Personalised Information Modelling Technologies for Personalised Medicine

Yingjie Hu, Nikola Kasabov and Wen Liang

Abstract Personalised modelling offers a new and effective approach for the study in pattern recognition and knowledge discovery, especially for biomedical applications. The created models are more useful and informative for analysing and evaluating an individual data object for a given problem. Such models are also expected to achieve a higher degree of accuracy of prediction of outcome or classification than conventional systems and methodologies. Motivated by the concept of personalised medicine and utilising transductive reasoning [34], personalised modelling was recently proposed as a new method for knowledge discovery in biomedical applications [13]. Personalised modelling aims to create a unique computational diagnostic or prognostic model for an individual. Here we introduce an integrated method for personalised modelling [15, 10] that applies global optimisation of variables (features) and an appropriate size of neighbourhood to create an accurate personalised model for an individual. This method creates an integrated computational system that combines different information processing techniques, applied at different stages of data analysis, e.g. feature selection, classification, discovering the interaction of genes, outcome prediction, personalised profiling and visualisation, etc. It allows for adaptation, monitoring and improvement of an individual's model and leads to improved accuracy and unique personalised profiling that could be used for personalised treatment and personalised drug design.

Yingjie Hu
Knowledge Engineering and Discovery Research Institute, Auckland University of Technology, Auckland, New Zealand, e-mail: rhu@aut.ac.nz

Nikola Kasabov
Knowledge Engineering and Discovery Research Institute, Auckland University of Technology, Auckland, New Zealand, e-mail: nkasabov@aut.ac.nz

Liang Wen
Knowledge Engineering and Discovery Research Institute, Auckland University of Technology, Auckland, New Zealand, e-mail: lliang@aut.ac.nz

1 Introduction

Contemporary medical and other data analysis and decision support systems use predominantly inductive global models for the prediction of a person's risk, or of the likely outcome of a disease for an individual [2, 20]. In such models, features are pre-processed to minimise learning function's error (usually a classification error) in a global way to identify the patterns in large databases. Pre-processing is performed to constrain the features used for training global learning models. In general, global modelling is concerned with deriving a global formula (e.g. a linear regression function, a "black box neural network", or a support vector machine) from a large group of data samples. Once an optimal global model is trained, a set of features (variables) are selected and then applied to the whole problem space (i.e. all samples in the given dataset). Thus, the assumption is made that the global model is able to work properly on any new data sample. In clinical research, therapeutic treatment designed to target a disease is assumed to be suitable for any new patients anywhere at anytime. However, such global modelling based medical treatment systems are not always applicable to the individual patients, as the molecular profiling information is not taken into account. The heterogeneity of diseases (e.g. cancer), means that there is different disease progress and different responses to the treatment, even when the patients have similar remarkably morphologically tumours in the same organ.

Statistic reports from the medical research community have shown that the treatment developed by such global modelling methods are only effective for approximately 70% of people, leaving the rest of patients with no effective treatment[28]. In the cases of aggressive diseases, e.g. cancer, any ineffective treatment of a patient (e.g. either a patient not being treated, or being incorrectly treated), can be the difference between life and death. Thus, more effective approaches are required that are capable of using a patient's unique information, such as protein, gene or metabolite profile to design clinical treatment specific to the individual patient.

1.1 Why Personalised Modelling?

In order to develop an understanding of personalised modelling for medical data analysis and biomedical applications, we must answer the question: *why we need personalised information modelling technologies?* For many common conditions a patient's health outcome is influenced by the complex interplay of genetic, clinical and environmental factors [25]. With the advancement of microarray technologies collecting personalised genetic data on a genome-wide (or genomic) scale has become quicker and cheaper [23, 9]. Such personalised genomic data may include: DNA sequence data (e.g. Single Nucleotide Polymorphisms (SNPs), gene and protein expression data. Many world-wide projects have already collected and published a vast amount of such personalised data. For example, Genome-wide Association Scan (GWAS) projects have so far been published for over 100 human

traits and diseases and many have made data available for thousands of people (`http://www.genome.gov/gwastudies`).

The advance of molecular profiling technologies, including microarray messenger ribonucleic acid (mRNA) gene expression data, proteomic profiling, and metabolomic information make it possible to develop "personalised medicine" based on new molecular testing and traditional clinical information for treating individual patient. According to the United States Congress, the definition of *personalised medicine* is given as "the application of genomic and molecular data to better target the delivery of health care, facilitate the discovery and clinical testing of new products, and help determine a person's predisposition to a particular disease or condition" [27]. The personalised medicine is expected to focus on the factors affecting each individual patient and for helping fight chronic diseases. More importantly, it could allow the development of medical treatment tailored to an individual's needs.

Motivated by the concept of personalised medicine and utilising transductive reasoning [34], personalised modelling was recently proposed as a new method for knowledge discovery in biomedical applications. For the purpose of developing medical decision support systems, it would be particularly useful to use the information from a data sample related to a particular patient (e.g. blood sample, tissue, clinical data and/or DNA) and tailor a medical treatment specifically for her/him. This information can also be potentially useful for developing effective treatments for another part of the patient population.

In a broader sense, personalised modelling offers a new and effective approach for the study in pattern recognition and knowledge discovery. The created models are more useful and informative for analysing and evaluating an individual data object for a given problem. Such models are also expected to achieve a higher degree of accuracy of outcome prediction or classification than conventional systems and methodologies [13]. In fact, being able to accurately predict an individual's disease risk or drug response and using such information for personalised treatment is a major goal of clinical medicine in the 21st century [11].

Personalised modelling has been reported as an efficient solution for clinical decision making systems [32], because its focus is not simply on the global problem space, but on the individual sample. For a new data vector, the whole (global) space usually contains much noise information that presents the learning algorithm working properly on this new data, though the same information might be valuable for other data samples. With personalised modelling, the noise (or redundant) information can be excluded within the local problem space that is only created for the observed data sample. This characteristic of personalised modelling makes it a more appropriate method for discovering more precise information specifically for the individual data sample than conventional models and systems.

1.2 Inductive vs. Transductive Reasoning

Inductive and transductive inference are two prevalent approaches used in the development of the learning models and systems in artificial intelligence. The original theory of inductive inference was proposed by Solomonoff [29, 30] in early 1960s and was used for predicting the new data based on observations of a series of given data. In the context of knowledge discovery, the inductive reasoning approach is concerned with the construction of a functional model based on the observations, e.g., predicting the next event (or data) based upon a series of historical events (or data) [4, 20]. Many of the statistical learning methods, such as, SVM, Multi Layer Perceptron (MLP) and neural network models have been implemented and tested on inductive reasoning problems.

Inductive inference approach is widely used in the development of models and systems for data analysis and pattern discovery in computer science and engineering. This approach creates the models based upon known historical data vectors and applicable to the whole problem space. However, the inductive learning and inference approach is only efficient when the whole problem space (global space) is searched for the solution of a new data vector. Inductive models generally neglect any information related to the particular new data sample, which raises an issue about the suitability of a global model for analysing new input data.

In contrast to inductive learning methods, transductive inference introduced by Vapnik [34] is a method that creates a model to test a specific data vector (a testing data vector) based on the observation of a specific group of data vectors (training data). The models and methods created from transductive reasoning focus on a single point of the space (the new data vector), rather than on the whole problem space. Transductive inference systems emphasise the importance of the utilisation of the additional information related to the new data point, which brings more relevant information to suit the analysis of the new data. Within the same given problem space, transductive inference methods may create different models, each of them specific for testing every new data vector.

Transductive inference systems have been so far applied to a variety of classification problems, such as heart disease diagnostics [36], promoter recognition in bioinformatics [16], microarray gene expression data classification [35]. Other examples using transductive reasoning systems include: evaluating the predicting reliability in regression models [5], providing additional reliability measurement for medical diagnosis [19], transductive SVM for gene expression data analysis [26] and a transductive inference based radial basis function (TWRBF) method for medical decision support system and time series prediction [31]. Most of these experimental results have shown that transductive inference systems outperform inductive inference systems, due to the former's ability to exploit the structural information of unknown data.

Some more sophisticated transductive inference approaches have been developed including: Transductive Neural Fuzzy Inference System with Weighted Data Normalization - TWNFI [32] and Transductive RBF Neural Network with Weighted Data Normalization - TWRBF [31]. These methods create a learning model based

on the neighbourhood of new data vector, and then use the trained model to calculate the output.

Transductive inference approach seems to be more appropriate to build learning models for clinical and medical applications, where the focus is not simply on the model, but on the individual patient's condition. Complex problems may require an individual or a local model that best fits a new data vector, e.g. a patient to be clinically treated; or a future time moment for a time-series data prediction, rather than a global model that does not take into account any specific information from the object data [32].

2 Global, Local and Personalised Modelling Approaches

Global, local and personalised modelling are currently the three main techniques for modelling and pattern discovery in the machine learning area. These three types of modelling techniques are derived from inductive and transductive inference and are the most commonly used learning techniques for building the models and systems for data analysis and patter recognition [13, 14]. This section will investigate these three techniques for data analysis and model design.

- **Global modelling** creates a model from the data that covers the entire problem space. The model is represented by a single function, e.g. a regression function, a radial basis function (RBF) , a MLP neural network, SVM, etc.
- **Local modelling** builds a set of local models from data, where each model represents a sub-space (e.g. a cluster) of the whole problem space. These models can be a set of rules or a set of local regressions, etc.
- **Personalised modelling** uses transductive reasoning to create a specific model for each single data point (e.g. a data vector, a patient record) within a localised problem space.

