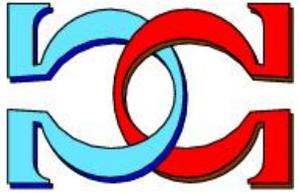
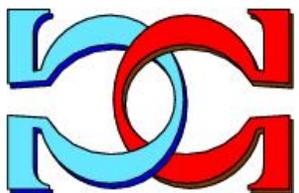
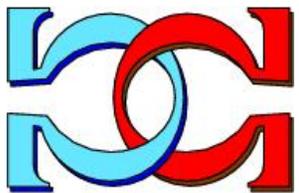


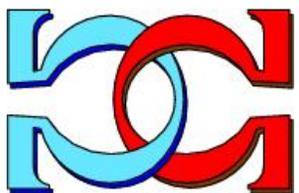
**CDMTCS
Research
Report
Series**



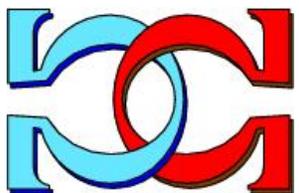
**SAIN: Search-And-INfer,
A Mathematical and
Computational Framework
for Personalised Multimodal
Data Modelling with
Applications in Health Care**



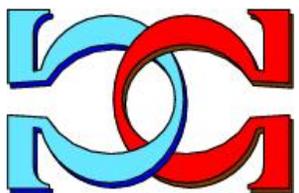
**C. S. Calude¹, P. Gladding¹,
A. Henderson², N. Kasabov³**



¹University of Auckland, New Zealand
²University of Queensland, Australia
Auckland University of Technology, New
Zealand



CDMTCS-583
June 2025



Centre for Discrete Mathematics and
Theoretical Computer Science

SAIN: Search-And-INfer, A Mathematical and Computational Framework for Personalised Multimodal Data Modelling with Applications in Health Care

Cristian S. Calude¹, Patrick Gladding², Alec Henderson³, Nikola Kasabov^{4,5}

¹ University of Auckland, New Zealand

²

³ University of Queensland, Australia

⁴ Auckland University of Technology, New Zealand

⁵ Knowledgeengineering.ai, New Zealand

Personalised modelling has become a dominant approach in personalised medicine and precision health. It creates a computational model for an individual based on large data repositories of existing personalised data, aiming to achieve the best possible personal diagnosis or prognosis and derive an informative explanation for it. Current methods are still working on a single data modality or treating all modalities with the same method. The proposed method, SAIN (Search-And-INfer), offers better results and an informative explanation for classification and prediction tasks on a new multimodal object (sample) using a database of similar multimodal objects. The method is based on different distance measures suitable for each data modality and introduces a new formula to aggregate all modalities into a single vector distance measure to find the closest objects to a new one and then use them for a probabilistic inference.

The paper describes SAIN and applies it to two types of multimodal data, cardiovascular diagnosis and EEG time series, modelled by integrating modalities, such as numbers, categories, images, and time series, and using a software implementation of SAIN.

Keywords: search in multimodal data; inference in multimodal data; personalised modelling; precision health.

1. Introduction

Multimodal data processing has become a new data science trend with applications in neuro-imaging, health diagnosis and prognosis, environmental modeling, and financial modelling, see [1–3].

Methods for searching relevant items in databases have been developed and used for decades, improving their accuracy and speed. These searches are a significant part of personalised modelling (e.g. precision medicine), where an optimal model is created for a given vector of person x data X and a database D of past personalised records with labelled outcomes to predict x behaviour. Most methods (e.g. [3,4]) select a subset D_X of closest vectors to X from the database D (for example, the K -nearest neighbours) and build a machine learning model using this subset D_X . To select a subset D_X of the closest vectors to X , the method searches D using predominantly Euclidean or Hamming distances to measure the similarity between the new vector X and the vectors in D . These methods have been applied in many applications and constitute the state-of-the-art in the field (e.g. [5–8]).

The enormous growth of personal multimodal data worldwide demands more advanced personalisation of search and inference methods. Most methods for multi-modal data represent all modalities of data for one object as a vector and then apply a single machine learning method, such as a deep neural network or a statistical regression (e.g. [9,10]). In these cases, the specificity of each data modality cannot be considered, which negatively impacts the inference results and explanation.

2. General Description of SAIN

The proposed new method offers new functionality and features for personalised search and model creation in multimodal data, some of which are listed below. The method

1. is suitable for multimodal data searches in heterogeneous data sets, e.g. numbers, text, images, sound, and categorical data,
2. uses a novel mathematical similarity measure superseding a single (e.g. Euclidean, Hamming) distance used in the existing methods. In this way, inaccurate measurement of similarity on a large number of heterogeneous variables is avoided,
3. search is fast even on large data sets, with millions of records and thousands of variables,
4. includes advanced personalised searches with multiple parameters and other features,
5. facilitates multiple solutions with corresponding probabilities,
6. is suitable for unsupervised clustering in multimodal heterogeneous data,
7. is suitable for personalised model creation to classify or predict specific outcomes based on multimodal and heterogeneous data.

3. Mathematical Description

In this section, we present the mathematical method.

3.1. Database

We will work with the multidimensional data described as follows:

1. $m > 1$ objects (samples) o_1, \dots, o_m ,
2. each object o_i ($1 \leq i \leq m$) is defined by $n > 1$ criteria (variables) c_1, \dots, c_n with values in linearly ordered domains D_i with $\min D_i$ and $\max D_i$; if some value $a_{i,j} \in D_i$ ($1 \leq i \leq m, 1 \leq j \leq n$) is either missing or uncertain, then its value is recorded as ∞ ,
3. $n > 1$ weights w_1, \dots, w_n in $(0, 1)$ with $\sum_{i=1}^n w_i = 1$, where each w_i ($1 \leq i \leq n$) quantifies the importance of the criterion c_i ; if $w_i = \frac{1}{n}$ for all $1 \leq i \leq n$, then all criteria are equally important; a criterion c_i is ignore if $w_i = 0$.

Data of independent variables are organised as follows:

Objects/Criteria	c_1	c_2	...	c_j	...	c_n
o_1	$a_{1,1}$	$a_{1,2}$...	$a_{1,j}$...	$a_{1,n}$
\vdots	\vdots	\vdots	...	\vdots	...	\vdots
o_i	$a_{i,1}$	$a_{i,2}$...	$a_{i,j}$...	$a_{i,n}$
\vdots	\vdots	\vdots	...	\vdots	...	\vdots
o_m	$a_{m,1}$	$a_{m,2}$...	$a_{m,j}$...	$a_{m,n}$
w	w_1	w_2	...	w_j	...	w_n

Table 1. Unlabelled database

3.2. Distance metrics

A distance metric on a space X is a positive real-valued function $d : X \times X \rightarrow \mathbf{R}_+$ satisfying the following three conditions for all $x, y, z \in X$: a) $d(x, y) = 0$ if and only if $x = y$, b) $d(x, y) = d(y, x)$, c) $d(x, z) \leq d(x, y) + d(y, z)$.

