Optimizing Multi-view Broad Learning Systems via Graph Embedding and Adaptive Membership Functions

M.Sc. Thesis

by

Moksha Pathak

DEPARTMENT OF MATHEMATICS

INDIAN INSTITUTE OF TECHNOLOGY INDORE

MAY 2025

Optimizing Multi-view Broad Learning Systems via Graph Embedding and Adaptive Membership Functions

A THESIS

Submitted in partial fulfillment of the

requirements for the award of the degree

of

Master of Science

by

Moksha Pathak

(Roll No. **2303141009**)

Under the guidance of

Prof. M. Tanveer and Dr. Priyamvada

DEPARTMENT OF MATHEMATICS

INDIAN INSTITUTE OF TECHNOLOGY INDORE

MAY 2025

INDIAN INSTITUTE OF TECHNOLOGY INDORE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **Optimizing Multi-view Broad Learning Systems via Graph Embedding and Adaptive Membership Functions** in the partial fulfillment of the requirements for the award of the degree of **Master of Science** and submitted in the **Department of Mathematics**, **Indian Institute of Technology Indore**, is an authentic record of my own work carried out during the time period from July 2024 to May 2025 under the supervision of **Prof. M. Tanveer and Dr. Priyamvada** , Department of Mathematics, IIT Indore.

The matter presented in this thesis by me has not been submitted for the award of any other degree of this or any other institute.

Moksha

10/06/25

Signature of the student with date

(**Moksha Pathak**)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.
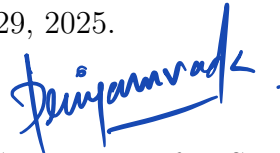
10/06/2025   13/06/2025

Signature of Thesis Supervisors with date

(**Prof. M. Tanveer and Dr. Priyamvada**)

Moksha Pathak has successfully given her M.Sc. Oral Examination held on May 29, 2025.

Signature of Supervisors of M.Sc Thesis

Date: **10/06/2025**

Signature of Convener, DPGC

Date:13/06/2025

"Dedicated to My Parents and Friends"

Acknowledgements

I extend my deepest gratitude to my Master's Thesis supervisors, **Prof. M. Tanveer and Dr. Priyamvada** , Department of Mathematics at IIT Indore, for their invaluable guidance, unwavering support, and profound mentorship throughout this memorable journey. Prof. Tanveer's expertise in the domain of machine learning and Dr. Priyamvada's expertise in graphs have been instrumental in shaping the trajectory of our project. Their profound insights, meticulous attention to detail, and unwavering commitment to excellence have been a constant source of inspiration for me. Beyond academia, their mentorship transcended conventional boundaries. They embraced the role of a parent figure on campus, offering academic guidance and a nurturing environment where I could thrive and grow personally and professionally.

Furthermore, I offer my thanks to Prof. Swadesh Kumar Sahoo, Dr. V. Antony Vijesh, Dr. Ashisha Kumar, and Dr. Charitha Cherugondi for their persistent assistance, motivation, and guidance during my stay at IIT Indore.

I would like to extend my heartfelt gratitude to Dr. Sanjeev Singh, Head of the Department of Mathematics, Dr. Vijay Kumar Sohani, DPGC, Department of Mathematics, and the entire Department of Mathematics for providing excellent facilities and support during my stay at IIT Indore.

I express my sincere appreciation to Mr. Abdul Quadir and Mr. Md Sajid members of OPTIMAL Lab, for their significant contributions to this endeavor. As mentors, collaborators, and, most importantly, as elder brothers, their consistent backing, inspiration, and direction were crucial in the development of this research project. Their readiness to exchange thoughts, knowledge, and assets has been priceless. They have undeniably served as the cornerstones of my academic progress.

I am deeply grateful to my parents for their constant love, unwavering support, and belief in my abilities throughout my academic journey. Their encouragement has been a steadfast pillar, providing me with the strength and motivation to pursue my goals. Additionally, I extend my heartfelt thanks to

<u>Publications</u>

1. *M. Tanveer, **M. Pathak**, A. Quadir, M. Sajid, and Priyamvada. "GrMv-BLS: Graph Embedded Multi-View Broad Learning System"* - **Under Review**

2. *M. Tanveer, **M. Pathak**, M. Sajid, A. Quadir, and Priyamvada. "Generalized Bell-Shaped Function with Broad Learning System and its Multi-View Extension"* - **Under Preparation**

Abstract

Randomized neural networks (RNNs) have emerged as efficient alternatives to traditional deep learning models by replacing iterative backpropagation with random feature projections and closed-form solutions. RNNs includes random vector functional link (RVFL) networks which offer faster training while maintaining competitive performance. However, their shallow architectures often limit feature representation capability. The broad learning system (BLS) addresses this limitation through a flat, incremental network structure that expands horizontally via randomly generated feature and enhancement nodes. Unlike deep networks, BLS computes output weights analytically in broader manner using pseudo-inverse methods, achieving remarkable efficiency. While successful in applications ranging from medical diagnosis to industrial fault detection, standard BLS suffers from some drawbacks, including sensitivity to outliers, poor handling of class imbalance as all samples contribute equally, limited data understanding from single-view processing, and inability to capture geometric properties of data. Limited understanding of the BLS causes not capturing the complex relationships within the datasets, resulting in poor generalization performance and less reliability on the traditional BLS. Furthermore, Noise and imbalanced classes are common challenges in real-world datasets.

To overcome these limitations of BLS, this thesis makes three fundamental contributions: the class probability-based generalized bell-shaped broad learning system (CPBS-BLS); its multi-view extension, the class probability-based generalized bell-shaped multi-view broad learning system (CPBS-MvBLS); and the graph embedded multi-view broad learning system (GrMv-BLS). First, the proposed CPBS-BLS enhances robustness through an integrated framework that combines generalized bell-shaped membership functions with class probability density estimation, enabling adaptive sample weighting that automatically suppresses outliers while emphasizing high-confidence regions near class centroids. Second, the proposed multi-view extension CPBS-MvBLS ex-

tends this capability to multi-view data sources by incorporating view-specific membership weighting that simultaneously understands the correlation between the views and addresses class imbalance across multiple feature spaces. Finally, the graph embedded multi-view BLS (GrMv-BLS) introduces a comprehensive architecture that unifies multi-view learning with graph regularized broad learning, incorporating both intrinsic and penalty graph embeddings for topological structure preservation and local fisher discriminant analysis (LFDA) weighted subspace learning for optimal discriminative projection. By addressing BLS's core limitations while preserving its efficiency advantages, these advances enable more reliable deployment in real-world scenarios involving imperfect, imbalanced, and multi-view data. The proposed frameworks establish new state-of-the-art performance while maintaining the simplicity that makes BLS practically valuable.

Contents

List of Figures

List of Tables

List of Abbreviations

BLS Broad Learning System

CPBS-BLS Class Probability based Bell Shaped Broad Learning System

CPBS-MvBLS Class Probability based Bell Shaped Multi View Broad Learning System

ELM Extreme Learning Machine

GE Graph Embedding

GrMv-BLS Graph Embedded Multi-View Broad Learning System

IID Identical and Independent

LFDA Local Fisher Discriminant Analysis

MVL Multi-View Learning

RNN Randomized Neural Network

RVFL Random Vector Functional Link Network

SVM Support Vector Machine

Chapter 1

Introduction

The broad learning system (BLS), introduced by Chen and Liu in 2018 [1], represents a significant advancement in the field of randomized neural networks. The BLS framework comprises four layers: an input layer, a feature mapping layer, an enhancement layer, and an output layer. Within the BLS framework, the input nodes are transformed to feature nodes through randomly generated weights, biases, and feature maps. Afterward, the feature nodes are processed to yield enhancement nodes with weights generated randomly and by applying an activation function. With all feature and enhancement nodes connected to the output nodes, a closed-form solution for the output weights is obtained through the least squares method. The BLS model garnered widespread attention as an influential framework that leads to several extensions and improvements for various real-world applications, including classification [2], regression [3], semi-supervised learning [4], unsupervised learning [5] and so on.

The motivation for studying and advancing BLS stems from several unresolved challenges and opportunities. While BLS offers fast training and strong generalization, its reliance on random weights can sometimes lead to instability, particularly in high-dimensional or noisy datasets. Furthermore, its inability to effectively capture the complex underlying relationships and critical information within datasets, which arises from a lack of understanding of the data. Additionally, the BLS model may overlook geometric structure, reducing its effectiveness in capturing subtle dataset features.

To address the aforementioned limitations, this thesis aims to explore the theoretical foundations of BLS, propose novel enhancements to improve its robustness and scalability, and evaluate its performance across diverse real-world applications. By addressing key limitations and expanding its applicability, this research seeks to contribute to the development of efficient, interpretable,

and adaptable machine learning models. The findings could have significant implications for industries requiring fast, reliable, and resource-efficient AI solutions, from healthcare diagnostics to autonomous systems.

1.1 Motivation

The rapid advancement of machine learning has led to the development of various neural network architectures, among which the BLS has emerged as a computationally efficient alternative to deep learning models. However, the standard BLS framework faces significant limitations, such as sensitivity to outliers, noise, and class imbalance, which degrade its performance in real-world scenarios. BLS often struggles to capture the underlying geometric and topological structures of data, limiting its ability to fully exploit complex relationships within datasets. In addition, single-view data alone may not adequately encapsulate the entirety of the content across various samples, leading to the common practice of gathering information through diverse and distinct feature sets. These shortcomings highlight the need for enhanced BLS variants that can address these challenges while retaining the model's inherent efficiency.

The integration of multi-view learning (MVL) [6] and graph embedding (GE) [7] techniques presents a promising direction for improving BLS. MVL leverages complementary information from diverse feature representations, enabling a more comprehensive understanding of data. The primary objective of MVL is to analyze datasets that can be represented through various feature sets. Information derived from multiple viewpoints typically provides complementary insights, enriching the overall understanding of the data. Meanwhile, graph embedding preserves the intrinsic geometric relationships among data points, enhancing the model's ability to capture local and global structures. These advancements are particularly relevant for applications where data is inherently multi-faceted or exhibits complex interdependencies, such as medical diagnosis, financial forecasting, and sensor data analysis. By combining these techniques with BLS, we develop a robust GrMv-BLS model that are not only computationally efficient but also improves the generalization performance of the traditional BLS.

In addition, the motivation behind this thesis is to bridge the gap between the theoretical strengths of BLS and its practical limitations by introducing innovative extensions that integrate probabilistic weighting mech-

anisms within BLS. The proposed models, CPBS-BLS and CPBS-MvBLS, address the challenges of outliers and class imbalance through adaptive sample weighting. These advancements aim to provide scalable, noise-resistant solutions for real-world applications, ensuring both robustness and generalization. By rigorously evaluating these models on benchmark datasets and providing theoretical guarantees, this thesis seeks to contribute to the ongoing evolution of BLS frameworks, paving the way for future research in adaptive and structurally aware learning systems.

1.2 Objectives

This thesis aims to achieve the following objectives:

- to present a comprehensive literature review on multiview learning, while also providing a concise explanation of BLS, membership functions, and the graph embedding framework.

- To develop a novel BLS model incorporating multiview learning with graph embedding to enhance the classification problem of the traditional BLS. Additionally, to develop a bell-shaped membership based BLS framework with class probability and imbalance ratio to handle the outliers and imbalanced classes.

- To rigorously evaluate the computational complexity of the proposed models and to derive comprehensive generalization and consistency bounds to substantiate their theoretical reliability. Additionally, we aim to provide statistically robust evidence demonstrating that our models significantly outperform baseline models.

1.3 Contributions of the Thesis

Here, we give a summary of the contribution of the thesis. The main contributions of the thesis are as follows:

- In this work, we propose a novel graph embedded multi-view broad learning system (GrMv-BLS). The proposed GrMv-BLS incorporates various perspectives of datasets through MVL together with intrinsic and penalty graph subspace learning through graph embedding into the BLS framework.

- By integrating different perspectives of the dataset through MVL approach, the proposed GrMv-BLS model enhances the overall understanding of the BLS model. Thereby, GrMv-BLS provides a more robust framework with improved adaptability and accuracy in processing diverse datasets.

- In order to effectively preserve geometric information from multiview data, the proposed GrMv-BLS model leverages intrinsic and penalty graphs to facilitate subspace learning (SL), which make use of graph regularization parameters alongside a local fisher discriminant analysis (LFDA) weighting scheme.

- We derive novel generalization and consistency error bounds for the proposed GrMv-BLS via Rademacher complexity. We conducted the statistical and hyperparameters analyses of the proposed GrMv-BLS model to ensure the reliability of the model.

- In addition, this thesis proposes a novel model, CPBS-BLS, designed to enhance the performance of the standard BLS framework. The proposed approach introduces an innovative integration of a bell-shaped membership function, class probability estimation, and imbalance ratio weighting to effectively improve classification accuracy, particularly in noisy and imbalanced datasets.

- Furthermore, we propose CPBS-MvBLS, which extends these innovations to the multi-view learning context by incorporating the gbell membership function and class probability based imbalance weighting within MvBLS.

- The proposed CPBS-BLS integrates a class probability based gbell function with the imbalance ratio into BLS to handle outliers and imbalanced classes effectively. It assigns adaptive weights by evaluating each sample's closeness to the class centers and neighborhood density, favoring clustered points near centers while suppressing distant or sparse outliers.

- The proposed CPBS-MvBLS tackles outliers and imbalanced classes while maintaining the model's ability to recognize complex data relationships via a multi-view learning approach. This allows the model to leverage complementary information from different feature sets, further enhancing predictive accuracy while handling the imbalanced noisy data.

- We derive novel generalization and consistency error bounds for the proposed CPBS-BLS and CPBS-MvBLS models, and perform statistical and hyperparameter analyses of these models.

1.4 Organization of the Thesis

This dissertation comprises five main chapters, with a concise overview of each provided below:

[1] In Chapter 2, we provide the mathematical formulation of BLS and present the literature review on multi-view learning. In addition, we explain the graph embedding framework and membership functions.

[2] In Chapter 3, we propose the graph embedded multi view broad learning system (GrMv-BLS) and present the experimental results. We also employ statistical analyses and determine the generalization and consistency bound to theoretically establish the proposed model.

[3] In Chapter 4, we introduce the class probability based bell-shaped BLS (CPBS-BLS) and its multi-view extension (CPBS-MvBLS). We present experimental results and show the reliability of the models through statistical and theoretical analyses.

[4] In Chapter 5, we present a comprehensive summary of this thesis and outline potential directions for future research to further advance the field.

Chapter 2

Related Work

This chapter conducts a systematic review of existing works in multi-view learning. Furthermore, it introduces a rigorous mathematical exposition of the BLS, generalized class probability based bell shaped membership functions, and the graph embedding (GE) framework. These formulations serve to establish a theoretical groundwork for subsequent discussions and analytical developments presented in this thesis.

2.1 Literature Review on Multi-view Learning

Applications in the real world often utilize data characterized by multiple perspectives. Single-view data alone may not adequately encapsulate the entirety of the content across various samples, leading to the common practice of gathering information through diverse and distinct feature sets. The primary objective of MVL is to analyze datasets that can be represented through various feature sets. Information derived from multiple viewpoints typically provides complementary insights, enriching the overall understanding of the data. Unlike conventional one-view learning approaches, which construct a learning function based on one feature view, MVL develops learning functions for each individual feature view. Subsequently, these functions are integrated to enhance the overall learning performance by leveraging the intersecting information inherent in the datasets from different perspectives. Thereby, MVL is dedicated to harnessing multi-view data to boost the efficacy and accuracy of learning models.

MVL has numerous applications across various domains. [8] develops an MVL framework that leverages multiple camera viewpoints to understand the 3D shape of pedestrians to enhance the person re-identification systems.

Sindhwani et al. [9] introduces a co-regularization framework that learns classifiers in each view using multi-view regularization. It optimizes agreement and smoothness across labeled and unlabeled data, extending standard methods like SVM [10] and BLS for multi-view semi-supervised learning. The time-aware balanced MVL (TABLE) is developed for stock ranking [11]. Another significant advancement is the MVL with twin parametric margin support vector machine (MvTPMSVM) [12], which enhances the performance of twin parametric margin support vector machine by capturing intricate non-linear relationships within the data. In order to improve the accuracy and reliability of the DNA binding proteins forecasting, Quadir et al. [13] develops the multi-view RVFL (MvRVFL) network.

2.2   Graph Embedding

Let $\mathcal{G} = (V, E)$ be the graph, where V and E are the vertex set and the edge set of the graph, respectively. Suppose d be the dimension of the projected space such that $d << |V|$. The goal of graph embedding [14] is to map the graph into a d-dimensional space while preserving its structural properties in the projected space. By offering a quick and effective way to compute graph similarities, graph embedding improves graph classification problems. Furthermore, graph embedding captures the underlying structural features of the graph. Let $X = \{x_1, x_2, ..., x_k\} \in \mathbb{R}^{N \times D}$ be the input matrix, where N is the number of data points and D is the number of features. Let $G_i = \{X, T_i\}$ and $G_p = \{X, T_p\}$ represent the intrinsic and penalty graphs corresponding to X, respectively. Here, $T_i \in \mathbb{R}^{N \times N}$ is the weight matrix that defines the relationships between pairs of graph vertices, and $T_p \in \mathbb{R}^{N \times N}$ adds penalty terms for specific pairs of vertices. The graph embedding optimization problem is stated as follows:

$$g^* = \underset{\mathrm{tr}(g_0^T X^T U X g_0)=s}{\arg\min} \sum_{k \neq N} \|g_0^T x_k - g_0^T x_N\| T_{i_{kN}}$$
$$= \underset{\mathrm{tr}(g_0^T X^T U X g_0)=s}{\arg\min} \mathrm{tr}(g_0^T X^T L X g_0), \tag{2.1}$$

where $\mathrm{tr}(\cdot)$ denotes the trace operator. $L = D_i - T_i \in \mathbb{R}^{N \times N}$ and $U = D_p - T_p \in \mathbb{R}^{N \times N}$ are the Laplacian matrices corresponding to the intrinsic graph G_i and the penalty graph G_p, respectively. $D_i \in \mathbb{R}^{N \times N}$ and $D_p \in \mathbb{R}^{N \times N}$ are the diagonal matrices with diagonal entries $d_{i_{kk}} = \sum_{l=1}^N T_{i_{kl}}$ and $d_{p_{kk}} = \sum_{j=1}^N T_{P_{kj}}$. g_0 denotes the projection matrix for the embedding process, while s represents

the constraint constant. The optimization problem (2.1) can be written as follows:

$$G_i z = \lambda G_p z, \tag{2.2}$$

where $G_i = X^T L X \in \mathbb{R}^{D \times D}$ and $G_p = X^T U X \in \mathbb{R}^{D \times D}$. λ is the eigenvalue. The eigenvector matrix of $G = G_p^{-1} G_i$ acts as the transformation matrix. In this formulation, G incorporates the connections of data samples from both the intrinsic and penalty graph matrices.

2.3 Generalized Bell-Shaped Membership Function

The generalized bell-shaped (gbell) [15] is a membership function which assigns sample-specific weights. The gbell membership function is formulated by integrating class probability with the bell-shaped function. This hybrid approach leverages probabilistic class information to enhance the interpretability and adaptability of the membership function, making it particularly suitable for classification and fuzzy inference systems. The bell-shaped function with class probability and imbalance ratio [16] calculates a membership value for each sample that reflects its reliability. Samples closer to the class center and surrounded by similar class neighbors receive higher weights, while potential outliers, those far from the center or in low-density regions, are assigned significantly lower weights. This adaptive weighting scheme [17] effectively diminishes the influence of outliers during model training. Specifically, the gbell component suppresses distant points smoothly, while the class probability term further penalizes inconsistent samples. As a result, the gbell function preserves the discriminative power of clean data while robustly handling datasets contaminated with outliers. Mathematically, the bell-shaped function $\mathcal{M}_{gb}(a_i)$ for a sample a_i is defined as:

$$\mathcal{M}_{gb}(a_i) = \begin{cases} \frac{1}{1+\left(\frac{\|\mathcal{K}(a_i)-Cen_+\|}{\omega}\right)^{2\tau}}, & \text{if } b_i = +1, \\ \frac{1}{1+\left(\frac{\|\mathcal{K}(a_i)-Cen_-\|}{\omega}\right)^{2\tau}}, & \text{if } b_i = -1, \end{cases} \tag{2.3}$$

where b_i is output label of a_i sample, $\mathcal{K}(\cdot)$ maps the sample to a higher-dimensional space, Cen_+ and Cen_- denote the geometric centers of the positive and negative classes (computed as $Cen_+ = \frac{1}{M_+} \sum_{b_i=+1} a_i$ and $Cen_- =$

11

$\frac{1}{M_-} \sum_{b_i=-1} a_i$), M_+ and M_- are the number of major and minor class samples, ω controls the function's spread, and τ adjusts the steepness of the membership curve.

To further refine the membership assignments, the gbell function is augmented with class probability $\mathcal{M}_p(a_i)$, which measures local class density. For a sample a_i, $\mathcal{M}_p(a_i)$ is computed as the ratio of neighboring samples sharing its class label within an ϵ-radius:

$$\mathcal{M}_p(a_i) = \frac{|\{a_r : \|\mathcal{K}(a_r) - \mathcal{K}(a_i)\| \leq \epsilon, b_r = b_i\}|}{|\{a_r : \|\mathcal{K}(a_r) - \mathcal{K}(a_i)\| \leq \epsilon\}|}. \tag{2.4}$$

where ϵ is any positive number. Low probability values flag potential outliers, while high values indicate strong class alignment.