To explain the concepts of global, local and personalised modelling, we hereby present a comparative study in which each type of model will be applied to a benchmark gene expression dataset, namely colon cancer data [1] for cancer classification. This comparative study applies several popular algorithms for modelling development and investigates the performance using three modelling techniques on a gene expression data. The data used in the comparative experiment originates from Colon cancer data that consists of 62 samples of colon epithelial cells from colon cancer patients. 40 samples are collected from tumors and labeled as "diseased", and 22 samples are labeled as "normal" collected from a healthy part of the colon of the same patient. Each sample is represented by $2,000$ genes selected out of total $6,500$ genes based on the confidence in measured expression levels. Since the goal of this experiment is to demonstrate the difference of classification performance generated by three modelling techniques, we simply select 15 out of 2,000 genes by a signal-noise-to-ratio (*SNR*) method according to their statical scores for the purpose of reducing computational cost. SNR is a simple statistical algorithm and widely

adopted to filter features. Let $\bar{x}_i$ and $\bar{y}_i$ denote the mean values of the i^{th} gene in the samples in class 1 and class 2 respectively, σ_{xi} and σ_{yi} are the corresponding standard deviations. Then each feature's *SNR* score can be calcualated as follows:

$$SNR(i) = \frac{|\bar{x}_i - \bar{y}_i|}{\sigma_{xi} + \sigma_{yi}}, i = 1, 2, \cdots, m \tag{1}$$

where m is the number of features in the given dataset. The greater the SNR value, the more informative the feature. Therefore, the preprocessed subset used in the experiment presented here constitutes 62 samples. Each sample contains 15 top features(genes) selected based on their statistical SNR ranking scores. The subset is denoted as $D_{colon15}$.

As our interest for this experiment is mainly in the comparison of the classification performance obtained from three different modelling techniques, we applied a simple validation approach (*Hold-out* method) to the classification on data $D_{colon15}$: the given data is split into training and testing data with a specified ratio, i.e. 70% of samples are used for training and the remaining 30% for testing.

2.0.1 Global Modelling

Linear and logistic regression modelling is one of the most popular global modelling techniques. They have been implemented in a variety of global methods for modelling gene expression data [7], and for modelling gene regulatory networks [6].

Multiple linear regression (MLR) is a global modelling technique that is among the simplest of all statistical learning algorithms. MLR analysis is a multivariate statistical technique that examines the linear correlations between a single dependent variable and two or more independent variables. For multiple linear regression analysis, the independent variable X is described by an m-dimensional vector: $X = \{x_1, x_2, \cdots, x_m\}$, and a MLR model can be formulated as:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_m x_{im} \varepsilon_i, \quad i = \{1, 2, \cdots, n\} \tag{2}$$

where:

- β is an m-dimensional parameter vector called effects or (regression coefficients);
- ε is the *"residual"* representing the deviations of the observed values y from their means $\bar{y}$, which are normally distributed with mean 0 and variance;
- n is the number of observations.

For the purpose of investigating the global modelling for classification problems, an MLR based approach is applied to the subset of colon cancer gene expression data ($D_{colon15}$). A global MLR-based classifier is created from the training data (70%) analysis, which is given as:

$$\begin{aligned}\mathscr{Y} = {} & 0.1997 + 0.1354 * \mathbf{X_1} + 0.70507 * X_2 + -0.42572 * X_3 - 0.19511 * X_4 \\ & + 0.0943 * \mathbf{X_5} - 0.6967 * \mathbf{X_6} - 1.0139 * X_7 + 0.9246 * \mathbf{X_8} \\ & + 0.1550 * \mathbf{X_9} + 0.6190 * X_{10} + 0.1793 * X_{11} + 1.123 * \mathbf{X_{12}} \\ & - 0.1615 * X_{13} - 0.4789 * X_{14} - 0.4910 * X_{15}\end{aligned} \tag{3}$$

where $\mathscr{Y}$ is an MLR model to predict the new input data vector (here is to predict whether a patient sample is "diseased" or "normal"), and $X_i, i = 1, 2, \ldots, 15$ denotes each variable (feature).

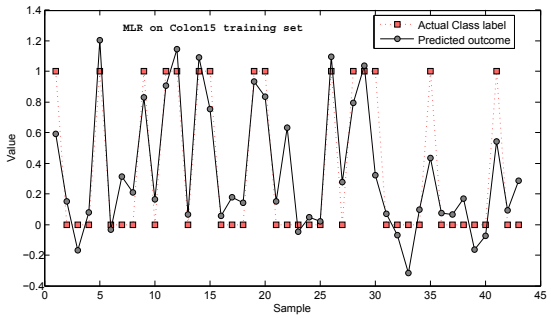
Function 3 constitutes a global model to be used for evaluating the output for any new data vector in the 15-dimensional space regardless of where it is located. This global model extracts a 'big' picture for the whole problem space, but lacks an individual profile [13]. It indicates to certain degree the genes' importance: X_6, X_8 and X_{12} show strong correlation to the corresponding output, while X_5, X_1, X_9 are less important in terms of outcome prediction.

Figure 1 shows the prediction result from the global multi-linear regression model over colon data with selected 15 genes. The results plotted in Figure 1 (a) and (b) demonstrate the inconsistent issue in microarray gene expression data analysis: the accuracy from testing data is significantly lower than that from training data - 95.3% vs. 73.7%, when the threshold of disease distinction is set to 0.5.

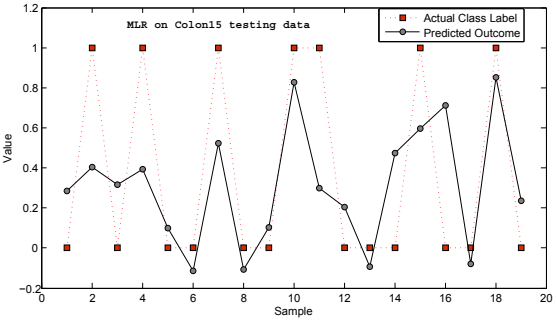
2.1 Local Modelling

Unlike global models, local models are created to evaluate the output function especially within a sub-space of the entire problem space (e.g. a cluster of data). Multiple local models can consist of the complete model across the entire problem space. Local models are usually based on clustering techniques. A cluster is a group of similar data samples, where similarity is measured predominantly as Euclidean distance in an orthogonal problem space. Clustering techniques can be found in the literature: classical k-means [21], Self-Organising Maps (SOM) [18, 8], fuzzy c-means clustering [3], hierarchical clustering for cancer data analysis [1], a simulated annealing procedure based clustering algorithm for finding globally optimal solution for gene expression data [22]. Fuzzy clustering is a popular algorithm used to implement local modelling for machine learning problems. The basic idea behind it is that one sample may belong to several clusters to a certain membership degree, and the sum of membership degree should be one.

Local learning models adapt to new data and discover local information and knowledge, that provide a better explanation for individual cases. However, these local modeling methods do not select specific subsets of features and precise neighbourhood of samples for individual samples that require a personalised modelling in the medical area. Evolving classification function (ECF) [12, 17] is a representative technique for local modelling. The classification result from ECF local model over dataset $D_{colon15}$ is shown in Figure 2(a) and 2(b). The classification accuracy from ECF model on the training set (70% of the whole data) appeared excellent - 100%



(a) The classification result using a global MLR model on $D_{colon15}$ training set (the training accuracy is 95.3%);



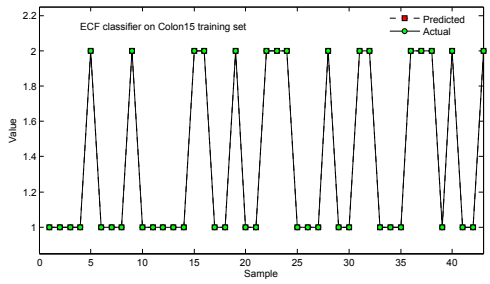
(b) The classification result using a global MLR model on $D_{colon15}$ testing set (the testing accuracy is 73.7%).

Fig. 1 An example of global modelling: the classification results from a multi-linear regression model(MLR) over colon cancer gene data, where x axis is the sample index, y axis represents the value of the actual class label and predicted outcome for each sample. The red square points represent the actual class labels of the samples, while the black circle points present the predicted outcome.

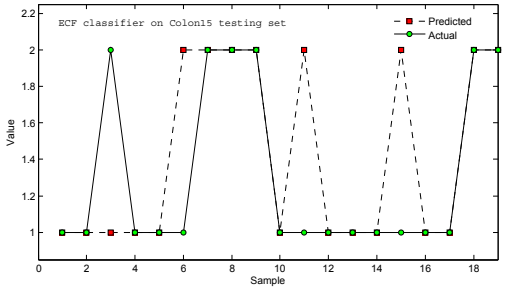
accurate, but the classification result from the testing set (30%) is only 78.95% (15 out of 19 samples are correctly classified). It seems that local modelling might not be an effective approach for analysing this particular gene expression dataset. Moreover, it is difficult to optimise the parameters during the learning process.

2.2 Personalised Modelling

The philosophy behind the proposed personalised modelling is the realisation that every person is different, and preferably each individual should have their own per-



(a) A local modelling: the outcomes from ECF model on the training set of colon cancer data (70%), the training accuracy is 100%.



(b) A local modelling: the outcomes from ECF model on the testing set of colon cancer data (30%), the testing accuracy is 79.0%.

Fig. 2 An example of local modelling: the experimental results from a local modelling method (ECF) on the training and testing set from data ($D_{colon15}$), respectively. Black solid line represents the actual label of the sample, while red dotted line is the predicted outcome.

sonalised models and tailored treatment. In the context of medical research, it has become possible to utilise individual data for a person with the advance of technology, e.g., DNA, RNA, protein expression, clinical tests, inheritance, foods and drugs intake, diseases. Such data is more readily obtainable nowadays, and is easily measurable and storable in electronic data repositories with less cost.

