Personalized Modeling for Medical Decision Support

Wen Liang

A thesis submitted to Auckland University of Technology
in fulfillment of the requirements
for the degree of Doctor of Philosophy - PhD

August, 2013

School of Computing and Mathematical Sciences

Primary Supervisor: Professor Nikola Kasabov
Secondary Supervisor: Professor Valery Feigin
Third Supervisor: Dr. Raphael Hu

Contents

List of Figures

List of Tables

List of Abbreviations

ACO	-	Ant Colony Optimization
ADCA	-	Adenocarcinoma
AI	-	Artificial Intelligence
AIS	-	Artificial Immune System
ANN	-	Artificial Neural Network
ARCOS	-	Auckland Regional Community Stroke
ASTRO	-	Auckland Stroke Outcomes Study
BCOS	-	Bakas Caregiving Outcomes Scale
BI	-	Barthel Index
BGSA	-	Binary-Valued Gravitational Search Algorithm
CFO	-	Central Force Optimization
DENFIS	-	Dynamic Evolving Neuro-Fuzzy Inference
DNA	-	Deoxyribonucleic Acid
EA	-	Evolutionary Algorithm
EC	-	Evolutionary Computation
ECF	-	Evolving Classifier Function
ECOS	-	Evolving Connectionist Systems
EESNN	-	Extended eSNN
EFuNNs	-	Evolving Fuzzy Neural Networks
EP	-	Evolutionary Programming
ES	-	Evolution Strategy
eSNN	-	Evolving Spiking Neural Network
evoPM	-	Evolving Personalized Modeling System and Framework

FAI	-	Frenchay Activity Index
FN	-	False Negative
FP	-	False Positive
GA	-	Genetic Algorithm
GDS-15	-	Geriatric Depression Scale
GHQ-28	-	General Health Questionnaire 28
GP	-	Genetic Programming
GSA	-	Gravitational Search Algorithm
HAMT	-	Hodkinson Abbreviated Mental Test
HMM	-	Hidden Markov models
HS	-	Heuristic Search
HTGS	-	Hydraulic Turbine Governing System
KNN	-	K-Nearest Neighbor
LIF	-	Leaky Integrate-and-Fire
LM	-	Levenberg-Marquardt
LOOCV	-	Leave-One-Out Cross-Validation
LSM	-	Liquid State Machine
LSSVM	-	Least Square Support Vector Machine
LTD	-	Long-Term Depression
LTP	-	Long-Term Potentiation
MRS	-	Modified Rankin Score
MPM	-	Malignant Pleural Mesothlioma
NN	-	Neural Network
NINDS	-	National Institute of Neurological Disorders and Stroke
NSVM	-	Newton Support Vector Machine
PSO	-	Particle Swarm Optimization
PSP	-	Post-Synaptic Potential
reSNN	-	Recurrent Network Reservoir Structure of eSNN
RGSA	-	Real-Valued Gravitational Search Algorithm
RNA	-	Ribonucleic Acid
ROC	-	Receiver Operating Characteristic

SF-36	-	Short Form 36 Questionnaire
SNN	-	Spiking Neural Network
SNR	-	Signal-to-Noise-Ratio
SOM	-	Self-organizing Maps
SSVM	-	Smooth Support Vector Machine
STD	-	Spatio-Temporal Data
STDP	-	Spike Time Dependent Plasticity
SVM	-	Support Vector Machine
TDNN	-	Time Delay Neural Networks
TIA	-	Transient Ischemic Attack
TN	-	True Negative
TP	-	True Positive
TSP	-	Traveling Salesman Problem
TWNFI	-	Transductive Neuro-Fuzzy Inference System with Weighted Data Normalization
WHO	-	World Health Organization
WKNN	-	Weighted K-Nearest Neighbor
WWKNN	-	Weighted-Weighted K-Nearest Neighbor
V-SVM	-	Virtual-Support Vector Machine

Attestation of Authorship

"I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning".

Auckland, July, 2013

Wen Liang (Linda)

Acknowledgement

I would like to extend my sincere thanks and gratitude to the following individuals who have contributed towards the completion of my PhD's study:

First and foremost, I would like to express my deeply-felt thanks to my primary supervisor Professor Nikola Kasabov, for taking me on board as a PhD student. Your support and comments were very important to my study and ensured the research advanced in the right direction. If it wasn't for your great ideas, your patience, your tolerance and your encouragement throughout my research, I do not know where I would be today. Thank you so much!

I would like to express my sincere appreciation to my secondary supervisor Professor Valery Feigin, for his advices in the experimental design, data analysis and general advice. His guidance, support and encouragement had contributed immensely to help and inspire me during my doctoral study.

I am very indebted to thank my third supervisor Dr. Raphael Hu, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my doctoral study.

I also would like to extend my gratitude to Joyce D' Mello, the manager of the Knowledge Engineering and Discovery Research Institute (KEDRI), who always provided her time, enthusiasm and energy to assist me for in solving various problems during my study at all times. Also for the many encouraging words were extremely helpful to me during my study.

I would like to thank the past and present honorary members of KEDRI, who have

contributed immensely to my personal and professional time at the Auckland University of Technology. The group has been a source of friendships as well as good advice and collaboration. Especially grateful to Dr. Raphael Hu, Dr. Stefan Schliebs, Kshitij Dhoble, Nuttapod Nuntalid, and Haza Nuzly for their thoughtful advice, friendship and a lot of insightful discussions.

My deepest gratitude goes to my parents in China and parents-in-law in Taiwan for their unflagging love and support throughout my life; my doctoral study is simply impossible without them. Special thanks to my husband Tsung Chun, Tsai and my lovely daughter Wan Ting, Tsai, for all your great ideas and support in many practical ways during my study, and also put up with my bad moods when I was stressed and tired.

I would like to acknowledge the generous support of Auckland University of Technology for providing me with a good study environment and condition, especially thanks for the financial support of the AUT Vice Chancellors PhD Scholarship.

Lastly, I offer my regards and blessings to all of those who have supported me in any respect during the completion of the study.

Abstract

Personalized modeling is an emerging approach, in which a model is created for every new input vector of the problem space based on its nearest neighbors using transductive reasoning (Kasabov, 2007c; Vapnik, 1998). The underlying philosophy of this approach when applied to medicine is that each patient is an individual. Therefore, each patient requires and deserves a personalized treatment model that predicts the best possible outcomes for the patient. This study proposes a novel integrated evolving framework and system for personalized modeling (evoPM); an extension of a model proposed by Kasabov and Hu (Kasabov & Hu, 2011). By allowing users to select the most important features, optimize nearest neighbors and model parameters, the model provides higher accuracy and personalized knowledge than global and local modeling approaches. The evoPM creates a personalized model for each test sample with unique optimal sets of features, neighborhood and model parameters. In addition, the system keeps evolving and is adaptable to any new incoming data vectors. The already created personalized model can be further evolved on new data entering in the neighborhood.

Currently, the amount of available spatio-temporal data (STD) is growing exponentially, thus suitable techniques to effectively and efficiently analyze and process this vast quantity information are urgently needed. Evolving spiking neural networks (eSNN), an extension of spiking neural networks (SNN), is an emerging computational technique for STD analysis. Evolving SNNs learn STD by first converting temporal changes in the input variables into spike trains, then applying learning procedures to map spatio-temporal patterns detected in the data into temporal spiking activity of spatially located neurons. This study introduces two recently proposed methods for spatio-temporal pattern recognition, the extended eSNN frame-

work (EESNN) (Hamed, Kasabov, Shamsuddin, Widiputra & Dhoble, 2011) and the recurrent network reservoir structure of eSNN (reSNN) using liquid state machine (LSM) (Schliebs, Hamed & Kasabov, 2011). Both methods are the first time applied to evaluate the spatio-temporal weather and stroke occurrence data as a case study. The evoPM is applied as a classifier to learn the responses from the reSNN model.

The novel evoPM framework and system brings several advances over existing personalized modeling methods. These are summarized below:

- The integrated evolving personalized modeling system is developed based on an emerging novel technology namely eSNN;
- A recently developed population-based heuristic optimization approach called gravitational search algorithm (GSA) is applied to improve the robustness and generalisability of feature selection, neighborhood, model and its parameters optimization for classification, diagnostic and prognostic problems;
- The standard diseased classification system is replaced by personalized risk evaluation.
- The evoPM system and framework is novel applied to stroke data as case studies.

The novel method is validated on several benchmark cancer gene expression datasets and stroke data. The model outputs are compared with those of traditional global, local and personalized modeling methods. The results of all studies show that evoPM performs consistently better than the traditional methods. In particular, it develops more useful knowledge discovery for medical decision support for cancer diagnosis and prognosis due to it selects the optimal sets of genes and disease classification parameters for each individual patient.

CHAPTER 1

Introduction

1.1 Motivation

Vast quantities of biomedical personal data are now available in a large volume, but these data are complex, challenging and require new methods for their analysis. For instance, the human brain as a complex system and network of multiply connected cells was recognized in the late nineteenth century. Throughout the entire body nerve cells are connected to other cells. Nerve cells deliver messages from the brain to other organ systems and to the brain itself, thereby controlling function, communication, or decision making. On the other hand, complex interactions between genes and neuronal functions may cause certain brain diseases.

In recent decades, stroke has become a major public health challenge and concern in industrialized countries, including New Zealand. 90% of the more than 15 million new cases of stroke occurring globally every year are preventable (Donnell et al., 2010). To date, many intelligent systems have been developed with the purpose of improving health care and providing better health care facilities at reduced cost. However, traditional predictive models use standard population statistics, and therefore cannot predict the degree of disability for individual stroke survivors. Statistical prediction methods incorporate the most recurrent or powerful variables with certain loss of

unique patient information (Wieslaw, Oczkowski & Barreca, 1997). For this reason, personalized modeling is worthy of exploration and integration in the medical system for diagnosis, prediction and prescription.

Personalized modeling aims to create a model and to output a value for a single point of the problem space, utilizing additional information related to this point (Vapnik, 1998). This approach has been successfully applied to a variety of modeling problems. For instance, in the field of personalized healthcare and therapy, the knowledge discovered by this approach has significantly improved diagnosis, prediction and therapy for individual patients. It has also resulted in improved patient safety (Kathryn & Charis, 2012; Deloitte, 2012). Furthermore, given the current advances in networking technologies, personalized mobile service delivers a more efficient service, which in turn benefits business (Alcatel-Lucent, 2012).

Personalized medicine and drug design are becoming a leading trend in medicine, health care and life sciences. Drugs in the current market are tested on global populations and are prescribed based on statistical averages. However, these drugs are only effective in about 50% of cases, such as among cancer patients, the rate of ineffectiveness jumps to 75% and 40% for the anti-depressants (Jackson-Laboratory, n.d.). The objective of personalized medicine is to determine a patient's disease based on his/her molecular profile, so that appropriate therapies can be administered to appropriate patients at appropriate time. Current trends are replacing the traditional form of medicine with more accurate marker-assisted diagnosis and treatment (Abrahams, 2007). Personalized medicine presents numerous benefits and possibilities in different disciplines. Patients and clinicians receive more effective, precise and safer diagnosis and treatment; in the pharmaceutical industry, the productivity and efficiency of product lines are improved; society as a whole is rewarded by more focused applications of valuable health care resources.

1.2 Research Objectives, Questions and Hypotheses

With proven efficiency of personalized modeling in contrast to global and local modeling (Vapnik, 1998; Kasabov, 2007c; Kasabov & Hu, 2011; Shabo, 2007), application of this approach in various modeling problems becomes imperative. The most widely used personalized modeling approaches are nearest neighbor method and its deriv-

atives (Vapnik, 1998; Kasabov, 2007c). The major objective of this research is to develop a principally new method for personalized modeling and prognostic decision support. The novel personalized modeling system and framework is able to select the most significant features, optimize the optimal number of nearest neighbors and model parameters for a single input sample.

In many cases sudden undesirable events are triggered by specific spatio-temporal patterns of interaction between multiple variables over a period preceding the event-occurrence (e.g. cardiovascular disease (Cornelissen et al., 2002; Stoupel et al., 2000), cancer (J. E. Anderson et al., 2006), stroke occurrence (V. L. Feigin, Nikitin & Vinogradova, 1997; V. L. Feigin, Lawes, Bennett, Barker & Varsha, 2009; V. L. Feigin, Lawes, Bennett & Anderson, 2003), ecological and environmental disasters, financial and economic crises, etc.). Such events may be preventable if predicted early enough. However, existing personalized modeling methods are applicable only to static data, and therefore cannot identify important spatio-temporal interactions between variables that affect the outcome of interest for an individual (Vapnik, 1998; Kasabov, 2007c; L. Li, 2006). This task is timely, important and challenging, with a broad range of potential applications across medical, environmental, ecological and social areas.

Based on the above considerations, this research will achieve the following main objectives:

- To further develop the personalized modeling framework introduced by Kasabov (2007b);
- To develop a novel integrated evolving personalized modeling system (evoPM) by utilizing novel technology such as evolving spiking neural networks (eSNN);
- To improve the robustness and generalisability of feature selection, neighborhood, model and its parameters optimization for classification, diagnostic and prognostic problems;
- To evaluate the feasibility of the novel integrated evolving personalized method on several gene expression benchmark datasets and compare the outputs with traditional global, local and personalized methods;

- To study the personalized risk for individual patients, as opposed to classify patients into normal or diseased groups. Accurately quantifying this risk is critical for medical decision support to ensure that patients receive treatment that is optimal for their individual profile.
- To facilitate new knowledge discovery that will help to understand the complex brain and to improve medical decision making. The methodology and computational method for personalized modeling will be applied to stroke outcome prognosis data in a preliminary case study;
- To build personalized reservoir based generic methods for learning dynamic spatio-temporal data (STD), which applied to stroke risk spatio-temporal data as another case study.

More specifically, this research aims to answer the following research questions:

- How to develop a novel integrated evolving personalized modeling system using incrementally new data from various sources?
- How to select optimal set of features, neighborhoods, model and its parameters?
- How can these complex personal data be visualized?
- How to accurately estimate personalized risk?
- How to encode the real-valued data into spike trains prior to feed into a spatio-temporal filter (reservoir) to accumulate the spatio-temporal information of all input signals into a single high-dimensional state?

We hypothesize that the new model

- will be fast, efficient and incrementally trainable with new data, thus the efficiency of the model will improve over time;
- will improve the robustness and generalisability of feature selection, neighborhood, model and its parameters optimization in personalized modeling problems;

- will provide more precise accuracy and new personalized knowledge that will advance our understanding of signal processing in the biological brain, as well as enhancement of medical decision making;
- be extendable to multivariate spatio-temporal data. The personalized reservoir based generic methods will provide more accurately prediction than existing prognostic models.

1.3 Organization of the Thesis

The study is organized into the following chapters:

- *Part 1 - Literature Review*
    1. Chapter 2 outlines and compares inductive modeling and transductive modeling approaches. This is followed by a more detailed discussion of the two approaches, including a detailed review on global, local and personalized modeling approaches.
    2. Chapter 3 introduces biological neurons, followed by a review of two emerging contemporary neural models: spiking neural network (SNN) and evolving spiking neural network (eSNN). Neural encoding methods, learning algorithms and applications are also discussed. In addition, this chapter provides a brief literature review of two recently proposed methods for spatio-temporal pattern recognition, extended eSNN framework (EESNN) and the recurrent network reservoir structure of eSNN (reSNN) using liquid state machine (LSM).
    3. Chapter 4 introduces evolutionary computation (EC) focusing on a recently developed population-based heuristic optimization approach called gravitational search algorithm (GSA) for feature, neighborhood and model parameters optimization in the scope of personalized modeling.
    4. Chapter 5 introduces biological background of the human brain. And then it reviews the disease of stroke together with information methods for predicting risk and outcome of stroke. These methods include conventional statistical methods and computational intelligent modeling methods. The chapter concludes with a comparative study using conventional

global, local, classical personalized modeling methods, and evoPM-based algorithms on the stroke outcome prognosis data as a case study.

- *Part 2 - Proposed Novel evoPM for Static Data and Applications*

    1. Chapter 6 first discusses the motivation behind the development of this novel personalized modeling framework and system. Thereafter, the novel evoPM is introduced, ranging from the simple implementation (with limited model parameter optimization) to the complicated implementation (with full feature, neighborhood and model parameter optimization). Finally, an experimental study is designed for verifying the strength of each evoPM prototype.
    2. Chapter 7 presents relevant biological background before introducing several information techniques used for evaluating gene expression data. The chapter concludes with a comparative study investigating the feasibility of the novel evoPM on several benchmark cancer gene expression datasets using global, local, and personalized modeling methods in classification tasks.

- *Part 3 - Proposed Generic Personalized Modeling for Dynamic STD and Application*

    1. Chapter 8 offers a comparative study of exploring associations between changes in weather conditions and stroke occurrence. Results of conventional algorithms (global, local and classical personalized modeling) are compared with the algorithms from evoPM. In particular, gender differences in weather and stroke occurrence are explored.
    2. Chapter 9 begins with a pilot statistical analysis on the weather and stroke occurrence STD, followed by two studies using the proposed EESNN and reSNN generic personalized models.

The study conclusion and suggestions for future work are presented in Chapter 10.

1.4 Thesis Contributions

A summary of the contributions made by this thesis is visualized two dimensionally in Figure 1.1. The X axis represents the information acquiring methods proposed in the study, while the purple dots relate the proposed novel integrated methods to the datasets used for testing.

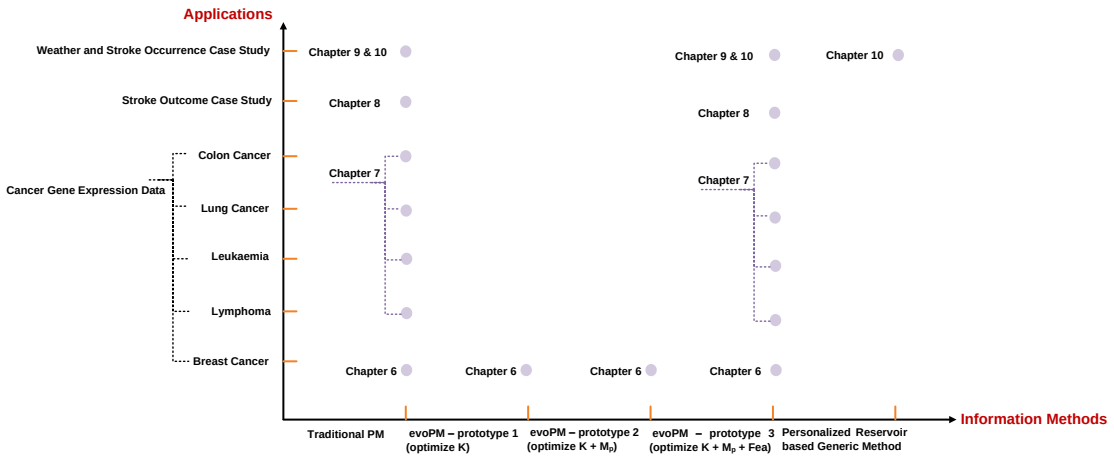


Figure 1.1: *A visual summary of contributions made by the proposed novel personalized modeling framework and system. The dots indicate the datasets to which each model was applied.*

1.5 Publication List

- **Book Chapter**

  1. Hu, Yingjie, Kasabov, N. & **Liang, W.** (2013). Personalized information modeling technologies for personalised medicine. In Springer Handbook of Bio-and Neuroinformatics (HBBNI). Berlin/Heidelberg: Springer.
  2. **Liang, W.**, Kasabov, N., Valery, F. & Rita, K. (2013). Information methods for predicting risk and outcome of stroke. In Springer Handbook of Bio-and Neuroinformatics (HBBNI). Berlin/Heidelberg: Springer.

- **Conference Paper**

1. **Liang, W.**, Hu, Yingjie., Kasabov, N., & Valery, F. (2011). Exploring associations between changes in ambient temperature and stroke occurrence: Comparative analysis using global and personalized modeling approaches. Proceedings of the 18th International Conference on Neural Information Processing, Shanghai, China, 129-137, Part I, Springer LNCS 7062.

- **Poster**

1. **Liang, W.**, Hu, Yingjie., & Kasabov, N. (2008). Integrated Feature, Neighborhood, and Model Optimization for Personalized Modeling and Knowledge Discovery. 15th International Conference on Neural Information Processing of the Asia Pacific Neural Network Assembly (ICONIP 2008), Auckland, New Zealand.
2. **Liang, W.**, Hu, Yingjie., & Kasabov, N. (2010). A framework for personalized modeling, profiling and prognosis on brain data. NZBIO Conference 2010, Auckland, New Zealand.

- **Journal Paper**

1. **Liang, W.**, Hu, Yingjie., Kasabov, N. (2012). Evolving Personalized Modeling System for Feature, Neighborhood and Parameter Optimization utilizing Gravitational Search Algorithm. Evolving Systems, Springer. (Accepted)
2. **Liang, W.**, Hu, Yingjie., Kasabov, N. (In preparation 2013). Evolving Spiking Neural Network (eSNN) for Personalized Modeling on Stroke Data.

CHAPTER 2

Personalized Modeling

2.1 Introduction

Before providing a detailed literature review on the concept of personalized modeling, a comparison between inductive modeling and transductive modeling approaches is given, and outline the basic theory behind these two approaches. Thereafter, inductive and transductive inference methods are described in depth, including a detailed review of global, local and personalized modeling approaches.

2.2 Inductive versus Transductive Reasoning Approaches

To date, most artificial intelligence (AI) learning methods, especially those employing neural fuzzy inference methods, are based on either inductive inference or transductive reasoning approaches. Figure 2.1 illustrates the difference between these two approaches. In the transductive inference method, data are trained and then tested in a problem space, while the inductive inference method first induces a function from the training data, which is then deducted and used to predict the testing data

(Vapnik, 1998). These two reasoning approaches are compared further in the following section.

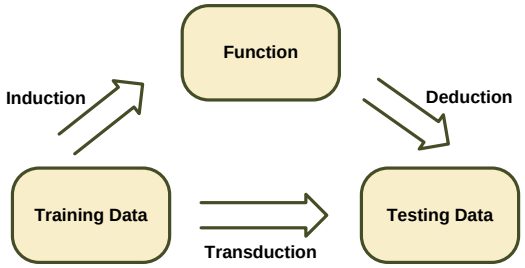


Figure 2.1: *The difference between inductive inference and transductive inference methods.*

2.2.1 Inductive Inference Method

The theory of inductive inference was pioneered by Ray Solomonoff around 1960 (Solomonoff, 1960). Inductive inference is defined as a process of inferring a general rule or law from the observations of a particular example (Angluin & Smith, 1983). For instance, given the binary string *"100, 111100, 11000, 1110, 1100"*, the following rule can be inferred: "any number of 1s are followed by any number of 0s".

In general, the inductive inference method concerns the creation of a model (generally a global model) from all available data. In other words, it focuses on the whole problem space. This model can be adapted to investigate new input vectors. Once a global model is created, no new information about a new input vector is considered. Instead, the extent to which the new input vector fits the model is estimated by an error calculation.

An overview of an inductive inference method is presented in Figure 2.2. A global model M is created from the dataset D, which is then recalled for every new input vector V_i. Model M computes an output Y_i for each new input vector.

2.2.2 Transductive Inference Method

The transductive inference approach was originated by Vapnik in 1998 (Vapnik, 1998). Transductive inference evaluates the potential value of a model for an in-

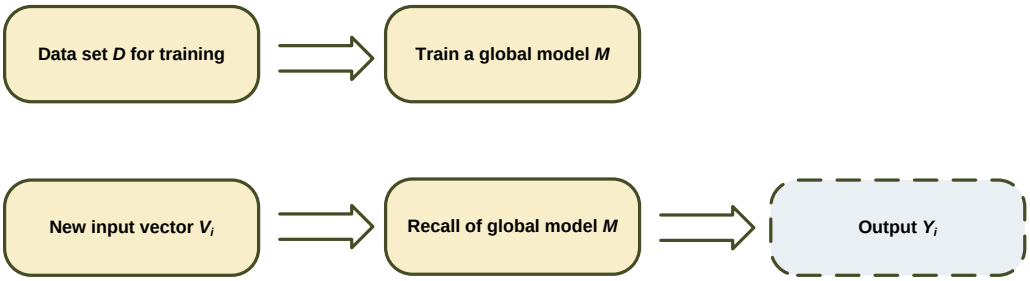


Figure 2.2: *Overview of the inductive inference method.*

dividual point of the problem space by using additional information related to the point. In contrast to the inductive inference approach which solves a general problem, the transductive inference approach is targeted to individual problem solving (Bosnic & Kononenko, 2003).

Figures 2.3 and 2.4 depict the transductive inference approach: every new input vector V_i is investigated by a classification or prediction task based primarily on its nearest neighbors. The nearest neighbors form a sub-data set D_i derived from the original training data set D. Based on these vectors, a new local model M_i is dynamically created and adapted to estimate the output Y_i for every new input vector V_i.

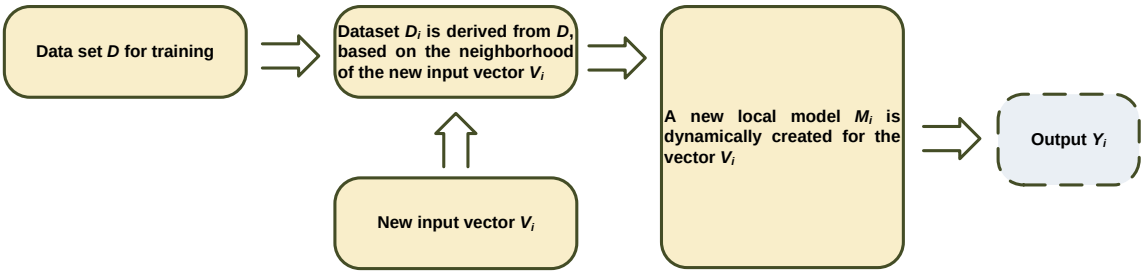


Figure 2.3: *Overview of the inductive inference method (a).*

2.3 Global, Local and Personalized Modeling

"Machine learning is the process of discovering and interpreting meaningful information, such as new correlations, patterns and trends by sifting through large amounts of data stored in repositories, using pattern recognition technologies as well as statistical and mathematical techniques"(Larose, 2005). Kasabov (2007c) classified com-

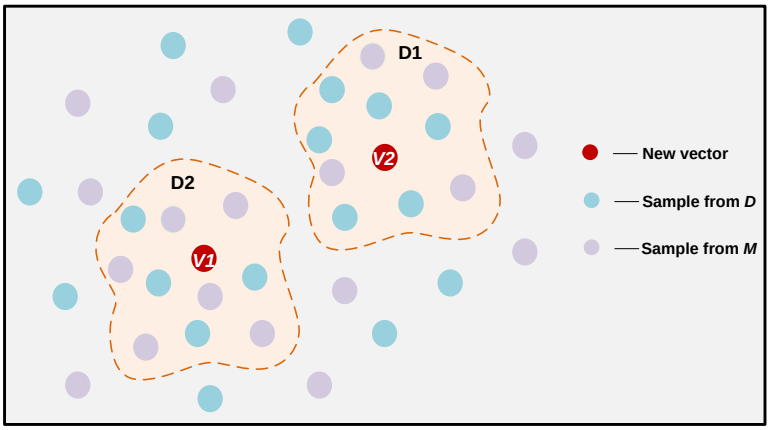


Figure 2.4: *Overview of transductive inference method (b): V_1 and V_2 represent two new input vectors surrounded by a number of nearest neighbors selected from the training data set D and generated from an existing model M.*

putational machine learning models into three categories (global, local, and personalized), such models have become widely used in data analysis and decision support, especially in the fields of medicine and bioinformatics.

2.3.1 Definition

- **Global modeling** - A global model is created from the entire data set for the whole problem space based on an inductive inference method. It focuses on the whole problem space rather than on individual vectors. This model is usually not readily adaptable to new input data.
- **Local modeling** - A local model is created to evaluate an output function in a sub-space of a problem space. Local modeling approaches are more amenable to individual vector interpretation than global modeling.
- **Personalized modeling** - A personalized model is evolved for every new input vector of the problem space based on its nearest neighbors using the transductive reasoning approach. Personalized modeling is tailored to solve the problem for an individual data point rather than a general problem across the whole data population.

2.3.2 Global Modeling

Support vector machine (SVM) is one of the most popular global modeling approaches used in machine learning. SVM is a fast optimization algorithm that achieves high-quality classification accuracy with few training samples. However, in dealing with a large, high-dimensional data set, the kernel computation time required to train the SVM classifier is prohibitively long.

Support Vector Machine (SVM)

SVM is a supervised learning algorithm based on small-sample statistical learning theory, was proposed by Vapnik (1998) and his co-workers. The algorithm has been widely applied to classification and regression problems. In addition, it has been successively extended by subsequence researchers. Adaptations of SVM include virtual-support vector machine (V-SVM) (Scholkopy & Smola, 2000), smooth support vector machine (SSVM) (Lee & Mangasarian, 2001), Newton support vector machine (NSVM) (Fung & Mangasarian, 2004), and least square support vector machine (LSSVM) (Suykens & Vandewalle, 1999).

The most widely used two SVM are linear SVM (Vapnik & Lerner, 1963) and non-linear SVM (Aizerman & Braverman, 1964). In cases where the data are linearly separable, SVM uses a hyperplane to separate a given set of training data, such that the distance from the hyperplane to the data is maximized (also known as "the maximum margin hyperplane"). If the data are non-linearly separable, SVM cooperates with the non-linear "kernel function" that automatically maps the data onto a feature space (possibly a high-dimensional feature space). Consequently, the hyperplane in the high-dimensional feature space corresponds to a non-linear decision boundary in the original input space.

Figure 2.5 displays the SVM process: the optimal hyperplane splits a set of vectors in such a way that the vectors within two separate categories are placed on either side of the plane.

Mathematically, the SVM can be formulated as the following equations, given a

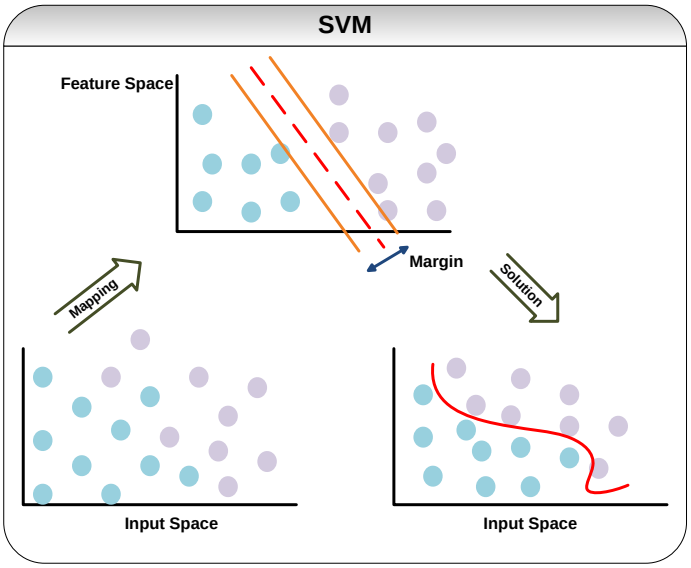


Figure 2.5: *Overview of a simple SVM process.*

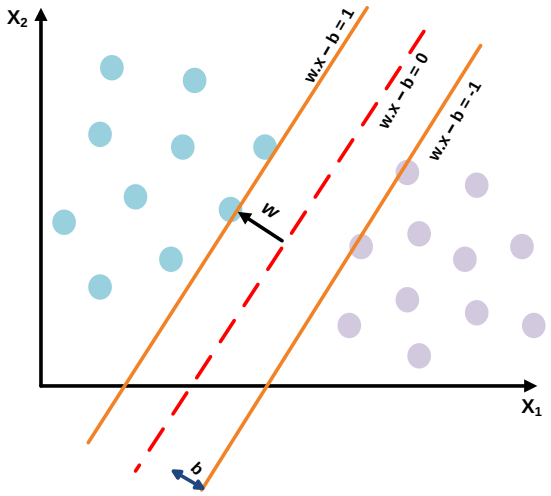


Figure 2.6: *Overview of a simple linearly separable SVM.*

two-class classification task:

$$D = \{((x_1, y_1), \ldots (x_i, y_i))|x \in R^n, y \in [-1, 1]\}_{i=1}^{m} \tag{2.1}$$

where D is a given training data set, x is a n-dimensional vector, and y is the class label that indicates the category of x.

As illustrated in Figure 2.6, when the data are linearly separable, the optimal hy-

perplane is defined as:

$$w^T_{(w_1,...w_n)}x + b = 0 \tag{2.2}$$

where w is the weight vector, and b is a scalar. Therefore, the optimal hyperplane separates the vectors into two distinct classes. Furthermore, both w and b can be constrained such that:

$$w(\wedge) = minL(w, b, \wedge) \tag{2.3}$$

where L is the Lagrange function, and $\wedge$ is the Lagrange multiplier. If w and b are to be chosen to maximize the margin, the hyperplane in Eq.(2.2) can be re-defined as:

$$w^T_{(w_1,...w_n)}x + b = 1 \tag{2.4}$$

$$w^T_{(w_1,...w_n)}x + b = -1 \tag{2.5}$$

Thus if the distance between the vectors belonging to the two different classes is maximized, those vectors are optimally separated by the hyperplanes given by Eq.(2.4) and Eq.(2.5). From Eq.(2.3), the parameters w, $\wedge$ and the optimal hyperplane are related by:

$$w = \sum_{i=1}^{n} \wedge_i x_i y_i \tag{2.6}$$

Therefore, the classifying function can be defined as:

$$f(x) = w^T_{(w_1,...w_n)}x + b \tag{2.7}$$

The result (either 1 or -1) obtained from Eq.(2.7) is ultimately used to assign a class to vector x.

2.3.3 Local Modeling

Local models are usually adaptable to new data vector and more suited to analysis of individual cases than global models. Evolving classifier function (ECF) is a representative approach for local modeling, was proposed by Kasabov (2002). The special characteristics of ECF are: (1) fast incremental and online learning, and (2)

dynamic allocation of rule nodes assists users in understanding and verifying the model's functionality.

Evolving Classifier Function (ECF)

As stated by Arbib (2003), traditional neural network models do not allow researchers to discover new patterns from the data, because they are essentially "black boxes". Kasabov (2002) introduced a novel type of neural network model, called evolving connectionist systems (ECOS) that enables fast incremental, online learning, as well as rule extraction and rule adaptation. According to Kasabov (2007b), "Evolving connectionist system (ECOS) is a connectionist architecture that facilitates modeling of an evolving process and knowledge discovery"; which represents ECOS imparts new information to neural network knowledge.

ECF is an implementation of ECOS that has been widely applied to pattern classification tasks. It comprises four layers of nodes (Kasabov, 2007b):

1. input variables;
2. fuzzy membership functions;
3. a set of data centers in the input space;
4. classes.

ECF produces rule nodes in a multi-dimensional input space, where each rule node is identified by its radius, center and class. Figure 2.7 illustrates the classification of data into clusters, where c denotes the class, v_i is the i^{th} data vector, and o_j and r_j are the center and radius of node j^{th}, respectively.

2.3.4 Personalized Modeling

The personalized modeling approach is a type of local modeling that is created for each new input vector of the problem space, based on its nearest neighbors using the

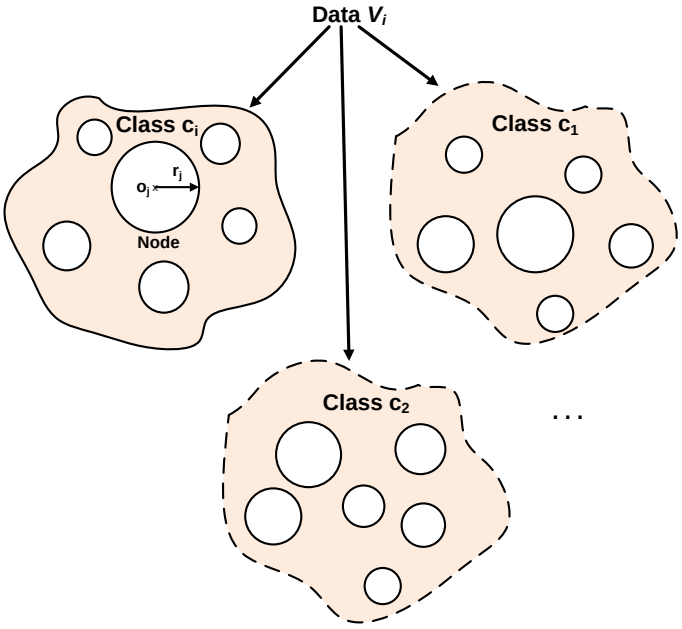


Figure 2.7: *An example of clusters evolved in ECF for a robotics classification task.*

transductive reasoning approach (Kasabov, 2007c; Vapnik, 1998). The basic principle and framework of personalized modeling is shown in Figure 2.8. Personalized modeling involves as following steps (Kasabov & Hu, 2011):

1. Select a feature subset S_i for the new input vector V_i from the given dataset D (the global problem space);
2. Select a group of (K) nearest neighbors of V_i and allocate this group to a local problem space D_i;
3. Create a personalized model M_i for V_i, which consists of a learning function $\mathcal{F}$ that measures the performance of M_i (e.g. a classifier);
4. Evaluate the feature subsets by the learning function based on the performance evaluated within a personalized problem space D_i;
5. Optimize personalized model M_i through iterations. The output is the optimal or near-optimal solution to V_i, when the termination criteria are reached. The solution includes an optimal personalized model M_i^* with a small set of features S_i^*;
6. Use model M_i^* to evaluate V_i and output the result;

7. Create a personalized profile for V_i.

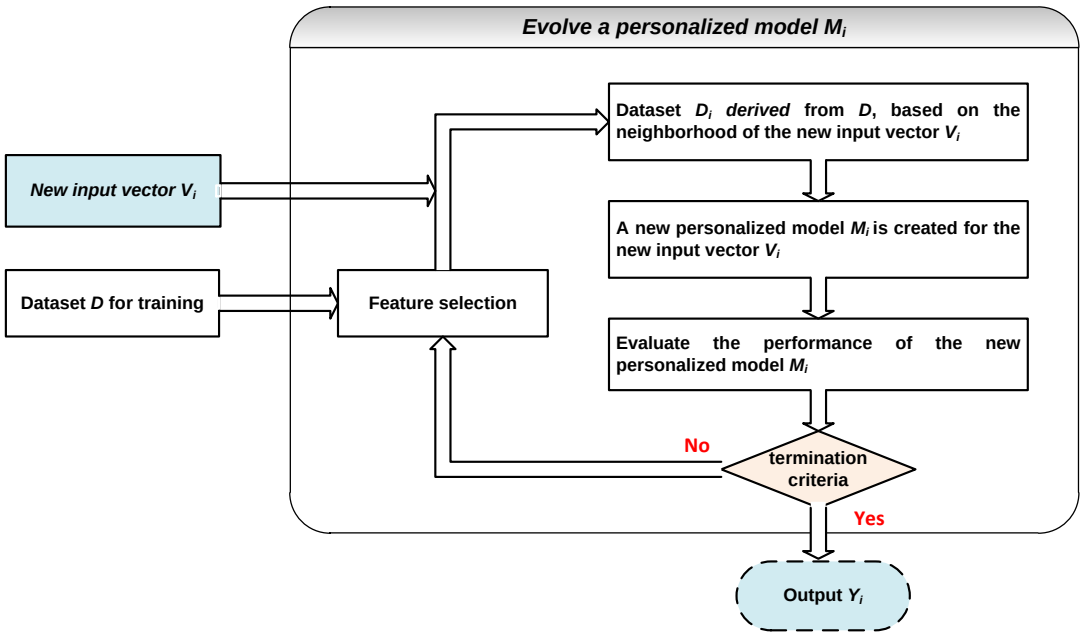


Figure 2.8: *Flowchart of personalized modeling.*

Several personalized modeling algorithms have been developed to date. such as k-nearest neighbor (KNN) is the simplest nearest neighbor algorithm and has been extended to weighted k-nearest neighbor (WKNN) (Dudani, 1976), weighted-weighted k-nearest neighbor (WWKNN) (Kasabov, 2007c), and transductive neuro-fuzzy inference system with weighted data normalization (TWNFI) (Song & Kasabov, 2006). These methods are described below.

K-Nearest Neighbor (KNN)

KNN has been successfully used for classifying sets of samples based on nearest training samples in a multi-dimensional feature space (Fix & Hodges, 1951). The basic idea behind the KNN algorithm is:

1. Data points are specified by feature pairs, set of pairs (e.g. $(x_1, y_1), \ldots, (x_n, y_n)$), and each data points is assigned a class label $C = c_1, \ldots, c_n$;

2. Similarity of the data points (considering all features) is measured by a chosen distance measurement (e.g. Euclidean distance (Eq.(2.8)), or Manhattan distance (Eq.(2.9)));

3. From the distance measurements, the k-nearest neighbors are found for a target data point. The data point is classified by majority voting rule.

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{2.8}$$

$$d(x,y) = \sum_{i=1}^{n}|x_i - y_i| \tag{2.9}$$

A overview of KNN is presented in Figure 2.9. If k=5, the target vector v is classified into class 1 based on the classification of its five nearest neighbors.

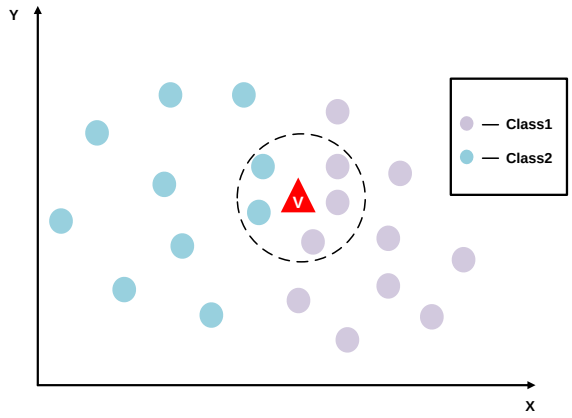


Figure 2.9: *An example of the KNN classification task. Each vector is represented by a two dimensional point within a Euclidean space.*

Weighted K-Nearest Neighbor (WKNN)

WKNN evaluates the output of a model focusing solely on an individual point in a problem space using information related to this point. In the WKNN algorithm, each new input vector is fitted to a local rather than a global model; thus each new input vector can be matched to an individual model without considering specific

information on existing vectors. In contrast to KNN, the output of a new input vector depends on the outputs of its k-nearest neighbor vectors, but also on the distance between these vectors and the new input vector, which is represented as a weight distance vector w.

Mathematically, the WKNN algorithm is expressed as:

$$Output = \sum_{j=1,\ldots,k_i} w_j y_j \tag{2.10}$$

where k_i represents the number of nearest neighbors, and w_j denotes the weight, calculated as:

$$w_j = \frac{max(d) - (d_j - min(d))}{max(d)} \tag{2.11}$$

where $d = [d_1, \ldots, d_{k_i}]$ represents the distance between the new input vector and k_i. The parameters $max(d)$ and $min(d)$ represent the maximum and minimum values in d, respectively.

Weighted-Weighted K-Nearest Neighbor (WWKNN)

WWKNN is a novel personalized modeling algorithm is proposed by Kasabov (2007c). In this algorithm, the output of each new input vector depend not only on the outputs of its k-nearest neighbors, and the distance between the existing vectors and the new input vector, but also on the power of each vector, which is weighted by its importance within the sub-space (local space) to which the new input vector belongs. We assume that all the variables from a data set are used and that distances between vectors are calculated in a v-dimensional space with all input variables having equal impact on the output variables. However, the importance of different variables might vary when classifying vectors into classes, if these variables are ranked by their discriminative power in classifying vectors over the entire v-dimensional Euclidean space. We note that the discriminative power of variables within a sub-space of the problem space may vary. The output of each new input vector is then assigned a power ranking within the neighborhood of k-nearest neighbor vectors.

The WWKNN algorithm uses the following formulas:

$$d_j = \sqrt{\sum_{l=1\ldots n}^{k} C_{i,l}(x_l - x_{j,l})^2} \tag{2.12}$$

$$C_i = (C_{i,l}, \ldots C_{i,n}) \tag{2.13}$$

where d_j is the distance between the new input vector x_i and its nearest neighbor vector x_j, k represents the number of nearest neighbors, and $C_{i,l}$ is the weighing coefficient between variable x_l and its nearest neighbor vector x_i. Each variable is ranked across all vectors in D_i by signal-to-noise-ratio (SNR) supervised method:

$$C_{i,l} = \frac{S_1}{\sum S_1(l = 1, 2, \ldots n)} \tag{2.14}$$

$$S_1 = \frac{|x_1^{(class1)} - x_1^{(class2)}|}{Std_1^{(class1)} + Std_1^{(class2)}} \tag{2.15}$$

where the parameters $x_1^{(class1)}$ and $x_1^{(class2)}$ represent the means of variable x from Class 1 and Class 2, respectively. The parameters $Std_1^{(class1)}$ and $Std_1^{(class2)}$ represent the standard deviations of Class 1 and Class 2 variables, respectively, in dataset D_i.

Transductive Neuro-Fuzzy Inference System with Weighted Data Normalization (TWNFI)

TWNFI is a dynamic neuro-fuzzy inference system with local generalization (Song & Kasabov, 2006), designed for solving problems requiring individual modeling analysis. This method creates a learning model based on the neighborhood of a new data vector, and calculates the output by applying the trained model on the new data.

In the TWNFI model, *Gaussian fuzzy membership functions* are used in each fuzzy rule for both antecedent and consequent parts. The parameters of the fuzzy membership functions are optimized by applying a *steepest descent (back-propagation) learning algorithm* (C. T. Lin & Lee, 1996; Wang, 1994). The distance between two

vectors a and b is computed using *weighted normalized Euclidean distance* defined as follows:

$$\|a - b\| = [\frac{1}{p}\sum_{j=1}^{p} w_i|a_j - b_j|^2]\frac{1}{2} \tag{2.16}$$

where w_j is the weight vector reflecting the importance of the variables to the specified problem.

Figure 2.10 is a general block diagram of the TWNFI algorithm. The algorithm is executed as follows:

1. Normalize the training data set and the new data vector V_i (range [0, 1]); set the initial weight for every input variable to 1;

2. Find N_v training samples (within an appropriate neighborhood D_v) that are closest to V_i, using the *weighted normalized Euclidean distance* defined in Eq. (2.16);

3. Compute the distances between the N_v training samples and V_i: $d_i, i = 1, 2, \ldots, N_v$, using Eq. (2.16), and calculate the weights for each sample: $w_i = 1 - (d_i - min(d)), i = 1, 2, \ldots, N_v$, $min(d)$ is the minimum element in the distance vector $d = [d_1, d_2, \ldots, d_{N_v}]$;

4. Cluster and partition the input subspace comprising the N_v selected training samples; create fuzzy rules and set their initial parameter values based on the clustering results. In each fuzzy rule, the centroid of a cluster is the center of the fuzzy membership function (e.g. *Gaussian membership function*) and the cluster radius is taken as the width;

5. Apply the *steepest descent (back-propagation)* approach to optimize the weights and the parameters of the fuzzy rules in a local model M_v;

6. Find a new neighborhood set D_v^* closest to V_i (Step 2): if the set contains the same samples as were found the previous search, the algorithm advances to the next step; otherwise, it repeats from Step 3;

7. Calculate the output value Y_v for the input vector V_i applying fuzzy inference over the set of fuzzy rules that constitute the local model M_v;

8. Algorithm terminates.

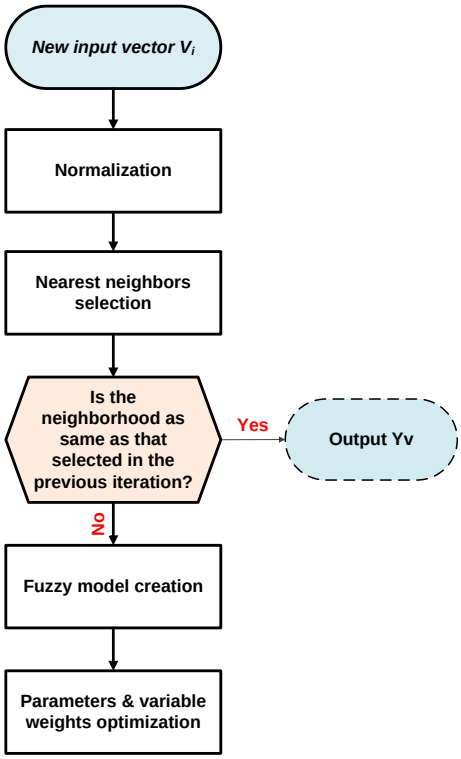


Figure 2.10: *A general block diagram of the TWNFI algorithm.*

The weights and parameters can be optimized as follows: Consider a system with V inputs, one output and M fuzzy rules initially defined by a clustering algorithm. The l^{th} rule is formed as:

R_l: if x_l is in F_{l1}, x_2 is in F_{l2} and $\ldots$ x_v is in F_{lv}, then y is in G_l, where F_{lj} are the fuzzy sets defined by the following Gaussian membership function:

$$Gaussian\ MF = \alpha\ exp(-\frac{(x-m)^2}{2\sigma^2}) \tag{2.17}$$

and G_l is defined as:

$$Gaussian\ MF = exp(-\frac{(y-n)^2}{2\delta^2}) \tag{2.18}$$

Thus, given an input vector $x_i = [x_1, x_2, \ldots, x_v]$, the output can be calculated by a

modified center average defuzzification function as:

$$f(x_i) = \frac{\sum_{l=1}^{M} \frac{n_l}{\delta_l^2} \prod_{j=1}^{V} \alpha_{li} \; exp[-\frac{w_j^2(x_{ij}-mli)^2}{2\sigma_{lj}^2}]}{\sum_{l=1}^{M} \frac{1}{\delta_l^2} \prod_{j=1}^{V} \alpha_{li} \; exp[-\frac{w_j^2(x_{ij}-mli)^2}{2\sigma_{lj}^2}]} \tag{2.19}$$

2.4 Open Questions in Personalized Modeling

Personalized modeling is an emerging technique applied in many disciplines, particularly the biomedical fields. However, numerous open questions must be addressed before a truly efficient personalized modeling system can be developed for data analysis. This section reviews a number of techniques relevant to the study, such as feature selection procedures, cross-validation techniques, performance measures and parameter optimization.

2.4.1 Feature Selection

In general, feature selection is regarded as a fundamental step in data mining, uses specific learning algorithms to find an optimal set of features among a given feature set. Throughout the past few years, feature selection techniques in machine learning have attracted much attention, and have become especially important in bioinformatics applications. Currently, this technique is applied in diverse fields, such as data mining (M. Chen, Han & Yu, 1996; Provost & Kolluri, 1999), pattern recognition (Ferri, Pudil, Hatef & Kittler, 1994), and text learning (Y. Yang & Pedersen, 1997). The primary goals of this technique are:

1. To improve classification or prediction accuracy;
2. To enhance speed and reduce the cost of learning stages;
3. To avoid over-fitting and improve classification or prediction model performance;
4. To reduce the dimensionality of the feature space and to identify the relevant features to be applied for a successful classification or prediction task.

In general, feature selection techniques are organized into two categories: *filter* and *wrapper*, depending on whether or not the selection method includes a learning function.

Filter Method

In a filter method, feature selection and classifier learning are separated in a feature subset, which means that features are selected prior to classification by a separate model. This type of feature selection approach is independent of machine learning algorithm. Figure 2.11 presents the basic structure of a simple filter model. Feature selection starts with a given training set characterized by the full feature set. Various feature subsets are generated and evaluated by the feature subset generator and evaluator. The final specific feature subset is evaluated by training and testing a specific classification model. Finally, ultimate classification accuracy is estimated from the test set.

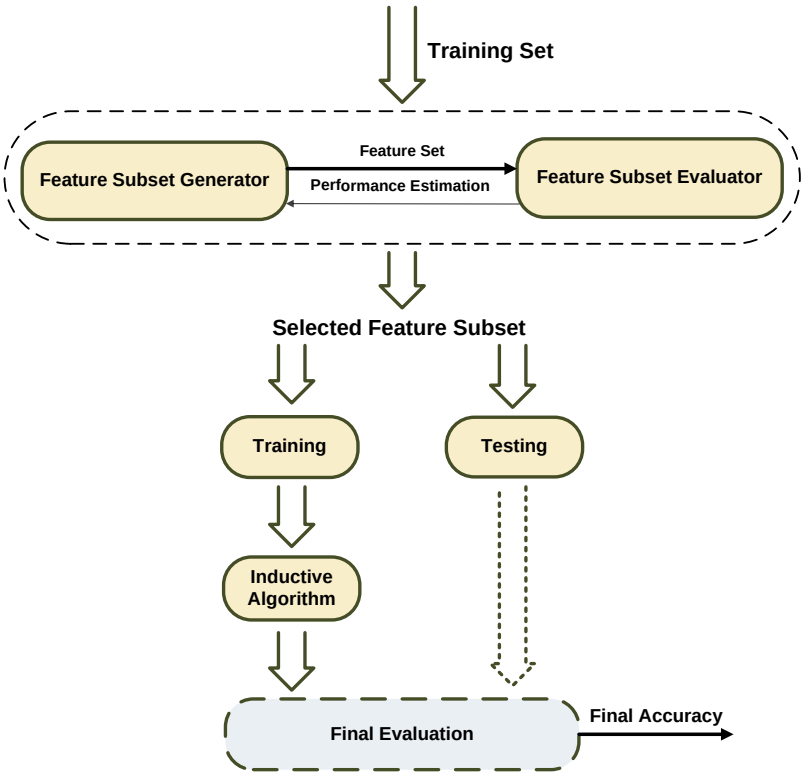


Figure 2.11: *Basic structure of a simple filter model.*

Filter feature selection is one of the simplest and most commonly used feature se-

lection techniques in microarray literature. The advantages of this model are that feature selection requires no machine learning process, and the model is time economical compared to the wrapper model. However, a major drawback of this method is that feature interactions are ignored, which compromises classification performance.

A typical type of filter model is signal-to-noise ratio (SNR) ranking procedure (see Eq.(2.15)). SNR is a supervised method, in which each variable is assigned a ranking number that indicates how well the variable distinguishes two different classes. Moreover, SNR can efficiently reduce the dimensionality of a data set. Basically, this approach begins with the evaluation of an individual feature and iteratively examines the remaining features in terms of statistic ranking score.

Wrapper Method

In the wrapper method, a feature subset procedure is defined, and various feature subsets are generated and evaluated using a feature subset generator and evaluator, respectively. Specific feature subsets are evaluated by training and testing with a specific classification model. The entire feature subset space is then searched by a search algorithm wrapped around the classification model. Figure 2.12 demonstrates the basic structure of a simple wrapper model.

The advantages of the wrapper method are that the features' importance is evaluated by a learning function, leading to much higher performance compared to the filter method. However, the method is more computationally expensive than the filter method, and the evaluation results depend largely on the inductive algorithm (also known as the central machine learning algorithm).

2.4.2 Cross-Validation Techniques

The choice of data splitting/sampling strategy is critical for the verification of final experimental results (BragaNeto & Hashimoto, 2004; Allison & Cui, 2006). Currently, cross-validation is the most popular data splitting method, having been successfully applied to microarray data analysis, performance evaluation of neural networks, and generalization ability estimation of a classifier (also known as generalization error).

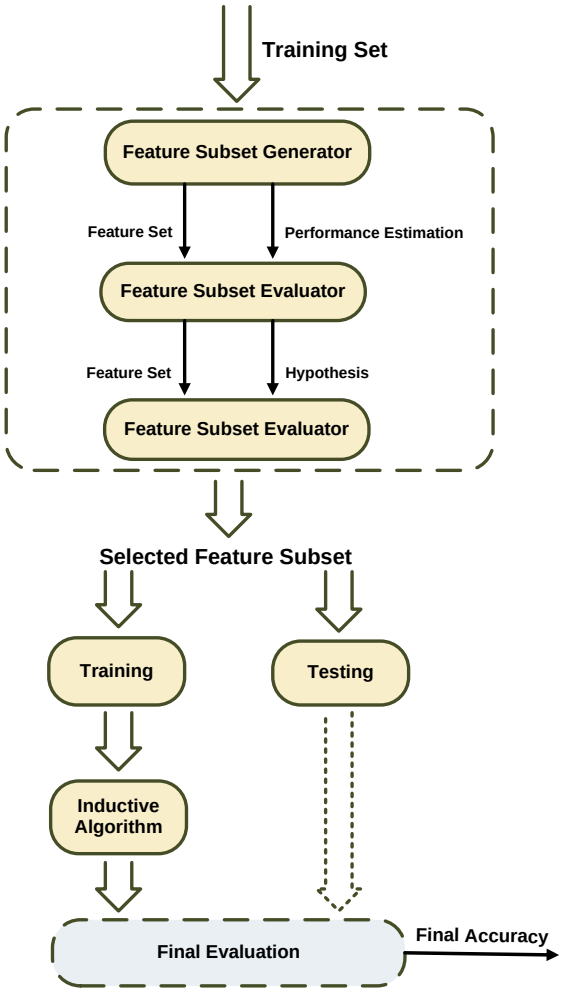


Figure 2.12: *Basic structure of a simple wrapper model.*

Cross-validation (also called "rotation estimation") is defined as an optimal method for measuring the extent to which statistical analysis results can generalize to an independent data set. The available training set is split into two parts: a training set to train the model, and a testing set to estimate the performance of the trained model. The primary goal of this method to reduce generalization error and the possibility of over-fitting is generally accomplished by sequentially omitting parts of the original sample in the available data set prior to perform a multi-variable analysis. The process iterates until all samples in the data set have been estimated (Ransohoff, 2004). A brief overview of two common cross-validation techniques is presented below.