The final membership score $\mathcal{M}(a_i)$ synthesizes the gbell function, class probability, and class imbalance adjustment. For majority-class samples, it combines $\mathcal{M}_{gb}(a_i)$, $\mathcal{M}_p(a_i)$, and the inverse imbalance ratio m; minority-class samples receive a fixed membership of 1 to preserve their influence:

$$\mathcal{M}(a_i) = \begin{cases} \mathcal{M}_{gb}(a_i) \times \mathcal{M}_p(a_i) \times \frac{1}{m}, & \text{if } a_i \in \text{majority class}, \\ 1, & \text{if } a_i \in \text{minority class}. \end{cases} \tag{2.5}$$

Finally, the score matrix S is as follows:

$$S = diag(\mathcal{M}(a_1), \mathcal{M}(a_2), ..., \mathcal{M}(a_M)). \tag{2.6}$$

2.4 Broad Learning System

The broad learning system (BLS) [1] comprises four distinct layers, namely, the input layer, the feature layer, the enhancement layer, and the output layer. The input layer is interconnected with the feature layer through randomly generated weights and biases. Similarly, the feature layer is linked to the enhancement layer using randomly assigned weights and biases. Finally, after the concatenation, the feature layer and the enhancement layer are connected with the output layer through an unknown weight matrix. To find the closed-form solution for the output weight matrix, we employ the pseudo-inverse technique. Consider the input matrix $X \in \mathbb{R}^{N \times D}$, where N represents the number of samples and D denotes the number of features, respectively. Let N_1 be the number of feature groups, where each feature group has N_2 nodes. Then, the ith feature group for the feature layer F is given as follows:

$$F_i = \mathcal{F}_i(XW_{1_i} + B_{1_i}) \in \mathbb{R}^{N \times N_2}, \ i = 1, 2, ..., N_1,$$
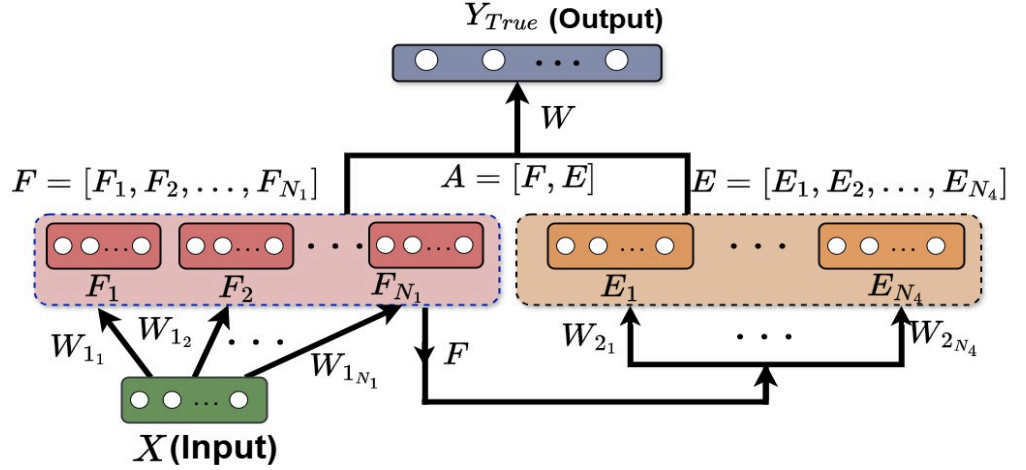
Figure 2.1: Architecture of the BLS model.

where $\mathcal{F}_i$ is the feature map, $W_{1_i} \in \mathbb{R}^{D \times N_2}$ is randomly generated weight matrix connecting the input matrix to i^{th} feature group, $B_{1_i} \in \mathbb{R}^{N \times N_2}$ is randomly generated bias matrix corresponding to the ith feature group. Then, the feature layer for N_1 number of feature groups is represented as:

$$F = [F_1, F_2, ..., F_{N_1}] \in \mathbb{R}^{N \times N_1 N_2}.$$

Let $W_{2_j} \in \mathbb{R}^{N_1 N_2 \times N_3}$ and $B_{2_j} \in \mathbb{R}^{N \times N_3}$ be the randomly generated weight matrix and bias matrix for j^{th} enhancement group, respectively. $\mathcal{A}_j$ is the activation function to generate non-linear features. The jth enhancement group for the enhancement layer E is given as follows:

$$E_j = \mathcal{A}_j(FW_{2_j} + B_{2_j}) \in \mathbb{R}^{N \times N_3}, \ j = 1, 2, ..., N_4,$$

where N_4 is the number of enhancement groups, and N_3 is the number of nodes in each enhancement group. Then, the enhancement layer for N_4 enhancement groups is represented as:

$$E = [E_1, E_2, ..., E_{N_4}] \in \mathbb{R}^{N \times N_3 N_4}.$$

Finally, the feature layer F and enhancement layer E are concatenated to form the matrix $A = [F, E] \in \mathbb{R}^{N \times (N_1 N_2 + N_3 N_4)}$. Let $W \in \mathbb{R}^{(N_1 N_2 + N_3 N_4) \times C}$ represents the weight matrix connecting the concatenated layer A to the predicted output layer Y_{Pred}. The predicted output Y_{Pred} is given as follows:

$$Y_{Pred} = [F, E]W = AW \in \mathbb{R}^{N \times C}. \tag{2.7}$$

13

The optimization problem for BLS is formulated as follows:

$$\arg\min_{W} \frac{1}{2}\|W\|_2^2 + \frac{c}{2}\|\zeta\|_2^2$$

$$\text{s.t.} \quad AW - Y_{True} = \zeta, \tag{2.8}$$

where c is a regularization parameter and ζ is the error.

The solution to the optimization problem (2.8) is given by:

$$W = \begin{cases} (A^T A + \frac{1}{c}I)^{-1} A^T Y_{True}, & \text{if } (N_1 N_2 + N_3 N_4) \leq N, \\ A^T \left(\frac{1}{c}I + AA^T\right)^{-1} Y_{True}, & \text{if } N < (N_1 N_2 + N_3 N_4). \end{cases} \tag{2.9}$$

where I denotes the suitable dimension's identity matrix.

Note: The notations introduced in this section are section-specific and do not carry over to other parts of this work.

Chapter 3

GrMv-BLS: Graph Embedded Multi-View Broad Learning System

The broad learning system (BLS) is a widely used framework for classification tasks, known for its strong generalization performance. However, it often overlooks the latent information within the datasets and fails to preserve the data's topological structure, limiting its depth of analysis. To address these issues, we propose a graph embedded multi-view broad learning system (GrMv-BLS), integrating multi-view learning (MVL) and graph embedding (GE). GrMv-BLS leverages multiple perspectives to extract richer information and retains graph structural data through graph subspace learning. GrMv-BLS's optimization problem includes a coupling term to minimize errors across all views simultaneously. Evaluated on UCI, KEEL, and AwA datasets, GrMv-BLS demonstrates superior generalization over baseline models, supported by statistical and hyperparameter sensitivity analyses. Moreover, theoretical analysis based on Rademacher complexity guarantees the proposed GrMv-BLS model's robust generalization capability.

3.1 Notations

Let the training dataset for two views be denoted as follows:
$V = \{(x_k^A, x_k^B, y_k)| \ x_k^A \in \mathbb{R}^{1 \times a}, \ x_k^B \in \mathbb{R}^{1 \times b}, \ y_k \in \{-1, 1\}, \ k = 1, 2, ..., N\}$,
where N is the number of training samples, x_k^A (x_k^B) stands for the feature vector for k^{th} sample of the dataset for view-A (view-B) and y_k stands for

the corresponding label of the kth feature vector for both views. We assume $X_A \in \mathbb{R}^{N \times a}$ as the input matrix of view-A, $X_B \in \mathbb{R}^{N \times b}$ as the input matrix of view-B and $Y_{true} \in \mathbb{R}^{N \times 1}$ is the label vector corresponding to all the samples. a and b represent the number of features in view-A and view-B, respectively, while C denotes the number of classes. **Note:** these notations will be used only for this chapter.

3.2 The Proposed Graph Embedded Multi View Broad Learning System (GrMv-BLS)

We describe the comprehensive mathematical derivation of the GrMv-BLS model, which incorporates MVL and integrates graph embedding through a concatenated layer formed by feature and enhancement layers. GrMv-BLS improves the basic BLS model by incorporating the subspace learning (SL) criterion and various features through different views to precisely understand the data. To find the output weights, we formulate an optimization problem that includes an L_2 regularization term for both the views, which minimizes the structural risk of the model. Optimization problem minimizes the error term, thereby improves the model's performance. Furthermore, we introduce an L_2 graph regularization term to preserve graph relationships, leveraging both intrinsic graphs and penalty graphs from both the views. The intrinsic graph captures the connections between vertices, while the penalty graph encodes specific relationships among them. To minimize errors simultaneously from both the views, we incorporate a coupling term to ensure improved model performance effectively. In addition, we integrate local fisher discriminate analysis (LFDA) [18] within the graph embedding framework to improve topological preservation and class separation while preserving the data graph structure. By combining these techniques, GrMv-BLS leverages the complementary strengths of each component to improve both the learning performance and generalization capabilities of the model, making it more robust in complex learning tasks. Therefore, GrMv-BLS can effectively handle the intricate data structures by integrating information from multi-view data and leveraging the LFDA scheme under the graph embedding framework. The formulation of the proposed GrMv-BLS model is shown below. Suppose the ith feature group for view-A is given as:

$$F_i^A = \mathcal{F}_i(X_A W_{1_i}^A + B_{1_i}^A) \in \mathbb{R}^{N \times N_2}, \ i = 1, 2, ..., N_1,$$

where $W_{1_i}^A \in \mathbb{R}^{a \times N_2}$ and $B_{1_i}^A \in \mathbb{R}^{N \times N_2}$ are the randomly generated weight and bias matrix for i^{th} feature group corresponding to view-A. $\mathcal{F}_i$ is a feature map for i^{th} feature group. Then, the feature layer is represented as:

$$F^A = [F_1^A, F_2^A, ..., F_{N_1}^A] \in \mathbb{R}^{N \times N_1 N_2}. \tag{3.1}$$

Let $W_{2_j}^A \in \mathbb{R}^{N_1 N_2 \times N_3}$ and $B_{2_j}^A \in \mathbb{R}^{N \times N_3}$ be the randomly generated weight matrix and bias matrix for the jth enhancement group corresponding to view A. $\mathcal{A}_j$ be the activation function utilized to generate non-linear features. The jth enhancement group for view A is given as follows:

$$E_j^A = \mathcal{A}_j(F^A W_{2_j}^A + B_{2_j}^A) \in \mathbb{R}^{N \times N_3}, \ \ i = 1, 2, ..., N_4.$$

Then, the enhancement layer is represented as:

$$E^A = [E_1^A, E_2^A, ..., E_{N_4}^A] \in \mathbb{R}^{N \times N_3 N_4}. \tag{3.2}$$

In a similar manner, view-B's feature layer and the enhancement layer can be formed and shown as $F^B \in \mathbb{R}^{N \times N_1 N_2}$ and $E^B \in \mathbb{R}^{N \times N_3 N_4}$, respectively. The optimization problem of the proposed GrMv-BLS model is formulated as below:

$$\underset{\gamma_1, \gamma_2}{\arg\min} \frac{c_1}{2}||\zeta_1||_2^2 + \frac{c_2}{2}||\zeta_2||_2^2 + \frac{c_3}{2}||\gamma_1||_2^2 + \frac{1}{2}||\gamma_2||_2^2 + \frac{\theta_1}{2}||G_1^{\frac{1}{2}}\gamma_1||_2^2$$

$$+ \frac{\theta_2}{2}||G_2^{\frac{1}{2}}\gamma_2||_2^2 + \rho\zeta_1^t\zeta_2 \tag{3.3}$$

$$\text{s.t.} \quad Z_1\gamma_1 - Y_{True} = \zeta_1, \quad Z_2\gamma_2 - Y_{True} = \zeta_2,$$

where $Z_1 = [F^A, E^A]$ and $Z_2 = [F^B, E^B]$. The detailed description for each component of the optimization problem (3.3) is given as follows:

1. ζ_1 and ζ_2 are the observational error, associated with view-A and view-B, respectively. By minimizing the observational error $||\zeta_1||_2^2$ and $||\zeta_2||_2^2$ terms, we can directly minimize the overall error of the model.

2. γ_1 is the weight matrix to connect Z_1 and the output layer, while γ_2 is the weight matrix to connect Z_2 and the output layer. Minimizing the L_2 regularization $||\gamma_1||_2^2$ and $||\gamma_2||_2^2$ terms to protect the model from the overfitting risk.

3. G_1 and G_2 are the graph embedded matrices for view-A and view-B, respectively. By minimizing graph regularization $||G_1^{\frac{1}{2}}\gamma_1||_2^2$ and $||G_2^{\frac{1}{2}}\gamma_2||_2^2$ terms, the model can preserve the graphical relationships of both the dataset on the projected space.

4. $\zeta_1^t\zeta_2$ represents the coupling term that effectively minimizes error while simultaneously integrating information from both views.

17

5. c_1, c_2, and c_3 are regularization coefficients, while ρ represents the coupling parameter and θ_1 and θ_2 represent the regularization parameters of the graph.

The corresponding Lagrangian of the optimization problem (3.3) is:

$$L = \frac{c_1}{2}||\zeta_1||_2^2 + \frac{c_2}{2}||\zeta_2||_2^2 + \frac{c_3}{2}||\gamma_1||_2^2 + \frac{1}{2}||\gamma_2||_2^2 + \frac{\theta_1}{2}||G_1^{\frac{1}{2}}\gamma_1||_2^2 + \frac{\theta_2}{2}||G_2^{\frac{1}{2}}\gamma_2||_2^2$$
$$+ \rho\zeta_1^t\zeta_2 - \lambda_1^t(Z_1\gamma_1 - Y_{True} - \zeta_1) - \lambda_2^t(Z_2\gamma_2 - Y_{True} - \zeta_2). \tag{3.4}$$

Partially differentiating (3.4) w.r.t. ζ_1, ζ_2, γ_1, γ_2, λ_1 and λ_2, we obtain:

$$\frac{\partial L}{\partial \zeta_1} = c_1\zeta_1 + \rho\zeta_2 + \lambda_1 = 0, \tag{3.5}$$

$$\frac{\partial L}{\partial \zeta_2} = c_2\zeta_2 + \rho\zeta_1 + \lambda_2 = 0, \tag{3.6}$$

$$\frac{\partial L}{\partial \gamma_1} = c_3\gamma_1 + \theta_1 G_1\gamma_1 - Z_1^t\lambda_1 = 0, \tag{3.7}$$

$$\frac{\partial L}{\partial \gamma_2} = \gamma_2 + \theta_2 G_2\gamma_2 - Z_2^t\lambda_2 = 0, \tag{3.8}$$

$$\frac{\partial L}{\partial \lambda_1} = Z_1\gamma_1 - Y_{True} - \zeta_1 = 0, \tag{3.9}$$

$$\frac{\partial L}{\partial \lambda_2} = Z_2\gamma_2 - Y_{True} - \zeta_2 = 0. \tag{3.10}$$

Substituting (3.5) and (3.6) in (3.7) and (3.8) respectively, we obtain:

$$c_3\gamma_1 + \theta_1 G_1\gamma_1 + Z_1^t(c_1\zeta_1 + \rho\zeta_2) = 0, \tag{3.11}$$

$$\gamma_2 + \theta_2 G_2\gamma_2 + Z_2^t(c_2\zeta_2 + \rho\zeta_1) = 0. \tag{3.12}$$

Substituting (3.9) and (3.10) in (3.11) and (3.12), we obtain:

$$c_3\gamma_1 + \theta_1 G_1\gamma_1 + Z_1^t(c_1(Z_1\gamma_1 - Y_{True}) + \rho(Z_2\gamma_2 - Y_{True})) = 0, \tag{3.13}$$

$$\gamma_2 + \theta_2 G_2\gamma_2 + Z_2^t(c_2(Z_2\gamma_2 - Y_{True}) + \rho(Z_1\gamma_1 - Y_{True})) = 0. \tag{3.14}$$

After simplification, we get:

$$c_3\gamma_1 + \theta_1 G_1\gamma_1 + c_1 Z_1^t Z_1\gamma_1 + \rho Z_1^t Z_2\gamma_2 = Z_1^t(c_1 + \rho)Y_{True}, \tag{3.15}$$

$$\gamma_2 + \theta_2 G_2\gamma_2 + c_2 Z_2^t Z_2\gamma_2 + \rho Z_2^t Z_1\gamma_1 = Z_2^t(c_2 + \rho)Y_{True}. \tag{3.16}$$

Finally, we obtain:

$$(c_3 I_1 + \theta_1 G_1 + c_1 Z_1^t Z_1)\gamma_1 + (\rho Z_1^t Z_2)\gamma_2 = Z_1^t(c_1 + \rho)Y_{True}, \tag{3.17}$$

$$(\rho Z_2^t Z_1)\gamma_1 + (I_2 + \theta_2 G_2 + c_2 Z_2^t Z_2)\gamma_2 = Z_2^t(c_2 + \rho)Y_{True}, \tag{3.18}$$

where I_1 and I_2 represent the identity matrices of appropriate dimensions. Equations (3.17) and (3.18) can be written in the matrix form as follows:

$$\begin{bmatrix} c_3 I_1 + \theta_1 G_1 + c_1 Z_1^t Z_1 & \rho Z_1^t Z_2 \\ \rho Z_2^t Z_1 & I_2 + \theta_2 G_2 + c_2 Z_2^t Z_2 \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} = \begin{bmatrix} Z_1^t(c_1 + \rho) \\ Z_2^t(c_2 + \rho) \end{bmatrix} Y_{True}.$$
$$\tag{3.19}$$

After taking the inverse of the equation (3.19), we obtain:

$$
\begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} = \begin{bmatrix} c_3 I_1 + \theta_1 G_1 + c_1 Z_1^t Z_1 & \rho Z_1^t Z_2 \\ \rho Z_2^t Z_1 & I_2 + \theta_2 G_2 + c_2 Z_2^t Z_2 \end{bmatrix}^{-1} \begin{bmatrix} Z_1^t (c_1 + \rho) \\ Z_2^t (c_2 + \rho) \end{bmatrix} Y_{True}.
$$

$$(3.20)$$

Algorithm 1 Proposed GrMv-BLS Model 3.2

Input: The training datasets X_A for view-A and X_B for view-B, the parameters c_1, c_2, c_3, θ_1 θ_2, and ρ.

1: **Compute:** T_1^i, T_1^p, T_2^i, and T_2^p using equations (3.21) and (3.22) for Z_1 and Z_2.
2: **Compute:** Laplacian matrices $L_l = D_l^i - T_l^i$ and $U_l = L_l^p - T_l^p$ for Z_l ($l = 1, 2$).
3: **Compute:** $G_1^i = Z_1^t L_1 Z_1$ and $G_1^p = Z_1^t U_1 Z_1$ for Z_1, and $G_2^i = Z_2^t L_2 Z_2$ and $G_2^p = Z_2^t U_2 Z_2$ for Z_2.
4: **Compute:** $G_1 = (G_1^p)^{-1} G_1^i$ and $G_2 = (G_2^p)^{-1} G_2^i$.
5: **Output:** γ_1 and γ_2, using equation (3.20).

3.2.1 Local Fisher Discriminant Analysis (LFDA)

In this subsection, the Z_1 and Z_2 matrices are utilized to compute the intrinsic and penalty graphs. For Z_1, we obtain $g_1^{int} = \{Z_1, T_1^i\}$ and $g_1^{pen} = \{Z_1, T_1^p\}$. Similarly, for Z_2, $g_2^{int} = \{Z_2, T_2^i\}$ and $g_2^{pen} = \{Z_2, T_2^p\}$. The intrinsic graph for Z_1 is $G_1^i = Z_1^T L_1 Z_1 \in \mathbb{R}^{(N_1 N_2 + N_3 N_4) \times (N_1 N_2 + N_3 N_4)}$, and the penalty graph is $G_1^p = Z_1^T U_1 Z_1 \in \mathbb{R}^{(N_1 N_2 + N_3 N_4) \times (N_1 N_2 + N_3 N_4)}$. The intrinsic and penalty graphs for Z_2 are $G_2^i = Z_2^T L_2 Z_2 \in \mathbb{R}^{(N_1 N_2 + N_3 N_4) \times (N_1 N_2 + N_3 N_4)}$ and $G_2^p = Z_2^T U_2 Z_2 \in \mathbb{R}^{(N_1 N_2 + N_3 N_4) \times (N_1 N_2 + N_3 N_4)}$, respectively. Here, L_1 and U_1 are the Laplacian matrices for view-A corresponding to intrinsic and penalty graph, respectively. L_2 and U_2 are the Laplacian matrices for view-B corresponding to intrinsic and penalty graph, respectively. We utilize graph embeddings through the LFDA weighting scheme. As a result, the intrinsic weights T_l^i and penalty weights T_l^p from LFDA technique are defined as below:

$$
[T_l^i]_{ij} = \begin{cases} \frac{a_{ij}}{N_{y_i}}, & y_i = y_j, \\ 0, & \text{otherwise,} \end{cases}
$$

$$(3.21)$$

$$
[T_l^p]_{ij} = \begin{cases} a_{ij} \left(\frac{1}{N} - \frac{1}{N_{y_i}} \right), & y_i = y_j, \\ \frac{1}{N}, & \text{otherwise,} \end{cases}
$$

$$(3.22)$$

where $l = 1, 2$, N_{y_i} represents the total number of samples in the y_i labeled class, and a_{ij} quantifies the similarity between z_i and z_j, with $z_i, z_j \in Z_k$ for

$k = 1, 2$. The kernel function used to compute the similarity measure is defined as follows:

$$a_{ij} = \exp\left(-\frac{\|z_i - z_j\|^2}{2\sigma^2}\right),$$

where σ represents the scaling parameter.

3.2.2 Time Complexity

We investigate the computational complexity of the GrMv-BLS framework. The initial step involves determining the time complexity associated with the computation of the graph embedding matrices G_1 and G_2, corresponding to Z_1 and Z_2. These embeddings are derived utilizing the graph embedding LFDA weighting scheme, incorporating intrinsic and penalty subspace learning methodologies from [19]. Specifically, the time complexity for calculating G_1 is given by: $\mathcal{O}((N_1N_2 + N_3N_4)^3 + (N_1N_2 + N_3N_4)^2 N)$ while for G_2, it is expressed as: $\mathcal{O}((N_1N_2 + N_3N_4)^3 + (N_1N_2 + N_3N_4)^2 N)$. Subsequently, when addressing the solution to equation (3.20), we find that the associated time complexity is: $\mathcal{O}(2[N_1N_2 + N_3N_4])^3$. Hence, the cumulative time complexity of the proposed GrMv-BLS model can be articulated as follows: $\mathcal{O}((N_1N_2 + N_3N_4)^3 + (N_1N_2 + N_3N_4)^2 N) + \mathcal{O}((N_1N_2 + N_3N_4)^3 + (N_1N_2 + N_3N_4)^2 N) + \mathcal{O}(2[N_1N_2 + N_3N_4])^3$. This can be approximated as: $\mathcal{O}((N_1N_2 + N_3N_4)^3)$.