The domains $X = D_i$ can vary greatly: they can be sets of logical values, rational numbers, percentages, digitally codified images, sounds, videos, and many others. We use a bounded distributive complemented lattice $(L, \vee, \wedge, \bar{\cdot}, 0, 1)$ to describe uniformly the domains D_i , [11,12].

Here is a list with illustrative, but far from exhaustive, examples of domains D_i :

- Logical Boolean domain: $(\{0, 1\}, \max, \min, \bar{\cdot}, 0, 1)$, where $\bar{x} = 1 - x, x \in \{0, 1\}$.
- Logical non-Boolean domain: $(\{0, \frac{1}{N-1}, \frac{2}{N-1}, \dots, \frac{N-2}{N-1}, 1\}, \max, \min, \bar{\cdot}, 0, 1)$, where $x \in \{0, \frac{1}{N-1}, \frac{2}{N-1}, \dots, \frac{N-2}{N-1}, 1\}$ and $\bar{x} = 1 - x$.
- Numerical domain with natural values: $(\{0, 1, \dots, N\}, \max, \min, \bar{\cdot}, 0, 1)$, where $\bar{x} = N - x, x \in \{0, 1, \dots, N\}$.
- Numerical domain with rational values: $(\{x \mid a \leq x \leq A\}, \max, \min, \bar{\cdot}, a, A)$, where $\bar{x} = A - x, a \leq x \leq A$.
- Binary code: $(\{0, 1\}^n, \max, \min, \bar{\cdot}, 00 \dots 0, 11 \dots 1)$, where the domain consists of all binary strings of length n , $\{0, 1\}^n = \{x_1 x_2 \dots x_n \mid x_i \in \{0, 1\}\}$ and for all $x_1 x_2 \dots x_n, y_1 y_2 \dots y_n \in \{0, 1\}^n$, $\max(x_1 x_2 \dots x_n, y_1 y_2 \dots y_n) = \max(x_1, y_1) \max(x_2, y_2) \dots \max(x_n, y_n)$, $\min(x_1 x_2 \dots x_n, y_1 y_2 \dots y_n) = \min(x_1, y_1) \min(x_2, y_2) \dots \min(x_n, y_n)$, $\bar{x}_1 \bar{x}_2 \dots \bar{x}_n = (1 - x_1)(1 - x_2) \dots (1 - x_n)$.

In the lattice $(L, \vee, \wedge, \bar{\cdot}, 0, 1)$ we introduce, following [11], the metric:

$$d(x, y) = \begin{cases} (x \wedge \bar{y}) \vee (\bar{x} \wedge y), & \text{if } x \neq y, \\ 0, & \text{otherwise,} \end{cases}$$

for $x, y \in L$. This metric d can be extended to $L \cup \{\infty\}$ as follows:

$$d_\infty(x, y) = \begin{cases} d(x, y), & \text{if } x, y \in L, \\ \sigma(x), & \text{if } x \in L \text{ and } y = \infty, \\ \sigma(y), & \text{if } y \in L \text{ and } x = \infty, \\ 0, & \text{otherwise,} \end{cases}$$

where $\sigma(x) = \max(x, \bar{x})$.

The metrics $d_{\infty,i}$ on $L_i \cup \{\infty\}$, $1 \leq i \leq n$, can be extended to $(L_i \cup \{\infty\})^n$, i.e. to n -dimensional vectors, as follows:

$$d_\infty(x_1 x_2 \dots x_n, y_1 y_2 \dots y_n) = \sum_{i=1}^n d_{\infty,i}(x_i, y_i), \quad (1)$$

where $x_i, y_i \in L_i \cup \{\infty\}, 1 \leq i \leq n$.

In what follows, we write d for d_∞ when the meaning is clear from the context.

3.3. Tasks specification

Data organised as in Table 2 consists of independent objects augmented with a column of labels, the weights of criteria, and a new unlabelled object, see Table 2 and Table 3.

Additional information associated with data in Table 2 may include the range of each criterion c_j and the associated specific distance, e.g. the Euclidean distance for real numbers and the distance d for binary strings or strings over a non-binary alphabet (e.g. for images or colours).

Objects/Criteria	c_1	c_2	...	c_j	...	c_n	Class label
o_1	$a_{1,1}$	$a_{1,2}$...	$a_{1,j}$...	$a_{1,n}$	l_1
\vdots	\vdots	\vdots	...	\vdots	...	\vdots	\vdots
o_i	$a_{i,1}$	$a_{i,2}$...	$a_{i,j}$...	$a_{i,n}$	l_i
\vdots	\vdots	\vdots	...	\vdots	...	\vdots	\vdots
o_m	$a_{m,1}$	$a_{m,2}$...	$a_{m,j}$...	$a_{m,n}$	l_m

Table 2. Labelled database

Criteria weights	c_1	c_2	...	c_j	...	c_n
w	w_1	w_2	...	w_j	...	w_n

Table 3. Weights

Object/Criteria	c_1	c_2	...	c_j	...	c_n
x	x_1	x_2	...	x_j	...	x_n

Table 4. New unlabelled object

We consider the following tasks:

Task 1: Calculate the distance (or similarity metric) between the new object and each object in Table 2. If the distance corresponding to c_i is d_i , then

$$d(o_j, x) = \sum_{i=1}^n w_i \cdot d_i(a_{i,j}, x_j).$$

Task 2: Given a threshold $\delta > 0$, calculate all objects o_i at a distance at most δ to x .

Task 3: Calculate the probability of a new object to belong to a labelled class (e.g. low risk vs. high risk) using a threshold δ and Table 2.

Task 4: Rank the criteria in Table 2 and calculate the marker or markers criterion/criteria, that is the most important one/ones.

Task 5: Assign alternative weights to criteria.

Task 6: Test the accuracy of data and method for Task 4.

3.4. Tasks Solutions

For Task 1 we calculate the distances $d_\infty(o_i, x)$ between each object o_i in Table 2 and x in Table 4.

For Task 2, given a threshold $\delta > 0$, we calculate all objects in Table 2 at a distance at most δ to x , that is, the objects which are δ -similar to x :

$$C_{\delta, x} = \{o_i \mid d(x, o_i) \leq \delta, 1 \leq i \leq m\},$$

and its complement $\overline{C_{\delta, x}}$.

For Task 3 we calculate the probability that x is in class label l_t , which is the ratio of the number of objects in $C_{\delta, x}$ with the label l_t to the size of the cluster $C_{\delta, x}$:

$$Prob(x \text{ has label } l_t) = \frac{\#\{o_i \in C_{\delta, x} \mid l_i = l_t\}}{\#(C_{\delta, x})},$$

where $\#\{\dots\}$ means the number of elements in the set $\{\dots\}$.