K-fold Cross-Validation

In K-fold cross-validation, the entire data set is randomly divided into K equal-sized subsets. For each of K experiments, the model is tested on an individual sub-sample, while the remaining K-1 sub-samples serve as training data. Cross-validation is iterated K times/folds (commonly 10-fold is used) with each of the K sub-samples being estimated exactly once as the testing data (Figure 2.13 shows a general K-fold cross-validation process). Once all samples have been estimated, the overall generalization error is calculated as the average error rate across the K experiments.

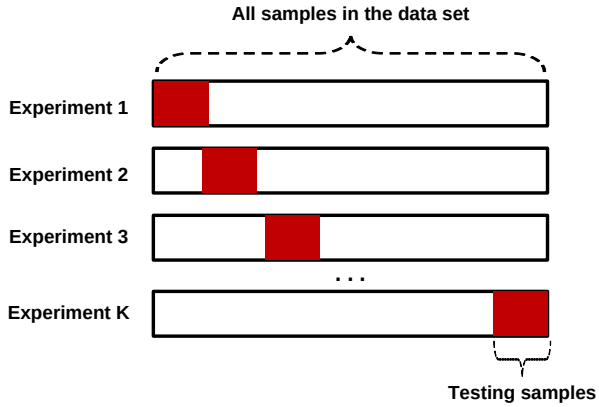


Figure 2.13: *Overview of a general K-fold cross-validation process.*

The advantage of this method is that all samples are used as both training and testing data, and each sample is validated exactly once. On the other hand, the disadvantage of this method is the training process needs to be repeated by K times computations to make an evaluation.

Leave-One-Out Cross-Validation (LOOCV)

The LOOCV algorithm proposed by Craven and Wahba (1979), is an almost unbiased validation schema for the optimal generalization ability of a classifier. LOOCV is a type of K-fold cross-validation, in which the number of folds (K) equals the number of samples (N) in an available data set. In each experiment, this algorithm uses N-1 samples for training and the remaining sample for testing. Thus, the LOOCV process is repeated N times, until every sample in the available data has been trained except

that which is left out for testing (Figure 2.14 shows the general LOOCV process). The final result is the average performance of the N experiments.

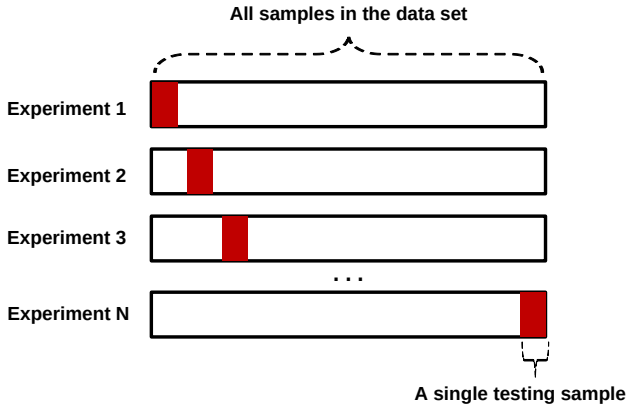


Figure 2.14: *Overview of a general leave-one-out cross-validation process.*

The method ensures economical use of the available data as each pattern is used as both training and testing data. However, the algorithm is very computationally expensive when applied to neural networks due to the large repeat number of the training process.

2.4.3 Performance Measurement

Machine Learning has recently benefited from attention to the performance measures used in classification. Evaluation of learning algorithms concentrates on two goals: algorithm comparison and the applicability of algorithms on specific domains. Various measuring techniques have been developed to evaluate classifier performance, the most popular being the confusion matrix and receiver operating characteristic (ROC).

Confusion Matrix

In general, the performance of a classification model is evaluated from counts of correct and incorrect model predictions. The counts are typically evaluated by a confusion matrix as illustrated in Figure 2.15: The columns represent the predicted class, while the rows represent the actual class. True Positive (TP) and True Negative

(TN) are the correct predictions, while False Positive (FP) and False Negative (FN) are the incorrect predictions.

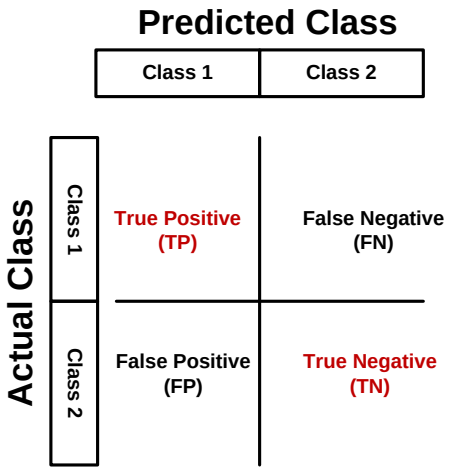


Figure 2.15: *Confusion matrix for 2-class classification problem.*

Although a confusion matrix provides the necessary information for evaluating classification model performance. The performance of different models can be more conveniently compared by summarizing this information as a single number termed the *Accuracy*.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.20}$$

Equivalently, the performance of a classification model can be evaluated terms of *Error rate (1-Accuracy)*:

$$Error\ rate = \frac{FP + FN}{TP + TN + FP + FN} \tag{2.21}$$

Receiver Operating Characteristic (ROC)

Receiver operating characteristic (ROC) first deployed by electrical and radar engineers during World War II to detect enemy objects in battle fields, is also known as signal detection theory. In a pioneering study, Spackman (1989) applied ROC curves in machine learning tasks.

More recently, the ROC curve has become extensively studied in medical decision making field, such as radiology (Obuchowski, 2003; Eng, 2005), bioinformatics (Lasko,

Bhagwat, Zou & Ohno Machado, 2005), and epidemiology (Shapiro, 1999).

In a ROC curve, the true positive rate (Sensitivity) is plotted as a function of the false positive rate (1-Specificity) for different cut-off points of a specified parameter. Each point on the ROC curve represents a sensitivity/specificity pair corresponding to a particular decision threshold. The area under the ROC curve is a measure of how well a parameter can distinguish between two diagnostic groups (e.g. healthy/diseased). *Sensitivity* and *specificity* are defined as follows:

- Sensitivity: the population of correctly identified diseased individuals.
- Specificity: the population of correctly identified healthy individuals.

Figure 2.16 illustrates a typical example of ROC test results for two populations, designated healthy and diseased groups.

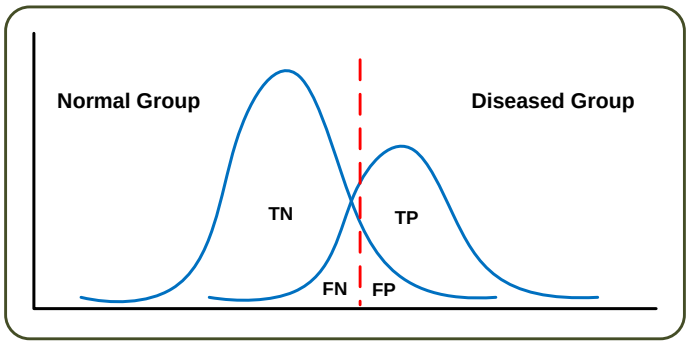


Figure 2.16: *An example of ROC test results for two populations.*

As shown in the figure, for every possible cut-off point selected to discriminate between the two groups, some diseased cases will likely be classified as healthy (FN = False Negative fraction), while others will be correctly classified as diseased (TP = True Positive fraction). On the other hand, some healthy cases will be correctly classified as healthy (TN = True Negative fraction), while others will be classified as diseased (FP = False Positive fraction).

In a ROC curve, FP is plotted on the X axis and TP is plotted on the Y axis. The blue line connecting the stars is a computed ROC curve, while the red solid line represents the perfect classification (see Figure 2.17). In general, the lower left point (0,0) represents the strategy of never issuing a positive classification, whereas the

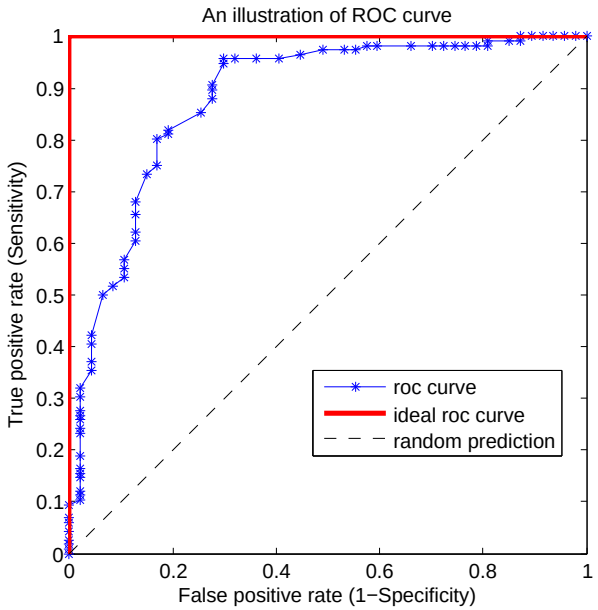


Figure 2.17: *An example of ROC curve.*

upper right point (1,1) represents the opposite strategy of unconditionally issuing positive classification. In addition, the point (0,1) represents perfect classification.

2.4.4 Parameter Optimization

Personalized modeling construction is a complex process that requires evolving and adaptive computational techniques. However, several open questions are raised during the implementation of the personalized modeling framework, including:

- *Which features are significant for every new input vector?*
- *How many nearest neighbors should be selected for every new input vector?*
- *How to find the best combination of model parameters for the learning function (e.g a classifier)?*

Hence, finding an efficient solution to optimize parameters becomes a formidable challenge personalized modeling development. Metaheuristic algorithms are available for solving such problems, and numerous methods for continuous optimization and

heuristics for discrete problems have been developed (Blum & Roli, 2003; T. Y. Chen & Cheng, 2008; X. S. Yang, 2008, 2010). The Greek word *heuristic* means "to find" or "to discover". According to Russell and Norvig (1995), "Heuristics are techniques which seek good (near-optimal) solutions at a reasonable computational cost without being able to guarantee either feasibility or optimality, or even in many cases to state how close to optimality a particular feasible solution is." The heuristic methods include population based, iterative based, stochastic, and deterministic.

In this study, a novel integrated evolving personalized modeling system and framework (evoPM) using a new population-based optimization approach termed gravitational search algorithm (GSA) is proposed. The purpose is to improve the robustness and generalisability of feature, neighborhood selection, and model and its parameters selection. The framework will be applied to classification, diagnostic and prognostic problems. The concept of GSA is introduced in Chapter 4.

2.5 Summary

This chapter compares the inductive and transductive inference approaches, including their operation and areas of application. In addition, it reviews global, local and personalized modeling approaches, explaining the basic theory behind these three approaches and the popular methods of implementing each approach. The inductive approach creates a global model derived from an entire problem space. The obtained model is then applied to every new input data point. In contrast, the transducive approach creates a local model for every new input data based on its nearest neighbors within the existing problem space. Personalized modeling is a type of local modeling that is tailored to an individual vector of the problem space.

More importantly, this chapter discusses several open questions that arise when implementing personalized modeling, such the appropriate data splitting strategy, how the performance of a classifier should be measured, and how to select optimal sets of feature selection, neighborhood and model parameters optimization. These questions and issues will be further studied and addressed in the remainder of this study.

One goal of this study is to further develop the personalized modeling framework

introduced by Kasabov (2007b). To this end, a new method must be established for improved prognostic decision support. The next chapter introduces two emerging contemporary neural models: spiking neural network (SNN) and its extended version, termed evolving spiking neural network (eSNN). The eSNN method will be applied as a classifier to develop the novel intergraded personalized modeling system and framework (evoPM).

CHAPTER 3

Neural Networks (NN)

3.1 Introduction

The remarkable information processing capabilities of the brain have inspired numerous mathematical descriptions of biological neurons. "Neural Network" (NN), also known as artificial neural network (ANN), is a hardware or software computational model that mimics information processing by the biological nervous systems, such as the brain. A number of ANN models have been successfully developed and applied across different disciplines, including medical and business decision support, time series prediction, and pattern recognition. (Kasabov, 2010). However, current ANN models perform rather poorly when applied to complex stochastic and dynamic processes such as biological, environmental, and brain disease process. For this reason, the development of more accurate and efficient biological neural networks is essential to knowledge discovery and information processing.

This chapter provides a brief review of two emerging contemporary neural models: spiking neural network (SNN) and its extended version evolving spiking neural network (eSNN). Since numerous spatio-temporal stroke data have been collected and are available for research purposes. Suitable techniques to properly analyze and process this complex information are imperative. This chapter also introduces

two recently proposed methods for spatio-temporal pattern recognition, namely the extended eSNN framework (EESNN) (Hamed et al., 2011) and the recurrent network reservoir structure of eSNN (reSNN), which uses liquid state machine (LSM) (Schliebs et al., 2011).

3.2 Biological Neurons

Its the center of the nervous system, the brain is extremely complex. The cerebral cortex of the human brain contains roughly $15-33$ billion neurons, perhaps more, depending on gender and age, each linked by up to 10,000 synaptic connections.

The neuron is the fundamental unit of the nervous system. The essential role of a neuron is to receive incoming information and based on that information send a signal to other neurons, muscles, or glands. Neurons are designed to rapidly send signals across physiologically long distances. Despite the diversing of shapes and sizes, typical vertebrate neurons are characterized by four functionally distinct parts: dendrites, soma, synapse and axon.

Essentially, the *dendrites* play the role of the 'input device' that collects signals from other neurons and transmits them to the soma. The main part of the neuron, *soma* contains the genetic information: if the total input exceeds a certain threshold, an output signal is generated. The output signal is intercepted by the 'output device', the *axon*, which delivers the signal to other neurons. The junction between two neurons (the *synapse*) transfers signals between two neurons. A typical biological neuron in the human brain is illustrated in Figure 3.1.

As is well-known, our brain is an exceptionally complex organ, yet we lack complete understanding of even a single neuron's functionality. In recent years, many neural networks researchers have brought their attention to develop more realistic computational neuron models inspired by biological neurons. The ultimate aim is to investigate and model the functionalities of the brain. Such are the cases of the first generation of ANN, the second generation of Neurons, and the third generation of SNN. These models have been applied successfully to diverse fields such as engineering, computer science, and physics.

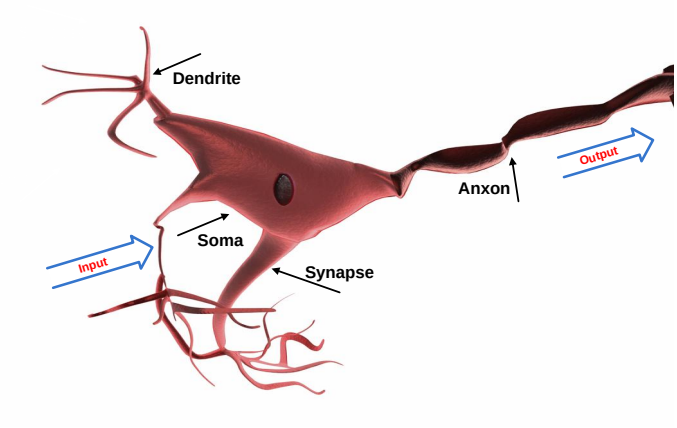


Figure 3.1: *Schematic drawing of a typical biological neuron (Adam, 2005).*

3.3 Evolving Connectionist Systems (ECOS)

"Evolving" is defined as revealing or developing over time in a continuous manner (Kasabov, 2002). Evolving processes are difficult to model as no prior knowledge may exist for some parameters, the results may not be long-term predictable, and unexpected changes might occur at some stage of the development. Thus to facilitate the modeling of evolving processes and knowledge discovery, a novel type of neural network model is proposed, based on the concept of *evolving connectionist systems (ECOS)* (Kasabov, 2002).

ECOS is an adaptive, incremental learning and knowledge representation system that evolves in structure and functionality. The core of the system contains a connectionist architecture consists of interconnected neural networks. ECOS learns local models from data through a set of clusters, where each cluster associated with a local output function. Cluster creation is based on the similarity between data samples in the input space or in both input and output space.

Thus far, several ECOS models have been developed, such as evolving classifier function (ECF) (Kasabov, 2002), dynamic evolving neuro-fuzzy inference (DENFIS) (Kasabov & Song, 2002), and evolving fuzzy neural networks (EFuNNs) (Kasabov, 2002). The eSNN is also based on the principle of ECOS. More information on ECOS can be found in (Kasabov, 1998) and (Watts, 2009).

3.4 Spiking Neural Networks (SNN)

Wolfgang Maass (1997) describes past and current neuron models into three generations. Spiking neural networks (SNNs) is the third generation of neural network models, such models are complex and biologically plausible connectionist model belonging to the ECOS family. All SNN models are compiled from artificial spiking neurons that represent and process pulse-coded information. Figure 3.2 is a simplified diagram of a spiking neuron model.

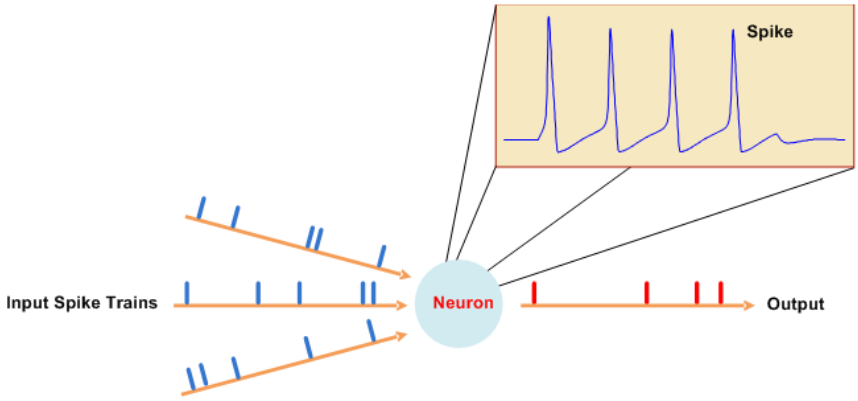


Figure 3.2: *Simplified diagram of spiking neuron model.*

A SNN comprises an encoding method, a neuron model, and a learning method, elucidated as follows:

3.4.1 Design of Encoding

The fundamental problem of any information processing system is information transmission. Biologists have known for over 100 years that neurons transmit information using electrical signals, but the code by which they transmit remains a challenging issue of neuronal coding. Traditionally, neuronal coding is achieved by one of two schemes: rate code and spike/pulse code.

Rate Code

Rate coding sometimes called frequency coding, assumes that the frequency or rate of spike firing increases with rising stimulus intensity.

A crucial factor in rate coding is precise computation of the firing rate. Firing rate may be conceptualized in different ways, depending on the selected averaging procedure. One averaging procedure (shown in Figure 3.3) is to average over a population of neurons (rate as a population activity).

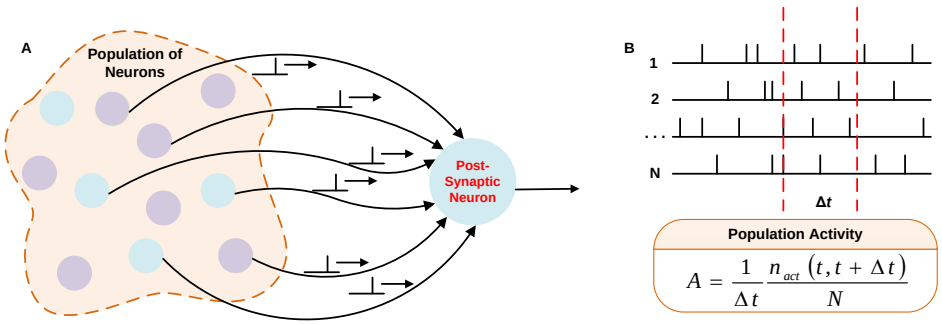


Figure 3.3: ***A*** *- A post-synaptic neuron receives spike input from a sub-population of active pre-synaptic neurons;* ***B*** *- The population activity is calculated as the fraction of neurons that are active within a short time interval* $[t, t + \Delta t]$*, divided by the time period* Δt *and the population size* N

A post-synaptic neuron receives spike input from the population of pre-synaptic neurons that are active. The population activity is defined as the fraction of neurons that are active within a short time interval $[t, t + \Delta t]$, divided by the time period Δt and the population size N:

$$A = \frac{1}{\Delta t}\frac{n_{act}(t, t + \Delta t)}{N} \tag{3.1}$$

where Δt represents the time period; N is the total number of neurons in the population; and $n_{act}(t, t + \Delta t)$ is the number of active spikes triggered within the interval $[t, t + \Delta t]$. The population activity may vary rapidly, allowing fast responses of the neurons to changes in stimulus (Gerstner, 2000).

Spike/Pulse code

Spike/Pulse coding also known as time-to-first-spike, is the second classical scheme of neuronal coding (see Figure 3.4). Its conception was inspired by the visual processing of the human eye. In a pure version of this coding scheme, each neuron transmits information by firing a single spike. If a neuron emits several spikes, the first spike after the reference signal is transmitted; succeeding spikes are ignored. As mentioned by Thorpe et al. (1996), the brain lacks sufficient time to evaluate more than one spike per neuron per processing step. Therefore the first spike should contain most of the relevant information. Since each neuron transmits exactly one spike per stimulus, information is clearly conveyed by timing rather than by spike numbers.

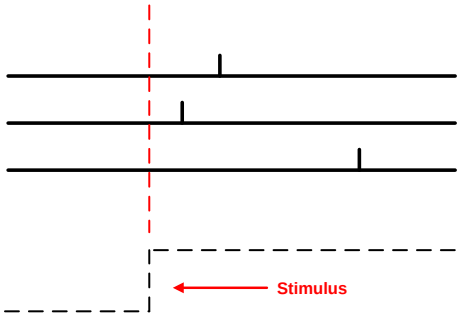


Figure 3.4: *Diagram of time-to-first spike. The second neuron from the top is the first one to fire a spike following a stimulus. The dashed line indicates the time course of the stimulus.*

The important consideration in spike/pulse coding are synchrony and correlation. The neurons representing a similar concept, object or label are "labeled" as firing synchronously (Malsburg, 1981). More generally, any precise spatio-temporal pulse pattern is potentially meaningful and could encode information. Neurons that fire with a specific relative time delay may signify a certain stimulus. An application of spike/pulse coding is population rank order encoding, introduced in the following section.

3.4.2 Neuron Model

To date, the activities of biological neurons have been described in numerous mathematical models. Since neurons are believed to communicate via action potentials,

these models express neuronal behavior in terms of membrane potential and action potential.

Among the existing neuron models of SNN are the Hodgkin Huxley model (Hodgkin & Huxley, 1952), spike response model (Gerstner, 1995; Kistler & Gerstner, 1997; Gerstner & Kistler, 2002a), integrate-and-fire model (Gerstner & Kistler, 2002a; Maass & Bishop, 1999), and Izhikevich model (Izhikevich, 2004, 2007; Izhikevich & Edelman, 2008). This section provides a brief overview of the leaky integrate-and-fire (LIF) model probably the best-known and widely used spiking neuron model. The desirable feature of LIF include clarity of coding, enabling of mathematical analysis of network dynamics, and relatively efficient simulation of large networks.

Leaky Integrate-and-Fire (LIF) Model

The LIF model is a simple model proposed by Louis Lapicque (1907). LIF is still extensively used to understand the behavior of many excitable systems. Lapicque modeled the neuron as an electric circuit consisting of a capacitor C in parallel with a resistor R driven by an external current $I(t)$, where both C and R are assumed constant (see Figure 3.5).

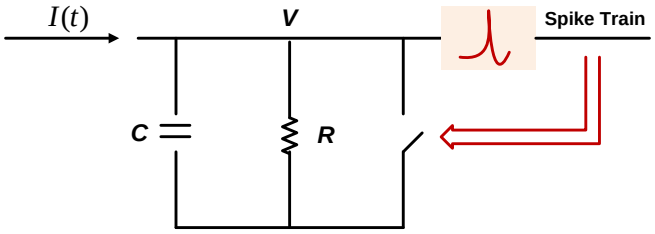


Figure 3.5: *Illustration of a leaky integrate and fire model. The discrete pulses of the rate neuron are replaced by a continuous output rate.*

The current $I(t)$ is split into two components:

$$I(t) = I_R + I_C \tag{3.2}$$

where I_C charges the capacitor C and I_R is the current passing through the resistor R. $I_C = C\frac{du}{dt}$ in terms of capacitance, I_C is expressed as where u is the voltage across the capacitor (equal to voltage across R). I_R is calculated from Ohm's law as

$I_R = \frac{u}{R}$. Thus we obtain

$$I(t) = \frac{u(t)}{R} + C\frac{du}{dt} \tag{3.3}$$

The voltage $u(t)$ across the capacitor represents the membrane potential. The voltage scale is chosen such that $u(t) = 0$ is the resting potential. The temporal evolution of $u(t)$ is:

$$T_m \frac{du}{dt} = -u(t) + RI(t) \tag{3.4}$$

where T_m is the membrane time constant of the neuron, and u is the membrane potential.

The form of an action potential in the integrate-and-fire model is not given explicitly. Spikes are events characterized by a firing time $t^{(f)}$:

$$t^{(f)} : u(t^{(f)}) = \vartheta \tag{3.5}$$

That is, the membrane potential $u(t)$ is compared with the threshold value ϑ. If $u(t) = \vartheta$ in time $t^{(f)}$, then an action potential is output.

Once a spike has fired, the next spike cannot occur during the refractory period, in other words, the potential is reset to a new value $u_{reset} < \vartheta$.

$$\lim_{t \to t^{(f)}, t > t^{(f)}} u(t) = u_{reset} \tag{3.6}$$

In fact, Thorpe's neuron model is the simplified LIF model, the general idea of this model is introduced in a later section.

3.4.3 Learning Algorithm

Learning how to recognize the temporal information contained in spike trains is a crucial factor in SNN. Various SNN learning algorithms have been proposed to date, enabling spike trains to be processed in close to real-time (Hopfield & Brody, 2000, 2001; Maass, Natschlager & Markram, 2003). Similarly to traditional neural networks, learning in SNN may be: supervised or unsupervised (Kasinski & Ponulak, 2006).

Spike Time Dependent Plasticity (STDP)

The most commonly used unsupervised learning rule in SNN is STDP derived from Hebbs law (Hebb, 1949). STDP embodies long-term potentiation (LTP) and depression (LTD), where depends on the output of a neuron spike time and transmission time. Efficacy of synapses is strengthened or weakened based on the relative timing between post-synaptic action potential and pre-synaptic spike. Through STDP, connected neurons learn consecutive temporal associations from data and new connections are evolved. If a pre-synaptic spike arrives at the synapse before the post-synaptic action potential, the synapse is potentiated as long-term potential (LTP); reversing this temporal order causes long-term depression (LTD) (Kempter, Gerstner & van Hemmen, 1999; Bi & Poo, 2001; Gerstner & Kistler, 2002b).

Mathematically, the function $W(t_{pre} - t_{post})$, also referred to as the STDP window describes the STDP learning rule. The change of synaptic weight depends on the difference between the arrival time t_{pre} of a pre-synaptic spike and the time t_{post} of an action potential emitted by the neuron.

$$W(t_{pre} - t_{post}) = \begin{cases} A_+ exp(\frac{t_{pre}-t_{post}}{\tau_+}) & \text{if } t_{pre} < t_{post} \\ A_- exp(-\frac{t_{pre}-t_{post}}{\tau_-}) & \text{if } t_{pre} > t_{post} \end{cases} \tag{3.7}$$

where parameters τ_+ and τ_- represent the time interval of the pre-synaptic and post-synaptic activity, respectively; and A_+ and A_- indicate the maximum fractions of synaptic modification at $t_{pre} - t_{post}$ close to zero.

3.4.4 Liquid State Machine (LSM)

Liquid state machine (LSM) can be best explained by a simple example: imagine a rock and a pond and throw the rock into the water. In fact, the rock is a low-dimensional temporal input: the rock and throw have some properties but these are only expressed very briefly. The resulting splash and ripples that are created can be seen as the reaction, or *liquid state*. These ripples propagate over the water's surface for a while and will interact with the ripples caused by other recent events. The water can thus be said to retain and integrate information about recent events, so if we're somehow able to read the water's surface we can extract information about

what has been recently going on in this pond. We refer to this trained spectator as a readout unit that we can ask at any one time what's going on in the pond, provided that we can show him a picture of the water's surface.

In general, LSM consists of two separate components: the *liquid*, which yields a complex time-varying vector state; and the *readout function* is a memory-free subsystem that extracts information from the liquid. Figure 3.6 is a schematic of the LSM: a reservoir of recurrently interacting nodes is stimulated by the input $u(t)$, a liquid state $x(t)$ is extracted and a readout function f converts the high-dimensional liquid state $x(t)$ into the desired output $y(t)$ for the given task.

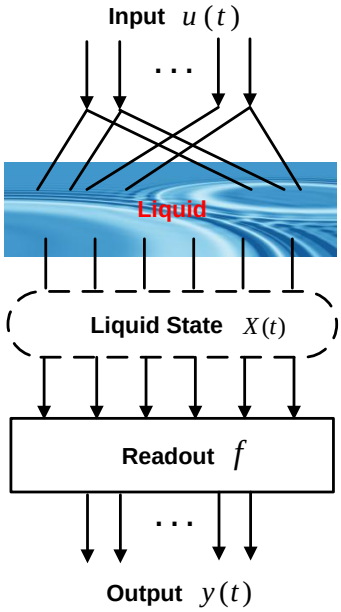


Figure 3.6: *Architecture of the liquid state machine (LSM).*

The LSM is a network of spiking neurons that can map complex spatio-temporal data (STD) into a high dimensional space where new patterns can be recognized from the firing of hundreds of thousands of neurons (Maass, Natschlager & Markram, 2002). It is a novel computation learning paradigm based on the transient dynamics of recurrent neural circuitry. As a form of reservoir computing, it constructs a recurrent neural network of spiking neurons, for which all parameters (such as connectivity, neural parameters, and synaptic weights) are randomly chosen and fixed during simulation.

As mentioned in the literature (Destexhe & Contreras, 2006; Yamazaki & Tanaka,

2007), some parts of the mammalian brain might act as a liquid generator while others learn how to interpret the liquid perturbations caused by external sensory stimuli. From this viewpoint, LSM mimics brain-like information processing, analysis may lead to very powerful computational tools, as well as providing further insights into the functioning of the mammalian brain.

The concept of reservoir as proposed by Maass et al. (2002) is composed of LIF neurons. When the network inputs are transferred into a high-dimensional space, they become easily separated. Thus, a readout function maps reservoir states into a desired class label. Because it uses recurrent networks, the reservoir can process temporal information present in the input signals. LSM integrated with the reservoir paradigm is becoming a popular means of processing STD.

3.4.5 Applications of SNN

SNN has been increasingly applied in various disciplines for solving complicated prediction and classification problems, such as speech recognition (Yau, Kumar & Arjunan, 2007), audio and video analysis (Fyfe, Barbakh, Ooi & Ko, 2008; Tsapatsoulis, Rapantzikos & Pattichis, 2007), and financial forecasting (Schneider & Graupe, 2008).

Currently, SNN is becoming to be a powerful computational tool that can diagnose and monitor the prognosis of a disease. Over 500 papers on neural network applications in medicine are now published per year (Gant, Rodway & Wyatt, 2001). SNN has been successfully used in breast cancer diagnosis (Kiyan & Yildirim, 2003), assessing prognoses for patients with congestive heart failure (Cowburn, Cleland, Coats & Komajda, 1996), predicting the risk of death for lung cancer patients (Bartfay, Mackillop & Pater, 2006), and predicting functional outcome associated with clinical variables of stroke rehabilitation (Oczkowski & Barreca, 1997).

As identified in some existing studies, the predictive accuracy of SNN is strong compared to that of the classical approaches. However, the ability of current SNN models to solve complex real world problems is limited. Therefore, novel SNN models are required.

3.5 Evolving Spiking Neural Networks (eSNN)

SNN has been successfully extended to eSNN which is modeled on the neural processing of the human eye. Proposed by Wysoski, Benuskova and Kasabov (2006), eSNN performs considerably better than previously published models in solving complex classification tasks, including taste recognition (Soltic, Wysoski & Kasabov, 2008), face recognition (Wysoski, Benuskova & Kasabov, 2008), and person authentication based on audiovisual information (Wysoski, Benuskova & Kasabov, 2007). A simplified diagram of evolving spiking neuron model is shown in Figure 3.7.

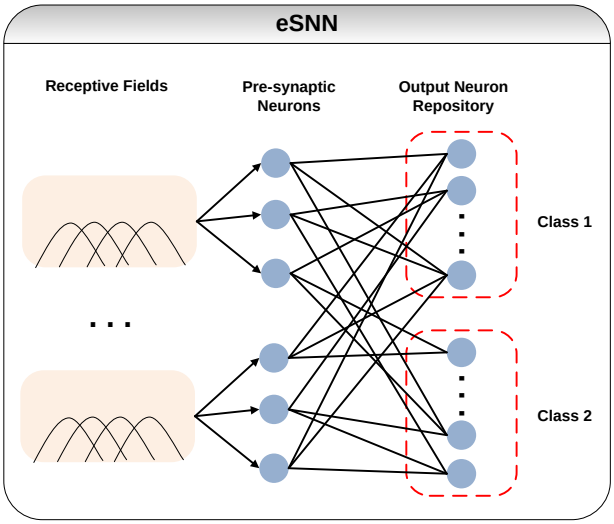


Figure 3.7: *Simplified diagram of evolving spiking neuron model.*

Like SNN, an eSNN model consists of an encoding method for transforming the real-valued data into a spike train, a neuron model, and a learning method.

3.5.1 Population Rank Order Encoding

Because information in an eSNN model is represented as series of spikes, real-valued data inputs must first be converted into spike train. A number of SNN encoding methods have been proposed such as poisson processes, rank order encoding, and frequency mappings.

A well-know eSNN encoding method is population rank order encoding, where a

single input value is encoded into multiple pre-synaptic neurons M. Each pre-synaptic neuron generates a spike at a certain firing time. Analogous to arrays of receptive fields, Bohte et al. (2002) suggest that the firing time could be calculated from the intersection of a Gaussian function and a neuron. This encoding approach allows the encoding of continuous values by using a population of neurons with overlapping sensitivity profiles. The Gaussian center and width are computed from Equations 3.8 and 3.9 respectively, where the variable interval is $[l_{min}, l_{max}]$. The parameter β controls the width of each Gaussian receptive field. Figure 3.8 illustrates the operation of population rank order encoding.

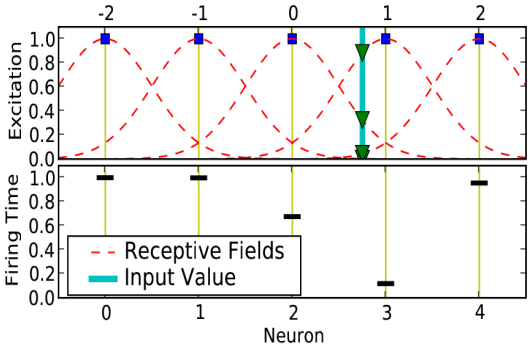


Figure 3.8: *Population rank order encoding based on Gaussian receptive fields.*

$$\mu = l_{min} + \frac{2i-3}{2}\frac{l_{max} - l_{min}}{M-2} \tag{3.8}$$

$$\sigma = \frac{1}{\beta}\frac{l_{max} - l_{min}}{M-2}, 1 \leq \beta \leq 2 \tag{3.9}$$

3.5.2 Spiking Neuron Model based on Population Rank Order Encoding

Thorpe's neuron model is adopted in eSNN due to its simplicity and effectiveness. The basic philosophy behind this model is that earlier spikes received by a neuron are weighted more heavily than later spikes. If the neuron intercepts a certain number of spikes and the post-synaptic potential (PSP) is larger than a threshold θ, a spike

is triggered, and the PSP is set to 0 for the rest of the simulation, even if the neuron remains stimulated by incoming spike trains (See Figure 3.9).

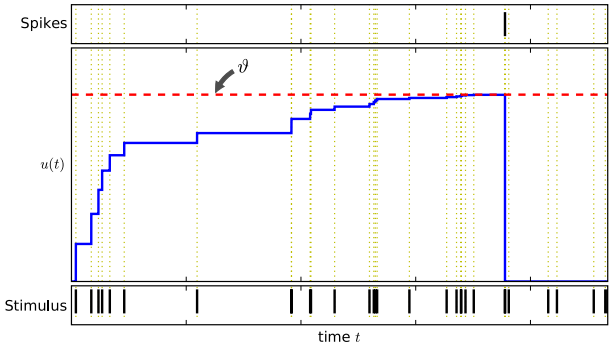


Figure 3.9: *A spike is triggered when the total spiking input-PSP exceeds the threshold θ, and the PSP set to 0 for the rest of the simulation.*

Equation 3.10 describes the PSP dynamics of a neuron i in the neuron model:

$$PSP_i(t) = \begin{cases} 0 & \text{if neuron fired} \\ \Sigma w_(ji) * M_i^{order(j)} & \text{else} \end{cases} \tag{3.10}$$

where $w_(ji)$ represents the weight of a pre-synaptic neuron j; $M_i \in [0, 1]$ is a parameter termed the modulation factor and $order^{(j)}$ represents the rank of the spike triggered by the neuron j. The $order^{(j)}$ is 0 if neuron j is the first pre-synaptic neuron to spike, and increases with firing time.

3.5.3 One-Pass Learning Algorithm

The learning algorithm applied to eSNN is a one-pass algorithm, in which the trained network learns new samples without retraining previously learned samples (S. J. Thorpe, 1997). In this algorithm, each training sample generates a new output neuron, which is then compared with existing neurons in the repository. If the newly trained neuron is very similar to those stored in the repository (according to a specified similarity threshold), it will be merged with the most similar stored neuron. Otherwise, it is added to the repository as a new output neuron. The merging process is implemented by modifying the connection weights and the threshold of the merged neurons to their average value. The pseudo code of the eSNN training algorithm is provided in Algorithm 1.

Algorithm 1 eSNN Training Algorithm

Require: $Mod \in [0,1]$, $Sim \in [0,1]$, $C \in [0,1]$

Initialize neuron repository $R = \{\}$
for Every input samples i belonging to the same output class **do**
  Encode every input sample into pre-synaptic neurons j
  Generate a new output neuron and compute the connection weights: $w_j = Mod^{order(j)}$
  $PSP_{max(i)} = \Sigma w_(j) * Mod_i^{order(j)}$
  $\theta(PSPthresholdvalue) = PSP_{max(i)*C}$
  if The new trained neuron is too similar to the ones already in the R (according to a specified similarity threshold) **then**
    $w^{(n)} \leftarrow$ merge $w^{(i)}$and $w^{(n)}$
    $\theta^{(n)} \leftarrow$ merge $\theta^{(i)}$and $\theta^{(n)}$
  else
    Add the new neuron to the output neuron repository R
  end if
end for

3.6 Personalized SNN Reservoir based Generic Method for Spatio-Temporal Data

Space and time are the two important components of real world phenomena. Conventional datasets generally contain either temporal or spatial information. In contrast, a spatio-temporal dataset (STD) manages both forms of information; the data change and evolve with time. Vast quantities of STDs, including medical, brain signals, weather forecast, environment monitoring, and audio/visual. Given the multifaceted nature of STD, the efficient and accurate analysis of these data presents a major challenge.

The reservoir acts as an intermediate recurrent neural network that captures an input and maps it into a high-dimensional output to enhance the separability of the incoming data. Next, an external classifier or a readout function transforms the responses from the reservoir into the desired class label for final decision making. Since the state of the reservoir depends on temporal information present in the input signals, it is an appropriate tool for STD analysis.

3.6.1 Spatio-Temporal Data (STD)

Approximately 80% of available datasets contain interrelated spatial and temporal components (Fayyad & Grinstein, 2001). Such data include:

- *Ecological data* - environment monitoring, moving storms, changes in atmospheric pressure level;
- *Biological data* - species relocation, mating behavior, and animal movements;
- *Forestry data* - forest fires, planning tree planting and cutting;
- *Transport data* - vehicle movement and traffic monitoring.

Recently, the quantity of available STDs is expanding exponentially; thus, suitable techniques that incorporate human expertise to effectively and efficiently analyze and process these data are urgently required. Spatio-temporal data mining is an emerging approach for discovering or extracting the "implicit knowledge, spatial and temporal relationships, or other patterns not explicitly stored in spatio-temporal datasets" (Koperski, Han & Adhikary, 1998).

Several conventional techniques have been developed for STD processing. Among the most popular are Hidden Markov models (HMM) (Rabiner, 1989), recurrent Elman networks (Elman, 1990), and time delay neural networks (TDNN) (Waibel, Hanazawa, Hinton, Shikano & Lang, 2002). However, existing statistical and computational methods are insufficient for the following reasons:

- Although STD is embedded in continuous space, conventional datasets are generally discrete;
- The patterns in the STD tend to be localized, whereas classical methods normally focus on global patterns;
- Because existing methods model either space or time separately, or mix both components in a simple way, they fail to capture some essential relations between STD variables.

To satisfy STD processing demands, the new generation of data modeling techniques must be able to train new STD efficiently, accurately and incrementally. As mentioned previously, eSNN as an extension of SNN is an emerging computational technique for STD analysis. This model can learn STD by first transferring temporal changes occurring in the input variables into spike trains (binary temporal events) and then applying learning procedures to map spatio-temporal patterns detected in the data into temporal spiking activity of spatially located neurons. The next subsection introduces two recently developed extended eSNN models that adopt the reservoir computing paradigm in solving STD classification problems.

3.6.2 Extended eSNN (EESNN)

The proposed EESNN model implemented by Hamed (Hamed et al., 2011), extends the original eSNN by adding a new layer that captures the entire STD pattern. The hybrid approach of EESNN is shown in Figure 3.10. The model comprises two major layers: (1) the first acts as a memory to capture the entire STD pattern; (2) and the second is the standard eSNN classifier (operating through the LOOCV schema) that learn the response from the first layer.

In the first layer, each real-value spatial data vector valued at every time moment is encoded into spike trains using the population rank order encoding scheme and is stored in memory. This encoding distributes a single input value into multiple neurons. The spike trains are then injected into the memory to capture their temporal information, and to map the entire STD input pattern into a single high-dimensional spiking neuron structure. The spiking time of the neurons reflects the values of the input variables at every time point of STD measurement. The obtained high-dimensional spiking neuron structures are then fed into the second layer for classification.

In the second layer, responses from the first layer are learned using a fast one-pass learning algorithm of eSNN that enables adaptive and incremental learning of spatio-temporal patterns. Thorpe's neuron model fires an output spike after sufficient spatio-temporal spike trains are received. In the classification process, the learned output for every sample is compared with the target output. The pseudo code of the EESNN algorithm is presented in Algorithm 2.

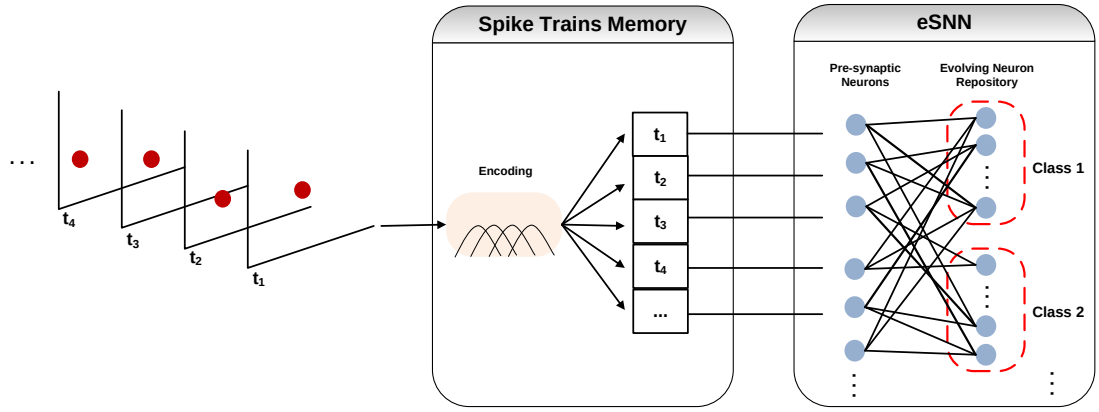


Figure 3.10: *The framework of the extended eSNN (EESNN) model.*

Algorithm 2 EESNN Algorithm

1: **for** all samples in class c **do**
2: **for** all time points **do**
3: Encode every real-value spatial data vector into spike trains
4: **end for**
5: Accumulate all spike trains for all time points in a memory
6: **end for**
7: Apply spike memory into standard eSNN for a classification task

3.6.3 Recurrent Network Reservoir Structure of eSNN (reSNN)

The concept of reSNN was introduced by Schliebs et al. (2011) and was applied to reservoir computing (Maass et al., 2002) for efficient processing of spatio-temporal data. The framework of reSNN is illustrated in Figure 3.11. Note that an additional layer exists between the reservoir and classifier, which denotes the liquid state.

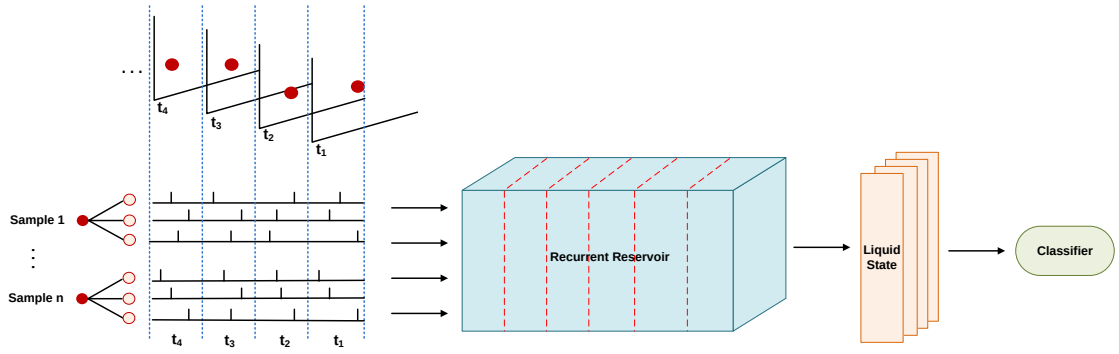


Figure 3.11: *The framework of recurrent network reservoir structure of reSNN model.*

In the first step, as for the EESNN model, each real-value of the spatio-temporal data vector is encoded into a spike train by population rank-order encoding. Thus, a series of spike trains for all pre-synaptic input neurons is generated, each attached to a input neuron within the reservoir. The complete input spike trains are continuously fed into the reservoir in temporal order; spikes that fire sooner are fed first followed by later ones.

Once all the spike trains have been injected into the reservoir layer, the reSNN reservoir acts as a large recurrent neural network whose topology and connection weight matrix is fixed during simulation. In this way, the reSNN accumulates the temporal information of all input spike trains and transforms them into a single high-dimensional intermediate liquid state. The recurrent reservoir is designed on the integrate-and-fire (LIF) neuron principle and is commonly referred to as a liquid state machine (LSM) that mimics the brain-like information processing of the human eye.

The recurrent reservoir generates unique accumulated neuron responses to different classes of input spike trains from different samples. Once the pre-defined simulation

time has elapsed, all reservoir responses are transformed into liquid states. However, before performing classification, the liquid states at a given time t must be read out from the reservoir. Three major types of readouts are in popular use, such as cluster, frequency and analog readouts. The reSNN adopts the analog readout approach, in which every spike is convolved by an α-kernel function according to Equation 3.11

$$\alpha(t) = e\tau_s^{-1} t e^{\frac{-t}{\tau_s}} \Theta(t) \tag{3.11}$$

where τ_s is the synaptic time constant, and $\Theta(t)$ is the Heaviside function defined as

$$\Theta(t) = \left\{ \begin{array}{ll} 0 & \text{if } t < 0 \\ 1 & \text{if } t \geq 0 \end{array} \right\} \tag{3.12}$$

Thus, a convolved spike train $\widetilde{s}(t)$ is computed as:

$$\widetilde{s}(t) = \sum_{t^f} e\tau^{-1}(t - t^f) e^{-\frac{(t-t^f)}{\tau}} \Theta(t - t^f) \tag{3.13}$$

where the parameter t^f represents the firing time of a neuron.

The final decision is made by passing liquid states at all time points to the classification layer. Algorithm 3 lists the pseudo code of the reSNN algorithm.

Algorithm 3 reSNN Algorithm

```
for all samples in class c do
    for all time points do
        Encode every real-value spatial data vector into spike trains
    end for
    Accumulate all spike trains for all time points in a memory
end for
for all spike trains do
    Inject into the recurrent reservoir
    Generate reservoir responses based on the neuron spikes
    Produce the liquid states from reservoir responses
end for
Apply liquid states to classifier/readout function for a classification task
```

As discussed above, several differences exist between EESNN and reSNN models. These are summarized below:

- The memory structure of EESNN comprises simple spike trains and requires no internal learning in the reservoir, thus it runs much faster than the reSNN model;
- The reSNN model possesses a more complex reservoir structure comprising an integrated recurrent network and LIF neurons. Therefore, it requires more computational time and resources than EESNN model;
- However, in the reSNN model, the liquid states can be extracted at any time point and passed to a classifier or a readout function to perform the classification task.

3.7 Summary

This chapter reviews in detail two novel neural technologies, namely SNN and eSNN, with focus on their encoding methods, neuron models, and learning methods. SNN is considered as the third generation of brain-inspired neural network methods. SNN learns temporal data by first transforming temporal changes occurring in the input variables into spike trains (binary temporal events) and then applying learning procedures to map spatio-temporal patterns detected in the data into temporal spiking activity of spatially located neurons. eSNN is a extension of SNN which has been successfully applied to complex classification tasks. In this study, eSNN is utilized as a classifier in novel integrated evolving personalized modeling systems. Its parameters will be optimized simultaneously with the features and neighborhood by gradational search algorithm (GSA), an evolutionary algorithm that enables effective and efficient decision making and knowledge discovery. Evolutionary computation and algorithms, in particular, the novel GSA optimization method will be described in the next chapter.

In addition, this chapter provides a brief literature review of two personalized SNN reservoir-based models: EESNN and reSNN. One aim of this PhD study is to apply these two models to spatio-temporal classification tasks and to evaluate their feasibility on a case study involving spatio-temporal weather and stroke occurrence data. The details of the study are presented in Chapter 9.

CHAPTER 4

Evolutionary Computation and Algorithms

4.1 Introduction

A vast diversity of species exists in nature. How does mankind evolve among such enormous variety? In other words, how does nature solve the optimization problem of perfecting mankind? This question may be answered in Charles Darwin's theory of evolution (1859). Evolution embodies the development of generations of individual populations governed by fitness criteria. Natural evolution has inspired the development of computational methods collectively known as evolutionary computation (EC).

This chapter provides a brief introduction to EC and to a recently developed population-based heuristic optimization approach called gravitational search algorithm (GSA). In this study, GSA is chosen to integrate with personalized modeling concept because: (1) It has been successfully applied to various types of complex optimization problems, and converges to the global optimum much more rapidly than many classical optimizers (Rashedi, Nezamabadi-pour & Saryazdi, 2009, 2010); (2) It can solve multi-objective optimization, where neighborhood and model parameters optimization is undertaken by a continuous (real-valued) RGSA, while feature selection uses

a discrete (binary-valued) BGSA; (3) GSA is a emerging technique not previously applied to personalized modeling before, thus it offers novelty compared to existing personalized modeling methods.

4.2 Evolutionary Computation (EC)

EC is the collective name for a range of problem-solving techniques inspired by biological mechanisms of evolution, such as natural selection and genetic inheritance. Improved optimization, robust adaptation, machine intelligence, and facilitating a greater understanding of biology are the main driving forces of EC development.

EC includes the evolutionary algorithm (EA), a powerful optimization method based on generic populations. Several types of evolutionary methods have been developed (Back, 1996), such as the genetic algorithm (GA) (Holland, 1975), which optimizes general combinatorial problems; evolution strategy (ES) (Rechenberg, 1973), which optimizes continuous functions with recombination; evolutionary programming (EP) (Fogel, Owens & Walsh, 1966), which optimizes continuous functions without recombination; and genetic programming (GP) (Koza, 1992), which evolves programs.

The mechanisms of EA are inspired by biological evolution, which operates by reproduction, mutation, recombination, and selection. The general scheme of an EA is given in Figure 4.1.

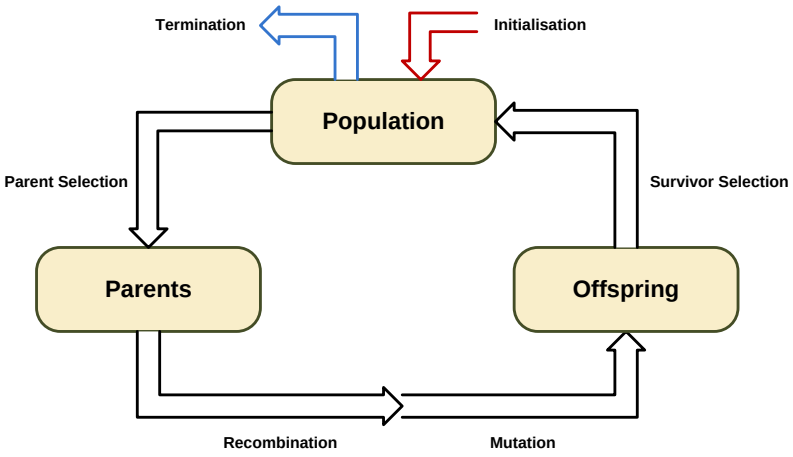


Figure 4.1: *Flow-chart of an evolutionary algorithm (EA).*

As is evident in the Figure, EA methods include two principle stages:

1. Creation of new population of individuals;
2. Development of the individual systems, such that a system develops and evolves through interaction with the environment, which itself depends on the genetic material embodied in the system.

4.2.1 Advantages of EA

Some of the advantages of using evolutionary algorithms rather than other global optimization techniques are given below (Fogel, 1999):

- The framework offered by EA much more easily accommodates prior knowledge on the problem. Incorporating such information focuses the evolutionary search, providing a more efficient exploration of the state space of possible solutions;
- EA can be combined with traditional optimization techniques. This may be as simple as a gradient minimization after primary search with an evolutionary algorithm (e.g. fine tuning of weights of an evolutionary neural network) or it may involve simultaneous application of other algorithms (e.g. hybridizing with simulated annealing or Tabu search to improve the efficiency of basic evolutionary search);
- Each solution can be evaluated in parallel and selection alone (which requires at least pair-wise competition) requires some serial processing. Implicit parallelism is not possible in many global optimization algorithms such as simulated annealing and Tabu search;
- Traditional methods of optimization are not robust to dynamic changes in the environment and often require a complete restart to provide a solution (e.g. dynamic programming). In contrast, evolutionary algorithms can adapt solutions to changing circumstance;
- The greatest advantage of evolutionary algorithms is that they can address problems unknown to human expertise. Although human expertise should be

used when available, it often proves less than adequate for automating problem solving routines.

4.2.2 Applications of EA

EA are ubiquitous to date, having been successfully applied to multi-domain applications, including:

- *Planning*

  One of the best known combinatorial optimization problems is the traveling salesman problem (TSP). A salesman must visit a number of cities, and then return home. In which order should the cities be visited to minimize the distance traveled? Optimizing the tradeoff between the speed and accuracy of solution is an ongoing aim of optimization algorithm (Verhoeven, Aarts, van de Sluis & Vaessens, 1992).

- *Controlling*

  Some researchers (Fogel et al., 1966; DeJong, 1980) have applied the adaptive qualities of EA to build on-line controllers for dynamic systems. Fonseca and Fleming (1993) used an EA to design a controller for a gas turbine engine that optimize the step response of the engine.

- *Economics*

  Oliver (1993) formulated rules to reflect the way in which consumers choose one brand rather than another, when a product can be judged by multiple criteria. A fuzzy hybrid system has been used for financial decision making, with applications to credit evaluation, risk assessment, and insurance underwriting.

- *Biology*

  EA has been applied to the difficult task of protein secondary-structure determination, for instance, classifying the locations of particular protein segments (Handley, 1993).

4.2.3 Methods of EA

To date, various evolutionary optimization techniques have been developed for tuning the optimal set of model parameters and/or the optimal feature set. Some of the more popular ones are given below:

- *Genetic Algorithm (GA)* (Holland, 1975)

  GA formally introduced in the 1970s by John Holland, was inspired by Darwin's theory of evolution. It works particularly very well on mixed (continuous and discrete) combinatorial problems.

  A common type of GA works operates as follows: a population is randomly created from a group of individuals. The individuals in the population are then evaluated by a provided evaluation function, and are scored based on their performance in the given task. The individuals are then selected by fitness, the higher the fitness, the higher the chance of being selected and vice-versa. These selected individuals then "reproduce", yielding one or more offspring, after which the offspring are mutated randomly. This continues until a optimal or near-optimal solution has been found or a certain number of generations have passed, depending on the termination conditions.

  However, GA is computationally time-expensive. Furthermore, GA does not guarantee globally optimal solution, for instance, when the populations have a lot of subjects. In addition, for noise data, convergence is rendered difficult and local optimization might yield a meaningless result.

- *Artificial Immune System (AIS)* (Farmer, Packard & Perelson, 1986)

  AIS is a population-based algorithm inspired by the biological immune system. It is applied to real-world problems such as numerical optimization and combinatorial optimization problems.

