based fuzzy clustering	The clustering algorithm MOGA-SVM has been used to identify groups of co-expressed genes from available gene expression datasets (Yeast Sporulation, Yeast cell cycle, Arabidopsis Thaliana, Human Fibroblasts Serum, and Rat CNS data). The work has not included any human disease data	[11]
Multi-class clustering using GA, k-means clustering, and the SVM technique	A combined approach is executed for cancer dataset classification (generate class level) in small Round Blood Cell Tumors, adult malignancy, and Brain tumors. Thus, some (gene-based) markers had been identified using the multi-class clustering method. The performance of this approach is better than the t-statistic. The work makes a classification between unrelated biological samples	[12]
Fuzzy logic	Finds information from the Lymphoma gene expression data, and subtypes are made using fuzzy rules on a test dataset. This method achieves ~77% accuracy	[13]
Combined classification schemes, namely decision trees, random forest, nearest neighbor, and multilayer perceptron	DNA microarray experiments generate a very large number of features (in the order of thousands) for a limited number of patients (in the order of hundreds or fewer), so the classification task becomes very complex using the feature-selection process. Complexity is reduced with a combined method (feature selection and classification), and the method is proposed for the identification of different DNA microarray datasets (namely Breast, Colon, Leukemia, Lymphoma, Lung, and Ovarian). The work employs classification with unrelated datasets	[14]
Multivariate feature ranking method	A multivariate feature ranking method has been applied to improve the quality of gene selection with an improvement of the accuracy of microarray data of Leukemia, Prostate, SRBCT, Lung, Ovarian, and MLL classification. The work makes a classification between unrelated datasets	[15]
GeneCloudOmics, a Web server	A web server is developed for both microarray and RNA-sequence data analysis. 23 different analytical tools are used, such as PCA, clustering (hierarchical, K-means, T-SNE, SOM) for gene expression analysis, along with the identification of protein-protein interactions. But presently the web server is not in operation	[16]
Naïve Bayes classification using forward selection	Naïve Bayes classification has been used to collect information from big data and then extract features using the forward selection method. This method achieved better accuracy using a heart disease dataset, predicting heart disease with an accuracy of 84.15%, compared to the Naive Bayes model, which achieved 82.84%	[17]
Artificial Bee Colony (ABC) and Deep Learning Method	A hybrid approach (an optimization algorithm (ABC) for feature selection and a deep learning (conventional Neural Network) method for classification) was used for the classification of DNA microarray gene expression Leukemia data. This approach achieved 98% accuracy in prediction	[18]

Recently, a work showed a correlation between the Shannon entropy function (calculated from tumor transcriptomic data) and tumor aggressiveness [24]. Different combinations of cellular cross-talks between the immune system can be derived quantitatively through IT to decipher the changes in cancer, thus its importance in immuno-oncology is suggested [25].

Table 2. IT-based different analytical software/packages

Software/ Package	Platform	Application	Comment	Reference
JAK/STAT	R	IL-6-mediated stimulation to downstream phosphorylation of the JAK/STAT pathway is determined as Mutual Information (MI) and channel capacity in a different heterogeneous cell population	Applied with cell line-based data	[26]
MERIDIAN	MATLAB	Predictive models using the maximum entropy value to understand the heterogeneity of the cell population. The joint probability distribution is applied to measure cell-to-cell variability in terms of gene expression (biochemical abundances)	Network model constructed with single-cell data	[27]
NOGEA	R	Joint Entropy (JE) values between gene sets are calculated, revealing that master genes with high entropy act as initiation points of the disease state, whereas redundant genes exhibit low entropy values within the disease-specific gene network	JE applied to the co-occurrence between drug and gene expression, thus, drug-induced comorbidity is revealed	[28]

Several IT-based analytical packages are available in platforms such as MATLAB and R, each offering distinct functionalities, as summarized in Table 2. These tools are primarily employed to infer interdependencies within a set of variables (e.g., genes), typically using measures such as JE or MI. Recent studies have applied IT approaches in the R environment to analyze constitutive gene expression data from the Neurospora experimental model, demonstrating that elevated entropy values are frequently associated with extreme physiological conditions [29]. All of these methods are developed using data from experimental models in a steady-state condition and do not support the analysis of differential effects arising from individual or combinatorial TFs. Furthermore, these implementations lack full automation and require substantial coding expertise.

An IT-based analytical framework has been developed on the MATLAB platform to analyze human leukemia-derived data and elucidate HLA gene regulatory mechanisms mediated by diverse combinations of transcription factors [30]. While the analytical workflow is fully automated, the absence of an integrated Graphical User Interface (GUI) in the original implementation necessitates a degree of programming proficiency for effective utilization. Furthermore, the proprietary licensing constraints associated with MATLAB limit accessibility for a substantial segment of the research community, particularly experimental biologists and biomedical specialists. To facilitate broader and more diverse applicability across disciplinary boundaries, we have implemented equivalent, ready-to-use automated workflows in both GNU Octave and Python, two widely adopted open-source environments. These implementations incorporate an integrated GUI designed to enhance user accessibility, thereby enabling researchers without programming expertise to conduct analyses efficiently.

2. Methodology

2.1. Information theoretic background

Entropy (from the Greek word meaning "change") is a concept that emerged in the 19-th century in thermodynamics by Clausius [31] and further developed by Boltzmann and Gibbs. It has been applied in biology [23, 32], economics [33], engineering [34], linguistics [35], and many other disciplines. Subsequently, in the 20th century, this essential innovation was incorporated in Communication Engineering by Claude Shannon, and this is now known as Information Theory (IT) [36]. For a discrete event x with probability P, information can be defined as

$$\text{Information}\,(x) = \log P(x).$$

Here, IT is applied to understand the propensity of different Transcription Factors (TFs) to regulate the Surface Expression (SE) of HLA [19, 38]. In statistical measurements, entropy concepts provide reliable tools for assessing the status and performance of the information systems [37]. It is interesting to note that many of the TFs that regulate SE of HLA are inducible in nature (i.e., expressed under infection or stimulation) and therefore, have no basal level of expression. Here HLA gene SE is considered a sink, and TFs behave like a source because the HLA gene is regulated by the TFs. So, for an understanding of this, IT-based analysis can be suitable [19]. Here analytical framework has been designed to understand the different combinations of TFs with SE of HLA; hence, these are considered as attributes, and the IT background can be explained as follows:

Shannon Entropy (H) is the property of the random distribution of a variable, and it denotes the measurement of the degree of randomness. Given the n data points about a variable, the range is subdivided into q intervals, and if f_i is the number of data points occurring in the *ith interval*, the $p_i=f_i/n$ defines a probability distribution for the variable over the chosen intervals. For any single attribute, like TF1, Entropy (H) is expressed as:

$$H(\text{TF1}) = \sum_{i=1}^{q} p_i \times \log_r \left(\frac{1}{p_i}\right), \tag{1}$$

Or,

$$H(\text{TF1}) = -\sum_{i=1}^{q} p_i \times \log_r p_i. \tag{2}$$

All logarithmic values were calculated using the base-10 logarithm ($\log_{10}$), with the logarithmic base defined as r=10. The following properties of the entropy function, as listed below, are useful in the present context.