3.2.3 Generalization Error Bound of GrMv-BLS

This subsection presents the analysis of the generalization capability of the proposed GrMv-BLS model via Rademacher complexity. The optimization problem can be rewritten as:

$$\underset{\gamma_1, \gamma_2}{\arg\min} \frac{c_1}{2} \sum_{k=1}^{N} \zeta_{1_k}^T \zeta_{1_k} + \frac{c_2}{2} \sum_{k=1}^{N} \zeta_{2_k}^T \zeta_{2_k} + \frac{c_3}{2} \|\gamma_1\|_2^2 + \frac{1}{2} \|\gamma_2\|_2^2 + \frac{\theta_1}{2} \|G_1^{\frac{1}{2}} \gamma_1\|_2^2 +$$

$$\frac{\theta_2}{2} \|G_2^{\frac{1}{2}} \gamma_2\|_2^2 + \rho \sum_{k=1}^{N} \zeta_{1_k}^T \zeta_{2_k} \tag{3.23}$$

$$\text{s.t.} \quad Z_1^{(x_k)} \gamma_1 - y_k = \zeta_{1_k}, \quad and \quad Z_2^{(x_k)} \gamma_2 - y_k = \zeta_{2_k}.$$

Applying y_k on both sides of the constraints of the rewritten optimization problem (3.23)

$$y_k Z_1^{(x_k)} \gamma_1 - 1 = y_k \zeta_{1_k}, \tag{3.24}$$

$$y_k Z_2^{(x_k)} \gamma_2 - 1 = y_k \zeta_{2_k}. \tag{3.25}$$

Rademacher Complexity: Given a data set $\mathcal{T} = \{x_1, ..., x_N\}$ from probability distribution $\mathcal{B}$ with uniformly independent value $\alpha = (\alpha_1, ..., \alpha_N)$ and independent samples x_i's, the empirical Rademacher complexity of a function class $\mathcal{H}$ in $\mathcal{T}$ is defined as the equation (3.26) and the Rademacher complexity is expressed as (3.27):

$$\hat{\mathcal{R}}_N(\mathcal{H}) = \mathbb{E}_\alpha \left[\sup_{h \in \mathcal{H}} | \frac{2}{N} \sum_{k=1}^{N} \alpha_k h(x_k) | \right], \qquad (3.26)$$

$$\mathcal{R}_N(\mathcal{H}) = \hat{\mathbb{E}}_{\mathcal{T}}[\hat{\mathcal{R}}_N(\mathcal{H})] = \hat{\mathbb{E}}_{\mathcal{T}_\alpha} \left[\sup_{h \in \mathcal{H}} | \frac{2}{N} \sum_{k=1}^{N} \alpha_k h(x_k) | \right]. \qquad (3.27)$$

Lemma 1. Let $\mathcal{H}$ be a class of functions mapping from an input space $\mathcal{T}$ to $[0, 1]$ and $\varphi \in (0, 1)$. Given a set of N independent samples $\{x_i\}_{i=1}^n$ drawn from a distribution $\mathcal{B}$, the following inequality holds with probability at least $1 - \varphi$ for all $h \in \mathcal{H}$:

$$\mathbb{E}_\mathcal{B}[h(x)] \leq \hat{\mathbb{E}}_\mathcal{B}[h(x)] + \hat{R}_N(\mathcal{H}) + 3\sqrt{\frac{\ln(2/\varphi)}{2N}},$$

where $\hat{\mathbb{E}}_\mathcal{B}[\cdot]$ is the empirical expectation and $\hat{R}_N(\mathcal{H})$ is the empirical Rademacher complexity of $\mathcal{H}$.

Lemma 2. (Rademacher Complexity of Bounded Linear Function Class). Let $\mathcal{T} = \{(x_1, y_1), ..., (x_N, y_N)\}$ be a dataset of N samples, and let $A^{(x)}$ denote the concatenated matrix of feature and enhancement layer for sample x. Consider the function set

$$\mathcal{H}_S = \left\{ h \mid h(x) = |A^{(x)} \gamma|, \|\gamma\| \leq S \right\},$$

where Euclidean norm of γ is bounded by S. Then, the empirical Rademacher complexity of $\mathcal{H}_S$ is given by:

$$\hat{R}_N(\mathcal{H}_S) = \frac{2S}{N} \sqrt{\sum_{k=1}^{N} A^{(x_k)^T} A^{(x_k)}}.$$

Lemma 3. (Rademacher Complexity of Lipschitz Function Composition). Let $\mathscr{S} : \mathbb{R} \to \mathbb{R}$ be a Lipschitz function with Lipschitz constant $\mathscr{D}$ such that $\mathscr{S}(0) = 0$. For any function class $\mathcal{H}$, the empirical Rademacher complexity of the composed class $\mathscr{S} \circ \mathcal{H}$ is bounded by:

$$\hat{R}_N(\mathscr{S} \circ \mathcal{H}) \leq 2\mathscr{D} \cdot \hat{R}_N(\mathcal{H}).$$

Theorem 1. Let $S \in \mathbb{R}^+$ be a norm bound, $\varphi \in (0, 1)$, and $\mathcal{T} = \{(x_k, y_k)\}_{k=1}^n$ a training set of N IID samples drawn from a probability distribution $\mathcal{B}$, where $y_k \in \{-1, +1\}$ and $x_k = (x_k^A, x_k^B)$. Define $A^{(x)} = \left(Z_1^{(x)}; -Z_2^{(x)} \right)$, where $Z_1^{(x)} = [F^A(x) \quad E^A(x)]$ and $Z_2^{(x)} = [F^B(x) \quad E^B(x)]$, with $F^A(x)[F^B(x)]$ feature layer

for data point x corresponding to view-A(B), and $E^A(x)[E^B(x)]$ enhancement layer for data point x corresponding to view-A(B). The function class $\mathcal{H}_S = \{h \mid h(x) = |A^{(x)}\gamma|, \|\gamma\| \leq S\}$, where $\gamma = (\gamma_1; \gamma_2)$. Then, every $d_s(x) = |Z_1^{(x)}\gamma_1 - Z_2^{(x)}\gamma_2| \in \mathcal{H}_S$ with probability greater than or equals to $1 - \varphi$ over $\mathcal{T}$ satisfies the bound:

$$\mathbb{E}_{\mathcal{B}}[d_s(x)] \leq 2S + 3S\mathcal{Q}_s\sqrt{\frac{\ln(2/\varphi)}{2N}} + \frac{4S}{N}\sqrt{\sum_{i=1}^{N}\left(\|Z_1^{(x_i)}\|^2 + \|Z_2^{(x_i)}\|^2\right)},$$

where $\quad \mathcal{Q}_s = \max_{x_i \in \sup(\mathcal{B})} \sqrt{\|Z_1^{(x_i)}\|^2 + \|Z_2^{(x_i)}\|^2}.$

Proof. Consider the piecewise function $\mathscr{O}: \mathbb{R} \to [0,1]$ defined as:

$$\mathscr{O}(x) = \begin{cases} -\frac{x}{S\mathcal{Q}_s}, & \text{if } -S\mathcal{Q}_s \leq x < 0, \\ \frac{x}{S\mathcal{Q}_s}, & \text{if } 0 \leq x \leq S\mathcal{Q}_s, \\ 1, & \text{otherwise.} \end{cases}$$

For any sample (x_k, y_k) drawn independently from $\mathcal{B}$, the boundedness condition $\|\gamma\| \leq S$ and the definition of $d_s(x_k)$ imply:

$$d_s(x_k) = |A^{(x_i)}\gamma| \leq S\|A^{(x_k)}\| = S\sqrt{\|Z_1^{(x_i)}\|^2 + \|Z_2^{(x_i)}\|^2} \leq S\mathcal{Q}_s,$$

where $\mathcal{Q}_s = \max_{x_i \in \sup(\mathcal{B})} \sqrt{\|Z_1^{(x_i)}\|^2 + \|Z_2^{(x_i)}\|^2}.$

The function $\hat{d}_s(x)$ ranges over $[-S\mathcal{Q}_s, S\mathcal{Q}_s]$, while $\mathscr{O}(\hat{d}_s(x))$ maps this interval to $[0,1]$. By Lemma 1, with probability at least $1 - \varphi$ over the training set $\mathcal{T}$, we have:

$$\mathbb{E}_{\mathcal{B}}[\mathscr{O}(\hat{d}_s(x))] \leq \hat{\mathbb{E}}_{\mathcal{T}}[\mathscr{O}(\hat{d}_s(x))] + \hat{R}_N(\mathscr{O} \circ \hat{\mathcal{H}}_S) + 3\sqrt{\frac{\ln(2/\varphi)}{2N}}. \tag{3.28}$$

Since $\mathscr{O}$ is Lipschitz with constant $\frac{1}{S\mathcal{Q}_s}$ and satisfies $\mathscr{O}(0) = 0$, Lemma 3 yield:

$$\hat{R}_N(\mathscr{O} \circ \hat{\mathcal{H}}_S) \leq \frac{2}{S\mathcal{Q}_s}\hat{R}_N(\hat{\mathcal{H}}_S).$$

After applying Lemma 2, we obtain:

$$\hat{R}_N(\mathscr{O} \circ \hat{\mathcal{H}}_S) \leq \frac{4}{N\mathcal{Q}_s}\sqrt{\sum_{k=1}^{N}(\|Z_1^{(x_k)}\|^2 + \|Z_2^{(x_k)}\|^2)}. \tag{3.29}$$

Combining results (3.28) and (3.29), we obtain:

$$\mathbb{E}_{\mathcal{B}}[\mathscr{O}(\hat{d}_s(x))] \leq \hat{\mathbb{E}}_{\mathcal{T}}[\mathscr{O}(\hat{d}_s(x))] + 3\sqrt{\frac{\ln(2/\varphi)}{2N}} \tag{3.30}$$

$$+ \frac{4}{N\mathcal{Q}_s}\sqrt{\sum_{k=1}^{N}\left(\|Z_1^{(x_k)}\|^2 + \|Z_2^{(x_k)}\|^2\right)}.$$

22

The relationship $d_s(x_k) = SQ_s\mathscr{O}(\hat{d}_s(x))$ allows us to rescale the expectation:

$$\mathbb{E}_\mathcal{D}[d_s(x)] \leq \hat{\mathbb{E}}_\mathcal{T}[d_s(x)] + 3SQ_s\sqrt{\frac{\ln(2/\varphi)}{2N}} + \frac{4S}{N}\sqrt{\sum_{k=1}^{N}\left(\|Z_1^{(x_i)}\|^2 + \|Z_2^{(x_i)}\|^2\right)}.$$

(3.31)

Finally, the boundedness of $|A^{(x_i)}\gamma| \leq 2S$ (from the triangle inequality) implies $\hat{\mathbb{E}}_\mathcal{T}[d_s(x)] \leq 2S$. Substituting this into the inequality (3.31) completes the proof. $\qquad\square$

Theorem 2. (Generalization Error Bound for GrMv-BLS) Let $S > 0$ be a norm constraint, $\varphi \in (0,1)$ a parameter, and $\mathcal{T} = \{(x_k, y_k)\}_{k=1}^{N}$ a training set of IID samples from distribution $\mathcal{B}$, where $y_k \in \{-1,+1\}$ and $x_k = (x_k^A; \eta x_k^B)$. Define $A^{(x)} = \left(Z_1^{(x)}; Z_2^{(x)}\right)$, where $Z_1^{(x)} = [F^A(x) \quad E^A(x)]$ and $Z_2^{(x)} = [F^B(x) \quad E^B(x)]$, with $F^A(x)[F^B(x)]$ feature layer for data point x corresponding to view-A(B), and $E^A(x)[E^B(x)]$ enhancement layer for data point x corresponding to view-A(B). The function class $\mathcal{H}_S = \{h \mid h(x) = |A^{(x)}\gamma|, \|\gamma\| \leq S\}$, where $\gamma = (\gamma_1; \gamma_2)$. Then, every $h \in \mathcal{H}$ with probability at least $1 - \varphi$ over $\mathcal{T}$ satisfies:

$$\mathbb{P}_\mathcal{B}[yh(x) \leq 0] \leq \frac{1}{N(1+\eta)}\sum_{k=1}^{N}(\zeta_{1_i} + \eta\zeta_{2_i}) + 3\sqrt{\frac{\ln(2/\varphi)}{2N}} + \quad (3.32)$$

$$\frac{4S}{N(1+\eta)}\sqrt{\sum_{k=1}^{N}\left(\|Z_1^{(x_i)}\|^2 + \eta^2\|Z_2^{(x_i)}\|^2\right)}.$$

Proof. Define the piecewise loss $\mathscr{O} : \mathbb{R} \to [0,1]$ as:

$$\mathscr{O}(x) = \begin{cases} 1, & x < 0, \\ 1 - \frac{x}{1+\eta}, & 0 \leq x \leq 1 + \eta, \\ 0, & x > 1 + \eta. \end{cases}$$

This ensures $\mathbb{P}_\mathcal{B}[yh(x) \leq 0] \leq \mathbb{E}_\mathcal{B}[\mathscr{O}(yh(x))]$.

By Lemma 1 and the Lipschitz property of $\mathscr{O}$ (with constant $\frac{1}{1+\eta}$), we derive:

$$\mathbb{E}_\mathcal{D}[\mathscr{O}(yh(x))] \leq \hat{\mathbb{E}}_T[\mathscr{O}(yh(x))] + 3\sqrt{\frac{\ln(2/\varphi)}{2n}} + \frac{2}{1+\eta}\hat{R}_n(\mathcal{H}). \quad (3.33)$$

We can find the bound for the empirical loss $\hat{\mathbb{E}}_T[\mathscr{O}(yg(x))]$ using equation (3.24) and (3.25):

$$\bar{\mathbb{E}}_{\mathcal{T}}[\mathcal{O}(\hat{h}(x,y))] \leq \frac{1}{N(1+\eta)} \sum_{k=1}^{N} [1 + \eta - y_k h(x_k)]_+$$

$$= \frac{1}{N(1+\eta)} \sum_{k=1}^{n} [1 - y_k h_A(x_k^A) + \eta(1 - y_k h_B(x_k^B))]_+$$

$$\leq \frac{1}{N(1+\eta)} \sum_{k=1}^{N} \left\{ [1 - y_k h_A(x_k^A)]_+ + \eta[1 - y_k h_B(x_k^B))]_+ \right\}$$

$$\leq \frac{1}{N(1+\eta)} \sum_{k=1}^{N} [y_k(\zeta_{1_k} + \eta\zeta_{2_k})]_+$$

$$\leq \frac{1}{N(1+\eta)} \sum_{k=1}^{N} (\zeta_{1_k} + \eta\zeta_{2_k}). \tag{3.34}$$

The empirical Rademacher complexity $\hat{R}_n(\mathcal{H})$ is bounded via Lemma 2:

$$\hat{R}_N(\mathcal{H}) \leq \frac{2S}{N} \sqrt{\sum_{k=1}^{N} \left(\|Z_1^{(x_k)}\|^2 + \eta^2 \|Z_2^{(x_k)}\|^2 \right)}. \tag{3.35}$$

Combining the results (3.33), (3.34), and (3.35) yields the theorem. □

By leveraging Rademacher complexity, these theorems establish rigorous bounds on the expected error of the model, thereby demonstrating the robust generalization capabilities of GrMv-BLS. Theorem 1 establishes a bound on the difference in outputs across multi-view data, while Theorem 2 provides a bound on the misclassification probability. As N increases, GrMv-BLS provides a tight consistency and generalization error bound for classification. These theoretical guarantees enhance the practical effectiveness of GrMv-BLS while maintaining the geometric and structural relationships of the data.

3.3 Experimental Results and Discussion

To examine the efficacy of the proposed GrMv-BLS model, we compare it to the baseline models SVM2K, MVLDM, ELM1 (ELM for 'view-A'), ELM2 (ELM for 'view-B'), MvTSVM, BLS1 (BLS for 'view-A'), BLS2 (BLS for 'view-B'), IF-BLS1 (IF-BLS for 'view-A'), and IF-BLS2 (IF-BLS for 'view-B') on publicly available AwA [20], UCI [21] and KEEL [22] datasets.

3.3.1 Experimental Setup

A personal computer (processor: intel(R) xeon(R) gold 6226R CPU) with Windows-11 and Python-3.11 with 128 GB of RAM is used to execute all

the experiments. In our experiment, the dataset is divided randomly, such that 30% of the samples are allocated for testing and 70% for training. A five-fold cross-validation approach is used to fine-tune and validate the hyperparameters. The graph regularization parameters θ_j (for $j = 1, 2$), ρ coupling parameter, and the regularization parameters c_i (for $i = 1, 2, 3$) are all tuned within the range $\{10^{-5}, 10^{-4}, \ldots, 10^5\}$. In our experiments, we assume $\theta_1 = \theta_2$ and $c_1 = c_2 = c_3 = \rho$. In addition, N_1, N_2, N_3 and N_4 represent the number of feature groups, the number of nodes in each feature group, the number of enhancement nodes in the enhancement group, and the number of enhancement groups, respectively. The hyperparameters $N_1 = N_3$ are selected from the range $\{5, 10, 15, \ldots, 50\}$, while N_2 is chosen from $\{1, 3, 5, \ldots, 21\}$, and N_4 is set to 1 (following [23]).

3.3.2 Results and Statistical Analyses on AwA Dataset

AwA is a repository that includes 30,475 photos of 50 distinct animal species. Each image has six representations of pre-extracted features. We use 10 distinct test classes selected from the AwA datasets for our experiment. Among the 6180 photos in these categories are leopards, chimpanzees, Persian cats, humpback whales, hippopotamuses, raccoons, giant pandas, rats, pigs, and seals. Using the one-against-one approach, we construct 21 datasets for the possible pairing of classes in the animal. Table 3.1 provides the optimal hyperparameter values and accuracies for each baseline model and the proposed GrMv-BLS model. The average accuracy for the existing models MVLDM, SVM2K, ELM1, ELM2, MvTSVM, BLS1, BLS2, IF-BLS1 and IF-BLS2 are 73.02%, 76.73%, 70.14%, 74.54%, 63.03%, 70.14%, 76.52%, 70.21% and, 76.74%, respectively. The proposed GrMv-BLS model surpasses all baseline models, achieving an average accuracy of 76.95%. The outcomes show that the proposed GrMv-BLS model beats the existing models.

Average accuracy may pose a misleading measure of performance, as a model's outstanding results on one dataset could offset its inadequate results on another. Consequently, depending exclusively on average accuracy may mask the model's actual effectiveness across various datasets, and thus due to the necessity for a more comprehensive evaluation of the proposed GrMv-BLS framework that accounts for performance differences in varying datasets, we perform a rank test. In the rank test, greater ranks are assigned to the worst-performing model, and lesser ranks are provided to the well-performing model. Assume $\mathcal{D}$ models are being evaluated using $\mathcal{K}$ datasets, and the dth

model's rank on the kth dataset is denoted as $R(d, k)$. The dth model's average rank can be calculated as: $R_d = (\sum_{i=1}^{\mathcal{K}} R(d, k))/\mathcal{K}$. We calculate the average rank of each model in order to compare our proposed model with baseline models. SVM2K (3.90), MVLDM (5.23), MvTSVM (8.25), BLS1 (6.98), BLS2 (3.78), ELM1 (6.78), ELM2 (5.50), IF-BLS1 (7.35), and IF-BLS2 (3.68) are the average ranks for the baseline models. On the other hand, our proposed GrMv-BLS model outperformed all of the baseline models with least average rank of 3.58. Further, we perform a number of statistical test on our proposed GrMv-BLS model to determine the statistical significance of the outcomes of our experiment.

Friedman Test

To evaluate the proposed GrMv-BLS model's statistical significance, the Friedman test is employed. By examining the average ranks of the models under comparison, this test aims to identify significant differences between them. Assume the null hypothesis, which states that models possess the same average rank, showcasing that there is no difference in the model's performance. A chi-squared distribution, represented as $\mathcal{X}_F^2$ with degrees of freedom $(\mathcal{D} - 1)$ is computed via: $\mathcal{X}_F^2 = \frac{12\mathcal{K}}{\mathcal{D}(\mathcal{D}+1)}[\sum_{d=1}^{\mathcal{D}} R_d^2 - \frac{\mathcal{D}(\mathcal{D}+1)^2}{4}]$, where R_d is the average rank of the model d, $\mathcal{K}$ and $\mathcal{D}$ are the number of datasets and the number of models, respectively. The Friedman statistic is defined as: $F_F = \frac{(\mathcal{K}-1)\mathcal{X}_F^2}{\mathcal{K}(\mathcal{D}-1)-\mathcal{X}_F^2}$, with degree of freedoms $(\mathcal{D} - 1)$ and $(\mathcal{D} - 1) \times (\mathcal{K} - 1)$. We have $\mathcal{D} = 10$ and $\mathcal{K} = 21$, the computed values of $\mathcal{X}_F^2$ and F_F are 34.17 and 4.14, respectively. According to the Friedman distribution table at $\alpha = 0.05$ significance level, the critical value of the 2-tailed Friedman statistic is $F_F(9, 180) = 1.93$. The calculated value of F_F is 4.14 which goes above the Friedman statistic $F_F(9, 180) = 1.93$. Hence, we reject the null hypothesis. This result highlights a significant statistical difference between the models being compared.

Nemenyi Post Hoc Test

In addition, to analyze the models, we use the Nemenyi post hoc test. Under this test, the model d_1 is considered to be significantly superior to the model d_2 if the variation between the average rank of d_1 and d_2 model is at least equal to the CD, which is calculated as: $CD = q_\alpha\sqrt{\frac{\mathcal{D}(\mathcal{D}+1)}{6\mathcal{K}}}$. Here, q_α represents the critical value for the 2-tailed post hoc test. In our experiment, we have $q_\alpha = 3.164$, $\mathcal{D} = 10$ and $\mathcal{K} = 21$, after calculating we obtain $CD = 2.95$.

The average rank differences of the proposed GrMv-BLS model from ELM1, MvTSVM, BLS1, and IF-BLS1 are 3.2, 4.67, 3.4, and 3.77, respectively. These variations are greater than CD, which implies that, according to the post hoc test, the GrMv-BLS model is superior to ELM1, MvTSVM, BLS1, and IF-BLS1. However, the GrMv-BLS model could not establish the required difference from other baseline models, based on the post hoc test result. Nevertheless, the proposed GrMv-BLS model is superior to all existing models, as demonstrated by their lower average ranking.

Win-Tie-Loss Sign Test

Furthermore, a statistical Win-Tie-Loss sign test is performed. Assume the null hypothesis, under which two models perform similarly. Each model is expected to win in approximately half of the total datasets, represented by $\frac{\mathcal{K}}{2}$, where $\mathcal{K}$ is the number of datasets. If the model outperform in approximately $\frac{\mathcal{K}}{2} + 1.96\frac{\sqrt{\mathcal{K}}}{2}$ datasets, then the model is considered to be significantly superior. The ties are divided equally in event when there are an even number of ties between the models. While, when there are an odd number of equal performances, one equal performance is disregarded, and the rest of the equal performances are distributed among the classifiers. In our case, $\mathcal{K} = 21$ which implies we require 14.99 wins in datasets to derive a meaningful distinction among the models.

In the Table 4.4, [a,b,c] represents [Number of wins, Number of ties, Number of loss]. After examining Table 4.4, results are evident that the proposed GrMv-BLS surpasses the existing MVLDM, ELM1, ELM2, BLS1, and MvTSVM models, by attaining more than 14.99 wins. The proposed GrMv-BLS establishes a significant difference to the baseline models, except for BLS2, IF-BLS2, and SVM2K. Nevertheless, the accuracy of the proposed GrMv-BLS model shows its superiority over the SVM2K, BLS2, and IF-BLS2 models.

3.3.3 Results and Statistical Analyses on UCI and KEEL Datasets

University of california irvine (UCI) and knowledge extraction based on evolutionary learning (KEEL) are two different datasets repositories. In this subsection, we present results and a detailed analysis of the comparison between the proposed GrMv-BLS and baseline models on 21 KEEL and UCI datasets. Since UCI and KEEL datasets do not intrinsically have multiview properties,

our experiment utilizes 95% of the principal components to minimize features of the input data, which is designated as 'view-B', while the original data is referred to as 'view-A'. Table 3.3 presents the accuracy and optimal hyperparameter value for each model corresponding to each dataset from the UCI and KEEL repositories. From Table 3.3, the average accuracy for the baseline models SVM2K, MVLDM, MvTSVM, BLS1, BLS2, ELM1, and ELM2 are 81.54%, 82.62%, 76.17%, 82.22%, 82.19%, 81.97%, and 81.62%, respectively. The proposed GrMv-BLS model achieves an average accuracy of 83.84%, outperforming all baseline models. Additionally, it is noteworthy that the proposed GrMv-BLS model has the lowest average rank among all models. The average ranks for the baseline models are as follows: SVM2K(4.14), MVLDM (4.26), MvTSVM (6.71), BLS1 (3.90), BLS2 (4.52), ELM1 (4.86), ELM2 (4.07) and our proposed GrMv-BLS model achieved an average rank of 3.52, which shows the superiority of our proposed GrMv-BLS model in terms of rank.