  In AIS, the initial population is randomly generated and its size can grow and shrink dynamically. In the cloning step, each antibody of the population generates a number of clones. Because no antibody has a selective advantage over the others, the algorithm can perform multi-model searching. These clones are assigned mutations at rates inversely proportional to their fitness: clones with higher fitness will be submitted to lower mutation rates and vice-versa.

Following the insertion of clones into the population, the antibodies form an interactive network. If two or more antibodies present a degree of similarity above a given threshold, all but one are eliminated from the population. Individuals with low fitness are also excluded. This process avoids redundancy and therefore tends to preserve population diversity.

The main weakness of AIS is normally the additional parameters required, some of which may be difficult to fine tune for an arbitrary problem. Furthermore, the current multi-objective artificial immune systems have focused on the solution of standard test functions, rather than on applications.

- *Ant Colony Optimization (ACO)* (Dorigo, 1992)

  ACO was first proposed by Marco Dorigo in his PhD thesis. The original algorithm searches for an optimal path in a graph, based on the behavior of ants seeking a path between their colony and food source. ACO is a population-based metaheuristic paradigm designed for solving combinatorial optimization problems.

  The essences of ACO algorithms are as follows: each path followed by an ant is associated with a candidate solution to a given problem. When an ant follows a path, the amount of pheromone deposited on that path is proportional to the quality of the corresponding candidate solution to the target problem. When an ant must choose between two or more paths, the path(s) with more pheromone are more likely to be chosen by the ant. As a result, the ants eventually converge to a short path, hopefully the optimum or a near-optimum solution to the target problem, as do natural ants.

  Although of ACO guarantees convergence, the time to convergence is uncertain, and the probability distribution changes with each iteration.

- *Particle Swarm Optimization (PSO)* (Kennedy & Eberhart, 1995)

  PSO is a population-based stochastic optimization technique proposed by Kennedy and Eberhart in 1995, inspired by social behavior of bird flocking, fish schooling and swarm theory. It has been developed for continuous, discrete, and binary problems.

  In PSO, the potential solution called *particle*, is assigned a random position and velocity. Each particle keeps track of its best-fit (up to the current iteration) coordinates in the problem space and records them as *pbest*. Another "best"

value that is tracked by the particle swarm optimizer is the best value obtained so far by any particle in the particle neighborhood. This location is called *lbest*. When the topological neighbors of a particle comprise the entire population, the best value is a global best and is designated *gbest*.

At each time step, particle swarm optimization alters the velocity of each particle toward its *pbest* and *lbest* locations. Acceleration is weighted randomly, with separate random numbers assigned for acceleration towards the *pbest* and *lbest* locations.

The main drawbacks of PSO when used for multi-objective optimization are: (1) The method easily suffers from partial optimization, which compromises the accuracy of its speed and direction regulation; (2) Diversity is not easily controllable. The loss of diversity is generally compensated by mutation operators. However, the role of the PSO parameters in algorithm convergence and loss of diversity are incompletely understood; (3) The criteria by which leaders are selected also seems to play a critical role in multi-objective optimization, but this effect has been little investigated.

The above algorithms are considered as classical optimization techniques and have been applied to different data analysis problems related to medical decision support, e.g. gene expression data for cancer diagnosis. Nevertheless, common drawbacks of these models are computational expense and the uncertain time to convergence, especially on complex real-world problems. For example, though GA is noted for its robustness at solving optimizing problems under different circumstances, its heavy computational cost may be prohibitive. Additionally, convergence towards optimum might be very slow and difficult in the presence of noisy data. Swarm intelligence based methods, such as PSO, tend and induce premature convergence and to reduce diversity within the swarm (Parrott & Li, 2006; Xinchao, 2010). As a result, new high performance heuristic algorithms are essentially required.

4.3 Gravitational Search Algorithm (GSA)

GSA was proposed by Rashedi et al. as a new population-based heuristic optimization approach (Rashedi et al., 2009) inspired by Newton laws of gravity and motion:

"Every particle in the universe attracts every other particle with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them".

4.3.1 Newton Laws of Gravity and Motion

Gravity is a natural phenomenon by which physical bodies attract with a force proportional to their mass. Gravitation is most familiar as the agent that gives weight to objects with mass and causes them to fall to the ground when dropped.

Gravity is responsible for retaining the earth and the other planets in their orbits around the sun and the moon in its orbit around the earth. The gravitational force acting on human bodies is perceived as our "weight".

Newton Laws of Gravity

According to the well-known story, Newton conceived his law of gravitation after a failing apple landed on his head. Immediately he realized that a force must have pulled the apple from the tree and towards the ground.

Newton's law states that every massive particle (M_1 and M_2) in the universe attracts every other massive particle with a force which is directly proportional to the product of their masses and inversely proportional to the square of the distance between them (R). The universal gravitational constant G is essentially a "fudge factor".

Mathematically, the law is formulated as follows (see Figure 4.2):

$$F = G\frac{M_1 M_2}{R^2} \tag{4.1}$$

where:

- F is the force between the masses;
- M_1 and M_2 are the mass of the first and second particles respectively;
- R is the distance between the centers of mass of the particles.

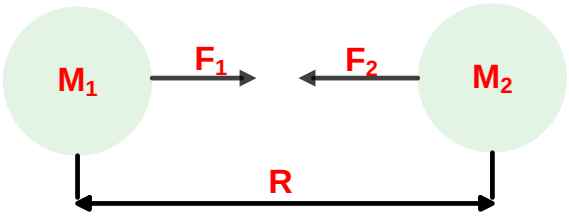


Figure 4.2: *Newton universal law of gravitation.*

Newton Laws of Motion

Newton developed his theories of gravitation in 1666, when he was only 23 years old. Around twenty years later, he published three laws of motion in his "*Principia Mathematica Philosophiae Naturalis*". The three laws of motion are summarized as follows:

1. **First law**: The velocity (v) of a body remains constant unless the body is acted upon by an external force.

   In its inertial form, this law sates that objects will remain in their state of motion unless their motion is altered by a force.

2. **Second law**: The acceleration a of a body is parallel and directly proportional to the net force F and inversely proportional to the mass m, i.e.

   $$F = ma \tag{4.2}$$

   Given an external applied force, the change in velocity depends on the mass of the object. A force induces a change in velocity; alternatively, a change in velocity will generate a force. The equation is invertible.

3. **Third law**: All forces in the universe occur as equal but oppositely directed pairs. No isolated forces exist; for every external force that acts on an object, the object exerts an opposite force of equal magnitude.

   The third law underlies the generation of lift by a wing and the production of thrust by a jet engine.

From Eq.(4.1) and Eq.(4.2), we note that an attracting gravitational force among all particles in the universe, whose magnitude increases with mass and decreases with distance. Due to the effect of decreasing gravity, the actual value of the gravitational constant (G) depends on the age of the universe. Eq.(4.3) gives the decrease of G with age (Mansouri, Nasseri & Khorrami, 1999):

$$G(t) = G(t_0)(\frac{t_0}{t})^{\beta},\ \beta < 1 \tag{4.3}$$

where $G(t)$ is the gravitational constant at time t and $G(t_0)$ is the gravitational constant at the first cosmic quantum time interval t_0.

4.3.2 Real-Valued Gravitational Search Algorithm (RGSA)

The original version of GSA (RGSA) was designed for optimizing the problems with real-valued parameters. The RGSA algorithm is provided in Algorithm 4. The principle of the algorithm is outlined below.

Given a system containing N agents (masses), the position of the i^{th} agent can be defined by:

$$X_i = (x_i^1, \ldots, x_i^d, \ldots x_i^n), \text{for } i = 1, 2, \ldots, N \tag{4.4}$$

where x_i^d presents the position of the i^{th} agent in the d^{th} dimension.

Following calculation of the current population's fitness, the mass of each agent is determined as:

$$M_i(t) = \frac{fit_i(t) - worst(t)}{\sum_{j=1}^{N}(fit_j(t) - worst(t))},\ worst(t) = min_{j\in(1\ldots N)} fit_j(t) \tag{4.5}$$

where $M_i(t)$ is the mass of agent i, $fit_i(t)$ is the fitness value of agent i at time t, and $worst(t)$ represents a maximization problem.

Based on the law of gravity, the force acting on mass i from mass j is calculated as (Eq.4.6):

$$F_i^d(t) = \sum_{j\in k_{best}, j\neq i} rand_j G(t) \frac{M_i(t)M_j(t)}{R_{ij}(t) + \varepsilon}(x_j^d(t) - x_i^d(t)) \tag{4.6}$$

The acceleration of agent i at the time moment t in the d^{th} direction is calculated

using Newton's second law as (Eq.4.7):

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)} = \sum_{j \in k_{best}, j \neq i} rand_j G(t) \frac{M_j(t)}{R_{ij}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) \tag{4.7}$$

The next velocity of an agent, calculated as a fraction of its current velocity, is added to its acceleration (Eq.4.8). Thus, the position of agent i at time moment $(t+1)$ is expressed as (Eq.4.9):

$$V_i^d(t+1) = rand_i v_i^d(t) + a_i^d(t) \tag{4.8}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{4.9}$$

where $rand_i$ and $rand_j$ is a random number within the interval [0,1], respectively; ε is a small constant; $R_{ij}(t)$ is the Euclidean distance between agent i and j; k_{best} is the set of K agents with the best fitness value and highest mass.

G is a gravitational constant is assigned an initial value G_0 and will decrease towards 0 after many iterations (Eq.4.10):

$$G = G_0 exp(-\frac{\alpha * t}{T}) \tag{4.10}$$

where T represents the total number of iterations.

Algorithm 4 Real-valued gravitational search algorithm (RGSA)

1: Generate initial population N
2: **Repeat**
3: **for** Every agent/mass $i = 1,2,\ldots,N$ **do**
4: Evaluate the fitness for each agent;
5: Calculate mass M for each agent (Eq.(4.5));
6: Calculate acceleration a for each agent (Eq.(4.7));
7: Update the velocity V (Eq.(4.8));
8: Update the position (Eq.(4.9));
9: **end for**
10: **Stop until termination conditions are met**

4.3.3 Binary-Valued Gravitational Search Algorithm (BGSA)

BGSA is a modification of the original GSA (Rashedi et al., 2010). Although both RGSA and BGSA similarly update the force, acceleration and velocity (Eq.4.6 - 4.8), the two essential differences exist between the methods:

1. Distance measurement between agents: RGSA uses Euclidian distance while BGSA uses Hamming distance;
2. Update of agent position (according to the velocity of its mass): Both algorithms use Eq.4.8 to update the position, but BGSA assigns the new position as either "0" or "1" (using Eq.4.11 to transform V_i^d into a probability), i.e. "1" represents the feature to be selected, whereas "0" represents the feature not to be selected.

$$S(V_i^d(t)) = |tanh(V_i^d(t))| \tag{4.11}$$

4.3.4 Applications of GSA

GSA has been applied to many complex real-world problems, and has proven a flexible and well-balanced mechanism by which to enhance exploration and exploitation abilities. Some examples of GSA use are:

- Slope stability analysis (Khajehzadeh, Taha, El Shafie & M., 2011);
- The DNA sequence design problem (Xiao & Cheng, 2011);
- Combined with neural network for solving the well-known Wessinger's equation problem (Ghalambaz et al., 2011);
- Parameter identification of hydraulic turbine governing system (HTGS) (C. Li & Zhou, 2011);
- Combined with heuristic search (HS) for clustering problems (Hatamlou, Abdullah & Othman, 2011);

- The optimization of retaining structures (Khajehzadeh & M., 2012);
- A global searcher to find the best positions of representatives (prototypes) (Bahrololoum, Nezamabadi pour, Bahrololoum & Saeed, 2012).

4.3.5 Advantages of GSA

GSA belongs to the class of swarm population-based heuristic algorithms. Rashedi et al. (2009) conducted a comparative study between GSA and a number of well-known algorithms, including genetic algorithm (GA), swarm theory inspired particle swarm optimization (PSO), and metaphor of gravitational kinematics inspired central force optimization (CFO). The results showed that algorithms inspired by Newton law of gravity and motion outperform other algorithms at rapidly finding the global optimum, suggesting that GSA are suitable for complex problems. GSA is efficient for the following reasons (Rashedi et al., 2009, 2010):

- It is memory-free, but works as efficiently as the memory-based algorithms;
- Similar to PSO, an agent can easily observe the performance of its neighboring agents, because it detects the gravitational force of its neighborhood agents. In other words, the force can be regarded as an information-transferring tool between the agents;
- Because the inertial mass decelerates the motion, heavier agents move more slowly than their lighter-weight counterparts. Hence, new agents are searched within a local space, which constitutes adaptive learning;
- A heavy gravitational mass is associated with a large effective attraction radius and high attraction. Superior-performing agents possess greater gravitational mass, which attracts other agents toward the optimal agent.

4.4 Summary

This chapter reviews the evolutionary algorithms for optimization, highlighting their applications and advantages. Various classical optimization methods are also introduced along with their theoretical backgrounds and limitations. The literature

reveals that all of these methods are computationally expensive and that convergence towards the global optimum is quite slow in highly complex real-world situations. Thus, in this study we adopt a recently proposed Newton law of gravity and motion inspired algorithm called GSA. Using this algorithm we develop a novel integrated evolving personalized modeling system (evoPM) for optimizing features, neighborhood and model parameters. Our choice was influenced by the following considerations: (1) GSA has not previously been integrated with a personalized modeling approach for complex optimization problems; (2) GSA is applicable to both continuous (real-valued) and discrete (binary-valued) multi-object optimization; (3) The construction of personalized models typically carries a heavy computational burden, because it creates a personalized model for each testing sample, thus requiring intensive optimization to find an optimal solution. Existing studies show that GSA can converge to the global optimum much faster than many classical optimizers (Rashedi et al., 2009; Sarafrazi, Nezamabadi-pour & Brahman, 2010).

To evaluate the feasibility of using the novel integrated evolving personalized method (evoPM), we test the model on stroke data as case studies. The next chapter introduces stroke and describes two large stroke datasets: the largest and most accurate spatio-temporal stroke dataset collected from stroke occurrences worldwide, and a long-term population-based stroke outcome dataset.

CHAPTER 5

The Case Study of Stroke Data

5.1 Introduction

Stroke is a major cause of disability and mortality in most economically developed countries. It is the second leading cause of death worldwide (after cancer and heart disease) (Johnston, Mendis & Mathers, 2009; Rothwell, 2001) and a major cause of adult disability in developed countries (Tobias, Cheung & McNaughton, 2002). Due to its prevalence and severity stroke has become a major public health challenge and concern in New Zealand and globally. Tobias et al. (2007) estimated that over 7,000 New Zealanders each year will experience a stroke event, and at least three-quarters of this population will die or be dependent on others for health care one year later.

Following a brief introduction to stroke, this chapter reviews various information methods, including conventional statistical methods and computational intelligent modeling methods for predicting stroke risk and outcome.

This chapter also introduces a population-based long-term Auckland Stroke Outcomes Study (ASTRO). Understanding long-term stroke outcomes, including body functioning (neurologic and neuropsychological impairments), activity limitations and participation, is essential for long-term evidence-based rehabilitation and service planning that could significantly improve health outcomes. However, most existing

neuropsychological stroke data are not population-based, examine limited outcomes, and are limited to short-term follow-up.

This chapter provides a pilot statistical analysis over entire population of the ASTRO dataset. In addition, the performance of evoPM-based algorithms are compared with that conventional global, local, and classical personalized modeling methods on the ASTRO dataset as a case study. The principle aim is to find the predictors of stroke outcomes in 5-year stroke survivors. Studying predictors of long-term outcomes in stroke survivors would allow the identification of patients who may benefit from specific rehabilitation services, may improve planning of stroke care and rehabilitation services and would facilitate information provision to patients and their families. Such measures would enhance the patient's potential for recovery and the likelihood of surviving in the long-term.

5.2 Biological Background of Human Brain

As part of the central nervous system, the human brain is responsible for receiving, analyzing, and storing information (forming memories). The average human brain weights about 3 pounds (1300-1400g), approximately 2% of our body weight. Human brain comprises three major parts: cerebrum, cerebellum, and brainstem (medulla), which are briefly described below (see Figure 5.1):

- **Cerebrum**

  This is the largest part of the brain, comprising 85% of the brain weight. The cerebrum facilitates complex behaviors such as thought, judgement, learning, working memory, speech and language, and social interactions.

- **Cerebellum**

  The cerebellum lies back of the brain, below the cerebrum. It is a mere 1/8 the size of the cerebrum, but controls of several bodily functions, such as muscle tone, balance and equilibrium, and fine movement coordination.

- **Brainstem**

  The brainstem sits beneath the cerebrum and in front of the cerebellum. It connects the rest of the brain to the spinal cord, which descends down the neck

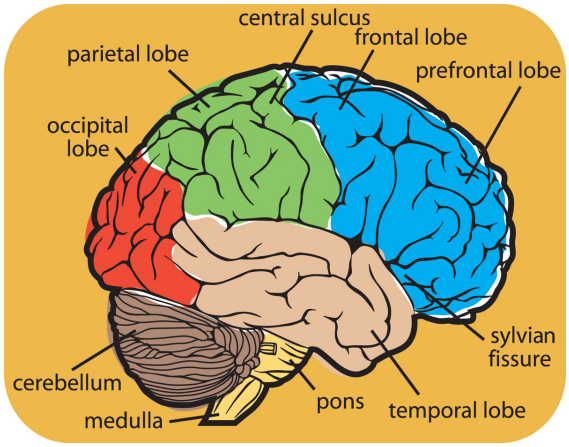


Figure 5.1: *A simplified diagram of the human brain (Michelon, 2008).*

and back, and is involved in the functions required to sustain life, including breathing, food digestion, and blood circulation.

The brain is a complex system that evolves its functions and structures during its lifetime (Kasabov, 2007a). Brain performance is governed by complex interactions between genes and neuronal functions. Abnormalities in these interactions may cause certain brain diseases, such as brain cancer, Parkinsons disease and Alzheimers disease.

5.3 Review of Stroke

5.3.1 What is a Stroke

The World Health Organization (WHO) defines stroke as "rapidly developing clinical signs of focal or global disturbance of cerebral function lasting more than 24 hours (unless interrupted by surgery or death) with no apparent cause other than of vascular origin" (Aho et al., 1980). It is generally accepted that the lifetime risk of stroke occurrence is 1 in 6, at least as high as the risk for developing Alzheimer disease (Seshadri, 2006).

Stroke exerts a large physical, psychological and financial impact on patients/families, the health care system, and society (Strong, Mathers & Bonita, 2007; Caro, Huy-

brechts & Duchesne, 2000). Lifetime costs per stroke patient range from US 59.8K to US 230K (Caro et al., 2000). The majority (about 75%) of stroke cases occur in the age group 65 years and over (Bonita, Broad & Beaglehole, 1993; Bonita et al., 1994), and about one third of patients die within a year of stroke onset (C. S. Anderson, Jamrozik, Broadhurst & Stewart, 1994; Bonita, Ford & Stewart, 1988). Over half of the survivors remain dependent on others for everyday activities, often with significant adverse effects on caregivers (C. S. Anderson, Linto & Stewart, 1995).

Family members of stroke victims are burdened by the suffering of their loved ones as well as by the responsibility caring for them, uncertainty regarding future plans and financial anxiety regarding the cost of the patient's treatment.

5.3.2 What are the Risk Factors

Stroke risk is elevated by a number of factors. More risk factors incur a greater chance of suffering a stroke. Stroke risk factors are broadly categorized into two classes (Wannamethee, Shaper, Whincup & Walker, 1995; Hankey, 1999; Reynolds, Lewis & Nolen, 2003; Thomson, 2009; Larsson, Virtamo & Wolk, 2011):

1. **Controllable**

   Controllable risk factors include: *lifestyle* risk and *medical* risk factors. Lifestyle risk factors can often be changed, while medical risk factors are usually treatable. Both types can be best managed by working with a doctor, who can prescribe medications and advise on adopting a healthy lifestyle.

   The most important risk factors are:

   - Smoking
   - Alcohol
   - High Cholesterol
   - Diabetes mellitus
   - Elevated blood pressure
   - Overweight (especially abdominal obesity)
   - Poor, unbalanced diet lacking fruits and vegetables

2. **Uncontrollable**

   Uncontrollable risk factors include increasing age (being 55 or older), gender (males are at greater risk than females), ethnicity (Asians/Pacific Islanders and African American are at increased risk), and family history of stroke, heart attack or transient ischemic attack (TIA).

5.3.3 What are the Symptoms

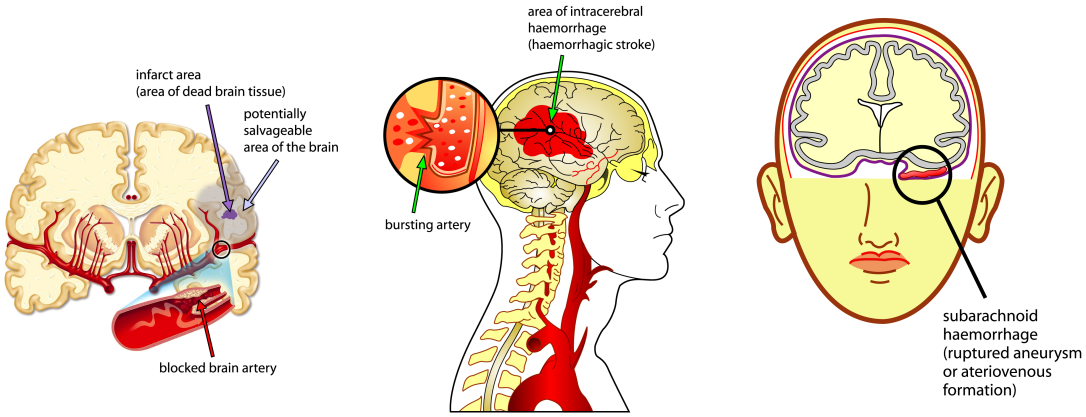
The signs and symptoms of a stroke depend on the area of the brain affected and the amount of brain tissue damaged. Although isolated small strokes may induce no significant focal neurological symptoms (so-called silent strokes), accumulated small strokes may lead to clinically significant consequences, such as vascular dementia. In general, strokes affecting the left side of the brain produce clinical symptoms on the right side of the body and vice versa.

According to the U.S. National Institute of Neurological Disorders and Stroke (NINDS), the common symptoms of stroke are typically sudden and may include:

- Loss of consciousness: patient may become stuporous or hard to arouse;
- Loss or disturbance of vision: difficulty with seeing in one or both eyes, such as blurriness;
- Headache: sudden onset of severe headache that may be accompanied by vomiting or dizziness (loss of balance);
- Trouble with muscle movements: difficulty with walking, moving arm or leg on one side of body, carrying or picking up objects;

5.3.4 How does Stroke Happen

Stroke is a heterogeneous disorder encompassing two major pathological types (ischemic and hemorrhagic). Each type is divisible into different sub-types with different causes and outcomes. Ischemic and hemorrhagic stoke are discussed in more detail below (see Figure 5.2):



(a) Ischemic stroke occurs when a blood vessel in the brain becomes blocked

(b) Intracerebral hemorrhage occurs when blood vessels within the brain become damaged and burst within the brain

(c) Subarachnoid hemorrhage occurs when a cerebral aneurysm ruptures, blood will fill the space surrounding the brain

Figure 5.2: *Types of Stroke (V. Feigin, 2004).*

Ischemic Stroke

Ischemic stroke is the most common type of stroke, accounting for almost 85% of all stroke cases. It results from a clot in the blood vessel of the brain that reduces or blocks the blood supply coming from the heart to the brain. Since the brain does not store nutrient/energy, it requires a constant supply of nutrients from the blood. The blood carries sugar and oxygen to the brain, and removes cellular waste and carbon dioxide. If an artery is blocked, the brain cells are deprived of essential oxygen and glucose, and the affected cells begin to shut down. If blood supply is absent for as little as 7 seconds, the affected brain cells may die.

At least four subtypes of ischemic stroke have been identified: cardioembolic stroke, ischemic stroke due to large artery disease (such as atherosclerosis), ischemic stroke due to small artery disease (such as hypertension, intracranial arteritis), and ischemic stroke due to haematological disorders and other rare conditions.

Hemorrhagic Stroke

Hemorrhagic stroke accounts for up to 15% of all stroke cases. It is a frequent complication of bleeding into brain from a burst artery (intracerebral hemorrhage) or bleeding around the brain (subarachnoid hemorrhage).

Intracerebral Hemorrhage

Intracerebral hemorrhage occurs when a diseased blood vessel breaks in the brain, causing blood leakage in the brain tissue. The resulting sudden increase in intracranial pressure may directly or indirectly damage the affected brain cells. Both precesses may cause unconsciousness, lost neurological function or even death. Intracerebral haemorrhage may be caused by different mechanisms (e.g. elevated blood pressure, amyloid angiopathy) in different parts of the brain (e.g. supratential, infratentirial hemorrhage) each of which carries a different prognosis and requires different management strategies.

Subarachnoid hemorrhage

Subarachnoid hemorrhage occurs when a blood vessel bursts in the area between the brain and the thin tissues surrounding the brain. This area, termed the subarachnoid space, lies outside of the brain tissue. Subarachnoid hemorrhage is characterized by loss of consciousness, vomiting, severe headache or neck pain, and neck stiffiness. Subarachnoid hemorrhage most often results from a rupture of the intracranial aneurysm but may also arise from a rupture of other brain arteries (so-called non-aneurysmal subarachnoid hemorrhage). The two forms of subarachnoid hemorrhage require very different management.

5.4 Information Methods for Predicting Risk and Outcome of Stroke

Many intelligent systems have been developed with the purpose of improving health care and providing better health care facilities at reduced cost. These technologies can be divided into two major categories: conventional statistical methods and computational intelligent machine learning methods. However, stroke occurrence and

outcomes literature reveals that traditional predictive models using standard population statistics apply to collective of patients and cannot predict the level of risk occurrence or disability for either an at-risk individual or a stroke survivor. Conventional statistical prediction methods employ only the most significant predictive variables, less statistically significant personal information that may be clinically significant for the particular patient is certainly lost (Wieslaw et al., 1997).

For this reason, effective computational intelligent machine learning approaches should be integrated into the medical system for diagnosis, prediction and management. Personalized modeling has already been adopted for knowledge discovery in biomedical applications. This computational intelligent approach aims to create a personalized diagnostic or prediction model for an individual person based on his/her nearest neighbors of predictive variables that are pertinent for that person.

5.4.1 Conventional Statistical Methods

Stroke prediction is usually analyzed by conventional statistical methods. For example, the frequency of strokes in the general population, across gender, and ethnic groups is estimated from descriptive statistics such as frequency statistics (V. L. Feigin et al., 2006); correlations between two different scales, such as Barthel Index and the SF-36 are found by spearman rank or some other correlation methods (Lai, Duncan & Keighley, 1998); the factors associated with the SF-36 sub-scales are analyzed by logistic regression to determine which of these variables best discriminates between patients scoring low and high on the SF-36 subscales (Kauhanen, 1999); and differences between stroke outcomes are generally analyzed by one-way of variance and $X2$ (Chi square) test (V. L. Feigin et al., 2010).

Compared to machine learning methods, conventional statistical methods are limited in efficiency and prediction accuracy. Khosla and his colleagues (Khosla et al., 2010) developed an integrated machine learning approach and compared its performance with that of the Cox proportional hazards model (one of the most commonly used conventional statistical methods in medical research) on the Cardiovascular Health Study (CHS) dataset for stroke risk prediction. They demonstrated that the machine learning methods significantly outperformed the Cox model in terms of stroke risk estimation.

5.4.2 Machine Learning Methods

"Machine learning is the process of discovering and interpreting meaningful information, such as new correlations, patterns and trends by sifting through large amounts of data stored in repositories, using pattern recognition technologies as well as statistical and mathematical techniques" (Larose, 2005). In other worlds, machine learning uses different analysis techniques to observe previously unknown, potentially meaningful information, and to discover strong patterns and relationships from a large dataset that can be accurately applied to a particular patient.

While vast volumes of biomedical data on stroke risk factors and prognosis are available, the interpretation of these data remains complex and challenging. These complex data have become increasingly explored, but mostly by conventional statistical methods. The need for computational models, especially with regard to personalized risk assessment is steadily growing. Such models will assist in unveiling the pathophysiology of individual and specific groups of stroke, and to achieve improved and reliable risk prediction for individuals.

Personalized modeling is an emerging effective approach for knowledge discovery in biomedical applications. As mentioned above, personalized modeling generally outperforms the conventional statistical methods at prediction and/or classification of conditions. The most popular personalized modeling methods are the nearest neighbor method and its derivatives (Vapnik, 1998; Kasabov, 2007c), which create a simple model for every individual entity based on 'neighboring' data points. However, published studies highlight the need for new methods that deliver more efficient personalized outputs.

As described in Chapter 3, SNN is emerging as a powerful computational machining learning tool that can successfully diagnose and monitor the prognosis of a disease. As such, it can be applied to stroke rehabilitation (Wieslaw et al., 1997), early diagnosing of ischemic stroke (Anita, Bhanot & Mishra, 2009), and to classify the gait patterns of post-stroke patients into homogenous groups (Kaczmarczyk, Wit, Krawczyk & Zaborski, 2009).

As proposed by Kasabov (2011), the development of a novel integrated evolving personalized modeling system using novel technology such as eSNN might facilitate more precise decision making, ensuring that patients receive optimal prognosis and

treatment. To this end, we aspire to develop a novel integrated eSNN-based evolving personalized classification method, and to evaluate its feasibility for medical decision support.

Most of the available stroke data are spatio-temporal data (STD), which are difficult to process. In this thesis, a recently proposed extended eSNN (EESNN) model (Hamed et al., 2011) and a recurrent network reservoir structure of eSNN (reSNN) using liquid state machine (LSM) (Schliebs et al., 2011) are tested in a case study involving spatio-temporal weather and stroke occurrence data. The results will provide new insights into the relationship between weather and stroke occurrence. We hypothesize that the EESNN and reSNN individualized prognostic models are applicable to multivariate STD and will outperform existing prognostic models, providing both better accuracy of individualized event prediction and new knowledge.

5.5 Stroke Outcome Data

5.5.1 Background

Auckland Stroke Outcomes Study (ASTRO) is a population-based long-term stroke follow-up study exploring the associations between neuropsychological deficits (memory, executive function, information processing speed, visuoperceptual/construction ability, language), depression, and a range of functional outcomes and their interrelationships 5 years post-stroke. The study sources its participants from the population-based Auckland Regional Community Stroke (ARCOS) study conducted in 2002-2003.

5.5.2 Dataset Description

418 patients participated in the ASTRO study, 318 Europeans, 37 Pacific Islanders, 35 Asians, 23 Maori, and individuals from 5 other ethnic groups. All stroke outcomes are measured by structured self-administered questionnaires and a face-to-face interview including a battery of neuropsychological tests. Among the questionnaires administered were the Short Form 36 questionnaire (SF-36), Geriatric Depression

Scale (GDS-15), Modified Rankin Score (MRS), Barthel Index (BI), Frenchay Activity Index (FAI), Hodkinson Abbreviated Mental Test (HAMT), Bakas Caregiving Outcomes Scale (BCOS), and General Health Questionnaire 28 (GHQ-28).

Since a complete study is impractical, a pilot study was conducted to find the predictors of depression in 5-year stroke survivors using the short form GDS-15. The short form GDS-15, introduced by Sheikh and Yesavage in 1986, has been extensively applied to older populations in community, acute and long-term care settings. The short form includes 15 yes/no questions (see Appendix D). The scale of the scores is 0-15, where a score above "5" suggests depression, and a score exceeding "10" almost always indicates depression.

In this study, 408 patients completed this questionnaire, scores of < 5 and $\geqslant 5$ were assigned into class 1 (287 patients) and class 2 (121 patients), respectively.

5.5.3 Statistical Analysis

The 408 patients participating the questionnaire comprise 213 male and 195 female patients. Patient age range from 20-99, with most patients in the 70-79 age group (see Figure 5.3).

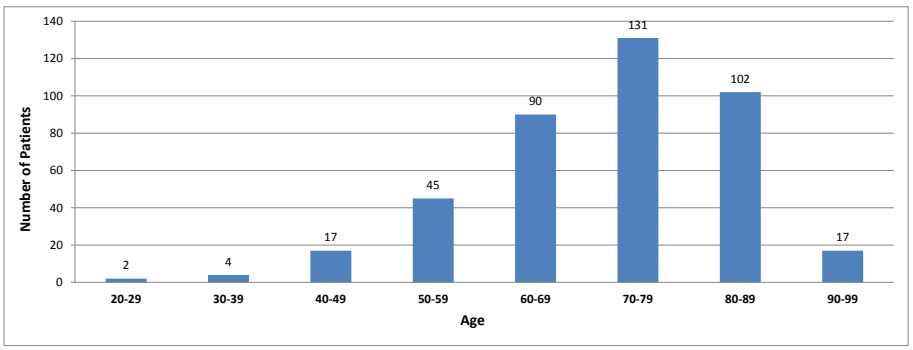


Figure 5.3: *Number of patients in each age group.*

The distribution of GDS score across the entire population is shown in Figure 5.4. 70% of patients present as non-depressible. The most common score among these patients is 2.

We next separate number of patients indicative of depression and non-depression by

Table 5.1: *Classification accuracy obtained by conventional global, local, personalized modeling approaches, and evoPM-based algorithms through LOOCV validation.*

Experimental Results		
Classifier	Overall Acc(%)	Class(1/2)(%)
SVM (RBF Kernal, gamma=0.5)	84.60	(86.80/82.40)
ECF	84.04	(85.54/82.54)
KNN(k=15)	84.93	(88.53/81.33)
WKNN(k=13, thre=0.3)	84.61	(86.64/82.58)
knnGSA(k=11 Ave)	89.63	(89.81/88.25)
svmGSA(gamma=0.74, c=50.96, k=17 Ave)	**91.91**	(94.60/89.34)
esnnGSA(mod=0.65, thre=0.34, sim=0.22, k=21 Ave)	89.22	(89.71/88.73)

Note: The parameters are selected through the same optimization process if they are employed in evoPM models. The parameters in SVM, ECF and KNN are selected based on the best classification performance. For the global SVM parameters, only the parameter γ is tuned.

gender (see Figure 5.5). Clearly, male patients are more likely to display signs of depression than female patients.

We also investigate the number of patients indicative of depression and non-depression in each age group (see Figure 5.6). The age groups 70-79 and 80-89 contain the same largest number of patients indicative of depression (32 patients score of $\geqslant 5$ in both age groups), followed by the 60-69 age group (29 patients report a score of $\geqslant 5$). From this study, we infer that the older population is more subject to depression than the young population.

5.5.4 Experimental Setup

To implement a performance comparison of the different methods, I have applied a *global modeling* method (SVM); a *local modeling* method (ECF); two classical *personalized modeling* methods (KNN and WKNN); and *evoPM*-based methods (knnGSA, svmGSA and esnnGSA). The performance of all experiments is evaluated by LOOCV. Significantly, irrelevant features are filtered out using a signal-to-noise-ratio (SNR) algorithm.

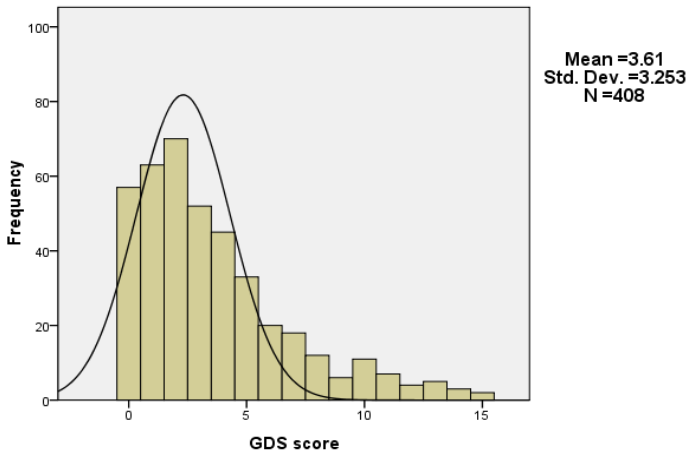


Figure 5.4: *Distribution of Geriatric Depression Scale (GDS) score over the entire population in the stroke dataset.*

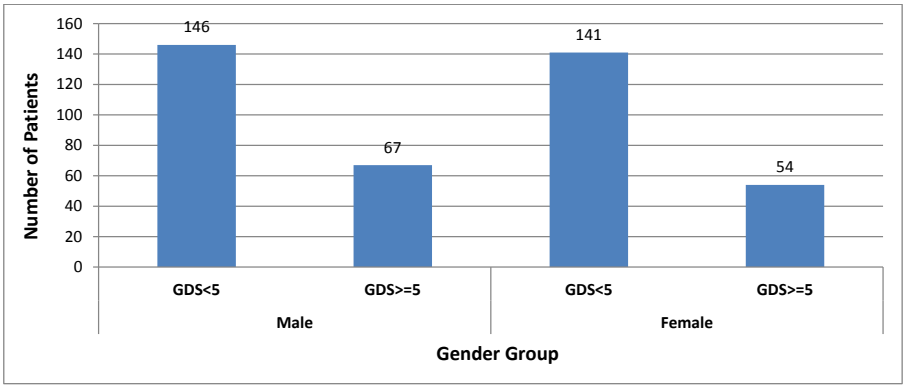


Figure 5.5: *Number of patients indicative of depression and non-depression in each gender group.*

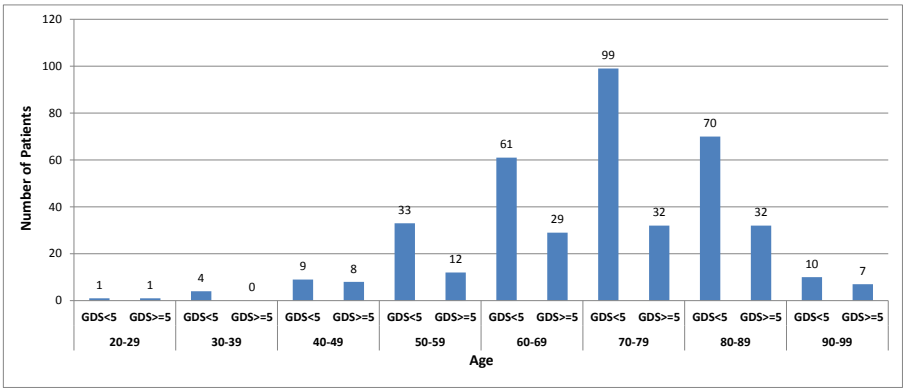


Figure 5.6: *Number of patients indicative of depression and non-depression in each age group.*

5.5.5 Experimental Result

Table 5.1 summarizes the results achieved by all methods, valuated by LOOCV. The performance is significantly improved under the evoPM-based methods, svmGSA yielding the best classification performance (91.91%), exceeding the best accuracy achieved by the conventional WKNN method by almost 7%.

Recall that evoPM creates a personalized profile for each individual patient. To demonstrate this efficacy, a personalized profile is created for sample 10 of the stroke data using svmGSA (see Figure 5.8). 6 features are selected as the best predictors of depression for this particular patient.

The global predictors are computed based on the selecting frequency over all samples obtained by svmGSA (see Figure 5.7). The features "5, 6, 8, and 12", selected as global predictors of depression, are presented below:

- **Feature 5** - Are you in good spirits most of the time?
- **Feature 6** - Are you afraid that something bad is going to happen to you?
- **Feature 8** - Do you often feel helpless?
- **Feature 12** - Do you feel pretty worthless the way you are now?

In conclusion, this experiment presents approximately 30% of patients are indicative of depression. The older population (age 60-89) are at increased risk of depression. Furthermore, according to the comparative study using various computational algorithms, the evoPM-based methods achieve higher classification accuracies than the conventional methods because they can select the optimal or near-optimal sets of features, nearest neighbors, and model parameters. Provided that a patient can remain happy and the risk of depression retain a sense of usefulness is largely reduced!

A single stroke outcome has been investigated in this chapter. Future studies must investigate more stroke outcomes and evaluate the correlations between various outcomes. As more data become available, the problem will become one of stroke risk prediction rather than classification.

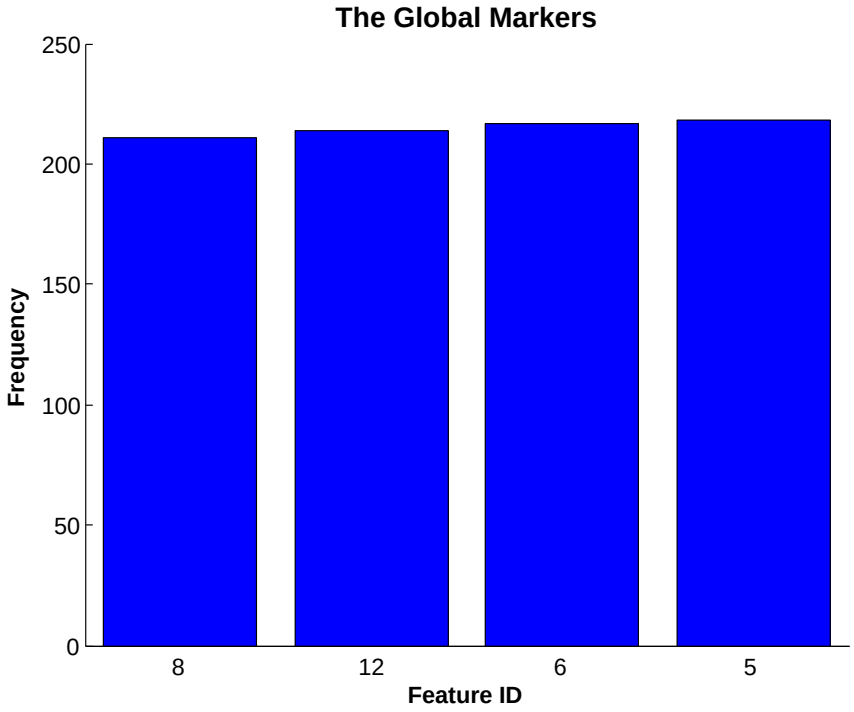


Figure 5.7: *A set of global markers are selected across all samples obtained by svmGSA.*

```
================================================================================ Result ================================================================================
Sample ID        Actual Class        Predicted Class        Predicted Class (based on Probability)        Probability in Class1        Probability in Class2
       10                   2                      2                                             2                      0.20238                      0.79762

--------------------------------------------------------

 Best Gamma          Best C
   0.18              244.94

--------------------------------------------------------

 # K      KNN Index
   7       9 28 52 229 231 73 86

--------------------------------------------------------

# Features      Feature Index
   6               2 4 3 13 7 5

================================================================================ End ================================================================================
```

Figure 5.8: *The personal profile for sample 10 obtained by svmGSA, after 50 testing runs.*

5.6 Summary

This chapter briefly reviews the medical condition of stroke, and identifies the risk factors, symptoms and aetiology of stroke. Several information methods for predicting risk and outcome of stroke are introduced, embracing both conventional statistical methods and machine learning methods. As the demand for suitable methods to extract essential information from complex stroke data, increases conventional statistical methods have been refined and supplemented with new computational approaches. Particularly, personalized modeling is regarded as ideal approach to individually tailored medical decision making. To this end, we propose and develop a novel personalized modeling system and framework, termed evolving personalized modeling system (evoPM), as discussed in the next chapter.

CHAPTER 6

Novel Integrated Evolving Personalized Modeling System (evoPM) for Feature Selection, Neighborhood and Parameter Optimization

6.1 Introduction

This chapter introduces a novel evolving personalized modeling system incorporating a gravitational search (GSA) inspired algorithm for selecting a small group of most informative features, optimizing neighborhoods and model parameters relevant to the learning functions. The system should exhibit superior diagnostic and prognostic performance and personalized knowledge relative to global and local modeling approaches. In addition, the obtained information and knowledge may significantly contribute to the design of individualized treatments, e.g. personalized medicine and personalized drug design.

This chapter states the motivation behind the development of this novel personalized modeling system and framework. Thereafter, the system itself termed *evolving personalized modeling system (evoPM)* is introduced, ranging from the simple im-

plementation (with limited model parameters optimization) to more comprehensive implementation (with full feature, neighborhood and model parameters optimization). Finally, the strength of each evoPM prototype is evaluated in an experimental study.

6.2 Motivation

As previously discussed, transductive approaches have been successfully implemented in medical and clinical decision support systems, and time-series prediction problems, where a personalized model is created for each new input vector. The model aims to predict the best outcome for the individual data vector. Many studies have shown that such characteristic is able to ensure personalized modeling to be a more appropriate method for solving complex problems rather than using the methods based on conventional global modeling approaches (Kasabov, 2007c; Ramaswamy & Perou, 2003; Kasabov et al., 2008; Hu, Song & Kasabov, 2009).

However, there are some opening questions raised in the development of the personalized modeling framework (Kasabov, 2007c), such as:

- *What features are significant for every new input vector?*
- *How many nearest neighbors should be selected for every new input vector?*
- *How to find the best combination of model parameters for the learning function (e.g a classifier)?*

Theoretically speaking, the performance of a personalized model largely relies on some specific parameters that might have different optimal values for every new input vector, such as the number of selected features, number of nearest neighbors and optimal sets of model parameters. Thus, it is essential to optimize these parameters in order to effectively improve the performance of a personalized model, as well as correctly derive personalized knowledge. For this reason, a novel system and framework for personalized modeling is developed to study and address these opening questions based on the existing personalized modeling framework introduced by Kasabov (2007c).

6.3 Methodology

In this study, a novel integrated evolving system for personalized modeling is proposed as an extension of Kasabov and Hu's model (Kasabov & Hu, 2011). The system aims to evolve a personalized model for every single new input vector based on its nearest neighbors. The concept is illustrated in Fig. 6.1 for a new patient V. At time t, a personalized model is constructed for the patient. Later on, another personalized model may be created for the same patient at time $t+1$, reflecting the changing status of the patient, for example his/her age or level of hypertension, etc.

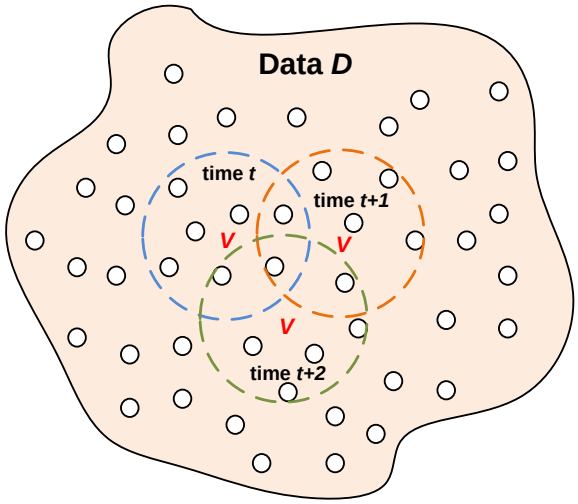


Figure 6.1: *The basic concept of the proposed novel integrated evolving personalized modeling system (evoPM).*

Alternatively, the system keeps evolving and is ready to accept any new incoming data vectors. The already created personalized model can be further evolved on new data entering in the neighborhood. The evolving process will include:

1. Comparison of the new data vectors with the individual vector for which a model is being developed;
2. If the new data vectors belong to the neighborhood, the personalized model is updated;
3. A new outcome is calculated for the individual and new profile is extracted.

6.3.1 The Principle of evoPM System and Framework

The novel proposed evoPM system is a hybrid approach consisting of six main processes, as summarized below:

1. *Pre-filtering feature subset* - A subset of relevant features Fea_i are selected for the new input vector V_i from a global space using signal to noise ratio (SNR);
2. *Selecting K-nearest neighbors* - A set of nearest neighbors of V_i are selected, and gathered into a local problem space D_i;
3. *Evaluating fitness function* - Each agent/chromosome comprises three parts: feature mask (Fea), k-nearest neighbors (K), and model parameters (M_p). The classification accuracy of each chromosome is evaluated by its fitness function;
4. *Meeting termination criteria* - If the termination criteria are met, the entire process is stopped; otherwise it continues processing the next generation of agents;
5. *Building personal profile* - A personalized model and personalized profile are built using the optimal sets of features, neighborhoods, and model parameters with known outcomes;
6. *Validation* - All performances obtained by evoPM in this study are validated by leave-one-out cross validation (LOOCV).

Figure 6.2 illustrates the flowchart of evoPM system, and the pseudo code of the novel system is given in Algorithm 5.

6.3.2 Chromosome Structure

As introduced above, the novel evoPM system can simultaneously select optimal or near-optimal sets of features, neighborhoods and model parameters. Therefore, the whole optimization problem space can be decomposed into three sub-components:

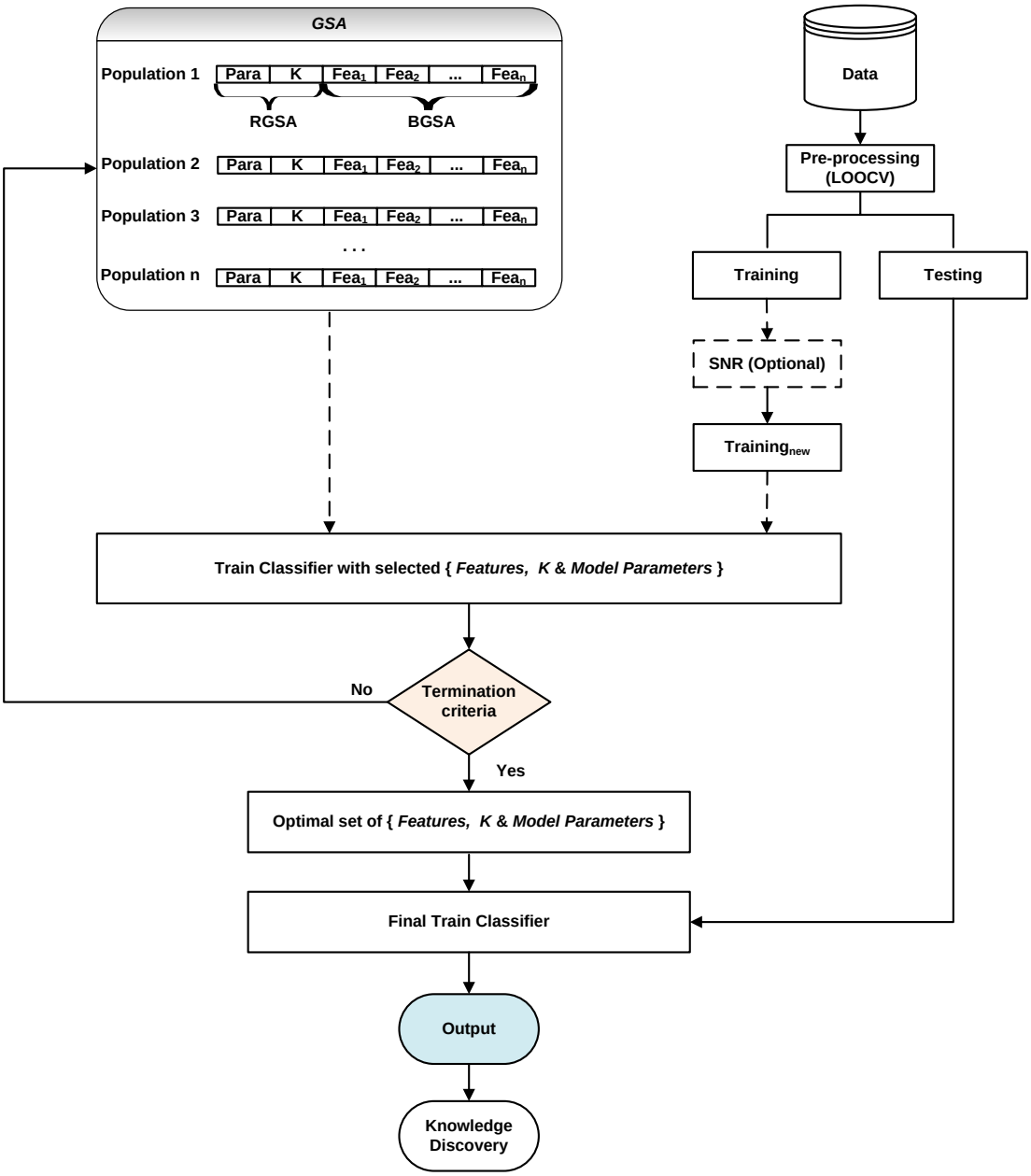


Figure 6.2: *Flowchart of the proposed novel integrated evolving personalized modeling system (evoPM)*

Algorithm 5 Evolving Personalized Modeling System (evoPM)

Require: Input a new data vector V_i and a training dataset D

1: **Pre-filtering feature subset**
$Fea_i = f_{rnk}(D)$
2: Generate a candidate feature pool Fea_p from the feature subset Fea_i
3: **Selecting K-nearest neighbors for** V_i
$D_i = f_k(V_i, D)$
4: **Evaluating fitness function**
$Opt_{sel} = f_{sel}(Fea_p, D_i, M_p)$
5: **if** termination criteria are met **then**
6: Output Opt_{sel} with best feature mask (Fea), k-nearest neighbors (K), and model parameters (M_p)
7: **else**
8: Return to *Step 4* to process the next generation
9: **end if**
10: **Building personal profile on the testing data vector** V_i
$output_i = f_{cls}(Opt_{sel}, V_i)$

11: ***where:***
f_{rnk}: a statistical function (e.g. SNR) for ranking all features;
f_k: a function (e.g. KNN) for searching the personalized space for V_i;
f_{sel}: a function for selecting optimal or near-optimal sets of Fea, K, and M_p;
f_{cls}: a classification function.

1. *Component 1 - Feature mask (Fea).* The features are encoded into a binary bit string, in which each bit denotes wether this feature is to be selected (1) or not (0).

2. *Component 2 - Neighborhood (K).* This component is used for finding the number of samples in the personalized problem space, and is real-value encoded.

3. *Component 3 - Model parameters (M_p).* This subcomponent is used for optimizing model parameters and is real-value encoded.

In this thesis, two types of GSA are adopted for feature, neighborhood selection and parameter optimization: the continuous (real-valued) RGSA and discrete (binary-valued) BGSA. RGSA is utilized to optimize the neighborhoods and model parameters, whereas BGSA is utilized to select the features (see Figure 6.3).

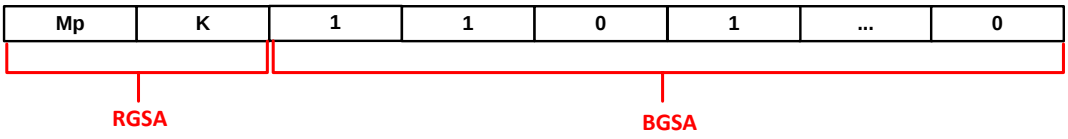


Figure 6.3: *A chromosome consists of three sub-components; feature mask (Fea), neighborhoods (K), and model parameters (M_p)*

6.3.3 Fitness Function

Each of the chromosomes in a generation must be evaluated based on their fitness function. A fitness function determines how well each chromosome solves the problem. In general, evaluation is accomplished by examining the classification accuracy of each chromosome containing an optimal or near-optimal number of features (Fea), neighborhoods (K) and model parameters (M_p). A chromosome with a high fitness value will very likely be selected in the next generation.

In the novel evoPM system, the fitness function is evaluated by the KNN and SVM classification algorithms, chosen for their simplicity and effectiveness. Because of the proven efficacy of eSNN at solving complex classification tasks, this technology is here applied to the novel integrated evolving personalized modeling system. The pseudo code of the GSA based hybrid personalized system is given in Algorithm 6.

Algorithm 6 GSA based Hybrid System

1: Generate initial population N
2: **Repeat**
3: **for** Every agent/mass $i = 1,2,\ldots,N$ **do**
4: Train classifier (e.g. KNN or SVM or eSNN) to evaluate the fitness for each agent;
5: Calculate mass M for each agent;
6: Calculate acceleration a for each agent;
7: Update the velocity V;
8: Update the position;
9: **end for**
10: **Stop until termination conditions are met**

In essence, the fitness function is evaluated by comparing the *actual* result and the *predicted* result. For instance, as discussed in section 2.4.4, the two-class classification problem yields four possible outcomes for prediction. True Positive (TP) and True Negative (TN) represent the correct classifications. A False Positive (FP) occurs when a negative outcome is incorrectly predicted as positive; while a False Negative (FN) occurs when a positive outcome is incorrectly predicted as negative. Recall from section 2.4.4 that the accuracy rate is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6.1}$$

6.3.4 Personalized Risk Evaluation

Accurate personalized risk evaluation is a crucial factor for medical decision making, if patients are to receive effective treatment. Up to date, several risk stratification models remain grounded in traditional statistical methods and in problem statements that have not evolved significantly over the years (Zeeshan & Ilan, 2010).

In addition, most clinical researches tend to study outcomes across the global population of patients, rather than by developing personalized profiles for individual patient. Thus, the outcomes are difficult to interpret in some situations. For example, the reported morbidity rate for a procedure may not apply to an individual patient, or might be elevated in patients suffering from other ailments.

In developing an individual-tailored medical profile, this thesis evaluates personal-

ized risk/probability associated with a classifier. In certain applications, users request not only a classification, but also the probabilities of belonging to each class. In this study, risk/probability (opposed to class label prediction) is assessed using *probability-based KNN* and *probability-based SVM*. These classifiers are discussed below:

Probability-based KNN

KNN is a completely nonparametric method: that is it makes no assumptions about the nature of the data and does not distinguish between high-risk and low-risk patients. Instead, it predicts patient risk from the outcomes of similar historical cases.