For Task 4, we work with Table 2. Recall that for each criterion c_i we have a domain D_i augmented with information "high" or "low," indicating whether higher or lower values are desirable. Based on this information, we can construct a hypothetical object which has as the most desirable values for each criterion: one could see this object as an "exemplar" one.

Object/Criteria	c_1	c_2	...	c_j	...	c_n	Class label
o_E	n_1	n_2	...	n_j	...	n_n	l_h

Table 5. Hypothetical exemplar object

Sometimes, criteria are interrelated or correlated. This means that in some cases, there is no unique "exemplar object", but a couple of them have to be studied in ranking the importance of criteria.

For example, fix an "exemplar object" o_E .

1. Compute the distances $d_\infty(o_i, o_E)$ between each object o_i in Table 2 and o_E , so obtain a vector with n non-negative real components $V_0 = (d_1^0, \dots, d_n^0)$.
2. For each $1 \leq t \leq m$, compute the distances $d_\infty(o_i, o_E)$ taking into consideration all criteria in Table 2 **except** c_t : obtain the vector $V_t = (d_1^t, \dots, d_n^t)$.
3. Compute the distances between $dist(V_0, V_t)$, $1 \leq t \leq m$ using the formula

$$dist(V_0, V_t) = \sum_{i=1}^n |d_{0,i} - d_{t,i}|,$$

and sort them in increasing order. The criterion c_t is a marker if $dist(V_0, V_t) \geq dist(V_0, V_j)$, for every $1 \leq j \leq m$.

We repeat this procedure for each "exemplar object" and study possible variations.

For Task 5, normalise the distances $dist(V_0, V_t)$ and use these values to construct the weights w_i^* , $1 \leq t \leq m$.

For Task 6 assume we have weights (w_i) associated to Table 2 (see Table 1). To test the accuracy of the data and method used for Task 4, compare the original weights (w_i) with (w_i^*). Serious discrepancies should signal issues either with the data or the choices made in the applications of the method.

3.5. An example

We illustrate the above tasks with an example of a labelled database in (see Table 6) and a new object (see Table 7), all having the following seven characteristics (the last column has the label classes 1 and 2):

c_1 : real number $\{0 - 100\}$, e.g. age, weight, BMI etc.;

c_2 : Boolean value $\{0, 1\}$, e.g. gender;

c_3 : integer number $\{0 - 10,000\}$, e.g. gene expression;

c_4 : categorical $\{\text{small, med, large}\}$, e.g. size of tumour, body size, keywords;

c_5 : colour $\{\text{red, yellow, white, black}\}$, e.g. colour of a spot on the body, on the heart;

c_6 : spike sequence of $\{-1, 0, 1\}$ e.g. encoded EEG, ECG;

c_7 : black and white image, e.g. MRI, face image.

In this example, for simplicity, we didn't use weights.

The first step is to code the data in Tables 6 and 7. The new data is in Table 8 and Table 9.

Then, we normalise the data in Table 8 and Table 9 – the entries in the first, third and fourth columns have been divided by 100, 10,000 and 2, respectively, and the entries in the last three columns have been transformed in reals in the unit interval, and the column of labels has been removed.

In this way, we have obtained Table 10 and Table 11.

Then, we choose an appropriate distance according to each criterion. In this example, we used the Euclidean distance for all criteria.

We can compute $C_{\delta,x} = \{o_i \mid d(o_i, x) \leq \delta\}$ and, accordingly, the probability that x would be labelled in class 1 or class 2.

If $\delta = 3.5$, then $C_{3.5,x} = \{o_1, o_2, o_3, o_5, o_6, o_7, o_8\}$ so the probability that x is in class 1 is $2/7$ and the probability that x is in class 2 is $5/7$. If $\delta = 2.5$, then its closest cluster is $C_{2.5,x} = \{o_2, o_3, o_5, o_6, o_7, o_8\}$, so the probability that x is in class 1 is $1/3$ and the probability that x is in class 2 is $2/3$.

which induces a ranking of the objects in the Table 8: $o_3, o_6, o_7, o_5, o_2, o_8, o_1, o_9, o_2$.

68.2	0	6789	small	red	0,1,-1,-1,1,1,0,0, 1,-1	1,1,0 0,0,1 0,0,1	1
93	1	98000	medium	yellow	0,-1,-1,-1,-1,0,0, 1,-1,1	1,0,0 0,0,1 0,0,1	1
44.5	1	5600	large	red	0,1,-1,1,-1,1,0,0, 1,-1	1,1,0 1,0,1 1,1,1	1
56.8	0	89	small	white	1,-1,-1,-1,-1,1,0,0, 1,-1	1,1,0 0,1,1 1,0,1	1
26.3	0	9456	large	black	1,-1,-1,-1,0,1,0,0, 1,-1	1,1,0 1,1,1 1,0,1	2
81.5	1	78955	medium	red	0, 1,-1,1,-1,-1,0,0, 1,-1	1,1,0 0,0,1 1,1,1	2
56.7	1	68900	small	black	1,- 1,-1,1,-1,1,0,0, 1,1	1,1,1 0,0,1 1,1,1	2
20	0	7833	large	yellow	1,1,-1,-1,1,1,0,-1, -1,1	1,0,0 0,0,1 1,1,1	2
20	0	7833	∞	yellow	1,1,-1,-1,1,1,0,-1, -1,1	1,0,0 0,0,1 1,1,1	2

Table 6. Example of labelled data

48.5	1	45679	large	red	1, 0, 0, -1, 1, -1, 1, 0, 0, 1	1,1,0 0,0,1 1,0,1
------	---	-------	-------	-----	--------------------------------	-------------------------

Table 7. Example of new unlabelled object

For Task 4, assume that the criteria c_1, \dots, c_7 in Table 10 have the additional information (m, m, m, m, m, M, M) ,
where m (M) means that the exemplar value is the minim (maximum) value. Based on this vector, we compute the
exemplar object:

Next we calculate V_0, \dots, V_t , see Table 14, and finally the distances $Dist(V_0, V_t)$, $t = 1, 2, \dots, 7$ and the weights
as their normalised values, see Table 16. The marker, in this case, is the criterion c_5 .

o_1	68.2	0	6789	0	FF0000 111111110000000000000000	0122110012	110001001	1
o_2	93	0	98000	1	FFFF00 111111111111111110000000	0222200121	110001001	1
o_3	44.5	1	5600	2	FF0000 111111110000000000000000	0121210012	110101111	1
o_4	56.8	0	89	0	FFFFFF 111111111111111111111111	1222210012	110011101	1
o_5	26.3	0	9456	2	000000 000000000000000000000000	1222010012	110111101	2
o_6	81.5	1	78955	1	FF0000 111111110000000000000000	0121220012	110001111	2
o_7	56.7	1	68900	0	000000 000000000000000000000000	1221210011	111001111	2
o_8	20	0	7833	2	FFFF00 111111111111111110000000	1122110221	100001111	2
o_9	20	0	7833	∞	FFFF00 111111111111111110000000	1122110221	100001111	2