To evaluate the outcomes of our experiment's statistical significance, we employ multiple statistical tests to evaluate the importance of our proposed GrMv-BLS models. The Friedman test, a non-parametric statistical method is employed. For significance level of 5%, we compute $\mathcal{X}_F^2 = 22.84$, $F_F = 3.67$ and $F_F(7, 140) = 2.07$. Since $F_F(7, 140)$ is lesser than calculated F_F, the null hypothesis is rejected. Now, the Nemenyi post hoc test is performed at a significance level of 5%, we calculate $CD = 2.29$. The average rank disparity between the proposed GrMv-BLS model with MvTSVM is 3.19. According to the post hoc test, the proposed GrMv-BLS shows a major difference from the baseline model MvTSVM. Although, the post hoc test could not establish the difference between the GrMv-BLS model and the baseline model, except for MvTSVM. However, the proposed GrMv-BLS model surpasses all the baseline models in terms of average ranks. Subsequently, based on the win-tie-loss sign test with $\mathcal{K} = 21$, the model must win at least 14.99 datasets to achieve statistical significance. Our proposed GrMv-BLS model demonstrates better performance against the baseline MvTSVM and ELM1 models by achieving 18 and 15 wins, respectively. Win-tie-loss test could not demonstrate the significant difference between the proposed GrMv-BLS and existing models, except for MvTSVM and ELM1. However, we demonstrate the superiority of the proposed GrMv-BLS model, through a greater winning percentage against the baseline models.

3.3.4 Sensitivity Analyses of Hyperparameters c_1 and N_1

(a) credit_approval
(from UCI and KEEL repository)

(b) echocardiogram
(from UCI and KEEL repository)

(c) Giantpanda vs Hippopotamous
(from AwA repository)

(d) Giantpanda vs Leopard
(from AwA repository)
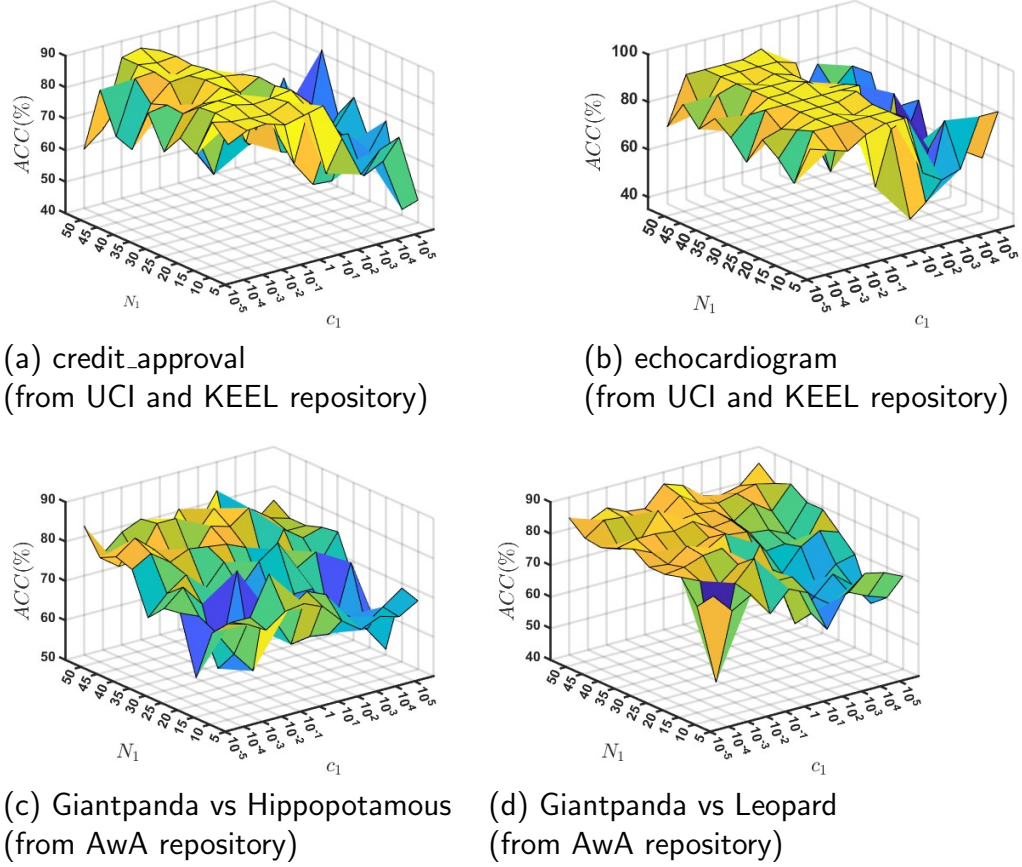
Figure 3.1: Effect of hyperparameters c_1 and the number of feature group N_1 on the accuracy of the proposed GrMv-BLS framework.

In this subsection, we investigate the sensitivity of the hyperparameters by adjusting the values of c_1 and N_1, in order to analyze the effect of hyperparameters on the proposed GrMv-BLS model's generalization capability. Figure 3.1 illustrates the model's accuracy changes as the hyperparameters are adjusted. It is observable that the accuracy of the credit_approval and echocardiogram dataset increases with the escalation of N_1. It is noteworthy that Figure 3.1a and 3.1b achieve optimized values for accuracy in the range of $40 - 50$ (for N_1) and $10^{-2} - 10^1$ (for c_1). Similarly, Figures 3.1c and 3.1d can be observed. Consequently, the observations emphasize the importance of carefully selecting hyperparameter values to achieve optimal model performance.

3.3.5 Sensitivity Analyses of Coupling Hyperparameter ρ

(a) credit_approval (from UCI and KEEL repository)

(b) breast_cancer_wisc_diag (from UCI and KEEL repository)

(c) Giantpanda vs Seal (from AwA repository)
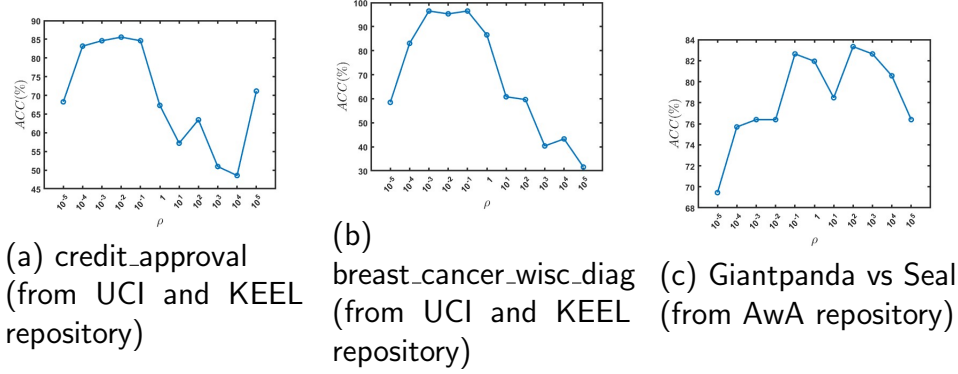
Figure 3.2: Effect of coupling hyperparameter ρ on the performance of the proposed GrMv-BLS model.

The optimization problem (3.3) includes a coupling term $\zeta_1^t \zeta_2$ with ρ acting as the coupling parameter. By adjusting ρ inside the range specified in the experimental setup and fixing other parameters at their optimal values, we examined the effect of ρ on the performance of our model. Figure 3.2 illustrates the performance of the proposed GrMv-BLS model on the datasets: credit_approval, breast_cancer_wisc_diag and Giantpanda vs Seal, achieving peak accuracy at different values of ρ. In the breast_cancer_wisc_diag dataset, the optimal accuracy increases with the increase in the value of ρ, after achieving a specific threshold, the accuracy stabilizes and then decreases with increasing value of ρ. Similarly, one can observe the Figures 3.2a and 3.2b. From the above analyses, we can observe that the performance of the proposed GrMv-BLS model affects with the different values of ρ. Our result highlights the influence of dataset characteristics on the accuracy of the proposed GrMv-BLS model, emphasizing the necessity of fine tuning the coupling parameter ρ, to achieve optimal performance.

3.3.6 Sensitivity Analyses of Hyperparameters N_1 and N_2

In this subsection, we investigate the influence of the hyperparameters N_1 and N_2 on the accuracy. Figure 3.3 depicts the variations in model accuracy as the hyperparameters are adjusted. In Figure 3.3a and 3.3b, the optimal accuracy is observed within the N_1 range of 25 - 50 and the N_2 range of
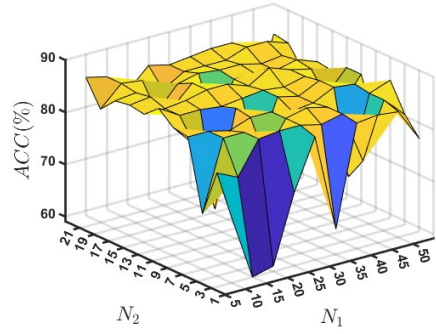
(a) credit_approval
(from UCI and KEEL repository)
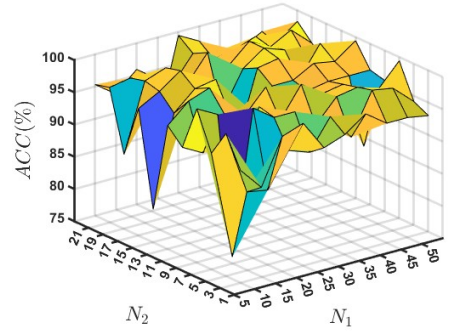
(b) breast_cancer_wisc_diag
(from UCI and KEEL repository)
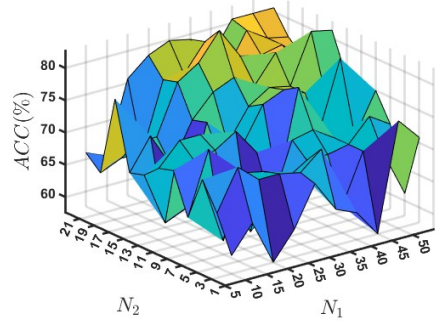
(c) Giantpanda vs Hippopotamus
(from AwA repository)
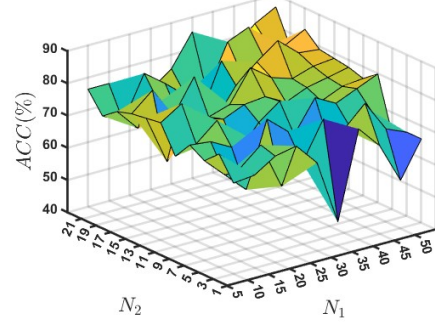
(d) Giant panda vs Seal
(from AwA repository)

Figure 3.3: Effect of the number of feature group N_1 and the number of nodes in feature group N_2 on the accuracy of the proposed GrMv-BLS model.

11 - 21. Additionally, the findings presented in Figure 3.3c shows that the optimal accuracy of the proposed GrMv-BLS model is observed at $N_1 = 35$ and $N_2 = 17$, whereas in Figure 3.3d, the optimal accuracy is observed at $N_1 = 50$ and $N_2 = 21$. Overall, these findings highlight the necessity of meticulously setting the values for the hyperparameters N_1 and N_2 to attain the optimal performance of the proposed GrMv-BLS model.

Table 3.1: Performance comparison of baseline and GrMv-BLS models on AwA datasets.

Dataset ↓ Model →	SVM2K [24] (C_1)	MVLDM [25] (C_1, C_2, C_3, σ)	MvTSVM [26] (C_1, C_2, D)	ELM1 [27] (C_1, N)	ELM2 [27] (C_1, N)	BLS1 [1] (C_1, N_1, N_2, N_3)	BLS2 [1] (C_1, N_1, N_2, N_3)	IF-BLS1 [23] (C_1, N_1, N_2, N_3, μ)	IF-BLS2 [23] (C_1, N_1, N_2, N_3, μ)	GrMv-BLS † (C_1, ρ, N)
Chimpanzee vs Leopard	80.11 (0.0001)	63.89 (100000,0.0001,0.001,4)	83.33 (0.00001,0.00001,0.00001)	68.75 (0.00001,163)	46.53 (0.00001,163)	75.00 (0.01,15,13,15)	81.94 (0.01,50,19,50)	68.75 (0.01,20,9,20,32)	72.92 (100000,15,21,15,0.5)	77.08 (0.0001,0.0001,35)
ChimpanzeevsPersiancat	70.86 (0.00001)	86.11 (1000,0.0001,1,4)	50 (0.0001,0.00001,0.00001)	79.86 (0.00001,143)	69.44 (69.44,0.00001,183)	68.75 (0.1,40,7,40)	80.56 (0.01,45,21,45)	68.75 (0.01,20,9,20,32)	72.92 (100000,15,21,15,0.5)	77.08 (0.01,0.1,50)
Chimpanzee vs Rat	77.08 (0.0001)	57.64 (100000,0.0001,1,4)	75 (0.0001,0.00001,0.00001)	68.06 (100000,43)	52.78 (0.00001,183)	63.89 (0.00001,10,17,10)	78.47 (0.1,35,13,35)	68.06 (0.01,15,9,15,2)	76.39 (1000,50,17,50,32)	74.31 (1.0,0.00001,45)
GiantpandavsHippopotamus	77.78 (0.001)	74.31 (100000,0.0001,0.00001,4)	54.17 (0.0001,0.00001,0.00001)	74.31 (0.0001,103)	81.25 (81.25,0.00001,203)	74.31 (0.01,50,13,50)	83.33 (0.01,45,21,45)	68.06 (0.001,15,9,15,2)	76.39 (1000,50,17,50,32)	79.86 (1.0,0.00001,50)
GiantpandavsLeopard	80.19 (0.01)	61.81 (100000,0.00001,0.01,0.25)	54.17 (0.0001,0.00001,0.00001)	61.11 (1,43)	78.47 (78.47,0.00001,203)	65.97 (0.01,50,13,50)	83.33 (0.01,35,19,35)	70.83 (0.1,20,3,20,16)	80.56 (1000,50,19,50,0.625)	81.25 (0.01,0.01,50)
GiantpandavsRaccoon	80.19 (0.0001)	64.58 (1000,0.00001,0.01,0.25)	52.78 (0.0001,0.00001,0.00001)	68.06 (0.0001,203)	74.31 (0.00001,163)	68.06 (0.01,35,19,35)	78.47 (0.01,35,19,35)	58.33 (100000,35,9,35,2)	83.33 (100000,50,19,50,0.625)	75.69 (0.01,0.01,50)
GiantpandavsSeal	85.89 (10000)	86.81 (10000,0.00001,0.00001,2)	56.94 (0.0001,0.00001,0.00001)	80.56 (0.0001,203)	77.08 (77.08,0.00001,183)	78.47 (0.001,35,13,35)	81.25 (0.001,30,19,30)	71.53 (0.00001,50,1,50,2)	76.39 (100000,20,21,20,0.25)	87.5 (0.001,0.1,50)
HippopotamusvsRat	75.33 (0.1)	64.58 (100000,0.001,0.01,2)	75.83 (0.0001,0.00001,0.00001)	65.28 (0.0001,143)	71.53 (71.53,0.00001,143)	65.97 (0.001,20,21,20)	68.75 (0.01,25,25,25)	77.78 (0.00001,10,21,10,0.625)	82.64 (100000,45,17,45,1)	70.14 (1.0,0.0001,50)
HumpbackwhalevsRat	82.28 (10)	77.78 (10000,0.0001,0.001,4)	80.31 (0.0001,0.00001,0.00001)	81.94 (0.0001,143)	86.11 (86.11,0.00001,183)	79.17 (0.01,50,15,50)	86.81 (0.01,40,17,40)	69.44 (0.1,20,21,20,32)	74.31 (100000,50,21,50,0.125)	82.64 (0.001,0.01,50)
HumpbackwhalevsSeal	76.39 (0.0001)	78.47 (10000,0.0001,0.00001,0.25)	72.08 (0.0001,0.00001,0.00001)	77.78 (10000,43)	73.61 (73.61,0.00001,143)	78.47 (0.1,45,11,45)	77.78 (0.01,40,17,40)	71.53 (100000,40,13,40,2)	75.08 (0.00001,30,21,30,1)	76.39 (10,0.0001,35)
LeopardvsHippopotamus	78.17 (10000)	75 (10000,0.0001,0.0001,4)	50.69 (0.0001,0.00001,0.00001)	73.61 (0.0001,123)	77.08 (77.08,0.00001,143)	63.89 (0.1,5,17,5)	77.08 (0.01,50,21,50)	68.75 (0.1,20,3,20,16)	79.86 (10,25,19,25,32)	79.17 (10,1,50)
LeopardvsHumpbackwhale	90.75 (0.001)	89.58 (100,0.00001,0.01,4)	79.31 (0.0001,0.00001,0.00001)	89.58 (0.0001,183)	91.67 (91.67,0.00001,163)	90.28 (0.1,5,19,5)	95.83 (0.01,50,15,50)	87.50 (0.1,20,3,20,16)	90.97 (100000,50,15,50,0.5)	95.14 (10,0.1,50)
LeopardvsPersiancat	82.19 (0.01)	80.56 (100000,0.001,0.00001,0.25)	79.31 (0.0001,0.00001,0.00001)	70.83 (1,63)	84.72 (84.72,0.00001,203)	74.31 (0.1,25,9,25)	86.11 (0.00001,50,21,50)	70.83 (1,10,9,10,2)	86.11 (1000,50,9,50,0.125)	86.11 (0.00001,0.0001,0.001,50)
LeopardvsRaccoon	80.56 (0.00001)	56.94 (0.01,0.00001,0.1,4)	55 (0.0001,0.00001,0.00001)	59.03 (0.001,183)	57.64 (57.64,100000,103)	61.11 (0.01,5,15,5)	76.39 (0.01,50,17,50)	75.72 (0.001,45,1,45,0.5)	78.47 (10,50,21,50,32)	71.53 (10,0.001,50)
LeopardvsSeal	80.42 (0.001)	81.25 (1000,0.00001,0.01,0.25)	63.47 (0.0001,0.00001,0.00001)	75.69 (0.0001,183)	79.86 (79.86,0.00001,203)	72.22 (0.001,20,5,20)	85.42 (0.01,40,19,40)	72.92 (0.1,30,9,30,32)	86.11 (0.001,40,3,40,2)	81.94 (0.0001,0.00001,40)
PersiancatvsHippopotamus	76.81 (0.0001)	75.69 (100000,0.001,0.0001,0.25)	76.53 (0.0001,0.00001,0.00001)	75.69 (0.00001,203)	79.86 (79.86,0.00001,203)	70.14 (0.1,10,7,10)	81.94 (0.01,30,19,30)	69.44 (100,25,3,25,1)	81.94 (10,40,17,40,32)	81.25 (0.001,0.1,45)
PersiancatvsRat	60.44 (0.01)	56.94 (10000,0.0001,0.0001,4)	64.17 (0.0001,0.00001,0.00001)	54.86 (0.00001,143)	56.25 (56.25,0.00001,143)	58.33 (0.001,40,15,40)	65.28 (0.01,50,19,50)	59.72 (100,25,13,25)	64.58 (10,40,17,40,32)	65.97 (0.001,0.1,45)
PigvsHippopotamus	71.53 (100)	72.22 (100000,0.00001,0.1,4)	65.83 (0.0001,0.00001,0.00001)	70.14 (1000,43)	65.97 (65.97,0.00001,183)	69.44 (0.01,40,11,40)	69.44 (0.01,40,19,40)	70.83 (0.001,5,19,5,1)	67.36 (100,10,21,10,2)	70.14 (0.0001,0.1,15)
PigvsHumpbackwhale	80.19 (0.01)	88.89 (0.001,0.01,0.00001,4)	77.92 (0.0001,0.00001,0.00001)	82.64 (0.1,63)	89.58 (89.58,0.00001,183)	82.64 (0.01,40,13,40)	70.83 (0.01,45,19,45)	84.03 (0.1,20,15,20,32)	86.81 (10,45,21,45,32)	84.03 (1.0,0.0001,35)
PigvsRat	71.53 (0.01)	64.58 (0.001,0.01,0.00001,4)	68.61 (0.0001,0.00001,0.00001)	62.5 (0.1,63)	59.72 (59.72,0.00001,203)	64.58 (0.01,40,17,40)	50 (0.001,25,17,25)	57.64 (1,30,9,30,32)	61.11 (1.25,17,25,0.0625)	61.11 (0.0001,0.01,35)
PigvsSeal	72.69 (0.01)	72.92 (100000,0.0001,0.001,0.25)	65.56 (0.0001,0.00001,0.00001)	70.14 (0.0001,143)	68.06 (68.06,0.00001,203)	70.83 (0.001,50,13,50)	65.97 (0.01,25,13,25)	67.36 (0.01,20,13,20,32)	69.44 (100000,50,13,50,2)	72.92 (0.1,1,10)
RaccoonvsRat	62.22 (100)	65.28 (100000,0.00001,10,4)	61.89 (0.00001,0.00001,0.00001)	59.72 (100,43)	70.83 (70.83,0.00001,163)	56.25 (0.01,15,11,15)	68.75 (0.00001,25,17,25)	59.03 (10000,5,11,5,0.25)	73.61 (0.00001,30,17,30,3)	67.36 (0.01,0.00001,45)
Average ACC	76.73	73.02	63.03	70.14	74.54	70.14	76.52	70.21	76.74	76.95

33

Table 3.2: Win-Tie-Loss test on AwA datasets.

	SVM2K [24]	MvTSVM [26]	ELM1 [27]	ELM2 [27]	BLS1 [1]	BLS2 [1]	MVLDM [25]	IF-BLS1 [23]	IF-BLS2 [23]
MvTSVM [26]	[2, 0, 19]								
ELM1 [27]	[3, 0, 18]	[15, 0, 6]							
ELM2 [27]	[7, 0, 14]	[18, 0, 3]	[14, 0, 7]						
BLS1 [1]	[2, 0, 19]	[15, 0, 6]	[11, 0, 10]	[7, 0, 14]					
BLS2 [1]	[12, 0, 9]	[18, 0, 3]	[16, 0, 5]	[15, 0, 6]	[16, 1, 4]				
MVLDM [25]	[8, 0, 13]	[17, 0, 4]	[15, 3, 3]	[8, 0, 13]	[12, 0, 9]	[7, 0, 14]			
IF-BLS1 [23]	[2,0,19]	[14,0,7]	[9,0,12]	[5,0,16]	[11,0,10]	[5,0,16]	[4,1,16]		
IF-BLS2 [23]	[11,0,10]	[19,0,2]	[16,0,5]	[16,0,5]	[16,0,5]	[8,1,12]	[14,0,7]	[19,0,2]	
GrMv-BLS†	[13, 0, 8]	[19, 0, 2]	[17, 0, 4]	[15, 0, 6]	[19, 0, 2]	[8, 1, 12]	[15, 0, 6]	[18,0,3]	[10,0,11]

† represents the proposed model.