In contrast to global and local modelling, personalised modelling creates a model for every new input data vector based on the samples that are closest to the new data vector in the given dataset. Figure 3 gives an example for personalised problem spaces. With a transductive approach, each individual data vector that represents a patient in any given medical area obtains a customised, local model that best fits the new data. This is contrary to using a global modeling approach where new data is matched to a model (function) averaged for the entire dataset. A global model may fail to take into account the specific information particular to individual data

samples. Moreover, there are no efficient methods for identifying important features that assist complex disease classification, e.g. which genes, SNPs, proteins and other clinical information contribute to the disease diagnosis. Hence, a transductive approach seems to be a step in the right direction when looking to devise personalised modelling useful for analysing individual data sample, e.g. disease diagnosis, drug design, etc.

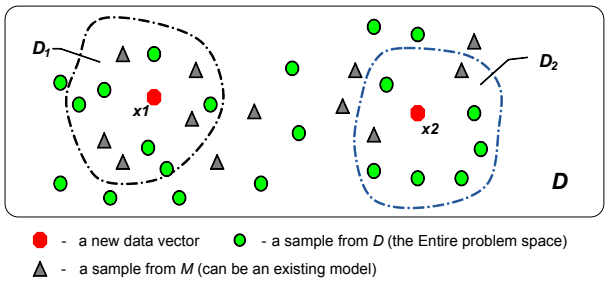


Fig. 3 An example of personalised space, where x_1 and x_2 represent two new input vectors, D is the entire (global) problem space, D_1 and D_2 denote the two personalised spaces for x_1 and x_2, respectively.

A personalised modelling framework (PMF) is initially designed for medical data analysis and knowledge discovery. However, PMF can be extended for solving various types of data analysis problems that require personalised modelling. PMF can be briefly described as follows:

1. Apply feature selection on the object data D (the global problem space) to identify which features are important to a new input vector x_v. The selected features are grouped into a candidate gene pool;
2. Select K_v nearest samples for x_v from D to form a local (personalised) problem space D_{pers};
3. Create a personalised model candidate M_x specifically for x_v, which includes a learning function (usually a classifier or a clustering function) denoted by f;
4. Evaluate the candidate feature subset S by a learning function f based on their performance within the personalised problem space D_{pers};
5. Optimising model M_x through an evolving approach until termination conditions are met. The output is the optimal or near-optimal solution to vector x_v. The solution includes an optimal personalised model M_x^* and a selected feature subset S^*;
6. Use the model M_x^* to test the new vector x_v and calculate the outcome y_v;
7. Create a personalised profile for the input vector x_v, visualize the outcome with the selected important features S^*, and provide an improvement scenario for data vector x_v for a given problem if it is possible.

An outline of PMF is depicted in Figure 4.

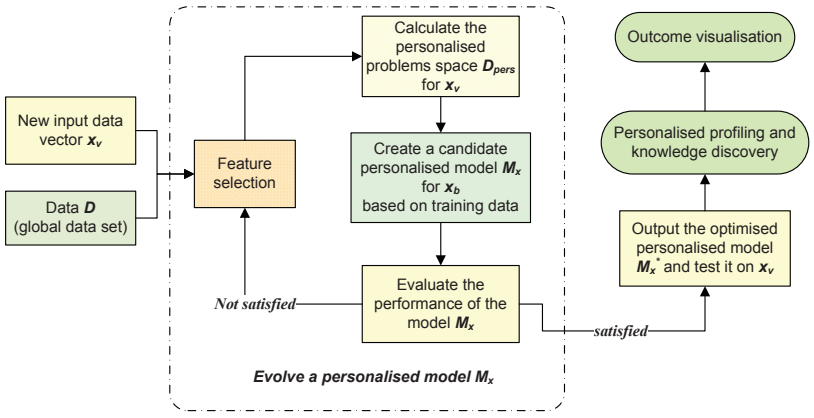


Fig. 4 A PMF for data analysis and knowledge discovery.

KNN method is probably the simplest techniques to use for personalised modelling. In a KNN model, the K nearest samples for every new sample x_i are derived from the given dataset through a distance measurement (usually Euclidean distance), and the class label for the new sample x_i is assigned based on a voting scheme [24]. The classical KNN method calculates the output value y_i according to the determination made by the majority vote of its neighbours, i.e. the new data vector is assigned to the class most common amongst its k nearest neighbours.

KNN algorithm is one of the most popular algorithms in machine learning, because it is simple to implement and works fast and effectively on many machine learning problems. However, the parameter selection is a critical factor impacting on KNN classifier's performance, e.g., the choice of value for K. In general, more nearest neighbours (K) used in KNN method can reduce the effect of noise over the classification, but would make the boundaries between classes less distinct. If too few neighbours are selected, there can be insufficient information for decision making. Also, the performance of the KNN algorithm can be severely degraded by the presence of noisy features which is a very common issue in biomedical data.

2.2.1 Weighted Nearest Neighbour Algorithms for Personalised Modelling: WKNN & WWKNN

In a weighted distance KNN algorithm (WKNN) , the output y_i is calculated not only based on the output values (e.g. class label) y_j, but is also dependent on the weight w_j measured by the distance between the nearest neighbours and the new data sample x_i:

$$y_i = \frac{\sum_{j=1}^{K_i} w_j \cdot y_j}{\sum_{j=1}^{K_i} w_j} \tag{4}$$

where:

- y_i is the predicted output for the new vector x_i;
- y_j is the class label of each sample in the neighbourhood of x_i.
- K_i is the number of K nearest samples to x_i;
- w_j is the weight value calculated based on the distance from the new input vector x_j to its K nearest neighbours.

The weight w_j can be calculated as follows:

$$w_j = \frac{\max(d) - (d_j - \min(d))}{\max(d)}, \qquad j = 1, \cdots, K \tag{5}$$

where:

- the value of weights w_j ranges from $\frac{\min(d)}{\max(d)}$ to 1;
- $d = [d_1, d_2, \cdots, d_K]$ denotes the distance vector between the new input data d_i and the its K nearest neighbouring samples;
- $\max(d)$ and $\min(d)$ are the maximum and minimum values for vector d.

The distance vector d is computed as:

$$d_j = \sqrt{\sum_{l=1}^{m} (x_{i,l} - x_{j,l})^2}, \qquad j = 1, \cdots, K \tag{6}$$

where m is the number of variables (features) representing the new input vector x_i within the problem space; $x_{i,l}$ and $x_{j,l}$ are the l^{th} variable values corresponding to the data vector x_i and x_j, respectively.

The output from a WKNN classifier for the new input vector x_i is a "*personalised probability*" that indicates the probability of vector x_i belonging to a given class. For a two-class classification problem, a WKNN classifier requires a threshold θ to determine the class label of x_i, i.e., if the output (*personalised probability*) is less than the threshold θ, then x_i is classified into the group with "small" class label, otherwise into the group with "big" class label. For example, in a case of a two-class problem, the output from WKNN model for *sample#1* of data $D_{colon15}$ is *0.1444*, so that this testing sample is classified into class **1** ("small" class label) when the threshold θ is set to *0.5*.

Weighted distance and weighted variables K-nearest neighbours (WWKNN) is a personalised modelling algorithm introduced by Kasabov [13]. The main idea behind WWKNN algorithm is: the K nearest neighbour vectors are weighted based on their distance to the new data vector x_i, and also the contribution of each variable is weighted according to their importance within the local area where the new vector

belongs [13]. In WWKNN, the assumption is made that the different variables have different importance to classifying samples into different classes when the variables are ranked in terms of their discriminative power of class samples over the whole *m-dimensional* space. Therefore, it will be more likely that the variables have different ranking scores if the discriminative power of the same variables is measured for a sub-space (localised space) of the entire problem space. The calculation of Euclidean distance d_j between a new vector x_i and a neighbour x_j is mathematically formulated by:

$$d_j = \sqrt{\sum_{l=1}^{K} c_{i,l}(x_{i,l} - x_{j,l})^2}, \qquad j = 1, \cdots, K \tag{7}$$

where: $c_{i,l}$ is the coefficient weighing x_l in relation with its neighbourhood of x_i, and K is the number of the nearest neighbours. The coefficient $c_{i,l}$ can be calculated by a SNR function that ranks variables across all vectors in the neighbourhood set $D_{nbr}(x_i)$:

$$\begin{aligned} c_{i,l} &= \{c_{i,1}, c_{i,2}, \cdots, c_{i,K}\} \\ c_{i,l} &= \frac{|\bar{x_l}^{class1} - \bar{x_l}^{class2}|}{\sigma_l^{class1} + \sigma_l^{class2}} \end{aligned} \tag{8}$$

where:

- $\bar{x_l}^{classi}$, $i = \{1,2\}$: the mean value of the l^{th} feature belonging to class i across the neighbourhood $D_{nbr}(x_i)$ of x_j;
- σ_l^{classi}, $i = \{1,2\}$: the standard deviation of l^{th} feature belonging to class i across the neighbourhood $D_{nbr}(x_i)$ of x_j.

Comparing to a conventional KNN algorithm, the contribution of WWKNN lies in the new distance measurement: all variables are weighted according to their importance as discriminating factors in the neighbourhood area (personalised subspace), which might provide more precise information for classification or prediction of the new data vector.

The experimental results from the classification of $D_{colon15}$ data using WKNN and WWKNN are illustrated in Figure 5. It shows that WWKNN produced better predicting result for colon cancer data classification, as the predicted outcome from WWKNN Both WKNN and WWKNN can create an outcome vector indicating the testing sample's probability of being diseased, which provides the important information for clinical decision making.