Table 8. Coded labelled data

x	48.5	1	45679	2	FF0000 111111110000000000000000	1002121001	110001101
-----	------	---	-------	---	------------------------------------	------------	-----------

Table 9. New unlabelled object coded

o_1	0.682	0	0.06789	0	0.2	0.0122110012	0.110001001
o_2	0.93	1	0.98	0.5	0.6	0.0222200121	0.100001001
o_3	0.445	1	0.056	1	0.2	0.0121210012	0.110101111
o_4	0.568	0	0.00089	0	1	0.1222210012	0.110011101
o_5	0.263	0	0.09456	1	0	0.1222010012	0.110111101
o_6	0.815	1	0.78955	0.5	0.2	0.0121220012	0.110001111
o_7	0.567	1	0.689	0	0	0.1221210011	0.111001111
o_8	0.2	0	0.07833	1	0.6	0.1122110221	0.100001111
o_9	0.2	0	0.07833	∞	0.6	0.1122110221	0.100001111

Table 10. Coded labelled normalised data

x	0.485	1	0.45679	1	0.2	0.1002121001	0.110001101
-----	-------	---	---------	---	-----	--------------	-------------

Table 11. New unlabelled object coded normalised

$d(o_1, x)$	0.197	1	0.3889	1	0	0.4	0.11111111	3.09701111
$d(o_2, x)$	0.445	0	0.52321	0.5	0.33333333	0.6	0.22222222	2.62376556
$d(o_3, x)$	0.04	0	0.40079	0	0	0.5	0.22222222	1.16301222
$d(o_4, x)$	0.083	1	0.4559	1	0.66666667	0.45	0.11111111	3.76667778
$d(o_5, x)$	0.222	1	0.36223	0	0.33333333	0.45	0.22222222	2.58978556
$d(o_6, x)$	0.33	0	0.33276	0.5	0	0.45	0.11111111	1.72387111
$d(o_7, x)$	0.082	0	0.23221	1	0.33333333	0.45	0.22222222	2.31976556
$d(o_8, x)$	0.285	1	0.37846	0	0.33333333	0.45	0.22222222	2.66901556
$d(o_9, x)$	0.285	1	0.37846	1	0.33333333	0.45	0.22222222	3.66901556

Table 12. Normalised distances from the new object to all objects

$d(o_3, x)$	0.04	0	0.40079	0	0	0.5	0.22222222	1.16301222
$d(o_6, x)$	0.33	0	0.33276	0.5	0	0.45	0.11111111	1.72387111
$d(o_7, x)$	0.082	0	0.23221	1	0.33333333	0.45	0.22222222	2.31976556
$d(o_5, x)$	0.222	1	0.36223	0	0.33333333	0.45	0.22222222	2.58978556
$d(o_2, x)$	0.445	0	0.52321	0.5	0.33333333	0.6	0.22222222	2.62376556
$d(o_8, x)$	0.285	1	0.37846	0	0.33333333	0.45	0.22222222	2.66901556
$d(o_1, x)$	0.197	1	0.3889	1	0	0.4	0.11111111	3.09701111
$d(o_9, x)$	0.285	1	0.37846	1	0.33333333	0.45	0.22222222	3.66901556
$d(o_4, x)$	0.083	1	0.4559	1	0.66666667	0.45	0.11111111	3.76667778

Table 13. Ranking of distances in increasing order in Table 12

o_E	0.2	0	0.00089	0	1	0.1222210012	0.100001001
-------	-----	---	---------	---	---	--------------	-------------

Table 14. Exemplar object

V_0	V_1	V_2	V_3	V_4	V_5	V_6	V_7
1.469	0.987	1.469	1.402	1.469	0.669	1.359	1.459
3.709	2.979	2.709	2.730	3.209	3.309	3.609	3.709
3.220	2.975	2.220	3.165	2.220	2.420	3.110	3.210
0.378	0.010	0.378	0.378	0.378	0.378	0.378	0.368
2.167	2.104	2.167	2.073	1.167	1.167	2.167	2.157
3.824	3.209	2.824	3.035	3.324	3.024	3.714	3.814
3.066	2.699	2.066	2.378	3.066	2.066	3.066	3.055
1.487	1.487	1.487	1.410	0.487	1.087	1.477	1.487
1.487	1.487	1.487	1.410	0.487	1.087	1.477	1.487

Table 15. Vectors V_0, \dots, V_m , rounded to two decimals

Distances	2.870	4.00	2.826	5.00	5.60	0.450	0.061
Weights	0.137	0.192	0.135	0.240	0.269	0.021	0.002

Table 16. Distances $Dist(V_0, V_i)$ and (normalised) weights

4. Survival analysis in SAIN

Medical survival analysis evaluates the time until an event of interest occurs, like death or disease recurrence, in a group of patients. This analysis is often used to compare treatment outcomes or predict prognosis.

4.1. Data and tasks

We are given the following data:

- Table 17 in which the first column lists the patients treated for the same disease with the same method under strict conditions and the last column records the times till the patient's deaths.

Patients/Criteria	c_1	c_2	...	c_j	...	c_n	Units of time
p_1	$a_{1,1}$	$a_{1,2}$...	$a_{1,j}$...	$a_{1,n}$	t_1
\vdots	\vdots	\vdots	...	\vdots	...	\vdots	\vdots
p_i	$a_{i,1}$	$a_{i,2}$...	$a_{i,j}$...	$a_{i,n}$	t_i
\vdots	\vdots	\vdots	...	\vdots	...	\vdots	\vdots
p_m	$a_{m,1}$	$a_{m,2}$...	$a_{m,j}$...	$a_{m,n}$	t_m

Table 17. Survival database

- Table 18, which includes the record of the new patient p .
- A threshold δ which defines the acceptable similarity between p and the relevant p_i 's in the Survival database (i.e. $d(p, p_i) \leq \delta$).

We consider the following tasks:

Task 1: What is the life expectancy of p ?

Task 2: What is the probability that the life expectancy of p is greater than or equal to a given T ?