Table 3.3: Performance comparison among the existing models and proposed GrMv-BLS model on KEEL and UCI datasets.

Dataset ↓ Model →	SVM2K [24] (C_1)	MVLDM [25] (C_1, C_2, C_3, σ)	MvTSVM [26] (C_1, C_2, D)	BLS1 [1] (C_1, N_1, N_2, N_3)	BLS2 [1] (C_1, N_1, N_2, N_3)	ELM1 [27] (C_1, N)	ELM2 [27] (C_1, N)	GrMv-BLS † (C_1, ρ, N)
breast_cancer	62.45 (1)	70 (0.1,0.001,0.1,0.25)	55.58 (0.00001, 0.00001, 0.00001)	68.6 (1,15,13,15)	69.77 (10,5,9,5)	69.77 (0.01,43)	65.12 (1,83)	**76.75** (0.01,0.00001,35)
breast_cancer_wisc_diag	95.49 (100)	93.15 (0.1,1,0.1,4)	88.6 (0.00001, 0.00001, 0.00001)	93.57 (0.0001,45,1,45)	91.81 (0.0001,20,9,20)	92.42 (0.001,163)	**97.08** (0.01,163)	**97.08** (0.1,1,5)
checkerboard_Data	87.02 (1)	84.06 (1,0.1,0.01,2)	43.75 (0.00001, 0.00001, 0.00001)	**89.91** (0.01,15,17,15)	88.46 (0.01,50,7,50)	86.98 (0.001,163)	86.06 (0.01,123)	88.46 (1,0.001,5)
congressional_voting	61.54 (0.1)	**62.31** (0.1,0.001,0.1,0.25)	55.41 (0.00001, 0.00001, 0.00001)	61.83 (100000,5,3,5)	60.31 (100000,5,21,5)	61.06 (0.001,63)	61.06 (10,3)	61.83 (0.001,0.01,5)
credit_approval	85.99 (0.1)	**86.47** (1,0.01,1,4)	76.4 (0.01, 0.0001, 1)	84.13 (0.01,20,11,20)	84.13 (1,5,15,5)	84.61 (0.001,83)	84.61 (1000,63)	85.58 (0.001,0.00001,45)
echocardiogram	79.48 (0.1)	89.74 (0.01,0.1,1,0.25)	76.08 (10, 0.00001, 0.00001)	80 (10,40,19,40)	85 (0.00001,35,21,35)	82.5 (0.01,103)	**87.5** (0.001,203)	80 (0.00001,0.01,25)
haberman_survival	74.72 (100)	71.42 (1,0.001,0.1,0.25)	71.16 (0.00001, 0.0001, 0.01)	78.26 (0.0001,10,9,10)	**79.35** (0.0001,45,1,45)	76.08 (0.001,103)	78.26 (0.001,123)	79.35 (1,0.00001,5)
heart_hungarian	82.95 (1)	**84.09** (0.1,0.01,1000,0.25)	78.64 (0.01, 0.0001, 0.00001)	82.02 (0.001,25,1,25)	78.65 (1,25,13,25)	77.52 (0.01,83)	77.53 (0.001,123)	78.65 (1,0.1,5)
hepatitis	80.85 (1)	78.26 (0.1, 1, 0.01, 4)	78.72 (0.03125)	80.85 (0.0001,15,13,15)	78.72 (0.001,10,17,10)	78.85 (0.01,183)	80.85 (1,23)	**80.86** (0.00001,1,20)
ionosphere	**91.42** (1)	88.57 (0.1,0.01,0.1,0.25)	95.52 (0.1, 0.01, 0.001)	85.85 (0.001,25,19,25)	86.79 (0.001,10,17,10)	85.84 (0.01,163)	86.79 (0.1,183)	85.85 (0.1,0.0001,25)
new-thyroid1	78.46 (1000)	95.13 (0.1,0.1,0.1,0.25)	82.31 (0.00001, 0.00001, 0.00001)	95.38 (0.01,30,3,30)	95.38 (1,5,9,5)	**98.46** (10, 103)	96.92 (1, 43)	95.38 (0.1,0.0001,5)
pima	**76.19** (1)	69.13 (1,0.0001,0.1,4)	73.33 (0.0001, 0.00001, 0.00001)	74.46 (1,35,19,35)	74.46 (0.1,15,9,15)	74.89 (0.1,63)	74.03 (0.01,143)	72.29 (0.1,0.0001,15)
ripley	89.07 (1000)	89.07 (10,0.0001,0.1,0.5)	80.67 (0.0001,0.0001,0.0001)	88.8 (10,20,13,20)	88.8 (10,15,17,15)	86.13 (0.1,63)	87.53 (100, 203)	**90.14** (0.1,0.0001,45)
shuttle-6_vs_2-3	97.1 (0.1)	93.98 (0.1,1,0.0001,0.25)	96 (0.00001, 0.1, 1)	**100** (10,40,15,40)	**100** (10,15,7,15)	**100** (0.00001,23)	**100** (0.00001,23)	**100** (0.00001,0.00001,45)
spect	73.41 (0.1)	**75.94** (0.1,0.1,0.01,4)	61.29 (0.00001, 0.1, 1)	68.75 (10,40,15,40)	68.75 (10,15,7,15)	55 (10000,23)	63.75 (0.01,103)	65 (0.01,10,5)
statlog_australian_credit	69.56 (1)	71.98 (0.1,0.1,0.0001,0.25)	67.08 (0.00001, 0.00001, 0.00001)	53.85 (0.01,25,3,25)	68.75 (0.0001,35,7,35)	76.34 (10000,23)	70.67 (0.01,103)	**84.75** (0.0001,100000,25)
statlog_heart	75.72 (1)	84.56 (0.00001,1,1,0.25)	73.54 (100, 0.1, 0.0001)	83.95 (0.001,5,5,5)	**90.12** (0.0001,45,7,45)	82.71 (0.00001,63)	88.89 (0.001,163)	81.4 (1,0.0001,5)
tic_tac_toe	**100** (10000)	98.94 (0.1,10,0.1,0.25)	98.51 (0.0001, 0.00001, 1)	99.65 (0.1,50,15,50)	81.6 (1,50,1,50)	99.65 (10,143)	70.48 (0.001,83)	99.65 (0.0001,0.00001,40)
transfusion	76.33 (10)	75 (100,0.1,1,0.5)	**77.09** (0.01, 0.00001, 0.01)	76.89 (1,20,17,20)	76 (10,5,13,5)	75.11 (0.001,103)	76.89 (0.0001,3)	76 (0.1,0.01,5)
votes	93.08 (10)	90.58 (0.01,1,0.1,0.25)	93.77 (0.01, 0.0001, 0.00001)	97.71 (100,20,7,20)	96.95 (10,25,19,25)	95.41 (0.1,183)	**98.47** (100000,23)	97.71 (0.0001,10,10)
Average ACC	81.54	82.62	76.17	82.22	82.19	81.97	81.62	**83.84**

† represents the proposed model. The accuracy of the best-performing model is shown in boldface.

Table 3.4: Win-Tie-Loss test on KEEL and UCI datasets.

	SVM2K [24]	MVLDM [25]	MvTSVM [26]	BLS1 [1]	BLS2 [1]	ELM1 [27]	ELM2 [27]
MVLDM [25]	[10, 1, 10]						
MvTSVM [26]	[4, 0, 17]	[6, 0, 15]					
BLS1 [1]	[11, 0, 10]	[10, 0, 11]	[18, 0, 3]				
BLS2 [1]	[8, 0, 13]	[9, 0, 12]	[18, 0, 3]	[6, 6, 9]			
ELM1 [27]	[8, 0, 13]	[10, 0, 11]	[17, 0, 4]	[7, 1, 13]	[10, 1, 10]		
ELM2 [27]	[11, 1, 9]	[10, 0, 11]	[17, 0, 4]	[10, 1, 10]	[10, 1, 10]	[12, 3, 6]	
GrMv-BLS †	[13, 0, 8]	[13, 0, 8]	[18, 0, 3]	[6, 8, 7]	[10, 6, 5]	[15, 1, 5]	[11, 1, 9]

† represents the proposed model.

Chapter 4

Class Probability based Bell-Shaped Broad Learning System and its Multi-View Extension

This chapter introduces two innovative models: the class probability based bell-shaped broad learning system (CPBS-BLS) and its multi-view extension (CPBS-MvBLS), designed to address critical limitations in traditional broad learning systems (BLS) and multi-view BLS (MvBLS) frameworks. While BLS and MvBLS excel in computational efficiency and multi-perspective data analysis, their performance degrades significantly in the presence of outliers, label noise, and class imbalance, common challenges in real-world datasets. To mitigate these issues, the proposed models integrate a bell-shaped membership function with class probability metrics and imbalance ratio weighting, creating an adaptive sample weighting mechanism. The CPBS-BLS framework dynamically assigns weights to training samples based on their proximity to class centroids and local neighborhood density. Samples near class centers in high-density regions receive higher weights, reinforcing their contribution to model training, while potential outliers-distant from centroids or situated in sparse regions-are systematically downweighted. This approach enhances the robustness against data imperfections. Extending this paradigm to multi-view learning, CPBS-MvBLS incorporates view-specific gbell membership functions across diverse feature representations, enabling the model to leverage complementary information from multiple data perspectives while maintaining outlier resilience. Theoretical analysis provides generalization bounds for both

models, ensuring their reliability. Extensive experiments on UCI, KEEL, and AwA datasets demonstrate that CPBS-BLS and CPBS-MvBLS outperform baseline methods in term of accuracy and robustness, validated by statistical tests. Our work advances BLS frameworks by addressing critical challenges in real-world data, offering a scalable solution for applications requiring noise-resistant learning.

4.1   Notations

Assume that the training set for singular view is represented as $\{(a_1, b_1), (a_2, b_2), ..., (a_M, b_M)\}$ and the training set for multi view is expressed as $\{(a_1^1, a_2^1, b_1), (a_1^2, a_2^2, b_2), ..., (a_1^M, a_2^M, b_M)\}$, where a_r stands for the rth sample corresponding to singular view. a_1^r and a_2^r represent the feature vector for view-1 and view-2, respectively. b_r is the label vector for r^{th} sample. Suppose $A \in \mathbb{R}^{M \times L}$ is the input matrix for singular view, $A_1 \in \mathbb{R}^{M \times l_1}$ and $A_2 \in \mathbb{R}^{M \times l_2}$ are the input matrices for view-1 and view-2, respectively, and $B \in \mathbb{R}^{M \times 2}$ is the output matrix. l_1 and l_2 are the number of features in view-1 and view-2, respectively. L is the number of features of a single-view input sample. **Note:** these notations are used only for this chapter.

4.2   Proposed Work

This section presents our two proposed models: CPBS-BLS and CPBS-MvBLS. These models integrate the CPBS membership function with the broad learning system (BLS) and multi-view broad learning system (MvBLS) to effectively assign weights to data points. This integration enables the models to manage naturally impure data by mitigating the effects of noise, outliers, and class imbalance. Consequently, the proposed models enhance robustness and generalization, making them well-suited for real-world datasets with inherent imperfections.

4.2.1   Class Probability based Bell Shaped Broad Learning System (CPBS-BLS)

In this subsection, we present the mathematical formulation of the proposed CPBS-BLS model, which amalgamates the CPBS membership function with BLS to robustly mitigate the impact of noise and outliers. To determine the

output weights, we construct an optimization problem incorporating an L_2 regularization term, effectively reducing the model's structural risk. The optimization problem minimizes a membership-weighted error term, ensuring that the contribution of each input sample is scaled proportionally to its assigned membership value. This formulation not only reduces the reconstruction error but also improves the overall stability and predictive performance of the BLS framework. The optimization problem of the proposed CPBS-BLS model is formulated as follows:

$$\arg\min_{\Omega} \frac{1}{2}\|\Omega\|_2^2 + \frac{c}{2}\|S\xi\|_2^2$$

$$\text{s.t.} \quad G\Omega - B = \xi. \tag{4.1}$$

In the optimization problem (4.1), S represents the score matrix computed from the equation (2.6). ξ is the error and G is the concatenated layer obtained from the concatenation of feature and enhancement layers. G denotes the concatenation of the feature and enhancement layers, defined as: $G = [Z, H] \in \mathbb{R}^{M \times (m_Z n_Z + m_H n_H)}$ The Lagrangian of the optimization problem (4.1) is as follows:

$$\mathcal{L}_B = \frac{1}{2}\|\Omega\|_2^2 + \frac{c}{2}\|S\xi\|_2^2 - \alpha^t(G\Omega - B - \xi). \tag{4.2}$$

Partial differentiation of the Lagrangian (4.2) is:

$$\frac{\partial \mathcal{L}_B}{\partial \Omega} = \Omega - G^T\alpha = 0, \tag{4.3}$$

$$\frac{\partial \mathcal{L}_B}{\partial \xi} = cS^T S\xi + \alpha = 0, \tag{4.4}$$

$$\frac{\partial \mathcal{L}_B}{\partial \alpha} = G\Omega - B - \xi = 0. \tag{4.5}$$

Finding the value of α from (4.4) and (4.5), then put into equation (4.3), we obtain:

$$\Omega + G^t(cS^t S(G\Omega - B)) = 0, \tag{4.6}$$

$$\implies \Omega + cG^t S^t SG\Omega = cG^t S^t SB, \tag{4.7}$$

$$\implies \Omega(I + cG^t S^t SG) = cG^t S^t SB, \tag{4.8}$$

$$\implies \Omega = \left(\frac{I}{c} + G^t S^t SG\right)^{-1} G^t S^t SB, \tag{4.9}$$

where I is the identity matrix of appropriate dimension. From equation (4.3) and (4.5), we obtain:

$$GG^t\alpha - B - \Omega = 0. \tag{4.10}$$

Furthermore, from equation (4.4) and (4.10), we obtain:

$$cS^tS(GG^t\alpha - B) + \alpha = 0, \tag{4.11}$$

$$\implies \alpha(I + cS^tSGG^t) = cS^tSB, \tag{4.12}$$

$$\implies \alpha = \left(\frac{I}{c} + S^tSGG^t\right)^{-1} S^tSB. \tag{4.13}$$

Finally, from eqaution (4.3) and (4.13), we get:

$$\Omega = G^t \left(\frac{I}{c} + S^tSGG^t\right)^{-1} S^tSB. \tag{4.14}$$

We derive two distinct expressions for computing Ω, given by equations (4.9) and (4.14). Both formulations involve matrix inversion, but their applicability depends on the relationship between the feature dimensions $(m_Z n_Z + m_H n_H)$ and the number of samples (M). Specifically, if the feature dimension $(m_Z n_Z + m_H n_H)$ is less than or equal to the sample size (M), we compute Ω using Equation (4.9). Otherwise, Ω will be determined using the equation (4.14). This approach provides a computational advantage by allowing matrix inversion to be performed in either the feature space or the sample space, depending on the problem's dimensionality. Thus, the optimal solution to (4.1) can be given by:

$$\Omega = \begin{cases} \left(G^tS^tSG + \frac{I}{c}\right)^{-1} G^tS^tSB, & \text{if} \quad (m_Z n_Z + m_H n_H) \leq M, \\ G^t \left(S^tSGG^t + \frac{I}{c}\right)^{-1} S^tSB, & \text{if} \quad M < (m_Z n_Z + m_H n_H). \end{cases} \tag{4.15}$$

Algorithm 2 Proposed CPBS-BLS Model 4.2.1

Input: The training datasets X_A for view-1 and X_B for view-2, the parameters c_1, c_2, c_3, θ_1 θ_2, and ρ.

 1: **Compute:** Concatenated matrix $G = [Z, H]$.
 2: **Compute:** The gbell membership value $\mathcal{M}_{gb}(\cdot)$ 2.3 for each training data points.
 3: **Compute:** The class probability $\mathcal{M}_p(\cdot)$ 2.4 of training data.
 4: **Compute:** The CPBS membership value $\mathcal{M}(\cdot)$ using subsection 2.5
 5: **Compute:** The score matrix S (2.6) for CPBS-BLS model.
 6: **Output** : Use equation (4.15) to obtain the output weight matrix Ω for the CPBS-BLS model.

4.2.2 Class Probability based Bell Shaped Multi View Broad Learning System (CPBS-MvBLS)

In this subsection, we present the mathematical derivation of the proposed CPBS-MvBLS, which integrates the CPBS membership function into a multi-

view BLS. This integration enhances the model's ability to effectively capture underlying data patterns while robustly handling noise and outliers. The proposed optimization problem for CPBS-MvBLS incorporates L_2 regularization terms for both views to mitigate overfitting. To minimize prediction error, the formulation includes membership-weighted error terms for each view, ensuring improved model performance by adaptively prioritizing reliable samples. Additionally, a membership-weighted coupling term is introduced to jointly minimize errors across both views while preserving the relative contribution of each data sample. By leveraging these components, the CPBS-MvBLS achieves enhanced data representation through meticulous noise-resilient learning, improved robustness against outliers, and superior predictive performance compared to conventional approaches. The optimization problem of the proposed CPBS-MvBLS can be formulated as below:

$$\underset{\Omega_1, \Omega_2}{\arg\min} \frac{c_1}{2}\|S_1\xi_1\|_2^2 + \frac{c_2}{2}\|S_2\xi_2\|_2^2 + \frac{c_3}{2}\|\Omega_1\|_2^2 + \frac{1}{2}\|\Omega_2\|_2^2 + \rho(S_1\xi_1)^t(S_2\xi_2)$$

$$\text{s.t.} \quad G_1\Omega_1 - B = \xi_1 \quad \text{and} \quad G_2\Omega_2 - B = \xi_2, \tag{4.16}$$

where $G_1 = [Z_1, H_1]$, $G_2 = [Z_2, H_2]$, S_1, and S_2 denote the score matrices corresponding to view-1 and view-2, respectively. The weight matrices Ω_1 and Ω_2 connect the concatenated layers G_1 and G_2 to the output layer B. To enhance the model's performance, a coupling term $(S_1\xi_1)^t(S_2\xi_2)$ is incorporated into (4.16). This term facilitates the preservation of an optimal membership distribution for input samples while simultaneously minimizing errors across both views. Additionally, the optimization problem incorporates regularization hyperparameters c_1, c_2, and c_3, along with a coupling hyperparameter ρ. The Lagrangian of the optimization problem (4.16) can be computed as follows:

$$\mathcal{L}_{MB} = \frac{c_1}{2}\|S_1\xi_1\|_2^2 + \frac{c_2}{2}\|S_2\xi_2\|_2^2 + \frac{c_3}{2}\|\Omega_1\|_2^2 + \frac{1}{2}\|\Omega_2\|_2^2 + \rho(S_1\xi_1)^t(S_2\xi_2)$$
$$- \alpha_1^t(G_1\Omega_1 - B - \xi_1) - \alpha_2^t(G_2\Omega_2 - B - \xi_2). \tag{4.17}$$

Partial Differentiation of (4.17) is:

$$\frac{\partial \mathcal{L}_{MB}}{\partial \xi_1} = c_1 S_1^t S_1 \xi_1 + \rho S_1^t S_2 \xi_2 + \alpha_1 = 0, \tag{4.18}$$

$$\frac{\partial \mathcal{L}_{MB}}{\partial \xi_2} = c_2 S_2^t S_2 \xi_2 + \rho S_2^t S_1 \xi_1 + \alpha_2 = 0, \tag{4.19}$$

$$\frac{\partial \mathcal{L}_{MB}}{\partial \Omega_1} = c_3 \Omega_1 - G_1^t \alpha_1 = 0, \tag{4.20}$$

$$\frac{\partial \mathcal{L}_{MB}}{\partial \Omega_2} = \Omega_2 - G_2^t \alpha_2 = 0, \tag{4.21}$$

$$\frac{\partial \mathcal{L}_{MB}}{\partial \alpha_1} = G_1 \Omega_1 - B - \xi_1 = 0, \tag{4.22}$$

$$\frac{\partial \mathcal{L}_{MB}}{\partial \alpha_2} = G_2 \Omega_2 - B - \xi_2 = 0. \tag{4.23}$$

By solving for α_1 and α_2 from (4.18) and (4.19), respectively, and substituting them into (4.20) and (4.21), we derive the analytical expressions:

$$c_3 \Omega_1 + G_1^t(c_1 S_1^t S_1 \xi_1 + \rho S_1^t S_2 \xi_2) = 0, \tag{4.24}$$

$$\Omega_2 + G_2^t(c_2 S_2^t S_2 \xi_2 + \rho S_2^t S_1 \xi_1) = 0. \tag{4.25}$$

The variables ξ_1 and ξ_2 are determined from (4.21) and (4.22), respectively, and subsequently substituted into (4.24) and (4.25) to obtain:

$$c_3 \Omega_1 + G_1^t(c_1 S_1^t S_1(G_1 \Omega_1 - B) + \rho S_1^t S_2(G_2 \Omega_2 - B)) = 0, \tag{4.26}$$

$$\Omega_2 + G_2^t(c_2 S_2^t S_2(G_2 \Omega_2 - B) + \rho S_2^t S_1(G_1 \Omega_1 - B)) = 0. \tag{4.27}$$

The above equations (4.26) and (4.27) can be written as follows:

$$c_3 \Omega_1 + c_1 G_1^t S_1^t S_1 G_1 \Omega_1 + \rho G_1^t S_1^t S_2 G_2 \Omega_2 = G_1^t(c_1 S_1^t S_1 + \rho S_1^t S_2)B, \tag{4.28}$$

$$\Omega_2 + c_2 G_2^t S_2^t S_2 G_2 \Omega_2 + \rho G_2^t S_2^t S_1 G_1 \Omega_1 = G_2^t(c_2 S_2^t S_2 + \rho S_2^t S_1)B. \tag{4.29}$$

The equations (4.28) and (4.29) imply the following equations:

$$\Omega_1(c_3 I_1 + c_1 G_1^t S_1^t S_1 G_1) + \Omega_2(\rho G_1^t S_1^t S_2 G_2) = G_1^t(c_1 S_1^t S_1 + \rho S_1^t S_2)B, \tag{4.30}$$

$$\Omega_1(\rho G_2^t S_2^t S_1 G_1) + \Omega_2(I_2 + c_2 G_2^t S_2^t S_2 G_2) = G_2^t(c_2 S_2^t S_2 + \rho S_2^t S_1)B, \tag{4.31}$$

where I_1 and I_2 are the identity matrices of appropriate dimensions. (4.30) and (4.31) can be shown as:

$$\begin{bmatrix} c_3 I_1 + c_1 G_1^t S_1^t S_1 G_1 & \rho G_1^t S_1^t S_2 G_2 \\ \rho G_2^t S_2^t S_1 G_1 & I_2 + c_2 G_2^t S_2^t S_2 G_2 \end{bmatrix} \begin{bmatrix} \Omega_1 \\ \Omega_2 \end{bmatrix} = \begin{bmatrix} G_1^t(c_1 S_1^t S_1 + \rho S_1^t S_2) \\ G_2^t(c_2 S_2^t S_2 + \rho S_2^t S_1) \end{bmatrix} B. \tag{4.32}$$

Finally, we get the expression for output weight matrices Ω_1 and Ω_2 in the matrix form:

$$\begin{bmatrix} \Omega_1 \\ \Omega_2 \end{bmatrix} = \begin{bmatrix} c_3 I_1 + c_1 G_1^t S_1^t S_1 G_1 & \rho G_1^t S_1^t S_2 G_2 \\ \rho G_2^t S_2^t S_1 G_1 & I_2 + c_2 G_2^t S_2^t S_2 G_2 \end{bmatrix}^{-1} \begin{bmatrix} G_1^t(c_1 S_1^t S_1 + \rho S_1^t S_2) \\ G_2^t(c_2 S_2^t S_2 + \rho S_2^t S_1) \end{bmatrix} B. \tag{4.33}$$

Algorithm 3 Proposed CPBS-MvBLS Model [4.2.1]

Input: The training dataset A_1 for view-1 and A_2 for view-2, the parameters c_1, c_2, c_3, ρ, and membership function parameter τ, and ω.