3 A Methodology to Build a Personalised Modelling System

We introduce a methodology for using the proposed PMF to build a personalised modelling system (PMS) to create the personalised model for each new input data sample based on its unique information. Given a dataset D pertaining to a bioinformatics problem, $D = \{x_{ij}, y_i, i = 1, \cdots, n, j = 1, \cdots, m\}$, where x is a data sample,

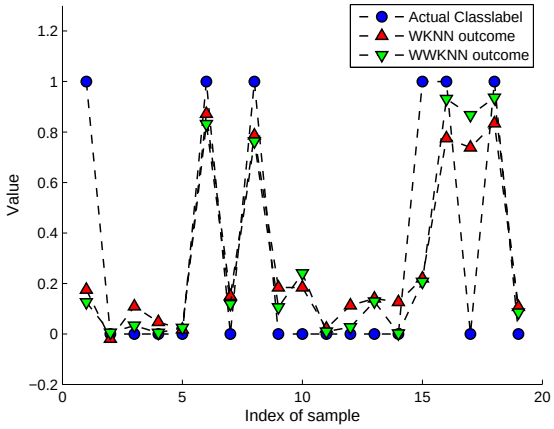


Fig. 5 The experimental results computed by two personalised models - WKNN and WWKNN on the colon cancer $D_{colon15}$ testing set (it contains 19 samples). x axis is the sample index and y axis shows value of the predicted outcome. $K = 15$ and the classification threshold is 0.5. Both of the models yielded 84.2% classification accuracy.

y is the responding outcome, n is the number of samples, m denotes the number of features (variables). The proposed method aims to optimise a model M_x suitable for analysing data, specific to every new input data vector x_v, e.g. to calculate y_v - the outcome of x_v. Data x_v contains a number of features that are related to the same scenario as the data samples in the global data D.

In order to obtain the optimal or near optimal personalised model M_x^* specifically for a new data sample x_v, the proposed method aims to find the solutions to the following objectives:

1. Determine how many and which features (variables) S are most suitable for building the model M_x^* that is able to successfully predict the outcome for the new data vector x_v;
2. Determine the appropriate number K_v for the neighbourhood of x_v to form a personalised problem space D_{pers};
3. Identify K_v samples from the global data set D which have the pattern most similar to the data x_v, and use these K_v samples to form the neighbourhood (a personalised problem space D_{pers});
4. Calculate the importance of selected features S within the personalised problem space D_{pers}), based on their contribution to the outcome prediction of the data vectors in D_{pers}. Compute a weight vector w_v for all selected features S;
5. Create the optimal personalised model M_x^* with the optimised parameters obtained in Steps 1∼4;
6. Validate the obtained model M_x^* by calculating the outcome y_v for the new data x_v;
7. Profile the new input data x_v within its neighbourhood D_{pers} using the most important features associated with a desired outcome;
8. If possible, provide the scenarios for improving the outcome for the new data vector x_v, which can be helpful for clinical use.

This is a method for determining a profile of a subject (new input vector x_v) using an optimal personalised model M_x^*, and for recommending the possible changes to the profile in relation to a scenario of interest in order to improve the outcome for x_v. The method comprises the following steps:

- Create a personalised profile for a new data vector x_v;
- Compare each important feature of input data vector x_v to the average value of important features of samples having the desired outcome;
- Determine which important features of input vector x_v can be altered in order to improve the outcome.

Principally, the decision of which variables should be changed will be based on the observation of the weight vector W_x of features (i.e. the contribution of the features to the classification). The term "*personalised profile*" used here refers to an input vector x_v and to its predicted outcome and related information, such as the size of its neighbourhood, its most important features specifically, etc.

Within the scope of PMS, the proposed method for building an optimal model M_x requires the following functional modules:

- A module for selecting most relevant V_v features (variables) S^* and ranking their weighter w_x by importance for x_v;
- the module for the selection of a number K_v of neighbouring samples of x_v and for the selection of neighbouring samples D_{pers};
- A module for creating a prediction model M_x, defined by the a set of parameters P_v, such as K_v, V_v, D_{pers} which were derived in the previous modules;
- A module for calculating the final output y_v responding to the new data x_v
- A module for the creation of personalised profile and the design of scenarios for potential improvement.

4 An Integrated Optimisation Method for Implementing a PMS

There has been very few implementations for PMSs using the computational intelligence for solving complex biomedical applications. In this section, we introduce an integrated method that has been recently developed for PMS implementations. The integrated method for personalised modelling (IMPM) [15] is developed based on the methodology described in Sec 3. For every new individual sample (new data vector), all aspects of their personalised model (variables, neighbouring samples, type of models and model parameters), are combined together to be optimised based on the accuracy of the outcome achieved within the local neighbourhood of the sample. Next, a personalised model and personalised profile are derived that use the selected variables and the neighbouring samples with known outcomes. The sample's profile is compared with average profiles of the other outcome classes in the neighbourhood (e.g. positive outcome, or negative outcome of disease or treatment). The difference between the points and average profiles based on important variables that may need to be modified through treatment and can be utilised in personalised drug design.

Algorithm 1 The algorithm of IMPM

1: Data pre-processing stage:
Include data collection, storage, update, etc.
2: Feature selection:
Identify a subset of features (variables) V_x relevant to the new data sample x_v from all features V;
3: Local problem space creation:
Select a number K_x of samples from the global dataset D to create a neighbourhood D_x. D_x consists of a group of similar samples to x with the features from V_x;
4: Evaluate the V_x features within the local neighbourhood D_x in terms of their contribution and obtaining a weight vector W_x;
5: Training model optimisation:
Optimise a local prognostic/classification model M_x that has a model-parameter set P_x, a variable set V_x and local training data set D_x;
6: Testing the new data sample:
Apply the optimised personalised model $M_x^*(P_x, V_x, D_x)$ on the new data sample x and output the prediction result;
7: Profiling:
Generate a functional profile F_x for the new data sample x using the selected set V_x of variables, along with the average profiles of the samples from D_x that belong to different outcome classes, e.g. F_i and F_j.
8: Perform a comparative analysis between F_x, F_i and F_j to define what variables from V_x are the most important for the person x that make him/her very differential from the desired class. These variables may be used to define a personalised course of treatment, such as personalised medicine.

4.1 A detailed description of the IMPM

The IMPM consists different functional modules and is summarised in Algorithm 1. Steps **2**-**5** is an iterative learning (training) process to optimise the local model M_x. The optimisation continues until the termination criteria are reached, e.g. the maximum number of iterations is reached or a desired local accuracy of the model for a local data set D_x is achieved. The optimisation of the parameters of the personalised model V_x, K_x and D_x is global and is achieved through multiple runs of cEAP that has been described in the previous section. The resulting competing personalised models for x form a population of such models that are evaluated over iterations (generations) using a fitness criterion - the best accuracy of outcome prognosis for the local neighbourhood of new testing sample x. All variables and parameters of the personalised model form to an integrated single 'chromosome' (refer to Figure 6) where variable values are optimised together as a global optimisation.

Initially, the assumption is made that all feature (variable) set V have equal absolute and relative importance for a new sample x in relation to predicting its unknown output y:

$$w_{v1} = w_{v2} =, \cdots, = w_{vq} = 1 \tag{9}$$

and

$$w_{v1,norm} = w_{v2,norm} =, \cdots, = w_{vq,norm} = 1/q \tag{10}$$

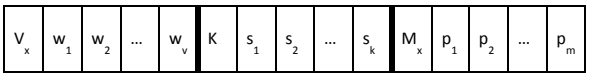


Fig. 6 A chromosome used in IMPM for the global optimisation of the parameters ('genes'): the variables V_x to be selected; their corresponding weights W_x; number K of nearest neighbours to x_v; the neighbourhood D_x with selected K samples $s_1 - s_K$; a local prognostic model M_x (e.g. classifier); a parameter set P_m for the M_x.

The initial numbers for the variables V_x and K_x may be determined in a variety of different ways without departing from the scope of the method. For example V_x and K_x may be initially determined by an assessment of the global dataset in terms of size and/or distribution of the data. The values of these parameters may be constrained according to the available data. For example, $V_{x\_min} = 3$ (minimum three variables used in a personalised model) and $V_{x\_max} < K_x$ (the maximum variables used in a personalised model should be smaller than the number of samples in the neighbourhood D_x of x), usually $V_{x\_max} < 20$. The initial set of variables may include expert knowledge, i.e. variables which are referenced in the literature as highly correlated to the outcome of the problem (disease) in a general sense (over the whole population). Such variables for example are the BRCA genes in the study for breast cancer prediction [33]. For an individual patient the BRCA genes may interact with some other genes, which interaction will be specific for the person or a group of people and is likely to be discovered through local or/and personalised modelling only [13].

IMPM has a major advantage over global and local modelling methods, as its modelling process can start with all relevant variables available for a person, rather than with a pre-fixed set of variables in a global model. Such global models may be statistically representative for the whole population, but not necessarily representative for a single person in terms of optimal model and best profiling and prognosis for this person.

Selecting the initial number K_x of neighbouring samples and the minimum and the maximum numbers K_{x_min} and K_{x_max} can also depend on the data available and on the problem in hand. A general requirement is that $K_{x\_min} > V_x$, and, $K_{x\_max} < cN$, where c is a ratio, e.g. 0.5, and N is the number of samples in the neighbourhood D_x of x. Several formulas have been already suggested and experimented [34], e.g.:

- K_{x_min} equals the number of samples that belong to the class with a smaller number of samples when the data is imbalanced (one class has many more samples, e.g. 90%, than the another class) and the available data set D is of small or medium size (e.g., several tens or several hundreds of samples);
- $K_{x\_min} = \sqrt{N}$, where N is the total number of samples in the data set D;

At subsequent iterations of the method, the parameters V_x and K_x along with all other parameters are optimised via an optimisation procedure, usually an evolutionary based algorithm, such as cEAP [10] that optimises all or part of parameters form the 'chromosome' in Figure 6.

