IN PARTNERSHIP WITH PLYMOUTH UNIVERSITY

Name: PMB Jayakody
Student Reference Number: 10899554

Module Code: PUSL 3123	Module Name: AI & Machine Learning
Coursework Title: PUSL 3123 AI & Machine Learning Final Coursework Report	
Deadline Date: 15th of December 2024	Member of staff responsible for coursework: Mr. Neamah Al Naffakh
Programme: BSc (Hons) Software Engineering	

Please note that University Academic Regulations are available under Rules and Regulations on the University website www.plymouth.ac.uk/studenthandbook.

Group work: please list all names of all participants formally associated with this work and state whether the work was undertaken alone or as part of a team. Please note you may be required to identify individual responsibility for parts.

Student Name	NSBM ID	Plymouth ID
RAVL Perera	26852	10899656
PMB Jayakody	27353	10899554
YGA Amarasinghe	24735	10899158
DNL Prematthilaka	27509	10899668
DPGD Dayoda	28118	10898439

We confirm that we have read and understood the Plymouth University regulations relating to Assessment Offences and that we are aware of the possible penalties for any breach of these regulations. We confirm that this is the independent work of the group.

Signed on behalf of the group: PMB Jayakody

Individual assignment: ***I confirm that I have read and understood the Plymouth University regulations relating to Assessment Offences and that I am aware of the possible penalties for any breach of these regulations. I confirm that this is my independent work.***

Signed :

Use of translation software: failure to declare that translation software or a similar writing aid has been used will be treated as an assessment offense.

I *have used/not used translation software.

If used, please state the name of
the software……

Overall mark _____% Assessors Initials _____ Date_____

PUSL 3123 AI & Machine Learning

Coursework Final Report

Group 49

Plymouth Batch 11 Software Engineering Degree Program

Faculty of Computing

NSBM Green University Town

Acknowledgment

We would like to thank our module creator, whose vision and dedication transformed what would have been a purely academic project into a meaningful learning process. Their commitment to growth and stimulation of curiosity has been instrumental in shaping our journey into this assignment.

Special thanks go to our project mentor, Mr. Neamah AI Naffakh, for the precious advice, encouragement, and sharing of insight. His experience has been an important inspiration and substantially affected our project development & Final report.

Equally, we would like to express our gratitude to our Shadow Lecturer, Ms. Chathurma Wijesinghe, who stood by us from the very commencement of this task. Indeed, her considerate advice and constructive criticism have guided us through problems and guaranteed the quality of our output.

Finally, we would like to thank all those who, in one way or another, supported us with their valuable ideas and views. Every one of your inputs added to our project and stabilized the effort put into finishing it. We highly appreciate the generosity, advice, and assistance given by each of these individuals involved in this undertaking.

Table of Contents

Acknowledgment....4
01) Introduction....7
02) Background Literature Review....8
03) Testing Methodology....10
04) Evaluation....11
04.01) Analyzing Data Variability: Inter and Intra- User Variance....11
04.01.01) FreqD Analysis....11
04.01.02) TimeD Analysis....14
04.01.03) Time FreqD Analysis....17
04.02) Pre-Processing the Data Before Training Process....20
04.02.01) Training Process....20
04.02.02) Evaluation Process....26
05) Optimization....30
06) Conclusion & References....32
07) Appendix....33

Table Of Figures

Figure 1: FreqD Features FDay 11
Figure 2 : FreqD Features FDay 11
Figure 3 : FreqD Features FDay 12
Figure 4: Temporal Stability FreqD Features 12
Figure 5 : Dimensionality Analysis FreqD Features FDay 13
Figure 6 : Distribution Analysis TimeD Features FDay 14
Figure 7 : Variance Analysis TimeD Features FDay 14
Figure 8 : Feature Importance TimeD FDay 15
Figure 9 : Temporal Stability TimeD Features 15
Figure 10 : Dimensionality Analysis TimeD Features FDay 16
Figure 11: Distribution Analysis Time FreqD FDay 17
Figure 12 : Variance Analysis Time FreqD FDay 17
Figure 13 : Feature Importance Time FreqD Features FDay 18
Figure 14 : Temporal Stability Time FreqD Features 18
Figure 15 : Dimensionality Analysis Time FreqD Features FDay 19
Figure 16 : Result of the Training FreqD FDay 20
Figure 17 : Result of the Training TimeD FDay 21
Figure 18 : Result of the Time FreqD FDay 22
Figure 19 : Result of the FreqD MDay 23
Figure 20 : Result of the TimeD MDay 24
Figure 21 : Result of the Time FreqD MDay 25
Figure 22: FreqD FDay Binary Model 27
Figure 23: TimeD FDay Binary Model 28
Figure 24: Time FreqD FDay Binary Model 28
Figure 25: FreqD MDay Binary Model 29
Figure 26: TimeD MDay Binary Model 30
Figure 27: Time FreqD MDay Binary Model 30