1. H is symmetric and continuous. Thus, with the vast amount of data, changes in sub-interval (bin-size) produce the same uncertainty measure.

2. $H_{n+1}(p_1, p_2, p_3, \ldots, p_{n-1}, 0) = H_n(p_1, p_2, p_3, \ldots., p_n)$, i.e, if an interval is empty, it does not affect entropy. This indicates changes in the global range (i.e., max, min) do not affect the sample data point-based calculation. Thus, all the different groups (for example, normal, disease, etc.) can be governed with the same bin size without affecting the desired metric.

3. $H_{n+1}(p_1, p_2, p_3, \ldots, p_{n-1}, 0) \leq H_n(1/n, 1/n, 1/n, \ldots.,1/n)$. This means that if the data is uniformly distributed, the entropy is maximum, and it falls when the data is more clustered.

Joint Entropy (JE) is the amount of average information provided by the joint probability of two or more discrete random attributes (s) between TF and SE. If TF1 (like RFXB) and SE1 (like HLA-ABC) are two random variables, entropy (H) can be obtained by

(3) $$H = \sum_{\mathrm{RFXB}} \sum_{\mathrm{ABC}} p(\mathrm{RFXB}, \mathrm{ABC}) \times \log \left(\frac{1}{p(\mathrm{RFXB},\ \mathrm{ABC})}\right),$$

Or,

$$H(\mathrm{RFXB}, \mathrm{ABC}) = \sum_{\mathrm{RFXB}} \sum_{\mathrm{ABC}} p(\mathrm{RFXB}, \mathrm{ABC}) \times \log \left(\frac{1}{p(\mathrm{RFXB},\ \mathrm{ABC})}\right).$$

In IT, the Gibbs Inequality (J. Gibbs, 1839-1903) can be applied in the information entropy of a discrete probability distribution (p_i, p'_i) (G i b b s [39]). This theory states that the entropy is smaller than or equal to any other average formed using the same probabilities (MaxEnt) but a different function in the logarithm. Specifically,

(4) $$\sum_i \ p_i \ \log_2 \frac{1}{p_i} \leq \sum_i p_i \log_2 \left(\frac{1}{p'_i}\right).$$

This equation can be proven that the natural logarithm has the property that it is less than or equal to a straight line that is tangent to it at any point.

Conditional Entropy quantifies the amount of information needed to describe the outcome of an attribute (RFXB) when the value of another attribute (HLA-ABC) is known,

(5) $$H(\mathrm{RFXB}|\mathrm{ABC}) = -\sum_{\mathrm{RFXB}} p_{(\mathrm{RFXB},\ \mathrm{ABC})} \log\ p\left(\frac{\mathrm{RFXB}}{\mathrm{ABC}}\right).$$

The conditional entropy is equivalently given by the expression

(6) $$H(\mathrm{RFXB}|\mathrm{ABC}) = H(\mathrm{RFXB}, \mathrm{ABC}) - H(\mathrm{RFXB}).$$

Mutual Information (MI) of two random variables is a measure of the mutual dependence between the two variables. It quantifies the "amount of information" (its unit is in bits) obtained about one random variable, with respect to another random variable. Thus gene network is denoted through a numeric metric. It is a measure of the amount of information that one attribute (RFXB) contains about another attribute (HLA-ABC),

(7) $$I(\mathrm{RFXB}; \mathrm{ABC}) = \sum_{\mathrm{RFXB,ABC}} p_{(\mathrm{RFXB,ABC})} \log \left(\frac{p(\mathrm{RFXB,ABC}}{p(\mathrm{RFXB})p(\mathrm{ABC})}\right).$$

It is equivalently given by the expression

(8) $$I(\mathrm{RFXB}; \mathrm{ABC}) = H(\mathrm{RFXB}) - H(\mathrm{RFXB}|\mathrm{ABC}).$$

Multivariate information theory is used to calculate MaxEnt (Δ) with the simultaneous effect of two attributes on one attribute,

(9) $$\Delta = I[(\mathrm{CIITA}; \mathrm{ABC}|\mathrm{RFXB}) - I(\mathrm{CIITA}; \mathrm{ABC}) + I(\mathrm{CIITA}; \mathrm{RFXB})].$$

2.2. Sample data

We used published data to run the codes. The data was obtained from D a s and M a j u m d e r [40]. The dataset comprises gene expression of different Transcription Factors (TFs) that regulate the Surface Expression (SE) of HLA class I and II under leukemic conditions. The results were compared with data reported by D a s and M a j u m d e r [21, 40], and M a j u m d e r [19].

2.3. Installation of open-source platform

MATLAB is a proprietary coding platform available at Mathworks, which can be installed with default settings as per the instructions (**https://in.mathworks.com/help/install/install-products.html**). GNU Octave and Python are both open-source coding platforms, installed on 64-bit Windows OS computers. Step-by-step installation procedures are available from URLs: displayed as enlarged images, **https://www.ssbtr.net/org_research_details.php?item=PRJ-07**. The details are as follows.

Python, PyCharm, and installation of Python packages. Python Version 3.6.3 can be installed from https://www.python.org/downloads/, followed by the installation of the Community edition of PyCharm IDE from **https://www.jetbrains.com/pycharm/download/**. In PyCharm new project is started, and then install the following packages: tkinter, pandas, numpy, math, time, and sys from File → Settings → Project → Python Interpreter → Packages → +. The package tkinter is used for enabling graphical user interfaces (GUIs), while time and sys are used for getting an idea of the system's performance time. All the packages are imported to the editor window by the import command.