The closest K_x neighbouring vectors to x from D are selected to form a new data set D_x. A local weighted variable distance measure is used to weigh the importance of each variable V_l ($l = 1,2,\cdots,q$) to the accuracy of the model outcome calculation for all data samples in the neighbourhood D_x. For example, the distance between x and z from D_x is measured as a local weighted variable distance:

$$d_{x,z} = \frac{\sqrt{\sum_{l=1}^{q}(1-w_{l,norm})(x_l - z_l)^2}}{q} \tag{11}$$

where: w_l is the weight assigned to the variable V_l and its normalised value is calculated as:

$$w_{l,norm} = \frac{w_l}{\sum_{i=1}^{q} w_i} \tag{12}$$

Here the distance between a cluster centre (in our case it is the vector x) and cluster members (data samples from D_x) is calculated not only based on the geometrical distance, as it is in the traditional nearest neighbour methods, but on the relative variable importance weight vector W_x in the neighbourhood D_x as suggested in [13]. After a subset D_x of K_x data samples are selected based on the variables from V_x, the variables are ranked in a descending order of their importance for prediction of the output y of the input vector x and a weighting vector W_x is obtained. Through an iterative optimisation procedure the number of the variables V_x to be used for an optimised personalised model M_x will be reduced, and only the most appropriate variables that lead to the best local prediction accuracy forM_x will be selected. For weighting W_x (i.e. ranking) of the V_x variables, alternative algorithms can be used, such as t-test, Signal-to-Noise ratio (SNR), etc.

In the SNR algorithm, W_x are calculated as normalised coefficients and the variables are sorted in descending order: $V_1, V_2, \cdots, V_v$, where: $w_1 >= w_2 >= \cdots >= w_v$, using the Equation 8. This method is very fast, but evaluates the importance of the variables in the neighbourhood D_x one by one and does not take into account a possible interaction between the variables, which might affect the model output.

A learning model, usually a classification or prediction model is applied to the neighbourhood D_x of K_x data samples to derive a personalised model M_x using the already defined variables V_x, variable weights W_x and a model parameter set P_x. A variety of classification or prediction models can be used here such as: MLR, SVM, KNN, WKNN, WWKNN [13], TWNFI [32], etc. The outcome produced by the weighted KNN (WKNN) classifier for the new sample is calculated based on the weighted outcomes of the individuals in the neighbourhood according to their distance to the new sample. In the WWKNN model [13] variables are ranked and weighted according to their importance for separating the samples of different classes in the neighbourhood area in addition to the weighting according to the distance as in WKNN. In the TWNFI model - transductive, weighted neuro-fuzzy inference system [32], the number of variables in all personalised models is fixed, but the neighbouring samples used to train the personalised neuro-fuzzy classifica-

tion model are selected based on the variable weighted distance to the new sample the same as it is in the WWKNN.

The vector distance $d = [d_1, d_2, \cdots, d_K]$ is defined as the distances between the new input vector x and the nearest samples (x_j, y_j) for $j = 1$ to K_x; $max(d)$ and $min(d)$ are the maximum and minimum values in d respectively. Euclidean distance d_j between vector x and a neighbouring one x_j is calculated as:

$$d_j = \sqrt{\sum_{l=1}^{V} (1 - w_l)(x_l - x_{jl})^2} \tag{13}$$

where: w_l is the coefficient weighing variable x_l in the neighbourhood D_x of x (e.g. w_l can be calculated by a SNR algorithm, refer to Eq.**??**.

When using the TWNFI classification or prediction model [32], the output y for the input vector x is calculated as follows:

$$y = \frac{\sum_{l=1}^{m} \frac{n_l}{\delta_l^2} \prod_{j=1}^{P} \alpha_{lj} \cdot \exp[-\frac{w_j^2 (x_{ij} - m_{lj})^2}{2\sigma_{lj}^2}]}{\sum_{l=1}^{m} \frac{1}{\delta_l^2} \prod_{j=1}^{P} \alpha_{lj} \cdot \exp[-\frac{w_j^2 (x_{ij} - m_{lj})^2}{2\sigma_{lj}^2}]} \tag{14}$$

where: m is the number of the closest clusters to the new input vector x; each cluster l is defined as a Gaussian function G_l in a V_x dimensional space with a mean value m_l as a vector and a standard deviation δ_l as a vector too; $x = (x_1, x_2, \cdots, x_v)$; α_l (also a vector across all variables V) is membership degree to which the input vector x belongs to the cluster Gaussian function G_l; n_l is a parameter of each cluster [32].

A local accuracy (local error E_x), that estimates the personalised accuracy of the personalised prognosis (classification) for the data set D_x using model M_x is evaluated. This error is a local one, calculated in the neighbourhood D_x, rather than a global accuracy, that is commonly calculated for the whole problem space D. Different methods can be used for calculating the error, such as: absolute error (AE), root-mean square error (RMSE) and area under the receiving operating characteristic curve (AUC).

We propose another method to calculate local error specific for model optimisation:

$$E_x = \frac{\sum_{j=1}^{K_x} (1 - d_{xj}) \cdot E_j}{K_x} \tag{15}$$

where: d_{xj} is the weighted Euclidean distance between sample x and sample S_j from D_x that takes into account the variable weights W_x (see Eq.11); E_j is the error between what the model M_x calculates for the sample S_j from D_x and what its real output value is.

Based on a weighted distance measured by the above formula, the closer the data sample S_j to x is, the higher its contribution to the error E_x will be. The calculated personalised model M_x accuracy is then formulated as:

$$A_x = 1 - E_x \tag{16}$$

The best accuracy model obtained is stored for the purpose of future improvement and optimisation. The optimisation procedure iteratively returns to all previous procedures (*step2* to *step5*) to select another set of parameter values for the parameter vector (refer to Figure 6) until the termination criteria are reached. The method also optimises parameters P_x of the classification/prediction procedure. The output value y for the new input vector x is then calculated by the optimal model M_x^* Next, a personalised profile F_x for the person can be assessed against possible desired outcomes for the scenario, and the possible ways to achieve an improved outcome can be designed, which is a major novelty of this method. The profile F_x for x is formed as a vector:

$$F_x = \{V_x, W_x, K_x, D_x, M_x, P_x, t\} \tag{17}$$

where the variable t represents the time of the model M_x creation. At a future time $(t + \Delta t)$ the person's input data will change to x^* (due to changes in variables such as age, weight, protein expression values, etc.), or the data samples in the data set D may be updated and new data samples added. A new profile F_x^* derived at time $(t + \Delta t)$ may be different from the current one F_x.

The average profile F_i for every class C_i in the data D_x is a vector containing the average values of each variable of all samples in D_x from class C_i. The importance of each variable (feature) is indicated by its weighting in the weight vector W_x. The weighted distance from the person's profile F_x to the average class profile F_i (for each class i) is defined as:

$$D(F_x, F_i) = \sum_{l=1}^{v} |V_{lx} - V_{li}| \cdot w_l \tag{18}$$

where w_l is the weight of the variable V_l calculated for dataset D_x (see Eq.12).

Assuming that F_d is the desired profile (e.g. normal outcome), the weighted distance $D(F_x, F_d)$ will be calculated as an aggregated indication of how much the person's profile should change to reach the average desired profile F_d:

$$D(F_x, F_d) = \sum_{l=1}^{v} |V_{lx} - V_{ld}| \cdot w_l \tag{19}$$

A scenario for a person's improvement through changes made to variables (features) towards the desired average profile F_d can be produced as a vector of required variable changes, defined as:

$$\Delta F_{x,d} = \Delta V_{lx,d} \mid l = 1, \cdots, v \tag{20}$$

$$\Delta V_{lx,d} = |V_{lx} - V_{ld}|, \quad \textit{with an importance of } w_l. \tag{21}$$

In order to find a smaller number of variables, as global markers that can be applied to the whole population X, procedures *Step2-step7* are repeated for every

individual x. All variables from the derived sets V_x are then ranked based on their likelihood to be selected for all samples. The top m variables (most frequently selected for testing individual models) are considered as a set of global markers V_m. The procedures P2-P5 will be applied again with the use of V_m as initial variable set (instead of using the whole initial set V of variables). In this case personalised models and profiles are obtained within a set of variable markers V_m that would make treatment and drug design more universal across the whole population X.

4.2 An Optimisation Algorithm (cEAP) for PMS

The method IMPM employs a coevolutionary based algorithm for personalised modelling (cEAP) to optimise related parameters, selecting informative features and finding appropriated neighbourhood for personalised modelling [10]. Given a general objective optimisation problem $f(x)$ to minimize (or maximize), $f(x)$ is subject to two constraints $g_i(x)$ and $h_j(x)$. A candidate solution is to minimize the objective function $f(x)$ where x represents a n-dimensional decision (or optimisation) variable vector $X = \{x_i \mid i = 1, \cdots, n\}$ from the sample space Ω. The two constraints describe the dependence between decision variables and parameters involved in the problem, and must be satisfied in order to optimise $f(x)$. The constraints $g_i(x)$ and $h_j(x)$ are denoted as inequalities and equalities respectively and mathematically formulated as:

$$g_i(x) \leq 0 \mid i = 1, \ldots, n \tag{22}$$

$$h_j(x) = 0 \mid j = 1, \ldots, p \tag{23}$$

The number of degrees of freedom is calculated by $n - p$. Note the number of equality constraints must be smaller than the number of decision variables (i.e. $p < n$). The *overconstrained* issue, occurs when $p \geq n$, because there is no degrees of freedom left for optimising objective function.

The algorithm aims to find the optimal solution to an objective function. Given an objective function $f(x)$: for $x \in \Omega, \Omega \neq \emptyset$, a global minimum of the objective problem $f(x)$ can be mathematically defined as $f^* \triangleq f(x^*) > -\infty$, ***only if***

$$\forall x \in \Omega : \quad f(x^*) \leq f(x) \tag{24}$$

where x^* denotes the minimum solution, Ω is the sample universe of x.

