T.C.
REPUCLIC OF TURKEY
HACETTEPE UNIVERSITY
INSTITUTE OF HEALTH SCIENCES

BINARY CLASSIFICATION VIA GMDH-TYPE NEURAL NETWORK ALGORITHM

Osman DAĞ

Programme of Biostatistics

INTEGRATED DOCTOR OF PHILOSOPHY THESIS

ANKARA

2018

T.C.
REPUCLIC OF TURKEY
HACETTEPE UNIVERSITY
INSTITUTE OF HEALTH SCIENCES

BINARY CLASSIFICATION VIA GMDH-TYPE NEURAL NETWORK ALGORITHM

Osman DAĞ

Programme of Biostatistics
INTEGRATED DOCTOR OF PHILOSOPHY THESIS

ADVISOR OF THE THESIS
Prof. Dr. Celal Reha ALPAR

CO-ADVISOR OF THE THESIS
Prof. Dr. Erdem KARABULUT

ANKARA
2018

Binary Classification Via GMDH-Type Neural Network Algorithm

Osman Dağ

Supervisor: Prof. Dr. Celal Reha Alpar

Co-supervisor: Prof. Dr. Erdem Karabulut

This thesis study has been approved and accepted as an integrated PhD dissertation in "Biostatistics Program" by the assesment committee, whose members are listed below, on December 27, 2018.

Chairman of the Committee : *Prof. Dr. Ahmet Ergun KARAAĞAOĞLU*
Hacettepe University

Member : *Prof. Dr. Atilla Halil ELHAN*
Ankara University

Member : *Assoc. Prof. Dr. Jale KARAKAYA*
Hacettepe University

Member : *Assoc. Prof. Dr. Ceylan YOZGATLIGİL*
Middle East Technical University

Member : *Assist. Prof. Dr. Sevilay KARAHAN*
Hacettepe University

This dissertation has been approved by the above committee in conformity to the related issues of Hacettepe University Graduate Education and Examination Regulation.

0 2 Ocak 2019

Prof. Diclehan ORHAN, MD, PhD
Institute Manager

YAYINLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin / raporumun tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanım iznini Hacettepe Üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alınarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan **"Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge"** kapsamında tezim aşağıda belirtilen koşullar haricinde YÖK Ulusal Tez Merkezi / H. Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

o Enstitü / Fakülte yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren 2 yıl ertelenmiştir. (1)

o Enstitü / Fakülte yönetim kurulunun gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren ... ay ertelenmiştir. (2)

o Tezimle ilgili gizlilik kararı verilmiştir. (3)

27/12/2018

Osman DAĞ

"Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge"

(1) Madde 6. 1. Lisansüstü tezle ilgili patent başvurusu yapılması veya patent alma sürecinin devam etmesi durumunda, tez danışmanının önerisi ve enstitü anabilim dalının uygun görüşü üzerine enstitü veya fakülte yönetim kurulu iki yıl süre ile tezin erişime açılmasının ertelenmesine karar verebilir.

(2) Madde 6. 2. Yeni teknik, materyal ve metotların kullanıldığı, henüz makaleye dönüşmemiş veya patent gibi yöntemlerle korunmamış ve internetten paylaşılması durumunda 3. Şahıslara veya kurumlara haksız kazanç imkanı oluşturabilecek bilgi ve bulguları içeren tezler hakkında tez danışmanının önerisi ve enstitü anabilim dalının uygun görüşü üzerine enstitü ve fakülte yönetim kurulunun gerekçeli kararı ile altı ayı aşmamak üzere tezin erişime açılması engellenebilir.

(3) Madde 7. 1. Ulusal çıkarları veya güvenliği ilgilendiren, emniyet, istihbarat, savunma ve güvenlik, sağlık vb. konulara ilişkin lisansüstü tezlerle ilgili gizlilik kararı, tezin yapıldığı kurum tarafından verilir*. Kurum ve kuruluşlarla yapılan işbirliği protokolü çerçevesinde hazırlanan lisansüstü tezlere ilişkin gizlilik kararı ise, ilgili kurum ve kuruluşun önerisi ile enstitü veya fakültenin uygun görüşü üzerine üniversite yönetim kurulu tarafından verilir. Gizlilik kararı verilen tezler Yükseköğretim Kuruluna bildirilir.
Madde 7. 2. Gizlilik kararı verilen tezler gizlilik süresince enstitü veya fakülte tarafından gizlilik kuralları çerçevesinde muhafaza edilir, gizlilik kararının kaldırılması halinde Tez Otomasyon Sistemine yüklenir.

* Tez danışmanının önerisi ve enstitü anabilim dalının uygun görüşü üzerine enstitü veya fakülte yönetim kurulu tarafından karar verilir.

ETHICAL DECLARATION

In this thesis study, I declare that all the information and documents have been obtained in the base of the academic rules and all audio-visual and written information and results have been presented according to the rules of scientific ethics. I did not do any distortion in data set. In case of using other works, related studies have been fully cited in accordance with the scientific standards. I also declare that my thesis study is original except cited references. It was produced by myself in consultation with supervisor Prof. Dr. Celal Reha Alpar and co-supervisor Prof. Dr. Erdem Karabulut and written according to the rules of thesis writing of Hacettepe University Institute of Health Sciences.

Osman DAĞ

ACKNOWLEDGEMENTS

Throughout my PhD education, The Scientific and Technological Research Council of Turkey (TUBITAK) financially supported me with 2211/A Scholarship Program.

Hacettepe University Scientific Research Projects Coordination Unit (BAP Koordinasyon Birimi) supported this thesis with project number 16610.

I would like to thank to my advisor, Prof. Dr. Celal Reha Alpar, and my co-advisor, Prof. Dr. Erdem Karabulut for their endless support. Their support made me feel confident and encouraged.

I am also thankful to Assoc. Prof. Dr. Ceylan Yozgatlıgil and Assoc. Prof. Dr. Jale Karakaya for their contribution as being part of the thesis monitoring committee.

I would like to thank to Prof. Dr. A. Ergun Karaağaoğlu, Prof. Dr. Atilla H. Elhan, Assist. Prof. Dr. Sevilay Karahan for their relevant discussions, suggestions and comments.

I am thankful to all instructors who lectured me and made contribution to my skills. Within a special parenthesis, I would like to thank all members of Department of Biostatistics at Hacettepe University for their all support.

I would like to thank to Saman family (Özgür, Meryem ve Ahmet Enes), Ramazan Seyhan, Atilla Eyüpoğlu, Metin Yeşiltepe, Şükrü Keleş for their all support.

I would like to express my appreciations to my family, Murat, Döndü, Nazmi, Nazan, Murathan, Ceren, Sıla. Last but not least, many thanks to my wife, Özlem, for her all support. Also, many thanks to my little princess, Ada, for entering our life and making it colorful for us. This thesis is the product of endless support of Dağ family.

ABSTRACT

Dağ, O., Binary Classification via GMDH-Type Neural Network Algorithm, Hacettepe University Graduate School of Health Sciences Integrated Doctor of Philosophy Thesis in Biostatistics, Ankara, 2018. Group Method of Data Handling (GMDH) - type neural network algorithms are the self organizing algorithms for modeling complex systems. GMDH algorithms are used for different objectives; examples include regression, classification, clustering, forecasting, and so on. In this thesis, we propose a new algorithm named as diverse classifiers ensemble based on GMDH (dce-GMDH) algorithm for binary classification. Also, we develop an R package, GMDH2, to make our proposed algorithm available. The package offers two main algorithms, GMDH and dce-GMDH algorithms. GMDH algorithm performs binary classification and returns important variables. dce-GMDH algorithm performs binary classification by assembling classifiers based on GMDH algorithm. The package also provides a well-formatted table of descriptives in different format (R, LaTeX, HTML). Moreover, it produces confusion matrix and related statistics, and interactive scatter plot (2D and 3D) with classification labels of binary classes to assess the prediction performance. All properties of the package are demonstrated on Wisconsin Breast Cancer data. A Monte Carlo simulation study is also conducted to compare GMDH algorithms to the other well-known classifiers under the different conditions. Moreover, a user-friendly web-interface of the package is developed especially for non-R users. This web-interface is available at http://www.softmed.hacettepe.edu.tr/GMDH2.

Keywords: R Package, Web Tool, Data Mining, Machine Learning Algorithms, Monte Carlo Simulation.

ÖZET

Dağ, O., GMDH Türünde Sinir Ağı Algoritması ile İkili Sınıflandırma, Hacettepe Üniversitesi Sağlık Bilimleri Enstitüsü Biyoistatistik Programı Bütünleşik Doktora Tezi, Ankara, 2018. Veri işleme grup yöntemi (GMDH) türünde sinir ağı algoritmaları karmaşık sistemleri modellemeye yarayan kendi kendini organize eden yöntemlerdir. GMDH algoritmaları regresyon, sınıflandırma, kümeleme, öngörü gibi çeşitli amaçlar için kullanılmaktadır. Bu tez kapsamında GMDH temelli farklı sınıflandırıcıların birleştirilmesi (dce-GMDH) adında yeni bir algoritma önerilmektedir. Bu algoritmaya ulaşılabilmesi için GMDH2 adında bir R paketi geliştirilmiştir. Paket GMDH ve dce-GMDH adında iki temel algoritma sunmaktadır. GMDH algoritması ikili sınıflandırma yapmakta ve önemli değişkenleri bulmaktadır. dce-GMDH algoritması ise farklı sınıflandırıcıları GMDH temelli olarak birleştirerek ikili sınıflandırma yapmaktadır. Paket farklı formatlarda (R, LaTeX, HTML) tanımlayıcı istatistiklerin tablosunu üretmektedir. Ek olarak, paket sınıflandırma performansı değerlendirmek amacıyla karışıklık matrisi, ilgili istatistikleri ve sınıflandırma etiketleri ile birlikte etkileşimli saçılım grafiği (2 ve 3 boyutlu) üretmektedir. Paketin tüm özellikleri Wisconsin meme kanseri verisi ile sunulmaktadır. GMDH algoritmaları ile diğer iyi bilinen sınıflandırıcıları karşılaştırmak amacıyla Monte Carlo benzetim çalışması yapılmıştır. R kullanıcısı olmayanlar için paketin kullanıcı dostu bir web uygulaması geliştirilmiştir. Bu web uygulaması http://www.softmed.hacettepe.edu.tr/GMDH2 adresi ile kullanıma açılmıştır.

Anahtar Kelimeler: R Paketi, Web Aracı, Veri Madenciliği, Makine Öğrenmesi Algoritmaları, Monte Carlo Benzetim Çalışması.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

2D	2-dimensional
3D	3-dimensional
ann	Artificial Neural Network
CRAN	Comprehensive R Archive Network
dce-GMDH	Diverse Classifiers Ensemble Based on GMDH
EC	External Criterion
en	Elastic Net
FN	Number of False Negatives
FP	Number of False Positives
GMDH	Group Method of Data Handling
MCC	Matthews Correlation Coefficient
MAE	Mean Absoluate Error
MSE	Mean Square Error
nb	Naive Bayes
NIR	No Information Rate
NPV	Negative Predictive Value
pp	Proportion of Positives
PPV	Positive Predictive Value
rf	Random Forest
svm	Support Vector Machine
TN	Number of True Negatives
TP	Number of True Positives

LIST OF FIGURES

Figure **Page**

LIST OF TABLES

Table **Page**

1. INTRODUCTION

Binary classification is a classification problem where binary target labels can be assigned to each observation. Binary classification appears in different areas such as medical studies, economics, agriculture, meteorology, and so on. In literature, the traditional methods used for this purpose are logistic regression (1) and discriminant analysis (2). There exist certain assumptions of these models such as linearity between logit and continuous independent variables in logistic regression and multivariate normality in discriminant analysis. Moreover, these methods have some drawbacks especially when the number of independent variables is large or/and the variables are highly correlated. Penalized logistic regression models has been proposed to overcome these problems (3-5). At times, it is difficult for the researchers to select an appropriate model. Therefore, selecting an appropriate model in an automatic way may be extremely attractive for the researchers who do not have enough statistical knowledge or who are not experienced in statistics (6). For this purpose, there exist many machine learning algorithms of which the most commonly used ones are support vector machines (7), artificial neural network (8), random forest (9), naive bayes (7) and so on.

The objective of this thesis is to perform binary classification through Group Method of Data Handling (GMDH) - type neural network algorithms. Since there is no free available code for GMDH algortihms, we first code conventional GMDH algorithm for binary classification. Second, we propose a new method based on GMDH algorithm for binary classification. We name this method as diverse classifiers ensemble based on GMDH (dce-GMDH) algorithm. For the availability of these algorithms, we develop an R package, GMDH2 (10) which performs binary classification through GMDH-type neural network algorithms. The R package includes these aforementioned two main algorithms, GMDH and dce-GMDH algorithms. GMDH algorithm performs classification for a binary response and returns important variables dominating the system. dce-GMDH algorithm performs binary classification by assembling classifiers – support vector machines (7), random forest (9), naive bayes (7), elastic net logistic regression (5), artificial neural network (8) - based on GMDH algorithm.

The GMDH package also produces a well-formatted table of descriptives for a binary response. This table can be obtained in different formats. These are R, LaTeX and HTML. Furthermore, it produces confusion matrix and its related statistics to assess the prediction performance. There exist two functions in the package version 1.4 and later to draw 2-dimensional and 3-dimensional interactive scatter plots with classification labels of binary classes to evaluate the prediction performance. The GMDH2 package is publicly available on the Comprehensive R Archive Network (CRAN). All properties of the package are demonstrated on publicly available Wisconsin breast cancer data set. Also, we develop a web-interface of the R package especially for new R users or applied researchers. We also make Wisconsin breast cancer data available in the tool for the users to test it. This application is available at http://www.softmed.hacettepe.edu.tr/GMDH2.

In this study, we perform binary classification through GMDH-type neural network algorithms. We also conduct a Monte Carlo simulation study to compare the performances of GMDH and dce-GMDH algorithms with support vector machines, random forest, naive bayes, elastic net logistic regression, artificial neural network, and give some general suggestions on which classifier(s) should be used or avoided under different conditions.

The outline of this thesis is presented as follows. In chapter 2, we provide literature review of GMDH algorithms. In chapter 3, we present the methology of the algorithms. In chapter 4, we demonstrate our developed GMDH2 R package on Wisconsin breast cancer data. In chapter 5, the web-interface of the GMDH2 package is introduced. In chapter 6, a Monte Carlo simulation study is conducted for comparison purpose. Finally, the thesis is concluded with conclusion and discussion.

2. LITERATURE REVIEW

The historical development and usage of GMDH algorithm are presented in four parts. The origin of these algorithms is placed in the first part. Usage of GMDH algorithms in the different disciplines is stated in the second part. Methodological development of GMDH algorithms is presented in the third part. Finally, the studies related to classification through GMDH-type neural network algorithms are stated.

2.1. Origin

The origin of GMDH-type neural network algorithm depends on the end of the 1960s years. First, Ivakhnenko (11) proposed a polynomial to construct high order polynomials. After that, Ivakhnenko (12) presented heuristic self-organization methods specifying the architecture of GMDH algorithm by the rules such as external criterion. GMDH algorithms are convenient for complex and unstructured systems and also have benefits over high order regression (6).

2.2. Application Areas

Different problems that the GMDH algorithm handles were defined in the work done by Ivakhnenko and Ivakhnenko (13). Some of them are the identification of physical laws, extrapolation of physical fields, regression, classification, clustering, forecasting and so on.

The usage of GMDH algorithm has been increasing over years. GMDH algorithm was used in environmental study (14). In that study, GMDH algorithm was used to capture the non-linear relation between characteristics of wood obtained from the trees irrigated with processed wastewater and characteristics of wood obtained from the trees grown up in a common way. In an other study, GMDH algorithm was applied in material processing study (15). The relationship between considerable variables and depth penetration is investigated when explosive cutting process of plates is modeled. Astakhov and Galitsky (16) used GMDH algorithm to investigate the parameters affecting the tool life in gundrilling. Srinivasan (17) utilized GMDH-type neural network to forecast energy demand prediction. Xu et al. (18) used GMDH algorithm to forecast the daily power load. GMDH-type neural network algorithm was used in pipeline systems study (19). GMDH algorithm was used to explore the effect of magnetic field on heat transfer of Cu-water nanofluid (20). Depth of scour below pipelines exposed to waves was predicted through GMDH algorithm. Antanasijevic et al. (21) applied GMDH algorithm on feature selection for the prediction of transition temperatures of bent-core liquid crystals. Xiao et al. (22) applied GMDH-based multiple classifiers ensemble for churn prediction in customer relationship management. GMDH-based approach was utilized for human face recognition (23). Guo et al. (24) predict oilfield production via GMDH-type neural network algorithm.

2.3. Methodological Development

The development of GMDH algorithm increased in the last two decades. Kondo (25) used the heuristic self-organization method in GMDH algorithm. Muller et al. (26) used GMDH-type neural network to model complex systems. Sometimes, statistical models are not enough to handle some problems, such as high dimensional data. Obtaining the result in an automatic way is a compelling way for the researchers keen on the result and not having enough statistical knowledge and enough time. Kondo and Ueno (27) proposed GMDH algorithm with a feedback loop on medical image recognition of the brain. Sigmoid transfer function was integrated into GMDH algorithm with a feedback loop (28). Three transfer functions - sigmoid, radial basis and polynomial functions - were integrated into feedback GMDH algorithm (29). Dag and Yozgatligil (30) developed an R package, GMDH, for short term forecasting through GMDH algorithms.

2.4. The Studies Related to Classification through GMDH Algorithm

GMDH-type neural network was utilized for feature selection and classification of medical data (31). El-Alfy and Abdel-Aal (32) used GMDH algorithm for spam detection and email feature analysis. GMDH algorithm was applied for intelligent intrusion detection (33). In that study, network traffic was classified into two classes: normal and anomalous.

All in all, the origin of GMDH algorithm is presented. Different areas in which GMDH algorithm are applied are stated. Also, we present the works related to methodological development of GMDH algorithm and the studies using GMDH algorithm for the purpose of classification. In following chapters, the methodology of GMDH algorithms is presented. An R package and its web-interface are introduced. All properties of the R package are demonstrated on a real data set. Moreover, the simulation results are discussed.

3. METHODOLOGY

In this chapter, feature selection and classification through GMDH algorithm are presented. Also, dce-GMDH algorithm for classification is introduced.

3.1. Feature Selection and Classification through GMDH Algorithm

GMDH-type neural network algorithm is a heuristic self-organization method that investigates the relations among the variables. The algorithm defines its structure itself. Ivakhnenko (11) presented the following polynomial - known as the Ivakhnenko polynomial - to construct a high order polynomial.

$$y = a + \sum_{i=1}^{m} b_i x_i + \sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij} x_i x_j + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m} d_{ijk} x_i x_j x_k + \cdots \tag{3.1}$$

where *m* is the number of variables to be regressed in each neuron and a, b, c, d, ... are weights of variables in the polynomial. Here, y is a response variable, x_i, x_j and x_k are the exploratory variables. In this study, only the main effects are included in the model as presented below,

$$y = a + \sum_{i=1}^{m} b_i x_i \tag{3.2}$$

The GMDH algorithm, in general, investigates all pairwise combinations of p exploratory variables. Therefore, m is specified as 2 in equation 3.2. For this algorithm, there exist three weights to be estimated in each neuron. The weights are estimated via least square estimation. In model building and evaluation process, the data are divided into three sets; train (60%), validation (20%) and test (20%) sets. Train set is included in model building. Validation set is used for neuron selection. Test set is utilized to estimate the performance of the methods on unseen data. The GMDH algorithm can be depicted as follows:

i) Each pairwise combination goes into one neuron.

ii) Weights are estimated with least suare estimation on train set in each neuron at layer k.

iii) The predicted probabilities of train set are estimated in each neuron at layer k.

iv) The predicted probabilities of validation set are estimated in each neuron at layer k.

v) The external criterion (EC) (i.e., mean square error) is calculated using validation set in each neuron at layer k.

vi) Selection pressure (α) (varies between 0 and 1, is preferably chosen greater than 0.5 to give more weight to min EC) and the maximum number of neurons to be selected need to be specified.

vii) The neurons whose external criteria are smaller than $(\alpha \cdot \min(\mathrm{EC}) + (1 - \alpha) \cdot \max(\mathrm{EC}))/2$ are selected. If the number of selected neurons is larger than the specified maximum number of neurons, the neurons - as many as the specified maximum number of neurons - having smaller external criterion compared to the rest of them are selected.

viii) The predicted probabilities of train set obtained from selected neurons become the inputs for the next layer.

ix) This process (i) to (viii) continues until the stopping rule is realized.

x) There are three stopping rules to conclude the algorithm. The first one is an increase in minimum external criterion at consecutive layers. Second, the algorithm stops when the specified maximum number of layers is reached. The third one is that the algorithm stops if only one neuron in a layer is selected.

xi) At the last layer, only one neuron having minimum EC is selected.

GMDH algorithm is a system of layers where the neurons are present. The number of neurons in a layer is determined by the number of inputs. For example, providing that the number of inputs going into a layer is equal to p, the number of neurons in that layer becomes $h = \binom{p}{2}$, since all pairwise combinations of inputs are considered. This does not mean that all layers include h neurons. For instance, the number of inputs in the input layer defines just the number of neurons in first layer. The number of neurons selected in the first layer determines the number of neurons in second layer. The algorithm organizes the architecture itself. Sample architecture of GMDH algorithm is placed in Figure 3.1 when there exist three layers and four inputs.

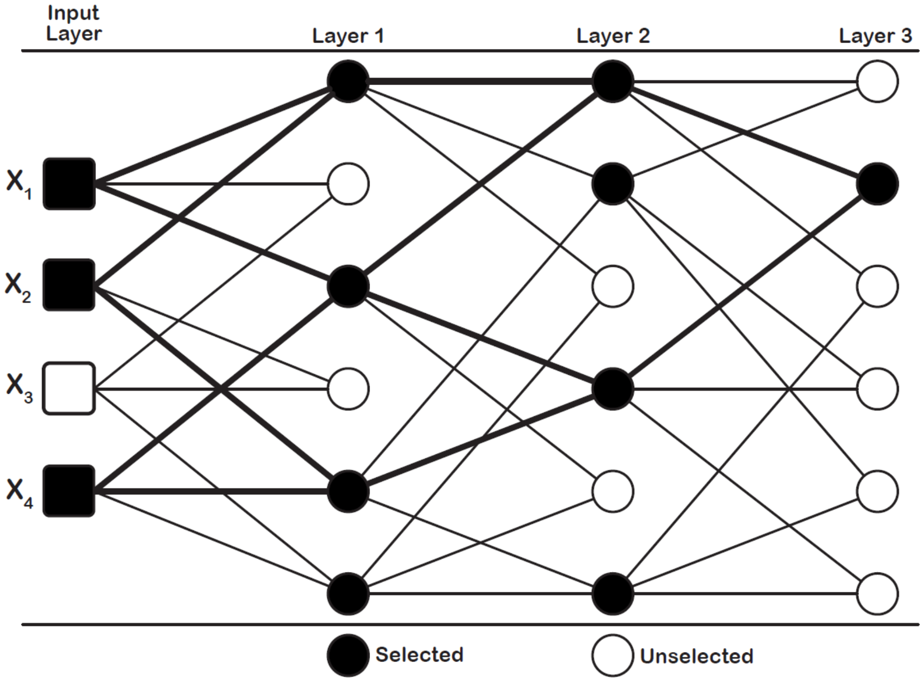


Figure 3.1. Architecture of GMDH algorithm

In the GMDH architecture shown in Figure 3.1, there exist four inputs (X_1, X_2, X_3, X_4). From these input variables, three of them (X_1, X_2, X_4) are dominating the system. X_3 does not have an impact on classification. In this study, GMDH algorithm selects these important features having an effect on classification.

3.2. Diverse Classifiers Ensemble Based on GMDH Algorithm

Diverse classifiers ensemble based on GMDH (dce-GMDH) algorithm is the GMDH algorithm which assemble the well-known classifiers - support vector machines, random forest, naive bayes, elastic net logistic regression, artificial neural network. These classifiers are available in e1071 (7), randomForest (9), e1071 (7), glmnet (5), nnet (8) packages, respectively. Specifically, these classifiers are available in svm (e1071), randomForest (randomForest), naiveBayes (e1071), cv.glmnet (glmnet), nnet (nnet) functions, respectively. Unlike GMDH algorithm, dce-GMDH algorithm includes base layer (Layer 0). The classifiers are placed at base layer. Predicted probabilities are obtained using all inputs through these classifiers. The predicted probabilities obtained from these classifiers continue their way as inputs of first layer without applying any neuron selection process. The rest of the algorithm is same as GMDH algorithm. The sample architecture of dce-GMDH algorithm is demonstrated in Figure 3.2.

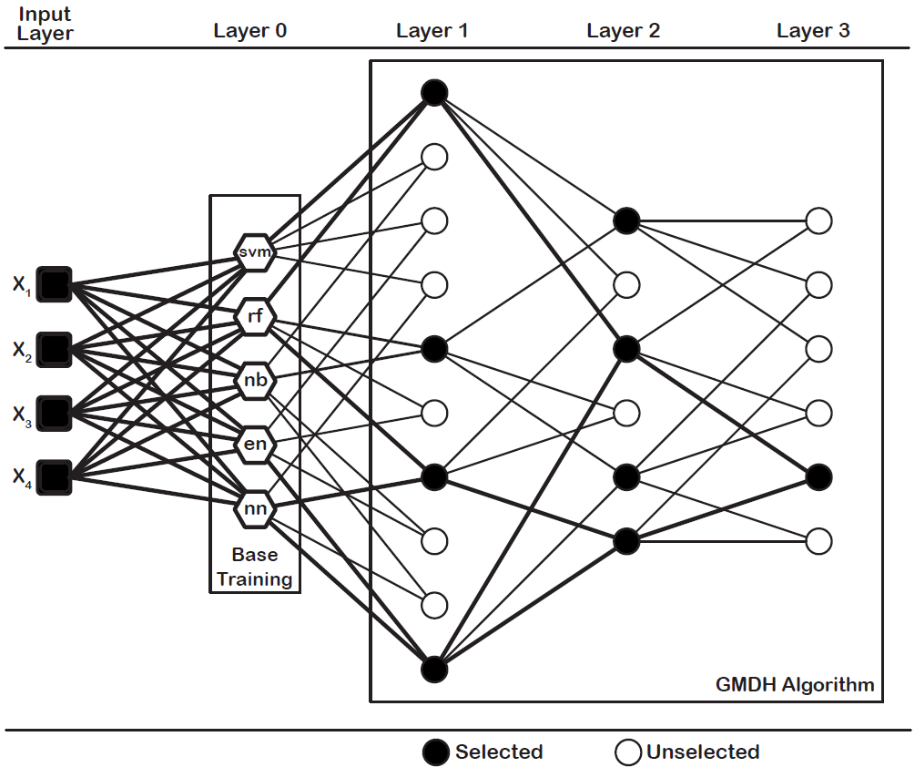


Figure 3.2. Architecture of dce-GMDH algorithm

The dce-GMDH algorithm is a system of layers where the neurons exist. The number of neurons in a base layer is five since the five classifiers are included. The number of neurons in other layers is defined by the number of inputs. The algorithm assembles the most appropriate classifiers by organizing itself. In the dce-GMDH architecture shown in Figure 3.2, there exist four inputs (X_1, X_2, X_3, X_4). These four inputs enter each neuron at base layer. There exists a different classifier in each neuron at base layer. Predicted probabilities are obtained by utilizing four inputs through the classifiers. These predicted probabilities obtained from these classifiers continue to first layer without applying any neuron selection process. Since five inputs will enter in the first layer, the number of neurons in that layer becomes $\binom{5}{2} = 10$. According to external criterion, four neurons are selected and six neurons are eliminated from the network. Since four neurons are selected in the first layer, the number of neurons in the second layer becomes $\binom{4}{2} = 6$. This process continues until one of the stopping rules is realized. Also, the algorithm returns which classifiers are assembled.

3.3. Methods Assembled in dce-GMDH Algorithm

Diverse classifiers ensemble based on GMDH (dce-GMDH) algorithm is the GMDH algorithm assembling the well-known classifiers - support vector machines, random forest, naive bayes, elastic net logistic regression, artificial neural network. In this part, we give some information about these classifiers for the readers to have an intuition for these classifiers.

3.3.1. Support Vector Machine

Support vector machine (svm) is the classifier that attempts to find a linear hyper-plane separating the observations into the two classes. After that, an extension of the method was developed for multi-class classification. The svm is known for its capacity to solve the large amount of problems, such as text classification and image recognition (34).

Support vector machine is the machine learning algorithm used for both classification and regression purposes. svm is more commonly utilized for the classification purpose. Therefore, the classification purpose is what we will focus on in this part. The main idea of svm is to find a hyperplane dividing a dataset into two classes in a best way. The sample illustration of svm classifier in 2d view is given in Figure 3.3. The ojective is to obtain the support vectors by maximizing the marjin between support vectors. Also, there exist some different kernel functions (linear, polynomial, radial basis, sigmoid) to transform the data in more suitable scale.

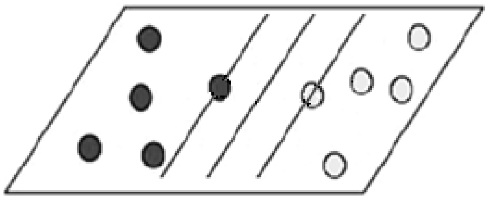


Figure 3.3. The illustration of svm classifier in 2d view

What if such a linear discrimination like in Figure 3.3 is not possible? In that case, it is needed to take the data from a 2d view of the data to a 3d view given in Figure 3.4. The discrimination of the classes is now in three dimension. The hyperplane is now a plane, not a line.

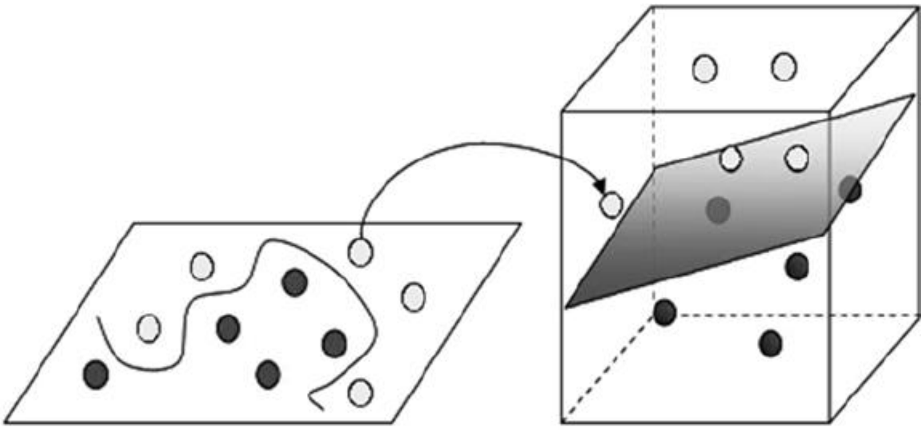


Figure 3.4. The illustration of svm classifier in 3d view

Until the discrimination of the data is completed via a hyperplane, the data are mapped into higher and higher dimensions.

3.3.2. Random Forest

A random forest (rf) (35) is a classifier composed of a collection of decision trees. Each tree is trained independently on a set of observations selected from the complete training set by using Bootstrap method. Some of variables are randomly selected and used in each tree. Random Forest is used for both classification and regression purposes. If Random Forest is utilized for classification purpose, the most frequent class of the individual trees becomes the predicted class. If Random Forest is utilized for regression purpose, the mean of outputs obtained from the individual trees becomes the predicted output. The sample architecture of Random Forest is given in Figure 3.5.

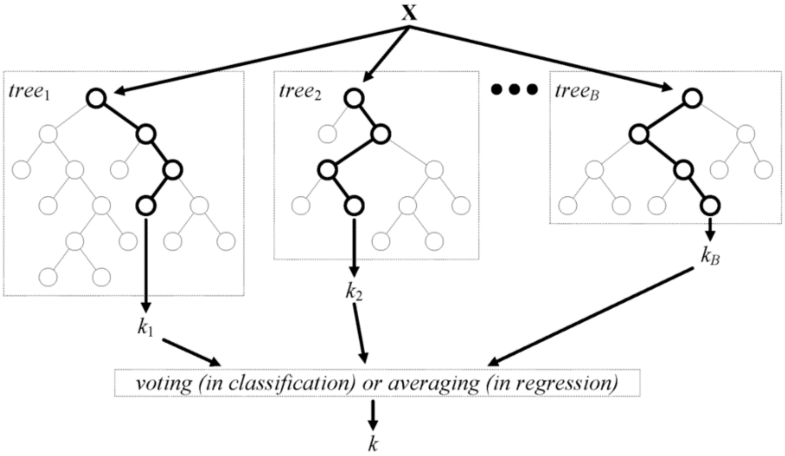


Figure 3.5. Architecture of the random forest model (36)

3.3.3. Naive Bayes

Naive Bayes (nb) classifier is a simple probabilistic classifier based on Bayes' theorem. It has strong independence assumptions between the variables. This helps to solve the problems occurring from high dimensionality. Naive Bayes model is easy to construct since it has no complicated iterative parameter estimation. Thus, it is also useful for large datasets. Basically, Bayes' theorem calculates the probability of each possible class given the predictors that has already occured. Then, it selects the class with highest probability.