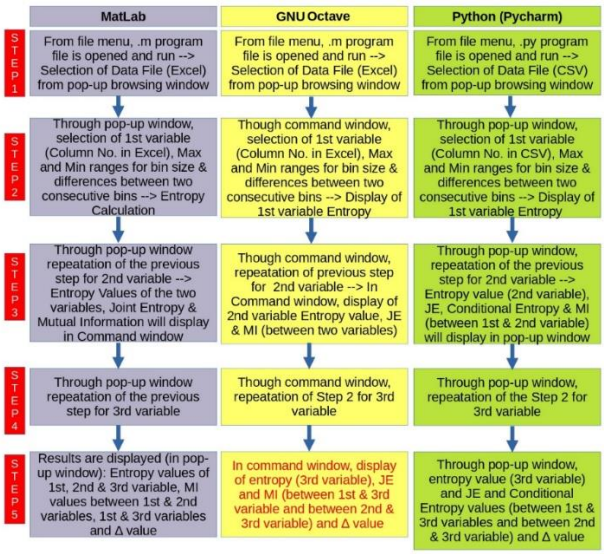

Fig. 1. Flowchart showing the procedural steps for running codes in different platforms (MATLAB, GNU Octave, and Python)

GNU Octave installation, including packages. GNU Octave can be installed from the official webpage of GNU Octave (**https://www.gnu.org/software/octave/index**). The package io is needed to read the workspace in XLSX format. On Windows, Octave version 6.2.0 has all the packages and can be installed with default settings and checked by the "pkg list" command. However, if needed, the io package can be installed by using the "pkg install-forge" and "pkg load" commands sequentially from the command window. Installation and loading of packages are ensured with the "pkg list" command (it displays the list of all the installed packages with * marked).

2.4. Code development

After installation, codes are developed. Programme codes are developed in the Editor of MATLAB and/or GNU Octave, and program files are saved as .m files. Python programme codes are developed in the Python editor and saved as a .py file (Sample Data File and Programme codes for different platforms – are all available from the URL: https://www.ssbtr.net/org_research_details.php?item=PRJ-07 as a zip file.

2.5. Analysis algorithm

Using the same dataset, the developed code is run on three different platforms, i.e., MATLAB, GNU Octave, and Python. A source file for the dataset is stored in .xlsx, and a copy of the file is also saved in .csv format (for Python). A step-by-step detailed procedure is available at the following **https://www.ssbtr.net/org_research_details.php?item=PRJ-07** as a zip file (figure files with sequential numbers), and through video (post date 2025-June-05) from **https://ssbtrthinktank.blogspot.com/2025/06/steps-towards-information-theoretic.html**. The sequential procedural steps for running the code on each platform are presented in a flowchart (Fig. 1).

3. Results

All gene expression data have been collected from the work of Majumder [38]. Two types of attributes have been considered – TF and SE. TF is the expression data of different transcription factors (TF1, TF2, TF3…TFn) that govern the surface expression of genes [(HLA-ABC (SE1) and HLA-DR (SE2)]. To evaluate our developed algorithm, we used values for the variables (RFX5, ABC, CIITA, HLA-DR) from NV and AML cases, and compared the resulting values for Entropy, Joint Entropy (JE), Mutual Information (MI), and Multivariate Delta (Δ), using the raw data available in the work of Majumder [38]. The output values are matched with the data published in the works of Das and Majumder [20, 40]. For finding the Entropy, JE, MI, and multivariate information, we made simulation runs with the following two combinations: SE1 (HLA-ABC) with TF1 (CIITA) and TF2 (RFX5); and SE2(HLA-DR) with TF1(CIITA) and TF2(RFX5) in NV and AML. The output of the analysis is summarized as follows:

3.1. Entropy

The computed entropy values using MATLAB and GNU Octave are in close agreement with the results reported by Das and Majumder [40], demonstrating the reproducibility of the method. However, the result produced in Python differs slightly because of its higher numerical precision compared to the other platforms (Tables 3 and 4). This difference arises because Python (with libraries like NumPy) uses double-precision floating-point numbers (up to ~16 decimal digits), while MATLAB and Octave often display results with only 6 digits of precision by default (**https://www.geeksforgeeks.org/python-float-type-and-its-methods/**). The obtained entropy values for each of the attributes in NV and in different leukemia combinations (using 7 and 10 intervals) are available at the following URL:

https://www.ssbtr.net/org_research_details.php?item=PRJ-07, in the file named **Fig8** (located within the **Figures** folder).

3.2. Joint entropy and mutual information

Table 3. Entropy, Joint Entropy, Mutual Information, and Multivariate Information of each attribute of NV data. In the last three columns, % values in parentheses show differences in results obtained with the published results in reference literature [20, 40]