List of Tables

Table 1: Table of the FreqD FDay Training Process 22
Table 2: Table of the TimeD FDay Training Process 23
Table 3: Table of the Time FreqD FDay Training Process 24
Table 4: Table of the FreqD MDay Training Process 25
Table 5: Table of the TimeD MDay Training Process 26
Table 6: Table of the Time FreqD MDay Training Process 27

01) Introduction

Digital devices such as smartphones and smartwatches are increasingly used to handle sensitive data, including online payments and banking information. Conventional authentication techniques like passwords and biometrics often lack the capability to provide ongoing verification, exposing users to potential security risks between authentications. Therefore, exploring continuous and transparent user authentication methods that leverage built-in sensors like accelerometers has become essential(Stylios et al., 2021).

This assessment investigates acceleration-based authentication, utilizing motion data from accelerometers to authenticate users based on their unique movement patterns. Specifically, we examine acceleration features in both time and frequency domains, evaluating them as potential indicators of user identity. The dataset contains various feature vectors, such as frequency domain features (U01_Acc_FreqD_FDay), time domain features (U01_Acc_TimeD_FDay), and combined time-frequency domain features (U01_Acc_TimeD_FreqD_FDay), captured across same-day and cross-day sessions for consistency evaluation. The feature counts (NF) and sample sizes (NS) within these vectors vary, providing a comprehensive set of attributes for model training.

To analyze and verify the effectiveness of these features, we first perform descriptive statistics to explore inter-user and intra-user variance, offering insights into how motion data differs between users and within individual users across sessions. For model training and evaluation, we employ a Feedforward Multi-Layer Perceptron (feedforward net) neural network, configured to handle the complexity of acceleration-based features. Additionally, we apply feature selection and classifier optimization techniques to enhance the neural network's performance, aiming for a model that can accurately and consistently authenticate users(Italis et al., 2023).

This study contributes to the development of non-intrusive, continuous authentication systems, highlighting the role of neural networks and acceleration-based features in enhancing digital security in an increasingly mobile world.

02) Background Literature Review

Acceleration-based user authentication has rapidly evolved with the integration of motion sensors in mobile devices, offering a dynamic and continuous alternative to static login methods. This approach leverages unique behavioral patterns, like gait and hand movements, to build biometric profiles that validate user identity throughout a session. Gait recognition, a prominent technique in this domain, identifies individual walking styles through accelerometer and gyroscope data, particularly improving accuracy with multi-sensor setups involving both smartphones and smartwatches. Hand-specific activities, such as typing, clapping, and eating, are also gaining attention as they create distinct movement patterns that classifiers like Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM) can effectively model. Among these, Random Forest is especially effective at handling complex, multi-dimensional motion data, achieving high accuracy rates often exceeding 95% (Verma et al., 2022).

Active Authentication (AA) methods are further advancing by focusing on secure, lightweight solutions for constrained environments like smart cards and SIM/eSIM cards. Mobile Match-On-Card (MMOC) techniques within AA often utilize Linear Discriminant Analysis (LDA) and Linear SVM (L-SVM) due to their computational efficiency and compatibility with resource-limited devices. Additionally, distance-based metrics such as Minkowski distance are frequently applied, striking a balance between low processing demand and favorable accuracy, with some gait-based systems achieving equal error rates (EER) as low as 11.4% (Keykhaie & Pierre, 2023).

Continuous authentication techniques distinguish themselves from traditional one-time logins by constantly monitoring user interactions. This approach is essential in sectors like finance and healthcare, where high security is paramount. Classifiers like one-class SVM, hidden Markov models (HMM), and deep learning architectures are employed to create baseline behavior models, detecting anomalies that may indicate unauthorized access. These systems capture diverse biometrics including keystroke dynamics, mouse movements, and touchscreen gestures, enabling a comprehensive analysis of user behavior. Support Vector Machines (SVM) are particularly effective for detecting subtle shifts in behavior, providing low false acceptance and rejection rates that enhance system reliability (Oduri, 2024).

To enhance classifier performance and differentiate genuine users from threats, distance metrics such as Euclidean, Manhattan, and Cosine are commonly used. Feature reduction methods, notably Principal Component Analysis (PCA) and LDA, further optimize system performance by reducing data dimensionality, with LDA achieving up to 94.3% accuracy when combined with Manhattan and Euclidean metrics. KNN

classifiers, assessing proximity to known behavior patterns, are also integral for real-time classification, making them effective in mobile environments where rapid authentication is needed (Dhaka et al., 2022).