3.3.4. Elastic Net Logistic Regression

Penalized logistic regression models have been proposed to overcome the problem of high correlations between independent variables (3-5). Penalized logistic regression models include ridge, lasso, elastic-net (mixture of ridge and lasso) logistic regression models. The main idea of these models is to shrink the coefficients of correlated predictors. If the mixing parameter is fixed to 0, the model is called “ridge logistic regression”. If the mixing parameter is fixed to 1, the model is called “lasso logistic regression”. If the mixing parameter is between 0 and 1, the model is called “elastic net logistic regression.”. Throughout this thesis, we fix the mixing parameter to 0.5. Elastic net is abbreviated with “en” throughout the thesis.

3.3.5. Artificial Neural Network

An artificial neural network (ann) is an information processing system inspired by biological nervous systems (37). Artificial neural networks are parallel architectures solving problems through connected artificial neurons. There exist three layer types in ann. These are input layer, hidden layer(s) and output layer. The data are presented in input layer for the network. Hidden layers are used to enable the networks between inputs. The response of the networks to the input is obtained in output layer. Each neuron is connected to all neurons at the next layer. The sample architecture of ann is given in Figure 3.6.

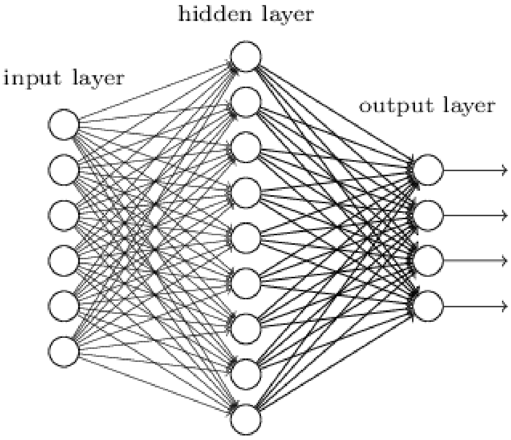


Figure 3.6. Architecture of ann classifier

Most of artificial neural networks use back-propagation paradigm. The weights of the neurons are updated in training process. These updates are made by reducing the error function. It utilizes the method of the gradient-descent while minimizing the error function.

3.4. Performance Measures

In this part, we give performance measures used for 2×2 confusion matrix. These are accuracy, no information rate, Kappa statistic, Matthews correlation coefficient, sensitivity, specificity, positive predictive value, negative predictive value, prevalence, balanced accuracy, Youden index, detection rate, detection prevalence, and F1 measure.

Suppose a 2×2 table with notation,

Table 3.1. The 2×2 confusion matrix

	Reference	
Predicted	Event	No Event
Event	TP	FP
No Event	FN	TN

TP is the number of true positives, FP is the number of false positives, FN is the number of false negatives and TN is the number of true negatives.

3.4.1. Accuracy

Accuracy is described as the percentage of correct predictions. Accuracy varies between 0 and 1. The values of this statistic which are close to 1 indicate high classification performance.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{3.3}$$

3.4.2. No Information Rate

No information rate (NIR) is the largest class percentage in the data. For binary classes, NIR varies between 0.5 and 1. The value of NIR increases as the unbalance in class increases. Also, NIR is used to assess the accuracy performance by investigating how larger accuracy is than NIR.

$$NIR = \max(Prevalence, 1 - Prevalence) \tag{3.4}$$

3.4.3. Kappa

Kappa measures the agreement between two categorical variables. Kappa statistic takes the maximum value of 1. If the Kappa statistic is equal to 1, there exists a complete agreement between two categorical variables. The Kappa statistic gets larger, as the agreement between two variables increases.

$$Kappa = \frac{Accuracy - \frac{(TP + FP)(TP + FN) + (FN + TN)(FP + TN)}{(TP + FP + FN + TN)^2}}{1 - \frac{(TP + FP)(TP + FN) + (FN + TN)(FP + TN)}{(TP + FP + FN + TN)^2}} \tag{3.5}$$

3.4.4. Matthews Correlation Coefficient

Matthews correlation coefficient (MCC) is the correlation coefficient between predicted and reference variables. MCC changes between -1 and 1. A coefficient of 1 indicates a perfect prediction, 0 represents no better than random prediction and −1 shows total disagreement between predicted and reference variables. The statistic is also known as the phi coefficient.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times (FN + TN) \times (TP + FN) \times (FP + TN)}} \tag{3.6}$$

3.4.5. Sensitivity

Sensitivity is the performance measure indicating the proportion of actual positives that are correctly classified. Sensitivity varies between 0 and 1.

Classification performance of actual positives increases as this statistic gets closer to 1. This performance measure is also known as recall.

$$Sensitivity = \frac{TP}{TP + FN} \tag{3.7}$$

3.4.6. Specificity

Specificity is the performance measure representing the proportion of actual negatives that are correctly classified. Specificity changes between 0 and 1. Classification performance of actual negatives increases as this statistic gets larger.

$$Specificity = \frac{TN}{TN + FP} \tag{3.8}$$

3.4.7. Positive Predictive Value

Positive predictive value (PPV) is the proportion of positives in prediction that are actual positive result. PPV varies between 0 and 1. This performance measure is also known as precision.

$$PPV = \frac{TP}{TP + FP} \tag{3.9}$$

3.4.8. Negative Predictive Value

Negative predictive value (NPV) is the proportion of negatives in prediction that are originally negative result. NPV changes between 0 and 1. As NPV increases, the performance gets higher.

$$NPV = \frac{TN}{TN + FN} \tag{3.10}$$

3.4.9. Prevalence

Prevalence is a proportion of the disease that are present in a particular population at a given time.

$$Prevalence = \frac{TP + FN}{TP + FP + FN + TN} \tag{3.11}$$

3.4.10. Balanced Accuracy

Balanced accuracy is the arithmetic mean of sensitivity and specificity. This performance measure changes between 0 and 1. The closer balanced accuracy to 1, the more classification performance.

$$Balanced\ accuracy = \frac{Sensitivity + Specificity}{2} \tag{3.12}$$

3.4.11. Youden Index

Like balanced accuracy, Youden index combines sensitivity and specificity into a single measure. This performance measure changes between 0 and 1 as well. Higher values of Youden index indicate higher performance.

$$Youden\ index = Sensitivity + Specificity - 1 \tag{3.13}$$

3.4.12. Detection Rate

Detection rate is the proportion of true positives in a particular population at a given time.

$$Detection\, rate = \frac{TP}{TP + FP + FN + TN} \tag{3.14}$$

3.4.13. Detection Prevalence

Detection prevalence is the proportion of positive predictions in a particular population at a given time.

$$Detection\,prevalence = \frac{TP + FP}{TP + FP + FN + TN} \tag{3.15}$$

3.4.14. F1 Measure

F1 measure is the harmonic mean of sensitivity and PPV. Therefore, this performance measure considers the effect of prevalence. F1 measure changes between 0 and 1. Higher values of F1 measure indicate higher performance.

$$F1 = \frac{2}{\frac{1}{Sensitivity} + \frac{1}{PPV}} \tag{3.16}$$

These performace measures are available in our confMat function under GMDH2 package. While comparing GMDH and dce-GMDH algorithms to other classifiers with a Monte Carlo simulation study, we reported accuracy, sensitivity, sprecificity, positive predictive value, negative predictive value, balanced accuracy and F1 measure.

4. DEMONSTRATION OF GMDH2 PACKAGE

The GMDH2 package includes several functions especially designed for binary response. In this part, we work with Wisconsin breast cancer data set, collected by Wolberg and Mangasarian (38), available under the mlbench package (39) in R. This data set includes nine exploratory variables - clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, mitoses - and a binary response variable (benign or malignant). After we put missing observations (16 observations) aside, we have a total of 683 observations (444 and 239 observations in each group, respectively).

After installing and loading GMDH2 package, the functions designed for binary response are available to be used.

```
# load Wisconsin breast cancer data
R> data(BreastCancer, package = "mlbench")
R> data <– BreastCancer

# obtain complete observations
R> data <– data[complete.cases(data),]

# select the exploratory variables
R> x <– data[,2:10]

# select the response variable
R> y <– data[,11]
```

4.1. Table of Descriptive Statistics: Table()

Table() produces a table for simple descriptive statistics for a binary response. It returns frequency (percentage) for the variables with class of factor/ordered. Also, this function returns mean ± standard deviation (median, minimum - maximum) or mean ± standard deviation (median, quartile1 - quartile3) for the variables with class of numeric/integer. The option argument is used to return minimum - maximum or quartile1 - quartile3 values. When this argument is set to "min-max", this function returns mean ± standard deviation (median, minimum - maximum). When this argument is set to "Q1-Q3", this function returns mean ± standard deviation (median, quartile1 - quartile3). The percentages can be specified with the percentages argument as row, column or total percentages. The ndigits argument is a vector of two numbers utilized to specify the number of digits. The first one is used to specify the number of digits for numeric/integer variables. The second one specifies the number of digits for percentages of factor/ordered variables. Default is set to ndigits = c(2,1). There exists output argument to return the output in a specified format (R, LaTeX, HTML). In this example, we use "R" output.

```
# obtain a table for simple descriptive statistics for a binary response
R> Table (x, y, option = "min-max", percentages = "column", ndigits = c(2,1), output = "R")
```

```
|============================================|
                        benign     malignant
             ---           ---           ---
    Observations           444           239
    Cl.thickness
               1   136 (30.6%)     3 ( 1.3%)
               2    46 (10.4%)     4 ( 1.7%)
               3    92 (20.7%)    12 ( 5.0%)
               4    67 (15.1%)    12 ( 5.0%)
               5    83 (18.7%)    45 (18.8%)
               6    15 ( 3.4%)    18 ( 7.5%)
               7     1 ( 0.2%)    22 ( 9.2%)
               8     4 ( 0.9%)    40 (16.7%)
               9     0 ( 0.0%)    14 ( 5.9%)
              10     0 ( 0.0%)    69 (28.9%)
       Cell.size
               1   369 (83.1%)     4 ( 1.7%)
               2    37 ( 8.3%)     8 ( 3.3%)
               3    27 ( 6.1%)    25 (10.5%)
               4     8 ( 1.8%)    30 (12.6%)
               5     0 ( 0.0%)    30 (12.6%)
               6     0 ( 0.0%)    25 (10.5%)
               7     1 ( 0.2%)    18 ( 7.5%)
               8     1 ( 0.2%)    27 (11.3%)
               9     1 ( 0.2%)     5 ( 2.1%)
              10     0 ( 0.0%)    67 (28.0%)
      Cell.shape
               1   344 (77.5%)     2 ( 0.8%)
               2    51 (11.5%)     7 ( 2.9%)
               3    30 ( 6.8%)    23 ( 9.6%)
               4    12 ( 2.7%)    31 (13.0%)
               5     2 ( 0.5%)    30 (12.6%)
               6     2 ( 0.5%)    27 (11.3%)
               7     2 ( 0.5%)    28 (11.7%)
               8     1 ( 0.2%)    26 (10.9%)
               9     0 ( 0.0%)     7 ( 2.9%)
              10     0 ( 0.0%)    58 (24.3%)
   Marg.adhesion
               1   363 (81.8%)    30 (12.6%)
               2    37 ( 8.3%)    21 ( 8.8%)
               3    31 ( 7.0%)    27 (11.3%)
               4     5 ( 1.1%)    28 (11.7%)
               5     4 ( 0.9%)    19 ( 7.9%)
               6     3 ( 0.7%)    18 ( 7.5%)
               7     0 ( 0.0%)    13 ( 5.4%)
               8     0 ( 0.0%)    25 (10.5%)
               9     0 ( 0.0%)     4 ( 1.7%)
              10     1 ( 0.2%)    54 (22.6%)
    Epith.c.size
               1    43 ( 9.7%)     1 ( 0.4%)
               2   355 (80.0%)    21 ( 8.8%)
               3    28 ( 6.3%)    43 (18.0%)
               4     7 ( 1.6%)    41 (17.2%)
               5     5 ( 1.1%)    34 (14.2%)
               6     1 ( 0.2%)    39 (16.3%)
               7     2 ( 0.5%)     9 ( 3.8%)
               8     2 ( 0.5%)    19 ( 7.9%)
               9     0 ( 0.0%)     2 ( 0.8%)
              10     1 ( 0.2%)    30 (12.6%)
     Bare.nuclei
               1   387 (87.2%)    15 ( 6.3%)
               2    21 ( 4.7%)     9 ( 3.8%)
               3    14 ( 3.2%)    14 ( 5.9%)
               4     6 ( 1.4%)    13 ( 5.4%)
               5    10 ( 2.3%)    20 ( 8.4%)
               6     0 ( 0.0%)     4 ( 1.7%)
               7     1 ( 0.2%)     7 ( 2.9%)
               8     2 ( 0.5%)    19 ( 7.9%)
               9     0 ( 0.0%)     9 ( 3.8%)
              10     3 ( 0.7%)   129 (54.0%)
     Bl.cromatin
               1   148 (33.3%)     2 ( 0.8%)
               2   153 (34.5%)     7 ( 2.9%)
               3   125 (28.2%)    36 (15.1%)
               4     7 ( 1.6%)    32 (13.4%)
               5     4 ( 0.9%)    30 (12.6%)
               6     1 ( 0.2%)     8 ( 3.3%)
               7     6 ( 1.4%)    65 (27.2%)
               8     0 ( 0.0%)    28 (11.7%)
               9     0 ( 0.0%)    11 ( 4.6%)
              10     0 ( 0.0%)    20 ( 8.4%)
 Normal.nucleoli
               1   391 (88.1%)    41 (17.2%)
               2    30 ( 6.8%)     6 ( 2.5%)
               3    11 ( 2.5%)    31 (13.0%)
               4     1 ( 0.2%)    17 ( 7.1%)
               5     2 ( 0.5%)    17 ( 7.1%)
               6     4 ( 0.9%)    18 ( 7.5%)
               7      2 ( 0.5%)    14 ( 5.9%)
               8      3 ( 0.7%)    20 ( 8.4%)
               9      0 ( 0.0%)    15 ( 6.3%)
              10      0 ( 0.0%)    60 (25.1%)
         Mitoses
               1    431 (97.1%)   132 (55.2%)
               2      8 ( 1.8%)    27 (11.3%)
               3      2 ( 0.5%)    31 (13.0%)
               4      0 ( 0.0%)    12 ( 5.0%)
               5      1 ( 0.2%)     5 ( 2.1%)
               6      0 ( 0.0%)     3 ( 1.3%)
               7      1 ( 0.2%)     8 ( 3.3%)
               8      1 ( 0.2%)     7 ( 2.9%)
              10      0 ( 0.0%)    14 ( 5.9%)
 |============================================|
```

4.2. Feature Selection and Classification through GMDH Algorithm: GMDH()

In this section, we demonstrate GMDH() function for feature selection and classification. It constructs GMDH algorithm, returns summary statistics of GMDH architecture and important variables. First, we randomly divide data into train, validation and test sets, and then call the GMDH() function. The first and second arguments in this function are a matrix of the exploratory variables and a factor of binary response in training set, respectively. The third and fourth arguments are a matrix of the exploratory variables and a factor in validation set, respectively. The alpha argument is the selection pressure. The maxlayers argument is the maximum number of layers requested. The maxneurons argument is the maximum number of neurons allowed in the second and the later layers. The exCriterion argument is the external criterion (mean square error or mean absolute error) to be used for neuron selection. The verbose argument is utilized to print the output in R console.

```
# change the class of x to a matrix
R> x <– data.matrix(x)

# the seed number is fixed to 12345 for reproducibility
R> seed <– 12345

# the number of observations
R> nobs <– length(y)
```

```
R> set.seed(seed)
# to split train, validation and test sets
# to shuffle data
R> indices <– sample(1:nobs)

# the number of observations in each set
R> ntrain <– round(nobs*0.6,0)
R> nvalid <– round(nobs*0.2,0)
R> ntest <– nobs-(ntrain+nvalid)

# obtain the indices of sets
R> train.indices <– sort(indices[1:ntrain])
R> valid.indices <– sort(indices[(ntrain+1):(ntrain+nvalid)])
R> test.indices <– sort(indices[(ntrain+nvalid+1):nobs])

# obtain train, validation and test sets
R> x.train <– x[train.indices,]
R> y.train <– y[train.indices]
R> x.valid <– x[valid.indices,]
R> y.valid <– y[valid.indices]
R> x.test <– x[test.indices,]
R> y.test <– y[test.indices]

R> set.seed(seed)

# construct model via GMDH algorithm
R> model <– GMDH(x.train, y.train, x.valid, y.valid, alpha = 0.6, maxlayers = 10,
maxneurons = 15, exCriterion = "MSE", verbose = TRUE)

Structure :
Layer           Neurons     Selected neurons     Min MSE
1               36          15                   0.063166774906096
2               105         15                   0.0531036043286508
3               105         15                   0.0518891571832988
4               105         15                   0.0516194168250014
5               105         15                   0.0512767947075964
6               105         15                   0.0511084021658896
7               105         15                   0.0509859596771523
8               105         11                   0.0509635614771722
9               55          15                   0.0509600557531984
10              105         1                    0.0509599306139545

External criterion : Mean Square Error

Feature selection : 8 out of 9 variables are selected.
Cl.thickness
Cell.size
```

```
Marg.adhesion
Epith.c.size
Bare.nuclei
Bl.cromatin
Normal.nucleoli
Mitoses
```

Here, the structure includes layer, neurons, selected neurons and min MSE in the output above. The layer shows the number of layer. The neurons represent the number of neurons in corresponding layer. The selected neurons mean the number of selected neurons. The min MSE respresents the minimum external criterion which is calculated for the neuron gives the minimum external criterion on validation set in the corresponding layer. There exist two options for the external criterion; namely, mean square error and mean absolute error.

In feature selection part of the output, eight variables - clump thickness, uniformity of cell size, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, mitoses - are selected by the algorithm. Minimum external criterion can be plotted across layers (presented in Figure 4.1) by the following code.

```
R> plot(model)
```

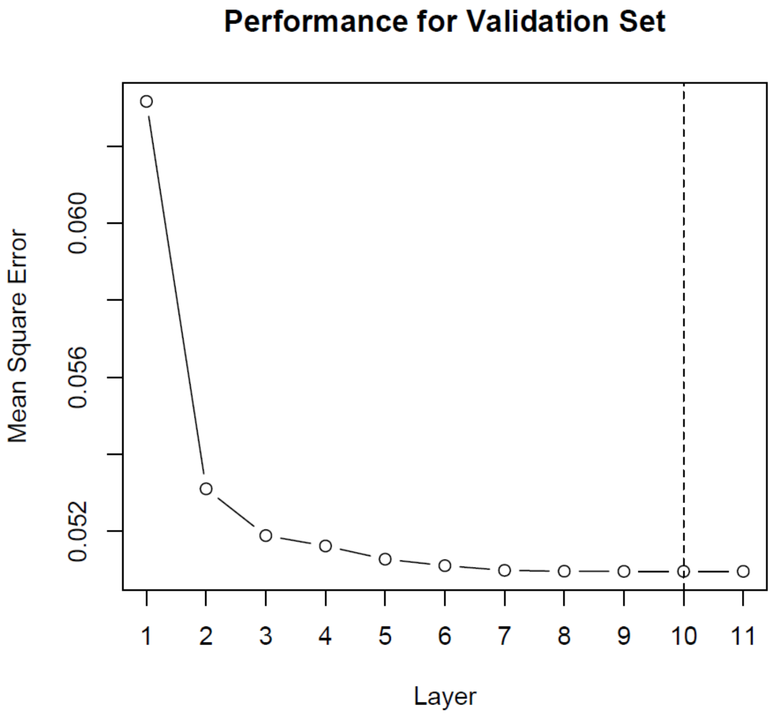


Figure 4.1. Minimum external criterion across layers (GMDH algorithm)

Predictions for test set can be made after model building process is completed. Test set has 136 observations, but only 10 of them are reported to save space.

```
R> predict(model, x.test, type = "class")

[1] benign benign benign benign benign benign malignant benign benign benign
Levels: benign malignant

R> predict(model, x.test, type = "probability")

             benign          malignant
[1,]         1.000000000     0.000000000
[2,]         0.643870382     0.356129618
[3,]         0.670641964     0.329358036
[4,]         0.974398179     0.025601821
[5,]         0.920988111     0.079011889
[6,]         0.994693987     0.005306013
[7,]         0.436033878     0.563966122
[8,]         0.951034736     0.048965264
[9,]         1.000000000     0.000000000
[10,]        0.994693987     0.005306013
```

The GMDH algorithm predicts that the probability of benign for the first and second persons are 100% and 64.4%, respectively. Since the predicted probability of benign is greater than the predicted probability of malignant, these persons are classified as benign.

4.3. Confusion Matrix and Related Statistics: confMat()

The confMat() function produces a confusion matrix for a binary response. It also returns some related statistics. These statistics are accuracy, no information rate, Kappa, Matthews correlation coefficient, sensitivity, specificity, positive predictive value, negative predictive value, prevalence, balanced accuracy, youden index, detection rate, detection prevalence, precision, recall and F1 measure. The formulation of these statistics are stated in section 3.4. The positive argument is an optional character string used to specify the positive factor level. The verbose argument is utilized to print the output in R console.

```
# obtain predicted classes for test set
R> y.test_pred <– predict(model, x.test, type = "class")

# obtain confusion matrix and some statistics for test set
R> confMat(y.test_pred, y.test, positive = "malignant")
```

```
Confusion Matrix and Statistics

                   reference
data               malignant    benign
malignant          51           1
benign             5            79

Accuracy                  :     0.9559
No Information Rate       :     0.5882
Kappa                     :     0.9079
Matthews Corr Coef        :     0.9097
Sensitivity               :     0.9107
Specificity               :     0.9875
Positive Pred Value       :     0.9808
Negative Pred Value       :     0.9405
Prevalence                :     0.4118
Balanced Accuracy         :     0.9491
Youden Index              :     0.8982
```

Detection Rate	:	0.375
Detection Prevalence	:	0.3824
Precision	:	0.9808
Recall	:	0.9107
F1	:	0.9444
Positive Class	:	malignant

Accuracy of GMDH algorithm is estimated to be 0.9559. This algorithm classifies 95.59% of persons in a correct class. Also, sensitivity and specificity are calculated as 0.9107 and 0.9875. The algortihm classifies 91.07% of the persons having breast cancer, 98.75% of the persons not having breast cancer.

4.4. Scatter Plots with Classification Labels: cplot2d() & cplot3d()

The cplot2d() and cplot3d() functions provide interactive 2-dimensional (Figure 4.2) and 3-dimensional (Figure 4.3) scatter plots with classification labels. These functions originally use the plot_ly function from plotly package (40). The first two arguments of cplot2d() are the exploratory variables stated in the x and y axes of Figure 4.2. The first three arguments of cplot3d() are the exploratory variables placed in the x, y and z axes of Figure 4.3. The ypred and yobs arguments are predicted and observed classes. The colors and symbols arguments are used to specify the colors and symbols of true/false classification labels, respectively. The size of symbols can be changed with the size argument. The names of axes can be changed with the arguments xlab, ylab, zlab and title.

```
# 2-dimensional scatter plot with classification labels for test set
R> cplot2d(x.test[,1], x.test[,2], y.test_pred, y.test, colors = c("red", "black"),
xlab = "clump thickness", ylab = "uniformity of cell size")
```

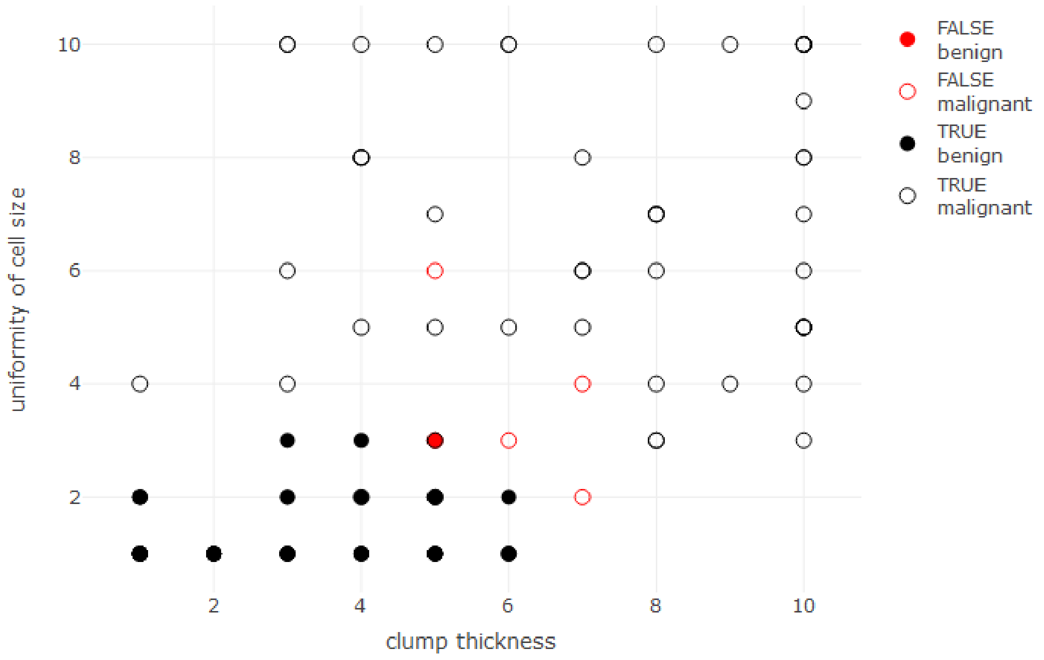


Figure 4.2. 2-dimensional scatter plots with classification labels

```
# 3-dimensional scatter plot with classification labels for test set
R> cplot3d(x.test[,1], x.test[,2], x.test[,6], y.test_pred, y.test, colors = c("red", "black"), xlab = "clump thickness", ylab = "uniformity of cell size", zlab = "bare nuclei")
```

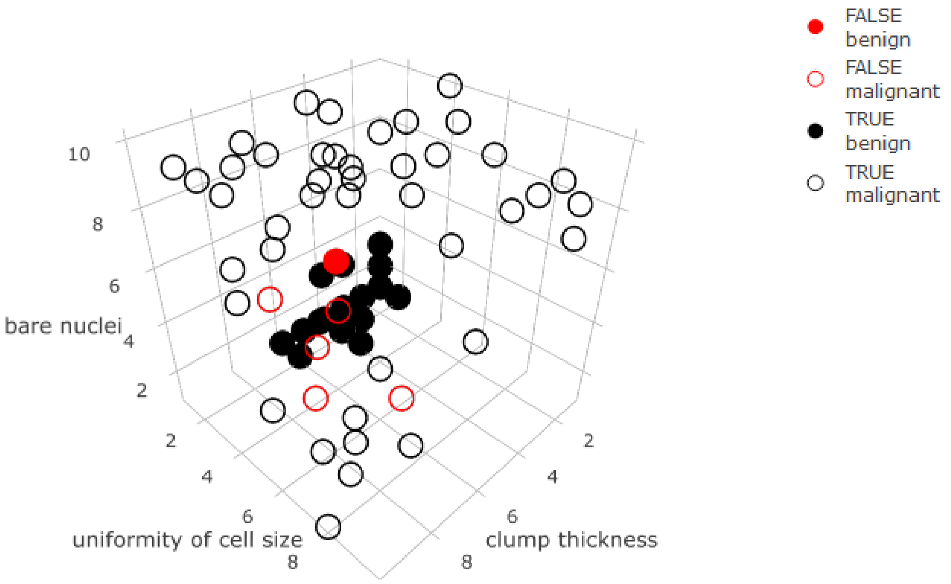


Figure 4.3. 3-dimensional scatter plots with classification labels

4.5. Diverse Classifiers Ensemble Based on GMDH Algorithm: dceGMDH()

In this part, we demonstrate dceGMDH() function for classification. It constructs dce-GMDH algorithm, returns summary statistics of dce-GMDH architecture and assembled classifiers. Like GMDH() function, the first and second arguments are a matrix of the exploratory variables and a factor of binary response in training set, respectively. The third and fourth arguments are a matrix of the exploratory variables and a factor of binary response in validation set, respectively. The alpha argument is the selection pressure. The maxlayers argument is the specified maximum number of layers. The maxneurons argument is the maximum number of neurons allowed in the second and later layers. The exCriterion argument is the external criterion to be utilized for neuron selection. The verbose argument is utilized to print the output in R console. Also, there are the arguments for options of classifiers. The svm_options argument is a list for options of svm. The randomForest_options argument is a list for options of randomForest. The naiveBayes_options argument is a list for options of naiveBayes. The cv.glmnet_options argument is a list for options of cv.glmnet (the elastic net mixing parameter is fixed to 0.5 as default). The nnet_options argument is a list for options of nnet.

```
R> set.seed(seed)
# construct model via dce-GMDH algorithm
R> model <- dceGMDH(x.train, y.train, x.valid, y.valid, alpha = 0.6, maxlayers = 10,
maxneurons = 15, exCriterion = "MSE", verbose = TRUE)

Structure :
Layer           Neurons    Selected neurons    Min MSE
0               5          5                   0.0466953323246936
1               10         1                   0.0464197640066751

External criterion      : Mean Square Error

Classifiers ensemble    : 2 out of 5 classifiers are assembled.
svm
cv.glmnet
```

In this example, two classifiers - support vector machine and elastic net logistic regression – are assembled by the algorithm. Minimum external criterion can be plotted across layers (presented in Figure 4.4) by the following line.

```
R> plot(model)
```

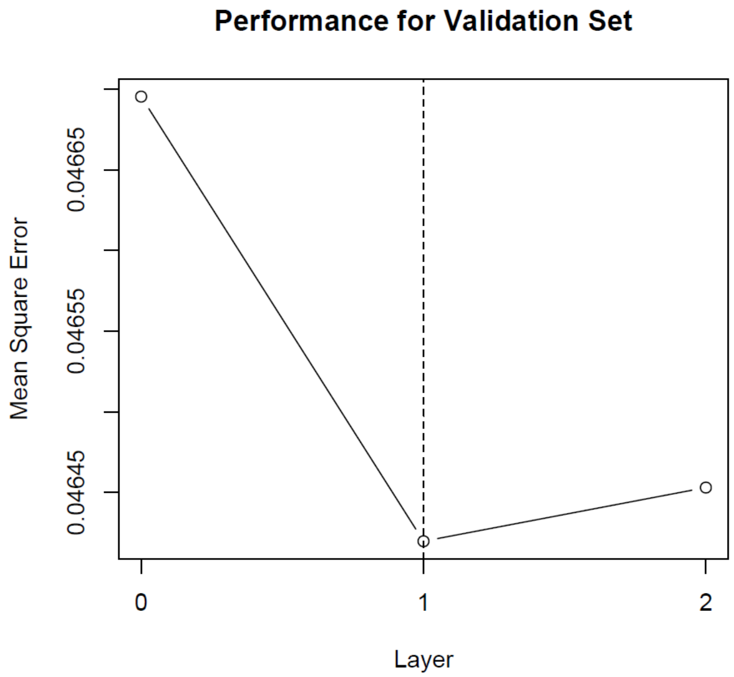


Figure 4.4. Minimum external criterion across layers (dce-GMDH algorithm)

Predictions for test set can be made after model building process is completed. Test set has 136 observations; therefore, 10 of them are reported to save space.

```
R> predict(model, x.test, type = "class")
[1] benign benign malignant benign benign benign malignant benign benign benign
Levels: benign malignant

R> predict(model, x.test, type = "probability")

                 benign          malignant
[1,]             0.9571287282    4.287127e-02
[2,]             0.8317147956    1.682852e-01
[3,]             0.3400820793    6.599179e-01
[4,]             1.0000000000    0.000000e+00
[5,]             0.9876416020    1.235840e-02
[6,]             1.0000000000    0.000000e+00
```

```
[7,]                 0.2762650840        7.237349e-01
[8,]                 1.0000000000        0.000000e+00
[9,]                 1.0000000000        0.000000e+00
[10,]                1.0000000000        0.000000e+00
```

The dce-GMDH algorithm predicts that the probability of benign for the first and second persons are 95.7% and 83.2%, respectively. Since the predicted probability of benign is greater than the predicted probability of malignant, these persons are classified as benign. Confusion matrix and related statistics are obtained through the following codes to investigate the performance measures for the test set.

```
# obtain predicted classes for test set
R> y.test_pred <- predict(model, x.test, type = "class")

# obtain confusion matrix and some statistics for test set
R> confMat(y.test_pred, y.test, positive = "malignant")

Confusion Matrix and Statistics

                     reference
data                 malignant     benign
malignant            54            1
benign               2             79

Accuracy                    :      0.9779
No Information Rate         :      0.5882
Kappa                       :      0.9543
Matthews Corr Coef          :      0.9545
Sensitivity                 :      0.9643
Specificity                 :      0.9875
Positive Pred Value         :      0.9818
Negative Pred Value         :      0.9753
Prevalence                  :      0.4118
Balanced Accuracy           :      0.9759
Youden Index                :      0.9518
Detection Rate              :      0.3971
Detection Prevalence        :      0.4044
Precision                   :      0.9818
Recall                      :      0.9643
F1                          :      0.973

Positive Class              :      malignant
```

Accuracy rate of dce-GMDH algorithm is estimated to be 0.9779. This algorithm classifies 97.79% of persons in a correct class. Moreover, sensitivity and specificity are calculated as 0.9643 and 0.9875. The algortihm correctly classifies 96.43% of the persons having breast cancer, 98.75% of the persons not having breast cancer.