Case	Variable	Bin Size	Result	Referent value	MATLAB	GNU Octave	Python
Entropy (Ent)							
NV	RFX5	7 BIN	Ent	0.4728	0.5074 (7.31%)	0.5073 (7.31%)	0.4604 (2.62%)
NV	CIITA	7 BIN	Ent	0.4579	0.4472 (2.33%)	0.44717 (2.34%)	0.2403 (47.52%)
NV	HLA-ABC	7 BIN	Ent	0.5159	0.5276 (2.26%)	0.5277 (2.28%)	0.37808 (26.71%)
NV	HLA-DR	7 BIN	Ent	0.4727	0.4472 (5.39%)	0.4471 (5.41%)	0.35987 (23.86%)
NV	RFX5	10 BIN	Ent	0.4470	0.5074 (13.51%)	0.5073 (13.48%)	0.3622 (18.97%)
NV	CIITA	10 BIN	Ent	0.7536	0.616 (18.25%)	0.617 (18.12%)	0.47813 (36.55%)
NV	HLA-ABC	10 BIN	Ent	0.5988	0.6198 (3.50%)	0.6197 (3.49%)	0.48899 (18.33%)
NV	HLA-DR	10 BIN	Ent	0.5556	0.5871 (5.66%)	0.5592 (6.64%)	0.34414 (38.05%)
Joint Entropy (JE)							
NV	HLA-ABC & CIITA	7 Vs 7	JE	0.6614*	0.886 (33.26%)	0.7987 (20.75%)	0.6931 (4.79%)
NV	HLA-ABC & CIITA	7 Vs 10	JE	0.7535*	0.886 (17.58%)	0.6366 (15.51%)	0.6167 (18.25.%)
NV	HLA-ABC & RFX5	7 Vs 7	JE	NA	1.6094	1.1020	0.6829
NV	HLA-ABC & RFX5	7 Vs 10	JE	NA	1.8344	1.18085	1.0789
NV	HLA-ABC & CIITA	7 Vs 7	JE	NA	1.5607	1.1214	1.0549
NV	HLA-ABC & CIITA	7 Vs 10	JE	NA	1.7918	1.2521	1.0549
NV	HLA-DR & CIITA	7 Vs 7	JE	0.6387*	0.6913 (8.24%)	0.7463 (16.84%)	0.809 (36.2%)
NV	HLA-DR & CIITA	7 Vs 10	JE	0.5558*	0.5189 (6.63%)	0.5188 (6.65%)	0.6569 (18.18%)
NV	HLA-DR & RFX5	7 Vs 7	JE	NA	1.7329	1.3198	1.0821
NV	HLA-DR & RFX5	7 Vs 10	JE	NA	1.3863	0.8077	1.3107
NV	HLA-DR & CIITA	7 Vs 7	JE	NA	1.0986	1.1512	1.1140
NV	HLA-DR & CIITA	7 Vs 10	JE	NA	1.9459	1.2728	1.2424
Mutual Information (MI)							
NV	HLA-ABC & CIITA	7 VS 7	MI	0.3124*	0.4013 (28.47%)	0.4013 (28.46%)	0.4123 (32.00%)
NV	HLA-ABC & RFX5	7 VS 7	MI	0.5158*	0.6197 (20.15%)	0.6197 (20.15%)	0.4132 (19.87%)
NV	HLA-DR &CIITA	7 VS 7	MI	0.2919*	0.1355 (53.57%)	0.2835 (62.85%)	0.4102 (40.54%)
NV	HLA-DR & RFX5	7 VS 7	MI	0.2693*	0.6866 (154%)	0.6866 (154.96%)	0.4056 (50.62%)
NV	HLA-ABC & CIITA	7 Vs 10	MI	0.3009*	0.2024 (32.73%)	0.4303 (43.010%)	0.8201 (172.57%)
NV	HLA-ABC & RFX5	7 Vs 10	MI	0.3175*	0.6197 (95%)	0.7197 (126.70%)	0.8260 (160.17%)
NV	HLA-DR & CIITA	7 Vs 10	MI	0.4554*	0.5861 (28.71%)	0.6773 (48.73%)	0.7773 70.69%
NV	HLA-DR & RFX5	7 Vs 10	MI	0.3264*	0.2255 (30.9%)	0.33562 (2.82%)	0.43749 (34.03%)
Multivariate Delta (Δ)							
NV	RFX5, HLA-ABC & CIITA	all in 7	Δ	NA	1.0624	1.2394	1.5727
NV	CIITA, HLA-ABC & RFX5	all in 7	Δ	NA	0.89193	0.89145	0.98214
NV	RFX5, HLA-ABC & CIITA	all in 10	Δ	NA	0.69261	0.69256	0.945474
NV	CIITA, HLA-ABC & RFX5	all in 10	Δ	0.2010*	0.3114 (54.92%)	0.2923 (45.42%)	0.28482 (41.69%)
NV	RFX5, HLA-DR & CIITA	all in 7	Δ	0.2092*	0.30979 (48.07%)	0.2878 (37.57%)	0.2314 (10.611%)
NV	CIITA, HLA-DR & RFX5	all in 7	Δ	NA	1.3015	0.9263	0.7003
NV	RFX5, HLA-DR & CIITA	all in 10	Δ	NA	0.45626	0.68723	1.3015
NV	CIITA, HLA-DR & RFX5	all in 10	Δ	NA	0.82863	0.72829	0.670225

*Note: Bin size or Max-Min values are not mentioned in [20, 40].

The results for JE and MI obtained using MATLAB, GNU Octave, and Python are presented in the tables below (Tables 3 and 4). These values are not compared with the mentioned literature [20, 40]. This is due not only to the absence of bin size information (or min–max values), but also to the lack of a clear indication of the attribute combinations used in the calculation of JE and MI values.

Table 4. Entropy, Joint Entropy, Mutual Information, and Multivariate Information of each attribute of AML data.