1: **Compute:** Concatenated matrices $G_1 = [Z_1, H_1]$ and $G_2 = [Z_2, H_2]$.
2: **Compute:** The gbell membership value $\mathcal{M}_{gb}(\cdot)$ [2.3] for each training data point across both the views.
3: **Compute:** The class probability $\mathcal{M}_p(\cdot)$ [2.4] of training data across both view.
4: **Compute:** The CPBS membership value $\mathcal{M}(\cdot)$ using subsection [2.5]
5: **Compute:** The score matrix S_1 for view-1 and S_2 for view-2 for CPBS-BLS model.
6: **Output:** Use equation ([4.33]) to obtain the output weight matrix Ω_1 for and Ω_2 the CPBS-BLS model.

4.2.3 Time Complexity Analysis

We analyze the computational complexity of the proposed CPBS-BLS and CPBS-MvBLS models by decomposing the analysis into two components: (1) membership value computation and (2) core BLS operations.

CPBS-BLS Complexity. Given an input dataset of size M, the membership value calculation for each sample exhibits a time complexity of $\mathcal{O}(M)$. However, the dominant computational cost arises from the BLS component. Specifically: if $M \geq (m_z n_z + m_H n_H)$, the BLS complexity is $\mathcal{O}\left((m_z n_z + m_H n_H)^3\right)$ and if $M < (m_z n_z + m_H n_H)$, the complexity scales as $\mathcal{O}(M^3)$. Since $\mathcal{O}(M)$ is negligible compared to the BLS complexity in both regimes, the overall time complexity of CPBS-BLS reduces to: $\mathcal{O}\left((m_z n_z + m_H n_H)^3\right)$ for $M \geq m_z n_z + m_H n_H$, and $\mathcal{O}(M^3)$ otherwise.

CPBS-MvBLS Complexity. Analogously, membership value computation requires $\mathcal{O}(M)$ time. The main complexity arises from the matrix inversion in the optimization solution (Eq. ([4.33])), resulting in $\mathcal{O}\left((m_Z n_Z + m_H n_H)^3\right)$. As $\mathcal{O}(M)$ is dominated by this term, the total complexity simplifies to: $\mathcal{O}\left((m_Z n_Z + m_H n_H)^3\right)$.

4.2.4 Theoretical Analysis of the Proposed Models

Guaranteeing the prediction accuracy of a learning algorithm on a theoretical basis is essential and indispensable to establishing the reliability of the algorithm. In this section, we analyze the generalization bound through the probabilities of the extreme bounds from the predicted error expressed by norms.

Definition 4.1. Let R_G define the *Stable rank* of G then $R_G = \frac{\|G\|_F^2}{\|G\|_2^2}$, where $\|\cdot\|_F$ is Frobenius norm of a matrix and $\|\cdot\|_2$ is L_2 norm of a matrix.

Definition 4.2. Let $V = (v_{ij})$ be an $n \times k$ matrix $(k \leq n)$ where each entry v_{ij} is an IID random variable following the standard normal distribution. The *Gaussian estimator* for the Frobenius norm of a matrix G is denoted by $\mathscr{E}(G)$ and defined as:

$$\mathscr{E}(G) = \frac{1}{\sqrt{k}}\|GV\|_F.$$

Lemma 4.1. For any $\mu > 1$, the following probabilistic bounds hold for the Gaussian estimator $\mathscr{E}(G)$:

$$P\left\{\mathscr{E}(G) \geq \mu\|G\|_F\right\} \leq \exp\left[-\frac{k}{2}R_G\left\{(1-\Gamma)(\mu^2-1) - \frac{(1-\Gamma)^2 R_G}{2\Gamma}\right\}\right],$$
$$(4.34)$$

$$P\left\{\mathscr{E}(G) \leq \frac{1}{\mu}\|G\|_F\right\} \leq \exp\left\{-\frac{k}{4}\frac{(\mu^2-1)^2}{\mu^4}\right\}, \qquad (4.35)$$

where $\Gamma = \sqrt{\dfrac{R_G}{2(\mu^2-1)+R_G}}$. Consequently, we have the combined probability bound:

$$P\left\{\frac{1}{\mu}\|G\|_F < \mathscr{E}(G) < \mu\|G\|_F\right\} \geq$$
$$1 - \exp\left[-\frac{k}{2}R_G\left\{(1-\Gamma)(\mu^2-1) - \frac{(1-\Gamma)^2 R_G}{2\Gamma}\right\}\right] - \exp\left\{-\frac{k}{4}\frac{(\mu^2-1)^2}{\mu^4}\right\}.$$
$$(4.36)$$

Let Ω be an unknown vector satisfying $B = G\Omega$. Consider B^{out} is the prediction of B, which imply there is an error between B and B^{out}. Assume $\mathbf{E}$ be the error vector. WLOG, we may assume that elements of $\mathbf{E}$ are IID with mean 0 and variance σ^2, i.e. $e \sim \mathcal{N}(0, \sigma^2 I)$, where I is the identity matrix. Then the B can be represented as: $B = G\Omega^{out} + \mathbf{E}$. Let's employ the L_2 norm for the prediction error, $\|B - B^{out}\|$.

Generalization Error Bound for the CPBS-BLS

Theorem 4.1.1. For any $\mu > 1$, the upper bound and the lower bound for the deviation $\|B - B^{out}\|_2$ are $P\left\{\|B - B^{out}\|_2 \geq \mu \cdot \frac{\sigma}{R_{G+\mathbf{E}}}\|G\|_2\|G^\dagger\|_F\right\} \leq \exp\left[-\frac{1}{2}R_{G^\dagger}\left\{(1-\Gamma^+)(\mu^2-1) - \frac{(1-\Gamma^+)^2 R_{G^\dagger}}{2\Gamma^+}\right\}\right]$, where $\Gamma^+ = \sqrt{\dfrac{R_{G^\dagger}}{2(\mu^2-1)+R_{G^\dagger}}}$ and $P\left\{\|B - B^{out}\|_2 \leq \frac{1}{\mu} \cdot \frac{\sigma}{R_{G+\mathbf{E}}}\|G\|_2\|G^\dagger\|_F\right\} \leq \exp\left\{-\frac{1}{4}\frac{(\mu^2-1)^2}{\mu^4}\right\}$, respectively. Then, by merging the lower and upper bounds, the probability of the generalization

bound for the predicted error $\|B - B^{out}\|$is:

$$P\left\{\frac{1}{\mu} \cdot \frac{\sigma}{R_{G+\mathbf{E}}}\|G\|_2\|G^\dagger\|_F < \|B - B^{\text{out}}\|_2 < \mu \cdot \frac{\sigma}{R_{G+\mathbf{E}}}\|G\|_2\|G^\dagger\|_F\right\} \geq$$

$$1 - \exp\left[-\frac{1}{2}R_{G^\dagger}\left\{(1-\Gamma^+)(\mu^2-1) - \frac{(1-\Gamma^+)^2 R_{G^\dagger}}{2\Gamma^+}\right\}\right] - \exp\left\{-\frac{1}{4}\frac{(\mu^2-1)^2}{\mu^4}\right\}.$$

$$(4.37)$$

Proof. We begin by establishing two key inequalities:

$$P\left\{\|\Omega - \Omega^{\text{out}}\|_2 \geq \mu\frac{\sigma}{R_{G^\dagger+\mathbf{E}}}\|G^\dagger\|_F\right\} \leq$$

$$\exp\left[-\frac{1}{2}R_{G^\dagger}\left((1-\Gamma^\dagger)(\mu^2-1) - \frac{(1-\Gamma^\dagger)^2 R_{G^\dagger}}{2\Gamma^\dagger}\right)\right], \qquad (4.38)$$

$$P\left\{\|\Omega - \Omega^{\text{out}}\|_2 \leq \frac{1}{\mu}\frac{\sigma}{R_{G^\dagger+\mathbf{E}}}\|G^\dagger\|_F\right\} \leq \exp\left\{-\frac{1}{4}\frac{(\mu^2-1)^2}{\mu^4}\right\}. \qquad (4.39)$$

Applying Lemma 1 with $k = 1$ and normalizing $\mathbf{E} = \mathbf{E}/\sigma \sim \mathcal{N}(0, I)$, we obtain:

$$P\{\mathscr{E}(G^\dagger) \geq \mu\|G^\dagger\|_F\}$$

$$= P\{\|G^\dagger\mathbf{E}'\|_F \geq \mu\|G^\dagger\|_F\} \qquad \{\because \mathscr{E}(G^\dagger) = \|G^\dagger\mathbf{E}'\|_F\}$$

$$\leq \exp\left[-\frac{1}{2}R_{G^\dagger}\left\{(1-\Gamma^\dagger)(\mu^2-1) - \frac{(1-\Gamma^+)^2 R_{G^\dagger}}{2\Gamma^+}\right\}\right],$$

which can be rewritten as:

$$P\{\|G^\dagger\mathbf{E}'\|_F \geq \mu\|G^\dagger\|_F\}$$

$$= P\left\{\|G^\dagger\mathbf{E}\|_2 \geq \mu\frac{\sigma}{R_{G^\dagger+\mathbf{E}}}\|G^\dagger\|_F\right\},$$

$$= P\left\{\|\Omega - \Omega^{\text{out}}\|_2 \geq \mu\frac{\sigma}{R_{G^\dagger+\mathbf{E}}}\|H^+\|_F\right\}.$$

This establishes the inequality:

$$P\left\{\|\Omega - \Omega^{\text{out}}\|_2 \geq \mu\frac{\sigma}{R_{G^\dagger+\mathbf{E}}}\|H^+\|_F\right\} \leq$$

$$\exp\left[-\frac{1}{2}R_{G^\dagger}\left\{(1-\Gamma^\dagger)(\mu^2-1) - \frac{(1-\Gamma^+)^2 R_{G^\dagger}}{2\Gamma^+}\right\}\right]. \qquad (4.40)$$

Similarly, for the lower bound (using (4.35) from Lemma 1):

$$P\left\{\|G^\dagger\mathbf{E}'\|_F \leq \frac{1}{\mu}\|G^+\|_F\right\} \leq \exp\left\{-\frac{1}{4}\frac{(\mu^2-1)^2}{\mu^4}\right\}.$$

Also, we have:

$$P\{\|G^{\dagger}\mathbf{E}\|_F \leq \frac{1}{\mu}\|H^+\|_F\}$$

$$=P\left\{\|G^{\dagger}\mathbf{E}\|_2 \geq \frac{1}{\mu}\frac{\sigma}{R_{G^{\dagger}+\mathbf{E}}}\|G^{\dagger}\|_F\right\},$$

$$=P\left\{\|\Omega - \Omega^{\text{out}}\|_2 \leq \frac{1}{\mu}\frac{\sigma}{R_{G^{\dagger}+\mathbf{E}}}\|G^{\dagger}\|_F\right\}.$$

Hence, we yield:

$$P\left\{\|\Omega - \Omega^{\text{out}}\|_2 \leq \frac{1}{\mu}\frac{\sigma}{R_{G^{\dagger}+\mathbf{E}}}\|G^{\dagger}\|_F\right\} \leq \exp\left\{-\frac{1}{4}\frac{(\mu^2-1)^2}{\mu^4}\right\} \quad (4.41)$$

Since $\|B - B^{out}\|_2 = \|G\|_2\|\Omega - \Omega^{out}\|_2$ (easy to prove by Cauchy–Schwarz inequality). The final result can be obtained directly by combining equations (4.40) and (4.41). □

Generalization Error Bound for the CPBS-MvBLS

The generalization bound for the proposed CPBS-MvBLS can be derived by adapting Theorem 4.1.1 through the substitution of G with the concatenated weight matrix $\begin{bmatrix} \Omega_1 \\ \Omega_2 \end{bmatrix}$, as defined in Subsection 4.2.2. This substitution preserves the theoretical framework while accounting for the multi-view structure of the model, ensuring that the bound remains valid under the same probabilistic guarantees.

4.3 Implementation and Evaluation

Implementation Details. In this section, we conduct a comprehensive evaluation of the proposed CPBS-BLS and CPBS-MvBLS models, benchmarking the performance of CPBS-BLS against BLS [28], extreme learning model (ELM) [27], intuitionistic fuzzy-BLS (IF-BLS) [29], neuro fuzzy BLS (NF-BLS) [30], fuzzy BLS (F-BLS) [29], intuitionistic fuzzy twin support vector machine (IF-TSVM) [31], and hierarchical ELM (H-ELM) [32] baseline models, whereas the proposed CPBS-MvBLS model compared against ELM1 (ELM on view-1) [27], ELM2 (ELM on view-2) [27], BLS1 (BLS on view-1) [28], BLS2 (BLS on view-2) [28], two view support vector machine (SVM2K) [24], multiview large margin distribution machine (MVLDM) [25], and multi-view twin support vector machine (MvTSVM)[26] baseline models. The experiments are carried out using a diverse array of datasets sourced from the UCI and KEEL

repositories, ensuring a robust assessment across multiple domains. Furthermore, we specifically evaluate the CPBS-MvBLS model using the AwA dataset to further validate its efficacy. For the implementation of these experiments, we utilize a computing environment configured with Windows 11 and Python version 3.11, running on an Intel® Xeon® Gold 6226® CPU, equipped with 128 GB of RAM. This setup provides the necessary computational resources to effectively handle the data processing and model training requirements of our study. To ensure a fair and rigorous comparison, all the hyperparameters are fine-tuned using 5-fold cross-validation combined with an exhaustive grid search strategy. The regularization parameters c, c_1, c_2, c_3, and the coupling parameter ρ in the optimization problem (4.16) are selected from the range $\{10^{-5}, 10^{-4}, \ldots, 10^5\}$ and set equal, while the number of feature groups m_Z is sampled from $\{1, 3, \ldots, 21\}$. The enhancement groups are fixed at $m_H = 1$ for simplicity, and the number of nodes per feature group (n_Z) and the number of nodes per enhancement group (n_H) are varied within the range $\{5, 10, \ldots, 50\}$ and set $n_Z = n_H$. The kernel parameter μ is chosen from the exponential set $\{2^{-5}, 2^{-4}, \ldots, 2^5\}$, and the steepness ($\omega$) and shape ($\tau$) parameters are jointly optimized from $\left\{\frac{1}{2}, \frac{5}{8}, \frac{3}{4}, \frac{7}{8}, 1\right\}$. The dataset is split into 70 % of the training set and 30% of the testing set. This systematic parameter selection framework ensures that the models are evaluated under optimal experimental setup, facilitating a reliable comparison with baseline models.

4.3.1 Evaluation on CPBS-BLS model with UCI and KEEL Repository

To evaluate the efficacy and efficiency of the proposed CPBS-BLS model, we conducted comprehensive experiments on 29 benchmark datasets from the university of california, irvine (UCI) repository. Performance assessment was based on two key metrics: classification accuracy and average rank, with detailed comparative results presented in Table 4.1. The baseline BLS, ELM, IF-BLS, NF-BLS, F-BLS, IF-TSVM, and H-ELM models achieved average accuracies of 75.87%, 76.64%, 77.02%, 77.78%, 76.74%, 68.22%, and 76.55%, respectively. In contrast, the proposed CPBS-BLS model demonstrated superior performance by attaining average accuracy of 81.09%. This significant improvement highlights the effectiveness of our proposed CPBS-BLS model. To further validate the results, we employed a rank-based evaluation scheme, where lower ranks indicate better performance. Given $\mathscr{T}$ models and $\mathscr{D}$

Table 4.1: Performance comparison of baseline and proposed CPBS-BLS model on UCI and KEEL datasets.

Dataset ↓ Model →	ELM [27]	BLS [1]	IF-BLS[29]	NF-BLS [30]	F-BLS [29]	IF-TSVM [31]	H-ELM [32]	CPBS-BLS †
aus	87.98	81.99	87.44	78.43	75.38	**89.9**	74.36	**89.9**
bank	89.54	87.91	89.41	89.58	88.76	89.5	88.63	**90.64**
blood	**77.33**	76.89	75.11	76.91	75.7	76.24	77.17	76.89
breast_cancer	69.77	68.6	67.79	70.18	72.29	70.18	**84.27**	74.42
breast_cancer_wisc	**98.1**	95.71	88.99	90.71	88.28	66.68	85.42	97.62
breast_cancer_wisc_diag	**99.42**	93.57	98.25	89.28	83.67	63.99	89.81	89.47
breast_cancer_wisc_prog	68.33	78.33	79.87	79.86	78.31	76.35	**79.9**	73.33
chess_krvkp	95.62	93.74	84.92	70.4	84.29	25.1	82.1	**98.69**
congressional_voting	61.07	61.83	59.93	**63.91**	60.23	61.38	62.76	61.83
cylinder_bands	**76.62**	70.13	72.85	69.53	69.13	60.87	63.24	66.23
echocardiogram	82.5	80	**90**	80.94	84.67	80.09	84.67	87.5
haberman_survival	76.09	78.26	**79.35**	73.49	69.66	73.49	74.15	79.35
heart_hungarian	77.53	64	79.78	70.36	78.94	82.02	**93.56**	77.53
hepatitis	85.11	80.85	78.6	**87.74**	84.52	79.35	82.58	80.85
ilpd_indian_liver	68.57	70.29	72.38	72.38	69.48	71.35	**72.39**	70.86
ionosphere	85.85	85.85	89.19	**91.46**	88.35	64.43	82.65	85.85
mammographic	80.28	78.89	79.82	79.5	78.67	82.01	72.32	**83.04**
monks_1	77.84	68.86	39.51	**86.13**	77.14	40.64	66.72	76.65
monks_2	64.64	43.09	67.22	**86.34**	68.98	65.73	66.23	73.2
musk_1	82.52	81.82	78.98	67.85	76.68	22.11	**89.49**	79.02
pima	58.61	74.46	76.96	72.79	65.1	71.86	70.31	**78.17**
pittsburg_bridges_T_OR_D	70.33	83.87	**90.24**	90.14	88.19	86.14	87.29	87.1
planning	41.82	70.24	74.23	72.49	71.41	71.38	71.38	**76.36**
shuttle-6_vs_2-3	45	**100**	44.93	56.38	54.34	**100**	43.98	**100**
spect	55	66.79	67.5	67.92	65	58.49	68.75	69.43
spectf	**83.95**	54.32	79.73	80.85	78.97	79.34	79.34	80.25
statlog_australian_credit	**70.67**	53.85	66.25	69.28	67.83	70.67	67.83	70.19
statlog_heart	82.72	83.95	85.19	**88.74**	82.22	55.56	80.74	86.42
tic_tac_toe	**99.65**	**99.65**	**99.65**	82.76	97.81	65.31	75.73	**99.65**
Average ACC	75.87	76.64	77.02	77.78	76.74	68.22	76.55	**81.09**

† represents the proposed model.

datasets, the average rank for model t is computed as: $\mathscr{R}_t = \frac{1}{\mathscr{D}}\sum_{i=1}^{\mathscr{D}}\mathscr{R}(t,i)$. The baseline models yielded average ranks of 5.17, 4.25, 4.14, 3.64, 5.25, 5.83, and 4.60 for BLS, ELM, IF-BLS, NF-BLS, F-BLS, IF-TSVM, and H-ELM, respectively. Our proposed CPBS-BLS model achieved significantly better rank of 3.08, further validating their competitive superiority of the proposed CPBS-BLS model. This rank-based analysis confirms that our models consistently perform in the top tier across diverse datasets.

To quantify the statistical significance of these results, we conducted three statistical tests: the Friedman test, the Nemenyi post hoc test, and the Win-tie-loss test. The Friedman test is a non-parametric statistical method for comparing multiple models across different datasets. The chi-sqaured value for Friedman test statistic is calculated as: $\chi_r^2 = \frac{12\mathscr{D}}{\mathscr{T}(\mathscr{T}+1)}\sum_{t=1}^{\mathscr{T}}\mathscr{R}_t^2 - \frac{\mathscr{T}(\mathscr{T}+1)^2}{4}$ and the value for Friedman static is given by the formula $F_F = \frac{(\mathscr{D}-1)\chi_r^2}{\mathscr{D}(\mathscr{T}-1)-\chi_r^2}$, where $\mathscr{D}$ is the number of datasets, $\mathscr{T}$ is the number of models, and $\mathscr{R}_t$ is the average rank for model t. Under the null hypothesis that all models perform equally, the test statistic follows a χ_r^2 distribution with $\mathscr{T} - 1$ and $(\mathscr{T} - 1)(\mathscr{D} - 1)$ degrees of freedom (dof) at $\alpha = 0.05$. We reject the null hypothesis if the computed F_F exceeds the critical value of Friedman's statistic (F_F), concluding that there are statistically significant differences among the

models' performances. In the case of our experiment, we have $\mathscr{T} = 8$ and $\mathscr{D} = 29$, thus we computed that $\chi_r^2 = 80.41$, Friedman static is $F_F = 18.367$, and the critical value of F_F for dof 7 and 196 is 2.06. Since $2.06 < 18.367$, we reject the null hypothesis and establish a significant statistical difference between the baseline and the proposed CPBS-BLS model. Following a significant Friedman test result, the Nemenyi post-hoc test is employed to identify specific pairwise differences between models. The test computes a critical difference CD $= q_\alpha \sqrt{\frac{\mathscr{T}(\mathscr{T}+1)}{6\mathscr{D}}}$ that determines whether the performance difference between any two models is statistically significant. If the absolute difference between the average ranks of any two models is greater than CD, we reject the null hypothesis of equal performance at significance level α, concluding that the models exhibit statistically significant differences in their overall performance across the evaluated datasets. Given $\mathscr{T} = 8$ and $\mathscr{D} = 29$, the CD is 1.95. The average rank difference between the proposed CPBS-BLS and the models BLS, ELM, IF-BLS, NF-BLS, F-BLS, IF-TSVM, and H-ELM is 2.09, 1.17, 1.06, 0.56, 2.17, 2.75, and 1.52, respectively. Thereby, the proposed CPBS-BLS model has established a significant difference from the BLS, F-BLS, and IF-TSVM models according to the post-hoc test. However, CPBS-BLS could not establish a significant difference from other models, yet the proposed CPBS-BLS is superior to all existing models, as we have demonstrated by average accuracy and average rank. The Win-tie-loss test compares multiple models across datasets by tracking pairwise performance outcomes. For each model pair, we record wins, ties, and losses. Under the null hypothesis, each model should win on $\mathscr{D}/2$ datasets. The significance threshold is calculated by: threshold $= \frac{\mathscr{D}}{2} + \frac{1.96\sqrt{\mathscr{D}}}{2}$. For $\mathscr{D} = 29$ datasets, models require ≥ 20 wins (19.77 rounded up) to demonstrate statistical significance. Furthermore, the ties are evenly distributed between the compared models when the number of ties are even, and one of the tie is disregarded when the number of ties are odd. Table 4.2 shows the wins, ties, and losses of the models in the first column against models in the first row. Upon observing Table 4.2, the proposed model CPBS-BLS wins over BLS, IF-BLS, F-BLS, and IF-TSVM models by winning 22, 21, 22, and 24 datasets. Although the proposed model could not demonstrate a statistical difference from the rest of the models, but outperformed all the models in terms of accuracy and rank.