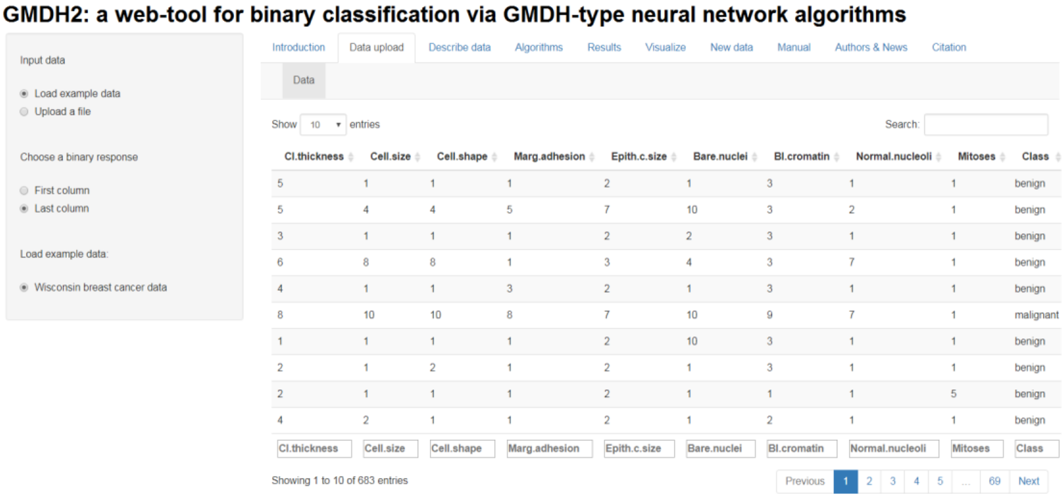
All in all, using dce-GMDH algorithm increases the classification performance approximately 2% in accuracy compared to GMDH algorithm for this data set.

5. GMDH2 WEB-INTERFACE

In the previous chapter, we introduce the GMDH2 package. The purpose of the package is to perform binary classification via GMDH-type neural network algorithms. This package presents two main algorithms, GMDH algorithm and dce-GMDH algorithm. GMDH algorithm performs binary classification and returns the variables dominating the system. dce-GMDH algorithm performs binary classification by assembling classifiers depending on GMDH algorithm. Moreover, the package provides a well-formatted table of descriptives in different format (R, LaTeX, HTML). Also, it produces confusion matrix, its related statistics and scatter plot (2D and 3D) with classification labels of binary classes to assess the contribution of the variables on the prediction performance. It is sometimes difficult for applied researchers to deal with R codes. Therefore, a web interface of GMDH2 package is developed by using shiny package (41). This web-interface is available at http://www.softmed.hacettepe.edu.tr/GMDH2.

In this section, we demonstrate the usage of the GMDH2 web-interface for especially non-R user and applied researchers. The web-interface includes ten tab panels – introduction, data upload, describe data, algorithms, results, visualize, new data, manual, authors & news, citation. In introduction tab panel, we give some general information on GMDH algorithms and the features of the tool.

In data upload tab panel, researchers can upload their data to the tool (Figure 5.1). The file including the data has to be text file in which the deliminater of the columns can be comma, tab, semicolon, or space. Also, the first row of the data has to be the header. Two-class response variable can be the first or the last column of the data. Moreover, we include Wisconsin breast cancer dataset on this tab for the researchers to test the tool.



GMDH2: a web-tool for binary classification via GMDH-type neural network algorithms

Input data

● Load example data
○ Upload a file

Choose a binary response

○ First column
● Last column

Load example data:

● Wisconsin breast cancer data

Show 10 entries

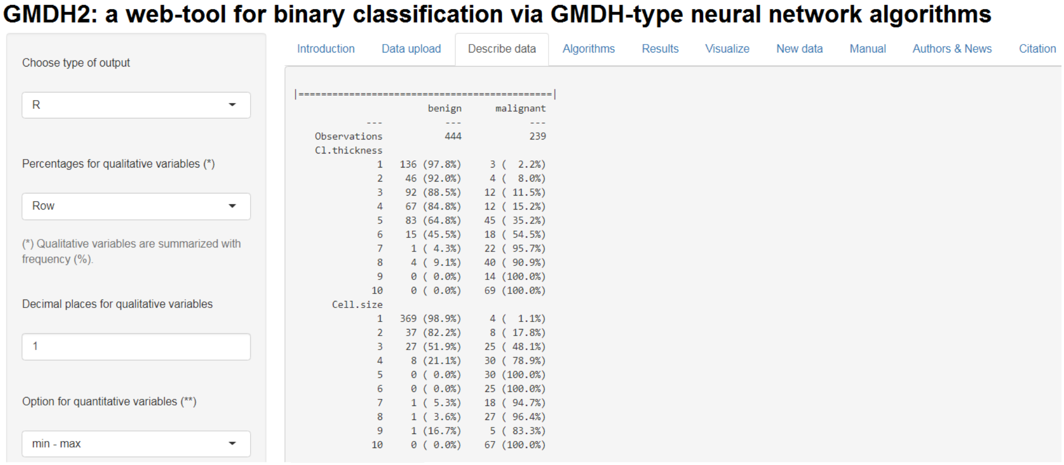
Search:

Cl.thickness	Cell.size	Cell.shape	Marg.adhesion	Epith.c.size	Bare.nuclei	Bl.cromatin	Normal.nucleoli	Mitoses	Class
5	1	1	1	2	1	3	1	1	benign
5	4	4	5	7	10	3	2	1	benign
3	1	1	1	2	2	3	1	1	benign
6	8	8	1	3	4	3	7	1	benign
4	1	1	3	2	1	3	1	1	benign
8	10	10	8	7	10	9	7	1	malignant
1	1	1	1	2	10	3	1	1	benign
2	1	2	1	2	1	3	1	1	benign
2	1	1	1	2	1	1	1	5	benign
4	2	1	1	2	1	2	1	1	benign
Cl.thickness	Cell.size	Cell.shape	Marg.adhesion	Epith.c.size	Bare.nuclei	Bl.cromatin	Normal.nucleoli	Mitoses	Class

Showing 1 to 10 of 683 entries

Figure 5.1. Web interface of GMDH2 package – Data upload

Researchers can obtain basic descriptive statistics via describe data tab (Figure 5.2). In this tab, we organize the output as a table format. For quantitative variables, mean ± standard deviation (median, minimum - maximum) or mean ± standard deviation (median, Quartile1 - Quartile3) are reported as desired. For qualitative variables, the statistics are documentated as frequency (percentage). Decimals of the statistics are able to be set via this tab panel. All these statistics can be obtained in different formats (R, LaTeX, HTML).



GMDH2: a web-tool for binary classification via GMDH-type neural network algorithms

Choose type of output

R

Percentages for qualitative variables (*)

Row

(*) Qualitative variables are summarized with frequency (%).

Decimal places for qualitative variables

1

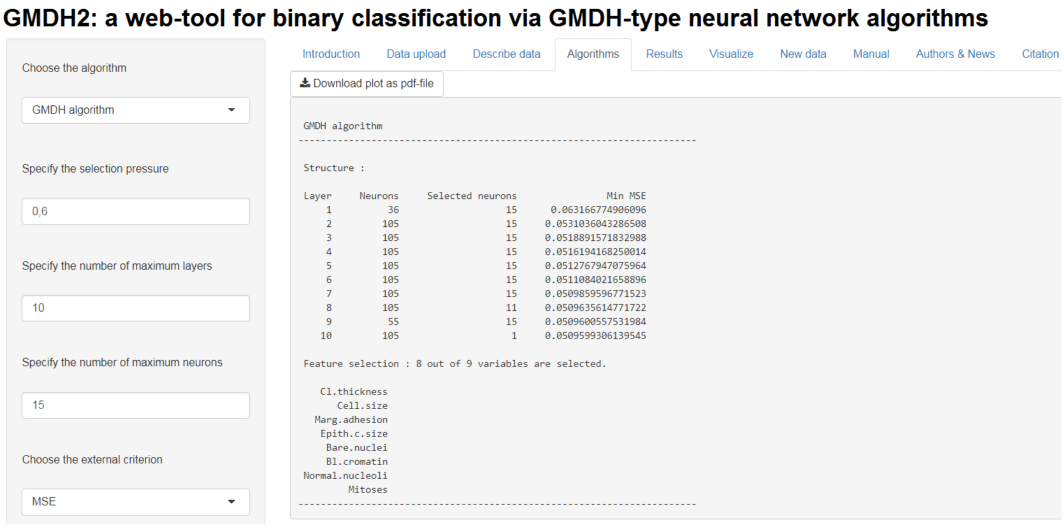
Option for quantitative variables (**)

min - max

```
|============================================|
                          benign      malignant
              ---            ---            ---
     Observations            444            239
     Cl.thickness
                1   136 (97.8%)     3 (   2.2%)
                2    46 (92.0%)     4 (   8.0%)
                3    92 (88.5%)    12 (  11.5%)
                4    67 (84.8%)    12 (  15.2%)
                5    83 (64.8%)    45 (  35.2%)
                6    15 (45.5%)    18 (  54.5%)
                7     1 ( 4.3%)    22 (  95.7%)
                8     4 ( 9.1%)    40 (  90.9%)
                9     0 ( 0.0%)    14 (100.0%)
               10     0 ( 0.0%)    69 (100.0%)
        Cell.size
                1   369 (98.9%)     4 (   1.1%)
                2    37 (82.2%)     8 (  17.8%)
                3    27 (51.9%)    25 (  48.1%)
                4     8 (21.1%)    30 (  78.9%)
                5     0 ( 0.0%)    30 (100.0%)
                6     0 ( 0.0%)    25 (100.0%)
                7     1 ( 5.3%)    18 (  94.7%)
                8     1 ( 3.6%)    27 (  96.4%)
                9     1 (16.7%)     5 (  83.3%)
               10     0 ( 0.0%)    67 (100.0%)
```

Figure 5.2. Web interface of GMDH2 package – Describe data

After describing the data, researchers can specify the algorithm desired through Algorithms tab (Figure 5.3). In this tab, there exist two main algorithms, GMDH and dce-GMDH algorithms. In this tab, it is possible to change the selection pressure (defaults to 0.6). Also, there exist panels to specify the number of maximum layers (default is set to 10), the number of maximum neurons (default is set to 15). Moreover, there exist two options to select the external criteria; namely, mean square error (MSE) and mean absoluate error (MAE) (default is set to MSE).



GMDH2: a web-tool for binary classification via GMDH-type neural network algorithms

Choose the algorithm: GMDH algorithm

Specify the selection pressure: 0,6

Specify the number of maximum layers: 10

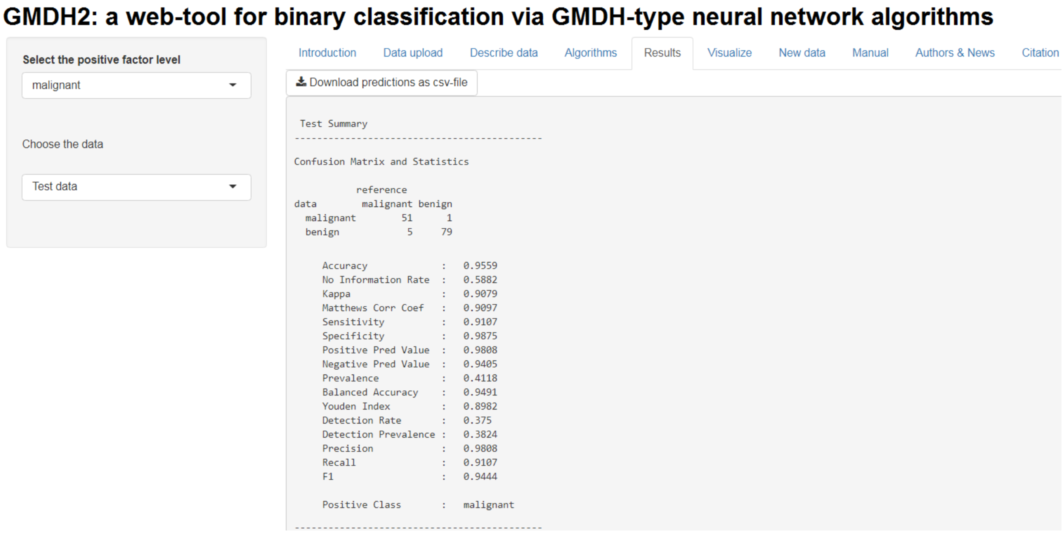
Specify the number of maximum neurons: 15

Choose the external criterion: MSE

```
GMDH algorithm
----------------------------------------------------------------------

Structure :

Layer     Neurons     Selected neurons                 Min MSE
    1          36                   15       0.063166774906096
    2         105                   15      0.0531036043286508
    3         105                   15      0.0518891571832988
    4         105                   15      0.0516194168250014
    5         105                   15      0.0512767947075964
    6         105                   15      0.0511084021658896
    7         105                   15      0.0509859596771523
    8         105                   11      0.0509635614771722
    9          55                   15      0.0509600557531984
   10         105                    1      0.0509599306139545

Feature selection : 8 out of 9 variables are selected.

   Cl.thickness
      Cell.size
  Marg.adhesion
   Epith.c.size
    Bare.nuclei
    Bl.cromatin
Normal.nucleoli
        Mitoses
----------------------------------------------------------------------
```

Figure 5.3. Web interface of GMDH2 package – Algorithms

Researchers can obtain the performance measures of classification through Results tab (Figure 5.4). It is possible to define the positive factor level in this tab. Also, there is an option to obtain the performance measures of classification for train, validation and test sets. Moreover, there exists an download button to download the predicted probabilities and classes.



GMDH2: a web-tool for binary classification via GMDH-type neural network algorithms

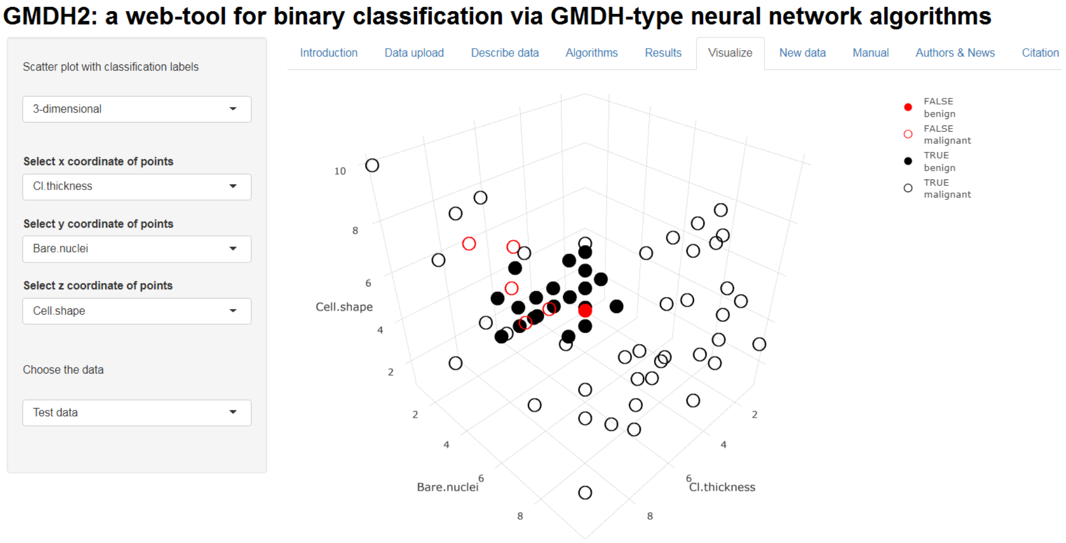
Select the positive factor level: malignant

Choose the data: Test data

```
 Test Summary
--------------------------------------------

Confusion Matrix and Statistics

           reference
data        malignant benign
  malignant        51      1
  benign            5     79

     Accuracy             :   0.9559
     No Information Rate  :   0.5882
     Kappa                :   0.9079
     Matthews Corr Coef   :   0.9097
     Sensitivity          :   0.9107
     Specificity          :   0.9875
     Positive Pred Value  :   0.9808
     Negative Pred Value  :   0.9405
     Prevalence           :   0.4118
     Balanced Accuracy    :   0.9491
     Youden Index         :   0.8982
     Detection Rate       :   0.375
     Detection Prevalence :   0.3824
     Precision            :   0.9808
     Recall               :   0.9107
     F1                   :   0.9444

     Positive Class       :   malignant

--------------------------------------------
```

Figure 5.4. Web interface of GMDH2 package – Results

Researchers can examine the interactive scatter plots with classification labels (Figures 4.2-3) via Visualize tab (Figure 5.5). There exist an option to draw interactive scatter plot in 2-dimensional or 3-dimensional. It is necessary to specify the coordinates of the graphic. These interactive scatter plots can be drawn for train, validation and test sets.



GMDH2: a web-tool for binary classification via GMDH-type neural network algorithms

Scatter plot with classification labels: 3-dimensional

Select x coordinate of points: Cl.thickness

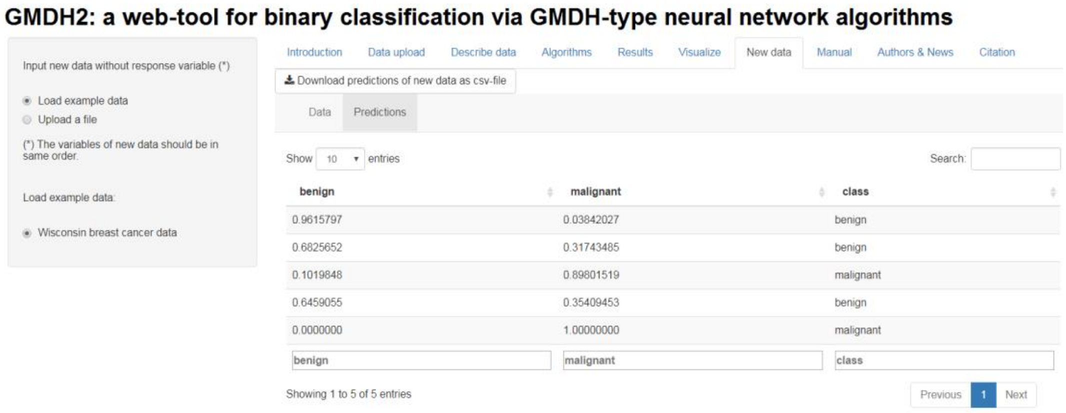
Select y coordinate of points: Bare.nuclei

Select z coordinate of points: Cell.shape

Choose the data: Test data

Figure 5.5. Web interface of GMDH2 package – Visualize

At last, researchers can upload new data, obtain predicted probabilities and classes through New data tab (Figure 5.6). Also, these predictions can be downloaded via download button in this tab panel. New data have to be inputted to the tool without the response variable. Also, the variables of new data have to be in same order with the data inputted in Data upload tab panel.



GMDH2: a web-tool for binary classification via GMDH-type neural network algorithms

Input new data without response variable (*)

● Load example data
○ Upload a file

(*) The variables of new data should be in same order.

Load example data:

● Wisconsin breast cancer data

benign	malignant	class
0.9615797	0.03842027	benign
0.6825652	0.31743485	benign
0.1019848	0.89801519	malignant
0.6459055	0.35409453	benign
0.0000000	1.00000000	malignant
benign	malignant	class

Showing 1 to 5 of 5 entries

Figure 5.6. Web interface of GMDH2 package – New data

In manual tab panel, we give some information on usage of web-interface. It is important to note that if there are missing values in the data, a listwise deletion will be applied and a complete-case analysis will be performed. The seed number is fixed to 12345 for reproducibility. The data are divided into three sets; train (60%), validation (20%) and test (20%) sets.

In authors & news tab panel, we give some information of authors and news for updates. In citation tab panel, the citation information of the tool is stated.

6. SIMULATION STUDY

In this chapter, the objective is to compare the performances of GMDH and dce-GMDH algorithms with support vector machines, random forest, naive bayes, elastic net logistic regression, artificial neural network, and give some general suggestions on which classifier(s) should be used or avoided under different conditions.

A Monte Carlo simulation study is conducted to investigate the effect of several conditions. The data are simulated under 216 different scenarios. The datasets include all possible combinations of the followings:

- Proportion of positives (pp) changing as 0.3, 0.5;
- number of exploratory variables (p) changing as 5, 10, 15;
- sample sizes (n) changing as 50, 100, 500, 1000;
- correlations between response and exploratory variables (ρ_{y,x_i}) changing as 0.2 - 0.3 (Low), 0.5 - 0.6 (Medium), 0.8 - 0.9 (High);
- correlations between exploratory variables (ρ_{x_i,x_j}) changing as 0 - 0.1 (Low), 0.4 - 0.5 (Medium), 0.8 - 0.9 (High).

Datasets are simulated using the jointly.generate.binary.normal function in the BinNor package (42) in R and manipulated based on the details given above. Exploratory variables are simulated in different variable types; binary (40%) and continuous (60%) variables.

In simulation study, the performance of classifiers are investigated through accuracy, sensitivity, specificity, positive predictive value, negative predictive value, balanced accuracy, F1 measure based on the confusion matrices of true and predicted classes for test sets. Simulation scenarios are repeated 10,000 times. In simulation scenarios, the seed number is fixed to ‘12345’ for reproducibility. All scenarios are summarized with accuracy rates and presented in Figures 6.1-6. A portion of the simulation results is stated in Table 6.1 to protect the content integrity. The rest of the simulation results is presented in Tables A.1-17 given in appendix.

The overall performance of each classifier increases as the sample size increases, as expected, since the classifiers need more observations to better learn from data. The pp changing from 0.3 to 0.5 does not have a serious effect on the classification performances. The number of exploratory variables does not have a severe impact on the performance of classifiers in most scenarios. However, as the number of variables gets larger, the accuracy rates of classifiers increase when the correlations among the exploratory variables (ρ_{x_i,x_j}) and the correlations between the response and exploratory variables (ρ_{y,x_i}) are low (Figures 6.1 and 6.4). Accuracy rates are increasing overall as the level of ρ_{y,x_i} increases. Accuracy rates range between 0.65 and 1.00 where the level of ρ_{y,x_i} is high, while they are between 0.50 and 0.80 for low ρ_{y,x_i}. The differences in the accuracies between medium and high ρ_{y,x_i} are more evident when the level of ρ_{x_i,x_j} is medium or high (Figures 6.2-3 and 6.5-6). Moreover, the accuracy rates are similar for different levels of ρ_{x_i,x_j}, with only a slightly increase for medium ρ_{x_i,x_j} (Figures 6.2 and 6.5).

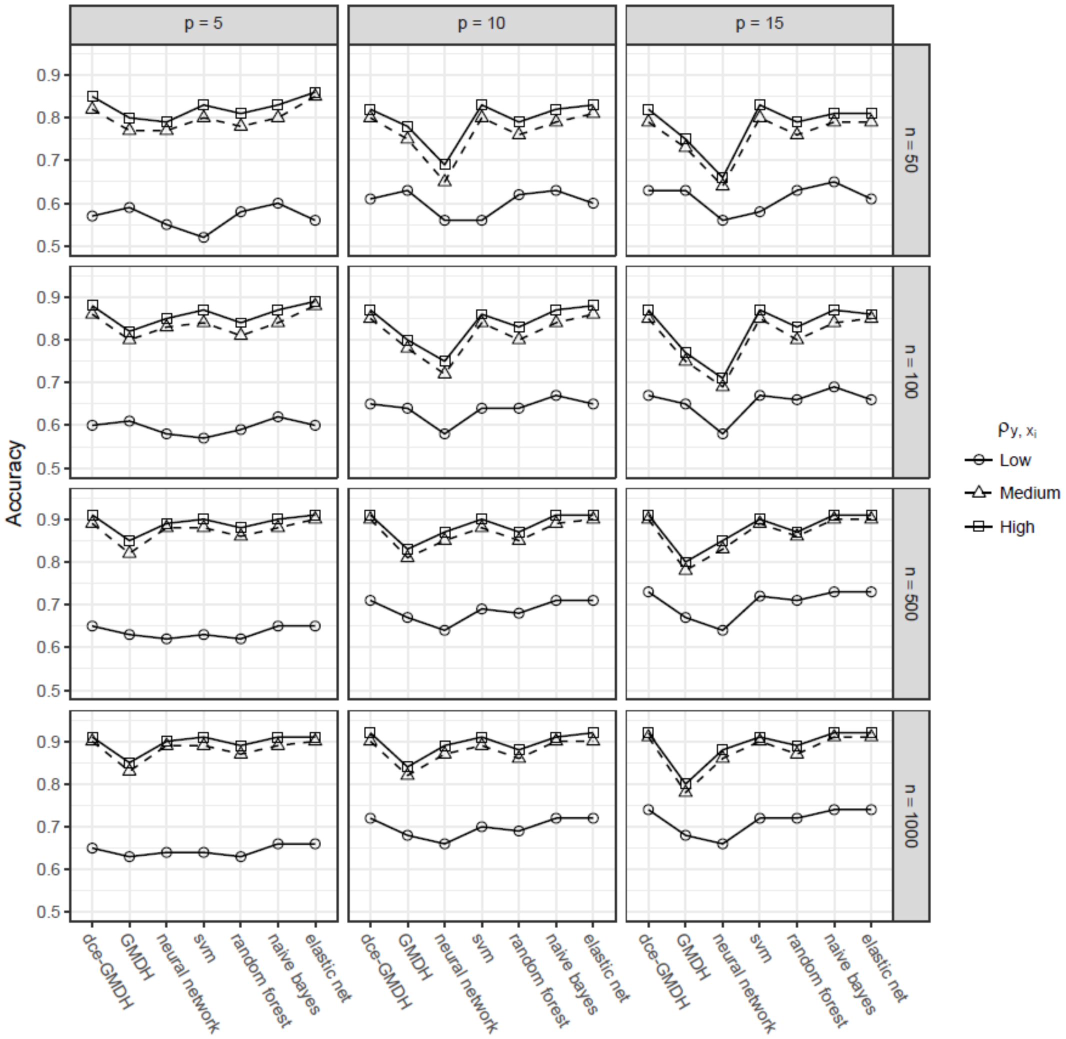


Figure 6.1. Accuracy rates of classifiers when ρ_{x_i,x_j} are low and pp is 0.5

When the level of ρ_{x_i,x_j} is low (Figures 6.1 and 6.4), dce-GMDH algorithm and elastic net logistic regression are the two competing and best classifiers under most sample sizes, exploratory variable numbers and correlation combinations. For example, the accuracies of dce-GMDH and elastic net logistic regression are estimated to be 0.91 under the scenario with high level of ρ_{y,x_i}, large n (n = 500), small p (p = 5), balanced pp (pp = 0.5) in Table 6.1. Under the same scenario, the accuracies of the other classifiers change between 0.85 and 0.90. The other performance measures are similar to the accuracy since the pp is equal to 0.5. GMDH algorithm and naive Bayes classifiers are performing particularly well under small n (n = 50), low level of ρ_{y,x_i} and pp = 0.5. For instance, the accuracies of GMDH algorithm and naive Bayes are estimated to be 0.59 and 0.60 respectively under the scenario with low level of ρ_{y,x_i}, small n (n = 50), small p (p = 5), balanced pp (pp = 0.5) in Table 6.1. Under the same scenario, the accuracies of the other classifiers vary between 0.52 and 0.58. Under small n (n = 50), low level of ρ_{y,x_i} and pp = 0.3, GMDH algorithm, support vector machine and elastic net logistic regression are performing particularly well.

As it can be seen from Figures 6.1 and 6.4, accuracy of artificial neural network is the lowest for small n ($n \leq 100$), medium and large p ($p \geq 10$). For larger sample sizes, this method also yields the worst result if the level of $\rho_{\mathrm{y,x_i}}$ is low.

Support vector machine is one of the best classifiers for all other scenarios although it gives the lowest accuracy for small n (n= 50), small and medium p ($p \leq 10$), low level of ρ_{y,x_i}, low level of $\rho_{\mathrm{x_i,x_j}}$ and pp = 0.5 (Figure 1). When the level of $\rho_{\mathrm{x_i,x_j}}$ is medium or high (Figures 6.2 and 6.3), similar patterns are observed as the ones observed in Figure 6.1. Support vector machine also gives the lowest accuracy for small n (n = 50), low level of ρ_{y,x_i} and pp = 0.5.

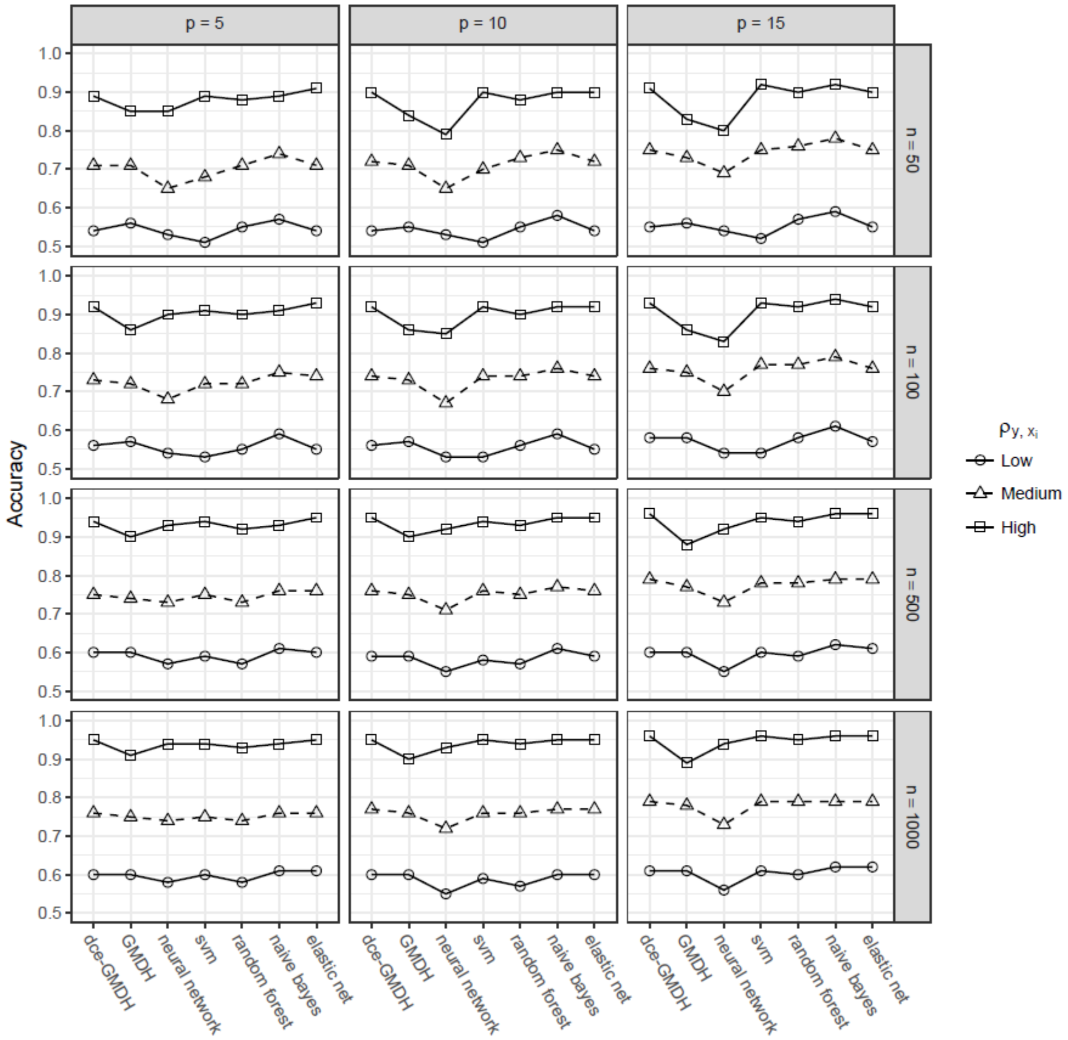


Figure 6.2. Accuracy rates of classifiers when ρ_{x_i,x_j} are medium and pp is 0.5