For instance, consider a classification problem with two classes $C_m, m = 1, 2$, and N training samples $x(n), n = 1, \ldots, N$. The class posterior probability is defined as $P(C_m|x)$. Given that K_m is the number of k-nearest neighbors to point x belonging to class C_1, the risk is estimated as:

$$P(C_1|x) = \frac{K_m}{K} \tag{6.2}$$

This method is best illustrated by a simple example, such as a training sample with a true class label "2". To predict the class label for this sample using personalized modeling, we firstly select its nearest neighbors by setting k=5 (e.g. 4 samples from class 2 and 1 sample from class 1). Thus, the label is predicted as "2" because most of the selected k-nearest neighbors are of class 2. The probability of this sample is computed as "0.8" (= 4/5) implying that 80% of the selected neighbors belong to class 2.

Probability-based SVM

SVM learns an optimal decision boundary by which to sperate historical cases with different outcomes. Standard SVM predicts class label by computing a decision function $f(x)$ rather than probabilities. Platt (1999) proposed that SVM predictions could be transformed to posterior probabilities by a sigmoid function. Numerical

difficulties in Platt's approach were averted in the extended SVM of Lin and Weng (2007). Both approaches are explained in more detail below:

- *Platt's Approach*

  Givn a set of training examples $x_i \in \mathbb{R}^n, i = 1, \ldots, l$, labeled by $y_i \in \{-1, +1\}$. Platt (1999) proposed a sigmoid function to compute a posterior class probability $P(y = class|input)$.

  Mathematically, Platt approach is defined as:

$$P(y|f) = \frac{1}{1 + exp(Af + B)},\ where\ f = f(x) \tag{6.3}$$

  where f is the decision function of the binary SVM, and A and B are two scalar values fitted by maximum likelihood from a training set (f_i, y_i), which is a cross-entropy error function:

$$arg\ min_{A,B}\{-\sum_i t_i log(p_i) + (1 - t_i) log(1 - p_i)\} \tag{6.4}$$

  with t_i target probabilities, defined as:

$$t_i = \frac{y_i + 1}{2} \tag{6.5}$$

  Two questions arise: *what is the origin of the sigmoid train set?* and *how is overfitting to this training set?*

  Platt (1999) addresses both questions by adopting the out-of-sample model. Given N_+ positive and N_- negative samples in the training set, for each training sample Platt replaces the binary assignment {0,1} with target values t_+ and t_- for all of the data in the sigmoid fit. The target values are defined as:

$$t_+ = \frac{N_+ + 1}{N_+ + 2};\ t_- = \frac{1}{N_- + 2} \tag{6.6}$$

  Hence,

$$p_i = \frac{1}{1 + exp(Af_i + B)},\ where\ t_i = \left\{ \begin{array}{l} \frac{N_+ + 1}{N_+ + 2}\ if\ y_i = +1 \\ \frac{1}{N_- + 2}\ if\ y_i = -1 \end{array} \right\} \tag{6.7}$$

- *Lin and Weng's Approach*

  Lin and Weng (2007) adopt Platt's approach to avoid the numerical difficulties in their implementation. The major difference between the two approaches is that Eq.(6.4) is solved by different optimization algorithms.

  Platt's approach uses the levenberg-marquardt (LM) method (Press, Flannery, Teukolsky & Vetterling, 1992), However, this method cannot efficiently converge to the minimum solution of Eq.(6.4) (for details see Lin et al. (2007)).

  In Lin and Weng's approach, the issues inherent in Platt's pseudo code are solved by Newton's method with backtracking line search (Nocedal & Wright, 1999). This proposed algorithm yields higher classification accuracy than Platt's approach on two UCI datasets (Sonar and Shuttle) (H. T. Lin et al., 2007).

  Lin and Weng's approach has been successfully integrated with the LibSVM (an integrated software for classification, regression and distribution estimation), it is used for estimating posterior class probabilities, rather than for SVM training and prediction (Chang & Lin, 2011).

Hence, this thesis hybridizes the GSA with probability-based KNN and LibSVM to construct a novel hybrid personalized system based on posterior class probabilities in binary problems. However, posterior probability cannot guarantee high classification accuracy. The main purpose is not to boost prediction accuracy, but to provide probability estimates for medical decision making to ensure that patients receive efficient diagnosis and treatment.

6.4 Prototypes of evoPM

This section presents three prototypes of evoPM that have been gradually developed, ranging from the simple implementation to the comprehensive implementation. These are:

1. **Prototype 1 - optimize** K

   In this prototype, only the number of nearest neighbor K is optimized for each sample. The fitness function is learned by probability-based KNN.

2. **Prototype 2 - optimize K and model parameters M_p**

   In this prototype, the nearest neighbor K and parameter(s) of a learning function (e.g. a classifier) M_p are optimized for each testing sample in D_i. Two classifiers are adopted as the learning function: probability-based SVM and eSNN.

   Model parameters are the regularization parameter (C) and the width of the Gaussian RBF (γ) for SVM; and modulation (m), threshold (θ), and similarity (s) for eSNN.

3. **Prototype 3 - optimize K, model parameters M_p, and features Fea**

   In this prototype, nearest neighbor K, model parameter(s) M_p, and feature mask Fea are optimized. Each sample in D_i is classified using the optimal or near-optimal sets of K, M_p and Fea. The fitness function is evaluated using all three classifiers: namely probability-based KNN, SVM and eSNN.

In this section, the efficacy of each evoPM is tested on the *Breast Cancer Wisconsin* dataset (Street, Wolberg & Mangasarian, 1993) achieved in the UCI Machine Learning Repository. This dataset contains 699 samples with 9 features: clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses. It includes 2 classes: benign (class 1) and malignant (class 2). The original dataset contains 16 samples with missing features; these are removed to construct a final dataset containing 683 samples.

As described in early chapters, personalized model construction normally incurs heavy computational burden. The creation of a personalized model for each testing sample requires intensive optimization processing. For this reason, the experimental study uses 100 randomly selected samples from the dataset to demonstrate whether the different evoPM prototypes can improve personalized modeling implementation.

To compare the performances of the novel proposed evoPM and classifiers with non-optimization, the classification accuracy is firstly obtained for classifiers without optimization. Table 6.1 illustrates the classification performance of a *global modeling* method (SVM); a *local modeling* method (ECF); and two classical *personalized modeling* methods (KNN and WKNN).

Table 6.1: *The classification accuracy of for Breast Cancer Wisconsin data obtained by different classifiers.*

Experimental Results			
Classifier	# selected genes	Overall Acc(%)	Class(1/2)(%)
SVM (RBF Kernal, gamma=0.5)	6	96.00	(98.18/93.33)
ECF	5	96.00	(96.36/95.56)
KNN(k=11)	4	96.00	(98.18/93.33)
WKNN(k=9, thre=0.5)	4	96.00	(98.18/93.33)
Original reported	-	94.00	-

All of the methods provide the same LOOCV classification accuracy (96.00%), slightly better than the result reported in the original publication (with 10-fold cross-validation on all data). However, no further information relevant to medical treatment design can be gleaned form these results. Hence, in the next section, the proposed novel evoPM will be applied to a personalized problem space for breast cancer classification.

6.4.1 Prototype 1 - Optimize K

In this experiment, probability-based knnGSA is used as the learning function for evaluating classification performance. Because this evoPM prototype does not automatically select optimal feature sets, but only searches the optimal numbers of K for each sample, the learning function is applied to data containing all features.

The overall classification accuracy achieved by knnGSA is 96.00% (98.18% for class1 and 93.33% for class2). The performance of prototype 1 is not improved relative to the best accuracy achieved by non-optimization classifiers. One possible reason is that in this prototype, the parameter K alone is optimized. However, the optimal K is searched for each sample and the simple prediction of class label is replaced by personalized risk/probability evaluation.

Table 6.2 summarises the classification performance of the top 10 samples processed using prototype 1. All of the benign samples (class 1) are correctly classified, but 2 malignant samples (class 2) are misclassified as benign. Furthermore, because the classifier estimates the risk/probability rather than class labels, predicted label evaluated as "1" (benign) based on its 5 nearest neighbors is indeed the true label.

Table 6.2: *The classification performance of the top 10 samples for Breast Cancer Wisconsin using knnGSA.*

knnGSA	Actual:	1	1	1	2	1	2	1	2	2	2
	Predicted:	1	1	1	1*	1	2	1	2	1*	2
	# of K:	9	7	4	12	9	9	7	12	11	9
	Probability:	1	0.8	0.8	0.8	0.667	1	1	1	0.909	1

Table 6.3: *The classification accuracy of Breast Cancer Wisconsin using svmGSA and esnnGSA.*

Experimental Results			
Classifier	# selected genes	Overall Acc(%)	Class(1/2)(%)
svmGSA (gamma=7.46, c=163.56, k=19 Ave)	9	97.00	(100/93.33)
esnnGSA (mod=0.75, thre=0.22, sim=0.32, k=11 Ave)	9	**98.00**	(98.18/97.78)

The probability of this sample is given as "0.8", meaning that 80% of the selected nearest neighbors belonging to class 1.

The next section will test the classification performance of prototype 2, and will investigate the effect of model parameters on accuracy.

6.4.2 Prototype 2 - Optimize K and model parameters M_p

This prototype is designed for optimizing numbers of K and the model parameters M_p. The fitness function is evaluated using probability-based svmGSA and esnnGSA as the learning functions.

The classification accuracy is seen to be improved in this prototype because the model parameters are optimized for efficient personalized modeling (see Table 6.3). In other words, the result answers the question raised in the previous section; classification performance dependents on appropriate choice of the model parameters.

Table 6.4 summaries the classification performance of the top 10 samples computed by svmGSA and esnnGSA, respectively. Figure 6.4 also shows that the probability is estimated by svmGSA. For each sample, the probability of each class is estimated. For example, the probability of sample 1 (in the benign group) is assigned to class

Table 6.4: *The classification performance of the top 10 samples for Breast Cancer Wisconsin using svmGSA and esnnGSA.*

svmGSA	Actual:	1	1	1	2	1	2	1	2	2	2
	Predicted:	1	1	1	2	1	1*	1	1*	2	1*
	# of K:	5	5	5	6	5	5	5	6	3	6
	C:	124.76	64.61	77.14	253.39	157.85	26.64	79.23	106.17	159.21	201.91
	γ:	0.62	0.61	0.59	0.29	0.51	0.25	0.49	0.53	0.67	0.33
esnnGSA	Actual:	1	1	1	2	1	2	1	2	2	2
	Predicted:	1	1	2*	2	1	2	1	2	2	2
	# of K:	5	5	6	5	5	6	5	5	5	6
	Mod:	0.81	0.13	0.67	0.26	0.44	0.54	0.68	0.51	0.60	0.35
	Thre:	0.34	0.38	0.19	0.44	0.34	0.21	0.13	0.21	0.20	0.15
	Sim:	0.37	0.21	0.24	0.19	0.13	0.33	0.35	0.27	0.06	0.43

1 and 2 is "0.8" and "0.2", respectively. Here the probability can be considered as a confidence value or threshold (0.5). Thus, the predicted label of sample 1 is class 1 with prob≥0.5.

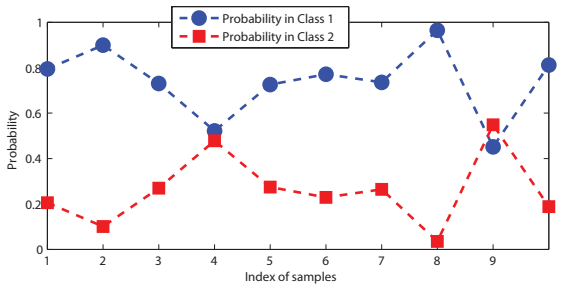


Figure 6.4: *The probability of each sample will be assigned to class 1 and class 2.*

6.4.3 Prototype 3 - Optimize K, model parameters M_p, and features Fea

As shown in the previous section, the personalized modeling approach has slightly improved the classification accuracy by optimizing relevant parameters K and model parameters M_p. Feature selection improves classification accuracy by reducing computational cost and noise, in essence, it selects the interpretable features that can help identify and monitor target diseases. Microarray gene expression analysis must process datasets containing tens of thousands of genes. Among these data, only a smaller number are strongly correlated with the targeted phenotypes. Thus, the

Table 6.5: *The classification accuracy of Breast Cancer Wisconsin using knnGSA, svmGSA and esnnGSA.*

Experimental Results			
Classifier	# selected genes	Overall Acc(%)	Class(1/2)(%)
knnGSA (k=11 Ave)	5	97.00	(98.18/95.56)
svmGSA (gamma=0.49, c=124.89, k=10 Ave)	5	**99.00**	(100/97.78)
esnnGSA (mod=0.52, thre=0.27, sim=0.24, k=12 Ave)	5	**99.00**	(100/97.78)

third prototype of evoPM with full optimization (K, M_p and Fea) is expected to offer the best classification of breast cancer samples.

As shown in Table 6.5, the classification accuracy is slightly improved as compared with the results achieved by the previous two prototypes. The results demonstrate the importance of feature selection, neighborhood and optimization of model parameters in advanced classification performance.

Figure 6.5 is an example of the classification result provided by svmGSA. All optimal sets of features, neighbors and model parameters are listed for the top 10 samples. The completed results for all 100 samples obtained by knnGSA, svmGSA and esnnGSA are presented in Appendices A, B and C respectively.

6.5 Summary

In conclusion, the experimental study has proved the hypothesis that the novel proposed evoPM can produce promising classification accuracy than global and local modeling methods through feature selection, neighborhood and model parameters optimization. Personalized modeling creates a unique model for each patient, ensuring that individuals receive a detailed medical profile. Such information will greatly assist personalized clinical decision system. In addition, the personalized risk for individual patient is evaluated by a classifier, as opposed to classifying patients into normal or diseased groups. A accurately quantifying this risk is critical for medical decision support, to ensure that patients receive the treatment that best matches their individual profile.

```
-----------------------------------------------------
  Sample ID       Best Gamma          Best C

       1              0.47           243.74
       2              0.78           103.18
       3              0.65           213.10
       4              0.42            84.90
       5              0.19           194.28
       6              0.77           172.58
       7              0.19            72.65
       8              0.64           151.25
       9              0.70           250.45
       10             0.39           229.70

Average number of Gamma:     0.52

Average number of C:  171.58

-----------------------------------------------------
  Sample ID       # K     KNN Index

       1         5       2 1 4 6 8
       2         5       2 1 6 4 8
       3         5       1 2 6 4 8
       4         5       9 5 7 8 4
       5         5       6 1 3 2 8
       6         5       9 7 8 4 6
       7         5       5 3 1 2 8
       8         6       8 6 9 4 5 7
       9         5       8 9 5 6 7
       10        5       6 8 4 9 5

Average number of K been selected:  6

-----------------------------------------------------
  Sample ID       # Features      Feature Index

       1         6              2 5 7 1 8 4
       2         6              6 5 9 3 1 8
       3         6              2 5 9 7 1 4
       4         5              6 5 3 8 1
       5         4              2 5 1 4
       6         3              2 3 7
       7         3              3 7 8
       8         3              5 8 1
       9         3              2 3 8
       10        6              2 5 1 8 7 4

Average number of Features been selected:  5
```

Figure 6.5: *Classification results of the top 10 samples for Breast Cancer Wisconsin, evaluated by svmGSA. All optimal sets of features, neighbors and model parameters are listed for each testing sample.*

To gain more insights into evoPM operation, the next chapter provides a comparative analysis of this novel personalized modeling system and framework. To this end, the feasibility of the system is tested on several gene expression benchmark datasets.

CHAPTER 7

Evolving Personalized Modeling System (evoPM) for Cancer Gene Expression Data Analysis

7.1 Introduction

Cancer, medically known as a malignant neoplasm, is the uncontrolled growth of abnormal cells in the body. In 2007, cancer caused about 13% of human deaths worldwide (7.9 million). Incidence are rising as more people live to an old age and as mass lifestyle changes occur in the developing world (Jemal et al., 2011). Cancer (of which more than 100 types have been identified) can develop in almost any organ or tissue in the human body, including lung, liver, colon, blood, breast, skin, and bones. Up to date, as new cancer gene expression data become available at an unprecedented speed, there is an increasing need for prognostic models to be continuously adaptive.

After introducing the relevant biological background, thus chapter discusses several information techniques used for evaluating gene expression data. Finally, the feasibility of the novel proposed personalized modeling system (evoPM) is evaluated. The classification performance of evoPM is compared with that of global and local modeling methods.

7.2 Biological Background

Molecular biology, conceptualized Warren Weaver in 1938, encompasses biology, chemistry, and especially biochemistry. Molecular biology attempts to explain the relationships intracellular between various systems, including the interactions between the different types of deoxyribonucleic acid (DNA), ribonucleic acid (RNA) and protein biosynthesis, as well as learning how these interactions are regulated.

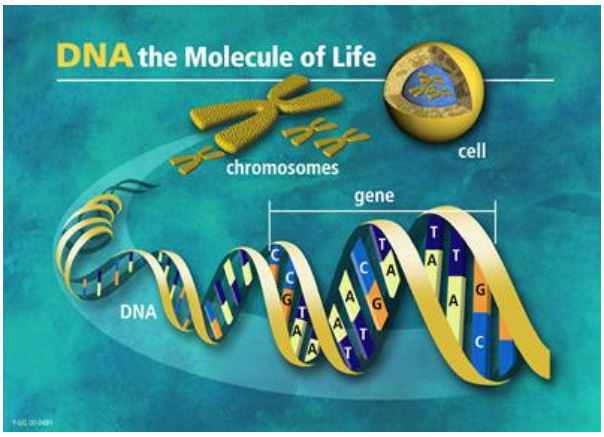
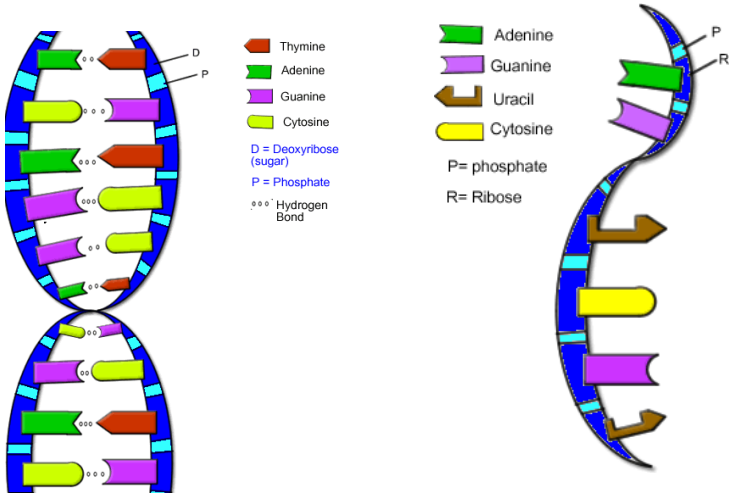


Figure 7.1: *The Molecule of life (Castellano, n.d.).*

The smallest structural and functional unit of all living organisms is the *Cell*. The human body contains about 10 trillion (10^{13}) cells of diverse shapes, sizes, and functions, but sharing a common basic structure. A typical human cell contains 25,000 to 35,000 *genes*, which carry trait-determinings information. Genes are embedded in thread-like structures called *chromosomes*, which collectively contain the information required for cell grow and reproduction. Humans cells contain two sets of chromosomes, one set inherited from each parent, yielding 23 pairs of chromosomes per cell. The chromosomes and genes are composed of DNA, a schematic of which is shown in Figure 7.1.



(a) Schematic of double helical DNA structure formed by base pairs attached to a sugar-phosphate backbone.

(b) RNA contains the bases A (adenine), C (cytosine), G (guanine) and U (uracil).

Figure 7.2: *A diagram of DNA and RNA (Biology-Corner, n.d.).*

7.2.1 Deoxyribonucleic Acid (DNA)

DNA constitutes the hereditary material in all organisms. The genetic instructions which determine the development and functioning of an organism are stored in a segment of nucleotides represented by four genetic codes, namely **A** (adenine), **T** (thymine), **C** (cytosine) and **G** (guanine). DNA in a cell is in a double helix structure formed by base pairs (A with T, C with G). Two individual DNA strands twist around each other in a right-handed spiral (see Figure 7.2 (a)).

7.2.2 Ribonucleic Acid (RNA)

RNA is a close cousin of DNA, is created from a DNA template in a process called transcription. RNA serves multiple roles in living cells; serving as a temporary copy of genes for protein synthesis *(messenger RNA-mRNA)*, functioning as an adaptor molecule that decode the genetic code *(transfer RNA-tRNA)* and catalyzing the synthesis of proteins *(ribosomal RNA-rRNA)*. Like DNA, RNA is made up of a

long chain of components called nucleotides, labeled **A** (adenine), **G** (guanine), **C** (cytosine) and **U** (uracil). However, unlike DNA, most RNA is a single-stranded molecule of considerably shorter chain structure (see Figure 7.2 (b)).

7.2.3 Gene Expression

Gene expression refers to the process that converts the genetic information contained in DNA into proteins. The gene conversion process, which occurs in two major stages, is summarized in Figure 7.3 below:

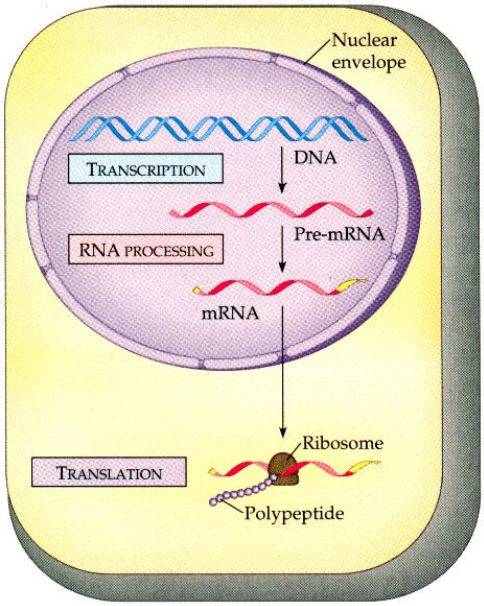


Figure 7.3: *Simplified overview of gene structure and expression (Berg, 2007).*

1. In the first stage, genetic information is *transcribed* from DNA to mRNA.

   In this process, the gene is copied to produce a RNA molecule (a primary transcript) with essentially the same sequence as the gene. In most human genes, the exons carry the information required for protein synthesis, are interspersed with non-translated sequences termed introns. Most primary transcripts are therefore processed by splicing to remove intron sequences and generate a mature transcript or mRNA containing exons alson.

2. In the second stage, genetic information is *translated* from mRNA to protein.

   In this process, no direct correspondence exists between the nucleotide sequence in DNA (and RNA) and the sequence of amino acids in the protein. In fact, each amino acid is encoded by three nucleotides. The chain of amino acids folds up to generate the final tertiary structure of the protein.

7.2.4 Techniques used for Evaluating Gene Expression Data

The numerous technologies available for analyzing gene expression levels in living cells are divided into two broad categories: DNA microarray-based techniques and computational techniques.

DNA Microarray-based Techniques

DNA microarray technology (also known as DNA chips), assesses mRNA levels in particular cells or tissues when many genes are activated simultaneously. This technology has been widely applied to tumor classification and prediction of clinical outcomes (Beer et al., 2002; Nielsen et al., 2002; van't Veer et al., 2002), identification of genes involved in various diseases, and elucidation of biological pathways (Yoshimoto et al., 2002).

Several types of DNA microarray-based technologies have been developed for measuring the thousands of genome-wide expression values in parallel. The two popular microarray technologies are those of complementary DNA (cDNA) (Schena, Shalon, Davi & Brown, 1995) and high-density oligonucleotides (Lockhart et al., 1996).

- *cDNA microarrays*

  The first reported DNA microarray technology, that of cDNA was developed by Patrick Brown and his colleagues at Stanford University. It involves the micro spotting of pre-fabricated cDNA fragments onto a glass slide.

  The advantages of this technology includ: (1) Readily accessible requiring no specific equipment for use and therefore incurring low cost, and (2) data can be captured using equipment that is frequently already available in the laboratory.

However, intensive labor is required for synthesizing, purifying and storing DNA solutions prior to microarray fabrication.

- *High-density oligonucleotide microarrays*

  This is a sophisticated platform of microarray technology, first developed by Stephen Fodor et al. in 1991. Presently, the main proponent of this technology is the commercial version of Affymetrix GeneChips, which holds up to 500,000 probes/sites in a 1.28-cm^2 chip area.

  This technology offers fast speed, high specificity and reproducibility, but its use is restricted by high cost and inflexibility. Expensive specialized equipments are required to carry out the hybridization, label staining, and washing.

The recent advance of microarray technologies has allowed the simultaneous monitoring of thousands of genes, with promising results. Nevertheless, several issues need to be addressed and understood, including:

- The dimensionality of gene expression data is very high. A gene expression dataset usually contains thousands to tens of thousands of genes; including numerous noise genes that impede the performance of computational models;
- The vast number of genes incurs heavy computational cost.

Clearly, existing microarray technologies cannot readily handle the above problems efficiently and effectively. Thus, biology and medicine would benefit greatly from automated analysis of complicated gene expression data with identification of relevant genes.

Computational Techniques

In recent years, many microarray data classification algorithms have been proposed for the diagnosis or prognosis of cancer diseases. Most of these are derived from computational machine learning algorithms, and fall into one of categories: supervised learning and unsupervised learning.

- *Supervised learning*

  Supervised learning is the search for a gene expression signature that predicts class membership. This approach begins with two data sets, a training set and a testing set. A model based on the chosen classification method is constructed using the training set; the testing set is then used to evaluate the classifier. To date, different supervised methods have been used to classify patient samples, such as SVM (Guyon, Weson, Barnhill & Vapnik, 2002), KNN (Yeang, Ramaswamy & Tamayo, 2001), and bagging and boosting (Tan & Gibert, 2003).

- *Unsupervised learning*

  Unsupervised learning is the search for a biologically relevant unknown taxonomy identified by a gene expression signature or a biologically relevant set of co-expressed genes. In other words, learning models of biological processes and relationships among genes are based entirely on their expression levels. Various unsupervised learning algorithms have been applied to gene expression data analysis, such as Bayesian networks (Hwang, Cho, Wook Park, Kim & Zhang, 2002), hierarchical clustering (Eisen, Spellman, Brown & Botstein, 1998), self-organizing maps (SOM) (Tamayo et al., 1999), and k-means clustering (Tavazoie, Hughes, Campbell, Cho & Church, 1999)

7.3 Cancer Gene Expression Data Analysis Using evoPM

Cancer is one of the major research fields in medical research. Accurate prediction of different tumor types could greatly benefit treatment provision and reduce toxicity to patients.

Existing computational methods facilitate more accurate diagnosis and prognosis of cancer. However, the treatments suggested by traditional global modeling systems for complex disease diagnosis and prognosis are effective for only 70% of the patients, leaving considerably large portion (approx. 30%) gaining no benefits from the treatment (Shabo, 2007). Hence, this section presents a comparative study that tests the

novel proposed personalized modeling system (evoPM) on four benchmark cancer gene expression datasets for classification tasks.

7.3.1 Datasets Description

The four benchmark datasets are: Colon cancer, Leukaemia, Lymphoma and Lung cancer. These datasets are publicly available and have been used in several published cancer classification studies.

- *Colon cancer data* (Alon et al., 1999)

  The dataset comprises 62 samples, 22 normal patients (class 1) and 40 cancer patients (class 2). In this data set, only 2,000 genes out of total 6,500 genes are selected based on confidence in the measured expression levels.

- *Leukaemia data* (Golub et al., 1999)

  This dataset consists of 72 samples and 7129 genes from 6817 human genes. 47 samples are labelled as Acute Lymphoblastic Leukaemia (ALL- class1); the remainder 25 samples are labelled as Acute Myeloid Leukaemia (AML-class2).

- *Lymphoma data* (Shipp et al., 2002)

  This dataset contains 77 samples, 58 Diffuse large B-cell lymphoma (DLBCL) samples and 19 Follicular lymphoma (FL) samples. Each sample is represented by 7129 genes.

- *Lung cancer data* (Gordon, Jensen, Hsiao, Hsiaox & JE, 2002)

  This dataset was originally used for distinguishing between malignant pleural mesothlioma (MPM) and adenocarcinoma (ADCA). The complete data set consists of 181 tissue samples (16 MPM /165 ADCA) and 12533 genes.

7.3.2 Experiment Setup and Results

The experimental results from four benchmark datasets through the unbiased validation schema are encouraging. The quality of the optimized classifier with the selected most important genes is validated by LOOCV. To compare the performance

of different methods, I here applied a *global modeling* method (SVM); a *local modeling* method (ECF); two classical *personalized modeling* methods (KNN and WKNN); and two *evoPM*-based methods (knnGSA and svmGSA).

7.3.3 Colon Cancer Dataset

Colon cancer results from uncontrolled growth of cells in the large intestine. Causing 655,000 deaths worldwide per year, this is the fourth most common cancer in the United States and the third leading cause of cancer-related death in the Western world (WHO, n.d.).

Table 7.1 summarizes the classification accuracy obtained from several classifiers on the colon cancer dataset. The highest accuracy obtained by svmGSA provides is 87.10%, approximately 5% better than the best accuracy achieved by tradition SVM classifier. In addition, knnGSA and svmGSA tend to select fewer genes than other algorithms.

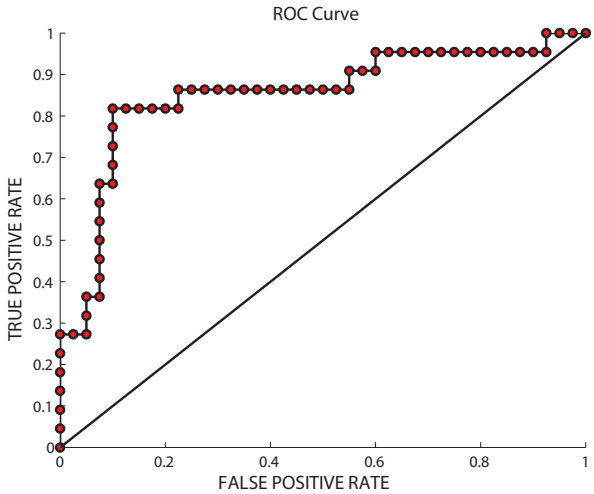


Figure 7.4: *ROC curve computed by svmGSA on the colon dataset.*

Figure 7.4 plots the classification performance of svmGSA on the colon cancer data. The classification performance is assessed by a ROC curve, where the x and y axes denote false positive rate (1-specificity) and true positive rate (sensitivity), respectively.

The novel evoPM can select the optimal or near-optimal sets of genes for each in-

Table 7.1: *Classification accuracy of different models, tested on the colon cancer dataset.*

Experimental Results			
Classifier	Number of selected genes	Overall Acc(%)	Class(1/2)(%)
SVM (RBF Kernal, gamma=0.5)	40	82.14	(95.00/59.09)
ECF	200	72.30	(75.00/68.18)
KNN(k=5)	100	82.14	(90.00/68.18)
WKNN(k=5, thre=0.5)	100	82.14	(90.00/68.18)
knnGSA(k=10 Ave)	75(Ave)	85.48	(90.00/77.27)
svmGSA(gamma=0.7, c=51.66, k=14 Ave)	91(Ave)	**87.10**	(90.00/81.82)
Original reported	-	87.00	-

Note: The parameters are selected through the same optimization process if they are employed in evoPM models. The parameters in SVM, ECF and KNN are selected based on the best classification performance. For the global SVM parameters, only the parameter γ is tuned.

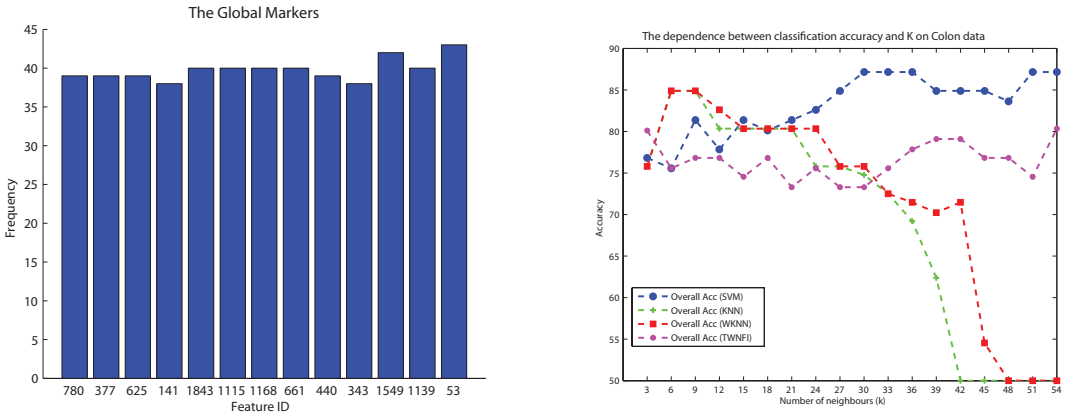
dividual patient. Thus, the global markers are computed based on the selection frequency over all samples obtained using svmGSA. The global markers with different numbers of neighbors then are re-evaluated by four classifiers (namely SVM, KNN, WKNN, and TWNFI) to investigate:

- Whether they are the most efficient features are selected;
- Whether they can significantly improve the classification performance.

Experimental results are valuated by LOOCV. Figure 7.5 shows the global markers and the results obtained using four classifiers with different number of neighbors. We note that the accuracy changes with K, and that SVM offers the highest classification accuracy followed by TWNFI. However, the accuracy of KNN and WKNN are dramatically decreased as K increases beyond 30.

In addition, the novel evoPM can create a personalized profile for each individual patient. Here one example - sample 20 of colon cancer data is given for demonstrating the profile of personalized modeling using svmGSA (see Table 7.2).

For ease of visualization, the 28 nearest neighbors of colon sample 20 are plotted in a 3D space (representing the 3 most important features/genes) (see Figure 7.6).



(a) 13 global markers of Colon cancer dataset with selecting frequency threshold set at 36 times.

(b) Classification accuracy achieved by 4 classifiers with different number of neighbors, assessed for 13 global markers of Colon cancer dataset.

Figure 7.5: *A set of global markers of Colon dataset and the results obtained using four classifiers with different number of neighbors.*

Table 7.2: *The optimal sets of features/genes, nearest neighbors, and model parameters, optimized solely for sample 20 of Colon cancer dataset based on one-run testing.*

Optimal set of features/genes	**Feature ID (7 total):** 249 1772 1423 1582 267 513 177
Optimal set of nearest neighbors	**KNN Index (28 total):** 18 7 8 5 26 40 27 43 45 19 2 58 10 32 13 9 59 34 49 42 61 38 60 12 33 57 44 41
Optimal SVM parameter	**Best γ:** 0.17; **Best C:** 204.08

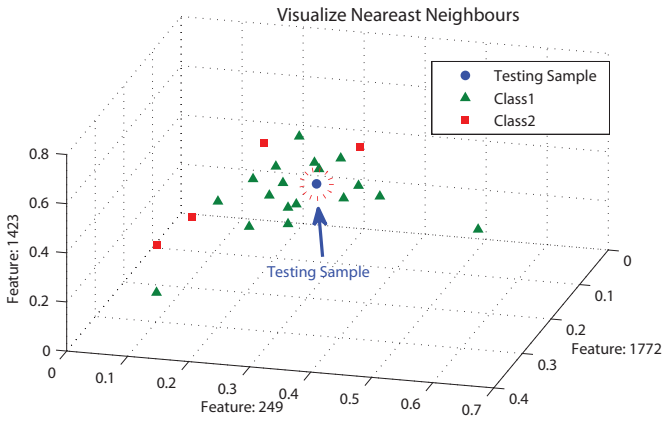


Figure 7.6: *K-nearest neighbors of sample 20 of colon data set.*

Based on the most common state of its nearest neighbors (normal group), sample 20 is more likely to be in the normal group.

7.3.4 Leukaemia Dataset

Leukaemia is a cancer of the blood or bone marrow, is characterized by an abnormal increase of immature white blood cells called "blasts". In 2000, approximately 256,000 children and adults worldwide developed some form of leukemia, and 209,000 died from it (Mathers, Boschi-Pinto, Lopez & Murray, 2001).

Table 7.3 lists the classification accuracy obtained by several classifiers tested on the leukaemia dataset. knnGSA and svmGSA achieve the same accuracy (97.22%), slightly better than that obtained by traditional SVM (95.83%). Figure 7.7 shows the classification performance obtained by svmGSA on the leukaemia data.

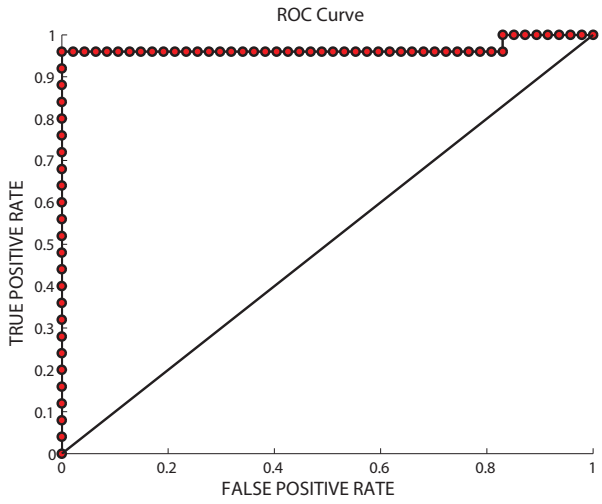
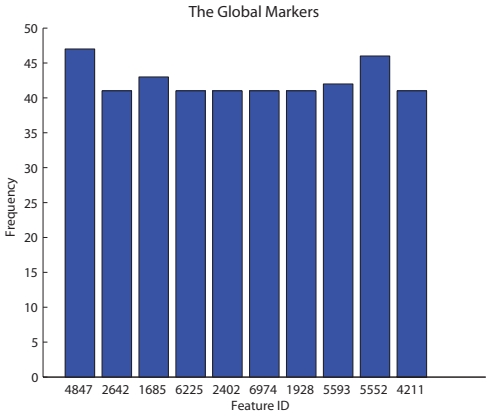


Figure 7.7: *ROC curve computed by svmGSA on the leukaemia dataset.*

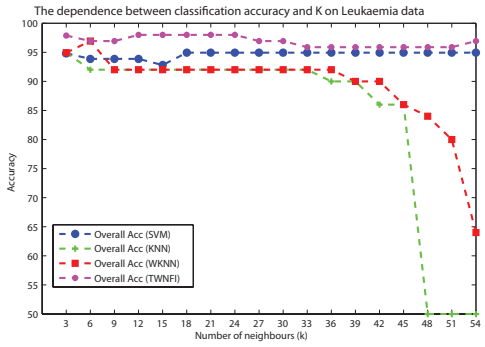
Figure 7.8 shows the global markers and the results obtained from four classifiers (SVM, KNN, WKNN, and TWNFI) assigned different numbers of neighbors. All classifiers are validated by LOOCV. TWNFI offers the best performance followed by SVM. We also note that accuracy is not significantly affected by K, except in the KNN and WKNN methods.

Table 7.3: *Classification results of different models, tested on the leukaemia dataset.*

Experimental Results			
Classifier	Number of selected genes	Overall Acc(%)	Class(1/2)(%)
SVM (RBF Kernal, gamma=0.6)	40	95.83	(95.74/96.00)
ECF	40	94.44	(97.87/88.00)
KNN(k=8)	30	94.44	(95.74/92.00)
WKNN(k=5, thre=0.5)	30	94.44	(95.74/92.00)
knnGSA(k=16 Ave)	41(Ave)	**97.22**	(97.87/96.00)
svmGSA(gamma=0.78, c=49.28, k=31 Ave)	35(Ave)	**97.22**	(97.87/96.00)
Original reported	-	85.00	-
Note: The parameters are selected through the same optimization process if they are employed in evoPM models. The parameters in SVM, ECF and KNN are selected based on the best classification performance. For the global SVM parameters, only the parameter γ is tuned.			



(a) 10 global markers of Leukaemia dataset with selecting frequency threshold set at 40 times.



(b) Classification accuracy achieved by 4 classifiers with different number of neighbors, assessed on 10 global markers of Leukaemia dataset.

Figure 7.8: *A set of global markers of Leukaemia dataset and the results obtained using four classifiers with different number of neighbors.*

7.3.5 Lymphoma Dataset

Lymphoma is the development of malignant tumors in the lymph system. It has become increasing common in the modern world and is estimated to become the second or third largest cancer by 2025 (Chris, n.d.).

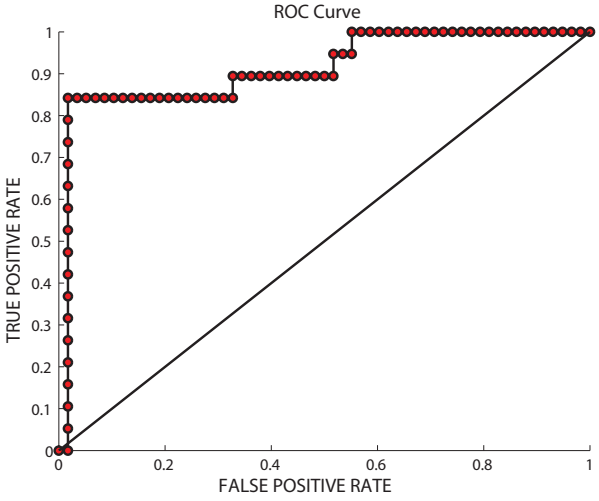


Figure 7.9: *ROC curve computed by svmGSA on the lymphoma dataset.*

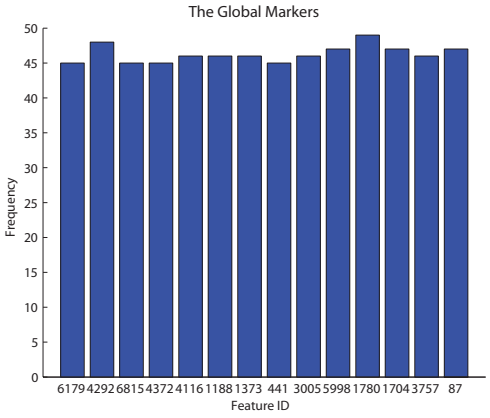
The classification performance of several classifiers tested on the lymphoma data is summarized in Table 7.4. Again, knnGSA and svmGSA achieve the same accuracy (94.81%), slightly better than that provided by traditional SVM (93.51%). Figure 7.9 plots the classification performance obtained by svmGSA on the lymphoma data.

Figure 7.10 illustrates the global markers and the results obtained from four classifiers (SVM, KNN, WKNN, and TWNFI) assigned different numbers of neighbors. All classifiers are validated by LOOCV. From the figure we observe that SVM, KNN and WKNN provide similar results as K increases from 3 to 33, and that accuracy is robust within this range. However, the accuracy of KNN and WKNN drop sharply at K larger than 33, and re-stabilizes at a low level once K reaches 39.

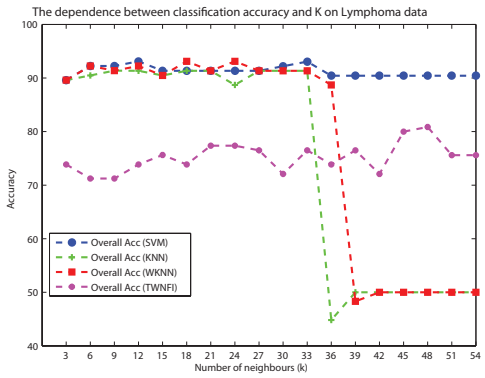
Table 7.4: *Classification accuracy of different models, tested on the lymphoma dataset.*

Experimental Results			
Classifier	Number of selected genes	Overall Acc(%)	Class(1/2)(%)
SVM (RBF Kernal, gamma=0.5)	80	93.51	(96.55/84.21)
ECF	67	92.21	(93.10/89.47)
KNN(k=3)	90	93.51	(93.10/94.74)
WKNN(k=3, thre=0.5)	90	93.51	(93.10/94.74)
knnGSA(k=3 Ave)	101(Ave)	**94.81**	(94.83/94.74)
svmGSA(gamma=0.8, c=50.72, k=17 Ave)	52(Ave)	**94.81**	(98.28/84.21)
Original reported	-	92.20	-

Note: The parameters are selected through the same optimization process if they are employed in evoPM models. The parameters in SVM, ECF and KNN are selected based on the best classification performance. For the global SVM parameters, only the parameter γ is tuned.



(a) 14 global markers of Lymphoma dataset with selecting frequency threshold set at 45 times.



(b) Classification accuracy achieved by 4 classifiers with different number of neighbors, assessed on 14 global markers of Lymphoma dataset.

Figure 7.10: *A set of global markers of Lymphoma dataset and the results obtained using four classifiers with different number of neighbors.*

Table 7.5: *Classification accuracy of different models, tested on the lung cancer dataset.*

Experimental Results			
Classifier	Number of selected genes	Overall Acc(%)	Class(1/2)(%)
SVM (RBF Kernal, gamma=0.7)	30	95.31	(94.77/95.86)
ECF	30	92.87	(92.15/93.58)
KNN(k=10)	25	96.60	(95.63/97.58)
WKNN(k=10, thre=0.6)	25	95.50	(94.25/96.76)
knnGSA(k=25 Ave)	25(Ave)	98.34	(90.32/100)
svmGSA(gamma=0.74, c=49.91, k=27 Ave)	26(Ave)	**98.90**	(96.77/99.33)
Original reported	-	90.00	-

Note: The parameters are selected through the same optimization process if they are employed in evoPM models. The parameters in SVM, ECF and KNN are selected based on the best classification performance. For the global SVM parameters, only the parameter γ is tuned.

7.3.6 Lung Cancer Dataset

Lung cancer is characterized by uncontrolled cell growth in tissues of the lung. As the most common cause of cancer-related death in men and women, it is responsible for 1.3 million deaths annually, as of 2004 (Ministry-Health, 2006).

The classification accuracy achieved using several classifiers on the lung cancer dataset is summarized in Table 7.5. Here, svmGSA provides the best accuracy is 98.90%, approximately 2% higher than the best accuracy achieved by the KNN classifier.

Figure 7.11 shows the classification performance obtained by svmGSA on the lung cancer dataset, plotted as a ROC curve. The global markers and the results of four classifiers (SVM, KNN, WKNN, and TWNFI) with different number of neighbors are illustrated in Figure 7.13. All classifiers are validated by LOOCV. In this experiment, all of the algorithms provide the same accuracy when K=3 and K=6, but the accuracy of SVM is slightly increased when K=9 and is retained thereafter. We note that the classification accuracy achieved by TWNFI is not affected by K, since it remains constant as K increases from 3 to 54. The accuracy of both KNN and WKNN decreases with increasing K.

Fig 7.12 summarizes the global marker genes of the Colon cancer and lymphoma

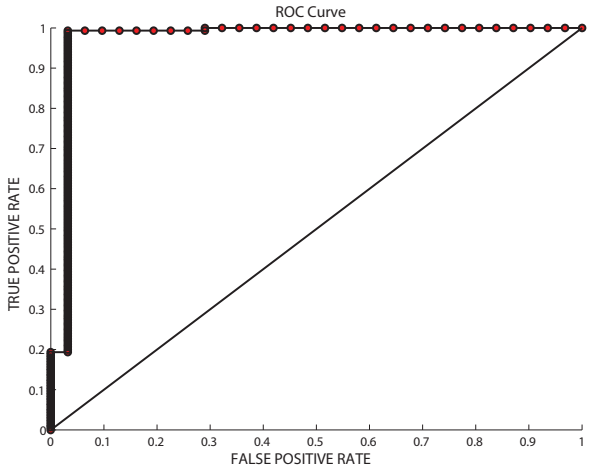
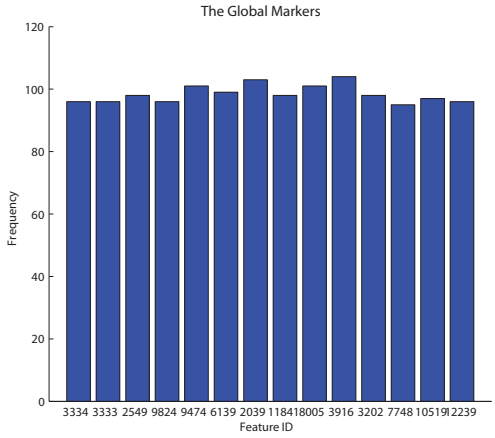


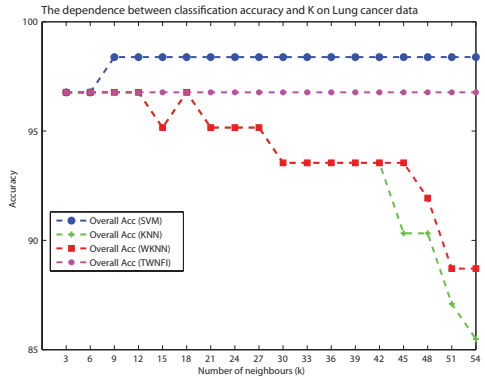
Figure 7.11: *ROC curve computed by svmGSA on the lung cancer dataset.*

Dataset	Index of Global Marker	Descriptions of Global Marker
Colon cancer	780	MACROPHAGE MIGRATION INHIBITORY FACTOR (HUMAN)
	377	H.sapiens mRNA for GCAP-II/uroguanylin precursor
	625	Human gene for heterogeneous nuclear ribonucleoprotein (hnRNP) core protein A1
	141	SM22-ALPHA HOMOLOG (HUMAN)
	1843	GELSOLIN PRECURSOR, PLASMA (HUMAN)
	1115	SERINE/THREONINE-PROTEIN KINASE IPL1 (Saccharomyces cerevisiae)
	1168	Human isoleucyl-tRNA synthetase mRNA, complete cds
	661	P02403 60S RIBOSOMAL PROTEIN
	440	SINGLE-STRANDED DNA-BINDING PROTEIN MITOCHONDRIAL PRECURSOR (HUMAN)
	343	Human mRNA for polyA binding protein
	1549	VASCULAR ENDOTHELIAL GROWTH FACTOR (Cavia porcellus)
	1139	RAN-SPECIFIC GTPASE-ACTIVATING PROTEIN (Homo sapiens)
	53	ELONGATION FACTOR 1-GAMMA (HUMAN)
Lymphoma	6179	ENO1
	4292	PKM2
	6815	Tubulin
	4372	GM2A
	4116	ALDOA
	1188	26S
	1373	Macrophage
	441	Proteasome
	3005	Bcl-2
	5998	mRNA
	1780	L-myc
	1704	ADA
	3757	Clone
	87	SLC

Figure 7.12: *The global marker genes discovered by evoPM for Colon cancer and Lymphoma data.*



(a) 14 global markers of Lung cancer dataset with selecting frequency threshold set at 95 times.



(b) Classification accuracy achieved by 4 classifiers with different number of neighbors, assessed on 14 global markers of Lung cancer dataset.

Figure 7.13: *A set of global markers of Lung cancer dataset and the results obtained using four classifiers with different number of neighbors.*

datasets (no supplementary information was available for the leukaemia and lung cancer datasets).

7.4 Summary

In conclusion, the proposed evoPM consistently improves the classification accuracy on four benchmark gene expression datasets, relative to previously published results. The results obtained by evoPM are significantly improved on lymphoma, leukaemia and lung cancer data, and are slightly improved on colon cancer data. The proposed system not only outperforms several global and local modeling methods in terms of diagnostic and prognostic accuracy, but finds the optimal or near-optimal solution to feature selection, neighborhood and model parameters optimization with significantly reduced computational cost. The results support the hypotheses that the classification accuracy for each dataset is improved using the discovered global markers. Furthermore, the classification performance is robust to K for SVM and TWNFI, but depends on K for KNN and WKNN. In other words, the global markers have been successfully selected.

In the next chapter, the novel evoPM system will be applied to stroke occurrence data

as a case study to explore the associations between changes in weather conditions and stroke occurrence.

CHAPTER 8

Evolving Personalized Modeling System (evoPM) for Weather and Stroke Occurrence Data Analysis

8.1 Introduction

Increasingly, number of studies have identified a link between weather conditions and stroke occurrence (Z. Y. Chen, Chang & Su, 1995; V. L. Feigin, Nikitin, Bots, Vinogradova & Grobbee, 2000). From early evidence, environmental triggers of different stroke are known as subtypes depend on age, gender and climatic factors. However, previous data are selection-biased (e.g. unclear CT (computed tomography)/MRI (magnetic resonance imaging) verification of different stroke subtypes), or reliable data is missing in various population groups (e.g. age, gender, and region).

Thus far, only a few studies have effectively explored the effect of weather on stroke occurrence, and in most of these have yielded inconsistent stroke occurrence predictions (Biller, Jones, Bruno, Adams & Banwart, 1988; Ricci et al., 1992; Nyquist, Brown, Wiebers, Crowson & OFallon, 2001). Thus, the effect of weather on stroke occurrence remains a matter of uncertainty and controversy. This chapter presents a comparative study in which associations between changes in weather conditions

and stroke occurrence are analyzed using conventional global, local, personalized modeling methods, and evoPM-based algorithms. Particular, attention is devoted to gender differences in weather and stroke occurrence.

8.2 Pilot Analysis

8.2.1 Background

As evidenced in several studies, sub-optimal ambient temperature and atmospheric pressure, as well as winter season, are associated with a rise in affect coronary heart disease death and incidences of heart attack (Z. Y. Chen et al., 1995; V. L. Feigin et al., 2000). The effect of these and other weather parameters on stroke occurrence remains a matter of controversy and uncertainty. Therefore, the significant associations between weather parameters and stroke occurrence must be clearly identified. This knowledge will contribute significantly to understanding the environmental triggers of stroke. In turn, other novel areas of research must be identified, such as physiological weather-stroke associations or clinical trials, from which preventive strategies against harmful weather conditions may be developed.

8.2.2 Study Areas

This international collaborative study is carried out under the auspices of six population regions: Auckland (NZ), Perth and Melbourne (Australia), Oxfordshire (UK), Dijon (France), Norrbotten and Vasterbotten (Northern Sweden).

The study areas are grouped in the Southern Hemisphere (Auckland, Perth, and Melbourne) and Northern Hemisphere (Oxfordshire, Dijon, Norrotten and Vasterbotten counties). Table 8.1 summarizes the number of patients in each region.

8.2.3 Dataset Description

The complete dataset consists of 11,453 samples (all with first occurrence of stroke) and 9 features (4 patient clinical features and 5 weather features):

Table 8.1: *Number of patients in each region participating in the global study.*

Region	**Number of patients**
Auckland	2805
Dijon	1756
Melbourne	1316
Oxfordshire	543
Perth	766
Sweden	4267
Total	**11453**

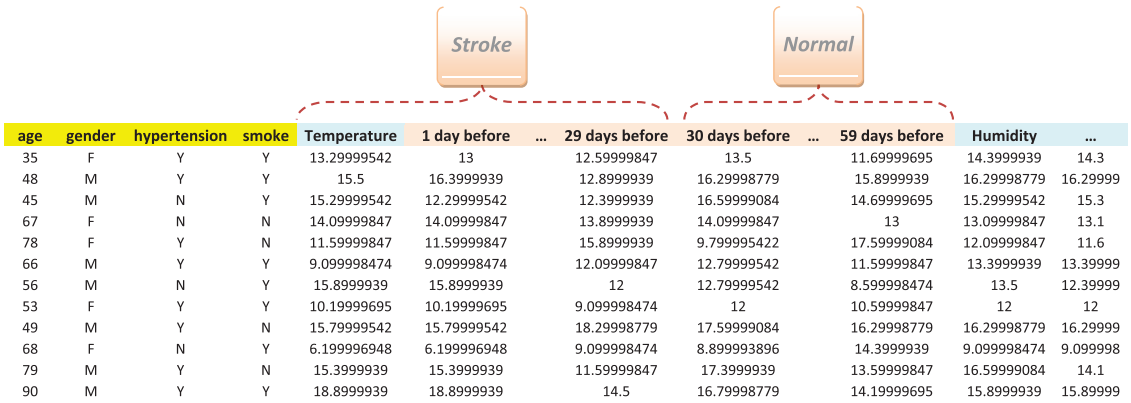
- *Patient clinical features (categorical data)* - age, gender, history of hypertension and smoking status.
- *Weather features (continuous data)* - temperature, humidity, wind speed, wind-chill and atmospheric pressure.

In fact, all the weather parameters are measured only for the day of stroke occurrence. As suggested by the medical expert, we make-up the 59 days pre-stroke occurrence data based on the measurement of day of stroke occurrence for spatio-temporal data analysis purpose. Figure 8.1 demonstrates how we make-up the pre-stroke occurrence data, by using an example of the patient high-lighted in the figure, whose age is 84 has stroke occurrence at the day "3-09-1981", the temperature for this day is measured as "23.19999695". Thus the 1 day pre-stroke occurrence is the day "3-08-1981", the temperature for this day is "22.29998779" according to the day of stroke occurrence. Furthermore, the day of 7 days pre-stroke is "3-02-1981", since there is no patient has stroke occurrence at this day, but the closest day is "3-03-1981", thus we use the temperature of this day instead which is "21.3999939", assuming that the temperature might not have significant differences between day "3-02-1981" and day "3-03-1981". The rest data is done in the similar manner.