4.2. Tasks solutions

Using a standard method of survival analysis

- For Task 1,
 - Compute the set of patients that are similar up to δ to p :

$$C_{\delta,p} = \{p_i \mid d(p, p_i) \leq \delta, 1 \leq i \leq m\}. \quad (2)$$

- Using $C_{\delta,p}$, compute the probability that p will survive the time t_j :

$$Prob_{\delta}(p \text{ survives time } t_j) = \frac{\#\{p_i \in C_{\delta,p} \mid t_i = t_j\}}{\#(C_{\delta,p})}. \quad (3)$$

Patient/Criteria	c_1	c_2	...	c_j	...	c_n
p	x_1	x_2	...	x_j	...	x_n

Table 18. New patient record

(c) Compute the life expectancy of p using the formula:

169

$$LE_\delta(p) = \sum_{j=1, t_j \in C_{\delta,p}}^m t_j \times Prob_\delta(p \text{ survives time } t_j). \quad (4)$$

2. For Task 2, calculate the probability that the life expectancy of p is at least time T :

170

$$Prob_\delta(LE(p) \geq T) = \sum_{j=1, t_j \in C_{\delta,p}, t_j \geq T}^m Prob_\delta(p \text{ survives time } t_j). \quad (5)$$

4.3. An example

171

We illustrate the above tasks with an example of a database in which columns 2–8 record patients' medical test results, and the last column records time to death (see Table prec) and a new patient (see Table 20):

172

173

174

patients	c_1	c_2	c_3	c_4	c_5	c_6	c_7	units of time
p_1	0.682	0	0.06789	0	0.2	0.012211001	0.110001001	12.3
p_2	0.93	1	0.98	0.5	0.6	0.022220012	0.100001001	15
p_3	0.445	1	0.056	1	0.2	0.012121001	0.110101111	68
p_4	0.568	0	0.00089	0	1	0.122221001	0.110011101	1.4
p_5	0.263	0	0.09456	1	0	0.122201001	0.110111101	40.5
p_6	0.815	1	0.78955	0.5	0.2	0.012122001	0.110001111	97.2
p_7	0.567	1	0.689	0	0	0.122121001	0.111001111	97.2
p_8	0.2	0	0.07833	1	0.6	0.112211022	0.100001111	55.7
p_9	0.2	0	0.07833	∞	0.6	0.112211022	0.100001111	63.7

Table 19. Patient records

x_p	0.485	1	0.45679	1	0.2	0.1002121001	0.110001101
-------	-------	---	---------	---	-----	--------------	-------------

Table 20. New patient records

The distance for column 4 is $d(x, y) = |x - y|$ and $d_\infty(x, \infty) = \max(x, 1 - x)$. For example, $d_\infty(1, \infty) = \max(1, 1 - 1) = 1$. For all other columns, the distance is $d(x, y) = |x - y|$. Finally, the total distance is the sum of individual distances (7 terms), with the results in Table 21.

175

176

177

The results for Task 1, (a), (b) and (c) are listed below:

178

1. For $\delta \geq 3.37$, $C_{\delta,p} = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9\}$, that is the entire database. Then

179

(a) $LE_\delta(p) = 50.11$,

180

(b) i. $Prob_\delta(p \text{ survives time } = 12.3) = 1/9$,

181

ii. $Prob(p \text{ survives time } = 15) = 1/9$,

182

iii. $Prob_\delta(p \text{ survives time } = 68) = 1/9$,

183

iv. $Prob_\delta(p \text{ survives time } = 1.4) = 1/9$,

184

v. $Prob_\delta(p \text{ survives time } = 40.5) = 1/9$,

185

vi. $Prob_\delta(p \text{ survives time } = 97.2) = 2/9$,

186

vii. $Prob_\delta(p \text{ survives time } = 55.7) = 1/9$,

187

viii. $Prob_\delta(p \text{ survives time } = 63.7) = 1/9$.

188

(c) i. $Prob_\delta(LE_\delta(p) \geq 1.4) = 1$,

189

ii. $Prob_\delta(LE_\delta(p) \geq 12.3) = 8/9$,

190

iii. $Prob_\delta(LE_\delta(p) \geq 15) = 7/9$,

191

iv. $Prob_\delta(LE_\delta(p) \geq 40.5) = 6/9$,

192

v. $Prob_\delta(LE_\delta(p) \geq 55.7) = 5/9$,

193

vi. $Prob_\delta(LE_\delta(p) \geq 63.7) = 4/9$,

194

vii. $Prob_\delta(LE_\delta(p) \geq 68) = 3/9$,

195

viii. $Prob_\delta(LE_\delta(p) \geq 97.2) = 2/9$,

196

	d_1	d_2	d_3	d_4	d_5	d_6	d_7	Distance d
$d(p_1, p)$	0.1970	1	0.388900	1.0	0.0	0.08800109890	0.000000100	2.67390119890
$d(p_2, p)$	0.4450	0	0.523210	0.5	0.4	0.07799208800	0.010000100	1.95620218800
$d(p_3, p)$	0.0400	0	0.400790	0.0	0.0	0.08809109890	0.000100010	0.52898110890
$d(p_4, p)$	0.0830	1	0.455900	1.0	0.8	0.02200890110	0.000010000	3.36091890110
$d(p_5, p)$	0.2220	1	0.362230	0.0	0.2	0.02198890110	0.000110000	1.80632890110
$d(p_6, p)$	0.3300	0	0.332760	0.5	0.0	0.08809009890	0.000000010	1.25085010890
$d(p_7, p)$	0.0820	0	0.232210	1.0	0.2	0.02190890100	0.001000010	1.53711891100
$d(p_8, p)$	0.2850	1	0.378460	0.0	0.4	0.01199892200	0.009999990	2.08545891200
$d(p_9, p)$	0.2850	1	0.378460	1.0	0.4	0.01199892200	0.009999990	3.08545891200

Table 21. Distances between all patients and the new patient

We can calculate other probabilities, for example, $Prob_\delta(LE_\delta(p) \geq 60) = Prob_\delta(LE_\delta(p) \geq 63.7) + Prob_\delta(LE_\delta(p) \geq 68) = 3/9 + Prob_\delta(LE_\delta(p) \geq 97.2) = 2/9 = 1/9 + 1/9 + 2/9 = 4/9$.

2. For $\delta \geq 2.5$, $C_{\delta,p} = \{v_2, v_3, v_5, v_6, v_7, v_8\}$. Then

(a) $LE_\delta(p) = 62.27$,

(b) i. $Prob(p \text{ survives time} = 15) = 1/6$,

ii. $Prob(p \text{ survives time} = 68) = 1/6$,

iii. $Prob(p \text{ survives time} = 40.5) = 1/6$,

iv. $Prob(p \text{ survives time} = 97.2) = 2/6$,

v. $Prob(p \text{ survives time} = 55.7) = 1/6$,

(c) i. $Prob_\delta(LE_\delta(p) \geq 15) = 1$,

ii. $Prob_\delta(LE_\delta(p) \geq 40) = 5/6$,

iii. $Prob_\delta(LE_\delta(p) \geq 55.7) = 4/6$,

iv. $Prob_\delta(LE_\delta(p) \geq 68) = 3/6$,

v. $Prob_\delta(LE_\delta(p) \geq 97.2) = 2/6$.