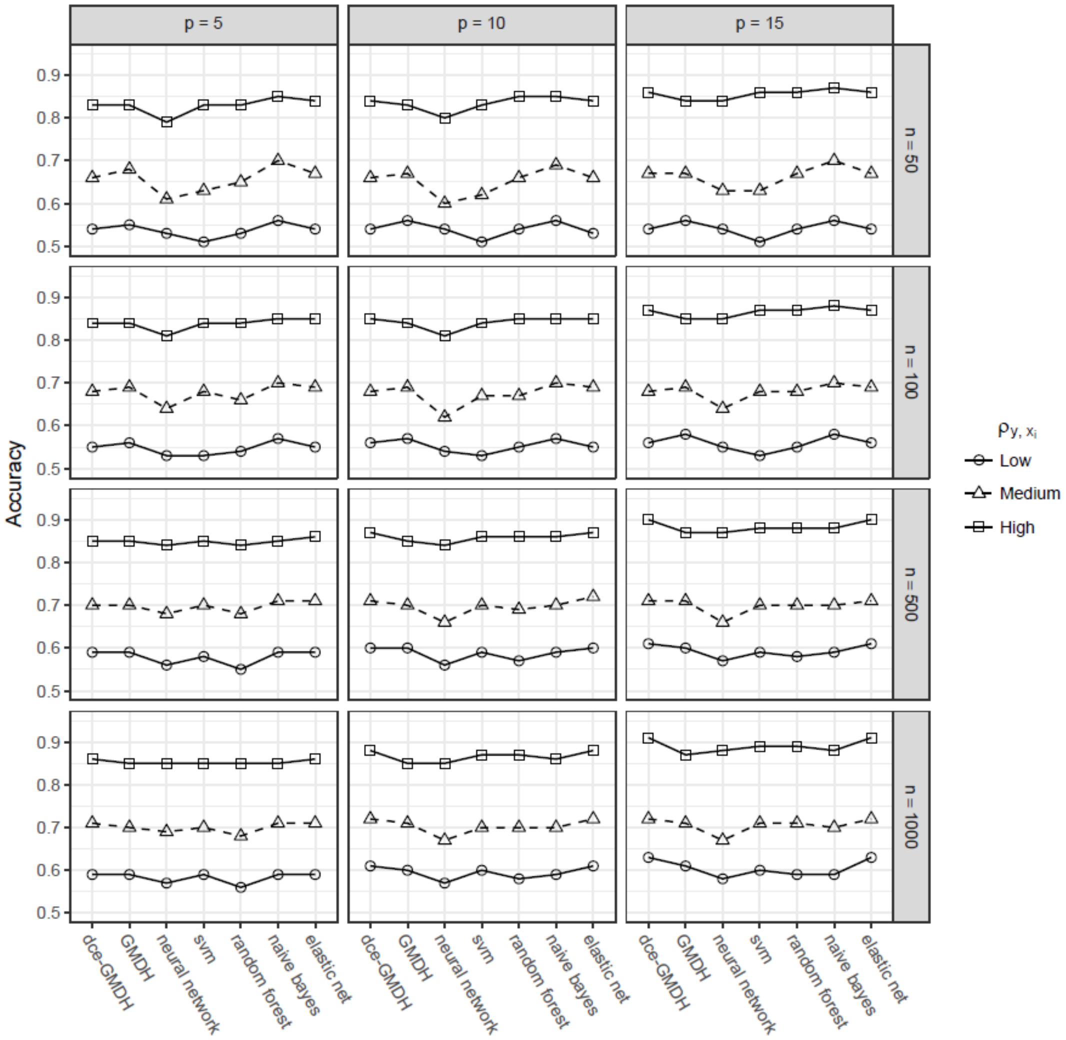


Figure 6.3. Accuracy rates of classifiers when ρ_{x_i,x_j} are high and pp is 0.5

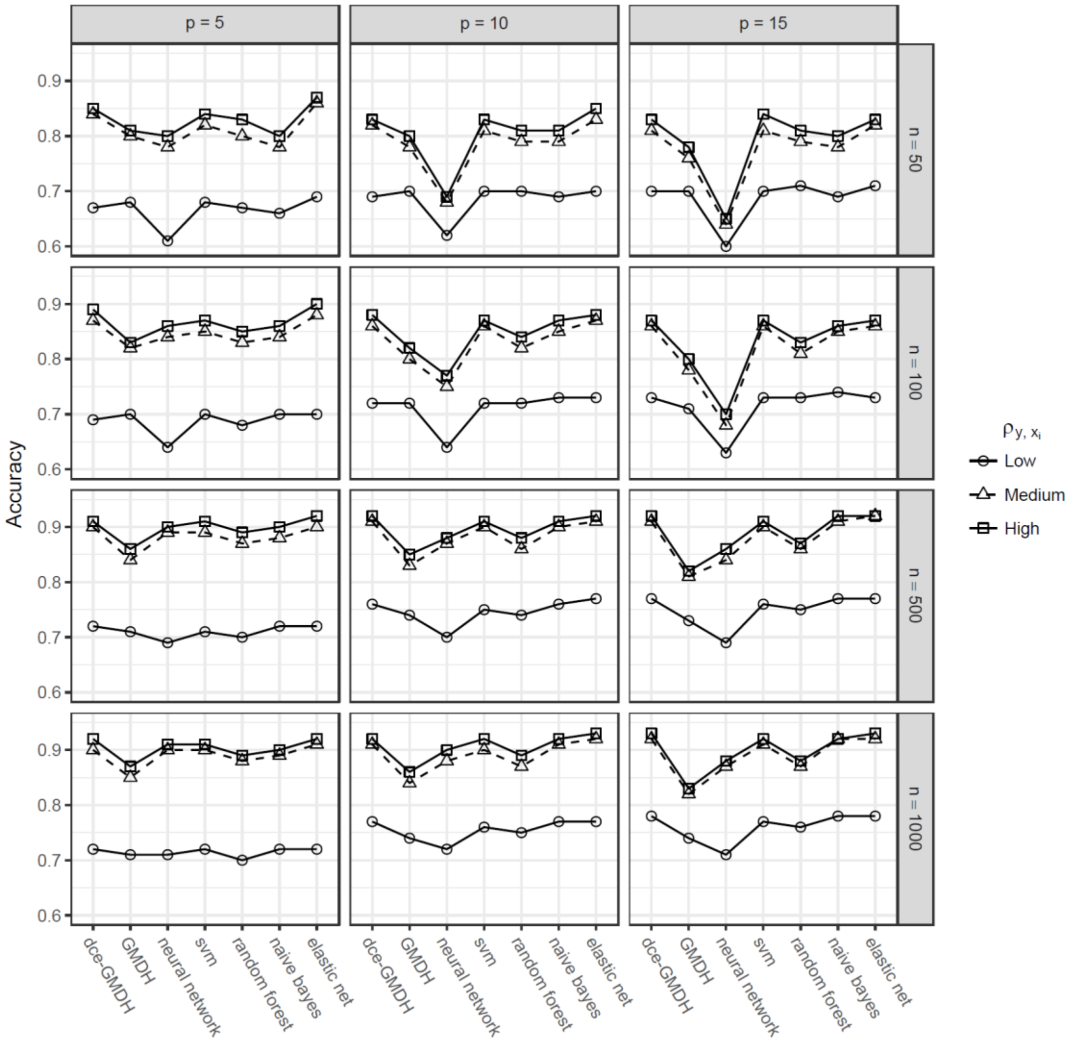


Figure 6.4. Accuracy rates of classifiers when ρ_{x_i,x_j} are low and pp is 0.3

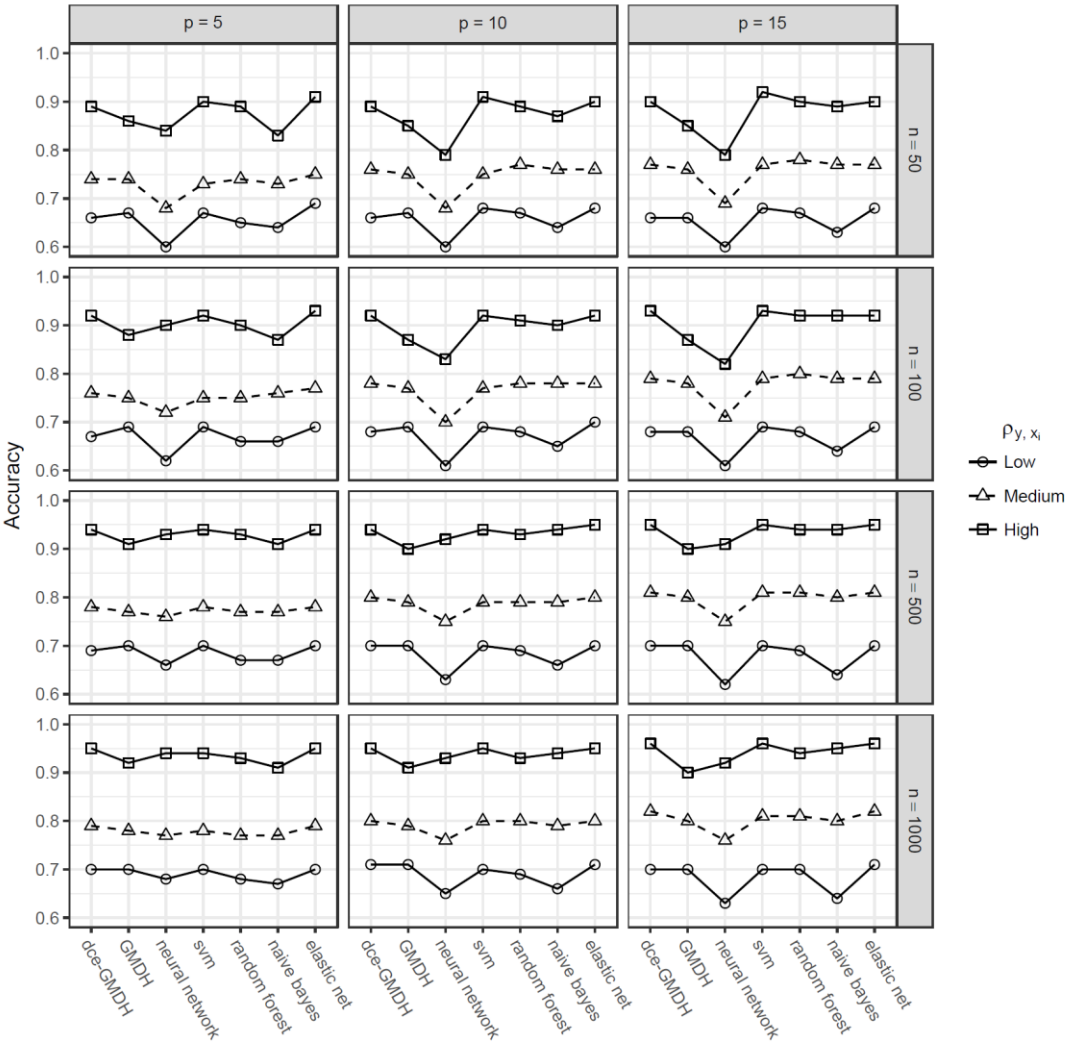


Figure 6.5. Accuracy rates of classifiers when ρ_{x_i,x_j} are medium and pp is 0.3

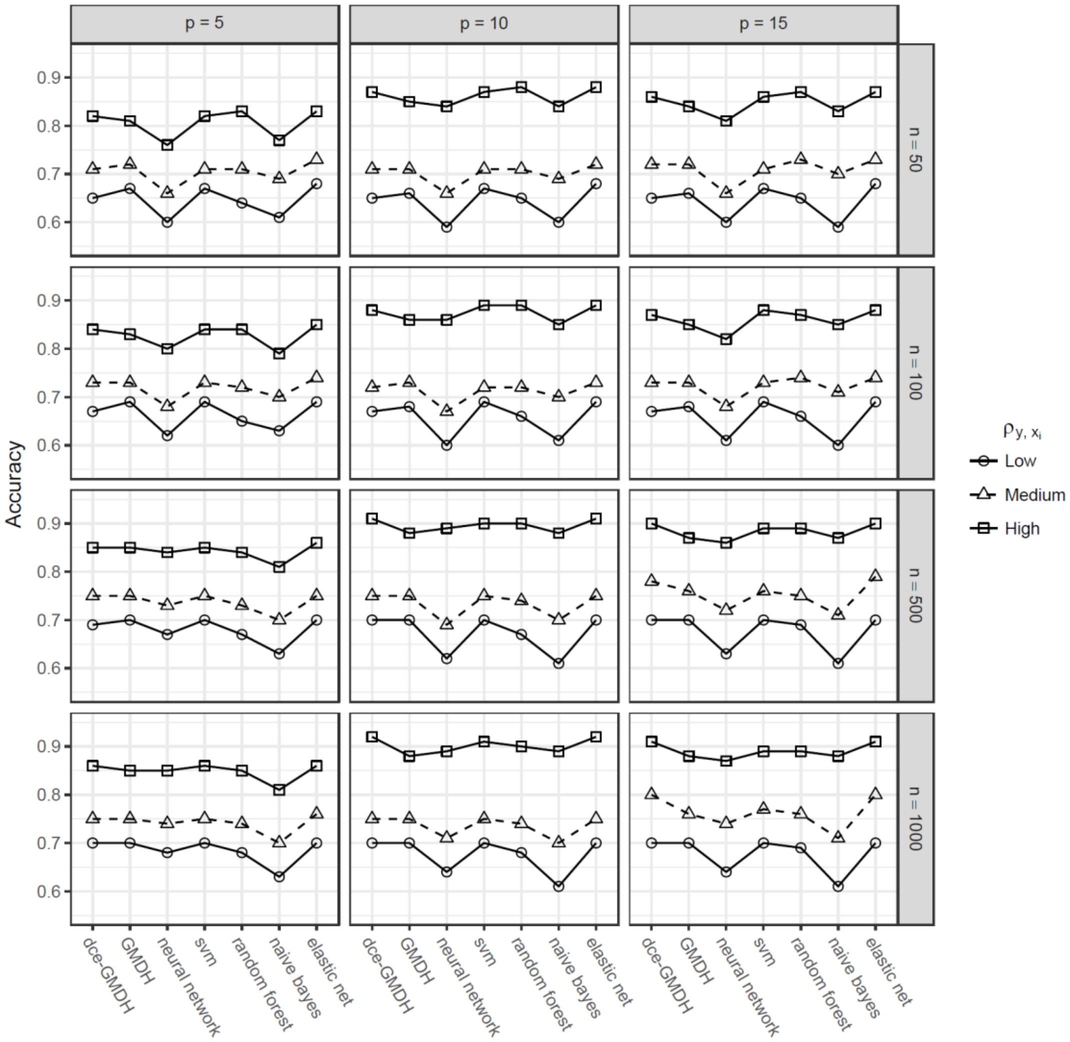


Figure 6.6. Accuracy rates of classifiers when ρ_{x_i,x_j} are high and pp is 0.3

Table 6.1. Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 5 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.59	0.59	0.58	0.59	0.60	0.58	0.56
		dce-GMDH	0.57	0.58	0.56	0.58	0.58	0.57	0.55
		svm	0.52	0.52	0.53	0.53	0.54	0.53	0.52
		random forest	0.58	0.58	0.57	0.58	0.59	0.58	0.55
		naive bayes	0.60	0.60	0.59	0.60	0.61	0.59	0.57
		elastic net	0.56	0.59	0.54	0.58	0.59	0.56	0.59
		neural network	0.55	0.56	0.54	0.55	0.56	0.55	0.52
	100	GMDH	0.61	0.62	0.59	0.61	0.62	0.60	0.59
		dce-GMDH	0.60	0.61	0.59	0.61	0.61	0.60	0.59
		svm	0.57	0.57	0.56	0.58	0.59	0.57	0.56
		random forest	0.59	0.60	0.59	0.60	0.60	0.59	0.58
		naive bayes	0.62	0.63	0.61	0.62	0.63	0.62	0.61
		elastic net	0.60	0.61	0.58	0.61	0.62	0.60	0.61
		neural network	0.58	0.60	0.55	0.58	0.59	0.58	0.57
	500	GMDH	0.63	0.63	0.62	0.63	0.63	0.63	0.62
		dce-GMDH	0.65	0.65	0.64	0.65	0.65	0.65	0.65
		svm	0.63	0.63	0.63	0.64	0.64	0.63	0.63
		random forest	0.62	0.62	0.61	0.62	0.62	0.62	0.61
		naive bayes	0.65	0.65	0.65	0.65	0.65	0.65	0.65
		elastic net	0.65	0.65	0.65	0.65	0.65	0.65	0.65
		neural network	0.62	0.64	0.60	0.62	0.63	0.62	0.62
	1000	GMDH	0.63	0.64	0.62	0.63	0.63	0.63	0.63
		dce-GMDH	0.65	0.66	0.65	0.65	0.66	0.65	0.65
		svm	0.64	0.64	0.64	0.64	0.65	0.64	0.64
		random forest	0.63	0.63	0.62	0.63	0.63	0.63	0.63
		naive bayes	0.66	0.66	0.65	0.66	0.66	0.66	0.65
		elastic net	0.66	0.66	0.65	0.66	0.66	0.66	0.66
		neural network	0.64	0.65	0.62	0.63	0.64	0.64	0.64
Medium	50	GMDH	0.77	0.78	0.77	0.78	0.79	0.77	0.76
		dce-GMDH	0.82	0.83	0.82	0.83	0.83	0.82	0.81
		svm	0.80	0.80	0.80	0.82	0.82	0.80	0.78
		random forest	0.78	0.78	0.77	0.79	0.79	0.78	0.76
		naive bayes	0.80	0.80	0.80	0.81	0.81	0.80	0.78
		elastic net	0.85	0.85	0.85	0.85	0.86	0.85	0.84
		neural network	0.77	0.77	0.76	0.77	0.78	0.77	0.75
	100	GMDH	0.80	0.80	0.79	0.80	0.80	0.80	0.79
		dce-GMDH	0.86	0.86	0.86	0.87	0.87	0.86	0.86
		svm	0.84	0.84	0.85	0.85	0.85	0.84	0.84
		random forest	0.81	0.81	0.81	0.82	0.82	0.81	0.80
		naive bayes	0.84	0.83	0.85	0.85	0.84	0.84	0.83
		elastic net	0.88	0.88	0.88	0.88	0.88	0.88	0.87
		neural network	0.83	0.82	0.83	0.84	0.83	0.83	0.82
	500	GMDH	0.82	0.83	0.82	0.82	0.83	0.82	0.82
		dce-GMDH	0.89	0.89	0.89	0.89	0.89	0.89	0.89
		svm	0.88	0.88	0.88	0.88	0.88	0.88	0.88
		random forest	0.86	0.86	0.86	0.86	0.86	0.86	0.86
		naive bayes	0.88	0.88	0.89	0.89	0.88	0.88	0.88
		elastic net	0.90	0.90	0.89	0.90	0.90	0.90	0.89
		neural network	0.88	0.88	0.88	0.88	0.88	0.88	0.88
	1000	GMDH	0.83	0.83	0.82	0.83	0.83	0.83	0.83
		dce-GMDH	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		svm	0.89	0.89	0.89	0.89	0.89	0.89	0.89
		random forest	0.87	0.87	0.87	0.87	0.87	0.87	0.87
		naive bayes	0.89	0.89	0.90	0.90	0.89	0.89	0.89
		elastic net	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		neural network	0.89	0.89	0.89	0.89	0.89	0.89	0.89

Table 6.1. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 5 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.80	0.80	0.79	0.80	0.81	0.80	0.78
		dce-GMDH	0.85	0.85	0.85	0.85	0.85	0.85	0.83
		svm	0.83	0.83	0.84	0.85	0.84	0.83	0.82
		random forest	0.81	0.81	0.81	0.82	0.82	0.81	0.80
		naive bayes	0.83	0.83	0.83	0.84	0.84	0.83	0.82
		elastic net	0.86	0.87	0.86	0.87	0.87	0.86	0.85
		neural network	0.79	0.80	0.79	0.80	0.80	0.79	0.78
	100	GMDH	0.82	0.82	0.81	0.81	0.83	0.82	0.81
		dce-GMDH	0.88	0.88	0.88	0.88	0.88	0.88	0.87
		svm	0.87	0.86	0.87	0.87	0.87	0.87	0.86
		random forest	0.84	0.84	0.84	0.84	0.84	0.84	0.83
		naive bayes	0.87	0.86	0.88	0.88	0.87	0.87	0.86
		elastic net	0.89	0.89	0.89	0.89	0.89	0.89	0.89
		neural network	0.85	0.84	0.85	0.85	0.85	0.85	0.84
	500	GMDH	0.85	0.85	0.84	0.84	0.85	0.85	0.85
		dce-GMDH	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		svm	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		random forest	0.88	0.88	0.88	0.88	0.88	0.88	0.88
		naive bayes	0.90	0.90	0.91	0.91	0.90	0.90	0.90
		elastic net	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		neural network	0.89	0.89	0.89	0.89	0.89	0.89	0.89
	1000	GMDH	0.85	0.86	0.85	0.85	0.86	0.85	0.85
		dce-GMDH	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		svm	0.91	0.90	0.91	0.91	0.91	0.91	0.91
		random forest	0.89	0.89	0.89	0.89	0.89	0.89	0.89
		naive bayes	0.91	0.90	0.91	0.91	0.90	0.91	0.91
		elastic net	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		neural network	0.90	0.90	0.90	0.90	0.90	0.90	0.90

7. DISCUSSION AND CONCLUSION

Binary classification is a problem in which binary factor labels can be predicted for each observation. Binary classification is used in different disciplines. Examples include medical studies, economics, agriculture, meteorology, and so on. In this thesis, we propose a new classifier for a binary response based on GMDH-type neural network algorithm. We name this classifier as diverse classifiers ensemble based on GMDH (dce-GMDH) algorithm. Also, we present GMDH algorithm for binary classification and develop an R package, **GMDH2**, for the availability of these classifiers. **GMDH2** package is publicly available on CRAN. The researchers over the world are able to reach these algorithms at https://CRAN.R-project.org/package=GMDH2. Moreover, we develop a web application of the package for especially non-R user researchers. This application is available at http://softmed.hacettepe.edu.tr/GMDH2.

In this thesis, we present **GMDH2** package to perform binary classification through GMDH-type neural network algorithms. The GMDH2 package offers two main algorithms; namely, GMDH and dce-GMDH algorithms. GMDH algorithm makes binary classification and determines which features are important for discrimination of classes. dce-GMDH algorithm assembles the classifiers - support vector machines, random forest, naive Bayes, elastic net logistic regression, artificial neural network - based on GMDH algorithm to perform classification for a binary response. Moreover, the package provides a table of descriptives for a binary factor in different formats (R, LaTeX, HTML). The package also produces confusion matrix, its related statistics and scatter plot (2D and 3D) with classification labels of binary classes to assess the prediction performance. All features of the package are demonstrated on Wisconsin breast cancer dataset. The package and its web-interface will be updated regularly.

In this study, we compared GMDH and dce-GMDH algorithms to support vector machines, random forest, naive Bayes, elastic net logistic regression, artificial neural network with a Monte Carlo simulation. In the light of this simulation study, dce-GMDH algorithm, elastic net logistic regression and support vector machine are the three competing classifiers under most sample size, feature number and correlation combinations. However, support vector machine usually gives the lowest accuracy when the sample size is small, the correlation between the response and covariates is low and the proportion of positives is balanced. Under small sample sizes and low level of correlations between response and exploratory variables, GMDH algorithm and naive Bayes classifiers are performing particularly well when the proportion of positives is balanced, GMDH algorithm, support vector machine and elastic net logistic regression are performing well when the proportion of positives is unbalanced. To sum up, the use of dce-GMDH algorithm seems to be beneficial, since it performs well under almost all scenerios and takes advantage from other classifiers when needed.

Future studies are planned in the direction of classification for ordinal and multinomial response variable. The algorithms presented in this paper will be organized for this type of variable. Monte Carlo simulation will be conducted to illustrate the performance comparison of these classifiers to the other well-known classifiers. Moreover, these algorithms can be used for the large number of variables, such as classification of genomics data. With especially GMDH algorithm, selection of important genes can be conducted.

8. REFERENCES

1. Agresti, A. An introduction to categorical data analysis. New York: Wiley; 1996.

2. Klecka, WR. Discriminant analysis. Sage; 1980.

3. Tibshirani, R. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society, Series B (Methodological). 1996:267–288.

4. Zou H, Hastie T. Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2005;67(2):301–320.

5. Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. Journal of Statistical Software. 2010;33(1):1-22.

6. Farlow SJ. The GMDH algorithm of Ivakhnenko. The American Statistician. 1981;35(4):210–215.

7. Meyer D, Dimitriadou E, Hornik K, Weingessel A, Leisch F. e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien; 2015.

8. Venables WN, Ripley BD. Modern applied statistics with S. 4th ed. New York: Springer; 2002.

9. Liaw A, Wiener M. Classification and regression by randomForest. R News. 2002;2(3):18–22.

10. Dag O, Karabulut E, Alpar R. GMDH2: binary classification via GMDH-type neural network algorithms; 2018.

11. Ivakhnenko A. The group method of data handling – a rival of the method of stochastic approximation. Soviet Automatic Control. 1966;13(3):43–55.

12. Ivakhnenko A. Heuristic self-organization in problems of engineering cybernetics. Automatica. 1970;6(2):207–219.

13. Ivakhnenko AG, Ivakhnenko GA. The review of problems solvable by algorithms of the Group Method of Data Handling (GMDH). Pattern Recognition and Image Analysis, 1995;5(4):527-535.

14. Kalavrouziotis I, Stepashko V, Vissikirsky V, Drakatos P. Group Method of Data Handling (GMDH) application for modelling of mechanical properties of trees irrigated with wastewater. International Journal of Environment and Pollution. 2002;18(6):589-601.

15. Nariman-zadeh N, Darvizeh A, Darvizeh M, Gharababaei H. Modelling of explosive cutting process of plates using GMDH-type neural network and singular value decomposition. Journal of Materials Processing Technology. 2002;128:80-87.

16. Astakhov VP, Galitsky, VV. Tool life testing in gundrilling: an application of the Group Method of Data Handling (GMDH). International Journal of Machine Tools & Manufacture. 2005;45:509-517.

17. Srinivasan D. Energy demand prediction using GMDH networks. Neurocomputing. 2008;72(1):625–629.

18. Xu H, Dong Y, Wu J, Zhao W. Application of GMDH to short-term load forecasting. Advances in Intelligent Systems. 2012;138:27–32.

19. Najafzadeh M, Barani G, Hessami Kermani M. Estimation of pipeline scour due to waves by GMDH. Journal of Pipeline Systems Engineering and Practice. 2014;5(3):06014002.

20. Sheikholeslami M, Sheykholeslami FB, Khoshhal S, Mola-Abasia H, Ganji DD, Rokni, HB. Effect of magnetic field on cu–water nanofluid heat transfer using GMDH-type neural network. Neural Computing and Applications. 2014;25:171-178.

21. Antanasijevic D, Antanasijevic J, Pocajt, V, Uscumlic G. A GMDH-type neural network with multi-filter feature selection for the prediction of transition temperatures of bent-core liquid crystals. RSC Advances. 2016;6(102):99676-99684.

22. Xiao J, Jiang X, He C, Teng G. Churn prediction in customer relationship management via gmdh-based multiple classifiers ensemble. IEEE Intelligent Systems. 2016;31(2):37–44.

23. El-Alfy, ESM, Baig ZA, Abdel-Aal RE. A novel approach for face recognition using fused GMDH-based networks. Int. Arab J. Inf. Technol. 2018;15(3):369-377.

24. Guo J, Huang W, Mao Q, Wang X, Wang X, Song T. Modified GMDH networks for oilfield production prediction. Geosystem Engineering. 2018;21(4):217-225.

25. Kondo T. GMDH neural network algorithm using the heuristic self-organization method and its application to the pattern identification problem. Proceedings of the 37th SICE Annual Conference. 1998:1143–1148.

26. Muller JA, Ivachnenko AG, Lemke F. GMDH algorithms for complex systems modelling. Mathematical and Computer Modelling of Dynamical Systems: Methods, Tools and Applications in Engineering and Related Sciences. 1998;4(4):275–316.

27. Kondo T, Ueno J. Medical image recognition of the brain by revised GMDH-type neural network algorithm with a feedback loop. International Journal of Innovative Computing, Information and Control. 2006;2(5):1039–1052.

28. Kondo T, Ueno J. Revised GMDH-type neural network algorithm with a feedback loop identifying sigmoid function neural network. International Journal of Innovative Computing, Information and Control. 2006;2(5):985–996.

29. Kondo T, Ueno J. Feedback GMDH-type neural network and its application to medical image analysis of liver cancer. 42th ISCIE international symposium on stochastic systems theory and its applications. 2012:81–82.

30. Dag O, Yozgatligil C. GMDH: an R package for short term forecasting via GMDH-type neural network algorithms. The R Journal. 2016;8(1):379–386.

31. Abdel-Aal R. GMDH-based feature ranking and selection for improved classification of medical data. Journal of Biomedical Informatics. 2005;38(6):456–468.

32. El-Alfy ESM, Abdel-Aal RE. Using GMDH-based networks for improved spam detection and email feature analysis. Applied Soft Computing, 2011;11(1):477–488.

33. Baig ZA, Sait SM, Shaheen A. GMDH-based networks for intelligent intrusion detection. Engineering Applications of Artificial Intelligence. 2013;26(7):1731-1740.

34. Joachims T. Making large-scale support vector machine learning practical. In Advances in kernel methods: support vector machines. Cambridge: MIT Press; 1998.

35. Breiman, L. Random Forests. Machine Learning. 2001;45(1):5-32.

36. Verikas A, Vaiciukynas E, Gelzinis A, Parker J, Olsson MC. Electromyographic patterns during golf swing: Activation sequence profiling and prediction of shot effectiveness. Sensors. 2016;16(4):592.

37. Bishop CM. Neural Networks for Pattern Recognition. Oxford: Oxford University Press; 1995.

38. Wolberg WH, Mangasarian OL. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. Proceedings of the national academy of sciences. 1990;87(23):9193–9196.

39. Leisch F, Dimitriadou E. mlbench: machine learning benchmark problems; 2010.

40. Sievert C, Parmer C, Hocking T, Chamberlain S, Ram K, Corvellec M, Despouy P. plotly: Create Interactive Web Graphics via ‘plotly.js’; 2017.