In addressing adversarial attacks, where subtle data manipulations can trick classifiers into misclassification, researchers have introduced confidence thresholding as a defense mechanism. This technique only accepts predictions above a certain confidence level, mitigating adversarial impacts and preserving system robustness in practical applications (Hasan & Abdulazeez, 2021).

In summary, recent advancements in acceleration-based user authentication have leveraged adaptive, continuous monitoring and optimized classifiers, resulting in high-accuracy systems suited for mobile and wearable technology. These methods offer a seamless and secure user experience with minimal intervention, proving especially effective in applications where constant verification is critical (d'Aspremont et al., 2021).

03) Testing Methodology

To assess and optimize the performance of Convolutional Neural Network (CNN) models, a rigorous methodology was used, which included data preparation, training, parameter adjustment, and testing across several datasets. The study focused on binary classification problems with three types of features: frequency domain, time domain, and time-frequency domain. For each feature category, datasets were created for both single-day and multi-day scenarios. Statistical analysis was used to better comprehend feature distributions such as mean, standard deviation, skewness, and kurtosis. Principal Component Analysis (PCA) was used to reduce dimensionality and find the most significant features, with measures such as intra- and inter-class variance and Fisher scores supporting the process.

Datasets were carefully constructed to maintain uniformity among studies. The frequency domain dataset includes 360 samples with 43 characteristics, molded into a 7× 7 matrix. The time domain dataset had 360 samples with 88 features, which were molded into a 10×10 matrix. The time-frequency domain dataset included 360 samples and 131 features, reshaped into a 12×12 matrix. Each dataset was divided into 80% training and 20% testing data, with labels assigned to 324 positive and 36 negative samples. This split technique ensured a fair assessment of training and testing performance.

The CNN models were trained with input dimensions that corresponded to the reshaped matrices for each dataset. The initial parameter choices comprised a learning rate of 0.001, binary cross-entropy as the loss function, and the Adam optimizer, which was selected for its adaptable learning rate capabilities. Early halting was enabled to evaluate validation loss and accuracy while avoiding overfitting. The activation functions were ReLU for intermediate layers and softmax for the output layer. Training was carried out across 50 epochs, with mini-batch sizes selected to balance computational efficiency and gradient stability.

The testing phase assessed models' performance using metrics like mini-batch accuracy, validation accuracy, and loss values. Factors like learning rate, epoch counts, and reshaping dimensions were investigated to balance overfitting and training efficiency.

The results showed good training accuracy across all datasets, with validation accuracy peaking between 98.61% and 100%. After optimization, validation loss was greatly reduced, showing improved model generalization. Among the datasets, frequency domain features demonstrated the best stability and consistency, with Feature 40 being the most stable. PCA investigation demonstrated that dimensionality reduction could retain 95% variance with 6-23 components, depending on the dataset, allowing for computing economy without sacrificing accuracy.

The CNN design was chosen for its ability to reliably capture spatial patterns in matrix-reshaped data, as demonstrated by established neural network training best practices. The chosen optimizer and learning rate were supported by their popularity and dependability in similar tasks (Abdolrasol et al., 2021). Validation monitoring and early stopping were critical in preventing the models from overfitting despite extensive training.

The study optimized CNN performance through iterative parameter tweaking and feature importance, achieving high classification accuracy and robustness. Future research could explore advanced structures or attention techniques.