Case	Variable	Bin Size	Result	Referent value	MATLAB	GNU Octave	Python
Entropy (Ent)							
AML	RFX5	7 BIN	Ent	0.7343	0.7502 (2.16%)	0.7502 (2.16%)	0.3902 (46.85%)
AML	CIITA	7 BIN	Ent	0.5853	0.5692 (2.78%)	0.5856 (0.06%)	0.3257 (44.34%)
AML	HLA-ABC	7 BIN	Ent	0.5914	0.5917 (0.05%)	0.5917 (0.05%)	0.3312 (43.94%)
AML	HLA-DR	7 BIN	Ent	0.6866	0.6871 (0.07%)	0.6871 (0.07%)	0.5713 (16.79%)
AML	RFX5	10 BIN	Ent	0.8012	0.8017 (0.06%)	0.8017 (0.06%)	0.5011 (37.45%)
AML	CIITA	10 BIN	Ent	0.7153	0.71 (0.74%)	0.7099 (0.74%)	0.5035 (29.60%)
AML	HLA-ABC	10 BIN	Ent	0.7116	0.7273 (2.20%)	0.7272 (2.20%)	0.3452 (51.48%)
AML	HLA-DR	10 BIN	Ent	0.7839	0.7846 (0.89%)	0.7845 (0.08%)	0.5475 (30.14%)
Joint Entropy (JE)							
AML	HLA-ABC & CIITA	7 Vs 7	JE	0.8398*	0.6774 (19.38%)	0.6775 (19.33%)	1.27703 (52.06%)
AML	HLA-ABC & CIITA	7 Vs 10	JE	0.9049*	0.85912 (5.50%)	0.85943 (5.02%)	1.0042 (10.97%)
AML	HLA-ABC & RFX5	7 Vs 7	JE	NA	0.6772	0.84936	0.8821
AML	HLA-ABC & RFX5	7 Vs 10	JE	NA	0.7110	0.9864	1.010
AML	HLA-ABC & CIITA	7 Vs 7	JE	NA	0.4956	0.6929	0.9559
AML	HLA-ABC & CIITA	7 Vs 10	JE	NA	0.6785	0.8599	1.0042
AML	HLA-DR & CIITA	7 Vs 7	JE	0.7817*	0.6468 (17.25%)	0.6465 (17.28%)	1.2770 (63.36%)
AML	HLA-DR Vs CIITA	7 Vs 10	JE	0.7522*	0.63481 (15.60%)	0.6345 (15.64%)	1.0042 (33.50%)
AML	HLA-DR & RFX5	7 Vs 7	JE	NA	0.7194	0.9730	1.2770
AML	HLA-DR & RFX5	7 Vs 10	JE	NA	0.7957	0.8145	1.2703
AML	HLA-DR & CIITA	7 Vs 7	JE	NA	0.9147	1.1281	1.0114
AML	HLA-DR & CIITA	7 Vs 10	JE	NA	0.9258	1.4081	1.0114
Mutual Information (MI)							
AML	HLA-ABC & CIITA	7 VS 7	MI	0.3369*	0.38886 (15.42%)	0.3885 (15.33%)	0.6254 (85.65%)
AML	HLA-ABC & RFX5	7 VS 7	MI	1.2068*	0.99018 (17.9%)	0.9901 (17.95%)	0.4750 (60.63%)
AML	HLA-DR & CIITA	7 VS 7	MI	0.4902*	0.68952 (40.6%)	0.6895 (40.66%)	0.6254 (27.58%)
AML	HLA-DR &RFX5	7 VS 7	MI	0.5008*	0.77923 (55.5%)	0.7792 (55.59%)	0.4993 (0.28%)
AML	HLA-ABC & CIITA	7 Vs 10	MI	0.522*	0.76312 (46.19%)	0.7634 (46.25%)	4.9161 (841.78%)
AML	HLA-ABC & RFX5	7 Vs 10	MI	0.3368*	1.1219 (233%)	0.9835 (192.03%)	0.1275 (62.14%)
AML	HLA-DR & CIITA	7 Vs 10	MI	0.4554*	0.5957 (30.82%)	0.5953 (30.73%)	0.4191 (7.95%)
AML	HLA-DR & RFX5	7 Vs 10	MI	0.3264*	1.15229 (253%)	1.1423 (249.99%)	0.2579 (20.98%)
Multivariate Delta (Δ)							
AML	**RFX5 HLA-ABC CIITA**	**all in 7**	**Δ**	**0.4653***	**0.4996 (7.38%)**	**0.4240 (8.87%)**	**0.4405 (5.32%)**
AML	CIITA HLA-ABC RFX5	all in 7	Δ	NA	0.7097	0.7098	0.3479
AML	RFX5 HLA-ABC CIITA	all in 10	Δ	NA	0.5761	0.7653	1.2925
AML	CIITA HLA-ABC RFX5	all in 10	Δ	NA	1.3608	1.4608	1.1854
AML	RFX5 HLA-DR CIITA	all in 7	Δ	0.405*	1.0941 (135.15%)	0.92324 (98.41%)	0.6568 (41.15%)
AML	CIITA HLA-DR RFX5	all in 7	Δ	NA	0.5980	0.5590	0.4023
AML	RFX5 HLA-DR CIITA	all in 10	Δ	NA	0.8875	0.9334	1.2163
AML	CIITA HLA-DR RFX5	all in 10	Δ	NA	1.1564	1.06241	0.3729

*Note: Bin size or Max-Min values are not mentioned in references [20, 40].

3.3. Multivariate information

Multivariate information (Δ) results from MATLAB, GNU Octave, and Python show strong agreement across the three different platforms but deviate from the values reported in literature. However, in AML cases (with a bin size 7) the results are quite similar – within 9% deviation (MATLAB 8%, GNU Octave 9%, and Python 6%) – compared to the values reported by Das and Majumder [20, 40] for the combinations of RFX5, HLA-ABC and CIITA (Table 3 and 4). These findings indicate that Python offers greater precision in numerical output, relative to MATLAB and GNU Octave. The discrepancy in values (in the NV case) is due to the much smaller sample size. The overall differences in results between existing literature and our algorithm are possibly due to the reason that the previous work was done through manual computation or with the consideration of a lesser number of floating points.

4. Discussion

From a systems biology perspective, normal and disease conditions are considered distinct states; therefore, a molecular understanding of the state transition from physiology to pathology is crucial. In this context, gene expression data provide characteristic signatures that help identify influential interactions among genes within a network. This approach is commonly used for reverse engineering [41]. The differential roles of individual molecular attributes exert a significant influence. This is particularly important for understanding immune gene regulation, as many genes become operative only under specific physiological or pathological states [42]. It is important to note that, in designing therapies under the purview of Systems Medicine, the role of each molecular attribute becomes critical for identifying control variables. In real-life clinical cases, considerable variation exists in the levels of molecular attributes. Therefore, emphasis is placed on capturing time-dependent data for each attribute [43, 44]; however, due to substantial financial constraints, this is not commonly practiced. Consequently, determining probability density functions and assigning entropy measures can be useful for concluding cross-sectional, heterogeneous data. In this context, IT-based analysis is advantageous, as it can be applied effectively to both undersampled datasets and high-throughput big data [23, 45].

Recently, several studies have employed IT-based analysis to investigate downstream effector activities in biochemical pathways, tumor aggressiveness, and alterations in immune function under malignant conditions [22, 24, 25]. These studies utilized entropy functions and/or MI. Furthermore, multivariate information-theoretic analysis has been proposed and successfully applied to differentiate signaling cascades that influence HLA gene regulation in human leukemia [20, 21]. However, to date, there remains a lack of automated analytical methods for multivariate information-theoretic analysis available on open-source platforms.