41. Chang W, Cheng J, Allaire J, Xie Y, McPherson J. shiny: web application framework for R; 2017.

42. Amatya A, Demirtas H. BinNor: simultaneous generation of multivariate binary and normal variates; 2016.

9. APPENDICES

Appendix-1: Performance Comparison of the Classifiers under Different Scenarios

Table A.1. Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 10 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.63	0.63	0.62	0.64	0.63	0.62	0.60
		dce-GMDH	0.61	0.61	0.61	0.63	0.62	0.61	0.59
		svm	0.56	0.56	0.56	0.58	0.57	0.56	0.55
		random forest	0.62	0.62	0.62	0.63	0.63	0.62	0.59
		naive bayes	0.63	0.63	0.64	0.65	0.64	0.63	0.61
		elastic net	0.60	0.61	0.58	0.62	0.62	0.60	0.61
		neural network	0.56	0.57	0.55	0.56	0.56	0.56	0.54
	100	GMDH	0.64	0.65	0.64	0.65	0.65	0.64	0.63
		dce-GMDH	0.65	0.65	0.65	0.66	0.66	0.65	0.64
		svm	0.64	0.63	0.64	0.66	0.65	0.64	0.62
		random forest	0.64	0.64	0.64	0.65	0.65	0.64	0.63
		naive bayes	0.67	0.67	0.67	0.67	0.67	0.67	0.65
		elastic net	0.65	0.65	0.64	0.66	0.67	0.65	0.64
		neural network	0.58	0.59	0.58	0.58	0.59	0.58	0.57
	500	GMDH	0.67	0.67	0.67	0.67	0.67	0.67	0.67
		dce-GMDH	0.71	0.71	0.71	0.71	0.71	0.71	0.70
		svm	0.69	0.69	0.69	0.69	0.69	0.69	0.69
		random forest	0.68	0.68	0.68	0.68	0.68	0.68	0.68
		naive bayes	0.71	0.71	0.71	0.71	0.71	0.71	0.71
		elastic net	0.71	0.71	0.71	0.71	0.71	0.71	0.71
		neural network	0.64	0.66	0.63	0.64	0.65	0.64	0.64
	1000	GMDH	0.68	0.68	0.67	0.68	0.68	0.68	0.68
		dce-GMDH	0.72	0.72	0.71	0.72	0.72	0.72	0.71
		svm	0.70	0.70	0.70	0.70	0.70	0.70	0.70
		random forest	0.69	0.69	0.69	0.69	0.69	0.69	0.69
		naive bayes	0.72	0.72	0.72	0.72	0.72	0.72	0.71
		elastic net	0.72	0.72	0.72	0.72	0.72	0.72	0.72
		neural network	0.66	0.68	0.65	0.66	0.67	0.66	0.67
Medium	50	GMDH	0.75	0.76	0.75	0.76	0.76	0.75	0.73
		dce-GMDH	0.80	0.80	0.79	0.80	0.81	0.80	0.78
		svm	0.80	0.80	0.80	0.81	0.81	0.80	0.78
		random forest	0.76	0.76	0.76	0.77	0.78	0.76	0.74
		naive bayes	0.79	0.78	0.79	0.80	0.80	0.79	0.77
		elastic net	0.81	0.81	0.81	0.82	0.82	0.81	0.79
		neural network	0.65	0.67	0.63	0.65	0.67	0.65	0.63
	100	GMDH	0.78	0.78	0.78	0.78	0.78	0.78	0.77
		dce-GMDH	0.85	0.85	0.85	0.85	0.85	0.85	0.84
		svm	0.84	0.84	0.84	0.85	0.84	0.84	0.83
		random forest	0.80	0.80	0.80	0.81	0.80	0.80	0.79
		naive bayes	0.84	0.83	0.84	0.85	0.84	0.84	0.83
		elastic net	0.86	0.86	0.86	0.86	0.86	0.86	0.85
		neural network	0.72	0.72	0.72	0.72	0.72	0.72	0.71
	500	GMDH	0.81	0.81	0.81	0.81	0.81	0.81	0.81
		dce-GMDH	0.90	0.90	0.90	0.90	0.90	0.90	0.89
		svm	0.88	0.88	0.88	0.88	0.88	0.88	0.88
		random forest	0.85	0.85	0.85	0.85	0.85	0.85	0.85
		naive bayes	0.89	0.89	0.89	0.89	0.89	0.89	0.89
		elastic net	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		neural network	0.85	0.84	0.86	0.86	0.85	0.85	0.85
	1000	GMDH	0.82	0.82	0.81	0.82	0.82	0.82	0.82
		dce-GMDH	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		svm	0.89	0.89	0.89	0.89	0.89	0.89	0.89
		random forest	0.86	0.86	0.86	0.86	0.86	0.86	0.86
		naive bayes	0.90	0.89	0.90	0.90	0.89	0.90	0.90
		elastic net	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		neural network	0.87	0.87	0.88	0.87	0.87	0.87	0.87

Table A.1. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 10 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.78	0.78	0.77	0.78	0.78	0.78	0.76
		dce-GMDH	0.82	0.83	0.82	0.83	0.83	0.82	0.81
		svm	0.83	0.83	0.83	0.84	0.84	0.83	0.82
		random forest	0.79	0.79	0.79	0.80	0.81	0.79	0.77
		naive bayes	0.82	0.81	0.82	0.83	0.82	0.82	0.80
		elastic net	0.83	0.83	0.83	0.84	0.84	0.83	0.82
		neural network	0.69	0.71	0.66	0.68	0.71	0.69	0.67
	100	GMDH	0.80	0.80	0.80	0.80	0.81	0.80	0.79
		dce-GMDH	0.87	0.87	0.87	0.87	0.87	0.87	0.86
		svm	0.86	0.86	0.86	0.87	0.87	0.86	0.86
		random forest	0.83	0.83	0.82	0.83	0.83	0.83	0.82
		naive bayes	0.87	0.86	0.87	0.87	0.87	0.87	0.86
		elastic net	0.88	0.88	0.88	0.88	0.88	0.88	0.87
		neural network	0.75	0.76	0.74	0.75	0.76	0.75	0.74
	500	GMDH	0.83	0.83	0.83	0.83	0.83	0.83	0.83
		dce-GMDH	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		svm	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		random forest	0.87	0.87	0.87	0.87	0.87	0.87	0.87
		naive bayes	0.91	0.90	0.91	0.91	0.90	0.91	0.91
		elastic net	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		neural network	0.87	0.86	0.88	0.88	0.87	0.87	0.87
	1000	GMDH	0.84	0.84	0.84	0.84	0.84	0.84	0.84
		dce-GMDH	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		svm	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		random forest	0.88	0.88	0.88	0.88	0.88	0.88	0.88
		naive bayes	0.91	0.91	0.92	0.92	0.91	0.91	0.91
		elastic net	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		neural network	0.89	0.89	0.89	0.89	0.89	0.89	0.89

Table A.2. Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 15 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.63	0.63	0.62	0.63	0.64	0.63	0.60
		dce-GMDH	0.63	0.63	0.63	0.64	0.64	0.63	0.60
		svm	0.58	0.58	0.58	0.60	0.60	0.58	0.58
		random forest	0.63	0.63	0.63	0.65	0.65	0.63	0.60
		naive bayes	0.65	0.64	0.66	0.66	0.65	0.65	0.62
		elastic net	0.61	0.62	0.59	0.63	0.63	0.61	0.61
		neural network	0.56	0.59	0.53	0.55	0.57	0.56	0.54
	100	GMDH	0.65	0.65	0.65	0.65	0.66	0.65	0.64
		dce-GMDH	0.67	0.67	0.67	0.68	0.68	0.67	0.66
		svm	0.67	0.67	0.67	0.68	0.68	0.67	0.65
		random forest	0.66	0.67	0.66	0.67	0.68	0.66	0.65
		naive bayes	0.69	0.69	0.69	0.70	0.69	0.69	0.68
		elastic net	0.66	0.67	0.66	0.67	0.68	0.66	0.66
		neural network	0.58	0.59	0.57	0.58	0.58	0.58	0.57
	500	GMDH	0.67	0.68	0.67	0.67	0.67	0.67	0.67
		dce-GMDH	0.73	0.73	0.73	0.73	0.73	0.73	0.73
		svm	0.72	0.72	0.72	0.72	0.72	0.72	0.71
		random forest	0.71	0.71	0.70	0.71	0.71	0.71	0.70
		naive bayes	0.73	0.73	0.73	0.73	0.73	0.73	0.73
		elastic net	0.73	0.73	0.73	0.73	0.73	0.73	0.73
		neural network	0.64	0.63	0.64	0.64	0.64	0.64	0.63
	1000	GMDH	0.68	0.68	0.68	0.68	0.68	0.68	0.68
		dce-GMDH	0.74	0.74	0.74	0.74	0.74	0.74	0.74
		svm	0.72	0.72	0.72	0.72	0.72	0.72	0.72
		random forest	0.72	0.72	0.71	0.72	0.72	0.72	0.71
		naive bayes	0.74	0.74	0.74	0.74	0.74	0.74	0.74
		elastic net	0.74	0.74	0.74	0.74	0.74	0.74	0.74
		neural network	0.66	0.66	0.66	0.66	0.66	0.66	0.66
Medium	50	GMDH	0.73	0.73	0.73	0.73	0.73	0.73	0.71
		dce-GMDH	0.79	0.79	0.79	0.80	0.80	0.79	0.78
		svm	0.80	0.80	0.81	0.82	0.81	0.80	0.79
		random forest	0.76	0.76	0.76	0.78	0.78	0.76	0.74
		naive bayes	0.79	0.78	0.79	0.80	0.79	0.79	0.77
		elastic net	0.79	0.79	0.79	0.80	0.80	0.79	0.77
		neural network	0.64	0.68	0.60	0.63	0.66	0.64	0.63
	100	GMDH	0.75	0.75	0.74	0.75	0.75	0.75	0.74
		dce-GMDH	0.85	0.85	0.85	0.85	0.85	0.85	0.84
		svm	0.85	0.85	0.85	0.85	0.85	0.85	0.84
		random forest	0.80	0.80	0.80	0.81	0.81	0.80	0.79
		naive bayes	0.84	0.84	0.85	0.85	0.84	0.84	0.83
		elastic net	0.85	0.85	0.85	0.85	0.85	0.85	0.84
		neural network	0.69	0.71	0.67	0.68	0.70	0.69	0.68
	500	GMDH	0.78	0.78	0.77	0.78	0.78	0.78	0.77
		dce-GMDH	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		svm	0.89	0.89	0.89	0.89	0.89	0.89	0.89
		random forest	0.86	0.86	0.85	0.86	0.86	0.86	0.85
		naive bayes	0.90	0.89	0.90	0.90	0.90	0.90	0.90
		elastic net	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		neural network	0.83	0.82	0.85	0.84	0.83	0.83	0.83
	1000	GMDH	0.78	0.78	0.78	0.78	0.78	0.78	0.78
		dce-GMDH	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		svm	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		random forest	0.87	0.87	0.87	0.87	0.87	0.87	0.87
		naive bayes	0.91	0.90	0.91	0.91	0.90	0.91	0.91
		elastic net	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		neural network	0.86	0.86	0.87	0.87	0.86	0.86	0.86

Table A.2. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 15 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.75	0.75	0.75	0.75	0.75	0.75	0.73
		dce-GMDH	0.82	0.82	0.82	0.83	0.83	0.82	0.80
		svm	0.83	0.83	0.83	0.85	0.84	0.83	0.82
		random forest	0.79	0.80	0.78	0.81	0.81	0.79	0.77
		naive bayes	0.81	0.81	0.82	0.83	0.82	0.81	0.80
		elastic net	0.81	0.82	0.81	0.82	0.82	0.81	0.80
		neural network	0.66	0.70	0.61	0.65	0.68	0.66	0.65
	100	GMDH	0.77	0.77	0.77	0.77	0.77	0.77	0.76
		dce-GMDH	0.87	0.87	0.86	0.87	0.87	0.87	0.86
		svm	0.87	0.87	0.87	0.87	0.87	0.87	0.86
		random forest	0.83	0.83	0.82	0.83	0.84	0.83	0.82
		naive bayes	0.87	0.86	0.87	0.87	0.87	0.87	0.86
		elastic net	0.86	0.86	0.86	0.86	0.87	0.86	0.86
		neural network	0.71	0.73	0.68	0.70	0.72	0.71	0.70
	500	GMDH	0.80	0.80	0.80	0.80	0.80	0.80	0.80
		dce-GMDH	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		svm	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		random forest	0.87	0.88	0.87	0.87	0.88	0.87	0.87
		naive bayes	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		elastic net	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		neural network	0.85	0.84	0.86	0.86	0.85	0.85	0.85
	1000	GMDH	0.80	0.80	0.80	0.80	0.80	0.80	0.80
		dce-GMDH	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		svm	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		random forest	0.89	0.89	0.88	0.88	0.89	0.89	0.88
		naive bayes	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		elastic net	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		neural network	0.88	0.87	0.89	0.89	0.87	0.88	0.88

Table A.3. Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 5 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.56	0.57	0.55	0.57	0.57	0.56	0.54
		dce-GMDH	0.54	0.56	0.52	0.55	0.55	0.54	0.54
		svm	0.51	0.52	0.51	0.52	0.52	0.51	0.51
		random forest	0.55	0.55	0.55	0.55	0.55	0.55	0.52
		naive bayes	0.57	0.58	0.56	0.58	0.58	0.57	0.55
		elastic net	0.54	0.58	0.49	0.55	0.56	0.54	0.58
		neural network	0.53	0.54	0.52	0.53	0.54	0.53	0.51
	100	GMDH	0.57	0.59	0.56	0.58	0.59	0.57	0.56
		dce-GMDH	0.56	0.57	0.55	0.57	0.58	0.56	0.56
		svm	0.53	0.53	0.53	0.54	0.54	0.53	0.52
		random forest	0.55	0.55	0.55	0.55	0.56	0.55	0.53
		naive bayes	0.59	0.60	0.58	0.59	0.60	0.59	0.58
		elastic net	0.55	0.58	0.53	0.57	0.58	0.56	0.58
		neural network	0.54	0.57	0.52	0.54	0.55	0.54	0.53
	500	GMDH	0.60	0.61	0.58	0.60	0.60	0.60	0.60
		dce-GMDH	0.60	0.61	0.59	0.60	0.61	0.60	0.60
		svm	0.59	0.60	0.58	0.59	0.60	0.59	0.58
		random forest	0.57	0.57	0.56	0.57	0.57	0.57	0.57
		naive bayes	0.61	0.62	0.60	0.61	0.61	0.61	0.61
		elastic net	0.60	0.61	0.59	0.60	0.61	0.60	0.60
		neural network	0.57	0.60	0.54	0.57	0.58	0.57	0.57
	1000	GMDH	0.60	0.61	0.59	0.60	0.60	0.60	0.60
		dce-GMDH	0.60	0.62	0.59	0.60	0.61	0.60	0.61
		svm	0.60	0.60	0.59	0.60	0.60	0.60	0.60
		random forest	0.58	0.58	0.57	0.57	0.58	0.58	0.58
		naive bayes	0.61	0.62	0.60	0.61	0.61	0.61	0.61
		elastic net	0.61	0.62	0.59	0.60	0.61	0.61	0.61
		neural network	0.58	0.60	0.56	0.58	0.59	0.58	0.58
Medium	50	GMDH	0.71	0.72	0.70	0.71	0.72	0.71	0.69
		dce-GMDH	0.71	0.71	0.71	0.72	0.72	0.71	0.69
		svm	0.68	0.67	0.68	0.70	0.70	0.68	0.66
		random forest	0.71	0.70	0.71	0.71	0.71	0.71	0.68
		naive bayes	0.74	0.74	0.73	0.74	0.74	0.74	0.71
		elastic net	0.71	0.71	0.70	0.72	0.73	0.71	0.69
		neural network	0.65	0.65	0.65	0.65	0.66	0.65	0.62
	100	GMDH	0.72	0.73	0.71	0.72	0.73	0.72	0.71
		dce-GMDH	0.73	0.73	0.73	0.73	0.74	0.73	0.72
		svm	0.72	0.72	0.72	0.73	0.73	0.72	0.71
		random forest	0.72	0.72	0.72	0.73	0.72	0.72	0.71
		naive bayes	0.75	0.75	0.75	0.75	0.75	0.75	0.74
		elastic net	0.74	0.74	0.73	0.74	0.75	0.74	0.73
		neural network	0.68	0.70	0.67	0.68	0.69	0.68	0.67
	500	GMDH	0.74	0.75	0.73	0.74	0.75	0.74	0.74
		dce-GMDH	0.75	0.76	0.75	0.75	0.76	0.75	0.75
		svm	0.75	0.75	0.74	0.75	0.75	0.75	0.74
		random forest	0.73	0.74	0.73	0.74	0.74	0.73	0.73
		naive bayes	0.76	0.76	0.75	0.76	0.76	0.76	0.76
		elastic net	0.76	0.76	0.75	0.75	0.76	0.76	0.76
		neural network	0.73	0.74	0.71	0.73	0.74	0.73	0.73
	1000	GMDH	0.75	0.75	0.74	0.74	0.75	0.75	0.75
		dce-GMDH	0.76	0.76	0.75	0.75	0.76	0.76	0.76
		svm	0.75	0.75	0.75	0.75	0.75	0.75	0.75
		random forest	0.74	0.74	0.74	0.74	0.74	0.74	0.74
		naive bayes	0.76	0.76	0.75	0.76	0.76	0.76	0.76
		elastic net	0.76	0.76	0.75	0.76	0.76	0.76	0.76
		neural network	0.74	0.75	0.73	0.74	0.75	0.74	0.74

Table A.3. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 5 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.85	0.86	0.84	0.84	0.86	0.85	0.83
		dce-GMDH	0.89	0.89	0.90	0.90	0.90	0.89	0.88
		svm	0.89	0.88	0.89	0.90	0.89	0.89	0.87
		random forest	0.88	0.88	0.88	0.88	0.89	0.88	0.87
		naive bayes	0.89	0.89	0.88	0.89	0.89	0.89	0.88
		elastic net	0.91	0.91	0.91	0.91	0.91	0.91	0.90
		neural network	0.85	0.86	0.84	0.85	0.86	0.85	0.84
	100	GMDH	0.86	0.88	0.85	0.86	0.88	0.86	0.86
		dce-GMDH	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		svm	0.91	0.91	0.91	0.92	0.91	0.91	0.91
		random forest	0.90	0.90	0.90	0.90	0.90	0.90	0.89
		naive bayes	0.91	0.91	0.92	0.92	0.91	0.91	0.91
		elastic net	0.93	0.93	0.93	0.93	0.93	0.93	0.93
		neural network	0.90	0.89	0.90	0.90	0.90	0.90	0.89
	500	GMDH	0.90	0.91	0.89	0.89	0.90	0.90	0.90
		dce-GMDH	0.94	0.94	0.94	0.94	0.94	0.94	0.94
		svm	0.94	0.94	0.94	0.94	0.94	0.94	0.94
		random forest	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		naive bayes	0.93	0.93	0.94	0.94	0.93	0.93	0.93
		elastic net	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		neural network	0.93	0.93	0.94	0.94	0.93	0.93	0.93
	1000	GMDH	0.91	0.92	0.90	0.90	0.91	0.91	0.91
		dce-GMDH	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		svm	0.94	0.94	0.94	0.94	0.94	0.94	0.94
		random forest	0.93	0.93	0.93	0.93	0.93	0.93	0.93
		naive bayes	0.94	0.93	0.94	0.94	0.93	0.94	0.94
		elastic net	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		neural network	0.94	0.94	0.94	0.94	0.94	0.94	0.94

Table A.4. Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 10 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.55	0.56	0.55	0.56	0.56	0.55	0.53
		dce-GMDH	0.54	0.55	0.53	0.55	0.55	0.54	0.53
		svm	0.51	0.51	0.51	0.52	0.51	0.51	0.50
		random forest	0.55	0.55	0.55	0.56	0.55	0.55	0.52
		naive bayes	0.58	0.57	0.57	0.58	0.58	0.57	0.55
		elastic net	0.54	0.57	0.50	0.55	0.56	0.54	0.58
		neural network	0.53	0.53	0.53	0.53	0.53	0.53	0.50
	100	GMDH	0.57	0.57	0.57	0.57	0.57	0.57	0.55
		dce-GMDH	0.56	0.56	0.56	0.57	0.57	0.56	0.55
		svm	0.53	0.52	0.53	0.54	0.54	0.53	0.52
		random forest	0.56	0.55	0.56	0.56	0.56	0.56	0.54
		naive bayes	0.59	0.59	0.59	0.59	0.59	0.59	0.57
		elastic net	0.55	0.56	0.54	0.57	0.57	0.55	0.57
		neural network	0.53	0.53	0.53	0.53	0.53	0.53	0.51
	500	GMDH	0.59	0.60	0.58	0.59	0.60	0.59	0.59
		dce-GMDH	0.59	0.60	0.58	0.59	0.60	0.59	0.59
		svm	0.58	0.58	0.58	0.59	0.59	0.58	0.57
		random forest	0.57	0.57	0.57	0.57	0.57	0.57	0.57
		naive bayes	0.61	0.61	0.60	0.60	0.61	0.60	0.60
		elastic net	0.59	0.60	0.59	0.60	0.60	0.59	0.59
		neural network	0.55	0.58	0.52	0.54	0.55	0.55	0.55
	1000	GMDH	0.60	0.61	0.59	0.60	0.60	0.60	0.60
		dce-GMDH	0.60	0.61	0.59	0.60	0.60	0.60	0.60
		svm	0.59	0.59	0.59	0.59	0.59	0.59	0.59
		random forest	0.57	0.58	0.57	0.58	0.58	0.57	0.57
		naive bayes	0.60	0.61	0.60	0.60	0.61	0.60	0.60
		elastic net	0.60	0.61	0.59	0.60	0.60	0.60	0.60
		neural network	0.55	0.59	0.51	0.55	0.56	0.55	0.57
Medium	50	GMDH	0.71	0.72	0.71	0.72	0.72	0.71	0.69
		dce-GMDH	0.72	0.73	0.72	0.73	0.73	0.72	0.70
		svm	0.70	0.70	0.70	0.73	0.72	0.70	0.69
		random forest	0.73	0.73	0.73	0.74	0.74	0.73	0.71
		naive bayes	0.75	0.75	0.75	0.76	0.76	0.75	0.73
		elastic net	0.72	0.72	0.72	0.74	0.73	0.72	0.71
		neural network	0.65	0.66	0.63	0.65	0.65	0.65	0.63
	100	GMDH	0.73	0.73	0.73	0.73	0.73	0.73	0.72
		dce-GMDH	0.74	0.74	0.74	0.74	0.74	0.74	0.73
		svm	0.74	0.73	0.74	0.75	0.74	0.74	0.72
		random forest	0.74	0.74	0.74	0.74	0.74	0.74	0.73
		naive bayes	0.76	0.76	0.76	0.76	0.76	0.76	0.75
		elastic net	0.74	0.74	0.74	0.75	0.75	0.74	0.73
		neural network	0.67	0.66	0.67	0.67	0.67	0.67	0.65
	500	GMDH	0.75	0.75	0.75	0.75	0.75	0.75	0.75
		dce-GMDH	0.76	0.76	0.76	0.76	0.76	0.76	0.76
		svm	0.76	0.76	0.76	0.76	0.76	0.76	0.75
		random forest	0.75	0.76	0.75	0.76	0.76	0.75	0.75
		naive bayes	0.77	0.77	0.77	0.77	0.77	0.77	0.77
		elastic net	0.76	0.76	0.76	0.77	0.77	0.76	0.76
		neural network	0.71	0.72	0.70	0.71	0.71	0.71	0.71
	1000	GMDH	0.76	0.76	0.76	0.76	0.76	0.76	0.76
		dce-GMDH	0.77	0.77	0.77	0.77	0.77	0.77	0.77
		svm	0.76	0.76	0.76	0.76	0.76	0.76	0.76
		random forest	0.76	0.76	0.76	0.76	0.76	0.76	0.76
		naive bayes	0.77	0.77	0.77	0.77	0.77	0.77	0.77
		elastic net	0.77	0.77	0.77	0.77	0.77	0.77	0.77
		neural network	0.72	0.73	0.70	0.71	0.73	0.72	0.72

Table A.4. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 10 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.84	0.84	0.83	0.84	0.84	0.84	0.82
		dce-GMDH	0.90	0.90	0.90	0.90	0.90	0.90	0.89
		svm	0.90	0.90	0.90	0.91	0.91	0.90	0.89
		random forest	0.88	0.89	0.88	0.89	0.89	0.88	0.87
		naive bayes	0.90	0.90	0.90	0.91	0.90	0.90	0.89
		elastic net	0.90	0.90	0.90	0.90	0.90	0.90	0.89
		neural network	0.79	0.82	0.75	0.77	0.81	0.79	0.78
	100	GMDH	0.86	0.87	0.86	0.86	0.87	0.86	0.86
		dce-GMDH	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		svm	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		random forest	0.90	0.90	0.90	0.90	0.91	0.90	0.90
		naive bayes	0.92	0.92	0.93	0.93	0.92	0.92	0.92
		elastic net	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		neural network	0.85	0.86	0.84	0.84	0.86	0.85	0.84
	500	GMDH	0.90	0.90	0.89	0.89	0.90	0.90	0.89
		dce-GMDH	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		svm	0.94	0.94	0.94	0.94	0.94	0.94	0.94
		random forest	0.93	0.93	0.93	0.93	0.93	0.93	0.93
		naive bayes	0.95	0.94	0.95	0.95	0.95	0.95	0.95
		elastic net	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		neural network	0.92	0.92	0.93	0.93	0.92	0.92	0.92
	1000	GMDH	0.90	0.91	0.90	0.90	0.91	0.90	0.90
		dce-GMDH	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		svm	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		random forest	0.94	0.94	0.93	0.93	0.94	0.94	0.94
		naive bayes	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		elastic net	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		neural network	0.93	0.93	0.94	0.94	0.93	0.93	0.93

Table A.5. Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 15 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.56	0.56	0.57	0.57	0.56	0.56	0.53
		dce-GMDH	0.55	0.56	0.55	0.56	0.56	0.55	0.54
		svm	0.52	0.51	0.52	0.52	0.52	0.52	0.51
		random forest	0.57	0.57	0.56	0.57	0.57	0.57	0.54
		naive bayes	0.59	0.58	0.59	0.59	0.59	0.59	0.56
		elastic net	0.55	0.57	0.52	0.56	0.56	0.54	0.57
		neural network	0.54	0.54	0.53	0.54	0.54	0.54	0.51
	100	GMDH	0.58	0.57	0.59	0.59	0.58	0.58	0.56
		dce-GMDH	0.58	0.58	0.57	0.58	0.58	0.57	0.57
		svm	0.54	0.55	0.54	0.56	0.56	0.54	0.54
		random forest	0.58	0.58	0.57	0.58	0.58	0.58	0.56
		naive bayes	0.61	0.61	0.61	0.61	0.61	0.61	0.59
		elastic net	0.57	0.57	0.56	0.59	0.59	0.57	0.58
		neural network	0.54	0.54	0.54	0.54	0.54	0.54	0.53
	500	GMDH	0.60	0.60	0.61	0.61	0.61	0.60	0.60
		dce-GMDH	0.60	0.60	0.60	0.61	0.61	0.60	0.60
		svm	0.60	0.60	0.60	0.60	0.60	0.60	0.59
		random forest	0.59	0.60	0.59	0.59	0.59	0.59	0.59
		naive bayes	0.62	0.61	0.62	0.62	0.62	0.62	0.61
		elastic net	0.61	0.60	0.61	0.61	0.61	0.61	0.60
		neural network	0.55	0.56	0.54	0.55	0.55	0.55	0.55
	1000	GMDH	0.61	0.60	0.62	0.61	0.61	0.61	0.61
		dce-GMDH	0.61	0.61	0.62	0.61	0.61	0.61	0.61
		svm	0.61	0.60	0.61	0.61	0.61	0.61	0.60
		random forest	0.60	0.60	0.60	0.60	0.60	0.60	0.60
		naive bayes	0.62	0.61	0.62	0.62	0.62	0.62	0.61
		elastic net	0.62	0.61	0.62	0.62	0.62	0.62	0.61
		neural network	0.56	0.57	0.54	0.56	0.56	0.56	0.56
Medium	50	GMDH	0.73	0.72	0.74	0.74	0.73	0.73	0.71
		dce-GMDH	0.75	0.75	0.75	0.76	0.76	0.75	0.73
		svm	0.75	0.75	0.75	0.76	0.77	0.75	0.73
		random forest	0.76	0.76	0.76	0.77	0.77	0.76	0.74
		naive bayes	0.78	0.78	0.78	0.78	0.78	0.78	0.76
		elastic net	0.75	0.75	0.75	0.76	0.76	0.75	0.73
		neural network	0.69	0.71	0.66	0.68	0.71	0.69	0.67
	100	GMDH	0.75	0.74	0.75	0.75	0.75	0.75	0.73
		dce-GMDH	0.76	0.76	0.76	0.77	0.77	0.76	0.75
		svm	0.77	0.77	0.77	0.78	0.78	0.77	0.76
		random forest	0.77	0.77	0.77	0.78	0.78	0.77	0.76
		naive bayes	0.79	0.79	0.78	0.79	0.79	0.79	0.78
		elastic net	0.76	0.76	0.77	0.77	0.77	0.76	0.75
		neural network	0.70	0.71	0.69	0.70	0.71	0.70	0.69
	500	GMDH	0.77	0.77	0.77	0.77	0.77	0.77	0.77
		dce-GMDH	0.79	0.78	0.79	0.79	0.79	0.79	0.78
		svm	0.78	0.78	0.78	0.78	0.79	0.78	0.78
		random forest	0.78	0.79	0.78	0.78	0.79	0.78	0.78
		naive bayes	0.79	0.79	0.79	0.79	0.79	0.79	0.79
		elastic net	0.79	0.79	0.79	0.79	0.79	0.79	0.79
		neural network	0.73	0.72	0.73	0.73	0.73	0.73	0.72
	1000	GMDH	0.78	0.78	0.78	0.78	0.78	0.78	0.78
		dce-GMDH	0.79	0.79	0.79	0.79	0.79	0.79	0.79
		svm	0.79	0.79	0.79	0.79	0.79	0.79	0.79
		random forest	0.79	0.79	0.79	0.79	0.79	0.79	0.79
		naive bayes	0.79	0.79	0.79	0.79	0.79	0.79	0.79
		elastic net	0.79	0.79	0.79	0.79	0.79	0.79	0.79
		neural network	0.73	0.74	0.73	0.73	0.74	0.73	0.73

Table A.5. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 15 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.83	0.83	0.84	0.84	0.83	0.83	0.82
		dce-GMDH	0.91	0.91	0.90	0.91	0.91	0.91	0.90
		svm	0.92	0.92	0.91	0.92	0.92	0.92	0.91
		random forest	0.90	0.90	0.90	0.91	0.91	0.90	0.89
		naive bayes	0.92	0.92	0.92	0.92	0.92	0.92	0.91
		elastic net	0.90	0.90	0.90	0.90	0.90	0.90	0.89
		neural network	0.80	0.84	0.75	0.78	0.83	0.80	0.79
	100	GMDH	0.86	0.85	0.86	0.86	0.85	0.86	0.85
		dce-GMDH	0.93	0.93	0.93	0.93	0.94	0.93	0.93
		svm	0.93	0.93	0.93	0.93	0.93	0.93	0.93
		random forest	0.92	0.92	0.92	0.92	0.92	0.92	0.91
		naive bayes	0.94	0.94	0.93	0.93	0.94	0.94	0.93
		elastic net	0.92	0.92	0.92	0.92	0.92	0.92	0.92
		neural network	0.83	0.86	0.81	0.82	0.85	0.83	0.83
	500	GMDH	0.88	0.88	0.89	0.89	0.88	0.88	0.88
		dce-GMDH	0.96	0.96	0.95	0.95	0.96	0.96	0.96
		svm	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		random forest	0.94	0.94	0.94	0.94	0.94	0.94	0.94
		naive bayes	0.96	0.96	0.95	0.95	0.96	0.96	0.96
		elastic net	0.96	0.96	0.96	0.96	0.96	0.96	0.96
		neural network	0.92	0.92	0.92	0.92	0.92	0.92	0.92
	1000	GMDH	0.89	0.89	0.90	0.90	0.89	0.89	0.89
		dce-GMDH	0.96	0.96	0.96	0.96	0.96	0.96	0.96
		svm	0.96	0.96	0.96	0.96	0.96	0.96	0.96
		random forest	0.95	0.95	0.95	0.95	0.95	0.95	0.95
		naive bayes	0.96	0.96	0.95	0.96	0.96	0.96	0.96
		elastic net	0.96	0.96	0.96	0.96	0.96	0.96	0.96
		neural network	0.94	0.94	0.94	0.94	0.94	0.94	0.94

Table A.6. Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 5 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.55	0.56	0.54	0.56	0.56	0.55	0.53
		dce-GMDH	0.54	0.56	0.52	0.54	0.55	0.54	0.53
		svm	0.51	0.52	0.50	0.51	0.52	0.51	0.52
		random forest	0.53	0.54	0.53	0.54	0.54	0.53	0.50
		naive bayes	0.56	0.58	0.55	0.56	0.57	0.56	0.54
		elastic net	0.54	0.57	0.49	0.55	0.56	0.53	0.59
		neural network	0.53	0.54	0.52	0.53	0.53	0.53	0.50
	100	GMDH	0.56	0.58	0.55	0.57	0.58	0.57	0.55
		dce-GMDH	0.55	0.57	0.53	0.55	0.57	0.55	0.55
		svm	0.53	0.53	0.52	0.53	0.54	0.53	0.53
		random forest	0.54	0.54	0.53	0.54	0.55	0.54	0.52
		naive bayes	0.57	0.59	0.56	0.57	0.58	0.57	0.56
		elastic net	0.55	0.58	0.52	0.56	0.57	0.55	0.58
		neural network	0.53	0.55	0.51	0.53	0.55	0.53	0.52
	500	GMDH	0.59	0.61	0.57	0.59	0.59	0.59	0.59
		dce-GMDH	0.59	0.61	0.56	0.58	0.59	0.59	0.59
		svm	0.58	0.60	0.56	0.58	0.59	0.58	0.58
		random forest	0.55	0.56	0.55	0.56	0.56	0.55	0.55
		naive bayes	0.59	0.61	0.57	0.59	0.59	0.59	0.59
		elastic net	0.59	0.61	0.56	0.59	0.60	0.59	0.59
		neural network	0.56	0.59	0.53	0.56	0.57	0.56	0.56
	1000	GMDH	0.59	0.61	0.57	0.59	0.60	0.59	0.60
		dce-GMDH	0.59	0.62	0.57	0.59	0.60	0.59	0.60
		svm	0.59	0.62	0.56	0.59	0.60	0.59	0.60
		random forest	0.56	0.58	0.55	0.56	0.57	0.56	0.57
		naive bayes	0.59	0.61	0.57	0.59	0.59	0.59	0.60
		elastic net	0.59	0.62	0.57	0.59	0.60	0.59	0.60
		neural network	0.57	0.60	0.55	0.57	0.58	0.57	0.58
Medium	50	GMDH	0.68	0.69	0.66	0.68	0.69	0.68	0.66
		dce-GMDH	0.66	0.68	0.65	0.67	0.68	0.66	0.65
		svm	0.63	0.64	0.62	0.65	0.65	0.63	0.63
		random forest	0.65	0.65	0.65	0.66	0.66	0.65	0.62
		naive bayes	0.70	0.71	0.68	0.69	0.71	0.70	0.68
		elastic net	0.67	0.68	0.65	0.68	0.69	0.67	0.66
		neural network	0.61	0.62	0.61	0.62	0.62	0.61	0.59
	100	GMDH	0.69	0.71	0.67	0.69	0.71	0.69	0.68
		dce-GMDH	0.68	0.70	0.67	0.68	0.70	0.68	0.67
		svm	0.68	0.69	0.67	0.68	0.70	0.68	0.67
		random forest	0.66	0.66	0.66	0.66	0.67	0.66	0.65
		naive bayes	0.70	0.72	0.69	0.70	0.72	0.70	0.70
		elastic net	0.69	0.71	0.68	0.69	0.71	0.69	0.68
		neural network	0.64	0.66	0.61	0.63	0.65	0.64	0.63
	500	GMDH	0.70	0.72	0.68	0.70	0.71	0.70	0.71
		dce-GMDH	0.70	0.72	0.69	0.70	0.71	0.70	0.70
		svm	0.70	0.72	0.68	0.69	0.71	0.70	0.70
		random forest	0.68	0.68	0.67	0.68	0.68	0.68	0.68
		naive bayes	0.71	0.73	0.69	0.70	0.72	0.71	0.71
		elastic net	0.71	0.72	0.69	0.70	0.71	0.71	0.71
		neural network	0.68	0.70	0.65	0.67	0.69	0.68	0.68
	1000	GMDH	0.70	0.72	0.69	0.70	0.71	0.70	0.71
		dce-GMDH	0.71	0.72	0.69	0.70	0.71	0.71	0.71
		svm	0.70	0.72	0.69	0.70	0.71	0.70	0.70
		random forest	0.68	0.69	0.68	0.68	0.69	0.68	0.68
		naive bayes	0.71	0.73	0.69	0.70	0.71	0.71	0.71
		elastic net	0.71	0.72	0.69	0.70	0.71	0.71	0.71
		neural network	0.69	0.71	0.67	0.69	0.70	0.69	0.70