04)Evaluation

04.01) Analyzing Data Variability: Inter and Intra-User Variance

04.01.01) FreqD Analysis

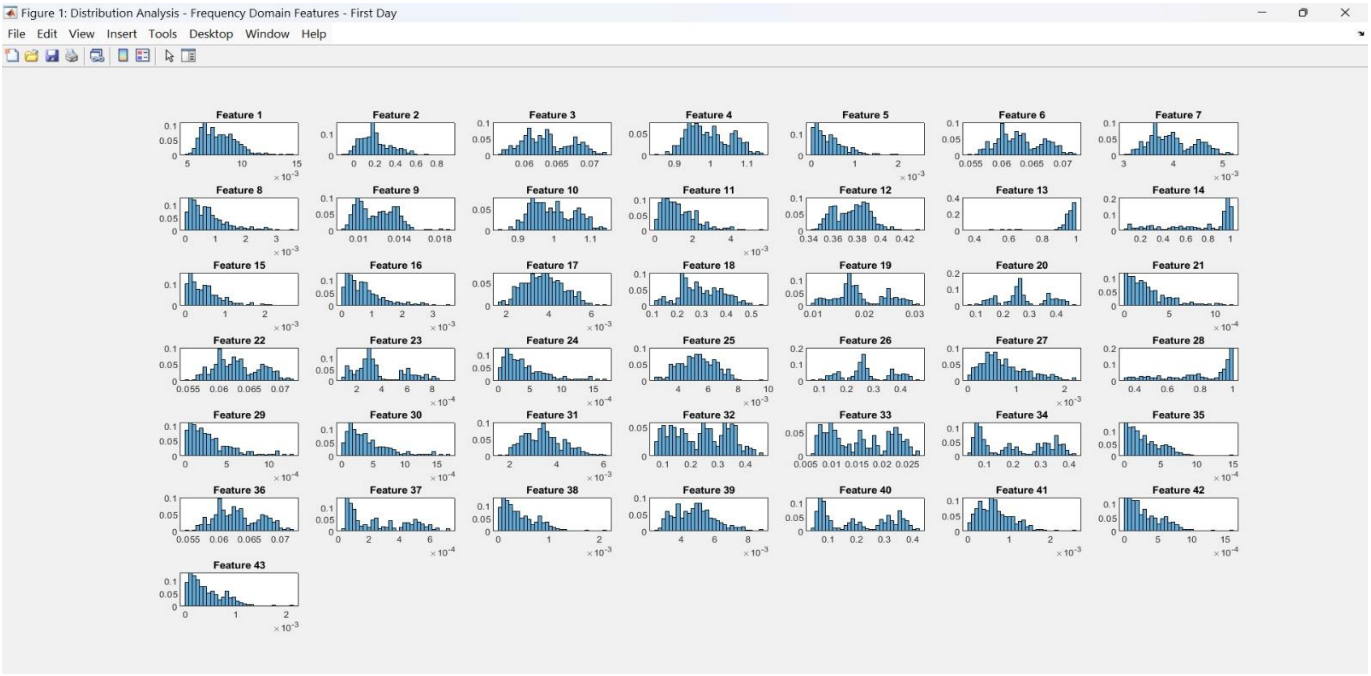

Figure 1: FreqD Features FDay

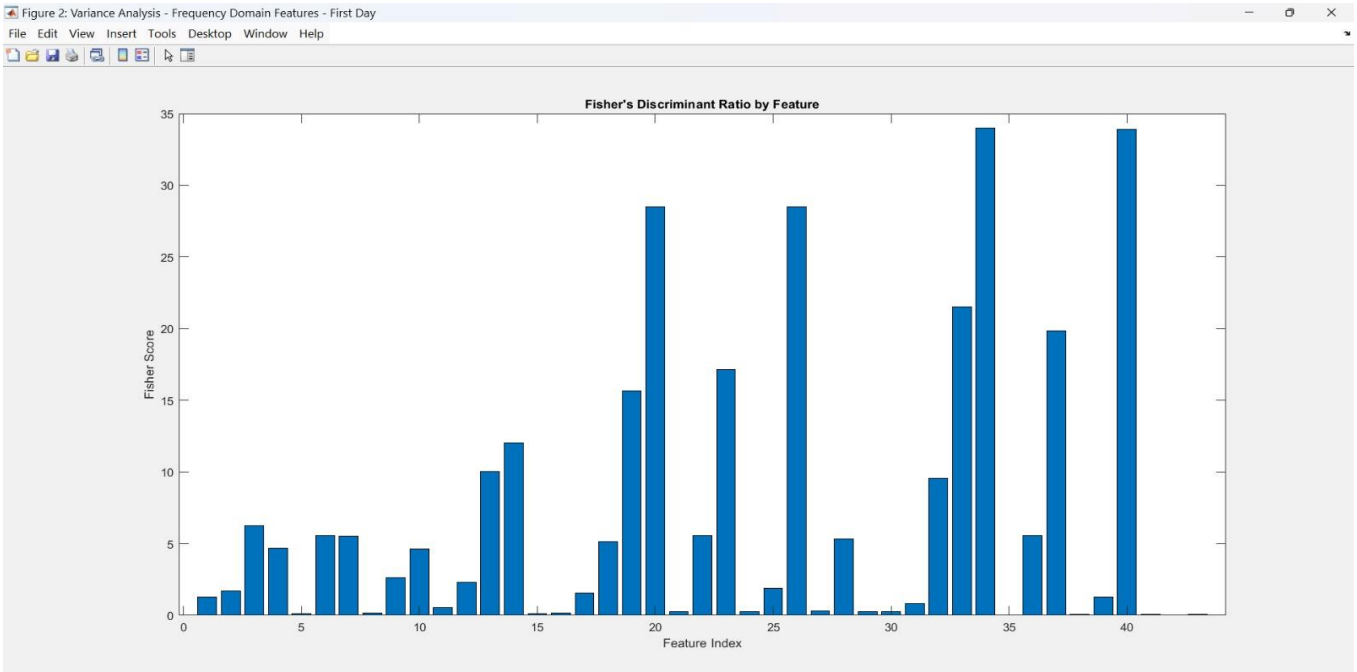

Figure 2 : FreqD Features FDay

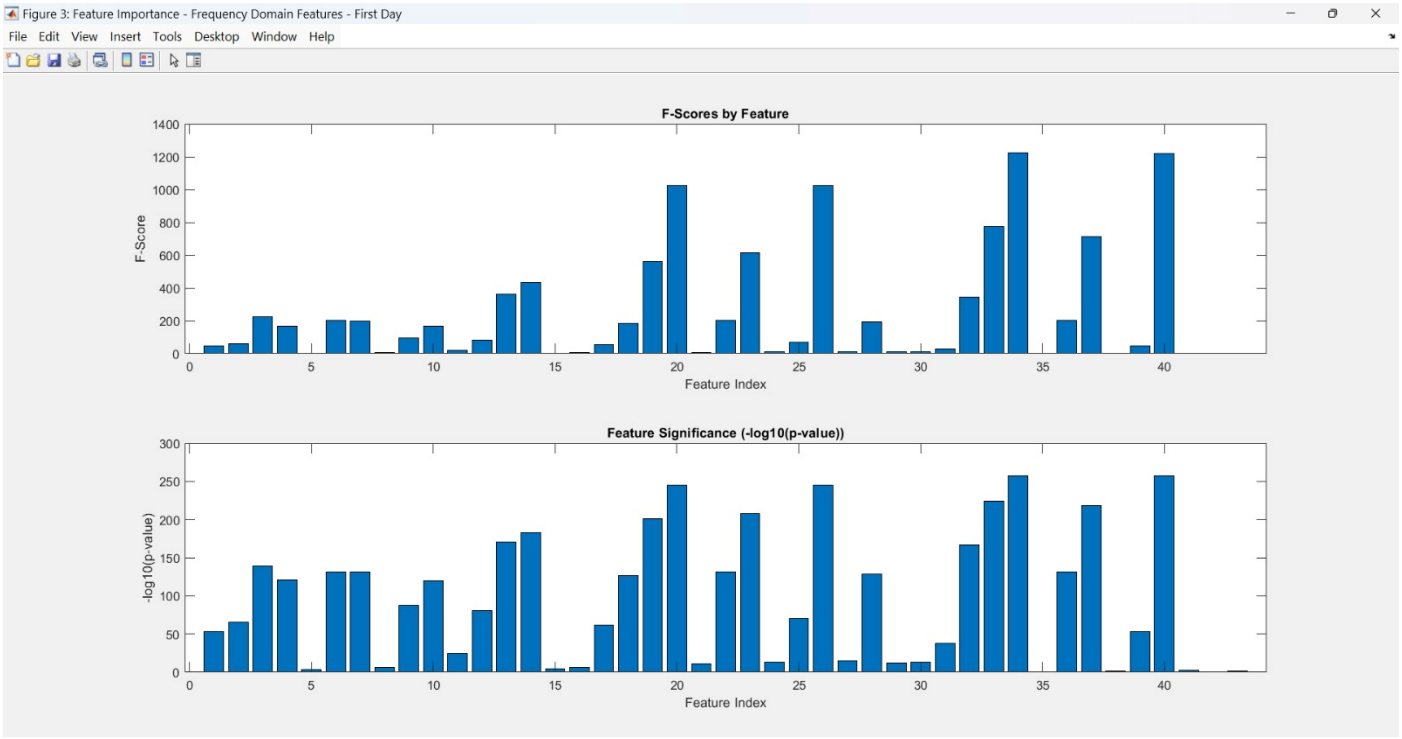

Figure 3 : FreqD Features FDay

Figure 4: Temporal Stability FreqD Features

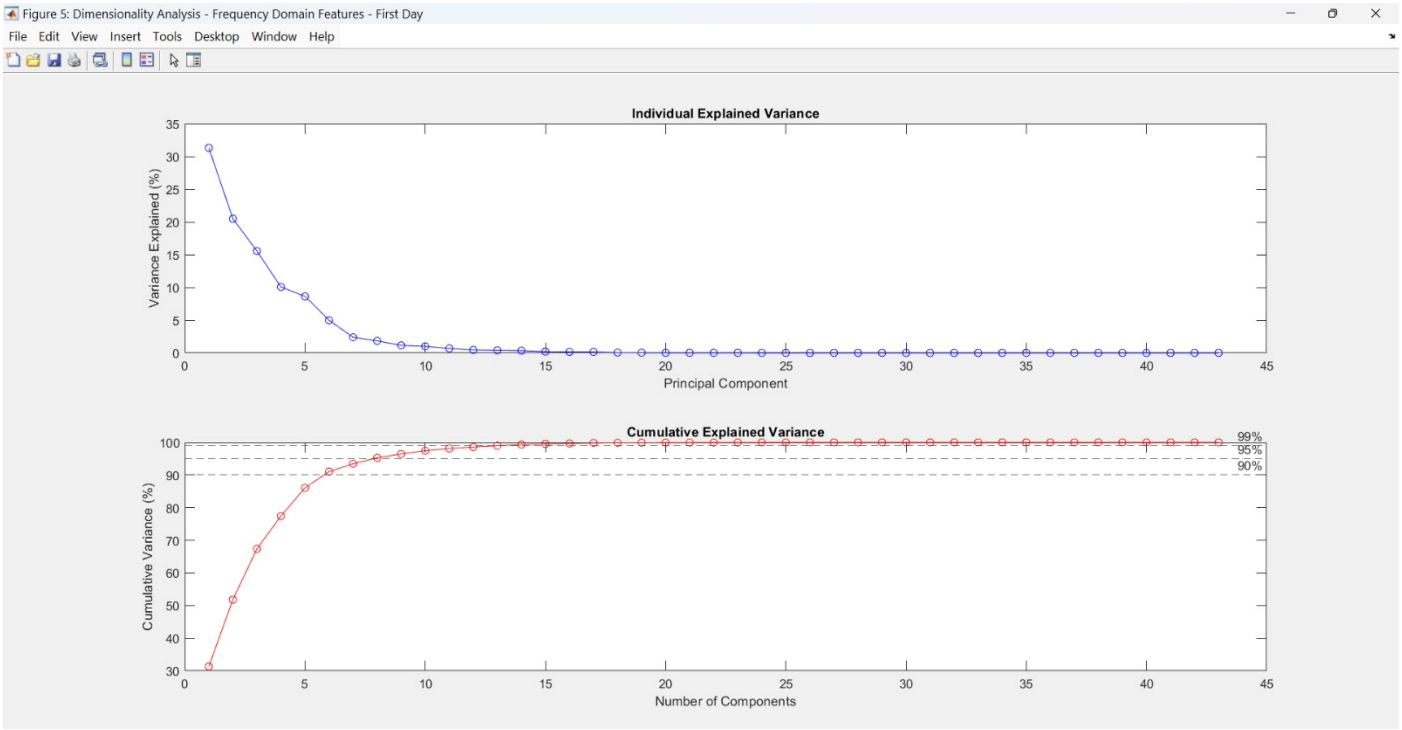

Figure 5 : Dimensionality Analysis FreqD Features FDay

The analysis presents various charts examining frequency domain features across multiple dimensions. The Explained Variance Chart reveals that the initial principal components account for over 90% of the cumulative variance, suggesting effective dimensionality reduction is achievable. The Temporal Stability Chart identifies several features, such as indices 15, 20, 35, and 40, exhibiting strong temporal correlations, with user 7 showing notable stability among users. The Feature Importance Chart highlights critical features with high F-scores and significant p-values, crucial for classification tasks. The Variance Analysis Chart, using Fisher's discriminant ratio, underscores features 20, 35, and 40 as highly discriminative. Lastly, the Distribution Analysis Chart demonstrates that many features possess skewed or distinct distributions, indicating their potential for effective class separation.