Similarly, we can calculate the probabilities $Prob_\delta(LE_\delta(p) \geq 45) = 4/6$, $Prob_\delta(LE_\delta(p) \geq 100) = 0$.

5. SAIN: A Modular Diagram and Functional Information Flow

In Figure 1 and Figure 2, we present the modular diagram and the functional information flow of SAIN.

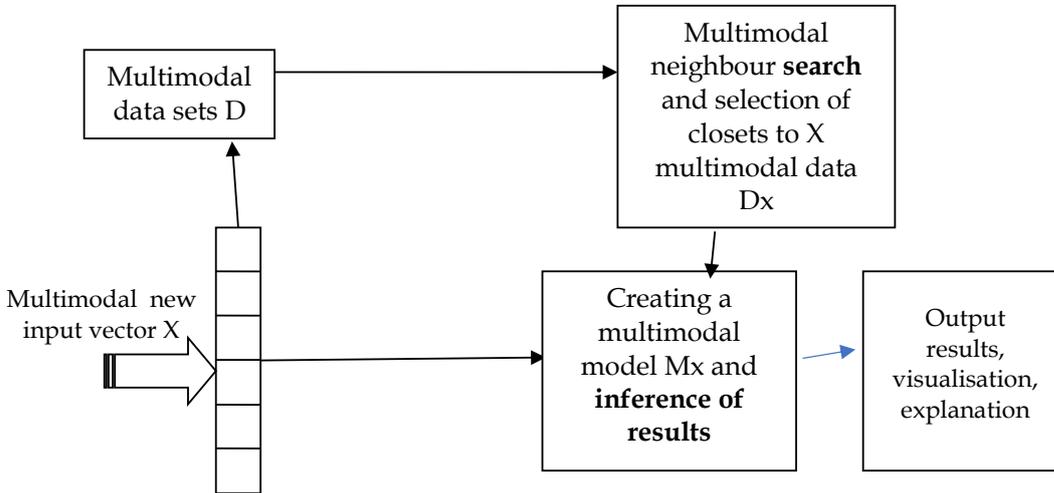


Figure 1. A modular diagram of the proposed SAIN computational framework

6. Case Studies for Medical Diagnosis and Prognosis

We present three case studies in which we applied SAIN.

6.1. Heart disease diagnosis

We worked with the well-known Cleveland dataset, which contains multiple data types [13]. The UCI Heart Disease data set contains 76 attributes. As in most articles, the attributes in our experiment data were restricted to 14, see Table 22.

The problem is a binary classification on whether the patient has or does not have heart disease.

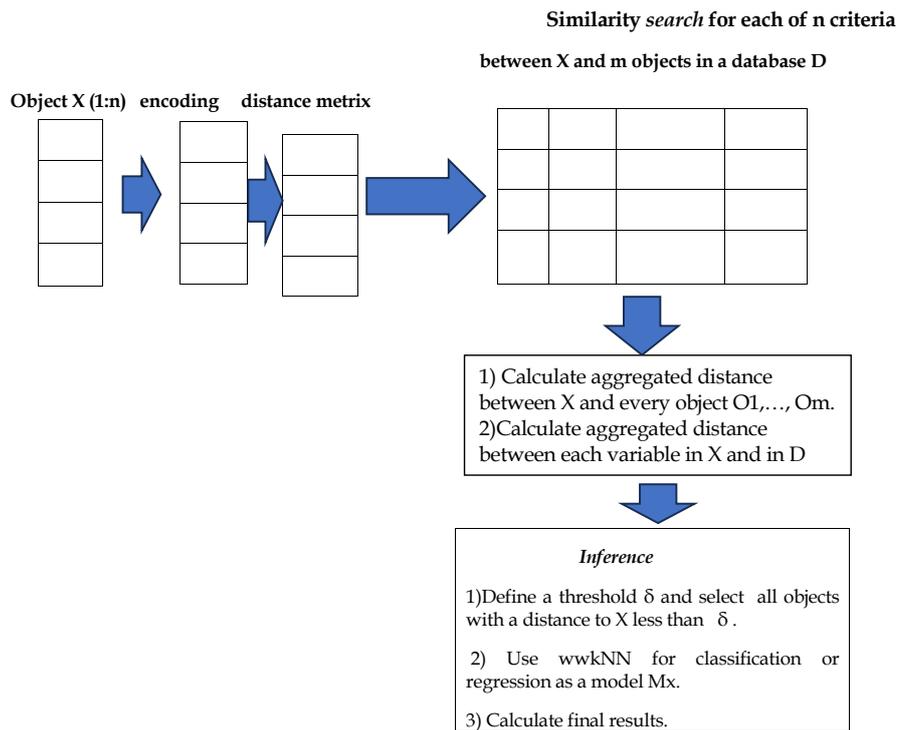


Figure 2. A flow of data and information processing in the SAIN computational framework

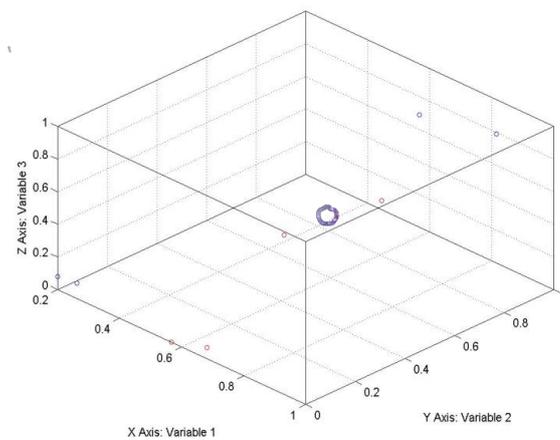


Figure 3. The neighborhood D_x of a new sample/vector X (as a circle) consists of samples of class 1 (in red) and class two (in blue). The sample X is classified in class 1 based on wwkNN method [14]. The samples/vectors are represented in a 3D X,Y,Z space of the top 3 ranked variables, which can belong to different modalities.

Name	Data type	Definition
age	integer	age in years
sex	binary	sex
cp	{1,2,3,4}	chest pain type
trestbps	integer	resting blood pressure
chol	integer	serum cholesterol in mg/dl
fbs	binary	fasting blood sugar > 120 mg/d
restecg	{0,1,2}	resting electrocardiographic results
thalach	I integer	maximum heart rate achieved
exang	binary	exercise-induced angina
oldpeak	float	ST depression induced by exercise relative to rest
slope	{1,2,3}	the slope of the peak exercise ST segment
ca	{0,1,2,3,}	number of major vessels colored by flourosopy
thal	{3,6,7}	heart status
num	{0,1,2,3,4}	diagnosis of heart disease

Table 22. The 14 variables used in the heart disease diagnosis case

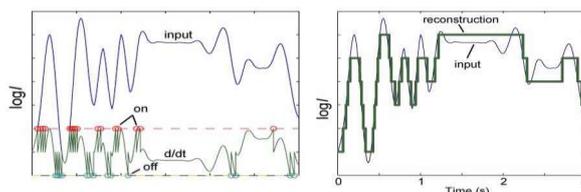


Figure 4. Every time series can be represented as a 3-value vector through a spike encoding method over time [15]. If at a time t the times series is increasing in value, there will be a positive spike (1), if decreasing – negative spike (-1) and if no change – no spike (0) (left figure). Each element in this vector represents the change of the signal at a time. If necessary, the original signal can be recovered over time using this vector (right figure). The length of the vector is equal the time points measured.