Case-crossover design is a longitudinal study, which represents a special situation in which no group exists for separate comparison. In effect, each subject serves as his/her own control. By assigning both treatments to the same subject covariates imbalance is precluded. This design has been widely applied in many medical and health studies. Mukamel and his colleagues (Mukamal, Wellenius, Suh & Mittleman, 2009) used this approach to compare measures of weather and ambient air pollution on the day of stroke presentation and on other days (as control) for each patient.

age	sex	stroke_d	hyp	smok	season	tempCels	temp_lag1	temp_lag2	temp_lag3	temp_lag4	temp_lag5	temp_lag6	temp_lag7	temp_lag8	temp_lag9	temp_lag10
44	1	1981/1/3	2	1	1	22.09999084										
88	2	3/01/1981	1	1	1	22.09999084	22.0999908									
50	1	1981/3/3	1	2	1	21.3999939	22.0999908	22.0999908								
72	1	3/03/1981	2	3	1	21.3999939	21.3999939	22.0999908	22.0999908							
62	2	1981/4/3	1	1	1	21	21.3999939	21.3999939	22.0999908	22.099991						
76	1	3/05/1981	2	1	1	19.79998779	21	21.3999939	21.3999939	22.099991	22.09999084					
56	1	3/06/1981	2	2	1	19.59999084	19.7999878	21	21.3999939	21.399994	22.09999084	22.0999908				
80	2	3/06/1981	1	1	1	19.59999084	19.5999908	19.7999878	21	21.399994	21.3999939	22.0999908	22.0999908			
68	1	3/07/1981	2	3	1	20.29998779	19.5999908	19.5999908	19.7999878	21	21.3999939	21.3999939	22.0999908	22.099991		
84	2	3/08/1981	2	1	1	22.29998779	20.2999878	19.5999908	19.5999908	19.799988	21	21.3999939	21.3999939	22.099991	22.09999084	
84	1	3/09/1981	2	3	1	23.19999695	22.2999878	20.2999878	19.5999908	19.599991	19.79998779	21	21.3999939	21.399994	22.09999084	22.09999084
80	2	3/10/1981	2	1	1	21.69999695	23.199997	22.2999878	20.2999878	19.599991	19.59999084	19.7999878	21	21.399994	21.3999939	22.09999084
80	2	3/10/1981	1	3	1	21.69999695	21.699997	23.199997	22.2999878	20.299988	19.59999084	19.5999908	19.7999878	21	21.3999939	21.3999939
64	1	3/12/1981	2	3	1	19.29998779	21.699997	21.699997	23.199997	22.299988	20.29998779	19.5999908	19.5999908	19.799988	21	21.3999939
70	1	3/12/1981	2	3	1	19.29998779	19.2999878	21.699997	21.699997	23.199997	22.29998779	20.2999878	19.5999908	19.599991	19.79998779	21
82	1	3/14/1981	2	3	1	18.8999939	19.2999878	19.2999878	21.699997	21.699997	23.19999695	22.2999878	20.2999878	19.599991	19.59999084	19.79998779
66	1	3/16/1981	2	3	1	20.09999084	18.8999939	19.2999878	19.2999878	21.699997	21.69999695	23.199997	22.2999878	20.299988	19.59999084	19.59999084

Figure 8.1: *Data pre-processing of spatio-temporal weather and stroke occurrence data.*



age	gender	hypertension	smoke	Temperature	1 day before	...	29 days before	30 days before	...	59 days before	Humidity	...
35	F	Y	Y	13.29999542	13		12.59999847	13.5		11.69999695	14.3999939	14.3
48	M	Y	Y	15.5	16.3999939		12.8999939	16.29998779		15.8999939	16.29998779	16.29999
45	M	N	Y	15.29999542	12.29999542		12.3999939	16.59999084		14.69999695	15.29999542	15.3
67	F	N	N	14.09999847	14.09999847		13.8999939	14.09999847		13	13.09999847	13.1
78	F	Y	N	11.59999847	11.59999847		15.8999939	9.799995422		17.59999084	12.09999847	11.6
66	M	Y	Y	9.099998474	9.099998474		12.09999847	12.79999542		11.59999847	13.3999939	13.39999
56	M	N	Y	15.8999939	15.8999939		12	12.79999542		8.599998474	13.5	12.39999
53	F	Y	Y	10.19999695	10.19999695		9.099998474	12		10.59999847	12	12
49	M	Y	N	15.79999542	15.79999542		18.29998779	17.59999084		16.29998779	16.29998779	16.29999
68	F	N	Y	6.199996948	6.199996948		9.099998474	8.899993896		14.3999939	9.099998474	9.099998
79	M	Y	N	15.3999939	15.3999939		11.59999847	17.3999939		13.59999847	16.59999084	14.1
90	M	Y	Y	18.8999939	18.8999939		14.5	16.79998779		14.19999695	15.8999939	15.89999

Figure 8.2: *Representative spatio-temporal weather and stroke occurrence data.*

Vlak et al. (2011) applied case-crossover design to identify trigger factors and their attributable risk for rupture of intracranial aneurysms.

In this study, case-crossover design is applied to the global dataset for data pre-processing, since no "non-stroke" patients exist in the original dataset. The period spanning 29 days pre-stroke occurrence to the stroke event (30 days time window) is considered as the "stroke" group. The "normal/control", for the same participant, spans days 30-59 post-stroke occurrence (30 days time window) (see Figure 8.2).

8.2.4 Experimental Setup

To our knowledge, this is the first study to use computational intelligent modeling techniques to investigate the associations between weather and stroke occurrence. Since this work is a pilot study focussing on real world medical data, the comparative study uses only data collected from the Auckland region.

The Auckland region comprises 2805 patients all experiencing first-ever occurrence of stroke. "Non-stroke" patients are absent in the original dataset. A case-crossover design is applied, in which the date of stroke occurrence (1 day lag) is considered as the *stroke* group, whereas 30 days prior to stroke occurrence (1 day lag) for the same participant is considered as the *normal/control* group, assuming that the weather parameters 30 days before the index stroke do not affect on the stroke occurrence 30 days later.

Due to the heavy computation burden of personalized modeling construction, this study uses a small section of the original dataset (500 randomly-selected patients). Hence, the data consist of 1,000 samples (500 "normal/control" patients (class1) and 500 "stroke" patients (class 2)).

Our experiments are carried out in three steps: (1) a simple comparative analysis using only 4 patient clinical features; (2) using all 9 features; (3) using 6 features (age and 5 weather features), as age is a continuous value and suggested to be used for the experiments by experts.

To provide a performance comparison from different methods, I have applied a *global modeling* method (SVM); a *local modeling* method (ECF); two classical *personalized modeling* methods (KNN and WKNN); and *evoPM*-based methods (knnGSA, svmGSA and esnnGSA) to the weather and stroke occurrence problem. All performances are validated by LOOCV. Irrelevant features are eliminated by a signal-to-noise-ratio (SNR) algorithm.

8.2.5 Experimental Result

Analysis Using 4 Patient Clinical Features

In this experiment, all modeling techniques are applied to the data assuming 4 relevant patient clinical features. The best classification accuracy manifested by the svmGSA model is 54.20% (53.80% for class 1-Normal/Control, and 54.60% for class 2-Stroke). The svmGSA model outperforms all other methods in terms of classification accuracy. However, the accuracy obtained from svmGSA is close to random, though the clinical variables selected namely age, gender, blood pressure and

smoking, are identified as very important stroke risk factors (Wannamethee et al., 1995; Hankey, 1999).

Analysis Using All 9 Features (4 Patient Clinical Features and 5 Weather Features)

In the second experiment, the same modeling techniques are applied to 9-feature data to explore whether accuracy is improved when taking weather features. Among the methods, esnnGSA obtains the best classification accuracy, averaged at 62.50% (61.94% for class 1 - Normal/Control, and 63.06% for class 2 - Stroke). Clearly, the classification accuracy of esnnGSA is significantly improved when 5 weather features are incorporated into the model. However, the result does not confirm whether weather conditions indeed have strong effect on stoke occurrence, possibly because three of the patient clinical features are categorical data (apart from age).

Analysis Using 6 Features (Age and 5 Weather Features)

Table 8.2 summarizes the classification performance of the tested modeling methods. Here esnnGSA yields the highest accuracy at 70.80% (68.60% for class 1 - Normal/Control, and 73.00% for class 2 - Stroke), almost 5% better than the highest accuracy achieved by conventional SVM method. In addition, the result is significantly improved relative to the cases of 4 features (17% improvement) and 9 features (8% improvement). We can more confident to state that weather and stroke occurrence are strongly correlated using 6 features.

8.2.6 Summary

As a general conclusion, the experiments suggest that: weather conditions significantly impact on stroke occurrence. The overall classification accuracy is significantly improved when weather features are incorporated into the experiments. This knowledge will contribute an understanding of environmental triggers of stroke. In addition, it will assist the health and medical experts in conducting new areas of research, such as physiological studies on weather-stroke associations, or preventive strategies to reduce the hazardous effects of harmful weather conditions.

Table 8.2: *Classification accuracy obtained by conventional global, local, personalized modeling approaches, and evoPM-based algorithms, assessed through LOOCV validation.*

Experimental Results		
Classifier	Overall Acc(%)	Class(1/2)(%)
SVM (RBF Kernal, gamma=0.6)	65.40	(60.80/70.00)
ECF	63.25	(62.50/64.00)
KNN(k=35)	64.80	(64.00/65.60)
WKNN(k=35, thre=0.5)	64.60	(65.20/64.00)
knnGSA(k=6 Ave)	66.20	(58.60/73.80)
svmGSA(gamma=0.76, c=49.36, k=15 Ave)	69.63	(66.50/72.76)
esnnGSA(mod=0.82, thre=0.34, sim=0.22, k=22 Ave)	**70.80**	(68.60/73.00)

Note: The parameters are selected through the same optimization process if they are employed in evoPM models. The parameters in SVM, ECF and KNN are selected based on the best classification performance. For the global SVM parameters, only the parameter γ is tuned.

8.3 Selected Case Analysis - by Gender

According to Framingham Heart Study (Seshadri, 2006), 1 in 6 men and 1 in 5 women aged over 55 will develop a stroke during their remaining lifetime. Increasing evidence is emerging for gender differences in stoke symptoms, prevention, diagnosis, treatment, and outcomes (Labiche, Chan, R. & Morgenstern, 2002; Di et al., 2003; Goto, Baba, Ito, Maekawa & Koshiji, 2007). This experiment aims to explore the gender differences in weather and stroke occurrence, as an extension of the previous pilot study.

8.3.1 Dataset Description

The same dataset is applied as in the previous pilot study. Figure 8.3 shows the number of strokes in each gender age-adjusted group across the population. The dataset contains 250 male patients and 250 female patients, of which stroke occurrence is much more likely in the 50-plus age group than those are younger than 50.

The important risk factors for stroke have been identified as age, gender, a history of hypertension and smoking. Thus, from the population of 500, this study selects

Table 8.3: *Classification accuracy of male group obtained by conventional global, local, personalized modeling approaches, and evoPM-based algorithms, assessed through LOOCV validation.*

Experimental Results - Male Group		
Classifier	Overall Acc(%)	Class(1/2)(%)
SVM (RBF Kernal, gamma=0.5)	66.22	(75.68/56.76)
ECF	66.22	(67.57/64.86)
KNN(k=9)	63.51	(72.97/54.05)
WKNN(k=15, thre=0.5)	64.86	(72.97/56.76)
knnGSA(k=16 Ave)	67.57	(70.27/64.86)
svmGSA(gamma=0.47, c=135.16, k=15 Ave)	67.57	(70.27/64.86)
esnnGSA(mod=0.76, thre=0.42, sim=0.25, k=18 Ave)	**68.74**	(71.35/66.13)
Note: The parameters are selected through the same optimization process if they are employed in evoPM models. The parameters in SVM, ECF and KNN are selected based on the best classification performance. For the global SVM parameters, only the parameter γ is tuned.		

the patients over 50 with a history of hypertension and smoking. These patients comprise 37 males and 21 females.

8.3.2 Experimental Setup

The setup of this experiment study is very similar to that of the previous pilot study; the difference is this case is that 5 weather parameters are used in a comparative analysis for both gender groups. These parameters were chosen for their significant impact on stroke occurrence. The following section details the analysis separated by gender.

8.3.3 Experimental Result

Male Group

Table 8.3 shows the classification results of the male group obtained by all modeling techniques. The evoPM-based esnnGSA provides the highest accuracy at 68.74%, a slightly improvement over other conventional methods.

Table 8.4: *Classification accuracy of female group obtained by conventional global, local, personalized modeling approaches, and evoPM-based algorithms, assessed through LOOCV validation.*

Experimental Results - Female Group		
Classifier	Overall Acc(%)	Class(1/2)(%)
SVM (RBF Kernal, gamma=0.4)	64.29	(66.67/61.90)
ECF	57.14	(47.62/66.67)
KNN(k=11)	66.67	(66.67/66.67)
WKNN(k=13, thre=0.5)	66.67	(61.90/71.43)
knnGSA(k=6 Ave)	69.05	(57.14/80.95)
svmGSA(gamma=0.42, c=114.91, k=13 Ave)	**71.43**	(66.67/76.19)
esnnGSA(mod=0.31, thre=0.23, sim=0.0, k=16 Ave)	70.25	(67.34/73.16)

Note: The parameters are selected through the same optimization process if they are employed in evoPM models. The parameters in SVM, ECF and KNN are selected based on the best classification performance. For the global SVM parameters, only the parameter γ is tuned.

To explore which weather parameters impact most strongly on the male patients, the global markers are computed based on the selecting frequency over all samples, obtained from svmGSA (see Figure 8.4). The feature "atmospheric pressure" is found as the marker parameter since it has been frequently selected, followed by "wind speed" and "wind chill".

Figure 8.5 presents an example of a personalized profile created by svmGSA for sample 10. Displayed are the optimal sets of features, nearest neighbors, and model parameters for sample 10 alone. Furthermore, the probability of stroke occurrence is estimated rather than assigning the sample to a particular group.

Female Group

Table 8.4 summarizes the classification performance for the female group. The highest accuracy (71.43%) is achieved by svmGSA, approximately 5% higher than that provided by the classical personalized modeling KNN method. Overall classification accuracy is higher for the female group than the male, suggesting that the weather parameters impact more heavily on the female group.

Figure 8.7 presents the global markers based on the selecting frequency over all

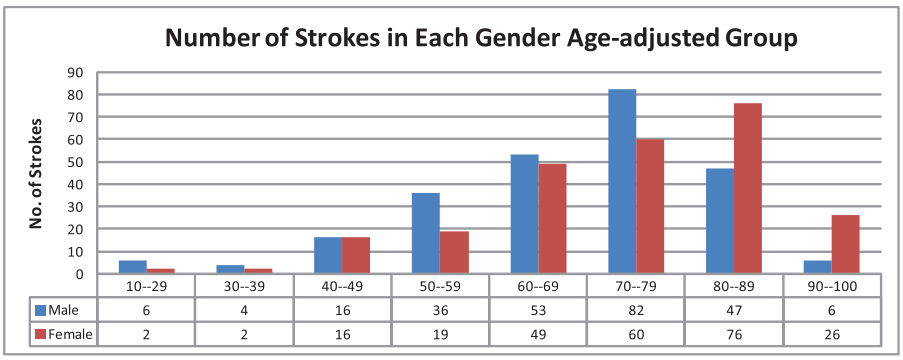


Figure 8.3: *Number of strokes in each gender age-adjusted group.*

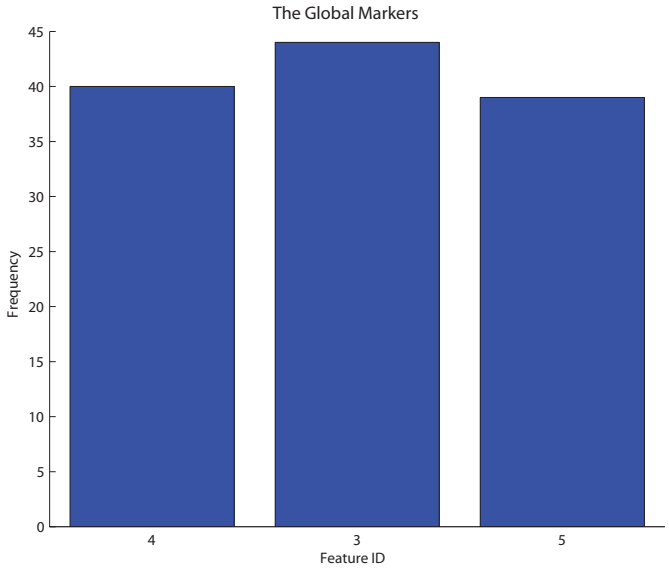


Figure 8.4: *The global markers of male group are computed based on the selecting frequency over all samples obtained using the svmGSA (G3=atmospheric pressure, G4=wind speed, and G5=wind chill).*

```
================================================================================ Result ================================================================================
 Sample ID       Actual Class       Predicted Class       Predicted Class (based on Probability)       Probability in Class1       Probability in Class2
        10                  1                     1                                            1                     0.85675                     0.14325

--------------------------------------------------------

 Sample ID       Best Gamma          Best C
        10             1.00          222.18

--------------------------------------------------------

 Sample ID       # K     KNN Index
        10       17      68 38 33 29 18 44 64 47 65 8 39 27 15 2 66 60 35

--------------------------------------------------------

 Sample ID       # Features      Feature Index
        10                2                4 3
================================================================================ End ================================================================================
```

Figure 8.5: *The optimal sets of features, nearest neighbors, and model parameters for sample 10 alone, based on 50 testing runs.*

samples obtained from svmGSA. "Wind speed" is the most frequently selected feature, followed by "temperature". The feature "wind speed" is consistently selected as the global marker for both gender groups.

```
================================================================================ Result ================================================================================
 Sample ID        Actual Class        Predicted Class        Predicted Class (based on Probability)        Probability in Class1        Probability in Class2
         6                   2                      2                                             2                      0.34927                      0.65073

-----------------------------------------------------------

  Sample ID       Best Gamma          Best C
          6             0.04          139.69

-----------------------------------------------------------

  Sample ID       # K     KNN Index
          6        10      18 5 21 3 23 32 22 37 25 4

-----------------------------------------------------------

  Sample ID       # Features      Feature Index
          6        3               4 1 3
================================================================================ End ================================================================================
```

Figure 8.6: *The optimal sets of features, nearest neighbors, and model parameters for sample 6 alone, based on 50 testing runs.*

Figure 8.6 presents an example of the optimal sets of features, nearest neighbors, and model parameters obtained by svmGSA for sample 6 alone.

8.3.4 Summary

To conclude the gender studies, weather parameters seem to strongly affect female patients according to the classification accuracy. However, "wind speed" is selected as the global marker for individuals of both genders aged over 50 with a history of hypertension and smoke.

This chapter presents a pilot study using just two time points to evaluate the relationship between weather conditions and stroke occurrence. Experimental results show that weather conditions impact significantly on stroke occurrence, hence the data is worthy of further investigation as STD, using recently proposed EESNN and reSNN methods to learn the whole spatio-temporal pattern. Details of this investigation are introduced in the next chapter.

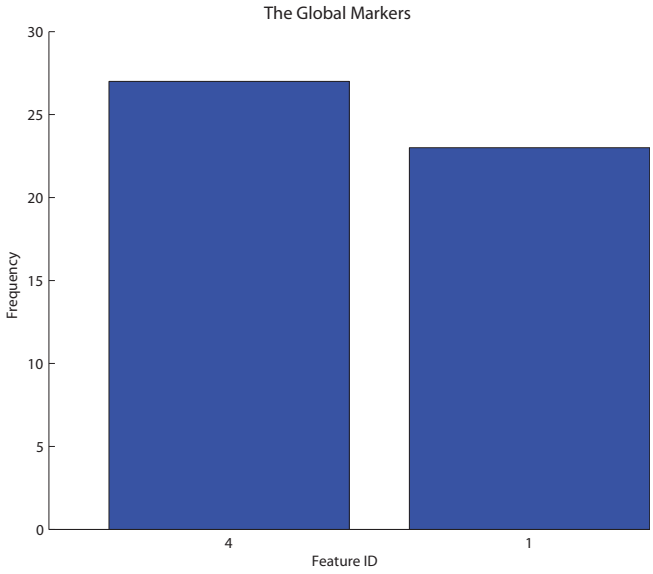


Figure 8.7: *The global markers of female group are computed based on the selecting frequency over all samples obtained using the svmGSA (G1=temperature and G4=wind speed).*

CHAPTER 9

Personalized Reservoir based Generic Method for Spatio-Temporal Weather and Stroke Occurrence Data Analysis

9.1 Introduction

As explained in Chapter 8, the weather and stroke occurrence dataset contains both temporal and spatial information. All weather parameters (temperature, humidity, wind speed, windchill and atmospheric pressure) are measured over time at different locations. To efficiently and effectively capture the whole STD pattern, rather than simply analyze the data by conventional statistical methods, the EESNN model (Hamed et al., 2011) and reSNN model (Schliebs et al., 2011) will be the first time applied to the spatio-temporal weather and stroke occurrence data.

This chapter begins with a pilot statistical analysis, followed by the application of EESNN model to the weather and stroke occurrence STD. The chapter concludes with the application of reSNN to the same data.

9.2 Statistical Analysis

Before applying the spatio-temporal weather and stroke occurrence data to reservoir based generic models, a statistical pilot study is conducted. Since this weather and stroke occurrence STD is here investigated for the first time, only a small group of patients are selected for analysis. We include data from the Auckland region only, focussing on the autumn season. The selected patients are aged from 60 to 69, with experience of hypertension and smoking. Recall that these have been identified as the important risk factors.

Based on the selection criteria, for triggering a stroke 20 patients were selected (see Figure 9.1). Since a case-crossover design is applied, 40 patients exist in the dataset, 20 in the "normal/control" group and 20 in the "stroke" group.

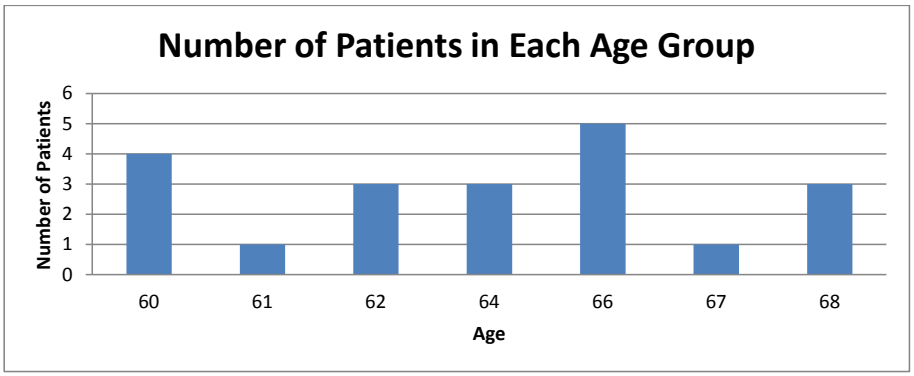


Figure 9.1: *Number of patients in each age group.*

As explained previously, all weather parameters are measured over a 60-day period, where the day of stroke occurrence is considered as day 0 and days -1 to -59 are the days prior to stroke occurrence. Thus, all weather changes over the 60-day period should be considered. To this end, we investigate weather changes for two age groups: 60 and 68.

Temperature Changes

Figure 9.2 illustrates the temperature changes over 60 days. Temperature changes smoothly 9 days before stroke occurrence for patients in 60-year old patients. In contrast, for patient ID 167 (age group 68), the temperature increases suddenly from day -25 to -24, and decreases gradually from day -9 to the day of stroke occurrence.

Other important knowledge abstracted from the Figure are that temperature remains almost stable from day -59 to -10 for patient ID 1600, but increases gradually from day -9 to the day of stroke occurrence. Thus, we could hypothesize that 9 days before stroke occurrence is an important stroke-triggering time window for both patient ID 167 and ID 1600.

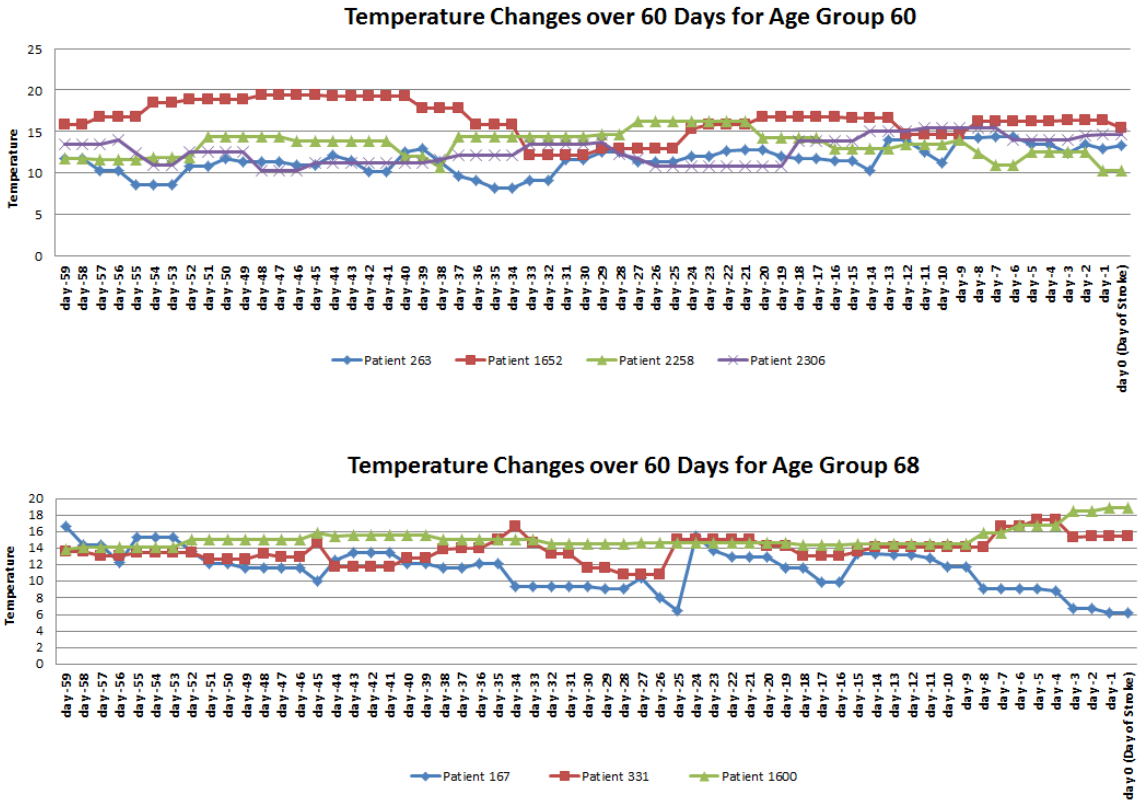


Figure 9.2: *The temperature changes over 60 days for patients aged 60 (top) and 68 (down).*

Humidity Changes

Figure 9.3 graphs the humidity changes to which patients from two age groups are exposed. Humidity appears to have no significant impact on most of the patients. However, for patient ID 1600, the humidity remains almost constant over 60 days, while for patient ID 2306 (age group 60), the humidity level suddenly increases from day -19 to -18.

Atmospheric Pressure Changes

As shown in Figure 9.4, the atmospheric pressure is quite changeable for some pa-

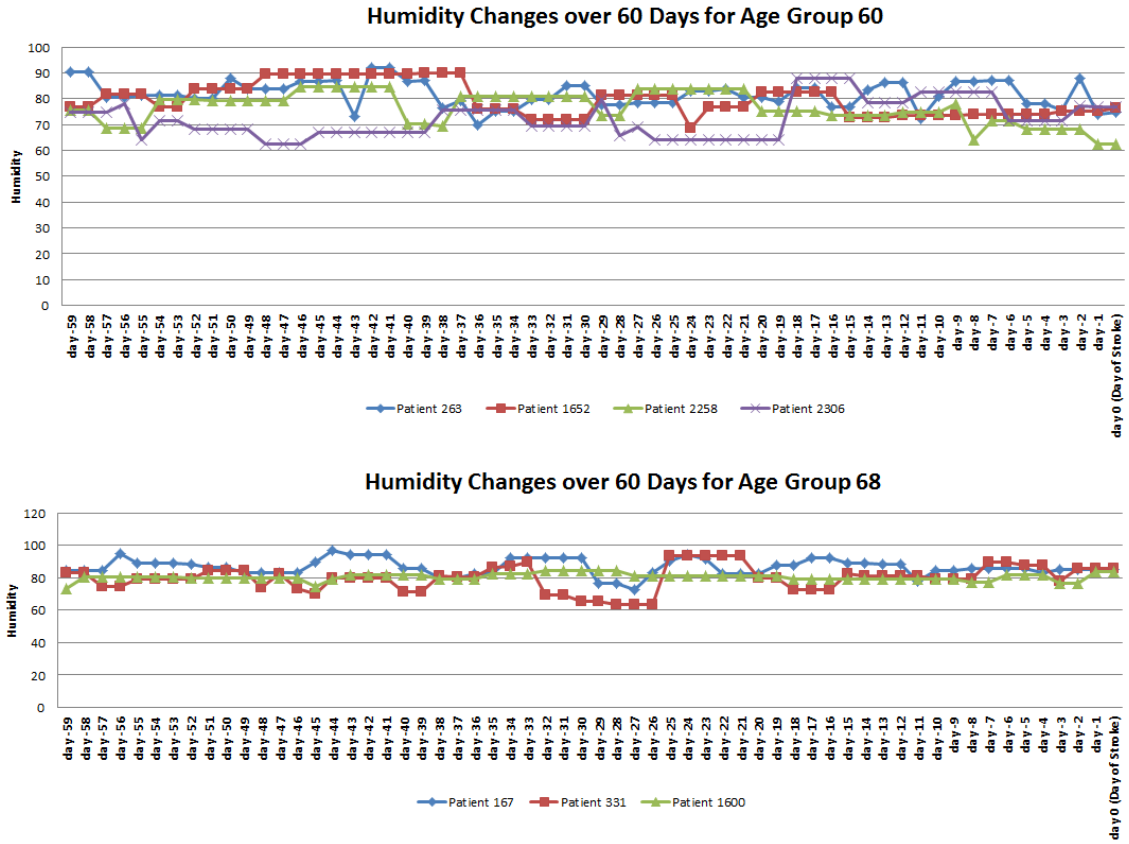


Figure 9.3: *The humidity changes over 60 days for patients aged 60 (top) and 68 (down).*

tients over the 60-day period. For instance, for patient ID 263, it increases suddenly from day -2 to the day of stroke occurrence, which might have trigged stroke in this patient. In addition, for patient ID 2258, the atmospheric pressure increases suddenly from day -58 to -57 and then varies slightly until the day of stroke occurrence. In the 68-year age group, the atmospheric pressure level for patient ID 331 changes significantly from day -41 to -32. The gradual decrease at the beginning of this period is followed by a sudden increase from day -33 to -32. Moreover, the atmospheric pressure drops dramatically from day -25 to -23 for patient ID 167, and then increases gradually from day -5 to the day of stroke occurrence. Patient ID 1600, whose stroke occurrence was found to be independent of humidity, is similarly insensitive to atmospheric pressure.

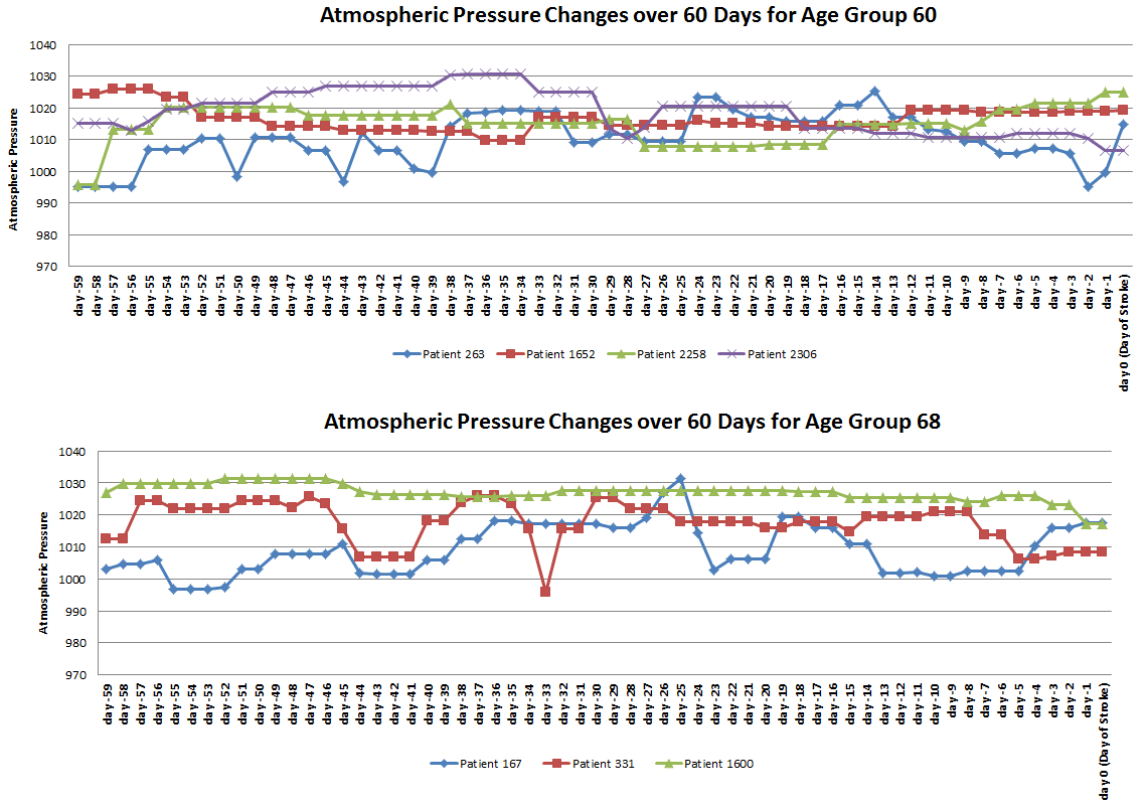


Figure 9.4: *The atmospheric pressure changes over 60 days for patients aged 60 (top) and 68 (down).*

Wind Speed Changes

Figure 9.5 illustrates the wind speed changes over 60 days. We note that wind speed is much more changeable than the previous weather parameters. Up/down variation

is frequent over the 60 days, specially for the patients from age group 60. According to the Figure, wind speed drops for all the patients a few days prior to stroke, suggesting that wind speed is an important stroke trigger. For example, the wind speed for patient ID 2258 drops sharply from day -9 to -7 and gradually declines until the day of stroke. For patient ID 263, the wind speed increases significantly from day -2 to -1 and drops dramatically on the day of stroke. Patients ID 331 and ID 167 are exposed to similar wind patterns 4 days before stroke: the wind speed drops significantly from day -4 to -3, and stabilizes until the day of stroke occurrence. Interestingly, the wind speed varies little for patient ID 1600, for this patient the wind speed is stable from day -59 to -7. Therefore, we can hypothesize that 6 days before stroke occurrence is an important stroke-triggering time window for patient ID 1600.

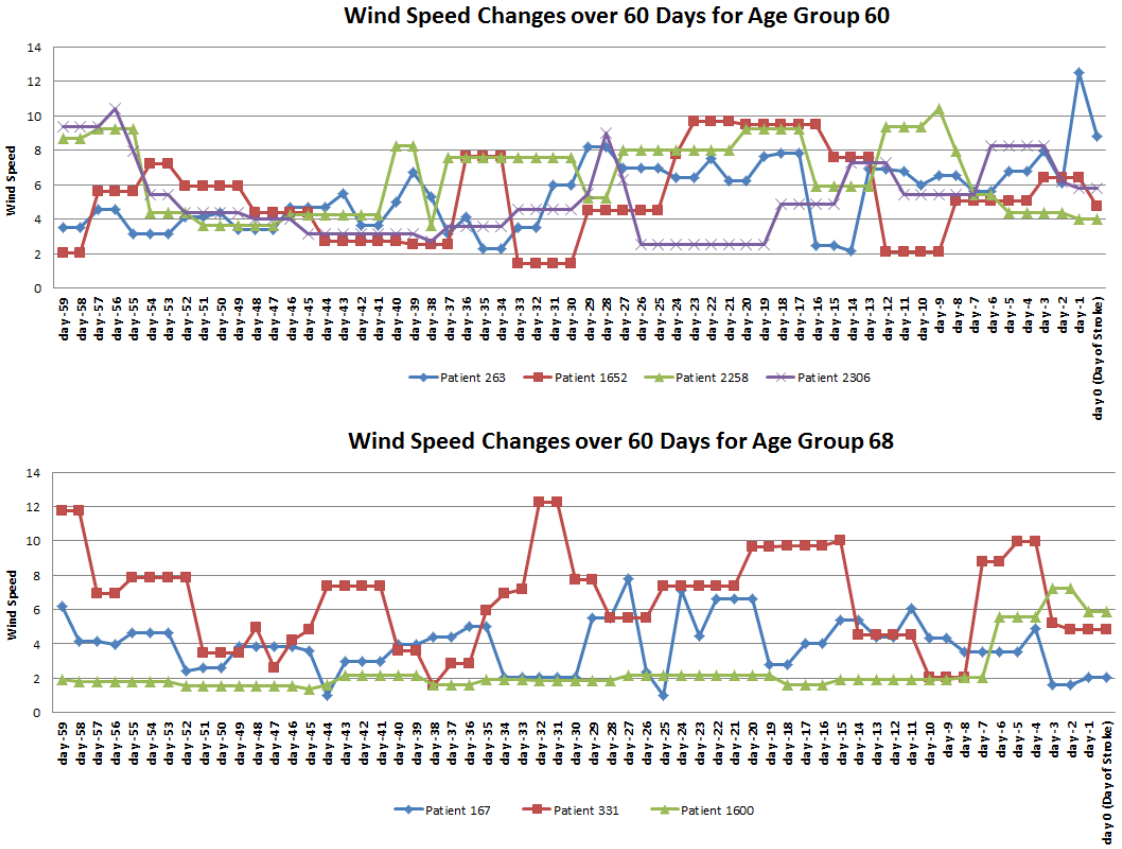


Figure 9.5: *The wind speed changes over 60 days for patients aged 60 (top) and 68 (down).*

Wind Chill Changes

Wind chill variation for both age groups is shown in Figure 9.6. This Figure shows that wind chill like wind speed, impacts strongly on patients. Among the 60-year group, the wind chill for patient ID 1652 decreases suddenly from day -37 to -36 and then stabilizes until the day of stroke. For patient ID 263, the wind chill is stable from day -59 to -20, but becomes variable from day -19 to the day of stroke, hence we identify 19 days before stroke occurrence exposure to varying wind chill as an important time window for triggering stroke in this patient. Moreover, for patient ID 2306, the wind chill changes little from day -59 to -30, but becomes more changeable from day -29 till the day of stroke occurrence, thus we hypothesize that 29 days before stroke occurrence might be an important stroke-triggering time window for this patient. Patients from age group 68 appear to be less affected by wind chill than the 60-year old patients. Especially for patient ID 1600, the wind chill remains stable from day -59 to -9. Other knowledge abstracted from the Figure is that patients ID 331 and ID 1600 are exposed to similar wind chill patterns 9 days before stroke occurrence, where wind chill gradually increases. In contrast, patient ID 167 experiences the opposite pattern, with the wind chill gradually decreasing 9 days prior to stroke.

To conclude this section, some new knowledge as regarding the weather patterns/stroke occurrence has been discovered:

- Wind speed emerges as the most significant stroke-triggering weather parameter followed by wind chill. Wind speed also presents as the global marker for both gender groups (see Chapter 9 for details). Thus, we can hypothesize that wind speed impacts strongly on stroke occurrence.
- Non of the weather parameters significantly impact on patient ID 1600 from day -59 to -10, since they are remains almost stable during this time period. However, the weather parameters for this patient change from day -9 to the day of stroke occurrence. Therefore, we can hypothesize that 9 days before stroke occurrence is an important stroke-triggering time window for patient ID 1600.
- Of all the patients, patient ID 263 is most obversely affected by the weather changes, especially since changes are more significant 17 days prior to stroke occurrence. Hence, we can hypothesize that 17 days before stroke is an important stroke-triggering time window for patient ID 263.

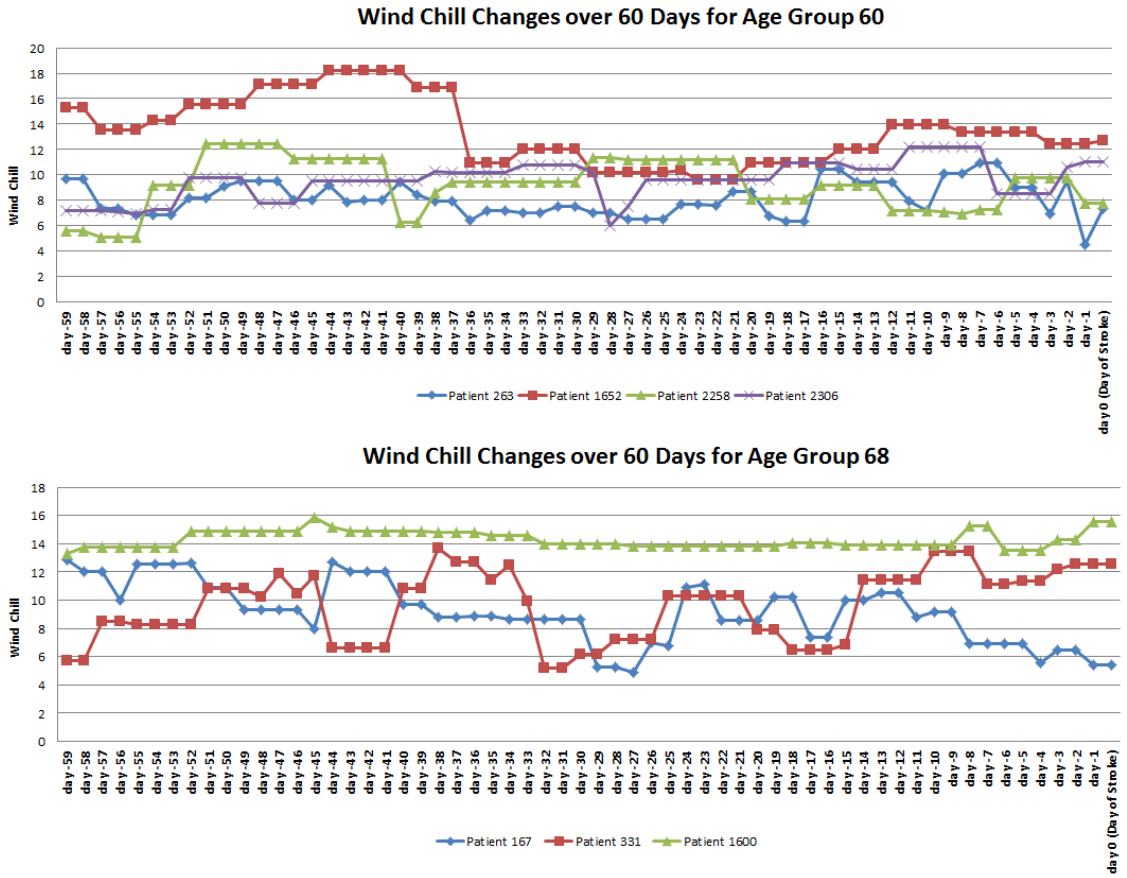


Figure 9.6: *The wind chill changes over 60 days for patients aged 60 (top) and 68 (down).*

In the following section, two studies are presented using two recently proposed generic methods EESNN and reSNN to learn the whole spatio-temporal pattern. The aim is to discover useful knowledge regarding the relationship between weather patterns and stroke occurrence.

9.3 Extended eSNN (EESNN) Performance Analysis

9.3.1 Setup

As explained in Chapter 3, the real-valued data must be encoded into spike trains using the population rank-order encoding scheme before it can be classified. However, prior to encoding, all weather parameters (temperature, humidity, wind speed, windchill and atmospheric pressure) are normalized individually in the interval [0,1], to account for their different units.

Figure 9.7 demonstrates how a single input value (e.g. temperature 20^o) is encoded into multiple neurons. Each neuron is encoded into a specific spike train calculated from the intersection of Gaussian functions. All data are encoded in the same way. In this study, each single input value is encoded into 40 receptive fields with β 1.5 (where β is the width of each Gaussian receptive field).

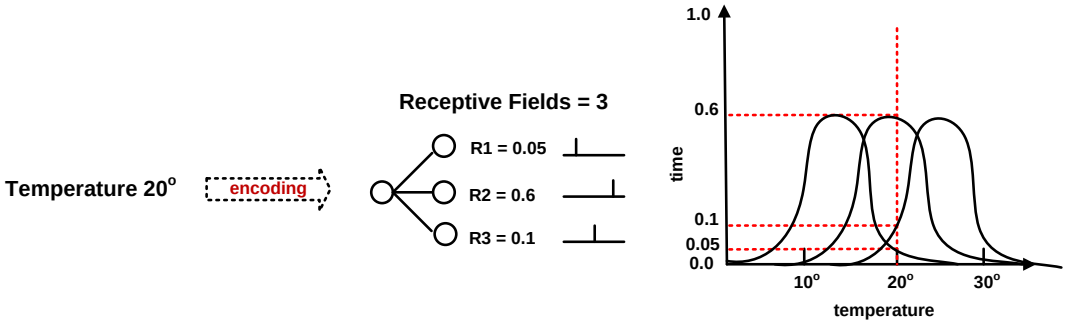


Figure 9.7: *Demonstration of a single input value (e.g. temperature 20^o) encoded into spike trains.*

Once all of the data have been encoded into spike trains, they are passed to the first layer of EESNN, which acts as a memory to capture the whole STD pattern.

Next, they are passed to eSNN for classification. eSNN contains three parameters, *Modulation factor (Mod)* of the Thorpe neural model, *Threshold/Proportion factor (C)* which dictates the percentage of the maximum post-synaptic potential (PSP) to be used for firing an output spike, and *Neuron Similarity (Sim)* which controls the similarity distance. If a certain neuron is considered too similar to others, it will be merged with the most similar existing neuron. All parameters are within the interval [0,1] (see Chapter 3 for details).

In this preliminary study, all parameters are manually as follows:

- *Mod* is defined as 0.9, since the spike trains are passed to EESNN time-sequentially, from day 0 (day of stroke occurrence) to -59. Because *Mod* is set to a high value, the first few days will tend to spike earlier than the later days. This scenario reflects the hypothesis that early days before stroke occurrence are important time windows for patients to develop stroke.
- *Sim* is set as 0.0, since we desire that every neuron produces a output.
- *C* is studied in the range between 0.1 to 1, incremented by 0.01 each time. From Figure 9.8, we observe that 0.32 is the most frequently selected threshold value.

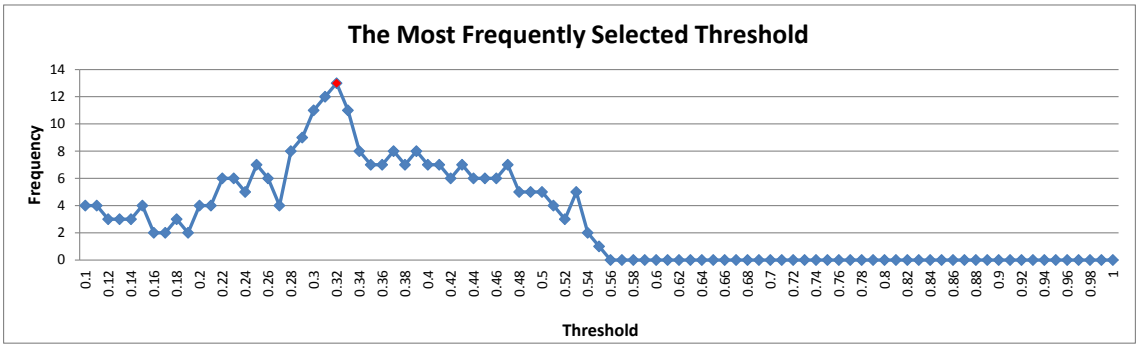


Figure 9.8: *The most frequently selected threshold.*

9.3.2 Result

As a pilot study prior to investigation by EESNN, the data are evaluated by conventional KNN. The best accuracy achieved by KNN (k=9) is 60% (50%-class Normal

& 70%-class Stroke). This almost random result arises because STD cannot be well learned by the conventional method. Thus, we assume that EESNN, which is especially designed for STD learning will provide higher accuracy than conventional method.

Figure 9.9 presents the classification accuracy using different C values (with *Mod=0.9, Sim=0.0*), where c=0.32 provides the best performance at 70% (65%-class Normal & 75%-class Stroke). This accuracy is 10% higher than that achieved by KNN. Therefore, we could say that the first 20 days presents an important stroke-triggering time window.

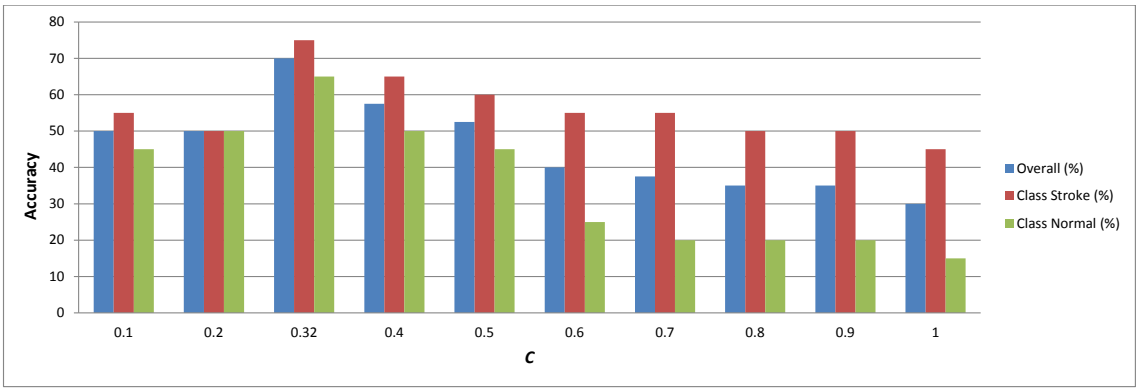


Figure 9.9: *The classification performance achieved by EESNN for different threshold/proportion factor (C).*

Having found the optimal or near-optimal threshold, the classification accuracy can be further evaluated by varying the *Mod* value. As shown in Figure 9.10, the overall accuracy and the accuracy of stroke class gradually increase as *Mod* increases. Thus, we can confidently predict that early days exert more impact on stoke occurrence than later days.

9.4 Recurrent Network Reservoir Structure (reSNN) Performance Analysis

9.4.1 Setup

Similar to the EESNN model, in the first step, each real-value of spatio-temporal data vector must be encoded into a spike train using the population rank-order encoding approach. However, unlike the previous encoding a threshold is added to eliminate all very late spikes. If the spike is less than the pre-determined threshold, it will not be used for the PSP computation. In this way, only a few earlier spikes carrying most of the information are used, boosting the accuracy and reducing the computation time.

Figure 9.11 demonstrates an example of spike trains after population encoding, with 40 receptive fields and β 1.5. The total number of neurons are 200 (40x5). Encoding reveals clear differences between sample 1 (Normal) and sample 40 (Stroke).

In this experiment, the LSM reservoir is constructed as a three dimensional network of grid size 5x5x10. The simulation time is set at 500 milliseconds. The reservoir responses are sampled using a time step of 10 milliseconds during analog readout process (see Chapter 3), hence the final liquid states are sampled in a series of 50 time intervals.

9.4.2 Result

Figure 9.12 shows typical reservoir responses to sample 1 (Normal) and sample 40 (Stroke). The obvious differences between the samples indicate the high separability capability of the reservoir.

In this study, time point t=50, 100, 150, 200, 250, 300, 350, 400, 450, and 500 milliseconds spaced at 50 time intervals, are abstracted and passed to the classifier sequentially to learn the responses from the reservoir. To adequately compare the different methods, I have a applied *global modeling* method (SVM); a *local modeling* method (ECF); and *evoPM*-based methods (svmGSA and esnnGSA). The perform-

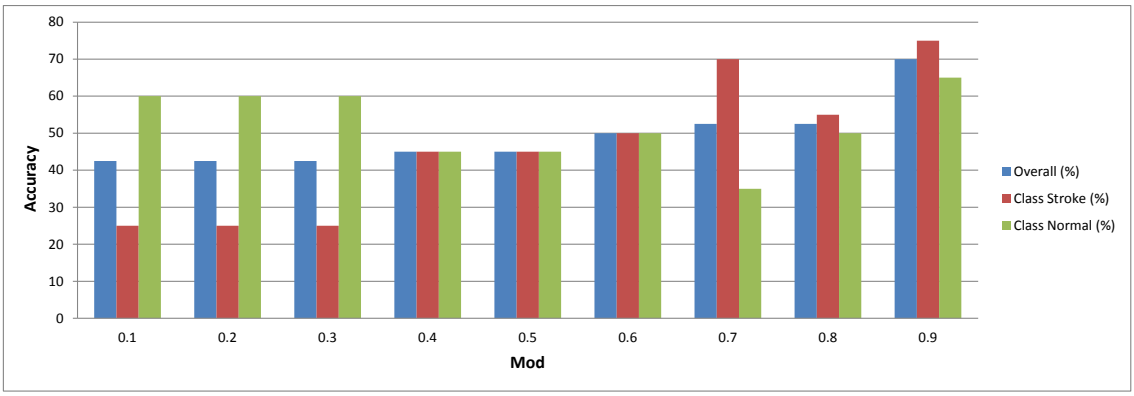
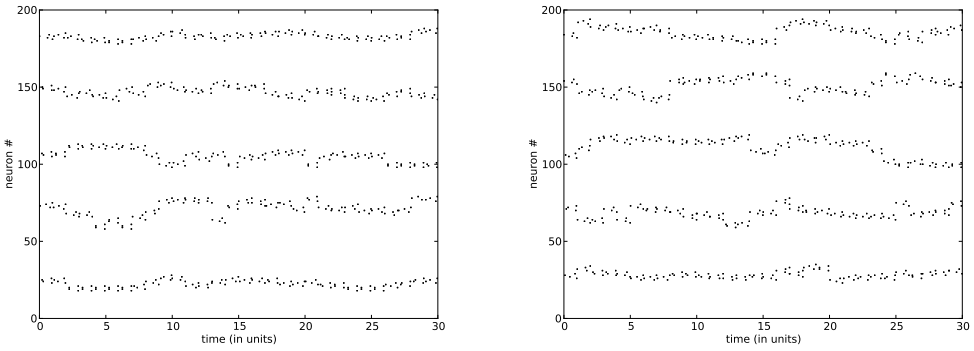
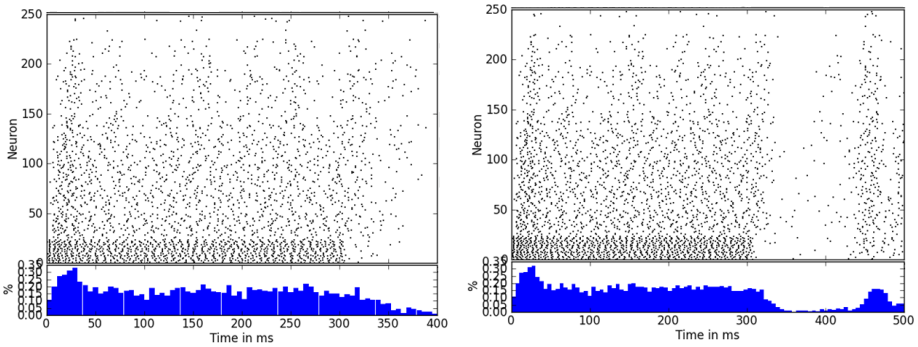


Figure 9.10: *The classification performance achieved by different modulation factor (Mod) value.*



(a) "sample 1" after encoding (b) "sample 40" after encoding

Figure 9.11: *Comparison of two samples from different class after encoding.*



(a) Reservoir response of "sample 1" (b) Reservoir response of "sample 40"

Figure 9.12: *The reservoir responses of two samples from different classes.*

Table 9.1: *Classification accuracy of different models, tested at the time point* $t = 200$ *milliseconds.*

Experimental Results		
Classifier	Overall Acc(%)	Class(1/2)(%)
SVM (Linear Kernal, gamma=1)	75.00	(85.00/65.00)
ECF	72.50	(80.00/65.00)
esnnGSA(Mod=0.5865, Threshold=0.2565, Sim=0.256, k=7 Ave)	**77.50**	(60.00/95.00)
svmGSA(gamma=0.84, c=44.65, k=5 Ave)	72.50	(75.00/70.00)
Note: The parameters are selected through the same optimization process if they are employed in evoPM models. The parameters in SVM and ECF are selected based on the best classification performance. For the global SVM parameters, only the parameter γ is tuned.		

ance of the methods used in all experiments is evaluated by LOOCV. Irrelevant features are filtered out by a signal-to-noise-ratio (SNR).

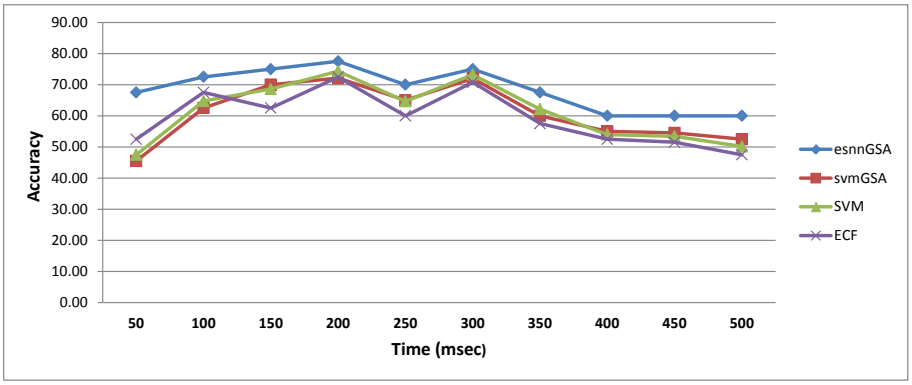


Figure 9.13: *The classification performance of each selected time point.*

The results at each time point obtained by the four methods are shown in Figure 9.13. The esnnGSA performs more accurately than the conventional methods across the entire simulation, followed by the svmGSA method. esnnGSA attains its highest accuracy at the time point $t = 200$ milliseconds (overall accuracy is 77.50%: 60.00% for class 1 - Normal/Control, and 95.00% for class 2 - Stroke). Table 9.1 summarizes all parameters that used for the best experiment. This suggests that the reservoir provides the best liquid state for distinguishing between output classes at this time point. However, the time points mentioned here are not related to the

real-time points, but are merely responses from the reservoir. Future studies should further implement the reSNN method, aiming to link the real-time points with the reservoir responses. In such a way, we could accurately discover which time window significantly impacts on stroke occurrence.