Though IT-based analysis offers several advantages, it has not been extensively applied to the analysis of clinical or human disease data. This limited adoption may be due to the lack of accessible platforms and/or automated algorithms. Existing

computational tools for gene expression data analysis generally focus on disease classification rather than elucidating the roles of individual molecular attributes. While IT-based analysis algorithms and packages are available in the R and MATLAB environments [26-29], they typically require coding expertise, which may limit their widespread use. Moreover, such algorithms are mostly concerned with identifying associations among attributes within a signaling pathway; hence, available programs typically employ channel capacity analysis using entropy functions or JE analysis. These IT-based computer algorithms are typically developed using data derived from cell line-based (in vitro) models, where the biological system is assumed to have reached a stable equilibrium state. As a result, only simple associations among attributes can be identified. However, combinations of different attributes affecting effector functions cannot be effectively studied using these programs. Consequently, the multivariable complexity of biological functionality remains largely unaddressed. Furthermore, analysis using these algorithms is not automated and therefore requires some level of coding expertise.

An algorithm implemented in MATLAB enables multivariate IT analysis to interrogate complex regulatory interactions [30]; however, its adoption within the experimental biological community is constrained by the proprietary restrictions of the MATLAB platform. Furthermore, the absence of an integrated GUI limits accessibility to users lacking programming expertise, thereby impeding broader application by experimental biologists and clinical researchers. To overcome these barriers, there exists an imperative need to develop automated, open-source analytical tools that facilitate reproducible and scalable data analysis workflows [19-21, 40, 46]. Here, the developed computational framework has been implemented in two open-source environments – GNU Octave and Python – to ensure broader accessibility and reproducibility. Integration of a GUI, together with automated data processing and analysis modules, enables researchers to execute analytical workflows without the need for programming proficiency. The framework has been successfully validated on a representative small-scale dataset comprising a substantial number of variables; however, it is inherently scalable and capable of efficiently accommodating larger datasets. Moreover, users can flexibly evaluate any combination of variables of their choice to explore complex regulatory interactions. Collectively, this platform provides a robust, extensible, and user-oriented solution for experimental biologists and biomedical researchers, enabling them to assess the functional significance of their data across diverse experimental conditions.

Both MATLAB and GNU Octave are widely used by the scientific community, including computational biologists [47, 48], while Python has now become increasingly popular among data scientists for complex data analysis in the 4IR/5IR era [49, 50]. The latter two are available as open-source. All of these have a wide range of function libraries that ease coding. A comparison between MATLAB, GNU Octave, and Python is available from url: **https://www.ssbtr.net/org_research_details.php?item=PRJ-07** as Table 5. Using the same dataset, our analysis shows that Python provides more accurate results than GNU Octave and MATLAB because Python uses 16-digit decimal precision, whereas MATLAB and GNU Octave use only 6-digit floating-point precision.

The developed code on different platforms has a runtime of less than a minute (ranging from 20 s up to 50 s based on the user's input performance for entry of variables as input); however, the memory usage of the code is 587 MB in MATLAB, 5.128 MB in Python, and 0.25 MB in GNU Octave. Both Octave and MATLAB are designed for numerical operations and rely on matrix-based computation. As a result, their memory management strategies are optimized for efficiently handling large arrays and matrices. In contrast, Python uses a private heap to manage memory, dynamically allocating space based on an object's type and size. Its garbage collection system automatically reclaims memory used by objects that are no longer in use. In GNU Octave, all inputs (except data file read) are given through the command window, and all outputs are displayed in the command window; therefore, no external package is used, and it consumes much less memory compared to other coding platforms. The very low execution time and acceptable memory usage across different open-source coding platforms demonstrate the practical relevance and usability of the developed code.

5. Conclusion

We have developed Information Theory-based algorithms across multiple open-source platforms to facilitate the application of multivariate IT analytical techniques for gene expression and related data analysis. The algorithm incorporates various molecular attributes and supports the analysis of both large-scale and undersampled datasets. Validation using human disease datasets demonstrated its capability to identify combinations of attributes whose relative weighted contributions may signify critical pathways or gene regulatory cascades. This enables network reconstruction to distinguish between disease and normal states – a classification process [51] – and may assist in the design of drug targets or transcription-based therapies. Given the growing emphasis on the availability of computational codes for advancing science and medicine [52, 53], we have made our code publicly accessible, with two implementations available on open-source platforms, allowing users to modify them according to their specific requirements. For example, in the case of time-varying data, different time points can be assigned to separate bins. Similarly, spatial (positional) data variations can also be binned for analysis. Thus, our developed algorithms can be applied to analyze both temporal and spatial data variations, assist in the reconstruction of networks based on variable dependencies, and be extended to various fields such as biology, public health, sociology, and economics.

***Acknowledgement*:** The authors acknowledge the critical comments of the expert members of SSBTR for the necessary improvement of this work. IM is a student of St. Xavier's University and is working as an Intern trainee in SSBTR. The authors acknowledge Ms. Sumita Biswas (Enfield, UK) for English editing and Dr. Abhik Mukherjee (IIEST, Shibpur) for scientific editing. We also acknowledge the constructive suggestions of the Editor and unknown Reviewers, which have enhanced the quality of the manuscript.