The optimisation algorithm for selecting genes and optimising the parameters of learning functions (e.g. a classifier threshold θ and the number of neighbours k_v) simultaneously. The basic idea underlying cEAP algorithm is to coevolve the search in multiple search spaces (here is for feature/variable selection and parameter optimisation).

The objective of IMPM is to build personalised models for data analysis and knowledge discovery which are able to minimise the prediction accuracy of disease distinction and create a personalised profile for individual patient. Given a data $D =$

Algorithm 2 The optimisation algorithm - cEAP

1: initialize the subindividuals in the subcomponent for feature selection:
generate a probability vector p with l bits, $p_i = 0.5$, where $i \in 1, \cdots, l$,
2: generate two subindividuals from the vector p, respectively:
$(G_a, G_b) = generate(p)$;
3: generate a pair of subindividuals K_a, K_b by a probability function f_p;
4: generate a pair of subindividuals: θ_a and θ_b using a probability function f'_p;
5: recombine the above subindividuals from three subcomponents into two individuals:
$\alpha = G_a + K_a + \theta_a$;
$\beta = G_b + K_b + \theta_b$;
6: evaluate individuals α and β by a fitness function F_c, respectively;
7: compete individual α and β:
$winner, loser = compete(\alpha, \beta)$
8: create new populations in three subcomponents:
(i) use GA to create the new generation for feature selection subcomponent
if $G_a(i) \neq G_b(i)$
if $winner(i) = 1$ ***then*** $p_i = p_i + \frac{1}{\mu}$
else $p_i = p_i - \frac{1}{\mu}$
(ii) use evolutionary strategy (ES) to create the new generation for K and θ in the other subcomponents:
Keep the winner of K and θ to form the offsprings K'_a and θ'_a; the other offsprings K'_b and θ'_b are generated through a mutation performed by probability functions f_p and f'_p.
9: check wether the termination criteria are reached:
if **yes**, ***then*** the winner individual represents the final solution ζ^*, including the selected features G^* and optimised parameters K^* and θ^*
otherwise iterate the process from step 2.

$\{X, Y\} \mid X = x_{ij}, Y = y_i, i = 1 \ldots n, j = 1 \ldots m$, the objective is therefore defined to optimise a classifier that involves the selected features and related parameters:

$$f(s^*) \leq f(s) \tag{25}$$

where f is a classification function, and s denotes an independent variables set. As s can be represented by the data vector X, Y with selected features and related parameters, Eq.25 is rewritten as follows:

$$f(X, Y, \zeta_l^*) \leq f(X, Y, \zeta_l), \quad |\zeta \in \Omega, l = \{1, 2, 3\}. \tag{26}$$

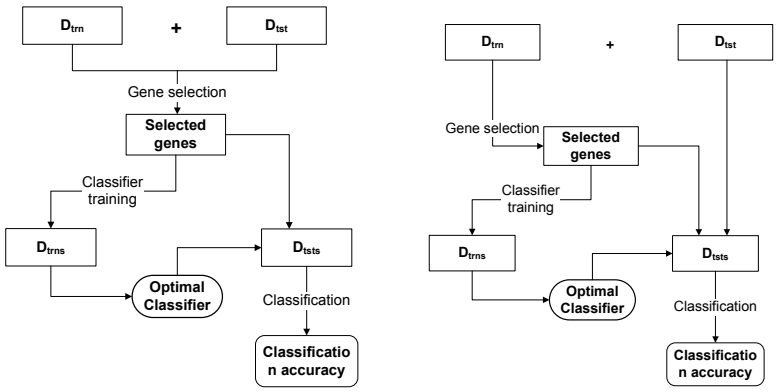
where ζ_l denotes the candidate solution from l different subcomponents. The final solution is obtained when Eq.25 is fulfilled, i.e. ζ_l^* is taken as the desired solution to the problem of gene selection and parameter optimisation when the classification error is less or equal to the value at any other conditions.

For clarity, the pseudo code of the optimisation algorithm cEAP is given in Algorithm 2.

5 Experiment

We present an experiment using personalised modelling with IMPM for diagnosis and profiling of cancer. A benchmark colon cancer gene expression dataset is used [1]. It consists of 62 samples, 40 collected from colon cancer patients and 22 from control subjects. Each sample is represented by 2000 gene expression variables. The objective is to create a diagnostic (classification) system that not only provides an accurate diagnosis, but also profiles the person to help define the best treatment.

An unbiased verification approach for personalised modelling data analysis should guarantee that generalisation errors occur in either feature selection or classification procedures as little as possible. To this end, an efficient data sampling method should be used in the two procedures to maximally decrease the generalisation error. In other words, the reliability and generalisability of the informative features should be evaluated on independent testing subsets, and then these features can be used for classification. The classification also needs to employ verification methods to estimate the bias error. Such procedure is shown in Figure 7(b). For comparison, a simple example of biased validation schema is demonstrated in Figure 7(a).



(a) An example of biased validation scheme; (b) The proposed unbiased validation scheme

Fig. 7 The comparison between a biased and an unbiased verification scheme, where D_{trn} and D_{tst} are the training and testing set, D_{trns} and D_{tsts} are the training and testing set with selected genes, respectively. In case (a) (biased verification scheme), the testing set is used twice in gene selection and classifier training procedure, which introduces a bias error from the gene selection stage into the final classification step. Whereas in case (b) (the unbiased scheme), the testing set is only used in the final classification(validation) stage, i.e. the testing set is independent all through gene selection and classifier training procedures.

5.1 Personalised Modelling with IMPM for Colon Cancer Diagnosis and Profiling on Gene Expression Data

An example of a personalised model of colon cancer diagnosis and profiling of a randomly selected person is given in Figure 8∼13 [15]. Figure 8 shows the evolution (GA) process of feature selection specifically for sample#32 from the colon cancer data through 600 generations. IMPM selects 18 genes (features) out of 2000 genes based the result from the GA optimisation. Figure 9 illustrates the weighted importance of the selected *18* genes from Figure 8. The weighted importance is calculated by a weighted SNR model (refer to Eq.12 and 1). The larger the importance value, the more informative the gene is.

Using the proposed IMPM, an optimised personalised model M_x for sample#32 from the colon cancer data is created. This personalised model M_x consists of the selected 18 informative genes, along with two parameters - classification threshold ($\theta = 0.40$) and the number of neighbouring samples (K = 18), which are optimised specific for sample#32. Figure 10 shows the data subset D_x with 18 samples (the neighbourhood with an appropriate size) of sample#32 using top 3 selected genes (gene 377, 1285 and 1892). These neighbouring samples are:
61, 41, 12, 1, 38, 22, 26, 31, 34, 28, 19, 44, 6, 49, 57, 3, 8, 43.
The predicted outcome computed by the optimised personalised model M_x^* is 0.51, which successfully classifies sample#32 into diseased class (class 2) (the classification threshold θ is optimised to 0.40 as a model parameter).

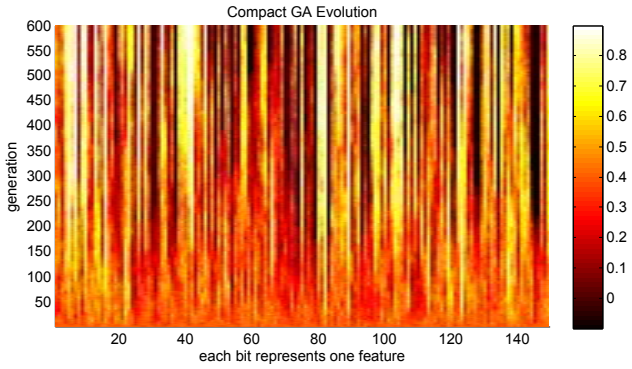


Fig. 8 The evolution of feature (variable) selection for sample#32 from the Colon cancer data (600 generations of GA optimisation; the lighter the colour, the higher the probability of the feature to be selected; each feature is represented as one bit on the horizontal axis; at the beginning all features are assigned equal probability to selected as 0.5)

Using the IMPM, a profile and a scenario of potential genome improvement for colon sample#32 was created shown in Figure 11. Desired average profile is the average gene expression level from healthy samples group and desired improvement

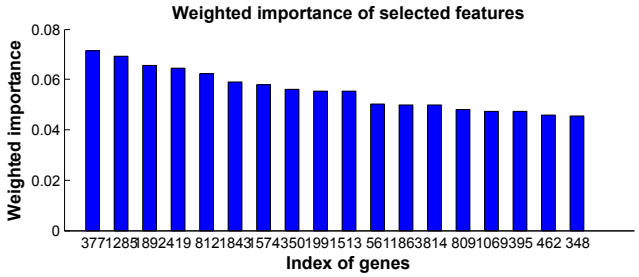


Fig. 9 The weighted importance of the selected features for sample#32 using weighted SNR based model (refer to Eq.12 $\sim$)

value identifies the change of the gene expression level that this patient (sample#32) should follow in order to recover from the disease. For example, the expression level of gene 377 of sample#32 is 761.3790, while the average class profile for class 1 (normal class) and class 2 (diseased class) are: *233.8870* (for class 1) and *432.6468* (for class 2). The distance between the gene expression level of gene 377 for sample#32 and the desired average class 1 profile is 527.4920, i.e. a potential solution can be given to the colon cancer patient (sample#32) to decrease his/her gene 377 expression level from 761.3790 to 233.8870. The information in the generated profile can be used for designing personalised treatment for cancer patients.