9.5 Summary

In the first study, the EESNN method is superior to the conventional KNN method in terms of classification accuracy. Particularly, we find that the first 20 days might be an important time window for stroke onset. However, in this study, all the parameters are manually adjusted. In future studies, EESNN will be further implemented to improve the robustness and generalisability of parameter optimization.

In the reSNN study, several time points are selected and fed into evoPM as a classifier to learn the reservoir responses. All methods provide the similar accuracy over the simulation period. The classification accuracy decreases as the time window enlarges due to fewer activities produced by the reservoir.

CHAPTER 10

Conclusions and Future Directions

The concept of personalized modeling is rooted in machine learning technologies that have been successfully utilized for understanding, evaluating and solving a variety of modeling problems. Fields that benefit from personalized modeling include personalized medicine and drug design, business, finance, and crime prevention. However, personalized modeling is not without problems, defining the correct number of neighbors and model parameters and an appropriate number of features remain a challenge. The goal of this research is to study and address these issues by creating a novel framework and system for personalized modeling that allows users to select and optimize the most important features, nearest neighbors and model parameters. The system promises more precise classification accuracy and personalized knowledge than standard global and local modeling approaches.

In brief, this thesis has presented the following main contributions for personalized modeling study:

1. Chapter 6 proposes the novel integrated evolving personalized modeling systems (evoPM), in which a recently developed population-based heuristic optimization approach termed gravitational search algorithm (GSA) is applied for feature selection, neighborhood and model parameters optimization. The evoPM can create a personalized model for each testing sample with its own optimal sets of features, neighborhood and model parameters.

This study has investigated a variety of classification methods during the development of evoPM, including KNN and SVM. In particular, a new technology evolving spiking neural networks (eSNN) is utilized in a novel way. Another novelty is that personalized risk is evaluated for individual patient, rather than generically classifying patients into normal or diseased groups. Accurately quantifying disease risk is critical for medical decision support to ensure that patients receive the optimal treatment for their individual profile;

2. To verify the strength of the novel method, it is applied to several benchmark cancer gene expression datasets, its performance is compared with that of traditional global, local and personalized modeling methods. evoPM consistently provides a more promising performance than traditional methods because it selects the optimal sets of genes and disease classification parameters for each individual patient (as detailed in Chapter 7). It discovers more useful knowledge for medical decision support in cancer diagnosis and prognosis;

3. The third novelty of this study is the first-time testing of the proposed method on stroke data as case studies. Chapter 5 presents a comparative study applying the conventional global, local, classical personalized modeling methods, and evoPM-based algorithms to stroke outcome prognosis data as a case study. Due to the limitation of the study, this study is conducted as a pilot study only to find the predictors of depression in 5-year stroke survivors. The evoPM-based methods were superior to the traditional methods in terms of classification accuracy. In addition, the system creates a personalized profile for individual patient and identifies the global markers computed from the selecting frequency over all samples;

4. Focussing on multivariate spatio-temproal data (STD) analysis, Chapter 3 introduces two recently proposed methods for spatio-temporal pattern recognition, namely the extended eSNN framework (EESNN) and the recurrent network reservoir structure of eSNN (reSNN) using Liquid State Machine (LSM). These two individualized generic prognostic models are for the first time applied to stroke risk of occurrence spatio-temporal data as another case study (See Chapter 9). The results show that personalized generic models are developed successfully for learning STD, as such, they make significant contributions to new knowledge and provide higher accuracy for predicting an individualized event than traditional prognostic models.

More specifically, the proposed personalized modeling system is the framework and system that integrates novel machine learning and modeling techniques for the following research problems:

- Develop a novel integrated evolving personalized modeling system using incrementally new data from various sources;
- Data sample profiling and results visualization;
- Estimate personalized risk;
- Encode the real-valued data into spike trains prior to feed into a spatio-temporal filter to accumulate the spatio-temporal information of all input signals into a single high-dimensional state;
- Knowledge discovery and model validation;
- Optimal set of features, neighborhoods, model and its parameters selection.

10.1 Future Directions

As mentioned above, personalized modeling promises better results in both static data analysis and dynamic STD analysis than global and local modeling methods. Thus, it is worthy of investigating further to generate new knowledge for enhanced understanding of complex phenomena occurring in nature and in human health. Suggestions for future work are bulleted below:

- In this study, a new optimization approach GSA is applied for feature selection, neighborhood and model parameter optimization. In a future study, several more evolutionary methods will be integrated with personalized modeling approach and evaluated for optimization, such as particle swarm optimization (PSO) (Kennedy & Eberhart, 1995) and cuckoo Search (CS) (X. S. Yang & Deb, 2009), etc.
- evoPM in this study adopts three classification methods, namely KNN, SVM and eSNN. In a future study, new technology will be studied and investigated,

such as probabilistic SNN (pSNN). Proposed by Kasabov (Kasabov, 2007a, 2007b), pSNN stores its information as both connection weights and probabilistic parameters under which spikes occur and propagate.

As hypothesized by Kasabov (Kasabov, 2010), a probabilistic connection between two neurons might enhance the computational power of SNN, leading to new pSNN. In addition, pSNN might provide probabilistic risk estimates that assist doctors in providing optimal prognosis and treatment to their patients.

- In two stroke case studies, select cases only are investigated as preliminary studies. The study limitations preclude an analysis of large populations. Therefore, in future:

  1. More stroke outcomes will be incorporated into stroke outcome prognosis analysis, including memory, executive function, and information processing. Such studies will benefit long-term evidence-based rehabilitation and service planning, thus improving health outcomes in stroke.
  2. The entire population from all six regions (Auckland (NZ), Perth and Melbourne (Australia), Oxfordshire (UK), Dijon (France), Norrbotten and Vasterbotten (Northern Sweden)) will be incorporated into stroke risk of occurrence analysis. Especially, a comparative study will be conducted to investigate the differences and similarities between different regions. We hypothesize that the study will contribute significantly to understanding of environmental triggers of stroke, as well as reducing the hazardous effects of harmful weather conditions on other diseases.

- To improve the efficiency of the generic personalized modeling for dynamic STD learning, EESNN and reSNN will be further implemented by integrating with evolutionary methods, such as GSA and PSO. We hypothesize that optimized personalized methods will generate more precise results and new discoveries.

- The "Brain-gene ontology (BGO)" model is used to characterize human brain, genes, and the relationships between them (Benuskova & Kasabov, 2006). In a future study, we will develop a Brain Injury Ontology (BIONT) repository to store all data, information and knowledge of brain injury, such as stroke data. BIONT will allow users to navigate and find genes expressed in different parts of the brain, or to study unknown interactions between variables related to any brain injury disease outcome and risk.

In addition, the BIONT and the existing personalized modeling framework and system will be combined into a Knowledge Engineering System (KESBI) (as shown in Figure 10.1). This system will support new knowledge discovery that facilitates understanding of the complex interactions occurring in the brain. In this way, we can predict the best possible outcome for a new patient, as well as provide more accurate diagnosis and prognosis of clinical results.

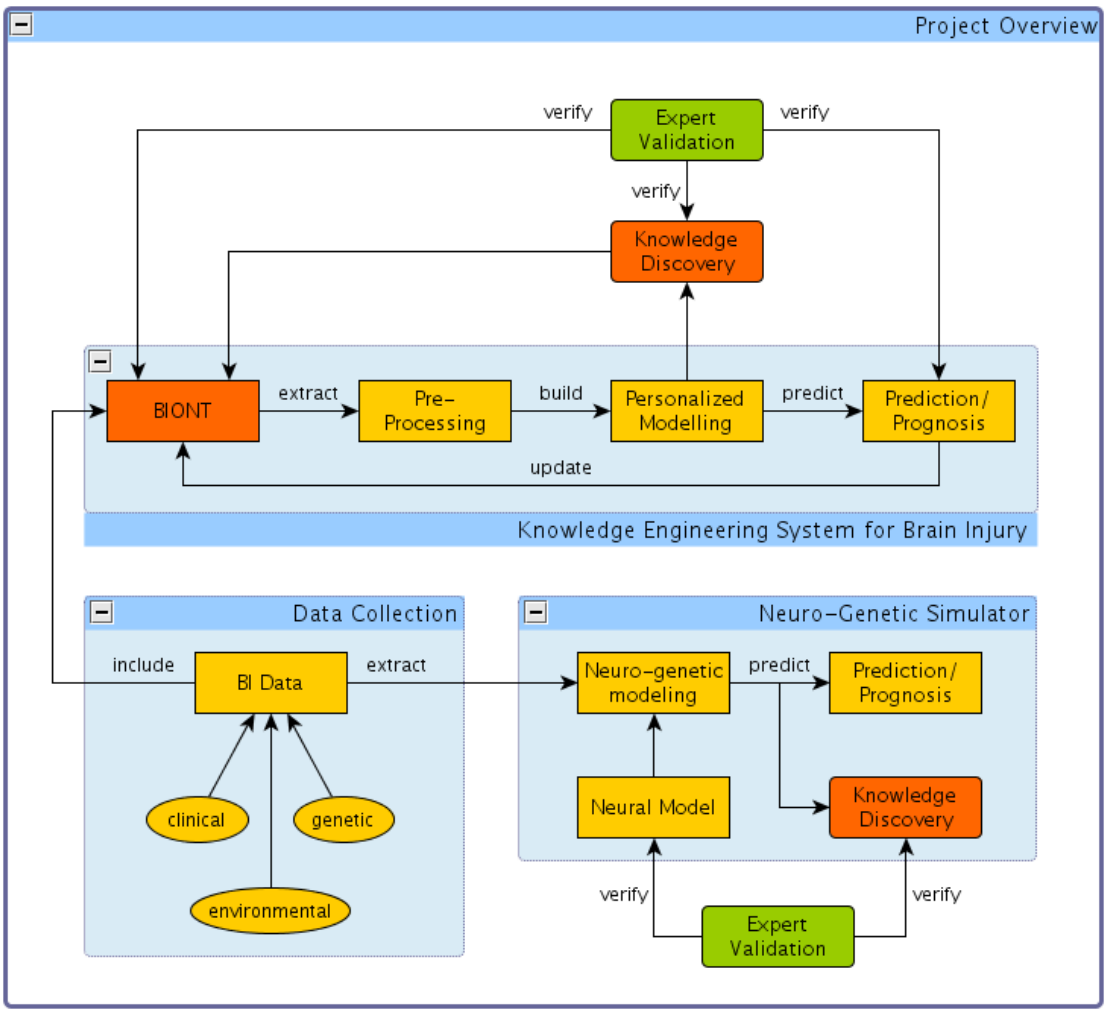


Figure 10.1: *Flowchart of the proposed novel integrated Knowledge Engineering System (KESBI).*

References

Abrahams, E. (2007). Personalized medicine: The changing landscape of healthcare. 2

Adam, G. (2005). *Introduction to neural networks.* Retrieved from `http://home.agh.edu.pl/~vlsi/AI/intro/` ix, 37

Aho, K., Harmsen, P., Hatano, S., Marquardsen, J., Smirnov, V. E. & Strasser, T. (1980). Cerebrovascular disease in the community: Results of a WHO collaborative study. *Bull World Health Organ*, *58*(1), 113-130. 72

Aizerman, E. M. & Braverman, L. R. (1964). Theoretical foundations of the potential function method in pattern recognition learning. *Automat Remote Control*, *25*, 821-837. 13

Alcatel-Lucent. (2012). *Alcatel lucent enables belgacom to deliver personalized mobile broadband service plans, giving customers more control over their spending.* Alcatel Lucent Press. Retrieved from `http://www.alcatel-lucent.com/wps/portal/!ut/p/kcxml/04_Sj9SPykssy0xPLMnMz0vM0Y_QjzKLd4w3MfQFSYGYRq6m-pEoYgbxjgiRIH1vfV-P_NxU_QD9gtzQiHJHR0UAAD_zXg!!/delta/base64xml/L0lJayEvUUd3QndJQSEvNElVRkNBISEvNl9BX0U4QS9lbl93dw!!?LMSG_CABINET=Docs_and_Resource_Ctr&LMSG_CONTENT_FILE=News_Releases_2012/News_Article_002672.xml` 2

Allison, D. & Cui, X. (2006). Microarray data analysis: from disarray to consolidation and consensus. *Nature Reviews Genetics*, *7*(1), 55-65. 26

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S. & Mack, D. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. In *National Academy of Sciences of the United States of America* (Vol. 96, p. 6745-6750). 111

Anderson, C. S., Jamrozik, K. D., Broadhurst, R. J. & Stewart, E. G. (1994). Predicting survival for 1 year among different subtypes of stroke. Results from the Perth community stroke study. *Stroke*, *25*(10), 1935-1944. 73

Anderson, C. S., Linto, J. & Stewart, E. G. (1995). A population based assessment of the impact and burden of caregiving for long-term stroke survivors. *Stroke*, *26*(5), 843-849. 73

Anderson, J. E., Hansen, L., Mooren, C., Post, M., Hug, H., Zuse, A. & Los, M.

(2006). Methods and biomarkers for the diagnosis and prognosis of cancer and other diseases: Towards personalized medicine. *Drug Resistance Updates*, *9*(4-5), 198-210. 3

Angluin, D. & Smith, C. H. (1983). Inductive inference: Theory and methods. *Computing Surveys*, *15*(3), 237-269. 10

Anita, T., Bhanot, S. & Mishra, S. N. (2009). Early diagnosis of ischemia stroke using neural network. In *2009 Proceedings of the International Conference on Man-Machine Systems (ICOMMS).* Batu Ferringhi, Penang, Malaysia. 78

Arbib, M. (2003). *The handbook of brain theory and neural networks.* Cambridge: MIT Press. 16

Back, T. (1996). *Evolutionary algorithms in theory and practice: Evolution strategies, evolutionary programming, genetic algorithms.* New York: Oxford University Press. 57

Bahrololoum, A., Nezamabadi pour, H., Bahrololoum, H. & Saeed, M. (2012). A prototype classifier based on gravitational search algorithm. *Applied Soft Computing*, *12*(2), 819-825. 68

Bartfay, E., Mackillop, W. J. & Pater, J. (2006). Comparing the predictive value of neural network models to logistic regression models on the risk of death for small-cell lung cancer patients. *European Journal of Cancer Care*, *15*, 115-124. 45

Beer, D. G., Kardia, S. L., Huang, C. C., Giordano, T. J., Levin, A. M., Misek, D. E., ... Hanash, S. (2002). Gene-expression profiles predict survival of patients with lung adenocarcinoma. *Nat Med*, *8*, 816-824. 108

Benuskova, L. & Kasabov, N. (2006). *Computational neurogenetic modeling.* New York, USA: Springer Verlag. 153

Berg, S. (2007). *Gene expression.* Retrieved from `http://course1.winona.edu/sberg/241f08/Lec-note/Prot-Syn.htm` xi, 107

Bi, G. & Poo, M. (2001). Synaptic modification by correlated activity: Hebb's postulate revisited. *Annual review of neuroscience*, *24*(1), 139-166. 43

Biller, J., Jones, M. P., Bruno, A., Adams, H. P. & Banwart, K. (1988). Seasonal variation of stroke: Does it exist? *Neuroepidemiology*, *7*, 89-98. 123

Biology-Corner. (n.d.). *DNA and RNA.* The Biology Corner. Retrieved from `http://www.biologycorner.com/bio1/DNA.html` xi, 106

Blum, C. & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptural comparison. *ACM Comput. Surv.*, *35*, 268-308. 33

Bohte, S. M., Kok, J. N. & Poutre, J. A. L. (2002). Error backpropagation in temporally encoded networks of spiking neurons. *Neurocomputing*, *48*(1-4), 17-37. 47

Bonita, R., Anderson, C. S., Broad, J. B., Jamrozik, K. D., Stewart, E. G. & Anderson, N. E. (1994). Stroke incidence and case fatality in Australasia. A comparison of the auckland and perth population-based stroke registers. *Stroke*, *25*(3), 552-557. 73

Bonita, R., Broad, J. B. & Beaglehole, R. (1993). Changes in stroke incidence and case-fatality in Auckland, New Zealand, 1981-91. *Lancet*, *342*(8885), 1470-1473. 73

Bonita, R., Ford, M. A. & Stewart, A. W. (1988). Predicting survival after stroke: A three-year follow-up. *Stroke*, *19*(6), 669-673. 73

Bosnic, Z. & Kononenko, I. (2003). Evaluation of prediction reliability in regression using the transduction principle. *Computer as A Tool*, *8*(2), 99-103. 11

BragaNeto, U. & Hashimoto, R. (2004). Is cross-validation better than resubstitution for ranking genes? *Bioinformatics*, *20*(2), 253-258. 26

Caro, J. J., Huybrechts, K. F. & Duchesne, I. (2000). Management patterns and costs of acute ischemic stroke: An international study. *Stroke*, *31*(3), 582-590. 72, 73

Castellano, S. (n.d.). *Genes and Genomes.* Retrieved from `http://genome.crg.es/courses/Madrid04/exercises/ensembl/index.html` xi, 105

Chang, C. C. & Lin, C. J. (2011). Libsvm : A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, *2*(3), 27:1-27. 96

Chen, M., Han, J. & Yu, P. (1996). Data mining: An overview from database perspective. *IEEE Transactions on Knowledge and Data Engineering*, *8*(6), 866-883. 24

Chen, T. Y. & Cheng, Y. L. (2008). Global optimization using hybrid approach. *WSEAS Transactions on Mathematics*, *7*(6), 254-262. 33

Chen, Z. Y., Chang, S. F. & Su, C. L. (1995). Weather and stroke in a subtropical area: Ilan, taiwan. *Stroke*, *26*, 569-572. 123, 124

Chris, W. (n.d.). *Lymphoma-symptoms, treatments, and therapies.* Retrieved from `http://www.canceractive.com/cancer-active-page-link.aspx?n=164` 117

Cornelissen, G., Halberg, F., Breus, T., Syutkina, V., Baevsky, R., Weydahl, A., ... Bakken, E. (2002). Non-photic solar associations of heart rate variability and

myocardial infarction. *Journal of Atmospheric and Solar Terrestrial Physics*, *64*(5-6), 707-720. 3

Cowburn, P. J., Cleland, J. G. F., Coats, A. J. S. & Komajda, M. (1996). Risk stratification in chronic heart failure. *Eur Heart J*, *19*, 696-710. 45

Craven, P. & Wahba, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, *31*(4), 377-403. 28

Darwin, C. (1859). *The origin of species by means of natural selection.* London: John Murray. 56

DeJong, K. (1980). Adaptive system design: A genetic approach. *IEEE Transactions on Systems Man and Cybernetics*, *10*, 566-574. 59

Deloitte. (2012). *Targeted therapies: Navigating the business challenges of personalized medicine.* Washington, DC. 2

Destexhe, A. & Contreras, D. (2006). Neuronal computations with stochastic network states. *Science*, *314*(5796), 85-90. 44

Di, C. A., Lamassa, M., Baldereschi, M., Pracucci, G., Basile, A. M., Wolfe, C. D., ... Inzitari, D. (2003). Sex differences in the clinical presentation, resource use, and 3 month outcome of acure stroke in europe: Data from a multicenter multinational hospital based registry. *Stroke*, *34*, 1114-1119. 129

Donnell, M., Xavier, D., Liu, L., Zhang, H., Chin, S., RaoMelacini, P., ... Yusuf, S. (2010). Risk factors for ischaemic and intracerebral haemorrhagic stroke in 22 countries: A case control study. *The Lancet Neurology*, *376*(9735), 112-123. 1

Dorigo, M. (1992). *Optimization, learning and natural algorithms.* PhD Thesis. 61

Dudani, S. A. (1976). The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on System, Man and Cybernetics*, 325-327. 18

Eisen, M., Spellman, P., Brown, P. & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. In *National Academy of Science* (Vol. 95, p. 14863-14868). Stanford, CA. 110

Elman, J. L. (1990). Finding structure in time. *Cognitive Sci.*, *14*, 179-211. 50

Eng, J. (2005). Receiver operating characteristic analysis: A primer. *Acad Radiol*, *12*, 909-916. 30

Farmer, J. D., Packard, N. H. & Perelson, A. S. (1986). The immune system, adaptation and machine learning. *Physica D*, *2*, 187-204. 60

Fayyad, U. M. & Grinstein, G. G. (2001). *Information visualization in data mining and knowledge discovery.* Los altos, CA: Morgan Kaufmann. 50

Feigin, V. (2004). *When lightning strokes: An illustrated guide to stroke prevention and recovery.* Auckland, NZ: HarperCollins. x, 75

Feigin, V. L., Barker-Collo, S., Parag, V., Lawes, C. M. M., Ratnasabapathy, Y. & Glen, E. (2010). Auckland stroke outcomes study. part 1: Gender, stroke types, ethnicity, and functional outcomes 5 years poststroke. *Neurology*, *75*(18), 1597-1607. 77

Feigin, V. L., Carter, K., Hackett, M., Barber, P., McNaughton, H. & Dyall, L. (2006). Ethnic disparities in incidence of stroke subtypes: Auckland regional community stroke study, 2002-2003. *The Lancet Neurology*, *5*(2), 130-139. 77

Feigin, V. L., Lawes, C., Bennett, D. & Anderson, C. (2003). Stroke epidemiology: A review of population based studies of incidence, prevalence, and case fatality in the late 20th century. *The Lancet Neurology*, *2*(1), 43-53. 3

Feigin, V. L., Lawes, C., Bennett, D., Barker, S. & Varsha, P. (2009). Worldwide stroke incidence and early case fatality reported in 56 population based studies: A systematic review. *The Lancet Neurology*, *8*(4), 355-369. 3

Feigin, V. L., Nikitin, Y. & Vinogradova, T. (1997). Solar and geomagnetic activities: Are there associations with stroke occurrence? *Cerebrovascular Diseases*, *7*(6), 345-348. 3

Feigin, V. L., Nikitin, Y. P., Bots, M. L., Vinogradova, T. E. & Grobbee, D. E. (2000). A population based study of the associations of stroke occurrence with weather parameters in siberia, Russia (1982-1992). *Eur J Neurol*, *7*, 171-178. 123, 124

Ferri, F. J., Pudil, P., Hatef, M. & Kittler, J. (1994). *Comparative study of techniques for large scale feature selection.* Amsterdam: Springer Verlag. 24

Fix, E. & Hodges, J. L. (1951). *Discriminatory analysis: Nonparametric discrimination: Consistency properties.* Randolph Field, Texas: UASF School of Aviation Medicine. 18

Fogel, L. J. (1999). *Evolutionary computation: Toward a new philosophy of machine intelligence* (2nd ed.). Piscataway, NJ: IEEE Press. 58

Fogel, L. J., Owens, A. J. & Walsh, M. J. (1966). *Artificial intelligence through simulated evolution.* New York: Wiley. 57, 59

Fonseca, C. M. & Fleming, P. J. (1993). Genetic algorithms for multiobjective optimization: formulation, discussion and generalization. In *5th International Conference on Genetic Algorithms* (p. 416-423). San Mateo, CA: Morgan Kaufmann. 59

Fung, G. M. & Mangasarian, O. L. (2004). A feature selection Newton method for support vector machine classification. *Computational Optimization and Applications*, *28*(2), 185-202. 13

Fyfe, C., Barbakh, W., Ooi, W. C. & Ko, H. (2008). Topological mappings of video and audio data. *International Journal of Neural Systems*, *18*(6), 481-489. 45

Gant, V., Rodway, S. & Wyatt, J. (2001). *Artificial Neural Networks: Practical considerations for clinical applications in Dybowski R and Gant V clinical applications of neural networks*. UK, Cambridge: Cambridge University Press. 45

Gerstner, W. (1995). Time structure of the activity of neural network models. *Physical Review*, *51*, 738-758. 41

Gerstner, W. (2000). Population dynamics of spiking neurons: Fast transients, asynchronous states, and locking. *Neural Comput.*, *12*(1), 43-89. 39

Gerstner, W. & Kistler, W. (2002b). *Spiking neuron models: An introduction*. New York, USA: Cambridge University Press. 43

Gerstner, W. & Kistler, W. M. (2002a). *Neuron models: Single neurons, populations, plasticity*. Cambridge, MA: Cambridge Univ. Press. 41

Ghalambaz, M., Noghrehabadi, A. R., Behrang, M. A., Assareh, E., Ghanbarzadeh, A. & Hedayat, N. (2011). A hybrid neural network and gravitational search algorithm (HNNGSA) method to solve well known Wessinger's equation. *World Academy of Science, Engineering and Technology*, *73*, 803-807. 67

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M. & Mersirov, J. P. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, *286*, 531-537. 111

Gordon, G. J., Jensen, R., Hsiao, L. L., Hsiaox, S. & JE, B. (2002). Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma. *Cancer Research*, *62*, 4963-4967. 111

Goto, T., Baba, T., Ito, A., Maekawa, K. & Koshiji, T. (2007). Gender differences in stroke risk among the elderly after coronary artery surgery. *Anesth Analg*, *104*, 1016-1022. 129

Guyon, I., Weson, J., Barnhill, S. & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, *46*(1-3), 389-422. 110

Hamed, H. N. A., Kasabov, N., Shamsuddin, S. M., Widiputra, H. & Dhoble, K.

(2011). An extended evolving spiking neural network model for spatio-temporal pattern classification. In *International Joint Conference on Neural Networks* (p. 2653-2656). San Jose, California, USA: IEEE. xxiii, 36, 51, 79, 135

Handley, S. (1993). Automated learning of a detector for helices in protein sequences via genetic programming. In *5th International Conference on Genetic Algorithms* (p. 271-278). San Mateo, CA: Morgan Kaufmann. 59

Hankey, G. J. (1999). Smoking and risk of stroke. *Journal of Cardiovascular Risk*, *6*(4), 207-211. 73, 128

Hatamlou, A., Abdullah, S. & Othman, Z. (2011). Gravitational search algorithm with heuristic search for clustering problems. In *3rd Conference on Data Mining and Optimization (DMO).* Selangor, Malaysia. 67

Hebb, D. (1949). *The organization of behavior.* New York: John Wiley and Sons. 43

Hodgkin, A. L. & Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology*, *117*, 500-544. 41

Holland, J. (1975). *Adaptation in natural and artificial systems.* Ann Harbor, MI: University of Michican Press. 57, 60

Hopfield, J. J. & Brody, C. D. (2000). What is a moment? Cortical sensory integration over a brief interval. *Proc. Natl. Acad. Sci*, *97*(25), 13919-13924. 42

Hopfield, J. J. & Brody, C. D. (2001). What is a moment? Transient synchrony as a collective mechanism for spatiotemporal integration. *Proc. Natl. Acad. Sci.*, *98*(3), 1282-1287. 42

Hu, Y., Song, Q. & Kasabov, N. (2009). Personalized modeling based gene selection for microarray data analysis. In *Advances in Neuro Information Processing* (p. 1221-1228). Springer LNCS. 87

Hwang, K. B., Cho, D. Y., Wook Park, S. W., Kim, S. D. & Zhang, B. Y. (2002). Applying machine learning techniques to analysis of gene expression data: Cancer diagnosis. *Methods of Microarray Data Analysis*, 167-182. 110

Izhikevich, E. M. (2004). Which model to use for cortical spiking neurons. *IEEE Transactions on Neural Networks*, *15*(5), 1063-1070. 41

Izhikevich, E. M. (2007). *Dynamical systems in neuroscience.* United States of America: The MIT Press. 41

Izhikevich, E. M. & Edelman, G. M. (2008). Large scale model of mammalian

thalamocortical systems. In *Proceedings of the National Academi of Sciences of the United States of America* (Vol. 105, p. 3593-3598). United States. 41

Jackson-Laboratory. (n.d.). *The benefits of personalized medicine.* Jackson Laboratory. Retrieved from `http://genetichealth.jax.org/personalized-medicine/what-is/benefits.html` 2

Jemal, A., Bray, F., Center, M. M., Ferlay, J., Ward, E. & Forman, D. (2011). Global cancer statistics. *CA: A Cancer Journal for Clinicians*, *61*(2), 69-90. 104

Johnston, S. C., Mendis, S. & Mathers, C. D. (2009). Global variation in stroke burden and mortality: Estimates from monitoring, surveillance, and modelling. *The Lancet Neurology*, *8*(4), 345-354. 70

Kaczmarczyk, K., Wit, A., Krawczyk, M. & Zaborski, J. (2009). Gait classification in post stroke patients using artificial neural networks. *Gait Posture*, *30*(2), 207-210. 78

Kasabov, N. (1998). ECOS: Evolving connectionist systems and the ECO learning paradigm. In *The Fifth International Conference on Neural Information Processing, ICONIP' 98* (p. 1232-1235). Kitakyushu, Japan: IOA Press. 37

Kasabov, N. (2002). *Evolving Connectionist Systems. Methods and applications in bioinformatics, brain study and intelligent machines.* London: Springer. 15, 16, 37

Kasabov, N. (2007a). *Brain, gene, and quantum inspired computational intelligence: Challenges and opportunities.* Heidelberg: Springer. 72, 153

Kasabov, N. (2007b). *Evolving Connectionist Systems: The Knowledge Engineering Approach.* London: Springer. 3, 16, 34, 153

Kasabov, N. (2007c). Global, local and personalised modelling and pattern discovery in bioinformatics: An integrated approach. *Pattern Recognition Letters*, *28*, 673-685. xxii, 2, 3, 11, 17, 18, 20, 78, 87

Kasabov, N. (2010). To spike or not to spike: A probabilistic spiking neuron model. *Neural Networks*, *23*, 16-19. 35, 153

Kasabov, N. & Hu, Y. J. (2011). Integrated optimisation method for personalised modelling and case studies for medical decision support. *Int. J. Functional Informatics and Personalised Medicine*, *3*(3), 236-256. xxii, 2, 17, 78, 88

Kasabov, N. & Song, Q. (2002). DENFIS: Dynamic Evolving Neural Fuzzy Inference System and its application for time series prediction. *Fuzzy Systems, IEEE Transactions on*, *10*(2), 144-154. 37

Kasabov, N., Song, Q., Benuskova, L., Gottgtroy, P. C. M., Jain, V., Verma, A.,

... MacDonell, S. G. (2008). *Integrating local and personalised modelling with global ontology knowledge bases for biomedical and bioinformatics decision support. In: T.G. Smolinski, M.G. Milanova, A.E. Hassanien (eds.) Computational Intelligence in Biomedicine and Bioinformatics*. Berlin: Springer. 87

Kasinski, A. J. & Ponulak, F. (2006). Comparison of supervised learning methods for spike time coding in spiking neural networks. *Int. J.of Applied Mathematics and Computer Science*, *16*, 101-113. 42

Kathryn, A. T. & Charis, E. (2012). Personalizing patient care. *Cleveland Clinic Journal of Medicine*, *79*(5), 329-330. 2

Kauhanen, M. L. (1999). Quality of life after stroke: Clinical, functiona, psychosocial and cognitive correlates. *Neurology*. 77

Kempter, R., Gerstner, W. & van Hemmen, J. L. (1999). Hebbian learning and spiking neurons. *Phys. Rev. E*, *59*(4), 4498-4514. 43

Kennedy, J. & Eberhart, R. C. (1995). Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks* (Vol. 4, p. 1942-1948). Perth, Western Australia,. 61, 152

Khajehzadeh, M. & M., E. (2012). Gravitational search algorithm for optimization of retaining structures. *Indian Journal of Science and Technology*, *5*(1). 68

Khajehzadeh, M., Taha, M. R., El Shafie, A. . & M., E. (2011). Search for critical failure surface in slope stability analysis by gravitational search algorithm. *International Journal of the Physical Sciences*, *6*(21), 5012-5021. 67

Khosla, A., Cao, Y., Lin, C. Y., Chiu, H. K., Hu, J. L. & Lee, H. (2010). An integrated machine learning approach to stroke prediction. In *16th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.* Washington, US. 77

Kistler, W. & Gerstner, W. (1997). Reduction of hodking huxley equations to a single variable threshold model. *Neural Computation*, *9*, 1015-1045. 41

Kiyan, T. & Yildirim, T. (2003). Breast cancer diagnosis using statistical neural networks. In *Turkish Symposium on Artificial Intelligence and Neural Networks.* Istanbul, Turkey. 45

Koperski, K., Han, J. & Adhikary, J. (1998). Mining knowledge in geographical data. *Communications of ACM*. 50

Koza, J. R. (1992). *Genetic programming.* Cambridge, MA: MIT Press. 57

Labiche, L. A., Chan, W., R., S. K. & Morgenstern, L. B. (2002). Sex and acute stroke presentation. *Ann Emerg Med*, *40*, 453-460. 129

Lai, S. M., Duncan, P. W. & Keighley, J. (1998). Prediction of functional outcome

after stroke : Comparison of the orpington prognostic scale and the nih stroke scale. *Stroke*, *29*, 1838-1842. 77

Lapicque, L. (1907). Recherches quantitatifs sur lexcitation electrique des nerfs traitee comme une polarisation. *J. Physiol. Pathol. Gen*(9), 620-635. 41

Larose, D. T. (2005). *Discovering knowledge in data: An introduction to data mining.* New Jersey, United States: John Wiley and Sons, Inc. 11, 78

Larsson, S. C., Virtamo, J. & Wolk, A. (2011). Red meat consumption and risk of stroke in swedish men. *American Journal of Clinical Nutrition*, *94*(2), 417-421. 73

Lasko, T. A., Bhagwat, J. G., Zou, K. H. & Ohno Machado, L. (2005). The use of receiver operating characteristic curves in biomedical informatics. *J Biomed Inform*, *38*, 404-415. 30, 31

Lee, Y. J. & Mangasarian, O. L. (2001). SSVM: A smooth support vector machine for classification. *Computational Optimization and Applications*, *22*(1), 5-21. 13

Li, C. & Zhou, J. (2011). Parameters indetification of hydraulic turbine governing system using improved gravitational search algorithm. *Energy Conversion and Management*, *52*(1), 374-381. 67

Li, L. (2006). Survival prediction of diffuse large B cell lymphoma based on both clinical and gene expression information. *Bioinformatics*, *22*(4), 466-471. 3

Lin, C. T. & Lee, C. S. G. (1996). *Neural fuzzy systems: A neuro fuzzy synergism to intelligent systems.* Upper Saddle River, NJ: Prentice Hall. 21

Lin, H. T., Lin, C. J. & Weng, R. C. (2007). A note on platt's probabilistic outputs for support vector machines. *Machine Learning*, *68*(3), 267-276. 95, 96

Lockhart, D., Dong, H., Byrne, M., Follettie, M., Gallo, M., Chee, M., ... Horton, H. (1996). Expression monitoring by hybridization to high-density oligonucleotide arrays. *Nature Biotechnology*, *14*, 1675-1680. 108

Maass, W. (1997). Networks of spiking neurons: The third generation of neural network models. *Neural Networks*, *10*(9), 1659-1671. 38

Maass, W. & Bishop, C. M. (1999). *Pulsed neural networks.* Cambridge, MA: The MIT Press. 41

Maass, W., Natschlager, T. & Markram, H. (2002). Real time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, *14*(11), 2531-2560. 44, 45, 53

Maass, W., Natschlager, T. & Markram, H. (2003). A model for real time com-

putation in generic neural microcircuits. In *NIPS 2002, Advances in Neural Information Processing Systems* (Vol. 15, p. 229-236). MIT Press. 42

Malsburg, C. v. d. (1981). *The correlation theory of brain function. In models of Neural Networks II.* Berlin: Springer. 40

Mansouri, R., Nasseri, F. & Khorrami, M. (1999). Effective time variation of G in a model universe with variable space dimension. *Physics Letters*, *259*, 194-200. 65

Mathers, C. D., Boschi-Pinto, C., Lopez, A. D. & Murray, C. J. L. (2001). Cancer incidence, mortality and survival by site for 14 regions of the world. *Geneva: World Health Organization.* 115

Michelon, P. (2008). *Brain games: Spot the difference.* Retrieved from `http://www.sharpbrains.com/blog/2008/06/11/brain-games-spot-the-difference/` x, 72

Ministry-Health. (2006). *Lung cancer.* Retrieved from `http://www.health.govt.nz/yourhealth-topics/diseases-and-illnesses/cancer/lung-cancer` 119

Mukamal, K. J., Wellenius, G. A., Suh, H. H. & Mittleman, M. A. (2009). Weather and air pollution as triggers of severe headaches. *Neurology*, *72*, 922-927. 125

Nielsen, T. O., West, R. B., Linn, S. C., Alter, O., Knowling, M. A., O'Connell, J. X., ... van de Rijn, M. (2002). Molecular characterisation of soft tissue tumours: A gene expression study. *Lancet*, *359*, 1301-1307. 108

Nocedal, J. & Wright, S. J. (1999). *Numerical optimization.* New York: Springer Verlag. 96

Nyquist, P. A., Brown, R. D., Wiebers, D. O., Crowson, C. S. & OFallon, W. M. (2001). Circadian and seasonal occurrence of subarachnoid and intracerebral hemorrhage. *Neurology*, *56*, 190-193. 123

Obuchowski, N. A. (2003). Receiver operating characteristic curves and their use in radiology. *Radiology*, *229*(3-8). 30

Oczkowski, W. J. & Barreca, S. (1997). Neural network modeling accurately predicts the functional outcome of stroke survivors with moderate disabilities. *Archives of Physical Medicine and Rehabilitation*, *78*(4), 5-340. 45

Oliver, J. R. (1993). Discovering individual decision rules: An application of genetic algorithms. In *5th International Conference on Genetic Algorithms* (p. 216-222). San Mateo, CA: Morgan Kaufmann. 59

Parrott, D. & Li, X. (2006). Locating and tracking multiple dynamic optima by

a particle swarm model using speciation. *IEEE Transactions on Evolutionary Computation*, 440-458. 62

Platt, J. (1999). Probabilistic outputs for support vector machines and comparisinos to regularized likelihood methods. *Advances in Large Margin*, 61-74. 94, 95

Press, W. H., Flannery, B. P., Teukolsky, S. A. & Vetterling, W. T. (1992). *Numerical recipes: The art of scientific computing*. Cambridge, UK: Cambridge University Press. 96

Provost, F. & Kolluri, V. (1999). A survey of methods for scaling up inductive algorithms. *Data Mining and Knowledge Discovery*, *2*, 131-169. 24

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proc. IEEE*, *77*, 257-286. 50

Ramaswamy, S. & Perou, C. (2003). DNA microarrays in breast cancer: The promise of personalised medicine. *Lancet*, *361*(9369), 1590-1596. 87

Ransohoff, D. F. (2004). Rules of evidence for cancer molecular marker discovery and validation. *Nature Reviews Cancer*, *4*, 309-314. 27

Rashedi, E., Nezamabadi-pour, H. & Saryazdi, S. (2009). GSA: A gravitational search algorithm. *Information Sciences*, *179*, 2232-2248. 56, 62, 68, 69

Rashedi, E., Nezamabadi-pour, H. & Saryazdi, S. (2010). BGSA: Binary gravitational search algorithm. *Nat Comput*, *9*, 727-745. 56, 67, 68

Rechenberg, I. (1973). *Evolutions strategie: Optimierung technischer systeme nach prinzipien der biologischen evolution*. Stuttgart: Fromman-Holzboog. 57

Reynolds, K., Lewis, B. & Nolen, J. D. (2003). Alcohol consumption and risk of stroke: A meta-analysis. *JAMA*, *289*(5), 579-588. 73

Ricci, S., Celani, M., Vitali, R., La Rosa, F., Righetti, E. & Duca, E. (1992). Diurnal and seasonal variations in the occurrence of stroke: A community based study. *Neuroepidemiology*, *11*, 59-64. 123

Rothwell, P. M. (2001). The high cost of not funding stroke research: A comparison with heart disease and cancer. *The Lancet Neurology*, *357*(9268), 1612-1616. 70

Russell, S. J. & Norvig, P. (1995). *Artificial intelligence a modern approach*. New Jersy: Prentice Hall. 33

Sarafrazi, S., Nezamabadi-pour, H. & Brahman, M. (2010). A GSA-SVM hybrid system for classfication of binary problems. In *4th Global Conference on Power Control and Optimization.* Malaysia. 69

Schena, M., Shalon, D., Davi, R. & Brown, P. (1995). Quantitative monitoring of

gene expression patterns with a complementary dna microarray. *Science*, *270*, 467-470. 108

Schliebs, S., Hamed, H. N. A. & Kasabov, N. (2011). Reservoir based evolving spiking neural network for spatio temporal pattern recognition. In *Proceedings of the 18th International Conference on Neural Information Processing* (p. 160-168). Shanghai, China: Springer Verlag. xxiii, 36, 53, 79, 135

Schneider, N. C. & Graupe, D. (2008). A modified lamstar neural network and its applications. *International Journal of Neural Systems*, *18*(4), 331-337. 45

Scholkopy, A. J. & Smola, P. L. (2000). New support vector algorithms. *Neural Computation*, *12*(5), 1207-1245. 13

Seshadri, S. e. a. (2006). The lifetime risk of stroke: Estimates from the framingham study. *Stroke*, *37*, 345-350. 72, 129

Shabo, A. (2007). Health record banks: Integrating clinical and genomic data into patient-centric longitudinal and cross-institutional health records. *Personalised Medicine*, *4*(4), 453-455. 2, 110

Shapiro, D. E. (1999). The interpretation of diagnostic tests. *Stat Methods Med Res*, *8*, 113-134. 31

Shipp, M. A., Ross, K. N., Tamayo, P., Weng, A. P., Kutok, J. L. & Aguiar, R. C. T. (2002). Diffuse large B-cell lymphoma outcome prediction by geneexpression profiling and supervised machine learning. *Nat Med*, *8*(1), 68-74. 111

Solomonoff, R. J. (1960). *A preliminary report on a general theory of inductive inference*. UK, Cambridge: Zator Company. 10

Soltic, S., Wysoski, S. G. & Kasabov, N. (2008). Evolving spiking neural networks for taste recognition. In *IEEE World Congress on Computational Intelligence* (p. 2091-2097). Hong Kong. 46

Song, Q. & Kasabov, N. (2006). TWNFI-a transductive neuro-fuzzy inference system with weighted data normalization for personliased modelling. *Neural Networks*, *19*, 1591-1596. 18, 21

Spackman, K. A. (1989). Signal detection theory: Valuable tools for evaluating inductive learning. In *Six International Workshop on Machine Learning* (p. 160-163). San Mateo, CA: Morgan Kaufman. 30

Stoupel, E., Israelevich, P., Gabbay, U., Abramson, E., Petrauskiene, J., Kalediene, B., ... Sulkes, J. (2000). Correlation of two levels of space proton flux with monthly distribution of deaths from cardiovascular disease and suicide. *J Basic Clin Physiol Pharmacol*, *11*, 63-71. 3

Street, W. N., Wolberg, W. H. & Mangasarian, O. L. (1993). Nuclear feature extraction for breast tumor diagnosis. In *1993 International Symposium on Electronic Imaging: Science and Technology* (Vol. 1905, p. 861-870). San Jose, CA. 97

Strong, K., Mathers, C. & Bonita, R. (2007). Preventing stroke: Saving lives around the world. *Lancet Neurology*, *6*, 182-187. 72

Suykens, J. A. K. & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters*, *9*(3), 293-300. 13

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., ... Golub, T. R. (1999). Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *Proc Natl Acad Sci USA*, *96*(6), 2907-2912. 110

Tan, A. C. & Gibert, D. (2003). Ensembel machine learning on gene expression data for cancer classification. *Applied Bioinformatics*, *2*(3), 75-83. 110

Tavazoie, S., Hughes, J. D., Campbell, M. J., Cho, R. J. & Church, G. M. (1999). Systematic determination of genetic network architecture. *Nat Genet*, *22*(3), 281-285. 110

Thomson, R. (2009). Evidence based implementation of complex interventions. *BMJ*, *339*. 73

Thorpe, S., Fize, D. & Marlot, C. (1996). Speed of processing in the human visual system. *Nature*, *381*, 520-522. 40

Thorpe, S. J. (1997). How can the human visual system process a natural scene in under 150ms? In *European Symposium on Artifical Neural Networks.* Bruges, Belgium. 48

Tobias, M., Cheung, J., Carter, K., Anderson, C. & Feigin, V. L. (2007). Stroke surveillance: Population based estimates and projections for New Zealand. *Aust N Z J Public Health*, *31*(6), 520-525. 70

Tobias, M., Cheung, J. & McNaughton, H. (2002). *Modelling stroke : A multi-state life table model.* PhD Thesis. 70

Tsapatsoulis, N., Rapantzikos, K. & Pattichis, C. (2007). An embedded saliency map estimator scheme: Application to video encoding. *International Journal of Neural Systems*, *17*(4), 289-304. 45

van't Veer, L. J., Dai, H., van de Vijver, M. J., He, Y. D., Hart, A. A., Mao, M., ... Friend, S. H. (2002). Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, *415*, 530-536. 108

Vapnik, V. (1998). *Statistical learning theory.* NY: Wiley Interscience. xxii, 2, 3, 10, 13, 17, 78

Vapnik, V. & Lerner, A. (1963). Pattern recognition using generalized portrait method. *Automation and Remote Control*, *24*, 774-780. 13

Verhoeven, M. G. A., Aarts, E. H. L., van de Sluis, E. & Vaessens, R. J. M. (1992). Parallel local search and the travelling salesman problem. *Parallel Problem Solving from Nature*, *2*, 543-552. 59

Vlak, M., Rinkel, G., Greebe, P., van der Bom, J. & Algra, A. (2011). Trigger factors and their attributable risk for rupture of intracranial aneurysms: A case-crossover study. *Stroke*, *42*, 1878-1882. 126

Waibel, A., Hanazawa, T., Hinton, G., Shikano, K. & Lang, K. (2002). Phoneme recognition using time delay neural networks. *IEEE Trans. Acoust Speech Signal Process*, *37*(3), 328-339. 50

Wang, L. X. (1994). *Adaptive fuzzy systems and control: Design and stability analysis.* Englewood Cliffs, NJ: Prentice Hall. 21

Wannamethee, S. G., Shaper, A. G., Whincup, P. H. & Walker, M. (1995). Smoking cessation and the risk of stroke in middle-aged men. *JAMA*, *274*(2), 155-160. 73, 128

Watts, M. J. (2009). A decade of Kasabov's evolving connectionist systems: A review. *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews*, *39*(3). 37

WHO. (n.d.). *Colon cancer.* Retrieved from `http://globocan.iarc.fr/` 112

Wieslaw, J., Oczkowski, M. D. & Barreca, B. A. (1997). Neural network modeling accurately predicts the functional outcome of stroke survivors with moderate disabilities. *Archives of Physical Medicine and Rehabilitation*, *78*(4), 340-345. 2, 77, 78

Wysoski, S. G., Benuskova, L. & Kasabov, N. (2006). Online learning with structural adaptation in a network of spiking neurons for visual pattern recognition. In *2006 International Conference on Artificial Neural Networks, LNCS* (Vol. 4131, p. 61-70). Berlin Heidelberg: Springer Verlag. 46

Wysoski, S. G., Benuskova, L. & Kasabov, N. (2007). Text independent speaker authentication with spiking neural networks. In *International Conference on Artificial Neural Networks* (p. 756-767). Porto, Portugal. 46

Wysoski, S. G., Benuskova, L. & Kasabov, N. (2008). Fast and adaptive network of spiking neurons for multi-view visual pattern recognition. *Neurocomputing*,

71(13-15), 2563-2575. 46

Xiao, J. H. & Cheng, Z. (2011). DNA sequences optimization based on gravitational search algorithm for reliable DNA computing. In *Sixth International Conference on Bio-Inspired Computing: Theories and Applications.* USM, Penang, Malaysia. 67

Xinchao, Z. (2010). A perturbed particle swarm algorithm for numerical optimization. *Applied Soft Computing*, *10*(1), 119-124. 62

Yamazaki, T. & Tanaka, S. (2007). 2007 special issue: The cerebellum as a liquid state machine. *Neural Netw*, *20*(3), 290-297. 44, 45

Yang, X. S. (2008). *Nature inspired metaheuristic algorithms.* Cambridge, United Kingdom: Luniver Press. 33

Yang, X. S. (2010). *Engineering optimization: An introduction with metaheuristic applications.* Cambridge, United Kingdom: John Wiley and Sons. 33

Yang, X. S. & Deb, S. (2009). Cuckoo search via lvy flights. In *World Congress on Nature and Biologically Inspired Computing* (p. 210-214). IEEE Publications. 152

Yang, Y. & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. In *14th International Conference on Machine Learning.* Nashville, TN. 24

Yau, W. C., Kumar, D. K. & Arjunan, S. P. (2007). Visual recognition of speech consonants using facial movement features. *Integrated Computer Aided Engineering*, *14*(1), 49-61. 45

Yeang, C., Ramaswamy, S. & Tamayo, P. (2001). Molecular classification of multiple tumor types. *Bioinformatics*, *17*(1), 316-322. 110

Yoshimoto, H., Saltsman, K., Gasch, A. P., Li, H. X., Ogawa, N., Botstein, D., ... Cyert, M. S. (2002). Genome-wide analysis of gene expression regulated by the Calcineurin/Crz1p signaling pathway in saccharomyces cerevisiae. *J Biol Chem*, *277*, 31079-31088. 108

Zeeshan, S. & Ilan, R. (2010). Personalized risk stratification for adverse surgical outcomes: Innovation at the boundaries of medicine and computation. *Personalized Medicine*, *7*(6), 695-701. 93

APPENDIX A

Appendix A - Result of 100 Breast Cancer Wisconsin Samples Achieved by knnGSA

================================== Result ==================================

Sample ID	Actual Class	Predicted Class	probRisk
1	1	1	1.000
2	1	1	0.800
3	1	1	0.800
4	2	1*	0.800
5	1	1	0.667
6	2	2	1.000
7	1	1	1.000
8	2	2	1.000
9	2	2	0.909
10	2	2	1.000
11	1	1	1.000
12	1	1	1.000
13	1	1	1.000
14	1	1	1.000
15	1	1	1.000
16	2	2	1.000
17	1	1	1.000
18	2	2	1.000
19	2	2	1.000
20	1	1	1.000
21	1	1	1.000
22	2	2	1.000
23	1	1	1.000
24	2	2	0.750
25	2	2	0.938
26	1	1	1.000
27	2	2	1.000
28	1	1	1.000
29	2	2	1.000
30	2	2	1.000
31	1	1	1.000
32	1	1	1.000
33	2	2	1.000
34	1	1	1.000
35	1	1	1.000
36	2	2	1.000
37	2	1*	0.917
38	1	1	1.000
39	1	1	1.000
40	1	1	0.909
41	1	1	1.000
42	2	2	1.000
43	2	2	0.615
44	1	1	1.000
45	1	1	1.000
46	1	1	1.000
47	1	1	1.000

48	1	1	1.000
49	2	2	1.000
50	2	2	0.556
51	2	2	1.000
52	1	1	1.000
53	2	2	1.000
54	1	1	1.000
55	2	2	1.000
56	1	1	1.000
57	1	1	1.000
58	1	1	1.000
59	2	2	1.000
60	2	2	1.000
61	1	1	1.000
62	2	2	0.900
63	2	2	0.933
64	2	2	1.000
65	1	1	1.000
66	2	2	1.000
67	2	2	1.000
68	1	1	1.000
69	1	1	1.000
70	1	1	1.000
71	1	1	1.000
72	1	2*	1.000
73	1	1	1.000
74	1	1	1.000
75	1	1	1.000
76	2	2	1.000
77	2	2	1.000
78	1	1	1.000
79	1	1	1.000
80	1	1	1.000
81	2	2	1.000
82	2	2	1.000
83	1	1	1.000
84	1	1	1.000
85	1	1	1.000
86	2	2	1.000
87	2	2	1.000
88	1	1	1.000
89	2	2	1.000
90	2	2	1.000
91	2	2	1.000
92	1	1	1.000
93	1	1	1.000
94	2	2	1.000
95	1	1	1.000
96	1	1	1.000
97	2	2	0.889
98	2	2	0.500

99	2	2	0.917
100	2	2	0.875

Overall Accuracy of Leave-one-out Crossvalidation: 97.00%

Class 1 Overall Accuracy: 98.18%
Class 2 Overall Accuracy: 95.56%
Class1 Confusion Table: 54(Correctly Classified) 55(Total)
Class2 Confusion Table: 43(Correctly Classified) 45(Total)

Sample ID	# K	KNN Index
1	9	1 2 11 13 14 16 19 20 22
2	7	12 14 19 25 33 47 67
3	4	37 2 13 14
4	12	93 29 52 63 90 86 65 5 58 81 80 9
5	9	15 54 6 97 1 33 64 94 3
6	9	9 76 18 50 17 81 58 42 86
7	7	64 94 10 33 5 2 3
8	12	76 81 36 29 59 75 21 96 49 24 4 90
9	11	28 54 96 98 49 24 36 59 71 75 76
10	9	6 26 35 58 62 75 80 88 99
11	5	34 51 60 2 13
12	11	39 2 14 34 51 11 16 25 31 38 43
13	14	19 3 20 25 30 37 40 53 67 70 74 7 38 43
14	14	27 55 82 83 31 44 46 47 56 60 68 73 77 79
15	15	2 13 19 25 33 47 67 74 91 11 12 14 20 22 27
16	15	5 64 1 16 94 7 11 33 40 72 95 3 13 20 22
17	20	30 40 2 15 19 20 22 25 31 34 38 43 45 46 47 51 53 56 57 67
18	3	36 10 9
19	3	59 54 9
20	7	13 20 25 30 53 67 70
21	9	1 2 13 15 17 20 22 25 30
22	12	59 76 49 32 8 98 18 6 10 54 9 36
23	11	57 78 84 67 3 21 37 53 70 94 13
24	8	28 16 9 18 54 5 19 23
25	16	4 93 63 71 75 61 62 99 29 58 90 36 96 86 26 41
26	20	2 15 20 47 67 74 91 1 3 5 7 11 12 13 14 21 23 27 31 33
27	15	76 6 96 66 50 88 18 35 48 62 28 65 80 75 85
28	16	14 55 82 83 17 46 60 68 77 79 87 92 95 34 47 51
29	7	9 54 8 98 59 36 49
30	13	4 93 65 86 50 27 58 76 8 10 22 52 80
31	15	20 21 53 70 3 13 26 37 38 40 43 45 67 69 74
32	17	14 28 46 47 55 56 68 73 77 79 82 83 87 91 92 2 12
33	10	8 59 22 49 9 36 54 99 29 98
34	24	11 13 64 2 12 14 15 20 21 23 26 28 32 34 38 43 45 46 47 51 53 55 56 57

35 2 10 6
36 13 8 54 96 6 27 36 49 59 76 9 29 75 98
37 12 75 71 81 61 25 30 98 99 10 19 58 86
38 11 3 7 40 1 2 5 11 13 14 15 17
39 5 2 6 13 20 9
40 22 12 8 44 14 28 55 60 82 83 84 95 97 1 2 11 13 15 17 20 21 23 26
41 18 3 20 21 31 38 53 70 13 26 39 43 45 67 69 74 78 84 1
42 16 66 22 33 9 18 37 90 19 59 50 49 54 16 24 75 8
43 13 19 81 64 7 16 52 22 94 98 34 59 5 6
44 11 26 39 45 69 74 1 2 13 15 20 21
45 16 14 28 55 60 82 83 84 95 1 2 11 12 13 15 20 21
46 18 26 39 44 69 74 2 12 13 15 20 21 31 35 51 53 60 70 3
47 8 14 28 55 68 77 79 82 83
48 7 1 2 11 14 15 20 21
49 7 18 42 66 80 24 63 4
50 18 59 22 98 33 19 8 54 16 10 72 9 23 57 94 5 13 1 67
51 11 42 58 96 4 65 93 10 66 18 81 75
52 10 11 35 60 2 14 15 17 28 39 44
53 16 6 10 36 19 27 98 51 59 90 81 58 61 75 4 37 93
54 3 1 3 21
55 10 10 33 42 51 58 80 6 49 24 29
56 5 14 28 60 82 83
57 16 32 45 73 34 48 91 2 15 14 17 28 46 47 56 64 68
58 13 23 94 72 78 84 7 67 1 21 54 70 3 20
59 9 37 75 63 10 86 89 65 99 61
60 4 50 22 19 98
61 18 14 28 56 82 83 84 17 21 23 35 39 44 47 52 54 58 68 69
62 10 71 99 10 75 98 25 50 37 60 22
63 15 43 27 49 36 85 18 80 75 24 29 88 42 40 4 93
64 7 59 42 66 51 80 86 90
65 3 14 17 28
66 16 30 88 90 86 75 66 37 25 89 36 4 93 99 85 53 59
67 10 76 42 27 51 43 6 10 49 53 62
68 12 2 13 15 20 26 48 74 91 1 3 17 21
69 10 1 3 14 21 23 28 35 38 39 44
70 21 26 39 44 46 74 2 12 13 15 20 21 35 52 54 70 3 11 38 61 41 32
71 17 3 14 17 21 23 28 35 38 39 41 44 47 52 54 56 58 61
72 6 98 25 9 67 90 50
73 11 1 95 5 3 21 23 35 38 39 44 45
74 13 2 11 15 17 20 21 23 26 31 32 35 39 44
75 6 2 13 15 20 26 48
76 4 37 42 59 90
77 3 81 6 22
78 14 2 3 7 11 12 15 20 21 23 26 32 35 38 39
79 10 21 23 54 58 68 71 84 3 20 38
80 8 1 14 17 21 23 28 35 39
81 4 25 36 66 4
82 20 37 43 18 19 8 22 63 9 76 86 6 90 29 27 60 88 66 36 67 10
83 18 3 7 11 14 17 21 23 28 35 38 39 41 44 47 52 54 56 58
84 14 14 28 56 83 47 61 69 78 80 87 92 35 48 52
85 17 7 79 3 21 23 38 54 58 68 71 94 20 41 13 39 44 70