First, we selected suitable distance metrics and weights to classify the attributes. For binary objects, the distance metric is simply whether they are equal; for non-binary discrete objects such as resting electrocardiographic results, the appropriate distance measure is not obvious and should be informed by an expert. We give electrocardiographic results 0 for normal, 1 for having ST-T wave abnormality, and 2 for showing probable or definite left ventricular hypertrophy following Estes' criteria.

Many studies with the Cleveland dataset have been tested with different machine learning techniques. For example, [14] lists different algorithms and performances ranging from 47% to 80% accuracy. SAIN achieved an 82% accuracy score. Why SAIN? The search is fast, it uses appropriate distances chosen by a medical expert, it provides explainability at a personal level including probabilities. It offers different scenarios for modelling by experimenting different sets of features, parameters and preferred outcome visualisations.

6.2. Time series classification

Many data sets for classifying outcomes of events consist of multiple time series. Each variable in a time series may depend on other variables that change in time. The proposed model can deal with this problem by encoding time series (signal) into binary vectors which can be processed for classification in the SAIN framework. The variables for this data set are 14 channels of temporal EEG data channels, located at places of interest on the human scalp.

The signals measured over the same time period are the EEG channels, fMRI voxels, ECG electrodes, seismic sensory signals, financial time series, gene expressions, voice, and music frequency bands [8]. Even when the variable (signal) measurements are independent, the signals may have an impact on each other as they represent the same object/person at the same time period. The number N of these signals can vary from just a few for a short time window T (Fig 4) to hundreds and thousands when the time varies from a few milliseconds to minutes, hours, days, etc.

Fig 5 shows an EEG experiment and Fig 6 shows one cardio-vascular disease signal.

Next we present a simple example how this search can be computed for a new record X consisting of only 3 variables/signals (e.g. EEG channels, ECG electrodes) over a short period of 5 time moments and the data base D constituting of only 6 such records which are labelled by an outcome labels 1,2,3 (e.g. diagnosis, prognosis).

In addition to the record X , a weight vector is supplied with the weighted importance of the signals at different time points, e.g. $W = [0.1, 0.2, 0.4, 0.2, 0.1]$, meaning that the most important and informative part of the measurements is at time point 3.

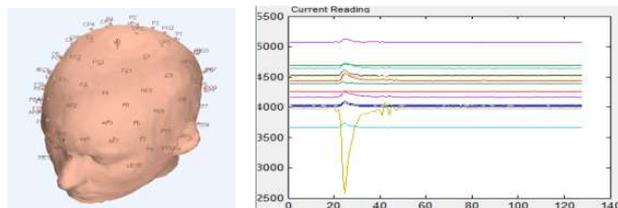


Figure 5. EEG signals taken from EEG electrodes spatially distributed on the scalp are spatio- temporal signals (left figure). Each time series signal from an electrode is measured every 1 millisecond. The figure on the right shows the measurements of 14 EEG electrodes over time of 124 milliseconds. Each signal can be encoded into 124 element vector according to Fig. 4, making altogether 14 such vectors to be processed in the SAIN framework.

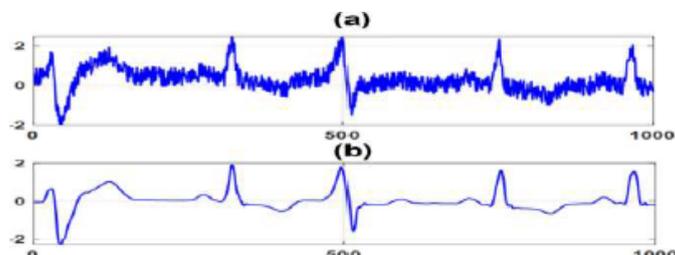


Figure 6. ECG (Electro cardiogram) signals (a- noisy and b- filtered) can be encoded into binary vectors according to the spike encoding methods from Fig. 4. Spike encoding is robust to noise, as any noise below a threshold would not cause the generation of a spike (either positive or negative) and the encoder will act as a filter. The length of this vector will be equal to the number of measurement time points. The vector data can be further processed in the SAIN framework.

The new record $X = [1, 1, -1, 0, 1]$ (signal, EEG channel 1) 0, 1, 1, 1, -1 (signal, EEG channel 2) 1, 1, -1, -1, 0 (signal, EEG channel 3), $W = [0.1, 0.2, 0.4, 0.2, 0.1]$

The database contains records ($Records, R1, R2, R3, R4, R5, LabelsL$) where:

Record	Channel 1	Channel 2	Channel 3	Label
R1	(1, 1, -1, 0, 1)	(0, 1, 1, 1, -1)	(1, 1, -1, -1, 0)	1
R2	(1, 0, -1, 0, 1)	(0, 1, 1, 1, -1)	(1, 0, -1, -1, 1)	1
R3	(1, 1, -1, 0, 1)	(0, -1, 1, 1, -1)	(1, 1, -1, 0, 1)	2
R4	(1, 1, -1, 0, 1)	(0, -1, 1, 0, -1)	(1, 1, -1, 0, 1)	2
R5	(1, 1, -1, 0, 0)	(0, -1, 0, 1, -1)	(1, 1, -1, 1, 1)	3
R6	(1, -1, -1, 0, 1)	(0, -1, 1, 0, -1)	(1, 1, -1, 0, 1)	3

Table 23. Caption

The new record X of EEG-signals will be classified in class 1 as it is closest according to the Euclidian distance class 1 data samples $R1$ and $R2$.

6.3. Predicting longevity in cardiac patients

We utilised a data set in which we applied a binary classification on whether the patient had an event (e.g. death) and further to those that had an event whether this would occur in the near future (within the next 180 days, e.g. approximately six months). The data set contained a set of 150 variables and an outcome, with 295 patients in the first data set and 49 in the second. The data included a mix of variables that could be grouped as follows:

- demographics, risk factors, disease states, medication and deprivation scores,
- echocardiography, cardiac ultrasound measurements,
- advanced ECG measurements,

The other data includes the days until the event occurred of the censor date for the Cox proportional hazard monitoring.

The objectives are to predict an arrhythmic event or death.