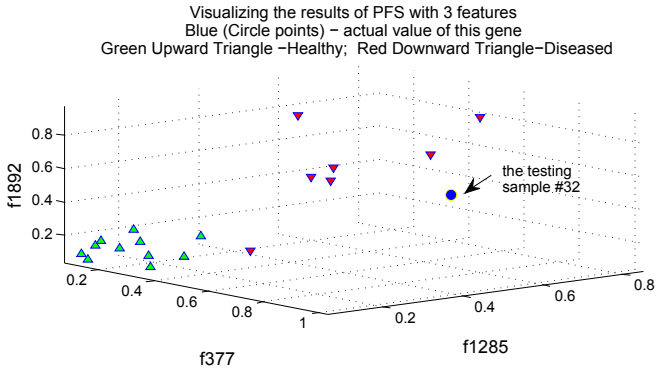


Fig. 10 Sample#32 (the blue dot) is plotted with its 18 neighbouring samples selected by IMPM (red triangles - cancer samples and green triangles - control) in the 3D space of the top 3 gene variables (genes 377, 1285 and 1892) from Fig.9

To find a small number of variables (potential markers) for the whole population of colon cancer data, we have used the approach as follows: Based on the experiment result for every sample, we selected 20 most frequently used genes as potential global markers. Table 1 lists these 20 global markers with their biological informa-

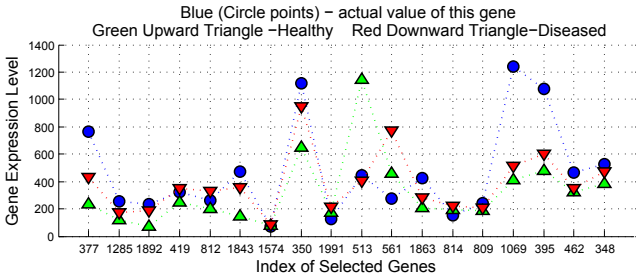


Fig. 11 The profile of sample#32 (blue dots) versus the average local profile of the control (green) and cancer (red) samples using the 18 selected genes from Fig.9 as derived through the IMPM.

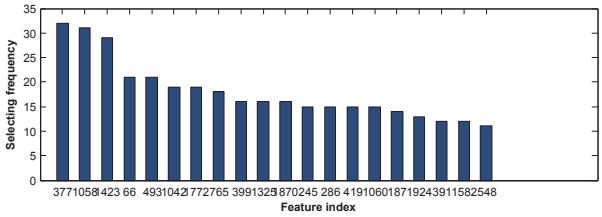


Fig. 12 The 20 most frequently selected genes using IMPM across all colon cancer data samples, where x axis represents the index of the gene in the data and y axis is the frequency of the gene as the marker of the optimised personalised models for which this gene has been selected.

tion. Here we use 20 selected genes as global markers. The number of 20 is based on the suggestion in Alon's work [1].

The next objective of our experiment is to investigate whether utilising these 20 potential marker genes can lead to improved colon cancer classification accuracy and what classification algorithm will perform best in the proposed IMPM. Four classification algorithms are tested as personalised models in this experiment, including WKNN, MLR, SVM and TWNFI. All the classification results from four classifiers are validated based on leave-one-out cross validation (LOOCV) across the whole dataset. Figure 13 shows the average accuracy obtained by these four algorithms with different size (K_x) of neighbourhood. Table 2 summarises the classification results from the four classification algorithms using 20 selected potential marker genes. WKNN and a localised SVM yielded improved classification accuracy (90.3%) when compared to the global model [1]. However, the TWNFI classifier obtained the best classification performance (91.9%). Our results suggest that a small set of marker genes selected by the IMPM could lead to improved cancer classification accuracy.

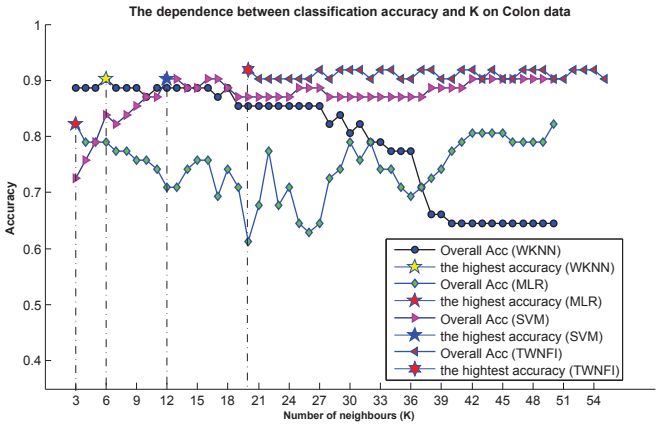


Fig. 13 A comparison of classification results obtained by 4 classification algorithms using 20 potential maker genes from Figure 12 , where x axis represents the size of neighbourhood and y axis is the average classification accuracy across all samples. The best accuracy is obtained with the use of the TWNFI classification algorithm (91.90%)

Table 1 The 20 most frequently selected genes (potential marker genes) using the proposed IMPM across all colon cancer gene data samples (see Figure 12)

Index of Gene	GenBank Accession Number	Description of the Gene (from GenBank)
G377	Z50753	H.sapiens mRNA for GCAP-II/uroguanylin precursor
G1058	M80815	H.sapiens a-L-fucosidase gene, exon 7 and 8, and complete cds.
G1423	J02854	Myosin regulatory light chain 2, smooth muscle ISOFORM (HUMAN)
G66	T71025	Human (HUMAN)
G493	R87126	Myosin heavy chain, nonuscle (Gallus gallus)
G1042	R36977	P03001 Transcription factor IIIA
G1772	H08393	COLLAGEN ALPHA 2(XI) CHAIN (Homo sapiens)
G765	M76378	Human cysteine-rich protein (CRP) gene, exons 5 and 6.
G399	U30825	Human splicing factor SRp30c mRNA, complete cds.
G1325	T47377	S-100P PROTEIN (HUMAN).
G1870	H55916	PEPTIDYL-PROLYL CIS-TRANS ISOMERASE, MITOCHONDRIAL PRECURSOR (HUMAN)
G245	M76378	Human cysteine-rich protein (CRP) gene, exons 5 and 6.
G286	H64489	Leukocyte Antigen CD37 (Homo sapiens)
G419	R44418	Nuclear protein (Epstein-barr virus)
G1060	U09564	Human serine kinase mRNA, complete cds.
G187	T51023	Heat shock protein HSP 90-BETA (HUMAN)
G1924	H64807	Placental folate transporter (Homo sapiens)
G391	D31885	Human mRNA (KIAA0069) for ORF (novel proetin), partial cds.
G1582	X63629	H.sapiens mRNA for p cadherin.
G548	T40645	Human Wiskott-Aldrich syndrome (WAS) mRNA, complete cds.

Table 2 The best classification accuracy obtained by four classification algorithms on colon cancer data with 20 potential maker genes. Overall - overall accuracy; Class 1 - class 1 accuracy; Class 2 - class 2 accuracy;

Classifier	**Overal**[%]	Class 1[%]	Class 2[%]	Neighbourhood size
MLR (Personalised)	82.3	90.0	68.2	3
SVM (Personalised)	90.3	95.0	81.8	12
WKNN	90.3	95.0	81.8	6
TWNFI	**91.9**	95.0	85.4	20
Original publication [1]	87.1	-	-	-

6 Conclusion and Future Development of Personalised Modelling System

When compared to global or local modelling, the proposed personalised modelling method (IMPM) has a major advantage. With personalised modelling methods, the modelling process starts with all relevant variables available for a person, rather than with a fixed number of features required by a global model. Such a global model may be statistically representative for a whole population (global problem space), but not necessarily representative for a single person in terms of best prognosis for this person. The proposed IMPM leads to a better prognostic accuracy and a computed personalised profile. With global optimisation, a small set of variables (potential markers) can be identified from the selected variable set across the whole population. This information can be utilised for the development of new more efficient drugs. A scenario for outcome improvement is also created by the IMPM, which can be utilised for the decision of efficient personalised treatment. We hope that this paper will motivate the biomedical applications of personalised modelling research.

Personalised modelling methods and systems are not going to substitute experts and current global or local modelling methods, but they are expected to derive information that is specifically relevant to a person and help individuals and clinicians make better decisions, thus saving lives, improving quality of life, and reducing cost of treatment. The IMPM method is capable of discovering more useful information, including selected informative genes and optimal disease classification parameters specifically for the observed patient sample, which are helpful to construct the clinical decision support systems for cancer diagnosis and prognosis. For biological reference, some of experimental findings are proofed in the literature, e.g. the selected genes of colon cancer data by our method are reported as biomarkers in other published papers.

In summary, personalised modelling offers a novel and integrated methodology that comprises different computational techniques for data analysis and knowledge discovery. Compared with the results obtained by other published methods, the new algorithms and methods based on personalised modelling have produced improved outcomes in terms of prediction accuracy and discovered more useful knowledge, because they take into account the location of new input sample in a subspace. The

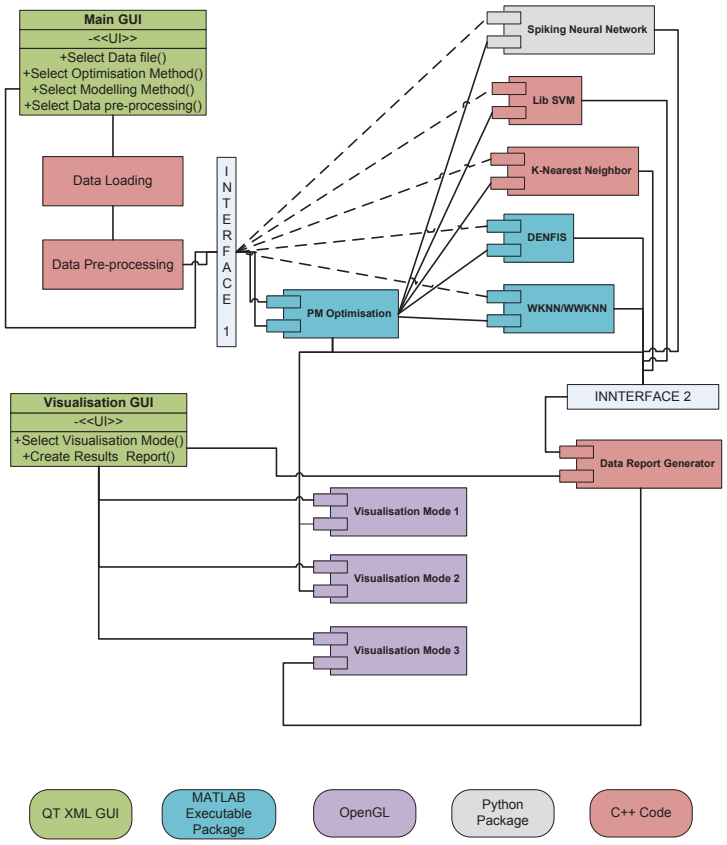


Fig. 14 A framework structure of the proposed ISPM system

subspace (local space) excludes noise data samples and provides more precise information for analysing new input data sample.