86	18	49 88 81 66 67 4 90 93 86 89 42 30 59 36 27 76 18 51
87	4	59 66 30 89
88	19	17 32 47 48 57 69 74 78 80 91 92 95 2 14 15 28 34 35 45
89	4	81 86 49 89
90	9	87 59 64 10 66 4 93 89 18
91	4	30 4 82 93
92	11	14 17 28 32 40 45 47 48 56 57 69
93	18	1 3 14 21 23 28 35 38 39 44 47 52 54 56 58 61 69 70
94	9	4 30 66 81 49 59 86 89 91
95	7	23 58 68 79 85 7 20
96	3	9 35 80
97	9	64 25 99 72 4 94 30 62 76
98	2	31 16
99	12	62 72 99 50 25 10 60 76 37 59 64 22
100	16	33 72 50 22 60 76 8 30 99 37 62 4 5 16 19 87

Average number of K been selected: 11

Sample ID	# Features	Feature Index
1	2	2 3
2	4	6 4 8 7
3	6	2 6 3 4 8 9
4	6	2 6 5 1 7 9
5	5	3 4 8 7 9
6	5	3 5 8 1 7
7	4	6 3 5 4
8	5	2 3 4 1 9
9	4	2 6 4 9
10	2	6 7
11	4	4 1 8 7
12	7	2 3 5 4 1 8 9
13	5	2 4 1 7 9
14	6	6 3 5 4 1 9
15	4	6 4 8 7
16	7	2 6 3 5 4 7 9
17	7	2 6 3 5 4 8 9
18	4	2 5 8 9
19	7	2 3 5 4 1 7 9
20	6	2 5 4 1 8 7
21	4	3 5 8 9
22	4	2 6 3 8
23	6	6 3 4 1 7 9
24	6	2 3 8 1 7 9
25	4	3 5 8 7
26	3	2 6 7
27	4	6 5 1 7

28	6	2 3 5 1 8 7
29	6	6 3 5 4 8 7
30	2	2 4
31	7	2 3 5 4 1 8 9
32	4	6 3 4 1
33	5	6 3 5 7 9
34	5	2 6 3 4 8
35	4	6 5 4 7
36	2	6 4
37	4	3 5 4 8
38	2	8 9
39	2	6 4
40	4	5 4 8 9
41	4	2 3 4 1
42	3	2 7 9
43	5	6 2 3 8 7
44	5	6 3 1 8 9
45	4	6 5 4 8
46	4	3 4 1 8
47	7	2 6 3 1 8 7 9
48	4	2 6 8 9
49	3	5 4 1
50	6	2 5 4 8 1 9
51	5	2 5 4 8 9
52	3	2 1 7
53	4	3 4 1 7
54	5	6 5 1 7 9
55	2	6 1
56	4	2 5 4 7
57	4	1 8 7 9
58	5	2 6 1 8 7
59	5	2 6 5 8 9
60	8	2 3 5 4 8 1 7 9
61	5	3 5 4 8 7
62	4	2 6 4 7
63	3	3 5 9
64	6	2 6 4 1 8 9
65	6	2 4 1 8 7 9
66	5	3 1 8 7 9
67	3	2 3 5
68	4	3 5 7 9
69	4	2 6 3 7
70	6	6 3 5 4 1 8
71	4	2 3 4 7
72	4	2 6 1 9
73	6	6 3 5 4 7 9
74	5	2 5 4 8 9
75	5	3 5 4 7 9
76	3	2 8 9
77	6	2 3 5 4 8 1
78	4	2 6 5 4

79	8	2 6 3 5 4 1 8 7
80	3	3 8 7
81	3	5 8 7
82	5	3 5 8 1 7
83	2	4 7
84	6	6 3 5 4 1 7
85	6	2 6 4 1 7 9
86	6	2 3 4 1 8 9
87	8	2 6 3 5 4 8 7 9
88	4	3 5 1 9
89	4	2 3 4 9
90	4	4 8 7 9
91	4	6 3 1 9
92	4	2 3 1 8
93	4	2 6 3 7
94	3	2 4 8
95	5	2 6 4 1 8
96	2	6 4
97	4	3 8 7 9
98	3	2 6 1
99	5	2 6 5 7 9
100	3	3 5 4

Average number of Features been selected: 5

========== End ==========

APPENDIX B

Appendix B - Result of 100 Breast Cancer Wisconsin Samples Achieved by svmGSA

==========Result==========

Sample ID	Actual Class	Predicted Class	Predicted Class (based on Probability)	Pro in Class1	Pro in Class2
1	1	1	1	0.95442	0.04558
2	1	1	1	0.97257	0.02743
3	1	1	1	0.96337	0.03663
4	2	2	2	0.27223	0.72777
5	1	1	1	0.60940	0.39060
6	2	2	1*	0.95759	0.04241
7	1	1	1	0.96051	0.03949
8	2	2	2	0.12078	0.87922
9	2	2	2	0.13689	0.86311
10	2	2	2	0.05301	0.94699
11	1	1	1	0.97173	0.02827
12	1	1	1	0.97151	0.02849
13	1	1	1	0.97072	0.02928
14	1	1	1	0.96782	0.03218
15	1	1	1	0.98417	0.01583
16	2	1*	2	0.07845	0.92155
17	1	1	1	0.96107	0.03893
18	2	2	2	0.16031	0.83969
19	2	2	2	0.50000	0.50000
20	1	1	1	0.98427	0.01573
21	1	1	1	0.97876	0.02124
22	2	2	2	0.07526	0.92474
23	1	1	1	0.96223	0.03777
24	2	2	2	0.37952	0.62048
25	2	2	2	0.41948	0.58052
26	1	1	1	0.96656	0.03344
27	2	2	2	0.07862	0.92138
28	1	1	1	0.96700	0.03300
29	2	2	2	0.44141	0.55859
30	2	2	2	0.13129	0.86871
31	1	1	1	0.95684	0.04316
32	1	1	1	0.96685	0.03315
33	2	2	2	0.07132	0.92868
34	1	1	1	0.96843	0.03157
35	1	1	1	0.96843	0.03157

36	2	2	2	0.09207	0.90793
37	2	2	2	0.20231	0.79769
38	1	1	1	0.96694	0.03306
39	1	1	1	0.96843	0.03157
40	1	1	1	0.96593	0.03407
41	1	1	1	0.96829	0.03171
42	2	2	2	0.07638	0.92362
43	2	2	2	0.04218	0.95782
44	1	1	1	0.96784	0.03216
45	1	1	1	0.90962	0.09038
46	1	1	1	0.96622	0.03378
47	1	1	1	0.96794	0.03206
48	1	1	1	0.96843	0.03157
49	2	2	2	0.04590	0.95410
50	2	2	2	0.13598	0.86402
51	2	2	2	0.09608	0.90392
52	1	1	1	0.96843	0.03157
53	2	2	2	0.06819	0.93181
54	1	1	1	0.98325	0.01675
55	2	2	2	0.18602	0.81398
56	1	1	1	0.98089	0.01911
57	1	1	1	0.96781	0.03219
58	1	1	1	0.96253	0.03747
59	2	2	2	0.08795	0.91205
60	2	2	2	0.08877	0.91123
61	1	1	1	0.96111	0.03889
62	2	2	2	0.11805	0.88195
63	2	2	2	0.10455	0.89545
64	2	2	2	0.14825	0.85175
65	1	1	1	0.96802	0.03198
66	2	2	2	0.11677	0.88323
67	2	2	2	0.06286	0.93714
68	1	1	1	0.96378	0.03622
69	1	1	1	0.97276	0.02724
70	1	1	1	0.96554	0.03446
71	1	1	1	0.97295	0.02705
72	1	1	1	0.96998	0.03002
73	1	1	1	0.96686	0.03314
74	1	1	1	0.96510	0.03490

75	1	1	1	0.96732	0.03268
76	2	2	2	0.47376	0.52624
77	2	2	2	0.05040	0.94960
78	1	1	1	0.97107	0.02893
79	1	1	1	0.97889	0.02111
80	1	1	1	0.96897	0.03103
81	2	2	2	0.07861	0.92139
82	2	2	2	0.04727	0.95273
83	1	1	1	0.96687	0.03313
84	1	1	1	0.96689	0.03311
85	1	1	1	0.96778	0.03222
86	2	2	2	0.05046	0.94954
87	2	2	2	0.10404	0.89596
88	1	1	1	0.96394	0.03606
89	2	2	2	0.04741	0.95259
90	2	2	2	0.04742	0.95258
91	2	2	2	0.19843	0.80157
92	1	1	1	0.96746	0.03254
93	1	1	1	0.96799	0.03201
94	2	2	2	0.25917	0.74083
95	1	1	1	0.94306	0.05694
96	1	1	1	0.97000	0.03000
97	2	2	2	0.75812	0.24188
98	2	2	2	0.06510	0.93490
99	2	2	2	0.42421	0.57579
100	2	2	2	0.50000	0.50000

Overall Accuracy of Leave-one-out Crossvalidation: 99.00%

Class 1 Accuracy: 100.00%
Class 2 Accuracy: 97.78%
Class1 Confusion Table: 55(Correctly Classified) 55(Total)
Class2 Confusion Table: 44(Correctly Classified) 45(Total)

Overall Accuracy of Leave-one-out Crossvalidation (base on probability): 99.00%

Class 1 Accuracy: 100.00%

Class 2 Accuracy: 97.78%
Class1 Confusion Table: 55(Correctly Classified) 55(Total)
Class2 Confusion Table: 44(Correctly Classified) 45(Total)

Sample ID	Best Gamma	Best C
1	0.23	91.50
2	0.13	205.59
3	0.12	113.03
4	0.58	8.52
5	0.17	201.65
6	0.35	13.65
7	0.74	105.16
8	0.80	210.93
9	0.30	169.11
10	0.43	198.31
11	0.12	159.88
12	0.23	233.80
13	0.40	154.77
14	0.66	80.49
15	0.02	236.73
16	0.77	185.58
17	0.59	51.60
18	0.37	238.62
19	0.93	72.67
20	0.03	246.42
21	0.09	166.96
22	0.94	124.32
23	0.33	39.57
24	0.92	76.22
25	0.75	78.80
26	0.63	100.89
27	0.36	51.94
28	0.88	239.94
29	0.71	175.98
30	0.29	214.35

31	0.54	28.21
32	0.73	234.07
33	0.38	177.03
34	0.67	215.62
35	0.87	160.63
36	0.57	155.19
37	0.83	67.65
38	0.14	163.57
39	0.39	151.90
40	0.23	20.36
41	0.01	11.62
42	0.36	61.44
43	0.01	50.18
44	0.71	213.75
45	0.46	76.92
46	0.32	28.64
47	0.48	80.39
48	0.37	64.92
49	0.87	94.58
50	0.77	93.57
51	0.75	30.17
52	0.94	67.94
53	0.30	135.12
54	0.11	97.28
55	0.79	113.03
56	0.32	154.74
57	0.40	225.64
58	0.42	226.63
59	0.98	198.74
60	0.11	173.30
61	0.38	152.06
62	0.89	1.19
63	0.97	168.19
64	0.33	132.32
65	0.57	87.84
66	0.33	179.65
67	0.21	54.20
68	0.72	111.83
69	0.06	237.72

70	0.66	52.77
71	0.27	162.11
72	0.80	195.69
73	0.72	145.23
74	0.98	23.14
75	0.33	29.76
76	0.14	111.68
77	0.81	14.37
78	0.15	86.20
79	0.25	102.01
80	0.45	101.21
81	0.57	213.38
82	0.49	108.33
83	0.68	90.81
84	0.48	30.26
85	0.22	155.91
86	0.79	41.92
87	0.58	206.45
88	0.40	228.18
89	0.32	65.76
90	0.59	208.95
91	0.19	81.80
92	0.70	141.90
93	0.85	57.59
94	0.83	107.60
95	0.53	92.49
96	0.76	61.28
97	0.14	130.64
98	0.68	74.80
99	0.13	178.23
100	0.51	244.06

Average number of Gamma: 0.49

Average number of C: 124.89

Sample ID	# K	KNN Index
1	17	72 20 53 70 2 19 37 38 43 69 78 12 25 40 67 74 84
2	13	14 25 34 47 51 74 91 10 31 38 43 45 46
3	9	37 20 40 53 70 19 38 43 69
4	12	93 29 90 86 63 65 52 58 80 75 61 71
5	2	6 15
6	10	76 9 59 26 50 7 36 81 75 35
7	6	94 5 3 37 78 20
8	16	59 21 49 36 98 75 81 86 8 32 6 9 76 99 17 71
9	8	28 54 36 98 59 49 32 71
10	9	58 86 75 98 6 50 36 59 61
11	8	34 51 2 14 38 43 46 60
12	6	39 2 14 31 45 56
13	9	19 20 25 53 67 70 74 3 37
14	7	27 55 82 83 46 60 68
15	11	2 25 34 47 51 74 91 11 31 38 43
16	17	5 7 40 3 37 1 72 94 78 13 20 22 53 57 70 84 97
17	12	46 68 77 79 87 92 14 27 34 47 51 55
18	10	23 50 75 86 41 58 21 62 36 71
19	2	16 5
20	9	13 20 25 53 67 70 74 3 37
21	11	53 70 3 20 37 38 43 69 78 13 25
22	19	59 49 8 32 36 98 18 76 10 75 86 9 54 81 6 23 99 28 71
23	9	57 78 67 84 21 53 70 94 3
24	3	18 32 71
25	2	71 99
26	10	74 2 15 20 38 43 45 69 13 21
27	13	35 75 50 36 96 6 41 48 80 18 88 62 85
28	11	14 55 82 83 46 60 68 77 79 87 92
29	5	9 54 98 49 71
30	13	65 4 93 90 86 75 25 36 99 58 88 89 71
31	20	20 45 13 26 40 67 74 21 53 70 2 3 15 37 38 43 69 78 84 97
32	9	56 73 47 91 2 15 44 33 45
33	15	49 59 54 22 98 24 9 10 29 8 97 18 16 36 71
34	8	47 64 91 2 11 15 32 46
35	10	51 2 11 15 38 43 46 60 68 69
36	23	75 27 36 85 80 65 88 30 90 6 48 86 25 58 99 10 4 41 93 96 81 50 71
37	6	75 99 8 25 98 71

38	8	3 21 40 53 70 20 38 43
39	8	43 69 21 26 35 51 53 70
40	6	12 44 17 32 56 73
41	12	3 38 21 53 70 20 39 43 69 78 13 26
42	15	66 50 58 18 80 90 61 75 37 10 27 63 88 86 71
43	6	18 62 19 81 22 90
44	7	39 69 21 26 35 51 53
45	11	32 56 73 47 91 2 15 14 28 34 55
46	9	26 74 2 15 20 13 32 39 44
47	6	68 77 79 87 92 14
48	8	91 2 15 32 47 56 68 73
49	27	85 80 88 27 58 36 65 66 86 4 93 62 75 63 89 42 50 18 10 90 30 61 6 37 99 96 71
50	12	59 98 22 33 8 54 29 37 9 10 75 71
51	21	42 58 10 66 18 75 27 86 37 80 6 61 90 63 65 4 93 88 96 76 71
52	8	35 2 11 15 39 44 47 60
53	21	4 93 90 81 30 65 19 36 61 86 25 51 80 27 42 58 6 75 10 37 71
54	6	21 70 3 20 38 39
55	11	33 29 9 50 59 98 97 22 16 24 31
56	7	14 28 82 83 47 60 68
57	9	32 73 48 91 2 15 45 34 46
58	7	23 78 67 84 21 54 70
59	19	10 86 80 75 51 63 61 42 65 90 37 4 18 93 85 99 88 89 71
60	16	50 22 8 33 98 37 55 10 9 76 29 6 75 18 86 71
61	5	14 28 35 52 56
62	9	59 10 42 63 90 51 4 93 71
63	14	18 85 27 24 75 88 49 99 80 36 66 42 37 71
64	6	59 4 93 86 62 71
65	8	34 11 47 68 77 79 87 92
66	10	88 86 30 90 85 75 59 89 4 93
67	9	42 51 88 80 59 18 90 66 85
68	7	20 78 13 21 23 54 58
69	5	47 77 79 87 92
70	6	39 44 21 26 35 52
71	11	21 54 3 20 38 39 44 70 78 13 26
72	7	99 25 98 75 37 90 29
73	7	1 78 21 23 54 58 68
74	10	32 57 48 91 2 15 45 34 46 47
75	7	26 2 15 20 39 44 46
76	3	37 99 72

77	4	6 60 22 50
78	7	47 69 79 87 92 14 28
79	10	21 23 54 58 68 71 84 3 20 38
80	11	47 69 78 87 92 14 28 35 48 52 56
81	8	85 49 88 59 66 86 42 36
82	8	37 8 90 6 86 30 22 53
83	7	14 28 56 83 47 61 69
84	15	14 28 56 83 47 61 69 78 80 87 92 35 48 52 91
85	13	79 21 23 54 58 68 71 3 20 38 13 41 39
86	8	81 88 49 66 86 59 36 89
87	9	59 66 30 89 76 90 10 4 93
88	8	47 69 78 80 92 14 28 35
89	11	86 66 81 49 87 89 67 76 59 36 90
90	11	87 66 89 86 59 10 76 30 81 37 99
91	7	4 30 93 66 87 76 72
92	7	48 2 15 32 47 57 69
93	10	47 69 78 80 88 14 28 35 48 52
94	12	4 30 91 87 64 66 53 59 81 76 62 72
95	6	7 23 58 79 68 85
96	8	47 69 78 80 88 93 14 28
97	5	25 76 27 37 72
98	10	31 41 1 5 17 73 20 13 21 26
99	3	50 60 72
100	2	76 72

Average number of K been selected: 10

Sample ID	# Features	Feature Index
1	3	2 1 7
2	6	2 6 3 5 4 8
3	3	3 7 9
4	6	2 3 1 8 7 9
5	4	2 6 1 7
6	4	2 4 1 7
7	4	2 3 4 9

8	4	2 4 1 9
9	6	6 3 4 7 8 9
10	3	3 5 8
11	4	2 4 1 7
12	5	5 1 8 7 9
13	5	2 6 3 1 9
14	3	2 3 9
15	6	6 5 1 8 7 9
16	6	2 3 5 4 8 9
17	4	3 4 7 9
18	2	4 1
19	5	3 4 8 1 9
20	6	6 3 5 1 8 9
21	5	2 5 4 1 9
22	6	6 3 5 4 1 9
23	6	6 3 1 8 7 9
24	4	6 5 8 7
25	3	3 1 9
26	7	2 6 5 4 8 7 9
27	4	1 8 7 9
28	4	2 6 7 9
29	3	6 4 1
30	4	3 5 7 9
31	3	2 6 4
32	4	2 6 4 9
33	5	2 4 8 1 9
34	3	3 5 8
35	4	2 3 5 8
36	5	6 3 5 7 9
37	4	6 5 1 9
38	4	3 5 1 9
39	4	3 4 8 9
40	5	6 4 1 8 9
41	4	6 5 4 7
42	4	3 4 8 9
43	4	3 5 4 1
44	3	5 4 7
45	4	2 5 4 7
46	5	6 3 4 1 9

47	3	2 4 1
48	5	2 6 8 7 9
49	7	2 6 3 5 1 7 9
50	6	5 4 8 1 7 9
51	4	6 3 8 9
52	1	7
53	2	6 2
54	5	3 5 4 8 9
55	2	7 9
56	6	2 3 5 4 8 9
57	5	2 6 5 4 7
58	4	2 3 1 8
59	3	2 6 8
60	4	2 6 5 7
61	4	2 5 1 9
62	3	3 1 7
63	4	2 8 7 9
64	3	2 3 8
65	5	2 5 4 8 9
66	4	3 5 1 7
67	5	2 6 1 7 9
68	3	5 7 9
69	5	2 6 3 5 1
70	6	6 3 5 1 8 7
71	5	5 1 8 7 9
72	4	2 6 8 7
73	5	2 5 8 7 9
74	4	6 3 4 7
75	3	6 3 7
76	5	6 5 4 1 7
77	5	6 3 8 7 9
78	4	2 6 4 9
79	4	2 3 5 1
80	6	3 5 4 8 7 9
81	4	2 6 5 7
82	6	2 6 3 5 8 7
83	4	2 5 4 1
84	4	6 1 7 9
85	5	3 4 1 8 9

86	7	2 6 5 4 1 7 9
87	4	3 5 4 7
88	3	3 8 7
89	6	2 6 3 5 1 9
90	5	2 3 4 8 9
91	4	4 1 8 7
92	5	2 6 5 1 9
93	2	4 1
94	4	6 5 4 7
95	5	6 3 5 4 9
96	5	6 4 8 7 9
97	4	3 1 7 9
98	4	4 1 7 9
99	2	1 7
100	2	4 8

Average number of Features been selected: 5

==========End==========

APPENDIX C

Appendix C - Result of 100 Breast Cancer Wisconsin Samples Achieved by esnnGSA

========== Result ==========

Sample ID	Actual Class	Predicted Class
1	1	1
2	1	1
3	1	1
4	2	2
5	1	1
6	2	2
7	1	1
8	2	1*
9	2	2
10	2	2
11	1	1
12	1	1
13	1	1
14	1	1
15	1	1
16	2	2
17	1	1
18	2	2
19	2	2
20	1	1
21	1	1
22	2	2
23	1	1
24	2	2
25	2	2
26	1	1
27	2	2
28	1	1
29	2	2
30	2	2
31	1	1
32	1	1
33	2	2
34	1	1
35	1	1
36	2	2
37	2	2
38	1	1
39	1	1
40	1	1
41	1	1
42	2	2
43	2	2
44	1	1
45	1	1
46	1	1
47	1	1

48	1	1
49	2	2
50	2	2
51	2	2
52	1	1
53	2	2
54	1	1
55	2	2
56	1	1
57	1	1
58	1	1
59	2	2
60	2	2
61	1	1
62	2	2
63	2	2
64	2	2
65	1	1
66	2	2
67	2	2
68	1	1
69	1	1
70	1	1
71	1	1
72	1	1
73	1	1
74	1	1
75	1	1
76	2	2
77	2	2
78	1	1
79	1	1
80	1	1
81	2	2
82	2	2
83	1	1
84	1	1
85	1	1
86	2	2
87	2	2
88	1	1
89	2	2
90	2	2
91	2	2
92	1	1
93	1	1
94	2	2
95	1	1
96	1	1
97	2	2
98	2	2

99	2	2
100	2	2

Overall Accuracy of Leave-one-out Crossvalidation: 99.00%

Class 1 Accuracy: 100.00%
Class 2 Accuracy: 97.78%
Class1 Confusion Table: 55(Correctly Classified) 55(Total)
Class2 Confusion Table: 44(Correctly Classified) 45(Total)

Sample ID	Best Mod	Best Threshold	Best Sim
1	0.79	0.32	0.19
2	0.46	0.19	0.34
3	0.46	0.43	0.37
4	0.87	0.32	0.42
5	0.67	0.04	0.02
6	0.83	0.01	0.38
7	0.74	0.50	0.19
8	0.06	0.04	0.46
9	0.45	0.25	0.28
10	0.92	0.03	0.12
11	0.57	0.18	0.46
12	0.62	0.34	0.39
13	0.50	0.30	0.01
14	0.64	0.04	0.47
15	0.62	0.44	0.48
16	0.84	0.11	0.28
17	0.19	0.07	0.16
18	0.75	0.50	0.40
19	0.07	0.47	0.44
20	0.98	0.35	0.05
21	0.91	0.17	0.25
22	0.69	0.15	0.26
23	0.52	0.40	0.12
24	0.75	0.30	0.08
25	0.74	0.38	0.15
26	0.98	0.43	0.34
27	0.67	0.20	0.47
28	0.11	0.45	0.48
29	0.61	0.26	0.28
30	0.03	0.41	0.09
31	0.88	0.33	0.47
32	0.31	0.19	0.23
33	0.00	0.14	0.39
34	0.66	0.11	0.03
35	0.05	0.24	0.03

36	0.84	0.28	0.03
37	0.69	0.41	0.05
38	0.95	0.12	0.01
39	0.52	0.13	0.46
40	0.52	0.23	0.11
41	0.02	0.22	0.07
42	0.11	0.37	0.09
43	0.51	0.48	0.32
44	0.86	0.49	0.07
45	0.03	0.34	0.08
46	0.09	0.12	0.10
47	0.65	0.46	0.41
48	0.39	0.03	0.00
49	0.79	0.36	0.34
50	0.39	0.28	0.14
51	0.58	0.45	0.22
52	0.46	0.06	0.02
53	0.43	0.38	0.17
54	0.05	0.47	0.24
55	0.70	0.03	0.40
56	0.29	0.49	0.27
57	0.14	0.27	0.04
58	0.56	0.16	0.03
59	0.72	0.45	0.04
60	0.85	0.30	0.36
61	0.80	0.19	0.47
62	0.72	0.35	0.02
63	0.78	0.32	0.15
64	0.24	0.04	0.43
65	0.55	0.35	0.39
66	0.29	0.31	0.03
67	0.81	0.15	0.36
68	0.28	0.49	0.11
69	0.30	0.06	0.36
70	0.58	0.19	0.45
71	0.48	0.21	0.22
72	0.48	0.20	0.46
73	0.04	0.22	0.18
74	0.25	0.17	0.41
75	0.15	0.48	0.38
76	0.90	0.13	0.01
77	0.86	0.41	0.43
78	0.58	0.25	0.17
79	0.05	0.37	0.29
80	0.62	0.34	0.38
81	0.22	0.49	0.06
83	0.70	0.23	0.27
84	0.46	0.28	0.32
86	0.97	0.01	0.29
87	0.63	0.32	0.29
88	0.04	0.16	0.38

89	0.81	0.29	0.03
90	0.10	0.28	0.47
91	0.61	0.33	0.40
92	0.14	0.17	0.06
93	0.87	0.27	0.22
94	0.83	0.15	0.13
95	0.01	0.38	0.27
96	0.56	0.21	0.20
97	0.97	0.21	0.38
99	0.03	0.21	0.29
100	0.54	0.26	0.23

Average number of Mod: 0.52

Average number of Threshold: 0.27

Average number of Sim: 0.24

Sample ID	# K	KNN Index
1	11	72 20 53 70 2 19 37 38 43 69 78
2	9	14 25 34 47 51 74 91 10 31
3	9	37 20 40 53 70 19 38 43 69
4	12	93 29 90 86 63 65 52 58 80 75 61 71
5	2	6 15
6	7	76 9 59 26 50 7 36
7	22	94 5 3 37 78 20 22 40 53 57 67 70 84 19 12 10 38 43 69 1 25 74 1
8	16	59 21 49 36 98 75 81 86 8 32 6 9 76 99 17 71
9	8	28 54 36 98 59 49 32 71
10	19	58 86 75 98 6 50 36 59 61 49 80 21 89 32 17 41 65 90 71
11	9	34 51 2 14 38 43 46 60 68
12	11	39 2 14 31 45 56 73 25 47 74 91
13	13	19 20 25 53 67 70 74 3 37 38 43 45 69
14	11	27 55 82 83 46 60 68 77 79 87 92
15	17	2 25 34 47 51 74 91 11 31 38 43 45 46 56 60 68 69
16	17	5 7 40 3 37 1 72 94 78 13 20 22 53 57 70 84 97
17	8	46 68 77 79 87 92 14 27
18	10	23 50 75 86 41 58 21 62 36 71
19	2	16 5
20	14	13 20 25 53 67 70 74 3 37 38 43 45 69 78
21	10	53 70 3 20 37 38 43 69 78 13
22	19	59 49 8 32 36 98 18 76 10 75 86 9 54 81 6 23 99 28 71
23	8	57 78 67 84 21 53 70 94
24	3	18 32 71
25	2	71 99
26	14	74 2 15 20 38 43 45 69 13 21 34 51 53 70
27	11	35 75 50 36 96 6 41 48 80 18 88
28	11	14 55 82 83 46 60 68 77 79 87 92

29	5	9 54 98 49 71
30	13	65 4 93 90 86 75 25 36 99 58 88 89 71
31	20	20 45 13 26 40 67 74 21 53 70 2 3 15 37 38 43 69 78 84 97
32	6	56 73 47 91 2 15
33	15	49 59 54 22 98 24 9 10 29 8 97 18 16 36 71
34	15	47 64 91 2 11 15 32 46 56 68 73 77 79 87 92
35	13	51 2 11 15 38 43 46 60 68 69 77 79 87
36	22	75 27 36 85 80 65 88 30 90 6 48 86 25 58 99 10 4 41 93 96 81 50
37	6	75 99 8 25 98 71
38	6	3 21 40 53 70 20
39	11	43 69 21 26 35 51 53 70 74 2 3
40	8	12 44 17 32 56 73 47 91
41	12	3 38 21 53 70 20 39 43 69 78 13 26
42	15	66 50 58 18 80 90 61 75 37 10 27 63 88 86 71
43	11	18 62 19 81 22 90 86 85 52 4 93
44	12	39 69 21 26 35 51 53 70 74 2 3 11
45	7	32 56 73 47 91 2 15
46	8	26 74 2 15 20 13 32 39
47	8	68 77 79 87 92 14 28 35
48	12	91 2 15 32 47 56 68 73 77 79 87 92
49	27	85 80 88 27 58 36 65 66 86 4 93 62 75 63 89 42 50 18 10 90 30 61 6 37 99 96 71
50	12	59 98 22 33 8 54 29 37 9 10 75 71
51	21	42 58 10 66 18 75 27 86 37 80 6 61 90 63 65 4 93 88 96 76 71
52	10	35 2 11 15 39 44 47 60 68 69
53	21	4 93 90 81 30 65 19 36 61 86 25 51 80 27 42 58 6 75 10 37 71
54	6	21 70 3 20 38 39
55	11	33 29 9 50 59 98 97 22 16 24 31
56	7	14 28 82 83 47 60 68
57	6	32 73 48 91 2 15
58	15	23 78 67 84 21 54 70 94 3 20 38 13 41 1 39
59	19	10 86 80 75 51 63 61 42 65 90 37 4 18 93 85 99 88 89 71
60	16	50 22 8 33 98 37 55 10 9 76 29 6 75 18 86 71
61	6	14 28 35 52 56 82
62	9	59 10 42 63 90 51 4 93 71
63	14	18 85 27 24 75 88 49 99 80 36 66 42 37 71
64	6	59 4 93 86 62 71
65	9	34 11 47 68 77 79 87 92 14
66	25	88 86 30 90 85 75 59 89 4 93 80 37 36 10 99 66 51 49 64 25 81 42 18 27 71
67	24	42 51 88 80 59 18 90 66 85 75 49 27 10 37 86 62 36 64 63 99 4 93 25 71
68	6	20 78 13 21 23 54
69	7	47 77 79 87 92 14 28
70	12	39 44 21 26 35 52 54 70 74 2 3 11
71	9	21 54 3 20 38 39 44 70 78
72	13	99 25 98 75 37 90 29 16 59 64 9 86 18
73	12	1 78 21 23 54 58 68 71 84 3 5 20
74	8	32 57 48 91 2 15 45 34
75	8	26 2 15 20 39 44 46 70
76	3	37 99 72

77	25	6 60 22 50 37 10 8 51 81 76 98 27 18 86 96 29 9 59 55 33 89 25 66 36 72
78	16	47 69 79 87 92 14 28 35 48 52 56 82 83 91 95 2
79	9	21 23 54 58 68 71 84 3 20
80	9	47 69 78 87 92 14 28 35 48
81	11	85 49 88 59 66 86 42 36 67 76 4
82	12	37 8 90 6 86 30 22 53 76 19 60 66
83	18	14 28 56 83 47 61 69 78 80 87 92 35 48 52 91 95 2 11
84	15	26 65 75 46 21 54 71 3 20 38 13 41 1 84 79
85	12	92 48 91 95 34 17 32 45 57 74 65 97
86	25	81 88 49 66 86 59 36 89 76 63 90 4 30 93 67 18 64 27 42 10 37 99 51 62 72
87	19	59 66 30 89 76 90 10 4 93 37 88 64 81 86 18 99 51 8 72
88	8	47 69 78 80 92 14 28 35
89	9	86 66 81 49 87 89 67 76 59
90	26	87 66 89 86 59 10 76 30 81 37 99 18 64 49 22 90 4 93 36 8 60 98 50 51 77 72
91	7	4 30 93 66 87 76 72
92	9	48 2 15 32 47 57 69 74 78
93	7	47 69 78 80 88 14 28
94	12	4 30 91 87 64 66 53 59 81 76 62 72
95	11	7 23 58 79 68 85 21 54 71 3 20
96	8	47 69 78 80 88 93 14 28
97	5	25 76 27 37 72
98	8	31 41 1 5 17 73 20 13
99	3	50 60 72
100	2	76 72

Average number of K been selected: 12

Sample ID	# Features	Feature Index
1	5	2 3 4 1 9
2	5	6 3 5 7 9
3	3	5 1 9
4	5	2 5 4 1 9
5	8	2 6 5 4 1 8 7 9
6	4	6 5 8 9
7	4	3 1 8 9
8	6	2 5 4 1 7 9
9	2	4 9
10	4	5 4 8 7
11	3	2 4 1
12	2	3 4
13	8	2 6 5 4 1 8 7 9
14	3	3 4 7
15	5	3 5 4 1 9
16	7	2 3 5 4 8 7 9

17	3	6 1 9
18	7	6 3 5 4 8 7 9
19	6	2 5 4 8 1 7
20	2	1 8
21	7	2 3 5 1 8 7 9
22	5	6 3 5 8 7
23	6	2 6 3 1 8 7
24	3	2 8 9
25	7	2 6 4 1 8 7 9
26	5	3 1 8 7 9
27	2	5 4
28	3	4 1 9
29	3	2 5 7
30	5	2 3 4 1 8
31	2	2 5
32	7	2 6 3 4 1 7 9
33	7	2 6 3 5 1 7 9
34	6	2 3 5 4 1 9
35	6	6 3 4 8 7 9
36	3	3 4 8
37	6	5 4 1 8 7 9
38	5	3 4 1 8 7
39	4	6 4 1 8
40	3	2 4 7
41	5	3 4 1 8 9
42	4	2 6 4 7
43	5	3 5 4 1 7
44	5	2 6 4 7 9
45	8	2 6 3 5 4 1 8 9
46	6	2 6 3 8 7 9
47	5	5 4 8 7 9
48	2	2 4
49	5	2 6 3 8 9
50	4	2 6 1 7
51	3	6 3 1
52	5	2 5 4 1 8
53	7	6 2 5 4 1 7 9
54	4	3 1 8 9
55	4	2 5 1 7
56	7	2 6 5 4 8 7 9
57	6	2 6 3 4 8 9
58	3	2 3 9
59	4	2 3 5 4
60	6	6 3 5 8 1 7
61	4	3 1 8 9
62	5	6 5 1 8 9
63	3	6 5 8
64	5	6 1 8 7 9
65	5	6 5 4 1 9
66	5	6 5 4 8 9
67	5	6 4 1 7 9

68	5	2 3 8 7 9
69	5	2 3 1 7 9
70	3	2 3 9
71	6	2 6 3 1 7 9
72	6	6 3 8 4 7 9
73	5	2 4 1 8 9
74	4	2 3 4 9
75	7	6 3 5 4 1 8 7
76	5	6 5 1 7 9
77	8	2 6 3 5 4 8 1 7
78	6	2 4 1 8 7 9
79	5	2 3 4 7 9
80	5	6 3 5 7 9
81	4	3 5 8 9
82	5	3 5 4 1 7
83	5	2 6 3 4 1
84	1	1
85	3	2 3 4
86	4	1 8 7 9
87	8	2 6 3 5 4 1 8 9
88	2	4 8
89	7	2 6 3 4 1 8 7
90	2	3 8
91	5	2 6 4 7 9
92	7	6 3 4 1 8 7 9
93	8	2 6 3 5 4 1 8 7
94	5	6 5 4 8 7
95	4	3 5 1 8
96	1	3
97	4	2 3 7 9
98	5	2 3 4 1 7
99	7	6 3 5 4 1 7 9
100	6	6 4 1 8 7 9

Average number of Features been selected: 5

========== End ==========

APPENDIX D

Appendix D - Geriatric Depression Scale: Short Form

Geriatric Depression Scale: Short Form

Choose the best answer for how you have felt over the past week:

1. Are you basically satisfied with your life? YES / NO

2. Have you dropped many of your activities and interests? YES / NO

3. Do you feel that your life is empty? YES / NO

4. Do you often get bored? YES / NO

5. Are you in good spirits most of the time? YES / NO

6. Are you afraid that something bad is going to happen to you? YES / NO

7. Do you feel happy most of the time? YES / NO

8. Do you often feel helpless? YES / NO

9. Do you prefer to stay at home, rather than going out and doing new things? YES / NO

10. Do you feel you have more problems with memory than most? YES / NO

11. Do you think it is wonderful to be alive now? YES / NO

12. Do you feel pretty worthless the way you are now? YES / NO

13. Do you feel full of energy? YES / NO

14. Do you feel that your situation is hopeless? YES / NO

15. Do you think that most people are better off than you are? YES / NO

APPENDIX E

Appendix E - 40 Samples used for Chapter 9

ID=263	age=60																													
Stroke																														
13	13.5	12.3999939	13.5	13.5	14.3999939	14.3999939	14.29999542	14.29999542	11.39999695	12.5	14	14	10.29999542	11.5	11.5	11.79999542	11.79999542	12	12.79999542	12.79999542	12.69999695	12	12	11.3999939	11.3999939	11.3999939	12.59999847	12.59999847	13.29999542
74	87.8906	75.5	78.09375	78.09375	87.09375	87.09375	86.5	86.5	80.796875	72.1875	86.09375	86.09375	83.5	76.6875	76.6875	84.09375	84.09375	78.796875	80.390625	80.390625	83.890625	82.890625	82.890625	78.5	78.5	78.5	77.796875	77.796875	74.6875
999.4	995.1	1005.7	1007.2	1007.2	1005.6	1005.6	1009.3	1009.3	1012.5	1013.1	1016.9	1016.9	1025.3	1020.9	1020.9	1015.9	1015.9	1015.9	1016.9	1016.9	1019.7	1023.5	1023.5	1009.4	1009.4	1009.4	1011.5	1011.5	1014.7
12.52734375	6.08301	7.944335938	6.75	6.75	5.583007813	5.583007813	6.52734375	6.52734375	5.944335938	6.75	6.916015625	6.916015625	2.138671875	2.444335938	2.444335938	7.8046875	7.8046875	7.610351563	6.221679688	6.221679688	7.5	6.388671875	6.388671875	6.971679688	6.971679688	6.971679688	8.193359375	8.193359375	8.8046875
4.552038568	9.54961	6.935700097	9.0472875	9.047288	10.9083593	10.9083593	10.0943474	10.0943474	7.148500329	7.9424125	9.477945665	9.477945665	9.483099115	10.43380136	10.43380136	6.364034282	6.364034282	6.734445922	8.677692237	8.677692237	7.600623586	7.670756029	7.670756029	6.55392222	6.55392222	6.55392222	6.979612859	6.979612859	7.349620525
Normal																														
11.59999867	11.6	9.099998474	9.099998474	8.199997	8.199998968	9.099998474	9.699996948	11.3999939	12.8999939	12.59999847	10.19999695	10.19999695	11.5	12.09999847	11	11	11.3999939	11.3999939	11.3999939	11.79999542	10.79999542	10.79999542	8.599998474	8.599998474	8.599998474	10.29999542	10.29999542	11.69999695	11.69999695
85	85	79.6875	79.6875	75.09375	75.09375	69.6875	79.1875	76.390625	87.09375	86.890625	92.09375	92.09375	73.1875	87.296875	86.6875	86.6875	84	84	84	87.890625	80.296875	80.296875	81.1875	81.1875	81.1875	80.5	80.5	90.296875	90.296875
1009.1	1009.1	1018.8	1018.8	1019.3	1019.3	1018.7	1018.3	1014	999.6	1000.9	1006.5	1006.5	1012.1	996.6	1006.4	1006.4	1010.7	1010.7	1010.7	998.4	1010.3	1010.3	1006.7	1006.7	1006.7	995	995	995.2	995.2
5.971679688	5.97168	3.5	3.5	2.277344	2.27734375	4.110351563	3.166503906	5.27734375	6.721679688	5	3.638671875	3.638671875	5.471679688	4.666015625	4.666015625	4.666015625	3.360839844	3.360839844	3.360839844	4.388671875	4.138671875	4.138671875	3.138671875	3.138671875	3.138671875	4.5546875	4.5546875	3.52734375	3.52734375
7.562624883	7.56262	7.002123406	7.002123406	7.209507	7.209506543	6.453245434	7.92908008	7.90805521	8.405774445	9.423493363	8.028652637	8.028652637	7.857255111	9.160544917	7.987593126	7.987593126	9.525559895	9.525559895	9.525559895	9.07124496	8.224159786	8.224159786	6.812380031	6.812380031	6.812380031	7.336584166	7.336584166	9.694999636	9.694999636

ID=1652	age=60																													
Stroke																														
16.3999939	16.4	16.3999939	16.29998779	16.29999	16.29998779	16.29998779	16.29998779	14.69999695	14.69999695	14.69999695	14.69999695	14.69999695	16.69999695	16.69999695	16.79998779	16.79998779	16.79998779	16.79998779	16.79998779	15.79999542	15.79999542	15.79999542	15.29999542	12.8999939	12.8999939	12.8999939	12.8999939	12.8999939	15.5
75	75	75	74.09375	74.09375	74.09375	74.09375	74.09375	73.390625	73.390625	73.390625	73.390625	72.890625	72.890625	72.890625	82.390625	82.390625	82.390625	82.390625	82.390625	76.796875	76.796875	76.796875	68.59375	81.5	81.5	81.5	81.5	81.5	76.390625
1019	1019	1019	1018.5	1018.5	1018.5	1018.5	1018.5	1019.3	1019.3	1019.3	1019.3	1014.2	1014.2	1014.2	1014.1	1014.1	1014.1	1014.1	1014.1	1015.2	1015.2	1015.2	1016.1	1014.6	1014.6	1014.6	1014.6	1014.6	1019.3
6.416015625	6.41602	6.416015625	5.02734375	5.027344	5.02734375	5.02734375	5.02734375	2.110839844	2.110839844	2.110839844	2.110839844	7.583007813	7.583007813	7.583007813	9.470703125	9.470703125	9.470703125	9.470703125	9.470703125	9.666015625	9.666015625	9.666015625	7.75	4.471679688	4.471679688	4.471679688	4.471679688	4.471679688	4.75
12.48365146	12.4837	12.48365146	13.37160887	13.37161	13.37160887	13.37160887	13.37160887	13.99249388	13.99249388	13.99249388	13.99249388	12.02033666	12.02033666	12.02033666	10.9247317	10.9247317	10.9247317	10.9247317	10.9247317	9.646015946	9.646015946	9.646015946	10.33690736	10.17101747	10.17101747	10.17101747	10.17101747	10.17101747	12.7223375
Normal																														
12.19999695	12.2	12.19999695	12.19999695	15.8	15.79999542	15.79999542	17.79998779	17.79998779	17.79998779	19.29998779	19.29998779	19.29998779	19.29998779	19.29998779	19.29998779	19.3999939	19.3999939	19.3999939	19.3999939	18.8999939	18.8999939	18.8999939	18.8999939	18.5	18.5	16.79998779	16.79998779	16.79998779	15.8999939	15.8999939
71.890625	71.8906	71.890625	71.890625	76.09375	76.09375	76.09375	90	90	90	89.5	89.5	89.5	89.5	89.5	89.5	89.390625	89.390625	89.390625	89.390625	83.890625	83.890625	83.890625	83.890625	76.6875	76.6875	81.6875	81.6875	81.6875	77	77
1016.9	1016.9	1016.9	1016.9	1009.7	1009.7	1009.7	1012.6	1012.6	1012.6	1012.8	1012.8	1012.8	1012.8	1012.8	1014	1014	1014	1014	1017.1	1017.1	1017.1	1017.1	1023.3	1023.3	1026	1026	1026	1024.2	1024.2
1.416503906	1.4165	1.416503906	1.416503906	7.660016	7.660015625	7.660015625	2.5	2.5	2.5	2.722167969	2.722167969	2.722167969	2.722167969	2.722167969	4.388671875	4.388671875	4.388671875	4.388671875	5.888671875	5.888671875	5.888671875	5.888671875	7.221679688	7.221679688	5.583007813	5.583007813	5.583007813	2.02734375	2.02734375
12.06460472	12.0646	12.06460472	12.06460472	10.95558	10.95558174	10.95558174	16.84973747	16.84973747	16.84973747	18.22466478	18.22466478	18.22466478	18.22466478	18.22466478	17.13628061	17.13628061	17.13628061	17.13628061	15.57831882	15.57831882	15.57831882	15.57831882	14.26672103	14.26672103	13.50823823	13.50823823	13.50823823	15.28352904	15.28352904

ID=111	age=62																											
Stroke																												
12.29999542	14.4	14.3999939	16.59999084	16.59999	16.59999084	15.29999542	15.29999542	11.59999847	11.59999847	11.59999847	9.799995422	9.799995422	9.799995422	10.79999542	10.79999542	12.09999847	11.59999847	11.59999847	10.19999695	9.599998474	10.3999939	15.3999939	15.3999939	15.3999939	15.3999939	13.8999939	12.3999939	12.3999939	15.29999542
95	84.6875	84.6875	84.59375	84.59375	95.59375	94.5	94.5	89.6875	89.6875	89.6875	91.390625	91.390625	91.390625	84.796875	84.796875	82.890625	87.59375	87.59375	83.1875	80.390625	80.390625	86.890625	86.890625	86.890625	86.890625	86.390625	92.796875	92.796875	88.890625
1006	1004.6	1004.6	1003	1003	1015.5	1018.3	1018.3	1019.6	1019.6	1019.6	1017.9	1017.9	1017.9	1016.9	1016.9	1018.8	1021.9	1021.9	1025.5	1024.6	1018.7	1003	1003	1003	1003	1018.7	1023	1023	996.8
3.944331938	4.13867	4.138671875	6.166015625	6.160216	3.138671875	3.416503906	3.416503906	1.083251953	1.083251953	1.083251953	0.722167969	0.722167969	0.722167969	2.138671875	2.138671875	4.8046875	5.471679688	5.471679688	1.916503906	2.694335938	1.972167969	6.5546875	6.5546875	6.5546875	6.5546875	2.610839844	1.416503906	1.416503906	4.666015625
9.971127009	12.0278	12.02779372	12.87810811	12.87811	15.11689555	13.5457981	13.5457981	11.76627617	11.76627617	11.76627617	10.31223226	10.31223226	10.31223226	9.992881183	9.992881183	9.046715063	7.965376069	7.965376069	9.586208691	8.255711264	9.73800887	11.28562958	11.28562958	11.28562958	11.28562958	12.7540997	12.26584271	12.26584271	12.57276979
Normal																												
12.19999695	12.2	14.59999847	17.19999695	17.2	17.19999695	17.59999084	15.8999939	15.8999939	14.5	13.79999542	13.69999695	13.69999695	13.69999695	13.19999695	14.5	16.8999939	16.19999695	16.19999695	15.3999939	15.3999939	15.3999939	15.8999939	15.8999939	15.8999939	16.59999084	16.59999084	16.59999084	14.69999695	15.8999939
72.890625	72.8906	79.890625	90.5	90.5	90.5	95.5	75.390625	75.390625	84.6875	82.296875	85.296875	85.296875	85.296875	85.890625	82.890625	82.796875	76.5	76.5	79.59375	79.59375	79.59375	72.5	72.5	72.5	71.390625	71.390625	71.390625	80.390625	85.59375
1017.1	1017.1	1017.1	1015.4	1015.4	1015.4	1015.5	1022.1	1022.1	1022.5	1024	1025.4	1025.4	1025.4	1026	1023.5	1017.6	1016.5	1016.5	1015.1	1015.1	1015.1	1013.2	1013.2	1013.2	1014.5	1014.5	1014.5	1017.4	1017
8.109375	8.10938	2.972167969	2.02734375	2.027344	2.02734375	4.666015625	7.583007813	7.583007813	3.305175781	0.416625977	0.555541992	0.555541992	0.555541992	0.333312988	2.52734375	4.5546875	3.972167969	3.972167969	3.52734375	3.52734375	3.52734375	5.52734375	5.52734375	5.52734375	2.388671875	2.388671875	2.388671875	0.666625977	3.02734375
6.588637079	6.58864	13.17666071	16.60628976	16.60629	16.60628976	15.02530387	11.12410473	11.12410473	12.80145906	14.55839306	14.33508627	14.33508627	14.33508627	14.04145994	13.44109111	14.36070989	14.05644501	14.05644501	13.56244313	13.56244313	13.56244313	12.57392708	12.57392708	12.57392708	15.70666638	15.70666638	15.70666638	15.22817696	14.47835705

ID=321	age=64																											
Stroke																												
14.09999847	14.1	14.09999847	14.09999847	13.6	13.09999847	13.09999847	13.09999847	14.29999542	14.29999542	15	15	15	15	15	10.8999939	10.8999939	10.8999939	11.59999847	11.59999847	11.59999847	13.3999939	13.3999939	14.59999847	16.69999695	15.09999847	14	14	13.8999939	14.09999847
81.296875	81.2969	81.296875	81.296875	82.39063	72.890625	72.890625	72.890625	79.890625	79.890625	93.5	93.5	93.5	93.5	93.5	63.3984375	63.3984375	63.3984375	65.296875	65.296875	65.296875	69.6875	69.6875	89.59375	87.09375	86.1875	80.5	80.5	80.890625	79.296875
1019.4	1019.4	1019.4	1019.4	1014.8	1017.7	1017.7	1017.7	1015.9	1015.9	1017.9	1017.9	1017.9	1017.9	1017.9	1021.9	1021.9	1021.9	1025.3	1025.3	1025.3	1015.8	1015.8	995.7	1015.6	1022.4	1026	1026	1023.9	1021.1
4.5	4.5	4.5	4.5	10.02734	9.693359375	9.693359375	9.693359375	9.638671875	9.638671875	7.333007813	7.333007813	7.333007813	7.333007813	7.333007813	5.5	5.5	5.5	7.75	7.75	7.75	12.25	12.25	7.166015625	6.916015625	5.944335938	2.833007813	2.833007813	1.555419922	2.02734375
11.42392338	11.4239	11.42392338	11.42392338	6.858895	6.49861936	6.49861936	6.49861936	7.925778168	7.925778168	10.29073983	10.29073983	10.29073983	10.29073983	10.29073983	7.185168398	7.185168398	7.185168398	6.180423286	6.180423286	6.180423286	5.205042635	5.205042635	9.963391038	12.46939726	11.39938583	12.67131552	12.67131552	13.65457183	13.45202315
Normal																												
12.79999542	11.7	11.69999695	11.69999695	11.7	14.59999847	13	13	13.29999542	12.69999695	12.69999695	12.69999695	13.5	13.5	13.5	13.5	13.09999847	13.09999847	13.59999847	13.59999847	15.69999695	14.69999695	15.8999939	15.8999939	12.79999542	12.79999542	12.79999542	13.29999542	13	13.5
71.09375	79.5938	79.59375	79.59375	79.59375	70	73.390625	82.1875	73.796875	84.5	84.5	84.5	79.5	79.5	79.5	79.5	74.5	74.5	82.890625	82.890625	88.296875	82	78.890625	78.890625	75.390625	75.390625	75.390625	74.6875	74	87.890625
1018.3	1006.8	1006.8	1006.8	1006.8	1015.6	1023.5	1025.7	1022.3	1024.4	1024.4	1024.4	1022	1022	1022	1022	1024.4	1024.4	1012.5	1012.5	1019.7	1025.8	1026.1	1026.1	1021.7	1021.7	1021.7	1014.7	998.4	995.1
3.583007813	7.36035	7.360351563	7.360351563	7.360352	4.8046875	4.221679688	2.583007813	4.971679688	3.444335938	3.444335938	3.444335938	7.833007813	7.833007813	7.833007813	7.833007813	6.916015625	6.916015625	11.77714375	11.77714375	7.360351563	3.083007813	5.0546875	5.0546875	4.944335938	4.944335938	4.944335938	8.8046875	12.52734375	6.083007813
10.7981869	6.58818	6.588183385	6.588183385	6.588183	11.71895186	10.48072215	11.85249622	10.19668331	10.81003196	10.81003196	10.81003196	8.255153164	8.255153164	8.255153164	8.255153164	8.480792314	8.480792314	5.735856662	5.735856662	11.0528494	13.18982695	12.92210124	12.92210124	9.682631563	9.682631563	9.682631563	7.340620525	4.552038568	9.549613125

ID=101	age=66

Stroke

11.59999847	9.8	9.799995422	9.799995422	10.8	10.79999542	12.09999847	11.59999847	11.59999847	10.19999695	9.599998474	10.3999939	15.3999939	15.3999939	15.3999939	15.3999939	13.8999939	12.3999939	12.3999939	10.8999939	12.19999695	12.19999695	14.59999847	17.19999695	17.19999695	17.19999695	17.59999084	15.8999939	15.8999939	11.59999847
89.6875	91.3906	91.390625	91.390625	84.79688	84.796875	82.890625	87.59375	87.59375	83.1875	80.390625	80.390625	86.890625	86.890625	86.890625	86.890625	86.390625	92.796875	92.796875	85.1875	72.890625	72.890625	79.890625	90.5	90.5	90.5	95.5	75.390625	75.390625	89.6875
1019.6	1017.9	1017.9	1017.9	1016.9	1016.9	1018.8	1021.9	1021.9	1025.5	1024.6	1018.7	1003	1003	1003	1003	1018.7	1023	1023	1025.2	1017.1	1017.1	1017.1	1015.4	1015.4	1015.4	1015.5	1022.1	1022.1	1019.6
1.083251953	0.72217	0.722167969	0.722167969	2.138672	2.138671875	4.8046875	5.471679688	5.471679688	1.916503906	2.694335938	1.972167969	6.5546875	6.5546875	6.5546875	6.5546875	2.610839844	1.416503906	1.416503906	0.472167969	8.109375	8.109375	2.972167969	2.02734375	2.02734375	2.02734375	4.666015625	7.583007813	7.583007813	1.083251953
11.76627617	10.3122	10.31223226	10.31223226	9.992882	9.99288183	9.046735063	7.965376069	7.965376069	9.586208691	8.255711264	9.73800887	11.28562958	11.28562958	11.28562958	11.28562958	12.7540997	12.26584271	12.26584271	11.64131194	6.588637079	6.588637079	13.17666071	16.60628976	16.60628976	16.60628976	15.02530387	11.12410473	11.12410473	11.76627617

Normal

13.79999542	13.7	13.69999695	13.69999695	13.2	14.5	16.8999939	16.19999695	16.19999695	15.3999939	15.3999939	15.3999939	15.8999939	15.8999939	15.8999939	15.8999939	16.59999084	16.59999084	16.59999084	16.69999695	15.8999939	15.8999939	16.29998779	16.29998779	16.59999084	16.59999084	17.5	17.59999084	17.59999084	17.59999084	18.5
82.296875	85.2969	85.296875	85.296875	85.89063	82.890625	82.796875	76.5	76.5	79.59375	79.59375	79.59375	72.5	72.5	72.5	72.5	71.390625	71.390625	71.390625	80.390625	85.59375	85.59375	84.390625	84.390625	68.890625	68.890625	73.59375	84.1875	84.1875	83	87.59375
1024	1025.4	1025.4	1025.4	1026	1023.5	1017.6	1016.5	1016.5	1015.1	1015.1	1015.1	1013.2	1013.2	1013.2	1014.5	1014.5	1014.5	1017.4	1017	1017	1017.8	1017.8	1023	1023	1015.6	1010.5	1010.5	1007.1	1012.2
0.416625977	0.55554	0.555541992	0.555541992	0.333313	2.52734375	4.5546875	3.972167969	3.972167969	3.52734375	3.52734375	3.52734375	5.52734375	5.52734375	5.52734375	2.388671875	2.388671875	2.388671875	0.666625977	3.02734375	3.02734375	1.194335938	1.194335938	6.694335938	6.694335938	7.194335938	3.166503906	3.166503906	2.222167969	10.94335938
14.55839306	14.3351	14.33508627	14.33508627	14.04146	13.44039111	14.36070989	14.05644501	14.05644501	13.56244313	13.56244313	13.56244313	12.57392708	12.57392708	12.57392708	15.70666638	15.70666638	15.70666638	15.70666638	15.22817696	14.47835705	14.47835705	16.37400925	16.37400925	12.51069736	12.51069736	13.17115292	16.13385829	16.13385829	16.86060094	12.05416937