04.01.02) TimeD Analysis

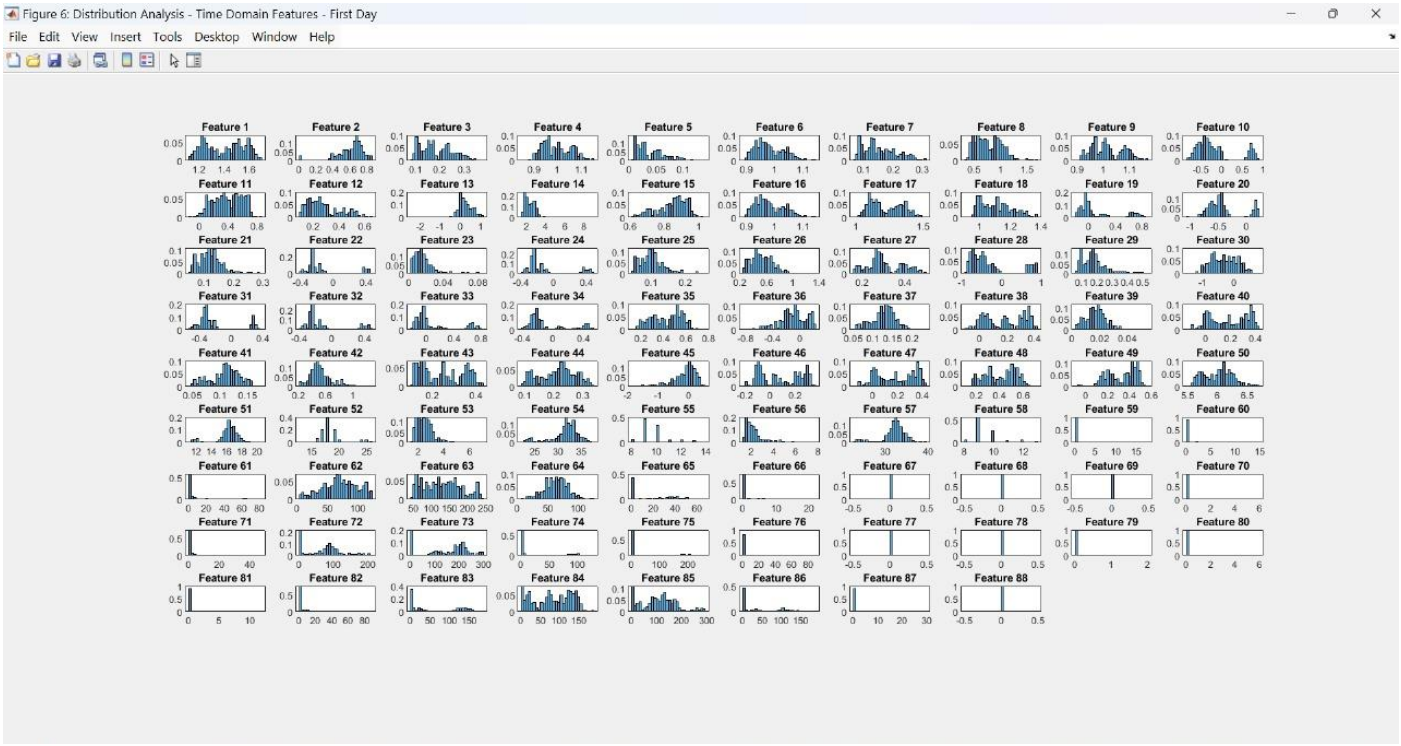

Figure 6 : Distribution Analysis TimeD Features FDay

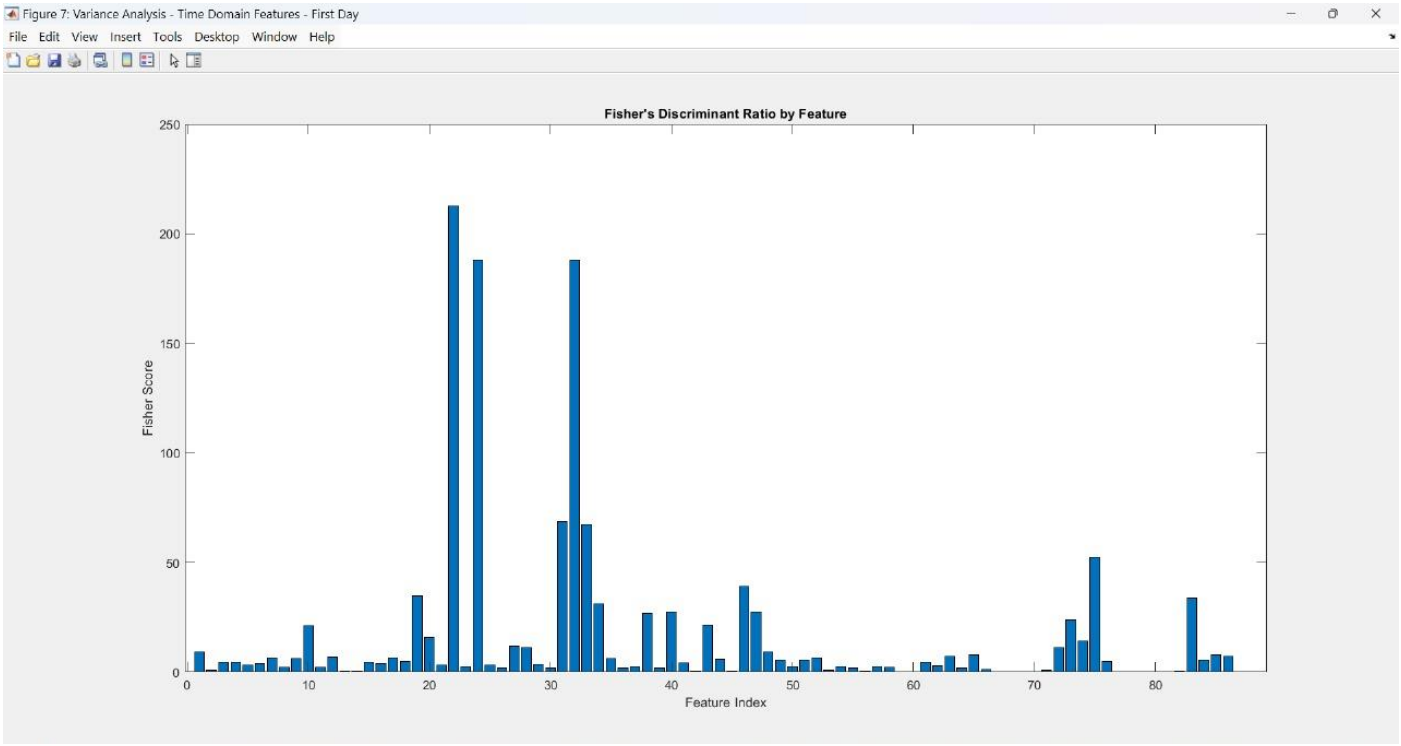