References

1. Schwab, J. D., S. D. Kühlwein, N. Ikonomi, M. Kühl, H. A. Kestler. Concepts in Boolean Network Modeling: What Do They All Mean? – Computational and Structural Biotechnology Journal, Vol. **18**, 2020, pp. 571-582.
2. Delgado, F. M., F. Gómez-Vela. Computational Methods for Gene Regulatory Networks Reconstruction and Analysis: A Review. – Artificial Intelligence in Medicine, Vol. **95**, 2019, pp. 133-145.
3. Milano, M., G. Agapito, M. Cannataro. Challenges and Limitations of Biological Network Analysis. – BioTech. (Basel), Vol. **11**, 2022, No 3, 24.
4. Golub, T. R., D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C D. Bloomfield, E. S. Lander. Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. – Science, Vol. **286**, 1999, pp. 531-537.
5. Furey, T. S., N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, D. Haussler. Support Vector Machine Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data. – Bioinformatics, Vol **16**, 2000, pp. 906-914.
6. Khan, J., J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, P. S. Meltzer. Classification and Diagnostic Prediction of Cancers Using Gene Expression Profiling and an Artificial Neural Network. – Nature Medicine, Vol. **7**, 2001, No 6, pp. 673-679.
7. Chen, W., H. Lu, M. Wang. Gene Expression Data Classification Using Artificial Neural Network Ensembles Based on Samples Filtering. – International Conference on Artificial Intelligence and Computational Intelligence, Shanghai, China, 2009, pp. 626-628.
8. Vanitha, C. D. A., D. Devaraj, M. Venkatesulu. Gene Expression Data Classification Using Support Vector Machine and Mutual Information-Based Gene Selection. – Procedia Computer Science, Vol. **47**, 2015, pp. 13-21.
9. Fan, L., K. L. Poh, P. Zhou. A Sequential Feature Extraction Approach for Naïve Bayes Classification of Microarray Data. – Expert Systems with Applications, Vol. **36**, 2009, pp. 9919-9923.
10. Fan, L., K. L. Poh. A Comparative Study of PCA, ICA, and Class-Conditional ICA for Naïve Bayes Classifier. – In: F. Sandoval, A. Prieto, J. Cabestany, M. Graña, Eds. Conference: Computational and Ambient Intelligence, Computational and Ambient Intelligence (IWANN), Lecture Notes in Computer Science. Vol. **4507**. 2007, Berlin, Heidelberg, Springer, pp. 16-22. ISBN: 978-3-540-73006-4.
11. Maulik, U., A. Mukhopadhyay, S. Bandyopadhyay. Combining Pareto-Optimal Clusters Using Supervised Learning for Identifying Co-Expressed Genes. – BMC Bioinformatics, Vol. **10**, 2009, pp. 1-16.
12. Mukhopadhyay, A., S. Bandyopadhyay, U. Maulik. Multi-Class Clustering of Cancer Subtypes through SVM-Based Ensemble of Pareto-Optimal Solutions for Gene Marker Identification. – PLoS One, Vol. **5**, 2010, pp. 1-14.
13. Bhuvaneswari, V., K. Vanitha. Classification of Microarray Gene Expression Data by Gene Combinations Using Fuzzy Logic (MGC-FL). – International Journal of Computer Science Engineering and Application, Vol. **2**, 2012, pp. 79-98.
14. Cilia, N. D., D. Stefano, C. F. Fontanella, S. Raimondo, A. Cotto. An Experimental Comparison of Feature-Selection and Classification Methods for Microarray Datasets. – Information, Vol. **10**, 2019, No 3, pp. 1-13. DOI: 10.3390/info10030109.
15. Lee, J., I. Choi, C. H. Jun. An Efficient Multivariate Feature Ranking Method for Gene Selection in High-Dimensional Microarray Data. – Expert Systems with Applications, Vol. **166**, 2021, pp. 1-9.
16. Helmy, M., R. Agrawal, J. Ali, M. Soudy, T. T. Bui, K. Selvarajoo. GeneCloudOmics: A Data Analytic Cloud Platform for High-Throughput Gene Expression Analysis. – Frontiers in Bioinformatics, Vol. **1**, 2021, pp. 1-14.

17. W i d i h a r t o, M., A. S o e l e m a n, A. S y u k u r. Performance Improvement of Naïve Bayes Algorithm Based on Information Gain and Forward Selection Features Selection for Heart Disease Classification. – IOSR Journal of Computer Engineering, Vol. **24**, 2022, No 3, pp. 69-79.
18. W a h i d, A., M. T. B a n d a y. Classification of DNA Microarray Gene Expression Leukemia Data through the ABC and CNN Methods. – International Journal of Intelligent Systems and Application in Engineering, Vol. **11**, 2023, No 75, pp. 119-131.
19. M a j u m d e r, D. Application of Information Theory for Understanding of HLA Gene Regulation in Leukemia. – In: Advances in Computing & Information Technology, Advances in Intelligent Systems and Computing. Vol. **177**. Berlin, Heidelberg, Springer, 2013, pp.161-173. ISBN: 978-3-642-31551-0.
20. D a s, B., D. M a j u m d e r. Maximum Entropy-Based Multivariate Dependence Analysis with a Case Study for HLA Gene Regulatory Network in Human Leukemia. – International Journal of Information Engineering, Vol. **3**, 2013, No 4, pp. 137-142.
21. D a s, B., D. M a j u m d e r. Differences of HLA Gene Regulatory Network in Human Myeloid and Lymphoid Leukemias. – In: Proc. of International Conference on Bioinformatics and Systems Biology, 2018, pp. 165-169. DOI: 10.1109/BSB.2018.8770568.
22. J e t k a, T., K. N i e n a l t o w s k i, S. F i l i p p i, M. P. H. S t u m p f, M. K o m o r o w s k i. An Information-Theoretic Framework for Deciphering Pleiotropic and Noisy Biochemical Signaling. – Nature Communications, Vol. **9**, 2018, No 4591, pp. 1-9.
23. M a r t i n o, A. D., D. M a r t i n o. An Introduction to the Maximum Entropy Approach and Its Application to Inference Problems in Biology. – Heliyon, Vol. **4**, 2018, No 4, pp. 1-33.
24. C o n f o r t e, A. J., J. A. T u s z y n s k i, F. D. B a r b o s a, N. C a r e l s. Signaling Complexity Measured by Shannon Entropy and Its Application in Personalized Medicine. – Frontiers in Genetics, Vol. **10**, 2019, pp. 1-14.
25. K a r o l a k, A., S. B r a n c i a m o r e, J. S. M c C u n e, P. P. L e e. Concepts and Applications of Information Theory to Immune-Oncology. – Trends in Cancer, Vol. **7**, 2021, No 4, pp. 335-346.
26. B i l l i n g, U., T. J e t k a, L. N o r t m a n n, N. W u n d r a c k, M. K o m o r o w s k i, S. W a l d h e r r, F. S c h a p e r, A. D i t t r i c h. Robustness and Information Transfer within IL-6-Induced JAK/STAT Signaling. – Communications Biology, Vol. **2**, 2019, No 27, pp. 1-14.
27. D i x i t, P. D., E. L y a s h e n k o, M. N i e p e l, D. V i t k u p. Maximum Entropy Framework for Predictive Inference of Cell Population Heterogeneity and Responses in Signaling Networks. – Cell Systems, Vol. **10**, 2020, No 2, pp. 204-212.
28. G u o, Z., Y. F u, C. H u a n g, C. Z h e n g, Z. W u, X. C h e n, S. G a o, Y. M a, M. S h a h e n, Y. L i, P. T u, J. Z h u, Z. W a n g, W. X i a o, Y. W a n g. NOGEA: A Network-Oriented Gene Entropy Approach for Dissecting Disease Comorbidity and Drug Respositioning. – Bioinformatics, Vol. **19**, 2021, No 4, pp. 549-564.
29. A m e r i, A. J., Z. A. L e w i s. Shannon Entropy as a Metric for Conditional Gene Expression in Neurospora Crassa. – G3 Genes| Genomes| Genetics, Vol. **11**, 2021, No 4, pp. 1-7.
30. D a s, M., D. M a j u m d e r. Development of an Algorithm for Gene Expression Analysis through MaxEnt-Based Multivariate Information Theory. – In: International Conference on Intelligent Communication and Computational Techniques (ICCT'17), New York, New Jersey, IEEE, 2017, pp. 217-222. DOI: 10.1109/INTELCCT.2017.8324048.
31. G r e v e n, A., G. K e l l e r, G. W a r n e c k e. Entropy. Princeton, NJ, USA, Princeton University Press, 2014, 384 p.
32. D e m i r e l, Y., V. G e r b a u d. Nonequilibrium Thermodynamics: Transport and Rate Processes in Physical, Chemical, and Biological Systems. – Amsterdam, The Netherlands, Elsevier, 2019.
33. J a k i m o w i c z, A. The Role of Entropy in the Development of Economics. – Entropy, Vol. **22**, 2020, No 4, p. 452. DOI: 10.3390/e22040452.
34. R o s t a g h i, M., H. A z a m. Dispersion Entropy: A Measure for Time-Series Analysis. – IEEE Signal Processing Letters, Vol. **23**, 2016, pp. 610-614.