Table 4.2: Win-tie-loss test on UCI and KEEL datasets for the proposed CPBS-BLS model.

	BLS [1]	ELM [27]	IF-BLS[29]	NF-BLS [30]	F-BLS [29]	IF-TSVM [31]	H-ELM [32]
ELM	[19, 0, 10]						
IF-BLS	[20, 0, 9]	[12, 0, 17]					
NF-BLS	[19, 0, 10]	[15, 0, 14]	[14, 0, 15]				
F-BLS	[12, 0, 17]	[12, 0, 17]	[8, 0, 21]	[6, 0, 23]			
IF-TSVM	[12, 1, 16]	[13, 1, 15]	[11, 0, 18]	[5, 2, 22]	[12, 0, 17]		
H-ELM	[16, 0, 13]	[12, 0, 17]	[11, 0, 18]	[11, 0, 18]	[12, 2, 15]	[21, 2, 6]	
CPBS-BLS	[22, 1, 6]	[17, 3, 9]	[21, 0, 8]	[17, 0, 12]	[22, 0, 7]	[24, 1, 4]	[19, 0, 10]

† represents the proposed model.

4.3.2 Evaluation on CPBS-MvBLS model with AwA Repository

Table 4.3: Performance comparison of baseline and proposed CPBS-MvBLS model on AwA datasets.

Dataset ↓ Model →	ELM1 [27]	ELM2 [27]	BLS1 [1]	BLS2 [1]	SVM2K [24]	MVLDM [25]	MvTSVM[26]	CPBS-MvBLS †
Chimpanzee vs Giantpanda	71.53	72.92	72.92	82.64	**84.03**	72.22	47.22	82.64
Giantpanda vs Leopard	61.11	78.47	65.97	**83.33**	80.19	61.81	54.17	82.64
Giantpanda vs Persiancat	71.53	76.39	77.78	81.25	81.81	66.67	52.08	**84.72**
Giantpanda vs Raccoon	68.06	74.31	68.06	78.47	**80.19**	64.58	52.78	74.31
Giantpanda vs Seal	80.56	77.08	78.47	81.25	85.89	**86.81**	56.94	77.78
Leopard vs Persiancat	70.83	84.72	74.31	**86.11**	82.19	80.56	79.31	85.42
Leopard vs Pig	68.75	75	61.11	72.92	75	68.75	61.39	**79.17**
Chimpanzee vs Leopard	63.89	**83.33**	75	81.94	80.11	68.75	46.53	78.47
Leopard vs Hippopotamus	63.89	77.08	73.61	77.08	**78.17**	75	50.69	70.14
Leopard vs Humpbackwhale	89.58	91.67	90.28	**95.83**	90.75	89.58	79.31	90.97
Leopard vs Rat	72.22	76.39	62.5	72.92	**76.42**	65.28	68.61	73.61
Persiancat vs Pig	63.89	67.36	61.81	**78.47**	70	69.44	69.31	71.53
Persiancat vs Hippopotamus	75.69	79.86	70.14	**81.94**	76.81	75.69	76.53	80.56
Persiancat vs Humpbackwhale	70.84	88.19	81.25	**93.06**	71.67	85.42	71.39	91.67
Persiancat vs Raccoon	65.28	81.25	73.61	**83.33**	82.64	65.97	79.31	81.94
Persiancat vs Seal	72.22	71.53	71.53	**84.72**	80.42	83.33	73.47	81.25
Pig vs Humpbackwhale	82.64	**89.58**	82.64	70.83	80.19	88.89	77.92	86.81
Pig vs Raccoon	61.11	72.92	59.72	60.42	71.69	62.5	69.31	**73.61**
Pig vs Rat	62.5	59.72	64.58	50	**71.53**	64.58	68.61	61.11
Pig vs Seal	70.14	68.06	70.83	65.97	72.69	**72.92**	65.56	69.44
Hippopotamus vs Raccoon	72.22	70.14	67.36	79.17	78.47	75.69	75.14	**81.25**
Hippopotamus vs Rat	65.28	71.53	65.97	68.75	75.33	64.58	**75.83**	75
Chimpanzee vs Pig	68.75	**81.25**	65.97	81.25	50.42	66.67	51.39	77.78
Raccoon vs Rat	59.72	**70.83**	56.25	68.75	62.22	65.28	61.89	67.36
Raccoon vs Seal	78.47	84.03	79.17	87.5	**90.28**	75.69	75.39	87.5
Rat vs Seal	68.75	68.75	69.44	65.28	70.86	69.87	65.17	**71.53**
Chimpanzee vs Hippopotamus	71.53	70.14	77.08	**79.86**	70.94	78.47	54.86	77.08
Chimpanzee vs Raccoon	69.44	63.89	68.06	75.69	**80.33**	72.22	63.47	77.78
Chimpanzee vs Rat	57.64	75	63.89	**78.47**	77.08	68.06	52.78	77.08
Chimpanzee vs Seal	79.17	76.39	75	81.11	70.69	75.69	53.47	**83.33**
Average ACC	69.91	75.93	70.81	77.61	76.63	72.7	64.33	**78.45**

† represents the proposed model.

This section presents the experimental results obtained from evaluating the performance of the proposed CPBS-MvBLS model on 30 Animal with Attribute (AwA) datasets. Table 4.3 displays the classification accuracy for all baseline and proposed models. The baseline models ELM1, ELM2, BLS1, BLS2, SVM2K, MVLDM, and MvTSVM achieve average accuracies of 69.91%, 75.93%, 70.81%, 77.61%, 76.63%, 72.7%, and 64.33%, respectively. In comparison, our proposed CPBS-MvBLS model demonstrates superior performance with average accuracy of 78.45% substantially outperforming all baseline models. Furthermore, the proposed models attained the most favorable average

rankings. CPBS-MvBLS achieve an average rank of 2.78, while the baseline models ELM1, ELM2, BLS1, BLS2, SVM2K, MVLDM, and MvTSVM obtained considerably higher average ranks of 5.97, 4.05, 5.8, 2.93, 3.1, 4.77, and 6.6, respectively. These results establish the superiority of the proposed CPBS-MvBLS model over all the baseline models.

Statistical analysis using the Friedman test with $\mathscr{T} = 8$ and $\mathscr{D} = 30$ yielded $\chi_r^2 = 88.84$ and Friedman static is $F_F = 21.26$. Given that the critical Friedman value for dof 7 and 203 is 2.05. Since $21.26 > 2.05$, we reject the null hypothesis and confirm the statistically significant differences between the performance of the model. The critical difference (CD) for the Nemenyi post-hoc test is calculated as 1.92. The average rank difference between the proposed CPBS-MvBLS and the existing models ELM1, ELM2, BLS1, BLS2, SVM2K, MVLDM, and MvTSVM is 3.91, 1.27, 3.02, 0.15, 0.32, 1.99, and 3.28, respectively. After comparing to the CD value, CPBS-MvBLS exihibit a difference from ELM1, BLS1, MVLDM, and MvTSVM. The remaining baseline models are outperformed by proposed model in terms of both average accuracy and average ranking. For the Win-tie-loss test, the established threshold is 21. As shown in Table 4.4, CPBS-MvBLS achieved significant differences from ELM1, ELM2, BLS1, MVLDM, and MvTSVM with victories of 27, 22, 25, 23, and 28 wins, respectively. These results further validate the superior performance of our proposed CPBS-MvBLS model. Both the average accuracy and average rank metrics indicate that the other two baselines are consistently outperformed.

Table 4.4: Win-tie-loss test on AwA datasets for the proposed CPBS-MvBLS model.

	ELM1 [27]	ELM2 [27]	BLS1 [1]	BLS2 [1]	SVM2K [24]	MVLDM [25]	MvTSVM [26]
ELM2 [27]	[21, 1, 8]						
BLS1 [1]	[16, 2, 12]	[7, 2, 21]					
BLS2 [1]	[25, 0, 5]	[18, 2, 10]	[26, 0, 4]				
SVM2K [24]	[26, 0, 4]	[19, 1, 10]	[25, 0, 5]	[14, 0, 16]			
MVLDM [25]	[20, 3, 7]	[9, 0, 21]	[20, 1, 9]	[6, 0, 24]	[9, 0, 21]		
MvTSVM [26]	[11, 0, 19]	[5, 0, 25]	[12, 0, 18]	[4, 0, 26]	[2, 0, 28]	[6, 0, 24]	
CPBS-MvBLS[†]	[27, 0, 3]	[22, 1, 7]	[25, 1, 4]	[10, 2, 18]	[16, 1, 13]	[23, 0, 7]	[28, 0, 2]

[†] represents the proposed model.

4.3.3 Evaluation on CPBS-MvBLS model with UCI and KEEL Repository

Our comprehensive evaluation of the proposed CPBS-MvBLS framework on standard benchmark datasets from UCI and KEEL repositories demonstrates

Table 4.5: Performance comparison of baseline and proposed CPBS-MvBLS model on UCI and KEEL datasets.

Dataset ↓ Model →	ELM1 [27]	ELM2 [27]	BLS1 [1]	BLS2 [1]	SVM2K [24]	MVLDM [25]	MvTSVM[26]	CPBS-MvBLS †
abalone9-18	95	**96.82**	95.91	96.36	94.52	93.15	95.7	95.45
acute_inflammation	83.33	97.22	**100**	**100**	**100**	**100**	90.48	97.22
acute_nephritis	**100**	**100**	91.67	**100**	**100**	**100**	**100**	**100**
aus	**89.9**	88.46	87.98	86.06	81.16	86.53	83.44	87.92
blood	76.89	76	77.33	76.11	78.57	**80.8**	75.57	77.33
breast_cancer_wisc	95.71	**96.19**	88.1	93.1	91.61	93.61	89.9	94.76
breast_cancer_wisc_diag	93.57	91.81	89.42	87.08	95.49	93.15	92.98	**97.66**
breast_cancer_wisc_prog	78.33	23.33	68.33	68.33	58.33	71.17	74.82	**81.96**
brwisconsin	94.63	95.12	91.07	96.1	**98.04**	95.59	95.82	96.45
bupa or liver-disorders	52.88	49.04	63.46	65.38	65.05	55.34	55.37	**72.63**
checkerboard_Data	**89.9**	88.46	83.98	86.06	81.16	84.06	81.16	86.54
credit_approval	84.13	84.13	82.62	84.62	**85.99**	85.51	79.3	83.65
cylinder_bands	70.13	66.23	71.62	**74.03**	73.2	71.9	73.26	72.61
echocardiogram	80	85	82.5	**87.5**	79.49	84.62	76.09	**87.5**
fertility	86.67	80	90	**100**	86.67	86.67	88.57	90
haberman_survival	78.26	79.35	76.09	78.26	74.73	71.43	71.16	**80.43**
horse_colic	76.58	76.58	80.78	**84.68**	81.82	83.64	77.91	81.3
ilpd_indian_liver	70.29	58.86	68.57	70.29	**72.99**	**72.99**	71.15	71.09
mammographic	78.89	76.47	70.28	72.01	**82.29**	80.9	81.58	79.93
molec_biol_promoter	71.88	59.38	56.25	81.25	**83.87**	**83.87**	73.33	81.25
shuttle-6_vs_2-3	**100**	**100**	**100**	**100**	97.1	95.65	96	**100**
sonar	65.08	63.49	77.78	79.37	**85.48**	79.03	79.45	81.38
statlog_australian_credit	53.85	68.75	70.67	70.67	69.57	**71.98**	67.08	69.71
statlog_german_credit	77.33	76	76.33	**78**	74	73.67	69.57	77.33
statlog_heart	83.95	**90.12**	82.72	81.89	82.72	82.72	73.54	87.65
tic_tac_toe	**99.65**	81.6	89.65	70.49	90	90	98.06	96.88
transfusion	76.89	76	75.11	76.89	76.34	75	77.1	**81.05**
vehicle1	78.74	72.05	79.13	**82.28**	75.89	65.61	81.11	79.22
votes	92.37	96.95	92.42	90.47	93.08	90.58	93.77	**97.71**
wpbc	35.59	35.59	77.97	79.66	72.41	61.01	75.74	**82.22**
Average ACC	80.35	77.63	81.26	83.23	82.72	82.01	81.3	**85.63**

† represents the proposed model.

its superior classification performance, as evidenced by the comparative accuracy results presented in Table 4.5. The experimental results reveal that CPBS-MvBLS achieves the highest average accuracy of 85.63%, outperforming all competing models, including ELM1 (80.35%), ELM2 (77.63%), BLS1 (81.26%), BLS2 (83.23%), SVM2K (82.72%), MVLDM (82.01%), and MvTSVM (81.3%). Further analysis of average rankings shows CPBS-MvBLS attaining the top position (rank 2.87), followed by ELM1 (4.75), ELM2 (5.25), BLS1 (5.22), BLS2 (3.82), SVM2K (4.28), MVLDM (4.7), and MvTSVM (5.12).

To statistically validate these findings, we conducted the Friedman test with $\mathscr{T} = 8$ models and $\mathscr{D} = 30$ datasets, yielding $\chi_r^2 = 82.68$ and $F_F = 18.83$, which significantly exceeds the critical value of 2.05 (dof: 7 and 203, $\alpha = 0.05$), leading us to reject the null hypothesis ($2.05 < 18.83$). Nemenyi post hoc analysis with a critical difference of 1.92 confirms that CPBS-MvBLS significantly outperforms ELM2 (rank difference: 2.38), BLS1 (rank difference: 2.35), and MvTSVM (rank difference: 2.25). However, CPBS-MvBLS could not establish the statistical difference from ELM1 (rank difference: 1.88), BLS2 (rank difference: 0.95), SVM2K (rank difference: 1.41), and MVLDM (rank difference: 1.83) yet CPBS-MvBLS demonstrate the superior performance over these baselines in terms of both average accuracy and ranking. The Win-tie-loss test with

threshold = 21 (round up of 20.367) further substantiates these results, showing CPBS-MvBLS achieving decisive victories against ELM1 (22 wins), ELM2 (22 wins), BLS1 (23 wins), MVLDM (21 wins), and MvTSVM (23 wins) while maintaining consistent superiority over all other baseline models across both accuracy and ranking metrics, thereby establishing its statistical significance evidence over existing frameworks.

Table 4.6: Win-Tie-loss test on UCI and KEEL datasets for the proposed CPBS-MvBLS model.

	ELM1 [27]	ELM2 [27]	BLS1 [1]	BLS2 [1]	SVM2K [24]	MVLDM [25]	MvTSVM [26]
ELM2	[9, 5, 16]						
BLS1	[13, 1, 16]	[14, 1, 15]					
BLS2	[17, 2, 11]	[18, 2, 10]	[20, 4, 6]				
SVM2K	[15, 1, 14]	[19, 1, 10]	[16, 1, 13]	[11, 2, 17]			
MVLDM	[14, 1, 15]	[17, 1, 12]	[17, 1, 12]	[13, 2, 15]	[8, 7, 15]		
MvTSVM	[16, 1, 13]	[15, 1, 14]	[14, 0, 16]	[8, 1, 21]	[10, 2, 18]	[14, 1, 15]	
CPBS-MvBLS	[22, 2, 6]	[22, 2, 6]	[23, 3, 4]	[17, 4, 9]	[19, 1, 10]	[21, 1, 8]	[23, 1, 6]

[†] represents the proposed model.

4.3.4 Sensitivity Analyses of Hyperparameter μ on CPBS-BLS and CPBS-MvBLS Model

By adjusting μ inside the range specified in the experimental setup and fixing other parameters at their optimal values, we examined the effect of the kernel parameter μ on the performance of our proposed models. Figure 4.1 illustrates the performance of the proposed CPBS-BLS model on the datasets: echocardiogram, haberman_survival, and the proposed CPBS-MvBLS model on the datasets: Persiancat vs Humpbackwhale, and Giantpanda vs Persiancat, achieving peak accuracy at different values of μ. In the echocardiogram dataset (Fig. 4.1a), the accuracy attains its optimal value on $\mu = 2^2$. The haberman_survival dataset (Fig. 4.1b) attains very low accuracy within the range $\{2^{-5}, 2^{-4}, ..., 2^{-1}\}$, but increases drastically on $\mu = 1$ and achieves optimal accuracy on $\mu = 2^3$. Similarly, one can observe the Figures 4.1c and 4.1d. From the above analyses, we can observe that the performance of the proposed CPBS-BLS and CPBS-MvBLS models affect with the different values of μ. Our result highlights the influence of dataset characteristics on the accuracy of the proposed CPBS-BLS and CPBS-MvBLS models, emphasizing the necessity of fine tuning the parameter μ, to achieve optimal performance.
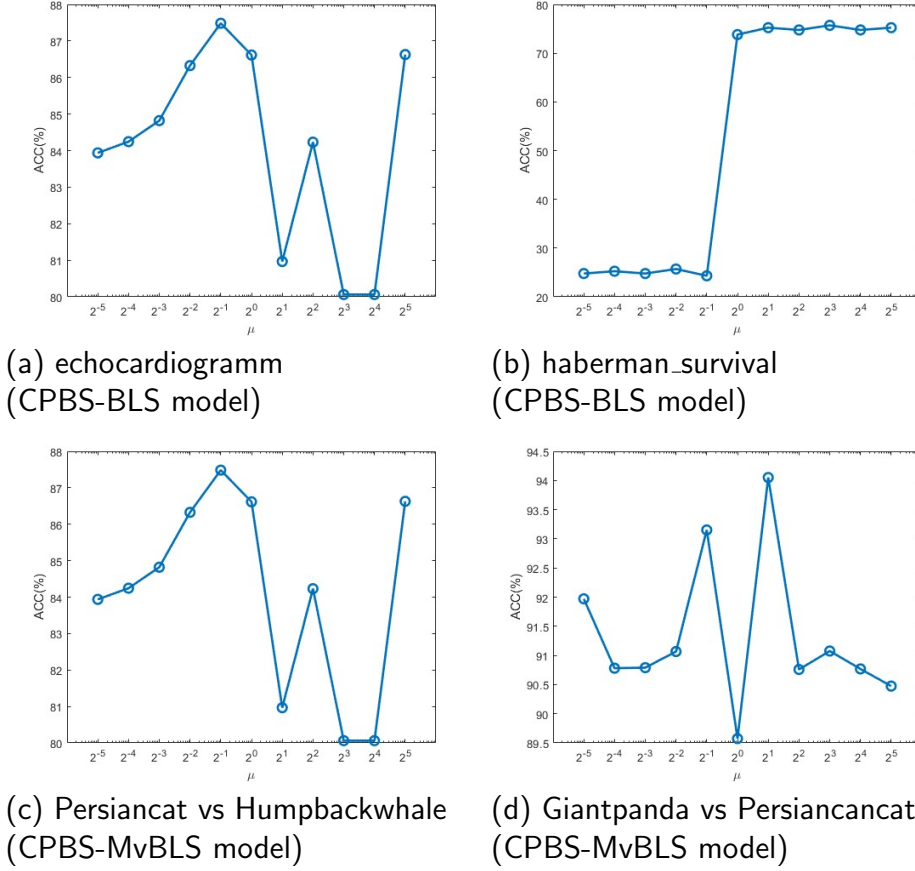
Figure 4.1: Effect of the hyperparameter μ on the accuracy of the proposed CPBS-BLS and CPBS-MvBLS models.

4.3.5 Sensitivity Analyses of Hyperparameters μ and τ on CPBS-BLS and CPBS-MvBLS Models

In this section, we investigate the sensitivity of the hyperparameters by adjusting the values of μ and τ to analyze the effect of hyperparameters on the proposed CPBS-BLS and CPBS-MvBLS model's generalization capability. Figure 4.2 illustrates the model's accuracy changes as the hyperparameters μ and τ are adjusted. It is observable that the accuracy of the aus dataset (Fig. 4.2a) achieves optimal values in the range of $\{2^{-5}, 2^{-4}..., 2^{-1}\}$ for μ and $\frac{7}{8}$ to 1 for τ. Figure 4.2b (echocardiogram dataset) attains its optimal accuracy within the range $\{2^1, 2^2..., 2^4\}$ (for μ) and $\frac{3}{4}$ to 1 (for τ). Similarly, Figures 4.2c and 4.2d (on CPBS-MvBLS) can be observed. Consequently, the observations emphasize the importance of carefully selecting hyperparameter μ and τ's values to achieve optimal model performance.

(a) aus
(CPBS-BLS model)

(b) echocardiogram
(CPBS-BLS model)

(c) Giant panda vs Humpback whale
(CPBS-MvBLS model)

(d) Giantpanda vs Persiancancat
(CPBS-MvBLS model)
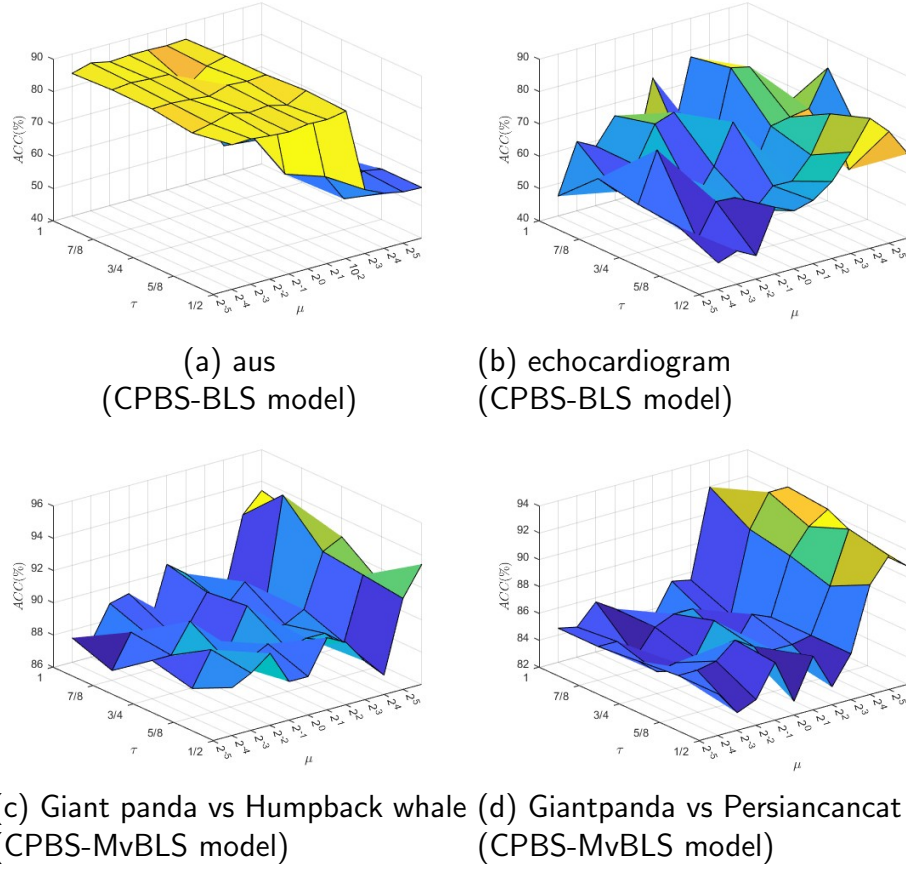
Figure 4.2: Effect of the hyperparameters μ and τ on the accuracy of the proposed CPBS-BLS and CPBS-MvBLS models.