Table A.6. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 5 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.83	0.86	0.81	0.82	0.86	0.83	0.82
		dce-GMDH	0.83	0.84	0.82	0.83	0.84	0.83	0.82
		svm	0.83	0.84	0.82	0.83	0.85	0.83	0.82
		random forest	0.83	0.83	0.83	0.83	0.84	0.83	0.81
		naive bayes	0.85	0.87	0.82	0.83	0.86	0.85	0.84
		elastic net	0.84	0.85	0.83	0.84	0.85	0.84	0.83
		neural network	0.79	0.80	0.78	0.79	0.80	0.79	0.77
	100	GMDH	0.84	0.87	0.81	0.82	0.86	0.84	0.84
		dce-GMDH	0.84	0.85	0.83	0.84	0.85	0.84	0.83
		svm	0.84	0.85	0.83	0.84	0.85	0.84	0.84
		random forest	0.84	0.84	0.83	0.84	0.84	0.84	0.83
		naive bayes	0.85	0.86	0.84	0.85	0.86	0.85	0.85
		elastic net	0.85	0.86	0.84	0.84	0.86	0.85	0.84
		neural network	0.81	0.82	0.80	0.80	0.82	0.81	0.80
	500	GMDH	0.85	0.88	0.81	0.83	0.87	0.85	0.85
		dce-GMDH	0.85	0.86	0.84	0.85	0.86	0.85	0.85
		svm	0.85	0.86	0.84	0.84	0.86	0.85	0.85
		random forest	0.84	0.85	0.84	0.84	0.85	0.84	0.84
		naive bayes	0.85	0.86	0.85	0.85	0.86	0.85	0.85
		elastic net	0.86	0.87	0.85	0.85	0.87	0.86	0.86
		neural network	0.84	0.85	0.83	0.83	0.85	0.84	0.84
	1000	GMDH	0.85	0.88	0.82	0.83	0.87	0.85	0.85
		dce-GMDH	0.86	0.87	0.85	0.85	0.87	0.86	0.86
		svm	0.85	0.86	0.84	0.85	0.86	0.85	0.85
		random forest	0.85	0.85	0.84	0.85	0.85	0.85	0.85
		naive bayes	0.85	0.86	0.85	0.85	0.86	0.85	0.85
		elastic net	0.86	0.87	0.85	0.85	0.87	0.86	0.86
		neural network	0.85	0.86	0.84	0.84	0.86	0.85	0.85

Table A.7. Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 10 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.56	0.56	0.56	0.57	0.56	0.56	0.53
		dce-GMDH	0.54	0.55	0.53	0.54	0.54	0.54	0.53
		svm	0.51	0.51	0.51	0.51	0.51	0.51	0.50
		random forest	0.54	0.54	0.54	0.54	0.55	0.54	0.51
		naive bayes	0.56	0.56	0.56	0.56	0.57	0.56	0.53
		elastic net	0.53	0.56	0.51	0.55	0.55	0.54	0.57
		neural network	0.54	0.53	0.54	0.54	0.54	0.54	0.51
	100	GMDH	0.57	0.57	0.57	0.58	0.58	0.57	0.55
		dce-GMDH	0.56	0.56	0.55	0.56	0.56	0.56	0.55
		svm	0.53	0.52	0.53	0.53	0.53	0.53	0.52
		random forest	0.55	0.55	0.55	0.55	0.55	0.55	0.53
		naive bayes	0.57	0.57	0.58	0.58	0.57	0.57	0.56
		elastic net	0.55	0.57	0.54	0.57	0.57	0.55	0.57
		neural network	0.54	0.54	0.54	0.55	0.54	0.54	0.53
	500	GMDH	0.60	0.60	0.60	0.60	0.60	0.60	0.59
		dce-GMDH	0.60	0.60	0.59	0.60	0.60	0.60	0.59
		svm	0.59	0.59	0.59	0.59	0.59	0.59	0.58
		random forest	0.57	0.57	0.57	0.57	0.57	0.57	0.57
		naive bayes	0.59	0.58	0.59	0.59	0.59	0.59	0.58
		elastic net	0.60	0.60	0.60	0.60	0.61	0.60	0.60
		neural network	0.56	0.57	0.55	0.56	0.56	0.56	0.56
	1000	GMDH	0.60	0.60	0.60	0.60	0.60	0.60	0.60
		dce-GMDH	0.61	0.61	0.61	0.61	0.61	0.61	0.61
		svm	0.60	0.60	0.60	0.60	0.60	0.60	0.60
		random forest	0.58	0.58	0.58	0.58	0.58	0.58	0.58
		naive bayes	0.59	0.58	0.59	0.59	0.59	0.59	0.59
		elastic net	0.61	0.62	0.61	0.62	0.62	0.61	0.61
		neural network	0.57	0.58	0.56	0.57	0.57	0.57	0.57
Medium	50	GMDH	0.67	0.67	0.68	0.68	0.68	0.67	0.64
		dce-GMDH	0.66	0.66	0.66	0.67	0.67	0.66	0.64
		svm	0.62	0.62	0.62	0.64	0.64	0.62	0.62
		random forest	0.66	0.66	0.66	0.67	0.67	0.66	0.63
		naive bayes	0.69	0.69	0.69	0.69	0.69	0.69	0.66
		elastic net	0.66	0.67	0.66	0.68	0.68	0.66	0.65
		neural network	0.60	0.61	0.60	0.61	0.61	0.61	0.58
	100	GMDH	0.69	0.68	0.69	0.69	0.69	0.69	0.67
		dce-GMDH	0.68	0.68	0.68	0.69	0.68	0.68	0.67
		svm	0.67	0.67	0.67	0.69	0.68	0.67	0.66
		random forest	0.67	0.67	0.67	0.68	0.68	0.67	0.66
		naive bayes	0.70	0.69	0.70	0.70	0.69	0.70	0.68
		elastic net	0.69	0.69	0.69	0.70	0.69	0.69	0.68
		neural network	0.62	0.61	0.62	0.62	0.62	0.62	0.60
	500	GMDH	0.70	0.70	0.71	0.71	0.70	0.70	0.70
		dce-GMDH	0.71	0.71	0.71	0.71	0.71	0.71	0.71
		svm	0.70	0.69	0.70	0.70	0.70	0.70	0.69
		random forest	0.69	0.69	0.69	0.69	0.69	0.69	0.69
		naive bayes	0.70	0.70	0.70	0.70	0.70	0.70	0.70
		elastic net	0.72	0.72	0.72	0.72	0.72	0.72	0.71
		neural network	0.66	0.67	0.64	0.65	0.66	0.66	0.66
	1000	GMDH	0.71	0.71	0.71	0.71	0.71	0.71	0.71
		dce-GMDH	0.72	0.73	0.72	0.72	0.73	0.72	0.72
		svm	0.70	0.70	0.71	0.71	0.70	0.70	0.70
		random forest	0.70	0.70	0.70	0.70	0.70	0.70	0.70
		naive bayes	0.70	0.70	0.70	0.70	0.70	0.70	0.70
		elastic net	0.72	0.73	0.72	0.72	0.73	0.72	0.72
		neural network	0.67	0.69	0.66	0.67	0.68	0.67	0.68

Table A.7. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 10 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.83	0.82	0.83	0.83	0.82	0.83	0.81
		dce-GMDH	0.84	0.84	0.84	0.84	0.84	0.84	0.82
		svm	0.83	0.83	0.84	0.84	0.84	0.83	0.82
		random forest	0.85	0.85	0.85	0.85	0.85	0.85	0.83
		naive bayes	0.85	0.85	0.86	0.86	0.85	0.85	0.84
		elastic net	0.84	0.84	0.85	0.85	0.84	0.84	0.82
		neural network	0.80	0.81	0.79	0.80	0.81	0.80	0.78
	100	GMDH	0.84	0.83	0.85	0.85	0.83	0.84	0.83
		dce-GMDH	0.85	0.84	0.85	0.85	0.85	0.85	0.84
		svm	0.84	0.84	0.85	0.85	0.84	0.84	0.83
		random forest	0.85	0.85	0.85	0.86	0.85	0.85	0.85
		naive bayes	0.85	0.85	0.86	0.86	0.85	0.85	0.85
		elastic net	0.85	0.85	0.86	0.86	0.85	0.85	0.85
		neural network	0.81	0.81	0.81	0.81	0.81	0.81	0.80
	500	GMDH	0.85	0.84	0.86	0.86	0.85	0.85	0.85
		dce-GMDH	0.87	0.87	0.87	0.87	0.87	0.87	0.87
		svm	0.86	0.86	0.86	0.86	0.86	0.86	0.86
		random forest	0.86	0.87	0.86	0.87	0.87	0.86	0.86
		naive bayes	0.86	0.86	0.86	0.86	0.86	0.86	0.86
		elastic net	0.87	0.87	0.88	0.88	0.87	0.87	0.87
		neural network	0.84	0.84	0.83	0.84	0.84	0.84	0.83
	1000	GMDH	0.85	0.84	0.86	0.86	0.85	0.85	0.85
		dce-GMDH	0.88	0.87	0.88	0.88	0.87	0.88	0.88
		svm	0.87	0.86	0.87	0.87	0.86	0.87	0.86
		random forest	0.87	0.87	0.87	0.87	0.87	0.87	0.87
		naive bayes	0.86	0.86	0.86	0.86	0.86	0.86	0.86
		elastic net	0.88	0.87	0.88	0.88	0.88	0.88	0.88
		neural network	0.85	0.85	0.84	0.85	0.85	0.85	0.85

Table A.8. Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 15 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.56	0.56	0.56	0.57	0.56	0.56	0.53
		dce-GMDH	0.54	0.55	0.54	0.55	0.55	0.54	0.53
		svm	0.51	0.51	0.51	0.52	0.51	0.51	0.51
		random forest	0.54	0.54	0.54	0.55	0.55	0.54	0.51
		naive bayes	0.56	0.56	0.57	0.57	0.56	0.56	0.54
		elastic net	0.54	0.56	0.52	0.55	0.55	0.54	0.57
		neural network	0.54	0.54	0.54	0.55	0.54	0.54	0.51
	100	GMDH	0.58	0.57	0.58	0.58	0.58	0.58	0.55
		dce-GMDH	0.56	0.56	0.55	0.56	0.57	0.56	0.55
		svm	0.53	0.53	0.52	0.54	0.53	0.53	0.52
		random forest	0.55	0.56	0.55	0.56	0.56	0.55	0.53
		naive bayes	0.58	0.57	0.58	0.58	0.58	0.58	0.56
		elastic net	0.56	0.57	0.54	0.57	0.57	0.56	0.57
		neural network	0.55	0.55	0.55	0.55	0.55	0.55	0.54
	500	GMDH	0.60	0.60	0.60	0.60	0.60	0.60	0.60
		dce-GMDH	0.61	0.61	0.60	0.61	0.61	0.61	0.60
		svm	0.59	0.59	0.59	0.60	0.60	0.59	0.58
		random forest	0.58	0.58	0.58	0.58	0.58	0.58	0.58
		naive bayes	0.59	0.59	0.60	0.59	0.59	0.59	0.59
		elastic net	0.61	0.61	0.61	0.62	0.62	0.61	0.61
		neural network	0.57	0.58	0.56	0.57	0.57	0.57	0.57
	1000	GMDH	0.61	0.61	0.61	0.61	0.61	0.61	0.60
		dce-GMDH	0.63	0.63	0.62	0.63	0.63	0.63	0.62
		svm	0.60	0.60	0.61	0.60	0.60	0.60	0.60
		random forest	0.59	0.59	0.59	0.59	0.59	0.59	0.59
		naive bayes	0.59	0.59	0.60	0.59	0.59	0.59	0.59
		elastic net	0.63	0.63	0.63	0.63	0.63	0.63	0.63
		neural network	0.58	0.59	0.56	0.58	0.58	0.58	0.58
Medium	50	GMDH	0.67	0.66	0.68	0.68	0.67	0.67	0.64
		dce-GMDH	0.67	0.67	0.67	0.68	0.68	0.67	0.65
		svm	0.63	0.63	0.63	0.66	0.65	0.63	0.63
		random forest	0.67	0.67	0.67	0.68	0.68	0.67	0.64
		naive bayes	0.70	0.69	0.70	0.70	0.70	0.70	0.67
		elastic net	0.67	0.67	0.67	0.69	0.68	0.67	0.66
		neural network	0.63	0.64	0.62	0.64	0.64	0.63	0.61
	100	GMDH	0.69	0.68	0.70	0.70	0.69	0.69	0.67
		dce-GMDH	0.68	0.68	0.69	0.69	0.69	0.68	0.67
		svm	0.68	0.67	0.68	0.69	0.69	0.68	0.66
		random forest	0.68	0.68	0.68	0.69	0.69	0.68	0.66
		naive bayes	0.70	0.70	0.71	0.70	0.70	0.70	0.69
		elastic net	0.69	0.68	0.70	0.70	0.70	0.69	0.68
		neural network	0.64	0.64	0.64	0.64	0.64	0.64	0.62
	500	GMDH	0.71	0.69	0.72	0.71	0.70	0.71	0.70
		dce-GMDH	0.71	0.70	0.72	0.71	0.71	0.71	0.71
		svm	0.70	0.69	0.71	0.71	0.70	0.70	0.70
		random forest	0.70	0.70	0.70	0.70	0.70	0.70	0.70
		naive bayes	0.70	0.70	0.71	0.71	0.70	0.70	0.70
		elastic net	0.71	0.71	0.72	0.72	0.71	0.71	0.71
		neural network	0.66	0.67	0.65	0.66	0.66	0.66	0.66
	1000	GMDH	0.71	0.70	0.72	0.71	0.70	0.71	0.70
		dce-GMDH	0.72	0.72	0.73	0.72	0.72	0.72	0.72
		svm	0.71	0.70	0.71	0.71	0.71	0.71	0.70
		random forest	0.71	0.71	0.70	0.71	0.71	0.71	0.70
		naive bayes	0.70	0.70	0.71	0.70	0.70	0.70	0.70
		elastic net	0.72	0.72	0.73	0.73	0.72	0.72	0.72
		neural network	0.67	0.68	0.65	0.66	0.67	0.67	0.67

Table A.8. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 15 and pp is 0.5.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.84	0.83	0.85	0.86	0.84	0.84	0.83
		dce-GMDH	0.86	0.86	0.86	0.86	0.86	0.86	0.85
		svm	0.86	0.86	0.86	0.87	0.86	0.86	0.85
		random forest	0.86	0.87	0.86	0.87	0.87	0.86	0.85
		naive bayes	0.87	0.87	0.88	0.88	0.87	0.87	0.86
		elastic net	0.86	0.86	0.86	0.87	0.86	0.86	0.85
		neural network	0.84	0.85	0.82	0.83	0.85	0.83	0.82
	100	GMDH	0.85	0.84	0.86	0.86	0.85	0.85	0.84
		dce-GMDH	0.87	0.87	0.86	0.87	0.87	0.87	0.86
		svm	0.87	0.87	0.87	0.87	0.87	0.87	0.86
		random forest	0.87	0.87	0.87	0.87	0.88	0.87	0.86
		naive bayes	0.88	0.88	0.87	0.88	0.88	0.88	0.87
		elastic net	0.87	0.87	0.87	0.88	0.87	0.87	0.86
		neural network	0.85	0.86	0.84	0.84	0.86	0.85	0.84
	500	GMDH	0.87	0.86	0.88	0.88	0.86	0.87	0.87
		dce-GMDH	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		svm	0.88	0.88	0.88	0.88	0.88	0.88	0.88
		random forest	0.88	0.88	0.88	0.88	0.88	0.88	0.88
		naive bayes	0.88	0.88	0.88	0.88	0.88	0.88	0.88
		elastic net	0.90	0.90	0.90	0.90	0.90	0.90	0.90
		neural network	0.87	0.88	0.87	0.87	0.88	0.87	0.87
	1000	GMDH	0.87	0.86	0.89	0.88	0.87	0.87	0.87
		dce-GMDH	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		svm	0.89	0.89	0.89	0.89	0.89	0.89	0.89
		random forest	0.89	0.89	0.89	0.89	0.89	0.89	0.89
		naive bayes	0.88	0.88	0.88	0.88	0.88	0.88	0.88
		elastic net	0.91	0.91	0.91	0.91	0.91	0.91	0.91
		neural network	0.88	0.88	0.88	0.88	0.88	0.88	0.88

Table A.9. Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 5 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.68	0.21	0.89	0.46	0.73	0.55	0.38
		dce-GMDH	0.67	0.26	0.85	0.43	0.73	0.55	0.37
		svm	0.68	0.11	0.93	0.40	0.71	0.52	0.32
		random forest	0.67	0.28	0.84	0.43	0.73	0.56	0.35
		naive bayes	0.66	0.37	0.78	0.44	0.75	0.58	0.39
		elastic net	0.69	0.16	0.92	0.47	0.72	0.54	0.39
		neural network	0.61	0.37	0.72	0.36	0.73	0.54	0.35
	100	GMDH	0.70	0.18	0.92	0.51	0.73	0.55	0.32
		dce-GMDH	0.69	0.25	0.88	0.49	0.74	0.57	0.35
		svm	0.70	0.10	0.95	0.50	0.72	0.53	0.31
		random forest	0.68	0.28	0.85	0.45	0.74	0.57	0.33
		naive bayes	0.70	0.33	0.85	0.49	0.75	0.59	0.38
		elastic net	0.70	0.17	0.93	0.54	0.73	0.55	0.35
		neural network	0.64	0.38	0.75	0.39	0.74	0.56	0.36
	500	GMDH	0.71	0.15	0.95	0.58	0.72	0.55	0.24
		dce-GMDH	0.72	0.28	0.91	0.57	0.75	0.59	0.36
		svm	0.71	0.16	0.95	0.59	0.73	0.55	0.26
		random forest	0.70	0.29	0.87	0.50	0.74	0.58	0.35
		naive bayes	0.72	0.31	0.89	0.57	0.75	0.60	0.39
		elastic net	0.72	0.26	0.92	0.59	0.74	0.59	0.35
		neural network	0.69	0.34	0.84	0.48	0.75	0.59	0.38
	1000	GMDH	0.71	0.16	0.95	0.59	0.73	0.55	0.24
		dce-GMDH	0.72	0.29	0.91	0.58	0.75	0.60	0.38
		svm	0.72	0.18	0.95	0.61	0.73	0.56	0.27
		random forest	0.70	0.28	0.88	0.51	0.74	0.58	0.36
		naive bayes	0.72	0.31	0.90	0.57	0.75	0.61	0.40
		elastic net	0.72	0.27	0.92	0.59	0.75	0.60	0.37
		neural network	0.71	0.32	0.87	0.53	0.75	0.60	0.39
Medium	50	GMDH	0.80	0.58	0.89	0.71	0.84	0.73	0.64
		dce-GMDH	0.84	0.68	0.90	0.76	0.87	0.79	0.71
		svm	0.82	0.56	0.93	0.79	0.84	0.74	0.69
		random forest	0.80	0.55	0.91	0.75	0.83	0.73	0.64
		naive bayes	0.78	0.68	0.83	0.68	0.86	0.75	0.65
		elastic net	0.86	0.71	0.92	0.80	0.89	0.82	0.75
		neural network	0.78	0.61	0.85	0.64	0.84	0.73	0.62
	100	GMDH	0.82	0.61	0.91	0.75	0.85	0.76	0.65
		dce-GMDH	0.87	0.75	0.92	0.82	0.90	0.84	0.76
		svm	0.85	0.68	0.93	0.82	0.87	0.80	0.72
		random forest	0.83	0.61	0.92	0.79	0.85	0.77	0.66
		naive bayes	0.84	0.68	0.91	0.80	0.87	0.80	0.71
		elastic net	0.88	0.77	0.93	0.84	0.91	0.85	0.78
		neural network	0.84	0.71	0.90	0.76	0.88	0.80	0.72
	500	GMDH	0.84	0.66	0.92	0.79	0.86	0.79	0.71
		dce-GMDH	0.90	0.81	0.94	0.85	0.92	0.88	0.83
		svm	0.89	0.78	0.94	0.85	0.91	0.86	0.81
		random forest	0.87	0.72	0.94	0.83	0.89	0.83	0.76
		naive bayes	0.88	0.72	0.95	0.87	0.89	0.84	0.78
		elastic net	0.90	0.82	0.94	0.85	0.92	0.88	0.83
		neural network	0.89	0.79	0.93	0.83	0.91	0.86	0.81
	1000	GMDH	0.85	0.67	0.93	0.79	0.87	0.80	0.72
		dce-GMDH	0.90	0.82	0.94	0.86	0.92	0.88	0.84
		svm	0.90	0.80	0.94	0.86	0.92	0.87	0.82
		random forest	0.88	0.74	0.94	0.84	0.90	0.84	0.79
		naive bayes	0.89	0.73	0.96	0.88	0.89	0.84	0.80
		elastic net	0.91	0.82	0.94	0.86	0.93	0.88	0.84
		neural network	0.90	0.81	0.93	0.84	0.92	0.87	0.82

Table A.9. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 5 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.81	0.62	0.89	0.73	0.85	0.76	0.67
		dce-GMDH	0.85	0.73	0.91	0.78	0.89	0.82	0.74
		svm	0.84	0.62	0.93	0.81	0.86	0.78	0.72
		random forest	0.83	0.61	0.92	0.78	0.85	0.77	0.69
		naive bayes	0.80	0.74	0.83	0.69	0.89	0.78	0.68
		elastic net	0.87	0.75	0.92	0.82	0.90	0.84	0.77
		neural network	0.80	0.65	0.86	0.67	0.86	0.76	0.65
	100	GMDH	0.83	0.65	0.91	0.78	0.86	0.78	0.68
		dce-GMDH	0.89	0.78	0.93	0.83	0.91	0.85	0.79
		svm	0.87	0.72	0.94	0.84	0.89	0.83	0.75
		random forest	0.85	0.67	0.93	0.82	0.87	0.80	0.71
		naive bayes	0.86	0.74	0.91	0.80	0.89	0.83	0.75
		elastic net	0.90	0.80	0.94	0.85	0.92	0.87	0.81
		neural network	0.86	0.74	0.91	0.78	0.89	0.82	0.75
	500	GMDH	0.86	0.70	0.93	0.81	0.88	0.82	0.75
		dce-GMDH	0.91	0.84	0.95	0.87	0.93	0.89	0.85
		svm	0.91	0.81	0.95	0.87	0.92	0.88	0.83
		random forest	0.89	0.76	0.94	0.85	0.90	0.85	0.80
		naive bayes	0.90	0.78	0.95	0.87	0.91	0.86	0.82
		elastic net	0.92	0.84	0.95	0.87	0.93	0.89	0.85
		neural network	0.90	0.82	0.94	0.85	0.92	0.88	0.83
	1000	GMDH	0.87	0.71	0.93	0.82	0.88	0.82	0.76
		dce-GMDH	0.92	0.85	0.95	0.87	0.93	0.90	0.86
		svm	0.91	0.83	0.95	0.87	0.93	0.89	0.85
		random forest	0.89	0.78	0.94	0.86	0.91	0.86	0.81
		naive bayes	0.90	0.79	0.95	0.88	0.91	0.87	0.83
		elastic net	0.92	0.85	0.95	0.87	0.94	0.90	0.86
		neural network	0.91	0.83	0.94	0.86	0.93	0.89	0.85

Table A.10. Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 10 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.70	0.35	0.85	0.52	0.76	0.60	0.44
		dce-GMDH	0.69	0.33	0.85	0.50	0.75	0.59	0.42
		svm	0.70	0.17	0.93	0.52	0.72	0.55	0.41
		random forest	0.70	0.29	0.88	0.54	0.74	0.59	0.41
		naive bayes	0.69	0.43	0.80	0.50	0.77	0.62	0.45
		elastic net	0.70	0.26	0.90	0.54	0.74	0.58	0.45
		neural network	0.62	0.38	0.72	0.37	0.73	0.55	0.35
	100	GMDH	0.72	0.34	0.88	0.57	0.76	0.61	0.41
		dce-GMDH	0.72	0.37	0.87	0.57	0.77	0.62	0.44
		svm	0.72	0.23	0.93	0.60	0.74	0.58	0.40
		random forest	0.72	0.31	0.90	0.58	0.75	0.60	0.39
		naive bayes	0.73	0.45	0.85	0.57	0.79	0.65	0.48
		elastic net	0.73	0.32	0.90	0.61	0.76	0.61	0.44
		neural network	0.64	0.40	0.75	0.41	0.74	0.57	0.38
	500	GMDH	0.74	0.33	0.91	0.63	0.76	0.62	0.42
		dce-GMDH	0.76	0.47	0.89	0.65	0.80	0.68	0.53
		svm	0.75	0.37	0.91	0.66	0.77	0.64	0.46
		random forest	0.74	0.37	0.91	0.63	0.77	0.64	0.45
		naive bayes	0.76	0.49	0.88	0.64	0.80	0.69	0.55
		elastic net	0.77	0.46	0.90	0.66	0.80	0.68	0.54
		neural network	0.70	0.47	0.80	0.51	0.78	0.64	0.49
	1000	GMDH	0.74	0.33	0.92	0.64	0.76	0.63	0.43
		dce-GMDH	0.77	0.48	0.89	0.66	0.80	0.69	0.56
		svm	0.76	0.39	0.92	0.67	0.78	0.65	0.49
		random forest	0.75	0.39	0.91	0.65	0.78	0.65	0.48
		naive bayes	0.77	0.50	0.89	0.66	0.81	0.69	0.56
		elastic net	0.77	0.48	0.90	0.67	0.80	0.69	0.55
		neural network	0.72	0.49	0.82	0.55	0.79	0.66	0.52
Medium	50	GMDH	0.78	0.57	0.87	0.68	0.83	0.72	0.61
		dce-GMDH	0.82	0.62	0.90	0.74	0.85	0.76	0.67
		svm	0.81	0.55	0.92	0.77	0.84	0.74	0.68
		random forest	0.79	0.47	0.93	0.77	0.81	0.70	0.61
		naive bayes	0.79	0.63	0.86	0.69	0.85	0.75	0.64
		elastic net	0.83	0.64	0.91	0.76	0.86	0.77	0.69
		neural network	0.68	0.48	0.76	0.47	0.78	0.62	0.45
	100	GMDH	0.80	0.59	0.89	0.72	0.84	0.74	0.62
		dce-GMDH	0.86	0.73	0.92	0.80	0.89	0.82	0.74
		svm	0.86	0.69	0.93	0.81	0.88	0.81	0.72
		random forest	0.82	0.52	0.94	0.81	0.83	0.73	0.61
		naive bayes	0.85	0.68	0.92	0.80	0.87	0.80	0.71
		elastic net	0.87	0.75	0.93	0.82	0.90	0.84	0.76
		neural network	0.75	0.55	0.83	0.58	0.82	0.69	0.55
	500	GMDH	0.83	0.63	0.92	0.77	0.85	0.78	0.69
		dce-GMDH	0.91	0.83	0.94	0.86	0.93	0.88	0.84
		svm	0.90	0.80	0.94	0.85	0.92	0.87	0.82
		random forest	0.86	0.64	0.95	0.86	0.86	0.80	0.73
		naive bayes	0.90	0.78	0.95	0.88	0.91	0.86	0.82
		elastic net	0.91	0.83	0.94	0.86	0.93	0.89	0.85
		neural network	0.87	0.75	0.92	0.79	0.90	0.83	0.77
	1000	GMDH	0.84	0.64	0.92	0.78	0.86	0.78	0.70
		dce-GMDH	0.91	0.84	0.95	0.87	0.93	0.89	0.85
		svm	0.90	0.81	0.94	0.86	0.92	0.88	0.83
		random forest	0.87	0.68	0.96	0.87	0.87	0.82	0.76
		naive bayes	0.91	0.79	0.96	0.89	0.91	0.87	0.83
		elastic net	0.92	0.84	0.95	0.87	0.93	0.89	0.86
		neural network	0.88	0.78	0.93	0.82	0.91	0.85	0.81

Table A.10. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 10 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.80	0.61	0.88	0.71	0.85	0.75	0.65
		dce-GMDH	0.83	0.67	0.90	0.76	0.87	0.79	0.70
		svm	0.83	0.62	0.93	0.80	0.86	0.77	0.72
		random forest	0.81	0.53	0.94	0.80	0.83	0.73	0.66
		naive bayes	0.81	0.70	0.86	0.71	0.87	0.78	0.69
		elastic net	0.85	0.68	0.92	0.79	0.88	0.80	0.73
		neural network	0.69	0.51	0.77	0.50	0.79	0.64	0.48
	100	GMDH	0.82	0.64	0.90	0.75	0.86	0.77	0.66
		dce-GMDH	0.88	0.76	0.93	0.82	0.90	0.84	0.77
		svm	0.87	0.74	0.93	0.83	0.89	0.84	0.76
		random forest	0.84	0.59	0.94	0.83	0.85	0.77	0.67
		naive bayes	0.87	0.75	0.92	0.81	0.90	0.84	0.76
		elastic net	0.88	0.78	0.93	0.83	0.91	0.85	0.78
		neural network	0.77	0.60	0.84	0.62	0.83	0.72	0.59
	500	GMDH	0.85	0.68	0.92	0.79	0.87	0.80	0.73
		dce-GMDH	0.92	0.85	0.95	0.88	0.94	0.90	0.86
		svm	0.91	0.82	0.94	0.87	0.93	0.88	0.84
		random forest	0.88	0.69	0.96	0.87	0.88	0.82	0.77
		naive bayes	0.91	0.83	0.95	0.87	0.93	0.89	0.85
		elastic net	0.92	0.85	0.95	0.88	0.94	0.90	0.86
		neural network	0.88	0.77	0.92	0.82	0.91	0.85	0.80
	1000	GMDH	0.86	0.69	0.93	0.80	0.88	0.81	0.74
		dce-GMDH	0.92	0.86	0.95	0.88	0.94	0.91	0.87
		svm	0.92	0.84	0.95	0.87	0.93	0.89	0.85
		random forest	0.89	0.73	0.96	0.88	0.89	0.84	0.79
		naive bayes	0.92	0.85	0.95	0.88	0.94	0.90	0.86
		elastic net	0.93	0.86	0.95	0.88	0.94	0.91	0.87
		neural network	0.90	0.80	0.93	0.84	0.92	0.87	0.83

Table A.11. Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 15 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.70	0.37	0.84	0.50	0.76	0.60	0.43
		dce-GMDH	0.70	0.33	0.86	0.52	0.76	0.60	0.44
		svm	0.70	0.17	0.93	0.52	0.73	0.55	0.41
		random forest	0.71	0.24	0.91	0.57	0.74	0.58	0.42
		naive bayes	0.69	0.45	0.79	0.50	0.78	0.62	0.45
		elastic net	0.71	0.25	0.90	0.54	0.75	0.58	0.44
		neural network	0.60	0.37	0.70	0.34	0.73	0.54	0.34
	100	GMDH	0.71	0.34	0.87	0.56	0.76	0.61	0.41
		dce-GMDH	0.73	0.38	0.88	0.58	0.77	0.63	0.44
		svm	0.73	0.26	0.93	0.63	0.75	0.60	0.43
		random forest	0.73	0.27	0.92	0.62	0.75	0.59	0.38
		naive bayes	0.74	0.46	0.85	0.58	0.79	0.66	0.49
		elastic net	0.73	0.33	0.90	0.60	0.76	0.61	0.44
		neural network	0.63	0.38	0.73	0.38	0.73	0.55	0.36
	500	GMDH	0.73	0.31	0.91	0.61	0.76	0.61	0.40
		dce-GMDH	0.77	0.50	0.89	0.66	0.81	0.69	0.56
		svm	0.76	0.43	0.91	0.67	0.79	0.67	0.51
		random forest	0.75	0.33	0.93	0.69	0.77	0.63	0.44
		naive bayes	0.77	0.54	0.88	0.65	0.82	0.71	0.58
		elastic net	0.77	0.49	0.90	0.67	0.80	0.69	0.56
		neural network	0.69	0.44	0.79	0.48	0.77	0.62	0.46
	1000	GMDH	0.74	0.31	0.92	0.63	0.76	0.61	0.40
		dce-GMDH	0.78	0.52	0.89	0.68	0.81	0.71	0.59
		svm	0.77	0.45	0.91	0.68	0.79	0.68	0.53
		random forest	0.76	0.35	0.94	0.70	0.77	0.64	0.46
		naive bayes	0.78	0.55	0.88	0.67	0.82	0.71	0.60
		elastic net	0.78	0.51	0.90	0.68	0.81	0.70	0.58
		neural network	0.71	0.48	0.81	0.53	0.79	0.65	0.51
Medium	50	GMDH	0.76	0.54	0.86	0.64	0.82	0.70	0.58
		dce-GMDH	0.81	0.61	0.90	0.75	0.85	0.76	0.67
		svm	0.81	0.56	0.92	0.78	0.84	0.74	0.69
		random forest	0.79	0.42	0.95	0.80	0.80	0.68	0.61
		naive bayes	0.78	0.64	0.85	0.68	0.85	0.74	0.63
		elastic net	0.82	0.60	0.91	0.75	0.85	0.75	0.67
		neural network	0.64	0.46	0.72	0.42	0.76	0.59	0.41
	100	GMDH	0.78	0.55	0.88	0.69	0.82	0.72	0.58
		dce-GMDH	0.86	0.71	0.92	0.81	0.88	0.82	0.73
		svm	0.86	0.71	0.93	0.81	0.88	0.82	0.73
		random forest	0.81	0.47	0.96	0.86	0.81	0.72	0.59
		naive bayes	0.85	0.68	0.92	0.80	0.87	0.80	0.71
		elastic net	0.86	0.73	0.92	0.81	0.89	0.82	0.74
		neural network	0.68	0.47	0.77	0.48	0.78	0.62	0.45
	500	GMDH	0.81	0.58	0.91	0.74	0.83	0.74	0.64
		dce-GMDH	0.91	0.84	0.95	0.87	0.93	0.89	0.85
		svm	0.90	0.81	0.94	0.86	0.92	0.88	0.83
		random forest	0.86	0.60	0.97	0.90	0.85	0.78	0.71
		naive bayes	0.91	0.81	0.95	0.87	0.92	0.88	0.84
		elastic net	0.92	0.84	0.95	0.87	0.93	0.89	0.85
		neural network	0.84	0.70	0.91	0.76	0.88	0.80	0.74
	1000	GMDH	0.82	0.59	0.92	0.75	0.84	0.75	0.65
		dce-GMDH	0.92	0.85	0.95	0.88	0.94	0.90	0.86
		svm	0.91	0.83	0.95	0.87	0.93	0.89	0.85
		random forest	0.87	0.64	0.97	0.91	0.86	0.80	0.74
		naive bayes	0.92	0.83	0.95	0.88	0.93	0.89	0.85
		elastic net	0.92	0.86	0.95	0.88	0.94	0.90	0.87
		neural network	0.87	0.75	0.93	0.82	0.90	0.84	0.79