35. R e y n a r, J. C., A. R a t n a p a r k h i. A Maximum Entropy Approach to Identifying Sentence Boundaries. – In: Proc. of 5th Conference on Applied Natural Language Processing. Association for Computational Linguistics, Stroudsburg, PA, USA, 1997, pp. 16-19.
36. S h a n n o n, C. E. A Mathematical Theory of Communication. – The Bell System Technical Journal, Vol. **27**, 1948, pp. 379-423.
37. P e t r o v, I. I. Information Systems Reliability in Traditional Entropy and Novel Hierarchy. – Cybernetics and Information Technologies, Vol. **22**, 2022, No 3, pp. 1-15.
38. M a j u m d e r, D. HLA Expression in Leukemia: Status, Regulation & Therapeutic Implications of HLA Expression in Leukemia.– USA & UK: LAMBERT Academic Publishing GmbH & Co., Canada, India, Germany, 2012. ISBN: 978-3-8484-3247-9.
39. G i b b s, J. W. Elementary Principles in Statistical Mechanics. – New York, Dover Publications, 1960 (Reprint of 1902). ISBN: 10: 0486607070.
40. D a s, B., D. M a j u m d e r. Information Theory-Based Analysis for Understanding the Regulation of HLA Gene Expression in Human Leukemia. – International Journal of Information Sciences and Techniques, Vol. **2**, 2012, No 5, pp. 39-50.
41. B a n s a l l, M., V. B e l c a s t r o, A. A. I m p i o m b a t o, D. D. B e r n a r d o. How to Infer Gene Networks from Expression Profiles. – Molecular Systems Biology, EMBO, Vol. **3**, 2007, No 78, pp. 1-10.
42. T e s c h e n d o r f f, A. E., S. S e v e r i n i. Increased Entropy of Signal Transduction in the Cancer Metastasis Phenotype. – BMC Systems Biology, Vol. **4**, 2010, No 1, 104.
43. M a j u m d e r, D., A. M u k h e r j e e. A Passage through Systems Biology to Systems Medicine: Adoption of Middle-Out Rational Approaches towards the Understanding of Clinical Outcome in Cancer Therapy. – Analyst**,** Vol. **136**, 2011, pp. 663-678.
44. M a j u m d e r, D., A. M u k h e r j e e. Multi-Scale Modeling Approaches in Systems Biology Towards the Assessment of Cancer Treatment Dynamics: Adoption of Middle-out Rationalist Approach. – In: Advances in Cancer: Research & Treatment, 2013, Article ID 587889.
45. W i e r i n g e n, V., V. D. V a a r t. Statistical Analysis of the Cancer Cell's Molecular Entropy Using High-Throughput Data. – Bioinformatics, Vol. **27**, 2011, No 4, pp. 556-563.
46. M a r g o l i n, A. A., K. W a n g, A. C a l i f a n o, I. N e m e n m a n. Multivariate Dependence and Genetic Networks Inference, – IET Systems Biology, Vol. **4**, No 6, 2010, pp. 428-440.
47. GNU Octave Wiki (Assessed on 06.05.2024). **https://wiki.octave.org/Publications_using_Octave**,
48. P r i n z, H. Numerical Methods for the Life Scientist: Binding and Enzyme Kinetics Calculated with GNU Octave and MATLAB. Springer, Heidelberg, Dordrecht, London, New York, 2011.
49. R a n j a n, M. K., K. B a r o t, V. K h a i r n a r, V. R a w a l, A. P i m p a l g a o n k a r, S. S a x e n a, A. M. S a t t a r. Python: Empowering Data Science Applications and Research. – Journal of Operating Systems Development & Trends, Vol. **10**, 2023, No 1, pp. 27-33.
50. S i n g h, P., A. E. O k e, A. F. K i n e b e r, O. I. O l a n r e w a j u, O. O m o l e, M. S. S a m s u r i j a n, R. A. R a m l i. A Mathematical Analysis of 4IR Innovation Barriers in Developmental Social Work – A Structural Equation Modeling Approach. – In: Article in Mathematics, Vol. **11**, 2023, No 1003, pp. 1-20.
51. W e s t, J., G. B i a n c o n i, S. S e v e r i n i, A. E. T e s c h e n d o r f f. Differential Network Entropy Reveals Cancer System Hallmarks. – Scientific Reports. Vol. **2**, 2012, No 1, p.802.
52. B a r n e s, N. Publish Your Computer Code: It Is Good Enough. – Nature, Vol. **467**, 2010, No 7317, 753.
53. R o b e r t s, M., D. D r i g g s, M. T h o r p e et al. Common Pitfalls and Recommendations for Using Machine Learning to Detect and Prognosticate for COVID-19 Using Chest Radiographs and CT Scans. – Nature Machine Intelligence, Vol. **3**, 2021, pp. 199-217.

Received: 27.06.2025, First revision: 16.08.2025, Second revision: 23.09.2025, Third revision: 13.10.2025, Accepted: 20.10.2025