4.3.6 Sensitivity Analyses of Hyperparameter τ on CPBS-BLS and CPBS-MvBLS Models

In this subsection, we investigate the influence of the hyperparameter τ on the accuracy of both proposed models. Figure 4.3 depicts the variations in model accuracy as the hyperparameter τ is adjusted. In Figure 4.3a, the optimal accuracy is observed at $\tau = \frac{3}{4}$, and in Figure 4.3b, the optimal accuracy is attained at two different values $\tau = \frac{5}{8}$ and $\tau = \frac{3}{4}$. Additionally, the findings presented in Figure 4.3c shows that the optimal accuracy of the proposed CPBS-MvBLS model is observed at $\tau = \frac{7}{8}$, whereas in Figure 4.3d, the optimal accuracy is achieved at $\tau = \frac{3}{4}$ and $\tau = 1$. Overall, these findings highlight the necessity of meticulously setting the values for the hyperparameter τ to attain the optimal performance of both proposed models.
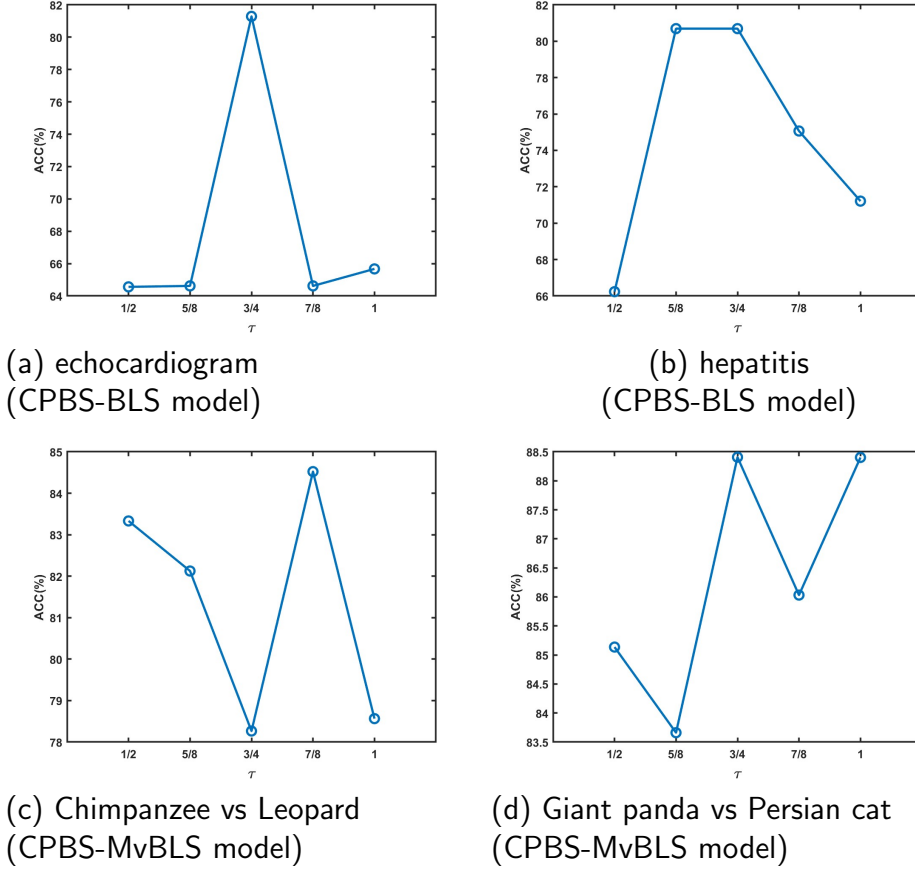
(a) echocardiogram
(CPBS-BLS model)

(b) hepatitis
(CPBS-BLS model)

(c) Chimpanzee vs Leopard
(CPBS-MvBLS model)

(d) Giant panda vs Persian cat
(CPBS-MvBLS model)

Figure 4.3: Effect of the hyperparameter τ on the accuracy of the proposed CPBS-BLS and CPBS-MvBLS models.

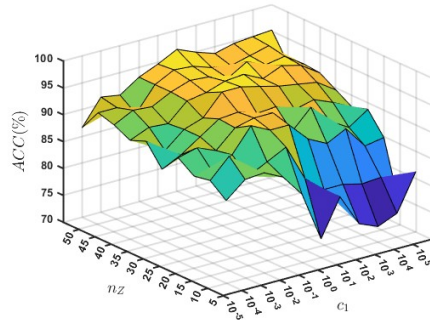### 4.3.7 Sensitivity Analyses of Hyperparameters $c(c_1)$ and n_Z on CPBS-BLS and CPBS-MvBLS Models

Figure 4.4 demonstrates the sensitivity of models accuracy to the hyperparameters n_Z and c or c_1, revealing distinct optimal regimes across different datasets. In Fig. 4.4a, peak accuracy occurs for $n_Z = \{30, ..., 50\}$ with $c = \{10^{-3}, ..., 10^1\}$. Fig. 4.4b shows optimal performance in the same n_Z range but with c in $\{10^1, ..., 10^5\}$. Figs. 4.4c and 4.4d exhibit optimal behavior in range $n_Z = \{35, ..., 50\}$ paired with $c_1 = \{10^1, ..., 10^4\}$ and $c_1 = \{10^{-3}, 10^0\}$, respectively. These results highlight two key patterns: first, the choice in the range $\{30, ..., 50\}$ for the number of feature groups (n_Z) provides the optimal accuracy across all datasets, suggesting this as a robust design choice; second, the regularization parameter c or c_1 shows greater variability in the whole range, indicating its stronger dependence on specific experimental setups.
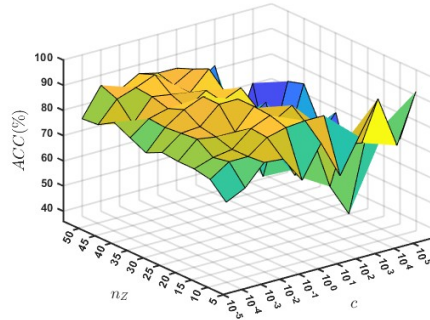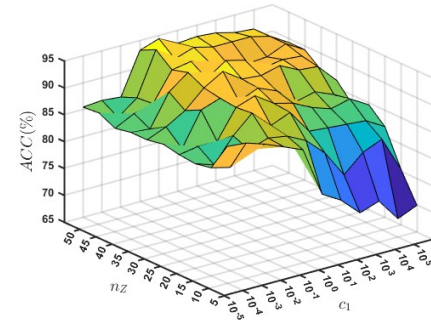
(a) shuttle-6_vs_2-3
(CPBS-BLS model)
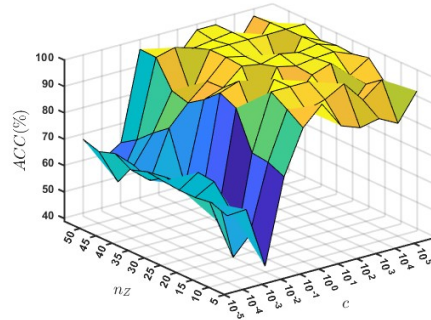
(b) tic_tac_toe
(CPBS-BLS model)

(c) Giantpanda vs Humpbackwhale
(CPBS-MvBLS model)

(d) Leopard vs Humpbackwhale
(CPBS-MvBLS model)

Figure 4.4: Effect of the hyperparameters $c(c_1)$ and n_Z on the accuracy of the proposed CPBS-BLS and CPBS-MvBLS models.

4.3.8 Sensitivity Analyses of Hyperparameter n_Z on CPBS-BLS and CPBS-MvBLS Models

(a) shuttle-6_vs_2-3
(CPBS-BLS model)

(b) new-thyroid1
(CPBS-BLS model)

(c) Leopard vs Humpbackwhale
(CPBS-MvBLS modl)

(d) Persiancat vs Humpbackwhale
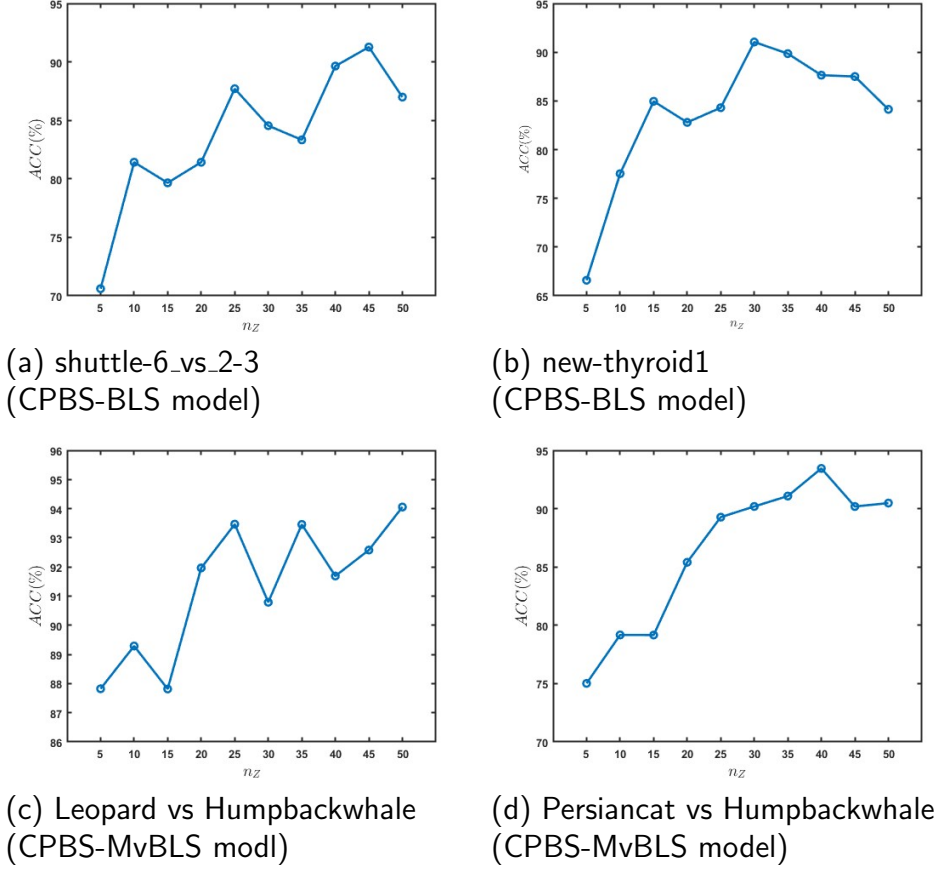(CPBS-MvBLS model)

Figure 4.5: Effect of the hyperparameter n_Z on the accuracy of the proposed CPBS-BLS and CPBS-MvBLS models.

To investigate the impact of the hyperparameter n_Z on model performance, we conduct a comprehensive sensitivity analysis by evaluating different values of n_Z across multiple datasets. As illustrated in Figure 4.5, the results demonstrate a clear relationship for the choice of n_Z to the optimal classification accuracy. Figures 4.5a and 4.5b reveal that the model's accuracy improves monotonically with increasing values of n_Z, reaching its peak performance at $n_Z = 45$ and $n_Z = 30$, respectively. Similarly, Figures 4.5c and 4.5d exhibit analogous behavior, with the highest accuracy achieved at $n_Z = 50$ and $n_Z = 40$, respectively. These findings underscore the importance of carefully tuning n_Z to strike a balance between model generalization capabilities.

Chapter 5

Conclusions and Future Directions

The study in this thesis has focused on examining state-of-the-art randomized neural networks, namely the BLS. This model have been widely applied in different scenarios and have demonstrated commendable performance. Nevertheless, it is important to acknowledge that BLS posses certain limitations and intricacies that have been addressed and mitigated in the research presented in this thesis. In this thesis, efforts have been made to enhance the performance of BLS algorithms by addressing inherent issues in the models and proposing solutions to improve overall effectiveness and applicability in real-world scenarios.

5.1 Conclusion and Future Work

In this section, we conclude the thesis and explore potential future directions arising from our findings. In this thesis, we proposed a novel graph embedded multi-view broad learning system (GrMv-BLS) model to enhance the classification performance and generalization capabilities of the BLS framework. The proposed GrMv-BLS model integrates GE and MVL approaches within the BLS framework, which leverages unique features from diverse perspectives to better interpretation of the intricate patterns and significant information of the datasets. In addition, the combination of graph embedding together with subspace learning and LFDA weighting scheme, improves the capability of preserving the inherent relationship and topological structure associated with the data, which increases the robustness of the proposed GrMv-BLS model.

Thereby, the GrMv-BLS model enriches the generalized classification performance of the BLS framework. The proposed model's performance is examined by contrasting it with a number of baseline models using the AwA, UCI, and KEEL datasets. Out of all the models taken into experiments, the proposed GrMv-BLS model achieves the highest accuracy and lowest ranking, flaunting the better performance of the GrMv-BLS model in comparison to existing frameworks. In addition, a thorough statistical and theoretical analysis is employed to confirm the efficacy and generalization performance of the proposed GrMv-BLS model. Comprehensive experimental results, coupled with rigorous statistical testing and theoretical analysis, show that the GrMv-BLS has superior generalization capability while providing provable learning guarantees through derived error bounds.

Furthermore, our research presents significant advancements in the realm of robust BLS frameworks through the development of two novel variants: CPBS-BLS and CPBS-MvBLS. The CPBS-BLS framework integrates a bell-shaped membership function into the traditional BLS model, effectively addressing the class imbalance problem by incorporating the imbalance ratio. This innovative approach leads to a more accurate assignment of membership to data points via class probability and enhances the overall performance of the BLS framework on datasets with high noise and imbalanced classes. Similarly, the CPBS-MvBLS model extends these concepts by merging MvBLS with a bell-shaped membership function that also factors in class probability and imbalance ratio, further refining its predictive capabilities by leveraging the multi-view approach to meticulously understand the data. Rigorous evaluations of the CPBS-BLS model against several baseline models across UCI and KEEL repositories demonstrate marked improvements in accuracy and ranking, underscoring its efficacy. On the other hand, the performance of the CPBS-MvBLS model was assessed using the AwA, UCI, and KEEL datasets, revealing comparable enhancements. Comprehensive statistical analyses, including a ranking scheme, Friedman test, Nemenyi post hoc test, and win-tie-loss test, provide robust evidence of the proposed models' superiority. The results from these tests strongly affirm that both CPBS-BLS and CPBS-MvBLS significantly outperform existing baseline models, marking an important contribution to the field and opening avenues for future research in class imbalance mitigation and membership function optimization in BLS frameworks. Furthermore, to evaluate the impact of hyperparameters on model performance, we conducted a comprehensive sensitivity analysis. The results underscored

the critical importance of fine-tuning these hyperparameters to achieve optimal performance. This finding highlights the need for careful observation of hyperparameters in model development, as even minor adjustments can lead to significant improvements in accuracy and efficiency. By prioritizing hyperparameter optimization, we can enhance the overall effectiveness of the model and ensure its robustness in various applications.

Our proposed GrMv-BLS, CPBS-BLS, and CPBS-MvBLS models have demonstrated outstanding performance in binary classification tasks for multiview and diverse datasets. Exploring the integration of the asymmetric and bounded loss function into BLS frameworks is an intriguing avenue for future research.

Bibliography

[1] CL Philip Chen and Zhulin Liu. Broad learning system: An effective and efficient incremental learning system without the need for deep architecture. *IEEE Transactions on Neural Networks and Learning Systems*, 29 (1):10–24, 2018.

[2] Zheng Liu, Shiluo Huang, Wei Jin, and Ying Mu. Graph-based broad learning system for classification. *Neurocomputing*, 463:535–544, 2021. ISSN 0925-2312.

[3] Junwei Duan, Shiyi Yao, Jiantao Tan, Yang Liu, Long Chen, Zhen Zhang, and C. L. Philip Chen. Extreme fuzzy broad learning system: Algorithm, frequency principle, and applications in classification and regression. *IEEE Transactions on Neural Networks and Learning Systems*, 36 (2):2946–2957, 2025. doi: 10.1109/TNNLS.2023.3347888.

[4] Jifeng Guo and CL Philip Chen. A robust semi-supervised broad learning system guided by ensemble-based self-training. *IEEE Transactions on Cybernetics*, 2024. doi: 10.1109/TCYB.2024.3393020.

[5] Xiaojia Wang, Wenjing Bian, and Xibin Zhao. Robust unsupervised anomaly detection for surface defects based on stacked broad learning system. *IEEE/ASME Transactions on Mechatronics*, 2024. doi: 10.1109/TMECH.2024.3465563.

[6] Jing Zhao, Xijiong Xie, Xin Xu, and Shiliang Sun. Multi-view learning overview: Recent progress and new challenges. *Information Fusion*, 38: 43–54, 2017.

[7] Palash Goyal and Emilio Ferrara. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*, 151:78–94, 2018.

[8] Zaiyang Yu, Lusi Li, Jinlong Xie, Changshuo Wang, Weijun Li, and Xin Ning. Pedestrian 3d shape understanding for person re-identification via multi-view learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024. doi: 10.1109/TCSVT.2024.3358850.

[9] Vikas Sindhwani, Partha Niyogi, and Mikhail Belkin. A co-regularizfation approach to semi-supervised learning with multiple views. In *Proceedings of ICML workshop on learning with multiple views*, volume 2005, pages 74–79. Citeseer, 2005.

[10] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

[11] Ying Liu, Cai Xu, Long Chen, Meng Yan, Wei Zhao, and Ziyu Guan. Table: Time-aware balanced multi-view learning for stock ranking. *Knowledge-Based Systems*, 303:112424, 2024.

[12] A. Quadir and M. Tanveer. Multiview learning with twin parametric margin SVM. *Neural Networks*, 180:106598, 2024.

[13] A. Quadir, M. Sajid, and M. Tanveer. Multiview random vector functional link network for predicting DNA-binding proteins. *arXiv preprint arXiv:2409.02588*, 2024.

[14] Hongyun Cai, Vincent W Zheng, and Kevin Chen-Chuan Chang. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering*, 30(9): 1616–1637, 2018.

[15] Anuradha Kumari, M. Tanveer, C. T. Lin, and the Alzheimer's Disease Neuroimaging Initiative. Class probability and generalized bell fuzzy twin svm for imbalanced data. *IEEE Transactions on Fuzzy Systems*, 32(5): 3037–3048, 2024. doi: 10.1109/TFUZZ.2024.3366936.

[16] MA Ganaie, Mohammad Tanveer, and Alzheimer's Disease Neuroimaging Initiative. KNN weighted reduced universum twin SVM for class imbalance learning. *Knowledge-Based Systems*, 245:108578, 2022.

[17] Yingnan Pan, Qi Li, Hongjing Liang, and Hak-Keung Lam. A novel mixed control approach for fuzzy systems via membership functions online learning policy. *IEEE Transactions on Fuzzy Systems*, 30(9):3812–3822, 2021.

[18] Jingyu Wang, Hengheng Yin, Feiping Nie, and Xuelong Li. Adaptive and fuzzy locality discriminant analysis for dimensionality reduction. *Pattern Recognition*, 151:110382, 2024.

[19] Alexandros Iosifidis, Anastasios Tefas, and Ioannis Pitas. Graph embedded extreme learning machine. *IEEE Transactions on Cybernetics*, 46(1): 311–324, 2015.

[20] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Animals with Attributes dataset. http://attributes.kyb.tuebingen.mpg.de/, 2009. Accessed: [Insert Last Accessed Date].

[21] Dheeru Dua and Casey Graff. UCI machine learning repository. 2017.

[22] J Derrac, S Garcia, L Sanchez, and F Herrera. KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing*, 17:255–287, 2015.

[23] M. Sajid, A. K. Malik, and M. Tanveer. Intuitionistic fuzzy broad learning system: Enhancing robustness against noise and outliers. *IEEE Transactions on Fuzzy Systems*, pages 1–10, 2024. doi: 10.1109/TFUZZ.2024. 3400898.

[24] Jason Farquhar, David Hardoon, Hongying Meng, John Shawe-Taylor, and Sandor Szedmak. Two view learning: SVM-2K, theory and practice. *Advances in Neural Information Processing Systems*, 18, 2005.

[25] Kun Hu, Yingyuan Xiao, Wenguang Zheng, Wenxin Zhu, and Ching-Hsien Hsu. Multiview large margin distribution machine. *IEEE Transactions on Neural Networks and Learning Systems*, 2024, 10.1109/TNNLS.2023.3349142.

[26] Xijiong Xie and Shiliang Sun. Multi-view twin support vector machines. *Intelligent Data Analysis*, 19(4):701–712, 2015.

[27] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: theory and applications. *Neurocomputing*, 70(1-3):489–501, 2006.

[28] C. L. Philip Chen and Zhulin Liu. Broad learning system: An effective and efficient incremental learning system without the need for deep architecture. *IEEE Transactions on Neural Networks and Learning Systems*, 29(1):10–24, 2018. doi: 10.1109/TNNLS.2017.2716952.

[29] M Sajid, Ashwani Kumar Malik, and M Tanveer. Intuitionistic fuzzy broad learning system: Enhancing robustness against noise and outliers. *IEEE Transactions on Fuzzy Systems*, 2024.

[30] Shuang Feng and CL Philip Chen. Fuzzy broad learning system: A novel neuro-fuzzy model for regression and classification. *IEEE Transactions on Cybernetics*, 50(2), 2018.

[31] Salim Rezvani, Xizhao Wang, and Farhad Pourpanah. Intuitionistic fuzzy twin support vector machines. *IEEE Transactions on Fuzzy Systems*, 27 (11):2140–2151, 2019.

[32] Jiexiong Tang, Chenwei Deng, and Guang-Bin Huang. Extreme learning machine for multilayer perceptron. *IEEE Transactions on Neural Networks and Learning Systems*, 27(4):809–821, 2015.