Table A.11. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are low, p is 15 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.78	0.59	0.86	0.66	0.84	0.72	0.61
		dce-GMDH	0.83	0.66	0.91	0.76	0.87	0.78	0.70
		svm	0.84	0.63	0.93	0.80	0.86	0.78	0.72
		random forest	0.81	0.47	0.95	0.83	0.81	0.71	0.65
		naive bayes	0.80	0.69	0.85	0.69	0.87	0.77	0.67
		elastic net	0.83	0.65	0.91	0.76	0.86	0.78	0.70
		neural network	0.65	0.48	0.73	0.43	0.77	0.60	0.43
	100	GMDH	0.80	0.59	0.89	0.71	0.84	0.74	0.61
		dce-GMDH	0.87	0.74	0.93	0.82	0.90	0.84	0.76
		svm	0.87	0.74	0.93	0.83	0.90	0.84	0.76
		random forest	0.83	0.52	0.96	0.87	0.83	0.74	0.63
		naive bayes	0.86	0.73	0.92	0.81	0.89	0.83	0.74
		elastic net	0.87	0.75	0.93	0.82	0.90	0.84	0.76
		neural network	0.70	0.51	0.79	0.50	0.79	0.65	0.48
	500	GMDH	0.82	0.61	0.91	0.75	0.85	0.76	0.67
		dce-GMDH	0.92	0.85	0.95	0.88	0.94	0.90	0.86
		svm	0.91	0.83	0.95	0.87	0.93	0.89	0.85
		random forest	0.87	0.64	0.97	0.91	0.86	0.81	0.75
		naive bayes	0.92	0.85	0.95	0.87	0.94	0.90	0.86
		elastic net	0.92	0.86	0.95	0.88	0.94	0.90	0.87
		neural network	0.86	0.72	0.91	0.78	0.89	0.82	0.76
	1000	GMDH	0.83	0.62	0.92	0.76	0.85	0.77	0.68
		dce-GMDH	0.93	0.87	0.95	0.89	0.94	0.91	0.88
		svm	0.92	0.85	0.95	0.88	0.93	0.90	0.86
		random forest	0.88	0.68	0.97	0.91	0.88	0.83	0.78
		naive bayes	0.92	0.86	0.95	0.88	0.94	0.91	0.87
		elastic net	0.93	0.87	0.95	0.89	0.94	0.91	0.88
		neural network	0.88	0.77	0.93	0.83	0.91	0.85	0.81

Table A.12. Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 5 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.67	0.16	0.89	0.40	0.72	0.53	0.33
		dce-GMDH	0.66	0.21	0.85	0.38	0.72	0.53	0.33
		svm	0.67	0.09	0.92	0.35	0.71	0.51	0.29
		random forest	0.65	0.25	0.82	0.38	0.72	0.54	0.31
		naive bayes	0.64	0.38	0.75	0.39	0.74	0.57	0.37
		elastic net	0.69	0.11	0.93	0.43	0.71	0.52	0.35
		neural network	0.60	0.34	0.72	0.34	0.72	0.53	0.33
	100	GMDH	0.69	0.11	0.94	0.45	0.71	0.52	0.27
		dce-GMDH	0.67	0.19	0.88	0.42	0.72	0.54	0.31
		svm	0.69	0.06	0.96	0.40	0.70	0.51	0.25
		random forest	0.66	0.24	0.84	0.39	0.72	0.54	0.28
		naive bayes	0.66	0.36	0.79	0.42	0.74	0.57	0.37
		elastic net	0.69	0.08	0.96	0.47	0.71	0.52	0.28
		neural network	0.62	0.32	0.74	0.35	0.72	0.53	0.32
	500	GMDH	0.70	0.05	0.98	0.53	0.71	0.51	0.14
		dce-GMDH	0.69	0.13	0.94	0.50	0.72	0.54	0.22
		svm	0.70	0.05	0.98	0.50	0.71	0.51	0.15
		random forest	0.67	0.21	0.87	0.41	0.72	0.54	0.27
		naive bayes	0.67	0.40	0.79	0.45	0.75	0.59	0.41
		elastic net	0.70	0.07	0.97	0.53	0.71	0.52	0.16
		neural network	0.66	0.22	0.86	0.38	0.72	0.54	0.27
	1000	GMDH	0.70	0.04	0.98	0.56	0.71	0.51	0.10
		dce-GMDH	0.70	0.11	0.95	0.53	0.72	0.53	0.18
		svm	0.70	0.05	0.98	0.53	0.71	0.51	0.13
		random forest	0.68	0.18	0.89	0.43	0.72	0.54	0.25
		naive bayes	0.67	0.40	0.79	0.45	0.75	0.59	0.42
		elastic net	0.70	0.08	0.97	0.55	0.71	0.52	0.14
		neural network	0.68	0.17	0.90	0.41	0.72	0.53	0.23
Medium	50	GMDH	0.74	0.44	0.87	0.62	0.79	0.65	0.54
		dce-GMDH	0.74	0.51	0.84	0.60	0.81	0.68	0.55
		svm	0.73	0.30	0.92	0.63	0.76	0.61	0.53
		random forest	0.74	0.49	0.86	0.61	0.80	0.67	0.54
		naive bayes	0.73	0.70	0.75	0.56	0.86	0.72	0.59
		elastic net	0.75	0.42	0.89	0.65	0.79	0.66	0.56
		neural network	0.68	0.49	0.77	0.48	0.78	0.63	0.47
	100	GMDH	0.75	0.44	0.89	0.66	0.79	0.67	0.51
		dce-GMDH	0.76	0.54	0.86	0.63	0.82	0.70	0.55
		svm	0.75	0.37	0.92	0.68	0.78	0.64	0.51
		random forest	0.75	0.50	0.86	0.62	0.80	0.68	0.53
		naive bayes	0.76	0.69	0.79	0.59	0.86	0.74	0.61
		elastic net	0.77	0.47	0.90	0.68	0.80	0.68	0.54
		neural network	0.72	0.52	0.80	0.54	0.80	0.66	0.50
	500	GMDH	0.77	0.48	0.90	0.68	0.80	0.69	0.55
		dce-GMDH	0.78	0.54	0.88	0.67	0.82	0.71	0.59
		svm	0.78	0.47	0.91	0.70	0.80	0.69	0.55
		random forest	0.77	0.52	0.87	0.64	0.81	0.70	0.57
		naive bayes	0.77	0.69	0.80	0.60	0.86	0.75	0.64
		elastic net	0.78	0.53	0.89	0.69	0.82	0.71	0.59
		neural network	0.76	0.55	0.85	0.62	0.82	0.70	0.57
	1000	GMDH	0.78	0.49	0.90	0.69	0.80	0.69	0.56
		dce-GMDH	0.79	0.54	0.89	0.68	0.82	0.72	0.60
		svm	0.78	0.48	0.91	0.71	0.80	0.70	0.56
		random forest	0.77	0.53	0.88	0.65	0.81	0.70	0.58
		naive bayes	0.77	0.70	0.80	0.61	0.86	0.75	0.65
		elastic net	0.79	0.54	0.89	0.69	0.82	0.72	0.60
		neural network	0.77	0.54	0.87	0.65	0.82	0.71	0.59

Table A.12. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 5 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.86	0.74	0.90	0.79	0.90	0.82	0.75
		dce-GMDH	0.89	0.84	0.92	0.83	0.93	0.88	0.82
		svm	0.90	0.77	0.95	0.88	0.91	0.86	0.82
		random forest	0.89	0.77	0.94	0.85	0.91	0.85	0.80
		naive bayes	0.83	0.92	0.79	0.67	0.96	0.86	0.76
		elastic net	0.91	0.85	0.94	0.86	0.94	0.89	0.84
		neural network	0.84	0.75	0.88	0.74	0.90	0.82	0.74
	100	GMDH	0.88	0.78	0.92	0.82	0.91	0.85	0.78
		dce-GMDH	0.92	0.86	0.94	0.87	0.94	0.90	0.85
		svm	0.92	0.82	0.96	0.90	0.93	0.89	0.84
		random forest	0.90	0.81	0.94	0.87	0.92	0.88	0.82
		naive bayes	0.87	0.91	0.85	0.74	0.95	0.88	0.80
		elastic net	0.93	0.88	0.95	0.89	0.95	0.91	0.87
		neural network	0.90	0.82	0.93	0.83	0.93	0.88	0.82
	500	GMDH	0.91	0.83	0.95	0.88	0.93	0.89	0.85
		dce-GMDH	0.94	0.90	0.96	0.91	0.96	0.93	0.90
		svm	0.94	0.88	0.96	0.91	0.95	0.92	0.89
		random forest	0.93	0.85	0.96	0.90	0.94	0.91	0.87
		naive bayes	0.91	0.88	0.92	0.82	0.95	0.90	0.85
		elastic net	0.94	0.90	0.96	0.91	0.96	0.93	0.91
		neural network	0.93	0.88	0.95	0.89	0.95	0.92	0.89
	1000	GMDH	0.92	0.84	0.96	0.89	0.94	0.90	0.86
		dce-GMDH	0.95	0.90	0.96	0.91	0.96	0.93	0.91
		svm	0.94	0.89	0.97	0.92	0.95	0.93	0.90
		random forest	0.93	0.87	0.96	0.90	0.94	0.91	0.88
		naive bayes	0.91	0.88	0.92	0.82	0.95	0.90	0.85
		elastic net	0.95	0.91	0.96	0.91	0.96	0.94	0.91
		neural network	0.94	0.89	0.96	0.91	0.95	0.93	0.90

Table A.13. Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 10 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.67	0.22	0.86	0.41	0.72	0.54	0.33
		dce-GMDH	0.66	0.23	0.85	0.40	0.72	0.54	0.34
		svm	0.68	0.10	0.92	0.36	0.71	0.51	0.29
		random forest	0.67	0.24	0.85	0.42	0.73	0.55	0.34
		naive bayes	0.64	0.43	0.73	0.41	0.75	0.58	0.39
		elastic net	0.68	0.13	0.92	0.44	0.72	0.53	0.35
		neural network	0.60	0.35	0.71	0.34	0.72	0.53	0.33
	100	GMDH	0.69	0.16	0.91	0.46	0.72	0.54	0.28
		dce-GMDH	0.68	0.21	0.88	0.44	0.72	0.54	0.32
		svm	0.69	0.07	0.96	0.43	0.71	0.51	0.25
		random forest	0.68	0.23	0.87	0.43	0.73	0.55	0.29
		naive bayes	0.65	0.45	0.74	0.43	0.76	0.60	0.41
		elastic net	0.70	0.10	0.95	0.49	0.71	0.53	0.29
		neural network	0.61	0.34	0.73	0.35	0.72	0.53	0.32
	500	GMDH	0.70	0.10	0.96	0.55	0.71	0.53	0.18
		dce-GMDH	0.70	0.15	0.94	0.53	0.72	0.54	0.23
		svm	0.70	0.07	0.97	0.54	0.71	0.52	0.17
		random forest	0.69	0.21	0.90	0.46	0.73	0.55	0.28
		naive bayes	0.66	0.50	0.73	0.44	0.77	0.61	0.46
		elastic net	0.70	0.11	0.96	0.56	0.72	0.54	0.20
		neural network	0.63	0.34	0.75	0.37	0.73	0.55	0.35
	1000	GMDH	0.71	0.10	0.97	0.57	0.71	0.53	0.16
		dce-GMDH	0.71	0.15	0.95	0.55	0.72	0.55	0.22
		svm	0.70	0.08	0.97	0.55	0.71	0.53	0.16
		random forest	0.69	0.20	0.90	0.47	0.73	0.55	0.28
		naive bayes	0.66	0.50	0.72	0.44	0.77	0.61	0.47
		elastic net	0.71	0.12	0.96	0.57	0.72	0.54	0.20
		neural network	0.65	0.31	0.79	0.38	0.73	0.55	0.34
Medium	50	GMDH	0.75	0.51	0.86	0.62	0.81	0.68	0.56
		dce-GMDH	0.76	0.55	0.85	0.62	0.82	0.70	0.57
		svm	0.75	0.37	0.91	0.66	0.78	0.64	0.56
		random forest	0.77	0.51	0.88	0.66	0.81	0.69	0.57
		naive bayes	0.76	0.73	0.77	0.59	0.87	0.75	0.63
		elastic net	0.76	0.47	0.89	0.66	0.80	0.68	0.57
		neural network	0.68	0.49	0.76	0.47	0.78	0.63	0.46
	100	GMDH	0.77	0.51	0.88	0.67	0.81	0.69	0.55
		dce-GMDH	0.78	0.57	0.86	0.66	0.83	0.72	0.58
		svm	0.77	0.44	0.91	0.71	0.80	0.68	0.55
		random forest	0.78	0.53	0.88	0.67	0.82	0.71	0.57
		naive bayes	0.78	0.73	0.80	0.62	0.87	0.76	0.65
		elastic net	0.78	0.51	0.89	0.69	0.81	0.70	0.56
		neural network	0.70	0.50	0.79	0.51	0.79	0.65	0.48
	500	GMDH	0.79	0.54	0.90	0.70	0.82	0.72	0.60
		dce-GMDH	0.80	0.59	0.89	0.70	0.83	0.74	0.63
		svm	0.79	0.52	0.91	0.73	0.82	0.72	0.60
		random forest	0.79	0.56	0.89	0.69	0.83	0.73	0.61
		naive bayes	0.79	0.74	0.81	0.63	0.88	0.78	0.68
		elastic net	0.80	0.58	0.90	0.71	0.83	0.74	0.63
		neural network	0.75	0.56	0.83	0.58	0.82	0.69	0.57
	1000	GMDH	0.79	0.55	0.90	0.70	0.82	0.73	0.61
		dce-GMDH	0.80	0.59	0.89	0.71	0.84	0.74	0.64
		svm	0.80	0.52	0.92	0.73	0.82	0.72	0.61
		random forest	0.80	0.57	0.89	0.70	0.83	0.73	0.62
		naive bayes	0.79	0.75	0.81	0.63	0.88	0.78	0.68
		elastic net	0.80	0.59	0.90	0.71	0.84	0.74	0.64
		neural network	0.76	0.58	0.84	0.61	0.82	0.71	0.59

Table A.13. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 10 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.85	0.75	0.89	0.77	0.90	0.82	0.75
		dce-GMDH	0.89	0.84	0.92	0.82	0.93	0.88	0.82
		svm	0.91	0.80	0.95	0.89	0.92	0.88	0.84
		random forest	0.89	0.75	0.95	0.88	0.90	0.85	0.80
		naive bayes	0.87	0.93	0.84	0.73	0.97	0.89	0.80
		elastic net	0.90	0.82	0.93	0.85	0.93	0.88	0.83
		neural network	0.79	0.68	0.83	0.64	0.87	0.76	0.64
	100	GMDH	0.87	0.78	0.91	0.80	0.91	0.85	0.77
		dce-GMDH	0.92	0.86	0.94	0.87	0.94	0.90	0.85
		svm	0.92	0.84	0.96	0.90	0.94	0.90	0.86
		random forest	0.91	0.79	0.96	0.89	0.92	0.87	0.82
		naive bayes	0.90	0.92	0.90	0.80	0.96	0.91	0.84
		elastic net	0.92	0.86	0.95	0.88	0.94	0.91	0.86
		neural network	0.83	0.74	0.87	0.72	0.89	0.81	0.71
	500	GMDH	0.90	0.83	0.93	0.85	0.93	0.88	0.83
		dce-GMDH	0.94	0.91	0.96	0.91	0.96	0.93	0.91
		svm	0.94	0.89	0.97	0.92	0.95	0.93	0.90
		random forest	0.93	0.85	0.96	0.91	0.94	0.91	0.87
		naive bayes	0.94	0.93	0.94	0.87	0.97	0.93	0.89
		elastic net	0.95	0.91	0.96	0.92	0.96	0.94	0.91
		neural network	0.92	0.84	0.95	0.87	0.94	0.89	0.86
	1000	GMDH	0.91	0.84	0.94	0.86	0.93	0.89	0.85
		dce-GMDH	0.95	0.92	0.96	0.92	0.96	0.94	0.92
		svm	0.95	0.90	0.97	0.92	0.96	0.93	0.91
		random forest	0.93	0.86	0.97	0.92	0.94	0.91	0.89
		naive bayes	0.94	0.94	0.94	0.87	0.97	0.94	0.90
		elastic net	0.95	0.92	0.97	0.92	0.96	0.94	0.92
		neural network	0.93	0.86	0.95	0.89	0.94	0.91	0.89

Table A.14. Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 15 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.66	0.24	0.84	0.40	0.72	0.54	0.33
		dce-GMDH	0.66	0.23	0.84	0.40	0.72	0.54	0.34
		svm	0.68	0.09	0.93	0.35	0.70	0.51	0.28
		random forest	0.67	0.22	0.87	0.43	0.72	0.54	0.33
		naive bayes	0.63	0.43	0.72	0.40	0.75	0.57	0.39
		elastic net	0.68	0.13	0.92	0.43	0.71	0.52	0.35
		neural network	0.60	0.35	0.71	0.34	0.72	0.53	0.32
	100	GMDH	0.68	0.18	0.90	0.44	0.72	0.54	0.28
		dce-GMDH	0.68	0.20	0.88	0.44	0.72	0.54	0.32
		svm	0.69	0.07	0.96	0.44	0.71	0.51	0.26
		random forest	0.68	0.21	0.88	0.44	0.72	0.55	0.29
		naive bayes	0.64	0.47	0.72	0.42	0.76	0.59	0.42
		elastic net	0.69	0.10	0.95	0.47	0.71	0.52	0.29
		neural network	0.61	0.34	0.72	0.35	0.72	0.53	0.32
	500	GMDH	0.70	0.10	0.96	0.55	0.71	0.53	0.18
		dce-GMDH	0.70	0.14	0.94	0.53	0.72	0.54	0.22
		svm	0.70	0.07	0.97	0.52	0.71	0.52	0.17
		random forest	0.69	0.18	0.91	0.48	0.72	0.55	0.25
		naive bayes	0.64	0.53	0.70	0.43	0.77	0.61	0.47
		elastic net	0.70	0.10	0.96	0.56	0.71	0.53	0.18
		neural network	0.62	0.34	0.74	0.36	0.72	0.54	0.35
	1000	GMDH	0.70	0.09	0.97	0.57	0.71	0.53	0.16
		dce-GMDH	0.70	0.14	0.95	0.55	0.72	0.54	0.21
		svm	0.70	0.08	0.97	0.55	0.71	0.52	0.15
		random forest	0.70	0.17	0.92	0.49	0.72	0.55	0.25
		naive bayes	0.64	0.53	0.69	0.43	0.77	0.61	0.47
		elastic net	0.71	0.11	0.96	0.57	0.72	0.54	0.18
		neural network	0.63	0.34	0.75	0.37	0.73	0.55	0.35
Medium	50	GMDH	0.76	0.54	0.85	0.63	0.82	0.70	0.57
		dce-GMDH	0.77	0.59	0.85	0.65	0.83	0.72	0.60
		svm	0.77	0.43	0.91	0.70	0.80	0.67	0.61
		random forest	0.78	0.53	0.89	0.70	0.82	0.71	0.61
		naive bayes	0.77	0.75	0.78	0.61	0.88	0.77	0.65
		elastic net	0.77	0.51	0.89	0.68	0.81	0.70	0.60
		neural network	0.69	0.53	0.76	0.49	0.79	0.65	0.49
	100	GMDH	0.78	0.55	0.88	0.68	0.82	0.71	0.58
		dce-GMDH	0.79	0.60	0.87	0.69	0.84	0.74	0.61
		svm	0.79	0.51	0.91	0.73	0.82	0.71	0.59
		random forest	0.80	0.56	0.90	0.71	0.83	0.73	0.60
		naive bayes	0.79	0.76	0.81	0.64	0.89	0.78	0.67
		elastic net	0.79	0.55	0.90	0.71	0.83	0.72	0.60
		neural network	0.71	0.53	0.79	0.53	0.80	0.66	0.51
	500	GMDH	0.80	0.57	0.90	0.71	0.83	0.73	0.62
		dce-GMDH	0.81	0.62	0.89	0.72	0.84	0.75	0.66
		svm	0.81	0.57	0.91	0.74	0.83	0.74	0.63
		random forest	0.81	0.59	0.90	0.72	0.84	0.75	0.65
		naive bayes	0.80	0.77	0.81	0.64	0.89	0.79	0.69
		elastic net	0.81	0.61	0.90	0.73	0.84	0.76	0.66
		neural network	0.75	0.56	0.83	0.59	0.82	0.69	0.58
	1000	GMDH	0.80	0.58	0.90	0.72	0.83	0.74	0.64
		dce-GMDH	0.82	0.62	0.90	0.73	0.85	0.76	0.67
		svm	0.81	0.57	0.92	0.75	0.83	0.74	0.64
		random forest	0.81	0.60	0.90	0.73	0.84	0.75	0.65
		naive bayes	0.80	0.78	0.81	0.64	0.89	0.79	0.70
		elastic net	0.82	0.62	0.90	0.73	0.85	0.76	0.67
		neural network	0.76	0.58	0.84	0.61	0.82	0.71	0.60

Table A.14. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are medium, p is 15 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.85	0.75	0.89	0.75	0.90	0.82	0.74
		dce-GMDH	0.90	0.86	0.92	0.83	0.94	0.89	0.84
		svm	0.92	0.83	0.95	0.89	0.93	0.89	0.86
		random forest	0.90	0.75	0.96	0.91	0.91	0.86	0.82
		naive bayes	0.89	0.94	0.86	0.75	0.97	0.90	0.83
		elastic net	0.90	0.83	0.94	0.85	0.93	0.88	0.83
		neural network	0.79	0.71	0.82	0.63	0.88	0.77	0.65
	100	GMDH	0.87	0.78	0.91	0.79	0.91	0.84	0.76
		dce-GMDH	0.93	0.88	0.95	0.88	0.95	0.91	0.87
		svm	0.93	0.87	0.96	0.91	0.94	0.91	0.88
		random forest	0.92	0.79	0.97	0.92	0.92	0.88	0.84
		naive bayes	0.92	0.93	0.92	0.83	0.97	0.92	0.87
		elastic net	0.92	0.86	0.95	0.89	0.94	0.91	0.86
		neural network	0.82	0.74	0.86	0.70	0.89	0.80	0.70
	500	GMDH	0.90	0.81	0.93	0.84	0.92	0.87	0.82
		dce-GMDH	0.95	0.92	0.96	0.92	0.97	0.94	0.92
		svm	0.95	0.91	0.97	0.93	0.96	0.94	0.92
		random forest	0.94	0.86	0.97	0.94	0.94	0.92	0.89
		naive bayes	0.94	0.96	0.94	0.87	0.98	0.95	0.91
		elastic net	0.95	0.92	0.97	0.93	0.97	0.94	0.92
		neural network	0.91	0.83	0.94	0.86	0.93	0.88	0.85
	1000	GMDH	0.90	0.82	0.94	0.85	0.92	0.88	0.83
		dce-GMDH	0.96	0.93	0.97	0.93	0.97	0.95	0.93
		svm	0.96	0.92	0.97	0.93	0.97	0.95	0.92
		random forest	0.94	0.87	0.98	0.94	0.95	0.92	0.90
		naive bayes	0.95	0.97	0.94	0.87	0.99	0.95	0.91
		elastic net	0.96	0.93	0.97	0.93	0.97	0.95	0.93
		neural network	0.92	0.85	0.96	0.89	0.94	0.90	0.89

Table A.15. Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 5 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.67	0.15	0.89	0.38	0.71	0.52	0.31
		dce-GMDH	0.65	0.20	0.85	0.36	0.72	0.52	0.32
		svm	0.67	0.08	0.93	0.33	0.70	0.50	0.28
		random forest	0.64	0.25	0.80	0.35	0.71	0.53	0.29
		naive bayes	0.61	0.41	0.70	0.37	0.74	0.55	0.37
		elastic net	0.68	0.10	0.93	0.39	0.71	0.51	0.33
		neural network	0.60	0.32	0.72	0.33	0.71	0.52	0.31
	100	GMDH	0.69	0.09	0.94	0.43	0.71	0.52	0.25
		dce-GMDH	0.67	0.15	0.89	0.38	0.71	0.52	0.28
		svm	0.69	0.04	0.97	0.38	0.70	0.51	0.23
		random forest	0.65	0.23	0.83	0.36	0.72	0.53	0.26
		naive bayes	0.63	0.40	0.73	0.38	0.74	0.56	0.37
		elastic net	0.69	0.07	0.96	0.42	0.71	0.51	0.26
		neural network	0.62	0.29	0.76	0.34	0.72	0.52	0.29
	500	GMDH	0.70	0.04	0.98	0.53	0.71	0.51	0.12
		dce-GMDH	0.69	0.08	0.96	0.47	0.71	0.52	0.17
		svm	0.70	0.03	0.99	0.48	0.70	0.51	0.13
		random forest	0.67	0.19	0.87	0.38	0.72	0.53	0.24
		naive bayes	0.63	0.46	0.70	0.40	0.76	0.58	0.42
		elastic net	0.70	0.04	0.98	0.51	0.71	0.51	0.13
		neural network	0.67	0.17	0.88	0.37	0.71	0.53	0.22
	1000	GMDH	0.70	0.03	0.99	0.56	0.70	0.51	0.09
		dce-GMDH	0.70	0.07	0.97	0.51	0.71	0.52	0.13
		svm	0.70	0.02	0.99	0.52	0.70	0.51	0.09
		random forest	0.68	0.16	0.90	0.40	0.71	0.53	0.22
		naive bayes	0.63	0.48	0.69	0.40	0.76	0.58	0.43
		elastic net	0.70	0.04	0.98	0.53	0.71	0.51	0.10
		neural network	0.68	0.12	0.93	0.40	0.71	0.52	0.18
Medium	50	GMDH	0.72	0.36	0.88	0.59	0.77	0.62	0.49
		dce-GMDH	0.71	0.43	0.83	0.54	0.78	0.63	0.49
		svm	0.71	0.21	0.92	0.56	0.74	0.57	0.47
		random forest	0.71	0.42	0.84	0.53	0.78	0.63	0.46
		naive bayes	0.69	0.68	0.69	0.49	0.84	0.69	0.55
		elastic net	0.73	0.32	0.90	0.61	0.76	0.61	0.50
		neural network	0.66	0.44	0.76	0.44	0.76	0.60	0.42
	100	GMDH	0.73	0.35	0.90	0.63	0.77	0.62	0.45
		dce-GMDH	0.73	0.44	0.85	0.59	0.78	0.64	0.48
		svm	0.73	0.26	0.93	0.64	0.75	0.59	0.44
		random forest	0.72	0.43	0.84	0.55	0.78	0.64	0.45
		naive bayes	0.70	0.70	0.70	0.51	0.85	0.70	0.57
		elastic net	0.74	0.35	0.91	0.65	0.77	0.63	0.46
		neural network	0.68	0.45	0.78	0.48	0.77	0.62	0.44
	500	GMDH	0.75	0.36	0.92	0.66	0.77	0.64	0.45
		dce-GMDH	0.75	0.41	0.90	0.64	0.78	0.65	0.49
		svm	0.75	0.34	0.93	0.68	0.77	0.63	0.44
		random forest	0.73	0.44	0.86	0.57	0.78	0.65	0.49
		naive bayes	0.70	0.71	0.70	0.50	0.85	0.70	0.58
		elastic net	0.75	0.39	0.91	0.65	0.78	0.65	0.48
		neural network	0.73	0.44	0.85	0.56	0.78	0.64	0.48
	1000	GMDH	0.75	0.37	0.92	0.66	0.77	0.64	0.47
		dce-GMDH	0.75	0.41	0.90	0.65	0.78	0.66	0.50
		svm	0.75	0.35	0.93	0.68	0.77	0.64	0.45
		random forest	0.74	0.44	0.86	0.58	0.78	0.65	0.50
		naive bayes	0.70	0.71	0.70	0.50	0.85	0.70	0.59
		elastic net	0.76	0.40	0.91	0.65	0.78	0.66	0.49
		neural network	0.74	0.43	0.87	0.60	0.78	0.65	0.49

Table A.15. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 5 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.81	0.65	0.88	0.72	0.87	0.77	0.68
		dce-GMDH	0.82	0.72	0.86	0.71	0.89	0.79	0.69
		svm	0.82	0.58	0.92	0.78	0.85	0.75	0.70
		random forest	0.83	0.69	0.89	0.74	0.87	0.79	0.70
		naive bayes	0.77	0.93	0.70	0.57	0.96	0.82	0.70
		elastic net	0.83	0.69	0.90	0.76	0.88	0.79	0.71
		neural network	0.76	0.62	0.82	0.61	0.84	0.72	0.60
	100	GMDH	0.83	0.69	0.89	0.75	0.88	0.79	0.69
		dce-GMDH	0.84	0.73	0.88	0.74	0.89	0.81	0.71
		svm	0.84	0.66	0.92	0.79	0.87	0.79	0.70
		random forest	0.84	0.71	0.89	0.75	0.88	0.80	0.70
		naive bayes	0.79	0.93	0.72	0.59	0.96	0.83	0.71
		elastic net	0.85	0.72	0.90	0.77	0.88	0.81	0.72
		neural network	0.80	0.67	0.85	0.66	0.86	0.76	0.65
	500	GMDH	0.85	0.72	0.90	0.77	0.88	0.81	0.74
		dce-GMDH	0.85	0.73	0.91	0.77	0.89	0.82	0.75
		svm	0.85	0.69	0.92	0.80	0.88	0.81	0.74
		random forest	0.84	0.72	0.90	0.75	0.88	0.81	0.73
		naive bayes	0.81	0.91	0.77	0.63	0.95	0.84	0.74
		elastic net	0.86	0.74	0.91	0.78	0.89	0.82	0.75
		neural network	0.84	0.72	0.89	0.74	0.88	0.81	0.73
	1000	GMDH	0.85	0.73	0.91	0.77	0.89	0.82	0.75
		dce-GMDH	0.86	0.74	0.91	0.78	0.89	0.82	0.75
		svm	0.86	0.70	0.92	0.80	0.88	0.81	0.74
		random forest	0.85	0.72	0.90	0.76	0.88	0.81	0.74
		naive bayes	0.81	0.90	0.77	0.63	0.95	0.84	0.74
		elastic net	0.86	0.74	0.91	0.78	0.89	0.83	0.76
		neural network	0.85	0.73	0.90	0.76	0.89	0.81	0.74