Before running the algorithm, the data was normalised, and to account for the data being unbalanced, we utilised smote each time we left one out (ensuring that we did not smote when the true data point was part of the data set). For the event classification data set, the model achieved an accuracy of 79%. This is broken down into classifying no event (198/247, 80%) and an event with (36/49, 73%) accuracy. It is worth noting the confidence of each individual could be explored with a sample of the confidence for classification in Figure 7.

For the second experiment, we normalised the dataset and removed any columns with unknown values. We then applied a genetic algorithm to find the set of features to use for classification. We found a set of 34 variables

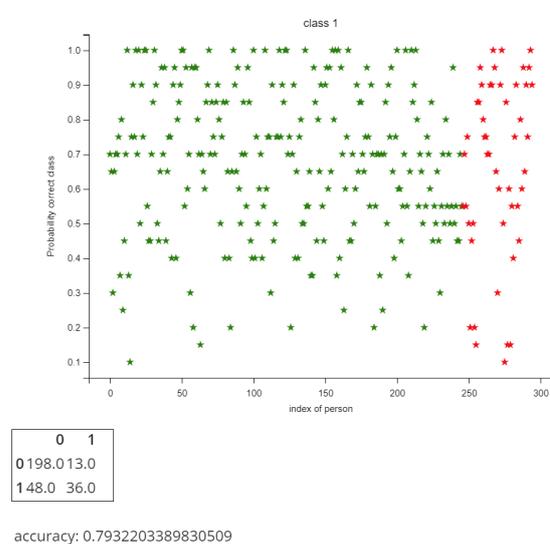


Figure 7. Sample of the classification breakdown and the confusion matrix (class 2 would be utilised when we wish to have an uncertain class, but here we have classified based on probability > 50%).

which would provide an accuracy of 81% with (34/34) for class 0 and (6/15) for class 1. Alternatively, we find that if we apply smote and focus more on the accuracy of class 1, we obtain 69% accuracy, however, more evenly distributed with (24/34) for class zero and (10/15) for class 1.

7. Data and Software Availability

The data has been obtained from the following: UCI Cleveland data available at <https://archive.ics.uci.edu/dataset/45/heart+disease> EEG data available at https://github.com/KEDRI-AUT/NeuCube-Py/tree/master/example_data Access to the software is available on request.

8. Conclusions

The paper presents a new method for search and inference, called here SAIN, for multi-modal data integration and personalised model creation based on these multi-modal data. The model not only evaluates the outcome for a person more accurately than traditional machine learning methods using a single modality data, but it also explains the proposed solution in terms of probability and visual explanation.

The proposed method is implemented as a computer system and applied to several case studies to illustrate its advantages and applicability. The SAIN method described in Section 2 was implemented as a software system.

9. Acknowledgement

We thank Dr. Elena Calude for the contributions to the mathematical model.

References

- AbouHassan, I.; Kasabov, N.K.; Jagtap, V.; Kulkarni, P. Spiking neural networks for predictive and explainable modelling of multimodal streaming data with a case study on financial time series and online news. *Sci. Rep.* **2023**, *13*, 18367.
- Rodrigues, F.; Markou, I.; Pereira, F.C. Combining time-series and textual data for taxi demand prediction in event areas: A deep learning approach. *Information Fusion* **2019**, *49*, 120–129. <https://doi.org/10.1016/j.inffus.2018.07.007>.
- Li, J.; Liu, J.; Zhou, S.; Zhang, Q.; Kasabov, N.K. GeSeNet: A General Semantic-Guided Network With Couple Mask Ensemble for Medical Image Fusion. *IEEE Transactions on Neural Networks and Learning Systems* **2024**, *35*, 16248–16261. <https://doi.org/10.1109/tnnls.2023.3293274>.
- Kasabov, N. Data Analysis and Predictive Systems and Related Methodologies, U.S. Patent 9,002,682 B2, 7 April 2015.
- Doborjeh, M.; Doborjeh, Z.; Merkin, A.; Bahrami, H.; Sumich, A.; Krishnamurthi, R.; Medvedev, O.N.; Crook-Rumsey, M.; Morgan, C.; Kirk, I.; et al. Personalised predictive modelling with brain-inspired spiking neural networks of longitudinal MRI neuroimaging data and the case study of dementia. *Neural Networks* **2021**, *144*, 522–539. <https://doi.org/10.1016/j.neunet.2021.09.013>.
- Budhraja, S.; Singh, B.; Doborjeh, M.; Doborjeh, Z.; Tan, S.; Lai, E.; Goh, W.; Kasabov, N. Mosaic LSM: A Liquid State Machine Approach for Multimodal Longitudinal Data Analysis. In Proceedings of the 2023 International Joint Conference on Neural Networks (IJCNN). IEEE, 2023, pp. 1–8. <https://doi.org/10.1109/ijcnn54540.2023.10191256>.
- Kasabov, N.K. *Evolving connectionist systems*, 2 ed.; Springer: London, England, 2007.
- Kasabov, N.K. *Time-Space, Spiking Neural Networks and Brain-Inspired Artificial Intelligence*; Springer Berlin Heidelberg, 2019. <https://doi.org/10.1007/978-3-662-57715-8>.

9. Santomauro, D.F.e.a. Global prevalence and burden of depressive and anxiety disorders in 204 countries and territories in 2020 due to the COVID-19 pandemic Global prevalence and burden of depressive and anxiety disorders in 204 countries and territories in 2020 due to the COVID-19 pandemic. *The Lancet*, 398, 1700 – 1712. 311
312
313
10. Swaddiwudhipong, N.; Whiteside, D.J.; Hezemans, F.H.; Street, D.; Rowe, J.B.; Rittman, T. Pre-diagnostic cognitive and functional impairment in multiple sporadic neurodegenerative diseases. *bioRxiv*, <https://doi.org/10.1101/2022.04.05.22273468>. 314
2022. 315
316
11. Calude, C.; Calude, E. A metrical method for multicriteria decision making. *St. Cerc. Mat* **1982**, 34, 223–234. 317
12. Calude, C.; Calude, E. On some discrete metrics. *Bulletin mathématique de la Société des Sciences Mathématiques de la République Socialiste de Roumanie* **1983**, pp. 213–216. 318
319
13. Gleeson, S.; Liao, Y.W.; Dugo, C.; Cave, A.; Zhou, L.; Ayar, Z.; Christiansen, J.; Scott, T.; Dawson, L.; Gavin, A.; et al. ECG-derived spatial QRS-T angle is associated with ICD implantation, mortality and heart failure admissions in patients with LV systolic dysfunction. *PLOS ONE* **2017**, 12, e0171069. <https://doi.org/10.1371/journal.pone.0171069>. 320
321
322
14. Kahramanli, H.; Allahverdi, N. Design of a hybrid system for the diabetes and heart diseases. *Expert systems with applications* **2008**, 35, 82–89. 323
324
15. Kasabov, N.K. *Time-space, spiking neural networks and brain-inspired artificial intelligence*; Vol. 750, Springer, 2019. 325