ID=160	age=66

Stroke

9.099998474	11.8	11.79999542	12.79999542	13.2	13.19999695	13.3999939	13.3999939	9.899993896	9.899993896	11.59999847	11.59999847	13	13	13	13.79999542	15.3999939	6.5	8.099998474	10.5	9.099998474	9.099998474	9.099998474	9.399993896	9.399993896	9.399993896	9.399993896	9.399993896	12.09999847	9.099998474
85.59375	84.6875	84.6875	78	88.1875	88.1875	89.09375	89.09375	92	92	87.6875	87.6875	82.59375	82.59375	82.59375	91.5	94.1875	90.59375	83.09375	72.390625	76.296875	76.296875	76.296875	92.59375	92.59375	92.59375	92.59375	92.59375	82.1875	85.59375
1002.3	1000.7	1000.7	1002.2	1001.8	1001.8	1011	1011	1015.9	1015.9	1019.3	1019.3	1006.1	1006.1	1006.1	1002.9	1014.4	1031.3	1026.9	1019	1016.1	1016.1	1016.1	1017.3	1017.3	1017.3	1017.3	1017.3	1018.1	1002.3
3.555175781	4.3069	4.3046875	6.083007813	4.416016	4.416015625	5.360351563	5.360351563	4	4	2.77734375	2.77734375	6.610351563	6.610351563	6.610351563	4.444335938	7.110351563	1	2.360839844	7.77734375	5.5	5.5	5.5	2.02734375	2.02734375	2.02734375	2.02734375	2.02734375	5	3.555175781
6.952124997	9.14144	9.141437703	8.784833173	10.53479	10.53479242	9.998389824	9.998389824	7.394993567	7.394993567	10.24239207	10.24239207	8.600400708	8.600400708	8.600400708	11.14924205	10.89241189	6.75065	7.027586593	6.923515326	5.238023349	5.238023349	5.238023349	8.669740952	8.669740952	8.669740952	8.669740952	8.669740952	8.887248363	6.952124997

Normal

11.59999847	11.6	12.09999847	12.09999847	13.5	13.5	13.5	12.5	10.09999847	11.59999847	11.59999847	11.59999847	11.59999847	12.09999847	12.09999847	13.59999847	15.29999542	15.29999542	15.29999542	12.29999542	14.3999939	14.3999939	16.59999084	16.59999084	16.59999084	15.29999542	15.29999542	11.59999847	11.59999847	11.59999847
80.09375	80.0938	85.796875	85.796875	94.29688	94.296875	94.296875	96.59375	89.59375	82.890625	82.890625	82.890625	82.890625	86.59375	86.59375	88.296875	88.890625	88.890625	88.890625	95	84.6875	84.6875	84.59375	84.59375	95.59375	94.5	94.5	89.6875	89.6875	89.6875
1012.5	1012.5	1005.9	1005.9	1001.4	1001.4	1001.4	1001.8	1011	1007.7	1007.7	1007.7	1007.7	1003.2	1003.2	997.5	996.8	996.8	996.8	1006	1004.6	1004.6	1003	1003	1015.5	1018.3	1018.3	1019.6	1019.6	1019.6
4.416015625	4.41602	3.972167969	3.972167969	2.944336	2.944335938	2.944335938	1	3.610839844	3.833007813	3.833007813	3.833007813	3.833007813	2.583007813	2.583007813	2.388671875	4.666015625	4.666015625	4.666015625	3.944335938	4.138671875	4.138671875	6.166015625	6.166015625	3.138671875	3.416503906	3.416503906	1.083251953	1.083251953	1.083251953
8.836079982	8.83608	9.73715768	9.73715768	12.06153	12.06152656	12.06152656	12.74165	7.948561167	9.328786961	9.328786961	9.328786961	9.328786961	10.92748158	10.92748158	12.6245029	12.57276979	12.57276979	12.57276979	9.971127009	12.02779372	12.02779372	12.87810811	12.87810811	15.11689555	13.5457981	13.5457981	11.76627617	11.76627617	11.76627617

ID=268	age=66

Stroke

15.8999939	12.8	12.76999542	12.76999542	13.3	13	13.5	12.3999939	13.5	13.5	14.3999939	14.3999939	14.29999542	14.29999542	11.19999695	12.5	14	14	10.29999542	11.5	11.5	11.79999542	11.79999542	12	12.79999542	12.79999542	12.69999695	12	12	15.8999939
78.890625	75.3906	75.390625	75.390625	74.6875	74	87.890625	75.5	78.09375	78.09375	87.09375	87.09375	86.5	86.5	80.796875	72.1875	86.09375	86.09375	83.5	76.6875	76.6875	84.09375	84.09375	78.796875	80.390625	80.390625	83.890625	82.890625	82.890625	78.890625
1026.1	1021.7	1021.7	1021.7	1014.7	999.4	995.1	1005.7	1007.2	1007.2	1005.6	1005.6	1009.3	1009.3	1012.5	1013.1	1016.9	1016.9	1025.3	1020.9	1020.9	1015.9	1015.9	1015.9	1016.9	1016.9	1019.7	1023.5	1023.5	1026.1
5.0546875	4.94434	4.944335938	4.944335938	8.804688	12.52734375	6.083007813	7.944335938	6.75	6.75	5.583007813	5.583007813	6.527734375	6.527734375	5.944335938	6.75	6.916015625	6.916015625	2.138671875	2.444335938	2.444335938	7.8046875	7.8046875	7.610351563	6.221679688	6.221679688	7.5	6.388671875	6.388671875	5.0546875
12.92210124	9.68263	9.682631563	9.682631563	7.340621	4.552038568	9.549613125	6.935700097	9.0472875	9.0472875	10.9083593	10.9083593	10.0943474	10.0943474	7.148500329	7.9426125	9.477945665	9.477945665	9.483099115	10.43380136	10.43380136	6.364034282	6.364034282	6.734445922	8.677692237	8.677692237	7.600621586	7.670756029	7.670756029	12.92210124
Normal																													
11.3999939	11.4	12.59999847	12.59999847	12.6	11.59999867	11.59999867	9.099998474	9.099998474	8.199996948	8.199996948	9.099998474	9.699996948	11.3999939	12.8999939	12.59999847	10.19999695	10.19999695	11.5	12.09999847	11	11	11.3999939	11.3999939	11.3999939	11.79999542	10.79999542	10.79999542	8.599998474	8.599998474
78.5	78.5	77.796875	77.796875	77.79688	85	85	79.6875	79.6875	75.09375	75.09375	69.6875	79.1875	76.390625	87.09375	86.890625	92.09375	92.09375	73.1875	87.296875	86.6875	86.6875	84	84	84	87.890625	80.296875	80.296875	81.1875	81.1875
1009.4	1009.4	1011.5	1011.5	1011.5	1009.1	1009.1	1018.8	1018.8	1019.3	1019.3	1018.7	1018.3	1014	999.6	1000.9	1006.5	1006.5	1012.1	998.6	1006.4	1006.4	1010.7	1010.7	1010.7	998.4	1010.3	1010.3	1006.7	1006.7
6.971679688	6.97168	8.193359375	8.193359375	8.193359	5.971679688	5.971679688	3.5	3.5	2.2734375	2.2734375	4.110351563	3.166503906	5.2734375	6.721679688	5	3.638671875	3.638671875	5.471679688	4.666015625	4.666015625	4.666015625	3.360839844	3.360839844	3.360839844	4.388671875	4.138671875	4.138671875	3.138671875	3.138671875
6.553922222	6.55392	6.979612859	6.979612859	6.979613	7.562624883	7.562624883	7.002123406	7.002123406	7.209506543	7.209506543	6.452245434	7.92908008	7.90805631	8.405774445	9.423498363	8.028652637	8.028652637	7.857255111	9.160544917	7.987593126	7.987593126	9.525559895	9.525559895	9.525559895	9.07123496	8.224159786	8.224159786	6.812180031	6.812180031

ID=2009	age=65

Stroke																											
10.19999695	10.2	12	12	12	12	12	12	12	12.79999542	12.79999542	13.29999542	13.29999542	13.29999542	13.29999542	13.29999542	12.3999939	12.3999939	12.3999939	12.3999939	12.3999939	13.5	13.5	13.5	13.5	11	9.099998474	9.099998474	9.099998474	10.19999695
75.1875	75.1875	81.296875	81.296875	81.29688	81.296875	81.296875	81.296875	81.296875	84.296875	84.296875	80.890625	80.890625	80.890625	80.890625	80.890625	91.5	91.5	91.5	91.5	91.5	92.59375	92.59375	92.59375	92.59375	86.296875	87.59375	87.59375	87.59375	75.1875
999.9	999.9	1000.7	1000.7	1000.7	1000.7	1000.7	1000.7	1000.7	1005.4	1005.4	1005.7	1005.7	1005.7	1005.7	1005.7	1013.4	1013.4	1013.4	1013.4	1013.4	1018.8	1018.8	1018.8	1018.8	1027.8	1032.9	1032.9	1032.9	999.9
8.2734375	8.2734	7.5546875	7.5546875	7.554688	7.5546875	7.5546875	7.5546875	7.5546875	3.610839844	3.610839844	5.971679688	5.971679688	5.971679688	5.971679688	5.971679688	5.971679688	3.083007813	3.083007813	3.083007813	3.083007813	3.083007813	3.5	3.5	3.5	3.5	4.110351563	1.222167969	1.222167969	1.222167969	8.2734375
4.199216673	4.19924	6.776310271	6.776310271	6.776303	6.776310271	6.776310271	6.776310271	6.776310271	10.77491942	10.77491942	9.416430881	9.416430881	9.416430881	9.416430881	9.416430881	10.80464181	10.80464181	10.80464181	10.80464181	10.80464181	11.599025	11.599025	11.599025	11.599025	8.459725903	9.13211308	9.13211308	9.13211308	4.199216673
Normal																													
9.099998474	9.1	9.299995422	9.299995422	9.299995	9.299995422	9.299995422	9.199996948	9.199996948	9.199996948	9.199996948	9.199996948	9.199996948	10.19999695	10.19999695	10.19999695	10.19999695	10.19999695	10.09999847	10.09999847	10.09999847	9.699996948	9.699996948	9.699996948	10.59999867	10.59999847	10.59999847	10.59999847	10.59999847	11.8999939
87.59375	87.5938	86.59375	86.59375	86.59375	86.59375	86.59375	84	84	84	84	84	84	83	83	83	83	71.390625	71.390625	71.390625	70.59375	70.59375	70.59375	66	66	66	83	83	83	77.390625
1032.9	1032.9	1032.5	1032.5	1032.5	1032.5	1032.5	1026.5	1026.5	1026.5	1026.5	1026.5	1026.5	1020	1020	1020	1020	1018.7	1018.7	1018.7	1020.3	1020.3	1020.3	1017.2	1017.2	1017.2	1013.8	1013.8	1013.8	1009.8
1.222167969	1.22217	1.138671875	1.138671875	1.138672	1.138671875	1.138671875	1.416503906	1.416503906	1.416503906	1.416503906	1.416503906	1.416503906	4	4	4	4	4	4	4	5.5546875	5.5546875	5.5546875	7.471679688	7.471679688	7.471679688	3.916503906	3.916503906	3.916503906	4.333007813
9.13211308	9.13211	9.412844708	9.412844708	9.412845	9.412844708	9.412844708	9.045988748	9.045988748	9.045988748	9.045988748	9.045988748	9.045988748	7.711196783	7.711196783	7.711196783	7.711196783	7.605798392	7.605798392	7.605798392	5.840834953	5.840834953	5.840834953	5.273145251	5.273145251	5.273145251	8.205250906	8.205250906	8.205250906	9.223759584

ID=2509	age=65

Stroke																											
15.79999542	17.6	17.59999084	17.59999084	17.59999	16.29998779	16.29998779	16.29998779	16.19999695	16.19999695	16.19999695	18.69999695	18.69999695	18.69999695	18.69999695	19.5	19.5	19.5	19.5	19.5	19.5	19.29998779	19.29998779	19.29998779	19.29998779	18.29998779	18.29998779	18.29998779	18.29998779	15.79999542

89.1875	71.7969	71.796875	71.796875	71.79688	69.390625	69.390625	69.390625	74.6875	74.6875	74.6875	91.1875	91.1875	91.1875	91.1875	86.796875	86.796875	86.796875	86.796875	86.796875	86.796875	76.6875	76.6875	76.6875	76.6875	72.796875	72.796875	72.796875	72.796875	88.1875
1013.5	1015.1	1015.1	1015.1	1015.1	1013.8	1013.8	1013.8	1006.7	1006.7	1006.7	1003.7	1003.7	1003.7	1003.7	1008.3	1008.3	1008.3	1008.3	1008.3	1008.3	1011	1011	1011	1011	1013	1013	1013	1013	1013.5
3.805175781	7.38867	7.388671875	7.388671875	7.388672	7.833007813	7.833007813	7.833007813	7.194335938	7.194335938	7.194335938	5.194335938	5.194335938	5.194335938	5.194335938	5.416015625	5.416015625	5.416015625	5.416015625	5.416015625	5.416015625	2.972167969	2.972167969	2.972167969	2.972167969	2.972167969	2.972167969	2.972167969	2.972167969	3.805175781
13.76417791	13.1551	13.15505127	13.15505127	13.15505	11.40488924	11.40488924	11.40488924	11.72412575	11.72412575	11.72412575	15.83172483	15.83172483	15.83172483	15.83172483	16.54614947	16.54614947	16.54614947	16.54614947	16.54614947	16.54614947	18.04107966	18.04107966	18.04107966	18.04107966	17.00609455	17.00609455	17.00609455	17.00609455	13.76417791
Normal																													
18.29998779	18.3	18.3999939	19.09999084	19.09999	19.09999084	19.09999084	19.09999084	19.09999084	18.29998779	18.29998779	18.29998779	18.29998779	18.29998779	18.29998779	18.29998779	18.29998779	18.29998779	16.5	16.5	16.5	16.5	16.5	16.5	16.5	15.09999847	15.09999847	15.09999847	15.09999847	16.29998779	16.29998779
72.79875	72.7969	76.796875	78.296875	78.29688	78.296875	78.296875	78.296875	78.296875	67.6875	67.6875	67.6875	67.6875	67.6875	67.6875	67.6875	67.6875	67.6875	85.09375	85.09375	85.09375	85.09375	85.09375	85.09375	85.09375	85	85	85	85	78.1875	78.1875
1013	1013	1015.2	1016.7	1016.7	1016.7	1016.7	1016.7	1016.7	1015.6	1015.6	1015.6	1015.6	1015.6	1015.6	1015.6	1015.6	1015.6	1021	1021	1021	1021	1021	1021	1021	1022.2	1022.2	1022.2	1022.2	1020.2	1020.2
2.972167969	2.97217	2.916503906	3.805175781	3.805176	3.805175781	3.805175781	3.805175781	3.805175781	4.166015625	4.166015625	4.166015625	4.166015625	4.166015625	4.166015625	4.166015625	4.166015625	4.166015625	3.194335938	3.194335938	3.194335938	3.194335938	3.194335938	3.194335938	3.194335938	2.2734375	2.2734375	2.2734375	2.2734375	2.7734375	2.7734375
17.00609455	17.0061	17.15126842	17.21047909	17.21048	17.21047909	17.21047909	17.21047909	17.21047909	16.12869204	16.12869204	16.12869204	16.12869204	16.12869204	16.12869204	16.12869204	16.12869204	16.12869204	14.96977958	14.96977958	14.96977958	14.96977958	14.96977958	14.96977958	14.96977958	14.26221103	14.26221103	14.26221103	14.26221103	15.0898711	15.0898711
ID=167	age=68																													
Stroke																													
6.199906948	6.8	6.790969237	8.899993896	9.099998	9.099998474	9.099998474	9.099998474	11.79999542	11.79999542	12.79999542	13.19999695	13.19999695	13.1999939	13.1999939	9.899993896	9.899993896	11.59999847	11.59999847	13	13	13	13.79999542	15.3999939	6.5	8.099998474	10.5	9.099998474	9.099998474	6.199906948
85.1875	85.1875	85.1875	83.09375	85.59375	85.59375	85.59375	85.59375	84.6875	84.6875	78	88.1875	88.1875	89.09375	89.09375	92	92	87.6875	87.6875	82.59375	82.59375	82.59375	91.5	94.1875	90.59375	83.09375	72.390625	76.296875	76.296875	85.1875
1017.4	1015.9	1015.9	1010.3	1002.3	1002.3	1002.3	1002.3	1000.7	1000.7	1002.2	1001.8	1001.8	1011	1011	1015.9	1015.9	1019.3	1019.3	1006.1	1006.1	1006.1	1002.9	1014.4	1031.3	1026.9	1019	1016.1	1016.1	1017.4
2.02734375	1.61108	1.611083984	4.860351563	3.555176	3.555175781	3.555175781	3.555175781	4.3046875	4.3046875	6.083007813	4.416015625	4.416015625	5.360351563	5.360351563	4	4	2.77734375	2.77734375	6.610351563	6.610351563	6.610351563	4.444335938	7.110351563	1	2.360839844	7.77734375	5.5	5.5	2.02734375
5.413725307	6.43678	6.436779541	5.577447254	6.952125	6.952124997	6.952124997	6.952124997	9.141437703	9.141437703	8.784833173	10.53479242	10.53479242	9.998389824	9.998389824	7.394993567	7.394993567	10.24193207	10.24193207	8.600400708	8.600400708	8.600400708	11.14924205	10.89241189	6.75065	7.027586593	4.925515226	5.238023349	5.238023349	5.413725307
Normal																													
9.399993896	9.3999	9.399993896	9.399993896	9.399994	12.09999847	12.09999847	11.59999847	11.59999847	12.09999847	12.09999847	13.5	13.5	13.5	12.5	10.09999847	11.59999847	11.59999847	11.59999847	11.59999847	12.09999847	12.09999847	13.59999847	15.29999542	15.2999542	15.29999542	12.29999542	14.3999939	14.3999939	16.59990084
92.59375	92.5938	92.59375	92.59375	92.59375	82.1875	82.1875	80.09375	80.09375	85.796875	85.796875	94.296875	94.296875	94.296875	96.59375	89.59375	82.890625	82.890625	82.890625	82.890625	86.59375	86.59375	88.296875	88.890625	88.890625	88.890625	95	84.6875	84.6875	84.59375
1017.3	1017.3	1017.3	1017.3	1017.3	1018.1	1018.1	1012.5	1012.5	1005.9	1005.9	1001.4	1001.4	1001.4	1001.8	1011	1007.7	1007.7	1007.7	1007.7	1003.2	1003.2	997.5	996.8	996.8	996.8	1006	1004.6	1004.6	1003
2.02734375	2.02734	2.02734375	2.02734375	2.027344	5	5	4.416015625	4.416015625	3.972167969	3.972167969	2.944335938	2.944335938	2.944335938	1	3.610839844	3.833007813	3.833007813	3.833007813	3.833007813	2.583007813	2.583007813	2.388671875	4.666015625	4.666015625	4.666015625	3.944335938	4.138671875	4.138671875	6.166015625
8.669740952	8.66974	8.669740952	8.669740952	8.669741	8.887248363	8.887248363	8.836079982	8.836079982	9.73715768	9.73715768	12.06152656	12.06152656	12.06152656	12.74165	7.948561167	9.328786961	9.328786961	9.328786961	9.328786961	10.92748158	10.92748158	12.6341029	12.57276979	12.57276979	12.57276979	9.975127009	12.02779372	12.02779372	12.87810831
ID=331	age=68																													
Stroke																													
15.3999939	15.4	15.29999542	17.3999939	17.39999	16.59999084	16.59999084	14.09999847	14.09999847	14.09999847	14.09999847	14.09999847	14.09999847	14.09999847	13.59999847	13.09999847	13.09999847	13.09999847	14.29999542	14.29999542	15	15	15	15	15	10.8999939	10.8999939	10.8999939	11.59999847	15.3999939
85.890625	85.8906	77.59375	87.890625	87.89063	89.59375	89.59375	79.296875	79.296875	79.296875	81.296875	81.296875	81.296875	81.296875	82.390625	72.890625	72.890625	72.890625	72.890625	79.890625	79.890625	93.5	93.5	93.5	93.5	93.5	63.3984375	63.3984375	63.3984375	65.296875	85.890625

1008.4	1008.4	1007.2	1006.1	1006.1	1013.8	1013.8	1021.1	1021.1	1021.1	1019.4	1019.4	1019.4	1019.4	1014.8	1017.7	1017.7	1017.7	1015.9	1015.9	1017.9	1017.9	1017.9	1017.9	1017.9	1021.9	1021.9	1021.9	1025.3	1028.4
4.833007813	4.83301	5.221679688	9.970703125	9.970703	8.8046875	8.8046875	2.02734375	2.02734375	2.02734375	4.5	4.5	4.5	4.5	10.02734375	9.693359375	9.693359375	9.693359375	9.638671875	9.638671875	7.333007813	7.333007813	7.333007813	7.333007813	7.333007813	5.5	5.5	5.5	7.75	4.833007813
12.55260108	12.5526	12.15249735	11.31835022	11.31835	11.11214147	11.11214147	13.45202315	13.45202315	13.45202315	11.42392338	11.42392338	11.42392338	11.42392338	6.855894993	6.49861936	6.49861936	6.49861936	7.925778168	7.925778168	10.29073983	10.29073983	10.29073983	10.29073983	10.29073983	7.185168398	7.185168398	7.185168398	6.180423286	12.55260108
Normal																													
11.59999847	13.4	13.3999939	14.59999847	16.7	15.09999847	16	14	13.8999939	12.79999542	12.79999542	11.69999695	11.69999695	11.69999695	11.69999695	14.59999847	13	13	13.29999542	12.69999695	12.69999695	12.69999695	12.69999695	13.5	13.5	13.5	13.5	13.09999847	13.09999847	12.59999847	13.59999847
65.296875	69.6875	69.6875	89.59375	87.09375	86.1875	80.5	80.5	80.890625	71.09375	71.09375	79.59375	79.59375	79.59375	79.59375	70	73.390625	82.1875	70.796875	84.5	84.5	84.5	79.5	79.5	79.5	79.5	74.5	74.5	82.890625	82.890625
1025.3	1015.8	1015.8	995.7	1015.6	1023.4	1026	1026	1023.9	1018.3	1018.3	1006.8	1006.8	1006.8	1006.8	1015.6	1023.5	1025.7	1022.3	1024.4	1024.4	1024.4	1022	1022	1022	1022	1024.4	1024.4	1012.5	1012.5
7.75	12.25	12.25	7.160215625	6.916016	5.944335938	2.833007813	2.833007813	1.555419922	3.583007813	3.583007813	7.360351563	7.360351563	7.360351563	7.360351563	4.8046875	4.221679688	2.583007813	4.971679688	3.444335938	3.444335938	3.444335938	7.833007813	7.833007813	7.833007813	7.833007813	6.916015625	6.916015625	11.77734375	11.77734375
6.180423286	5.20504	5.205042635	9.963391038	12.4694	11.39938583	12.67131552	12.67131552	13.65457183	10.7981869	10.7981869	6.588183385	6.588183385	6.588183385	6.588183385	11.71895186	10.48072215	11.85249322	10.19668331	10.81003196	10.81003196	10.81003196	8.255153164	8.255153164	8.255153164	8.255153164	8.480792314	8.480792314	5.735856662	5.735856662

ID<1600	age<68
Stroke	
18.8999939	18.5	18.5	16.79998779	16.79999	16.79998779	15.8999939	15.8999939	14.5	14.5	14.5	14.5	14.5	14.5	14.5	14.3999939	14.3999939	14.3999939	14.59999847	14.59999847	14.59999847	14.59999847	14.59999847	14.59999847	14.59999847	14.59999847	14.59999847	14.5	14.5	18.8999939
81.890625	76.6875	76.6875	81.6875	81.6875	81.6875	77	77	78.890625	78.890625	78.890625	78.890625	78.890625	78.890625	78.890625	79.5	79.5	79.5	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	84.296875	84.296875	81.890625
1017.1	1023.3	1023.3	1026	1026	1026	1024.2	1024.2	1025.4	1025.4	1025.4	1025.4	1025.4	1025.4	1025.4	1027.2	1027.2	1027.2	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1017.1
5.888671875	7.22168	7.221679688	5.583007813	5.58308	5.583007813	2.02734375	2.02734375	1.944335938	1.944335938	1.944335938	1.944335938	1.944335938	1.944335938	1.944335938	1.638671875	1.638671875	1.638671875	2.194335938	2.194335938	2.194335938	2.194335938	2.194335938	2.194335938	2.194335938	2.194335938	2.194335938	2.194335938	1.833251953	1.833251953	5.888671875
15.57831882	14.2667	14.26672103	13.50823823	13.50824	13.50823823	15.28352904	15.28352904	13.92912724	13.92912724	13.92912724	13.92912724	13.92912724	13.92912724	13.92912724	14.08769635	14.08769635	14.08769635	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	14.02320514	14.02320514	15.57831882
Normal																													
14.5	14.5	14.5	15.09999847	15.1	15.09999847	15.09999847	15.09999847	15.09999847	15.59999847	15.59999847	15.59999847	15.59999847	15.59999847	15.5	15.8999939	15.09999847	15.09999847	15.09999847	15.09999847	15.09999847	15.09999847	15.09999847	15.09999847	14.19999695	14.19999695	14.19999695	14.19999695	14.19999695	14.19999695	13.8999939
84.296875	84.2969	84.296875	82.5	82.5	82.5	79.1875	79.1875	79.1875	81.5	81.5	81.5	81.5	81.5	81.5	79	74.5	79.6875	79.6875	79.6875	79.6875	79.6875	79.6875	79.6875	80.5	80.5	80.5	80.5	80.5	80.5	73.59375
1027.7	1027.7	1027.7	1026	1026	1026	1025.6	1025.6	1025.6	1026.3	1026.3	1026.3	1026.3	1026.3	1026.3	1027.4	1029.9	1031.3	1031.3	1031.3	1031.3	1031.3	1031.3	1031.3	1029.9	1029.9	1029.9	1029.9	1029.9	1029.9	1026.9
1.833251953	1.83325	1.833251953	1.916503906	1.916504	1.916503906	1.638671875	1.638671875	1.638671875	2.166503906	2.166503906	2.166503906	2.166503906	2.166503906	2.166503906	1.638671875	1.363283984	1.555419922	1.555419922	1.555419922	1.555419922	1.555419922	1.555419922	1.555419922	1.805419922	1.805419922	1.805419922	1.805419922	1.805419922	1.805419922	1.944335938
14.02320514	14.0232	14.02320514	14.56194132	14.56194	14.56194132	14.79492177	14.79492177	14.79492177	14.8639763	14.8639763	14.8639763	14.8639763	14.8639763	14.8639763	15.19904949	15.83374874	14.86510677	14.86510677	14.86510677	14.86510677	14.86510677	14.86510677	14.86510677	13.74280108	13.74280108	13.74280108	13.74280108	13.74280108	13.74280108	13.31953891

ID<2258	age<60	temp	humid	atm	wsp	wc
Stroke						
10.29999542	10.3	12.5	12.5	12.5	12.5	10.8999939	10.8999939	12.3999939	14	13.5	13.5	13.5	13	13	13	13	14.19999695	14.19999695	14.19999695	14.19999695	16.19999695	16.19999695	16.19999695	16.19999695	16.19999695	16.19999695	14.59999847	14.59999847
62.59375	62.5938	68.390625	68.390625	68.39063	68.390625	71.59375	71.59375	64.09375	78.09375	74.6875	74.6875	74.6875	73.6875	73.6875	73.6875	73.6875	75	75	75	75	84	84	84	84	84	84	84	73.6875	73.6875
1024.9	1024.9	1021.5	1021.5	1021.5	1021.5	1019.5	1019.5	1015.6	1012.9	1015.2	1015.2	1015.2	1014.9	1014.9	1014.9	1014.9	1008.5	1008.5	1008.5	1008.5	1007.8	1007.8	1007.8	1007.8	1007.8	1007.8	1007.8	1016.4	1016.4

4.02734375	4.02734	4.388671875	4.388671875	4.388672	4.388671875	5.444335938	5.444335938	7.944335938	10.38867188	9.33203125	9.33203125	9.33203125	5.888671875	5.888671875	5.888671875	5.888671875	9.25	9.25	9.25	9.25	8.02734375	8.02734375	8.02734375	8.02734375	8.02734375	8.02734375	8.02734375	5.25	5.25
7.792753858	7.79275	9.814073119	9.814073119	9.814073	9.814073119	7.231071229	7.231071229	6.935700097	7.087293138	7.206730902	7.206730902	7.206730902	9.153576927	9.153576927	9.153576927	9.153576927	8.068446487	8.068446487	8.068446487	8.068446487	11.16305231	11.16305231	11.16305231	11.16305231	11.16305231	11.16305231	11.16305231	11.37729836	11.37729836
Normal																													
14.3999939	14.4	14.3999939	14.3999939	14.39999	14.3999939	14.3999939	14.3999939	10.69999695	12	12	13.79999542	13.79999542	13.79999542	13.79999542	13.79999542	13.79999542	14.3999939	14.3999939	14.3999939	14.3999939	14.3999939	11.8999939	11.8999939	11.8999939	11.59999847	11.59999847	11.59999847	11.69999695	11.69999695
81.09375	81.0938	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	69.296875	70.296875	70.296875	84.5	84.5	84.5	84.5	84.5	84.5	79.296875	79.296875	79.296875	79.296875	79.296875	79.796875	79.796875	79.796875	68.59375	68.59375	68.59375	75.796875	75.796875
1015.2	1015.2	1015.2	1015.2	1015.2	1015.2	1015.2	1015.2	1021.1	1017.7	1017.7	1017.8	1017.8	1017.8	1017.8	1017.8	1017.8	1020.2	1020.2	1020.2	1020.2	1020.2	1020.1	1020.1	1020.1	1013.3	1013.3	1013.3	995.6	995.6
7.583007813	7.58301	7.583007813	7.583007813	7.583008	7.583007813	7.583007813	7.583007813	3.610839844	8.25	8.25	4.27734375	4.27734375	4.27734375	4.27734375	4.27734375	4.27734375	3.610839844	3.610839844	3.610839844	3.610839844	3.610839844	4.388671875	4.388671875	4.388671875	9.220703125	9.220703125	9.220703125	8.693359375	8.693359375
9.443676258	9.44368	9.443676258	9.443676258	9.443676	9.443676258	9.443676258	9.443676258	8.576639892	6.258975	6.258975	11.28259388	11.28259388	11.28259388	11.28259388	11.28259388	11.28259388	12.44979869	12.44979869	12.44979869	12.44979869	12.44979869	9.177352384	9.177352384	9.177352384	5.096569044	5.096569044	5.096569044	5.593114327	5.593114327
2306	60																												
Stroke																													
14.69999695	14.7	14.5	14	14	14	14	15.5	15.5	15.5	15.5	15.5	15.09999847	15.09999847	15.09999847	13.8999939	13.8999939	13.8999939	13.8999939	10.79999542	10.79999542	10.79999542	10.79999542	10.79999542	10.79999542	10.79999542	10.79999542	11.79999542	12.29999542	13.69999695
76.6875	76.6875	77.09375	71.296875	71.29688	71.296875	71.296875	82.5	82.5	82.5	82.5	82.5	78.296875	78.296875	78.296875	88.09375	88.09375	88.09375	88.09375	64.09375	64.09375	64.09375	64.09375	64.09375	64.09375	64.09375	64.09375	64.09375	69.09375	65.6875	79.6875
1006.6	1006.6	1010.5	1011.8	1011.8	1011.8	1011.8	1010.7	1010.7	1010.7	1010.7	1010.7	1012	1012	1012	1013.5	1013.5	1013.5	1013.5	1020.6	1020.6	1020.6	1020.6	1020.6	1020.6	1020.6	1020.6	1020.6	1013.8	1010.3	1013.4
5.8046875	5.80469	6.166015625	8.27734375	8.277344	8.27734375	8.27734375	5.416015625	5.416015625	5.416015625	5.416015625	5.416015625	7.25	7.25	7.25	4.860351563	4.860351563	4.860351563	4.860351563	2.555175781	2.555175781	2.555175781	2.555175781	2.555175781	2.555175781	2.555175781	2.555175781	2.555175781	3.3046875	9	5.5
11.06734509	11.0673	10.58056842	8.505137096	8.505137	8.505137396	8.505137396	12.22336432	12.22336432	12.22336432	12.22336432	12.22336432	10.4605608	10.4605608	10.4605608	10.92702977	10.92702977	10.92702977	10.92702977	9.616363809	9.616363809	9.616363809	9.616363809	9.616363809	9.616363809	9.616363809	9.616363809	9.616363809	7.517150034	6.058344752	10.2140717
Normal																													
13.5	13.5	13.5	13.5	12.2	12.19999695	12.19999695	12.19999695	11.59999847	11.19999695	11.19999695	11.19999695	11.19999695	11.19999695	11.19999695	11.19999695	10.29999542	10.29999542	10.29999542	12.5	12.5	12.5	12.5	10.8999939	10.8999939	12.3999939	14	13.5	13.5	13.5
69.5	69.5	69.5	69.5	75.6875	75.6875	75.6875	75.6875	75.796875	66.890625	66.890625	66.890625	66.890625	66.890625	66.890625	66.890625	62.59375	62.59375	62.59375	68.390625	68.390625	68.390625	68.390625	71.59375	71.59375	64.09375	78.09375	74.6875	74.6875	74.6875
1024.8	1024.8	1024.8	1024.8	1030.8	1030.8	1030.8	1030.8	1030.5	1027	1027	1027	1027	1027	1027	1027	1027	1024.9	1024.9	1024.9	1021.5	1021.5	1021.5	1021.5	1019.5	1019.5	1015.6	1012.9	1015.2	1015.2	1015.2
4.5546875	4.55469	4.5546875	4.5546875	3.583008	3.583007813	3.583007813	3.583007813	2.722167969	3.138671875	3.138671875	3.138671875	3.138671875	3.138671875	3.138671875	3.138671875	3.138671875	4.02734375	4.02734375	4.02734375	4.388671875	4.388671875	4.388671875	4.388671875	5.444335938	5.444335938	7.944335938	10.38867188	9.33203125	9.33203125	9.33203125
10.74222654	10.7422	10.74222654	10.74222654	10.17042	10.17041711	10.17041711	10.17041711	10.29090295	9.511348564	9.511348564	9.511348564	9.511348564	9.511348564	9.511348564	9.511348564	9.511348564	7.792753858	7.792753858	7.792753858	9.814073119	9.814073119	9.814073119	9.814073119	7.231071229	7.231071229	6.935700097	7.087293138	7.206730902	7.206730902	7.206730902
1616	61																												
Stroke																													
15.79999542	15.8	17.79998779	17.79998779	17.79999	19.29998779	19.29998779	19.29998779	19.29998779	19.29998779	19.3999939	19.3999939	19.3999939	19.3999939	18.8999939	18.8999939	18.8999939	18.8999939	18.8999939	18.5	18.5	16.79998779	16.79998779	16.79998779	15.8999939	15.8999939	14.5	14.5	14.5	14.5	14.5
76.09375	76.0938	90	90	90	89.5	89.5	89.5	89.5	89.5	89.390625	89.390625	89.390625	89.390625	83.890625	83.890625	83.890625	83.890625	83.890625	76.6875	76.6875	81.6875	81.6875	81.6875	77	77	78.890625	78.890625	78.890625	78.890625	78.890625
1009.7	1009.7	1012.6	1012.6	1012.6	1012.8	1012.8	1012.8	1012.8	1012.8	1014	1014	1014	1014	1017.1	1017.1	1017.1	1017.1	1017.1	1023.3	1023.3	1026	1026	1026	1024.2	1024.2	1025.4	1025.4	1025.4	1025.4	1025.4
7.666015625	7.66602	2.5	2.5	2.5	2.722167969	2.722167969	2.722167969	2.722167969	2.722167969	4.388671875	4.388671875	4.388671875	4.388671875	5.888671875	5.888671875	5.888671875	5.888671875	5.888671875	7.221679688	7.221679688	5.583007813	5.583007813	5.583007813	2.02734375	2.02734375	1.944335938	1.944335938	1.944335938	1.944335938	1.944335938

10.95518174	10.9556	16.84973747	16.84973747	16.84974	18.22466478	18.22466478	18.22466478	18.22466478	18.22466478	17.13628061	17.13628061	17.13628061	17.13628061	15.57831882	15.57831882	15.57831882	15.57831882	14.26672103	14.26672103	13.50823823	13.50823823	13.50823823	15.28352904	15.28352904	13.92912724	13.92912724	13.92912724	13.92912724	13.92912724
Normal																														
14.5	14.4	14.3999939	14.3999939	14.6	14.59999847	14.59999847	14.59999847	14.59999847	14.59999847	14.59999847	14.59999847	14.59999847	14.5	14.5	14.5	14.5	14.5	14.5	15.09999847	15.09999847	15.09999847	15.09999847	15.09999847	15.09999847	15.59999847	15.59999847	15.59999847	15.59999847	15.59999847
78.890625	79.5	79.5	79.5	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	81.09375	84.296875	84.296875	84.296875	84.296875	84.296875	84.296875	82.5	82.5	82.5	79.1875	79.1875	79.1875	81.5	81.5	81.5	81.5	81.5
1025.4	1027.2	1027.2	1027.2	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1026	1026	1026	1025.6	1025.6	1025.6	1026.3	1026.3	1026.3	1026.3	1026.3
1.944319308	1.63867	1.638671875	1.638671875	2.194336	2.194319308	2.194319308	2.194319308	2.194319308	2.194319308	2.194319308	2.194319308	2.194319308	1.833251953	1.833251953	1.833251953	1.833251953	1.833251953	1.833251953	1.916503906	1.916503906	1.916503906	1.638671875	1.638671875	1.638671875	2.166503906	2.166503906	2.166503906	2.166503906	2.166503906
13.92912724	14.0877	14.08769635	14.08769635	13.82058	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	13.82057759	14.02320514	14.02320514	14.02320514	14.02320514	14.02320514	14.02320514	14.56194132	14.56194132	14.56194132	14.79492177	14.79492177	14.79492177	14.8639763	14.8639763	14.8639763	14.8639763	14.8639763

236	62

Stroke

11.3999939	12.6	12.59999847	12.59999847	11.6	11.59999847	9.099998474	9.099998474	8.199996948	8.199996948	9.099998474	9.699996948	11.3999939	12.8999939	12.59999847	10.19999695	10.19999695	11.5	12.09999847	11	11	11.3999939	11.3999939	11.3999939	11.79999542	10.79999542	10.79999542	8.599998474	8.599998474	8.599998474	8.599998474
78.5	77.7968	77.796875	77.796875	85	85	79.6875	79.6875	75.09375	75.09375	69.6875	79.1875	76.390625	87.09375	86.890625	92.09375	92.09375	72.1875	87.296875	86.6875	86.6875	84	84	84	87.890625	80.296875	80.296875	81.1875	81.1875	81.1875	81.1875
1009.4	1011.5	1011.5	1011.5	1009.1	1009.1	1018.8	1018.8	1019.3	1019.3	1018.7	1018.3	1014	999.6	1000.9	1006.5	1006.5	1012.1	996.6	1006.4	1006.4	1010.7	1010.7	1010.7	998.4	1010.3	1010.3	1006.7	1006.7	1006.7	1006.7
6.971679688	8.19336	8.191359375	8.191359375	5.97168	5.971679688	3.5	3.5	2.27734375	2.27734375	4.110351563	3.166503906	5.27734375	6.721679688	5	3.638671875	3.638671875	5.471679688	4.666015625	4.666015625	4.666015625	4.666015625	3.360839844	3.360839844	3.360839844	4.388671875	4.138671875	4.138671875	3.138671875	3.138671875	3.138671875
6.55392222	6.97961	6.979612859	6.979612859	7.562625	7.562624883	7.002123406	7.002123406	7.209506543	7.209506543	6.453245434	7.92908008	7.90805621	8.405774645	9.423498363	8.028652637	8.028652637	7.857255111	9.160544917	7.987593126	7.987593126	9.525559895	9.525559895	9.525559895	9.07123496	8.224159786	8.224159786	6.812380031	6.812380031	6.812380031	6.812380031
Normal																														
10.29999542	11.7	11.69999695	11.69999695	11.7	11.69999695	11.69999695	12	12	11.3999939	11.3999939	12.29999542	9.899993896	8.399993896	8.399993896	7.699999848	7.699999848	8.5	10	11.69999695	13.09999847	12.5	12.5	11	11	11	13.19999695	12.29999542	10.79999542	7.799999237
80.5	90.2969	90.296875	90.296875	90.29688	90.296875	90.296875	75.6875	75.6875	71.6875	71.6875	82.6875	77.09375	75.890625	75.890625	86.890625	86.890625	90.390625	87.59375	82.09375	88.890625	82.09375	82.09375	82.890625	82.890625	82.890625	85.390625	86.390625	90.59375	91.6875
995	995.2	995.2	995.2	995.2	995.2	995.2	1005.9	1005.9	999.1	998.1	995	1008.7	1011.4	1011.4	1006.8	1006.8	1008.5	1004.3	1001.5	1001.1	1004.2	1004.2	1012.9	1012.9	1012.9	1015.6	1023.4	1027.5	1029.1
4.5548875	3.52734	3.52734375	3.52734375	3.527344	3.52734375	3.52734375	5.471679688	5.471679688	9.916015625	9.916015625	6.083007813	6.52734375	2.916503906	2.916503906	3.972167969	3.972167969	2.333007813	3.694335938	5.666015625	5.833007813	8.638671875	8.638671875	3.944335938	3.944335938	3.944335938	4.833007813	3.888671875	0.638793945	0.888793945
7.336584166	9.695	9.694999636	9.694999636	9.695	9.694999636	9.694999636	8.397868148	8.397868148	4.370129775	4.370129775	8.238563351	5.251019936	6.801715196	6.811715196	5.1018216	5.1018216	7.463402676	7.769870452	7.916581905	9.305570059	6.544926634	6.544926634	8.60227055	8.60227055	8.60227055	10.19990093	10.01773569	11.38440784	8.19326908

729	62

Stroke

10.59999847	10.6	12.8999939	12.8999939	12.89999	12.69999695	12.69999695	11.79999542	11.79999542	12.69999695	14.29999542	14.29999542	14.29999542	14.29999542	14.29999542	12	12	12	12	13.59999847	13.59999847	13.29999542	13.29999542	13.29999542	13.29999542	13.29999542	8.899993896	8.899993896	8.899993896	8.899993896
76.5	76.5	77.890625	77.890625	77.89063	91.296875	91.296875	87	87	73	91.1875	91.1875	91.1875	91.1875	91.1875	77.1875	77.1875	77.1875	77.1875	70.5	70.5	77	77	77	77	77	80.6875	80.6875	80.6875	80.6875	80.6875
1006.3	1006.3	999.4	999.4	999.4	992.3	992.3	1004.4	1004.4	1001.2	999.7	999.7	999.7	999.7	999.7	1016.9	1016.9	1016.9	1016.9	1018.6	1018.6	1019.3	1019.3	1019.3	1019.3	1019.3	1020.5	1020.5	1020.5	1020.5	1020.5
9.359375	9.35938	9.416015625	9.416015625	9.416016	2.638671875	2.638671875	3.52734375	3.52734375	9.443359375	6.583007813	6.583007813	6.583007813	6.583007813	6.583007813	3.027343375	3.027343375	3.027343375	3.02734375	6.110351563	6.110351563	5.77734375	5.77734375	5.77734375	5.77734375	5.77734375	4.27734375	4.27734375	4.27734375	4.27734375	4.27734375
3.843991893	3.84399	6.457115671	6.457115671	6.457116	11.49615221	11.49615221	9.799523627	9.799523627	6.207325417	10.05344089	10.05344089	10.05344089	10.05344089	10.05344089	10.43794052	10.43794052	10.43794052	10.43794052	9.638105583	9.638105583	9.566120507	9.566120507	9.566120507	9.566120507	9.566120507	6.092851049	6.092851049	6.092851049	6.092851049	6.092851049

Normal

9.399993896	9.39999	9.399993896	10.09999847	10.7	10.69999695	14	14	14	14	14	14	15.59999847	15.59999847	13	13	13	13.09999847	13.09999847	13.09999847	13.09999847	13.09999847	13.09999847	14.5	14.5	14.5	13.59999847	13.59999847	13.59999847	13.3999939
74	74	74	71.6875	75.29688	75.296875	89.1875	89.1875	89.1875	89.1875	89.1875	89.1875	88.390625	88.390625	80.890625	80.890625	80.890625	72.390625	72.390625	72.390625	72.390625	72.390625	72.390625	69.390625	69.390625	83	72	72	72	87.59375
1017.4	1017.4	1017.4	1018.5	1019.4	1019.4	1021.3	1021.3	1021.3	1021.3	1021.3	1021.3	1025.6	1025.6	1025.3	1025.3	1025.3	1018.6	1018.6	1018.6	1018.6	1018.6	1018.6	1011.8	1011.8	1014.3	1018.3	1018.3	1018.3	1014.7
5.583007813	5.58301	5.583007813	5.416015625	4.554688	4.5546875	4.388671875	4.388671875	4.388671875	4.388671875	4.388671875	4.388671875	3.083007813	3.083007813	2.222167969	2.222167969	2.222167969	4.721679688	4.721679688	4.721679688	4.721679688	4.721679688	4.721679688	9.888671875	9.888671875	9.859375	7.583007813	7.583007813	7.583007813	3.75
5.491931074	5.49193	5.491931074	6.390302708	7.76229	7.762290478	11.40585876	11.40585876	11.40585876	11.40585876	11.40585876	11.40585876	14.12316062	14.12316062	12.16350379	12.16350379	12.16350379	10.18215446	10.18215446	10.18215446	10.18215446	10.18215446	10.18215446	7.994543822	7.994543822	8.013553809	8.547452871	8.547452871	8.547452871	11.2884936

71	64

Stroke

14.5	13.8	13.69999695	13.69999695	13.7	13.19999695	14.5	16.8999939	16.19999695	16.19999695	15.3999939	15.3999939	15.3999939	15.8999939	15.8999939	15.8999939	16.59999084	16.59999084	16.59999084	14.69999695	15.8999939	15.8999939	16.29998779	16.29998779	16.59999084	16.59999084	17.5	17.59999084	17.59999084	17.59999084
84.6875	82.2968	85.296875	85.296875	85.29688	85.890625	82.890625	82.796875	76.5	76.5	79.59375	79.59375	79.59375	72.5	72.5	72.5	71.390625	71.390625	71.390625	80.390625	85.59375	85.59375	84.390625	84.390625	68.890625	68.890625	73.59375	84.1875	84.1875	83
1022.5	1024	1025.4	1025.4	1025.4	1026	1023.5	1017.6	1016.5	1016.5	1015.1	1015.1	1015.1	1013.2	1013.2	1013.2	1014.5	1014.5	1014.5	1017.4	1017	1017	1017.8	1017.8	1023	1023	1015.6	1010.5	1010.5	1017.1
3.305175781	0.41663	0.555541992	0.555541992	0.555542	0.333312988	2.52734375	4.5546875	3.972167969	3.972167969	3.52734375	3.52734375	3.52734375	5.52734375	5.52734375	5.52734375	2.388671875	2.388671875	2.388671875	0.666625977	3.02734375	3.02734375	1.194335938	1.194335938	6.694335938	6.694335938	7.194335938	3.166503906	3.166503906	2.222167969
12.80145906	14.5584	14.10508627	14.33508627	14.33509	14.04145994	13.44039111	14.36070989	14.05644501	14.05644501	13.56244313	13.56244313	13.56244313	12.57392708	12.57392708	12.57392708	15.70666638	15.70666638	15.70666638	15.22817696	14.47835705	14.47835705	16.37400925	16.37400925	12.51069736	12.51069736	13.17115292	16.13385829	16.13385829	16.86060094

Normal

18.5	17.3	17.09999084	17.09999084	17	17	16.69999695	15.29999542	16.5	21.19999695	21.19999695	21.59999084	21.59999084	21.69999695	20.79998779	20.79998779	18.69999695	18.69999695	19	19	19	19.69999695	20.09999084	20.09999084	18.8999939	19.29998779	19.29998779	21.69999695	21.69999695	23.19999695
87.59375	79.3906	81.59375	81.59375	81.6875	81.6875	80.390625	70.390625	62.6953125	83.890625	83.890625	87.296875	87.296875	77.59375	79.5	79.5	72.296875	72.296875	74.390625	74.390625	74.390625	83.59375	70.5	70.5	71.1875	80.59375	80.59375	86.09375	86.09375	88.09375
1022.2	1021.4	1020.1	1020.1	1022.2	1022.2	1026.9	1025.5	1022.2	1017.9	1017.9	1023.4	1023.4	1029.6	1024	1024	1024.3	1024.3	1023.4	1023.4	1023.4	1020	1024.1	1024.1	1024.7	1012.4	1012.4	1012	1012	1008.7
10.94335938	1.83325	1.916503906	1.916503906	2.333008	2.333007813	1.944335938	1.833251953	2.77734375	3.916503906	3.916503906	6.52734375	6.52734375	6.360351563	4.666015625	4.666015625	3.694335938	3.694335938	3.77734375	3.77734375	3.77734375	2.166503906	2.388671875	2.388671875	3.305175781	2.02734375	2.02734375	4.888671875	4.888671875	12.27734375
12.05416937	16.8622	16.59284422	16.59284422	16.16027	16.16026855	16.16425861	14.83433243	14.90532392	19.36127572	19.36127572	18.12986013	18.12986013	18.34026176	18.43752874	18.43752874	16.89047062	16.89047062	17.14539013	17.14539013	17.14539013	19.04630406	19.29133288	19.29133288	17.38249402	18.74304275	18.74304275	19.25426634	19.25426634	17.01825012

286	64

Stroke

11.69999695	14.6	13	13	13.3	12.69999695	12.69999695	12.69999695	13.5	13.5	13.5	13.5	13.09999847	13.09999847	13.59999847	13.59999847	15.69999695	14.69999695	15.8999939	15.8999939	12.79999542	12.79999542	12.79999542	13.29999542	13	13.5	12.3999939	13.5	13.5	14.3999939
79.59375	70	73.390625	82.1875	73.79688	84.5	84.5	84.5	79.5	79.5	79.5	79.5	74.5	74.5	82.890625	82.890625	88.296875	82	78.890625	78.890625	75.390625	75.390625	75.390625	74.6875	74	87.890625	75.5	78.09375	78.09375	87.09375
1006.8	1015.6	1023.5	1025.7	1022.3	1024.4	1024.4	1024.4	1022	1022	1022	1022	1024.4	1024.4	1012.5	1012.5	1019.7	1025.8	1026.1	1026.1	1021.7	1021.7	1021.7	1014.7	999.4	995.1	1005.7	1007.2	1007.2	1015.6
7.360351563	4.80469	4.221679688	2.583007813	4.97168	3.444335938	3.444335938	3.444335938	7.833007813	7.833007813	7.833007813	7.833007813	6.916015625	6.916015625	11.77734375	11.77734375	7.360351563	7.083007813	5.0546875	5.0546875	4.944335938	4.944335938	4.944335938	8.8046875	12.52734375	6.083007813	7.944335938	6.75	6.75	5.583007813
6.588183385	11.719	10.48072215	11.85249022	10.19668	10.81003196	10.81003196	10.81003196	8.255153164	8.255153164	8.255153164	8.255153164	8.480792314	8.480792314	5.738546662	5.738546662	11.0528494	13.18982695	12.92210124	12.92210124	9.682631563	9.682631563	9.682631563	7.340620525	4.512038568	9.549613125	6.935700397	9.0472875	9.0472875	10.9083593

Normal

14.29999542	14.3	11.19999695	12.5	14	14	10.29999542	11.5	11.5	11.79999542	11.79999542	12	12.79999542	12.79999542	12.69999695	12	12	11.3999939	11.3999939	11.3999939	12.59999847	12.59999847	12.59999847	11.59999847	11.59999847	9.099998474	9.099998474	8.199996948	8.199996948	9.099998474
86.5	86.5	80.796875	72.1875	86.09375	86.09375	83.5	76.6875	76.6875	84.09375	84.09375	78.796875	80.390625	80.390625	83.890625	82.890625	82.890625	78.5	78.5	78.5	77.796875	77.796875	77.796875	85	85	79.6875	79.6875	75.09375	75.09375	69.6875
1009.3	1009.3	1012.5	1013.1	1016.9	1016.9	1025.3	1020.9	1020.9	1015.9	1015.9	1015.9	1016.9	1016.9	1019.7	1023.5	1023.5	1009.4	1009.4	1009.4	1011.5	1011.5	1011.5	1009.1	1009.1	1018.8	1018.8	1019.3	1019.3	1018.7
6.52734375	6.52734	5.944335938	6.75	6.916016	6.916015625	2.118671875	2.444335938	2.444335938	7.8046875	7.8046875	7.610351563	6.221679688	6.221679688	7.5	6.188671875	6.188671875	6.971679688	6.971679688	6.971679688	8.193359375	8.193359375	8.193359375	5.971679688	5.971679688	3.5	3.5	2.27734375	2.27734375	4.110351563
10.0943474	10.0943	7.148500229	7.9424125	9.477946	9.477945665	9.483099115	10.43380136	10.43380136	6.364034282	6.364034282	6.734445922	8.677692237	8.677692237	7.600621586	7.670756029	7.670756029	6.55392222	6.55392222	6.55392222	6.979612859	6.979612859	6.979612859	7.562624883	7.562624883	7.002123406	7.002123406	7.209506543	7.209506543	6.453245434

1579	67

Stroke

14.59999847	14.6	14.59999847	14.59999847	14.6	14.59999847	14.59999847	14.5	14.5	14.5	14.5	14.5	14.5	15.09999847	15.09999847	15.09999847	15.09999847	15.09999847	15.09999847	15.59999847	15.59999847	15.59999847	15.59999847	15.59999847	15.5	15.8999939	15.09999847	15.09999847	15.09999847	15.09999847
81.09375	81.0938	81.09375	81.09375	81.09375	81.09375	81.09375	84.296875	84.296875	84.296875	84.296875	84.296875	84.296875	82.5	82.5	82.5	79.1875	79.1875	79.1875	81.5	81.5	81.5	81.5	81.5	79	74.5	79.6875	79.6875	79.6875	79.6875
1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1027.7	1026	1026	1026	1025.6	1025.6	1025.6	1026.3	1026.3	1026.3	1026.3	1026.3	1027.4	1029.9	1031.3	1031.3	1031.3	1031.3
2.194335938	2.1943	2.194335938	2.194335938	2.194336	2.194335938	2.194335938	1.833251953	1.833251953	1.833251953	1.833251953	1.833251953	1.833251953	1.916503906	1.916503906	1.916503906	1.638671875	1.638671875	1.638671875	2.166503906	2.166503906	2.166503906	2.166503906	2.166503906	1.638671875	1.361083984	1.555419922	1.555419922	1.555419922	1.555419922
13.82057759	13.8206	13.82057759	13.82057759	13.82058	13.82057759	13.82057759	14.02320514	14.02320514	14.02320514	14.02320514	14.02320514	14.02320514	14.56194132	14.56194132	14.56194132	14.79492177	14.79492177	14.79492177	14.8639763	14.8639763	14.8639763	14.8639763	14.8639763	15.19904949	15.83374874	14.86510677	14.86510677	14.86510677	14.86510677

Normal

15.09999847	15.1	14.19999695	14.19999695	14.2	14.19999695	14.19999695	14.19999695	13.8999939	13.8999939	13.8999939	13.8999939	13.8999939	13.8999939	13.8999939	13.8999939	13.8999939	14.09999847	14.09999847	14.09999847	14.3999939	14.3999939	14.3999939	14.3999939	16.59999084	16.59999084	16.59999084	16.59999084	16.59999084	16.59999084	16
79.6875	79.6875	80.5	80.5	80.5	80.5	80.5	80.5	73.59375	73.59375	73.59375	73.59375	73.59375	73.59375	73.59375	73.59375	73.59375	76	76	76	74.296875	74.296875	74.296875	74.296875	74.5	74.5	74.5	74.5	74.5	74.5	81.890625
1031.3	1031.3	1029.9	1029.9	1029.9	1029.9	1029.9	1029.9	1026.9	1026.9	1026.9	1026.9	1026.9	1026.9	1026.9	1026.9	1026.9	1020	1020	1020	1010.8	1010.8	1010.8	1010.8	1006	1006	1006	1006	1006	1006	1002.1
1.555419922	1.5542	1.805419922	1.805419922	1.80542	1.805419922	1.805419922	1.805419922	1.944335938	1.944335938	1.944335938	1.944335938	1.944335938	1.944335938	1.944335938	1.944335938	1.944335938	2.305175781	2.305175781	2.305175781	2.972167969	2.972167969	2.972167969	2.972167969	2.971679688	2.971679688	2.971679688	2.971679688	2.971679688	2.971679688	8.609375
14.86510677	14.8651	13.74280108	13.74280108	13.7428	13.74280108	13.74280108	13.74280108	13.31953891	13.31953891	13.31953891	13.31953891	13.31953891	13.31953891	13.31953891	13.31953891	13.31953891	13.21658084	13.21658084	13.21658084	12.96965895	12.96965895	12.96965895	12.96965895	11.65099191	11.65099191	11.65099191	11.65099191	11.65099191	11.65099191	10.55338584