Table A.16. Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 10 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.66	0.20	0.85	0.37	0.72	0.53	0.31
		dce-GMDH	0.65	0.21	0.84	0.36	0.72	0.52	0.32
		svm	0.67	0.09	0.92	0.33	0.71	0.50	0.28
		random forest	0.65	0.22	0.82	0.35	0.72	0.52	0.29
		naive bayes	0.60	0.45	0.66	0.36	0.74	0.56	0.38
		elastic net	0.68	0.11	0.92	0.39	0.71	0.52	0.33
		neural network	0.59	0.34	0.70	0.32	0.72	0.52	0.31
	100	GMDH	0.68	0.13	0.92	0.42	0.71	0.52	0.26
		dce-GMDH	0.67	0.15	0.90	0.40	0.71	0.52	0.29
		svm	0.69	0.05	0.97	0.40	0.70	0.51	0.24
		random forest	0.66	0.21	0.85	0.37	0.71	0.53	0.26
		naive bayes	0.61	0.47	0.67	0.38	0.75	0.57	0.40
		elastic net	0.69	0.08	0.95	0.43	0.71	0.52	0.27
		neural network	0.60	0.33	0.72	0.34	0.71	0.52	0.31
	500	GMDH	0.70	0.04	0.98	0.51	0.71	0.51	0.12
		dce-GMDH	0.70	0.07	0.96	0.48	0.71	0.52	0.16
		svm	0.70	0.03	0.98	0.48	0.70	0.51	0.13
		random forest	0.67	0.18	0.89	0.40	0.72	0.53	0.24
		naive bayes	0.61	0.52	0.65	0.39	0.76	0.59	0.44
		elastic net	0.70	0.05	0.98	0.51	0.71	0.51	0.14
		neural network	0.62	0.29	0.77	0.34	0.72	0.53	0.30
	1000	GMDH	0.70	0.03	0.99	0.55	0.70	0.51	0.09
		dce-GMDH	0.70	0.06	0.97	0.51	0.71	0.52	0.12
		svm	0.70	0.03	0.99	0.51	0.70	0.51	0.10
		random forest	0.68	0.16	0.90	0.41	0.71	0.53	0.23
		naive bayes	0.61	0.53	0.65	0.39	0.76	0.59	0.45
		elastic net	0.70	0.04	0.98	0.53	0.71	0.51	0.10
		neural network	0.64	0.24	0.82	0.34	0.72	0.53	0.27
Medium	50	GMDH	0.71	0.39	0.85	0.55	0.77	0.62	0.48
		dce-GMDH	0.71	0.42	0.83	0.53	0.78	0.63	0.48
		svm	0.71	0.21	0.92	0.55	0.74	0.56	0.47
		random forest	0.71	0.41	0.84	0.54	0.78	0.63	0.47
		naive bayes	0.69	0.68	0.69	0.49	0.84	0.69	0.55
		elastic net	0.72	0.33	0.89	0.59	0.76	0.61	0.50
		neural network	0.66	0.45	0.75	0.44	0.76	0.60	0.42
	100	GMDH	0.73	0.37	0.88	0.59	0.77	0.62	0.45
		dce-GMDH	0.72	0.41	0.86	0.58	0.78	0.64	0.46
		svm	0.72	0.25	0.93	0.62	0.75	0.59	0.44
		random forest	0.72	0.41	0.86	0.56	0.77	0.63	0.45
		naive bayes	0.70	0.69	0.71	0.50	0.84	0.70	0.56
		elastic net	0.73	0.35	0.90	0.63	0.77	0.62	0.45
		neural network	0.67	0.44	0.77	0.45	0.76	0.61	0.43
	500	GMDH	0.75	0.37	0.91	0.64	0.77	0.64	0.46
		dce-GMDH	0.75	0.41	0.90	0.64	0.78	0.65	0.48
		svm	0.75	0.34	0.92	0.67	0.77	0.63	0.44
		random forest	0.74	0.42	0.87	0.59	0.78	0.64	0.48
		naive bayes	0.70	0.70	0.70	0.50	0.84	0.70	0.58
		elastic net	0.75	0.40	0.90	0.65	0.78	0.65	0.48
		neural network	0.69	0.46	0.79	0.50	0.78	0.63	0.48
	1000	GMDH	0.75	0.38	0.91	0.65	0.77	0.64	0.47
		dce-GMDH	0.75	0.41	0.90	0.64	0.78	0.66	0.50
		svm	0.75	0.36	0.92	0.66	0.77	0.64	0.46
		random forest	0.74	0.42	0.88	0.60	0.78	0.65	0.49
		naive bayes	0.70	0.70	0.70	0.50	0.85	0.70	0.58
		elastic net	0.75	0.41	0.90	0.64	0.78	0.66	0.50
		neural network	0.71	0.46	0.82	0.52	0.78	0.64	0.49

Table A.16. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 10 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.85	0.81	0.87	0.74	0.92	0.84	0.76
		dce-GMDH	0.87	0.83	0.89	0.77	0.93	0.86	0.78
		svm	0.87	0.74	0.93	0.83	0.90	0.84	0.79
		random forest	0.88	0.78	0.92	0.82	0.91	0.85	0.79
		naive bayes	0.84	0.96	0.79	0.66	0.98	0.87	0.77
		elastic net	0.88	0.81	0.91	0.81	0.92	0.86	0.80
		neural network	0.84	0.77	0.87	0.72	0.90	0.82	0.73
	100	GMDH	0.86	0.85	0.87	0.75	0.93	0.86	0.77
		dce-GMDH	0.88	0.82	0.91	0.80	0.93	0.87	0.79
		svm	0.89	0.79	0.93	0.84	0.91	0.86	0.80
		random forest	0.89	0.80	0.93	0.83	0.91	0.86	0.79
		naive bayes	0.85	0.95	0.81	0.69	0.97	0.88	0.78
		elastic net	0.89	0.83	0.92	0.83	0.93	0.88	0.81
		neural network	0.86	0.78	0.89	0.76	0.91	0.83	0.75
	500	GMDH	0.88	0.88	0.88	0.76	0.95	0.88	0.82
		dce-GMDH	0.91	0.85	0.94	0.85	0.94	0.89	0.85
		svm	0.90	0.82	0.94	0.86	0.92	0.88	0.84
		random forest	0.90	0.82	0.94	0.85	0.92	0.88	0.83
		naive bayes	0.88	0.91	0.87	0.75	0.96	0.89	0.82
		elastic net	0.91	0.86	0.94	0.85	0.94	0.90	0.85
		neural network	0.89	0.81	0.92	0.81	0.92	0.86	0.81
	1000	GMDH	0.88	0.89	0.88	0.77	0.95	0.89	0.82
		dce-GMDH	0.92	0.86	0.94	0.86	0.94	0.90	0.86
		svm	0.91	0.83	0.95	0.87	0.93	0.89	0.84
		random forest	0.90	0.83	0.94	0.85	0.93	0.88	0.84
		naive bayes	0.89	0.90	0.88	0.76	0.95	0.89	0.82
		elastic net	0.92	0.87	0.94	0.86	0.94	0.90	0.86
		neural network	0.89	0.82	0.93	0.83	0.92	0.87	0.83

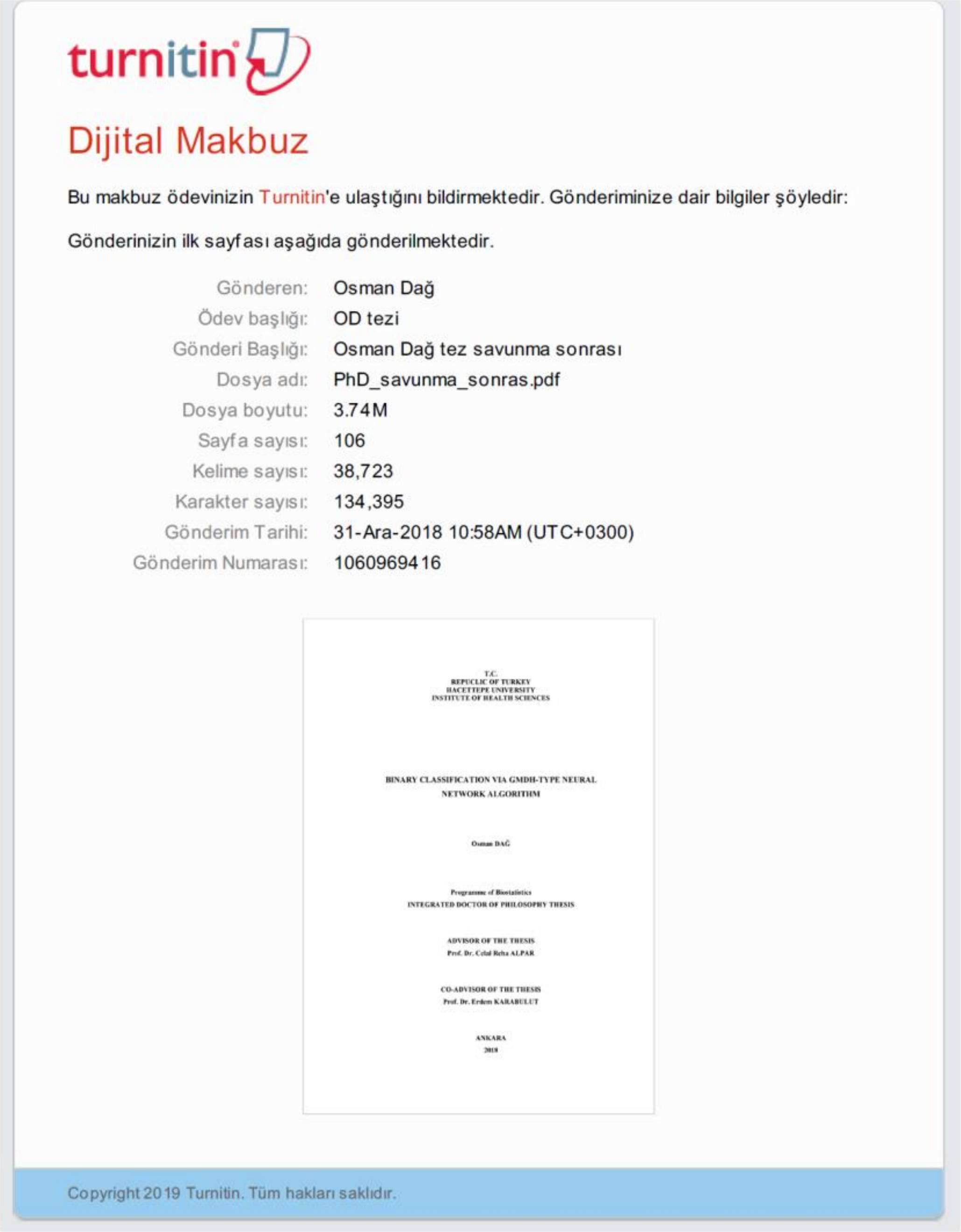
Table A.17. Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 15 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
Low	50	GMDH	0.66	0.23	0.84	0.38	0.72	0.53	0.31
		dce-GMDH	0.65	0.21	0.84	0.36	0.72	0.53	0.32
		svm	0.67	0.09	0.92	0.34	0.71	0.51	0.29
		random forest	0.65	0.22	0.84	0.37	0.72	0.53	0.30
		naive bayes	0.59	0.47	0.64	0.36	0.74	0.56	0.38
		elastic net	0.68	0.12	0.92	0.39	0.71	0.52	0.33
		neural network	0.60	0.36	0.71	0.34	0.72	0.53	0.33
	100	GMDH	0.68	0.16	0.90	0.42	0.72	0.53	0.27
		dce-GMDH	0.67	0.16	0.89	0.40	0.71	0.53	0.29
		svm	0.69	0.05	0.96	0.40	0.70	0.51	0.24
		random forest	0.66	0.19	0.87	0.39	0.71	0.53	0.26
		naive bayes	0.60	0.49	0.65	0.38	0.75	0.57	0.40
		elastic net	0.69	0.09	0.95	0.44	0.71	0.52	0.28
		neural network	0.61	0.35	0.72	0.35	0.72	0.53	0.33
	500	GMDH	0.70	0.07	0.97	0.52	0.71	0.52	0.15
		dce-GMDH	0.70	0.11	0.95	0.49	0.71	0.53	0.20
		svm	0.70	0.04	0.98	0.49	0.70	0.51	0.14
		random forest	0.69	0.16	0.91	0.44	0.72	0.54	0.23
		naive bayes	0.61	0.54	0.63	0.39	0.76	0.59	0.45
		elastic net	0.70	0.10	0.96	0.51	0.71	0.53	0.19
		neural network	0.63	0.34	0.75	0.37	0.73	0.55	0.35
	1000	GMDH	0.70	0.06	0.98	0.55	0.71	0.52	0.12
		dce-GMDH	0.70	0.14	0.94	0.52	0.72	0.54	0.21
		svm	0.70	0.05	0.98	0.52	0.71	0.51	0.12
		random forest	0.69	0.15	0.92	0.45	0.72	0.54	0.22
		naive bayes	0.61	0.55	0.63	0.39	0.76	0.59	0.45
		elastic net	0.70	0.13	0.95	0.53	0.72	0.54	0.21
		neural network	0.64	0.33	0.77	0.38	0.73	0.55	0.35
Medium	50	GMDH	0.72	0.43	0.84	0.55	0.78	0.63	0.49
		dce-GMDH	0.72	0.45	0.83	0.55	0.79	0.64	0.50
		svm	0.71	0.24	0.92	0.58	0.74	0.58	0.49
		random forest	0.73	0.43	0.85	0.57	0.78	0.64	0.49
		naive bayes	0.70	0.69	0.70	0.50	0.84	0.70	0.56
		elastic net	0.73	0.36	0.88	0.60	0.77	0.62	0.51
		neural network	0.66	0.46	0.74	0.44	0.77	0.60	0.43
	100	GMDH	0.73	0.42	0.87	0.60	0.78	0.64	0.47
		dce-GMDH	0.73	0.44	0.86	0.59	0.79	0.65	0.48
		svm	0.73	0.29	0.92	0.63	0.76	0.60	0.46
		random forest	0.74	0.43	0.87	0.59	0.78	0.65	0.47
		naive bayes	0.71	0.70	0.71	0.51	0.85	0.71	0.57
		elastic net	0.74	0.40	0.89	0.63	0.78	0.64	0.48
		neural network	0.68	0.46	0.77	0.46	0.77	0.61	0.44
	500	GMDH	0.76	0.43	0.90	0.65	0.79	0.66	0.50
		dce-GMDH	0.78	0.54	0.89	0.68	0.82	0.71	0.59
		svm	0.76	0.40	0.91	0.68	0.78	0.66	0.49
		random forest	0.75	0.44	0.89	0.63	0.79	0.67	0.51
		naive bayes	0.71	0.72	0.71	0.51	0.85	0.71	0.59
		elastic net	0.79	0.55	0.89	0.68	0.82	0.72	0.60
		neural network	0.72	0.51	0.81	0.54	0.80	0.66	0.52
	1000	GMDH	0.76	0.43	0.90	0.65	0.79	0.67	0.51
		dce-GMDH	0.80	0.56	0.90	0.70	0.83	0.73	0.62
		svm	0.77	0.42	0.91	0.68	0.79	0.67	0.52
		random forest	0.76	0.45	0.89	0.64	0.79	0.67	0.52
		naive bayes	0.71	0.72	0.70	0.51	0.85	0.71	0.59
		elastic net	0.80	0.56	0.90	0.70	0.83	0.73	0.62
		neural network	0.74	0.53	0.83	0.57	0.81	0.68	0.56

Table A.17. (Continued). Classification performances of the classifiers when ρ_{x_i,x_j} are high, p is 15 and pp is 0.3.

ρ_{y,x_i}	n	Method	Acc	Sens	Spec	PPV	NPV	Bacc	F1
High	50	GMDH	0.84	0.80	0.85	0.71	0.92	0.83	0.74
		dce-GMDH	0.86	0.80	0.88	0.75	0.92	0.84	0.76
		svm	0.86	0.71	0.93	0.81	0.89	0.82	0.77
		random forest	0.87	0.76	0.91	0.80	0.90	0.83	0.76
		naive bayes	0.83	0.94	0.79	0.66	0.97	0.87	0.76
		elastic net	0.87	0.79	0.90	0.79	0.92	0.85	0.78
		neural network	0.81	0.71	0.85	0.67	0.88	0.78	0.67
	100	GMDH	0.85	0.84	0.86	0.73	0.93	0.85	0.76
		dce-GMDH	0.87	0.80	0.90	0.78	0.92	0.85	0.77
		svm	0.88	0.77	0.93	0.83	0.91	0.85	0.77
		random forest	0.87	0.77	0.92	0.81	0.91	0.84	0.77
		naive bayes	0.85	0.93	0.82	0.69	0.97	0.87	0.78
		elastic net	0.88	0.81	0.91	0.80	0.92	0.86	0.79
		neural network	0.82	0.72	0.87	0.70	0.88	0.80	0.69
	500	GMDH	0.87	0.88	0.87	0.75	0.94	0.87	0.80
		dce-GMDH	0.90	0.83	0.93	0.83	0.93	0.88	0.82
		svm	0.89	0.80	0.93	0.83	0.92	0.87	0.81
		random forest	0.89	0.79	0.93	0.83	0.91	0.86	0.80
		naive bayes	0.87	0.89	0.87	0.75	0.95	0.88	0.81
		elastic net	0.90	0.84	0.92	0.83	0.93	0.88	0.83
		neural network	0.86	0.75	0.90	0.77	0.90	0.83	0.76
	1000	GMDH	0.88	0.88	0.88	0.76	0.94	0.88	0.81
		dce-GMDH	0.91	0.84	0.93	0.84	0.93	0.89	0.84
		svm	0.89	0.81	0.93	0.84	0.92	0.87	0.82
		random forest	0.89	0.80	0.93	0.83	0.92	0.87	0.81
		naive bayes	0.88	0.89	0.87	0.75	0.95	0.88	0.81
		elastic net	0.91	0.85	0.93	0.84	0.94	0.89	0.84
		neural network	0.87	0.76	0.91	0.79	0.90	0.84	0.79

Appendix-2: Report for Originality of Thesis Study



turnitin

Dijital Makbuz

Bu makbuz ödevinizin Turnitin'e ulaştığını bildirmektedir. Gönderiminize dair bilgiler şöyledir:

Gönderinizin ilk sayfası aşağıda gönderilmektedir.

Gönderen:	Osman Dağ
Ödev başlığı:	OD tezi
Gönderi Başlığı:	Osman Dağ tez savunma sonrası
Dosya adı:	PhD_savunma_sonras.pdf
Dosya boyutu:	3.74M
Sayfa sayısı:	106
Kelime sayısı:	38,723
Karakter sayısı:	134,395
Gönderim Tarihi:	31-Ara-2018 10:58AM (UTC+0300)
Gönderim Numarası:	1060969416

T.C.
REPUCLIC OF TURKEY
HACETTEPE UNIVERSITY
INSTITUTE OF HEALTH SCIENCES

BINARY CLASSIFICATION VIA GMDH-TYPE NEURAL NETWORK ALGORITHM

Osman DAĞ

Programme of Biostatistics
INTEGRATED DOCTOR OF PHILOSOPHY THESIS

ADVISOR OF THE THESIS
Prof. Dr. Celal Reha ALPAR

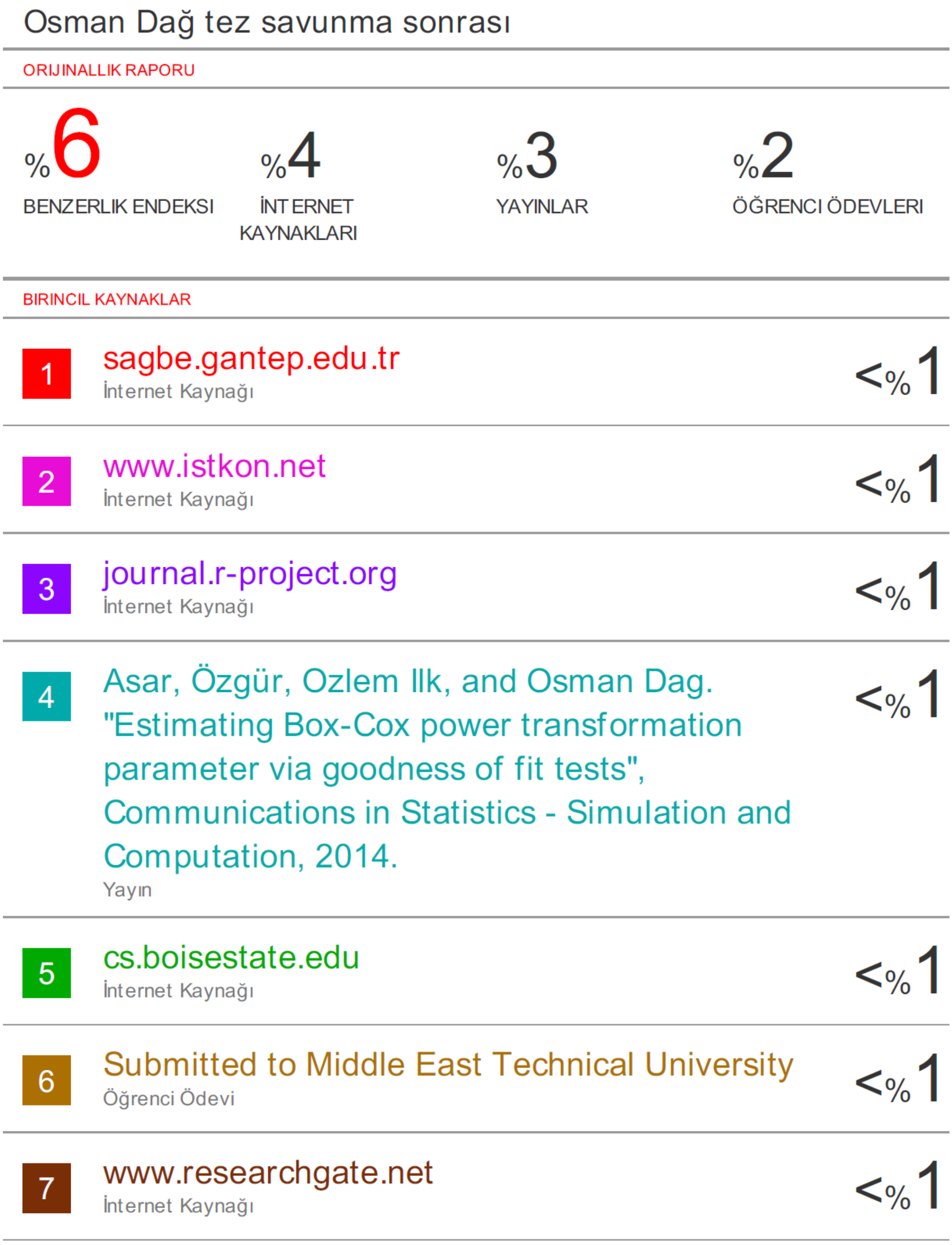
CO-ADVISOR OF THE THESIS
Prof. Dr. Erdem KARABULUT

ANKARA
2018

Thesis Title: Binary Classification Via GMDH-Type Neural Network Algorithm

Student Name and Surname: Osman Dağ

Total Page Number: 106



Osman Dağ tez savunma sonrası

ORIJINALLIK RAPORU

%6	%4	%3	%2
BENZERLIK ENDEKSI	İNTERNET KAYNAKLARI	YAYINLAR	ÖĞRENCI ÖDEVLERI

BIRINCIL KAYNAKLAR

1	sagbe.gantep.edu.tr İnternet Kaynağı	<%1
2	www.istkon.net İnternet Kaynağı	<%1
3	journal.r-project.org İnternet Kaynağı	<%1
4	Asar, Özgür, Ozlem Ilk, and Osman Dag. "Estimating Box-Cox power transformation parameter via goodness of fit tests", Communications in Statistics - Simulation and Computation, 2014. Yayın	<%1
5	cs.boisestate.edu İnternet Kaynağı	<%1
6	Submitted to Middle East Technical University Öğrenci Ödevi	<%1
7	www.researchgate.net İnternet Kaynağı	<%1

10. CURRICULUM VITAE

Osman DAĞ

Last Update: December, 2018

Contact Information

Hacettepe University
Faculty of Medicine
Department of Biostatistics
06100, Ankara, Turkey
E-mail: osman.dag@outlook.com
osman.dag@hacettepe.edu.tr
Phone: +90 312 305 14 67
Fax: +90 312 305 14 59
Home page: yunus.hacettepe.edu.tr/~osman.dag

Education

Doctor of Philosophy (Integrated) in Biostatistics (Candidate) at Hacettepe University, Ankara, Turkey (2014 – Present).

Master of Science in Statistics at Middle East Technical University, Ankara, Turkey (2012 – 2015), (Awarded by Middle East Technical University for completing all courses in one year with highest CGPA).

Bachelor of Science in Statistics at Middle East Technical University, Ankara, Turkey (2008 – 2012), (Graduated as a High Honor Student from Middle East Technical University).

Research Interests

Statistical Computing, especially with R
Computational Statistics
Machine Learning
GMDH-Type Neural Network Algorithms
Transformations
Time Series Analysis in Univariate Models
Numerical Optimization

Positions

Research Assistant, Department of Biostatistics, Faculty of Medicine, Hacettepe University, Ankara, Turkey (March 24, 2014 - present)

Research Assistant, Statistical Office, President’s Office, Middle East Technical University, Ankara, Turkey (February 4, 2013 – March 24, 2014)

Thesis

Dag, O. (ongoing). Binary Classification via GMDH-Type Neural Network Algorithm. Integrated Ph.D. Thesis. Under the Supervision of C. Reha Alpar and Erdem Karabulut.

Dag, O. (2015). GMDH-Type Neural Network Algorithms for Short Term Forecasting. M.S. Thesis. Under the Supervision of CeylanYozgatlıgil.

Articles in International Journals

Dag, O., Karabulut, E., Alpar, R. Binary Classification via GMDH-Type Neural Network Algorithms in R: the GMDH2 Package. Submitted.

Dag, O., Dolgun, A., Konar, N.M. (2018). onewaytests: An R Package for One-Way Tests in Independent Groups Designs. The R Journal, 10:1, 175-199.

Dag, O., Ilk, O. (2017). An Algorithm for Estimating Box-Cox Transformation Parameter in ANOVA. Communications in Statistics – Simulation and Computation, 46:8, 6424-6435.

Lafci, A., Gokcinar, D., Ornek, D., Yilmaz, S., Dikmen, B., Un, C., Kilci, O., **Dag, O.** (2017). Addition of Fentanyl to Levobupivacaine Decreases Postoperative Pain During Arthroscopic Shoulder Surgery Under Interscalene Brachial Plexus Block. Acta Medica Mediterranea, 33:5, 827-831.

Asar, O., Ilk, O., **Dag, O.** (2017). Estimating Box-Cox Power Transformation Parameter via Goodness-of-Fit Tests. Communications in Statistics - Simulation and Computation, 46:1, 91-105.

Dag, O., Yozgatligil, C. (2016). GMDH: An R Package for Short Term Forecasting via GMDH-Type Neural Network Algorithms. The R Journal, 8:1, 379-386.

Babaoglu, E., Kilic, H., Hezer, H., **Dag, O.**, Parlak, E., Senturk, A. Karalezli, A., Alisik, M., Erel, O., Hasanoglu, H.C. (2016). Comparison Of Thiol/Disulphide Homeostasis Parameters in Patients with COPD, Asthma and ACOS. European Review for Medical and Pharmacological Sciences, 20:8, 1537-1543.

Dag, O., Asar, O., Ilk, O. (2014). A Methodology to Implement Box-Cox Transformation When No Covariate is Available. Communications in Statistics – Simulation and Computation, 43:7, 1740-1759.

Papers in International Conferences

Ghahramani, M., **Dag, O.**, de Leon, A.R. (2014). Semi-Parametric Estimation of Count Time Series. International Work-Conference on Time Series Analysis, pp. 81-86, 25-27 June, Granada, Spain.

Abstracts in International Conferences

Dag, O., Karabulut, E., Alpar, R. (2018). Diverse Classifiers Ensembe Based on GMDH Algorithm for Binary Classification in R. 29th International Biometric Conference, pp. 72, 8-13 July, Barcelona, Spain.

Dag, O., Ilk, O. (2017). Asymmetric Confidence Interval with Box-Cox Transformation in R. 10th International Statistics Congress, pp. 215, 6-8 December, Ankara, Turkey [Poster].

Dag, O., Ilk, O. (2017). Box-Cox Transformation for Linear Models via Goodness-of-Fit Tests in R. 10th International Statistics Congress, pp. 219, 6-8 December, Ankara, Turkey [Poster].

Dag, O., Dolgun, A., Konar, N.M. (2017). One-Way Tests in Independent Groups Designs: the onewaytests Web Interface. 2nd International Biostatistics Congress, pp. 52-53, 25-28 October, Antalya, Turkey.

Bozer, A., **Dag, O.**, Karahan, S. (2017). Kohonen Öz Örgütlemeli Haritalama Yöntemi İle Psikotik Hastalıkların Kümelenmesi. 2nd International Biostatistics Congress, pp. 7-8, 25-28 October, Antalya, Turkey.

Dag, O., Yozgatligil, C. (2016). GMDH: An R Package for Short Term Forecasting Via GMDH-Type Neural Network Algorithms. 1st International Biostatistics Congress, pp. 59-60, 26-29 October, Antalya, Turkey.

Konar, N.M., **Dag, O.**, Basol, M. (2015). Comparison of Multiple Linear Regression and Ridge Regression on a Real Life Data Application. 9th International Statistics Congress, pp. 267-268, 28 October - 01 November, Antalya, Turkey [Poster].

Basol, M., **Dag, O.**, Konar, N.M. (2015). Estimation of Ridge Constant in Ridge Regression via K-Fold Cross Validation. 9th International Statistics Congress, pp. 271-272, 28 October - 01 November, Antalya, Turkey [Poster].

Konar, N.M., **Dag, O.**, Dolgun, A. (2015). Effects of Non-normality and Heterogeneity on Tests for One-Way Independent Groups Design: Type I Error and Power Comparisons. XVth Spanish Biometric Conference, pp. 113, 22-25 September, Bilbao, Spain.

Konar, N.M., **Dag, O.** (2015). Determining the Number of Clusters with an Application in R. European Meeting of Statisticians, pp. 102, 6-10 July, Amsterdam, Netherlands.

Dag, O., Ilk, O. (2015). MLE in A Feasible Region Is As Good As or Better Than The Usual MLE While Estimating Box-Cox Transformation Parameter in ANOVA. The 8th Conference of Eastern Mediterranean Region of International Biometric Society, pp. 15, 11-15 May, Cappadocia, Nevsehir, Turkey.

Dag, O., Yozgatligil, C. (2015). Forecasting Via GMDH Algorithm with Medical Applications in R. The 8th Conference of Eastern Mediterranean Region of International Biometric Society, pp. 54, 11-15 May, Cappadocia, Nevsehir, Turkey.

Dag, O., Asar, O., Ilk, O. (2013). Estimating Box-Cox Power Transformation Parameter Via Goodness-of-Fit Tests. y-BIS 2013: Joint Meeting of Young Business and Industrial Statisticians, pp. 66, 19-21 September, Istanbul, Turkey.

Dag, O., Asar, O., Ilk, O. (2012). A Methodology to Implement Box-Cox Transformation When No Covariate is Available. 8th World Congress in Probability and Statistics, pp. 188, 9-14 July, Istanbul, Turkey.

Abstracts in National Conferences

Konar, N.M., **Dag, O.**, Dolgun, A. (2015). onewaytests: Tek Yönlü Bağımsız Grup Tasarımı için Bir R Paketi. 17th National Biostatistics Congress, pp. 39-40, 5-9 November, Girne, Cyprus.

Refereed for the Following Journals

Computational Statistics
Earthquake Engineering and Engineering Vibration
Theoretical and Applied Climatology

Honors, Grants and Awards

Oral Presentation Award – Ranked 3rd among oral presentations – Dag, O., Dolgun, A., Konar, N.M. One-Way Tests in Independent Groups Designs: the onewaytests Web Interface. 2nd International Biostatistics Congress in Antalya, Turkey (2017).

Statement of Accomplishment for Machine Learning Course lectured by Andrew Ng, from Stanford University, in Coursera (2015).

Course Performance Award from METU – Ranked first in CGPA among the M.S. students who completed all courses in Department of Statistics during 2012-2013 academic year (2014).

Best Paper Award – Dag, O., Asar, O., Ilk, O. Estimating Box-Cox Power Transformation Parameter via Goodness-of-Fit Tests. y-BIS 2013: Joint Meeting of Young Business and Industrial Statisticians in Istanbul, Turkey (2013).

Travel and Accommodation Grant from The Scientific and Technological Research Council of Turkey (TUBITAK) to attend y-BIS 2013: Joint Meeting of Young Business and Industrial Statisticians in Istanbul, Turkey (2013).

Conference Grant from Bernoulli Society and Institute of Mathematical Statistics to attend 8th World Congress in Probability and Statistics in Istanbul, Turkey (2012), (exempted from all expenses).

Travel Grant from The Scientific and Technological Research Council of Turkey (TUBITAK) to attend 8th World Congress in Probability and Statistics in Istanbul, Turkey (2012).

Conference Grant from Bernoulli Society and Institute of Mathematical Statistics to attend Pre-world-congress Meeting of Young Researchers in Probability and Statistics in Istanbul, Turkey (2012), (exempted from all expenses).

Graduated as a High Honor Student from Middle East Technical University (2012).

Scholarship

National Scholarship for Ph.D. Students from The Scientific and Technological Research Council of Turkey (TUBITAK) (October, 2014 - present).

R Packages

Dag, O., Karabulut, E., Alpar, R. GMDH2: Binary Classification via GMDH-Type Neural Network Algorithms.

Dag, O., Dolgun, A., Konar, N.M. onewaytests: One-Way Tests in Independent Groups Designs.

Dag, O., Yozgatligil, C. GMDH: Short Term Forecasting via GMDH-Type Neural Network Algorithms.

Dag, O., Asar, O., Ilk, O. AID: Box-Cox Power Transformation.

Web-Tools

Dag, O., Karabulut, E., Alpar, R. GMDH2: A Web-Tool for Binary Classification via GMDH-Type Neural Network Algorithms.

Dag, O., Dolgun, A., Konar, N.M. onewaytests: A Web-Tool for One-Way Tests in Independent Groups Designs.

Professional Association Memberships

Institute of Mathematical Statistics

Bernoulli Society

International Society for Business and Industrial Statistics

International Statistical Institute

International Biometric Society

Biyoistatistik Dernegi (in Turkey)

Languages

Turkish (Native)

English (Fluent)

Computer Literacy

R, MATLAB
SPSS, Minitab, Statistica, NCSS
LATEX, Microsoft Office (Word, Excel, Power Point)
PASS