Personalised modelling is an adaptive and evolving technique, in which new data sample can be continuously added to the training dataset and subsequently contribute the learning process of personalised modelling. More importantly, the technique of personalised modelling offers a new tool to give a profile for each new individual data sample. Such characteristic makes personalised modelling based methods are promising for medical decision support systems and personalised medicine design, especially for complex human disease diagnosis and prognosis, such as cancer and brain disease.

However, as a personalised modelling system creates a unique (personalised) model for each testing data sample, it requires more computational power and performance time than traditional global modelling methods, especially to train the models on large data sets. The proposed methods have shown the great potential for

solving the problems that require individual testing. This study is the first step in this research direction and needs more in-depth understanding in bioinformatics for validating the experimental findings and knowledge discovery.

The next step for personalised modelling study is to develop a software platform, called Integrated Optimisation System for Personalised Modelling (ISPM) that utilises the proposed novel personalised modelling methodology for data analysis and medical decision support system. This platform offers an user-friendly environment for predicting the outcome of individual samples based on personal data and historical data of other similar cases, regardless of the type and the number of the available data and variables. It incorporates a variety of computational intelligent techniques for personalised modelling for solving different types of research problems. Figure 14 shows a framework structure of the proposed ISPM system.

The main feature of the ISPM system is that it optimises all the factors related to the given objective in an integrated way, such as the features (variables), the local problem space, the classification model and its model parameters, with an objective function - best accuracy of predicted results for every individual input vector (sample, patient). ISPM includes a cross-platform class library, integrated development tools and a cross-platform IDE. The system integrates a variety of modelling methods based on classical statistical algorithms and sophisticated models developed by KEDRI. The software system is expected to provide not only improved prediction accuracy, but reliable risk probability for disease diagnosis and prognosis and personalised profiles that would help define the best actions (e.g., treatment). ISPM will be available both as off-line system and as on-line web-based version.

References

1. U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences of the United States of America*, 96:6745–50, 1999.
2. Judy Anderson, Lise Lotte Hansen, Frank C. Mooren, Markus Post, Hubert Hug, Anne Zuse, and Marek Los. Methods and biomarkers for the diagnosis and prognosis of cancer and other diseases: Towards personalized medicine. *Drug Resistance Updates*, 9(4-5):198–210, 2006.
3. James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA, 1982.
4. Christopher Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, USA, 1995.
5. Z. Bosnic, I. Kononenko, M. Robnik-Sikonja, and M. Kukar. Evaluation of prediction reliability in regression using the transduction principle. In *EUROCON 2003. Computer as a Tool. The IEEE Region 8*, volume 2, pages 99–103 vol.2, 2003.
6. Patrik D'haeseleer, Shoudan Liang, and Roland Somogyi. Genetic network inference: from co-expression clustering to reverse engineering. *Bioinformatics*, 16(8):707–26, 2000.
7. T. Furey, N. Cristianini, Nigel Duffy, D. Bednarski, Michel Schummer, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.
8. Thore Graepel, Matthias Burger, and Klaus Obermayer. Self-organizing maps: Generalizations and new optimization techniques. *Neurocomputing*, 21:173–190, 1998.

9. Lucia A. Hindorff, Praveen Sethupathy, Heather A. Junkins, Erin M. Ramos, Jayashri P. Mehta, Francis S. Collins, and Teri A. Manolio. Potential etiologic and functional implications of genome-wide association loci for human diseases and traits. *Proc Natl Acad Sci USA*, 106(23):9362–9367, 2009.
10. Yingjie Hu and Nikola Kasabov. Coevolutionary method for gene selection and parameter optimization in microarray data analysis. In C.S. Leung, M. Lee, and J.H. Chan, editors, *Neural Information Processing*, pages 483–492. Springer-Verlag, Berlin / Heidelberg, 2009.
11. T. Jan Jorgensen. From blockbuster medicine to personalized medicine. *Personalized Medicine*, 5(1):55–64, January 2008.
12. Nikola Kasabov. Evolving connectionist systems. In *Methods and Applications in Bioinformatics, Brain Study and Intelligent Machines*. Springer-Verlag, London, 2002.
13. Nikola Kasabov. Global, local and personalized modelling and pattern discovery in bioinformatics: An integrated approach. *Pattern Recognition Letters*, 28(6):673–685, 2007.
14. Nikola Kasabov. Soft computing methods for global, local and personalised modeling and applications in bioinformatics. In Valentina Emilia Balas, Janos Fodor, and Annamaria Varkonyi-Koczy, editors, *Soft Computing Based Modeling in Intelligent Systems*, pages 1–17. Springer, Berlin Heidelberg, 2009.
15. Nikola Kasabov and Yingjie Hu. Integrated optimisation method for personalised modelling and case studies for medical decision support. *International Journal of Functional Informatics and Personalised Medicine*, 3(3):236–256, 2010.
16. Nikola Kasabov and Shaoning Pang. Transductive support vector machines and applications in bioinformatics for promoter recognition. In *Proc. of International Conference on Neural Network and Signal Processing*. IEEE Press, 2004.
17. Nikola Kasabov and Qun Song. Denfis: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *Fuzzy Systems, IEEE Transactions on*, 10(2):144–154, 2002.
18. Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59–69, 1982.
19. Matjaz Kukar. Transductive reliability estimation for medical diagnosis. *Artificial Intelligence in Medicine*, 29:2003, 2002.
20. A. S. Levey, J. P. Bosch, J. B. Lewis, T. Greene, N. Rogers, and D. Roth. A more accurate method to estimate glomerular filtration rate from serum creatinine: a new prediction equation. modification of diet in renal disease study group. *Annals of Internal Medicine*, 130:461–470, 1999.
21. Stuart Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.
22. Alexander V. Lukashin and Rainer Fuchs. Analysis of temporal gene expression profiles: clustering by simulated annealing and determining the optimal number of clusters. *Bioinformatics*, 17(5):405–414, 2001.
23. Mark I. McCarthy and Joel N. Hirschhorn. Genome-wide association studies: potential next steps on a genetic journey. *Human Molecular Genetics*, 17(R2):R156–R165, 2008.
24. Tom Mitchell, Richard Keller, and Smadar Kedar-Cabelli. Explanation-based generalization: A unifying view. *Machine Learning*, 1(1):47–80, 1986.
25. Joseph R. Nevins, Erich S. Huang, Holly Dressman, Jennifer Pittman, Andrew T. Huang, and Mike West. Towards integrated clinico-genomic models for personalized medicine: combining gene expression signatures and clinical factors in breast cancer outcomes prediction. *Human Molecular Genetics*, 12(2):R153–R157, 2003.
26. Shaoning Pang and Nikola Kasabov. Inductive vs transductive inference, global vs local models: Svm, tsvm, and svmt for gene expression classification problems. In *Neural Networks, 2004 IEEE International Joint Conference*, volume 2, pages 1197–1202, 2004.
27. Senate Health, Education, Labor, and Pensions. A bill to secure the promise of personalized medicine for all americans by expanding and accelerating genomics research and initiatives to improve the accuracy of disease diagnosis, increase the safety of drugs, and identify novel treatments, 2007.

28. Amnon Shabo. Health record banks: integrating clinical and genomic data into patient-centric longitudinal and cross-institutional health records. *Personalised Medicine*, 4(4):453–455, 2007.
29. Ray Solomonoff. A formal theory of inductive inference, part i. *Information and Control, Part I*, 7(1):1–22, 1964.
30. Ray Solomonoff. A formal theory of inductive inference, part ii. *Information and Control*, 7(2):224–254, 1964.
31. Qun Song and Nikola Kasabov. Twrbf: Transductive rbf neural network with weighted data normalization. *Lecture Notes in Computer Science*, 3316:633–640, 2004.
32. Qun Song and Nikola Kasabov. Twnfi - a transductive neuro-fuzzy inference system with weighted data normalization for personalized modeling. *Neural Networks*, 19(10):1591–1596, 2006.
33. Laura J. van't Veer, Hongyue Dai, Marc J. van de Vijver, Yudong D. He, Augustinus A. M. Hart, Mao Mao, Hans L. Peterse, Karin van der Kooy, Matthew J. Marton, Anke T. Witteveen, George J. Schreiber, Ron M. Kerkhoven, Chris Roberts, Peter S. Linsley, Rene Bernards, and Stephen H. Friend. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530–536, 2002.
34. V. N. Vapnik. *Statistical learning theory*. New York: Wiley, 1998.
35. Mike West, Carrie Blanchette, Holly Dressman, Erich Huang, Seiichi Ishida, Rainer Spang, Harry Zuzan, John A. Olson, Jeffrey R. Marks, and Joseph R. Nevins. Predicting the clinical status of human breast cancer by using gene expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 98(20):11462–11467, 2001.
36. Donghui Wu, Kristin P. Bennett, Nello Cristianini, and John Shawe-taylor. Large margin trees for induction and transduction, 1999.