Figure 7 : Variance Analysis TimeD Features FDay

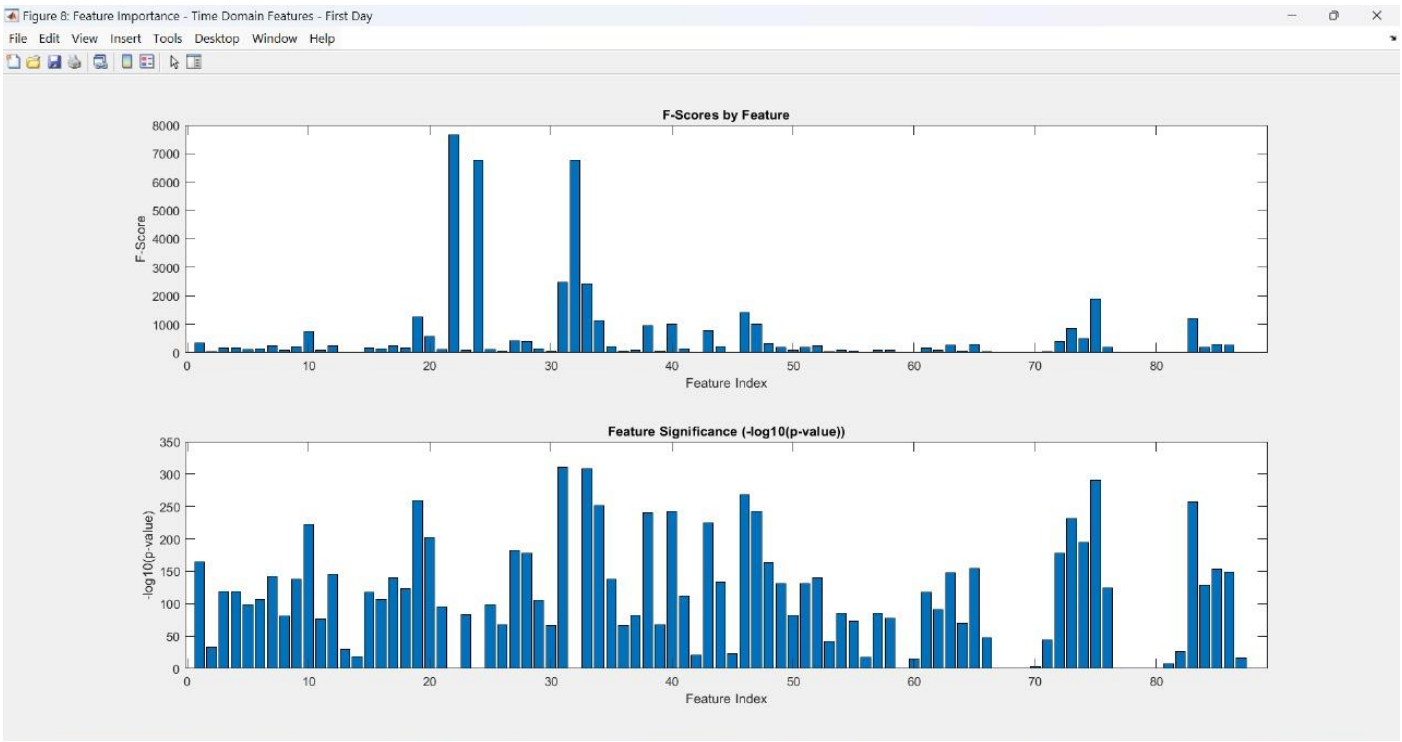

Figure 8 : Feature Importance TimeD FDay

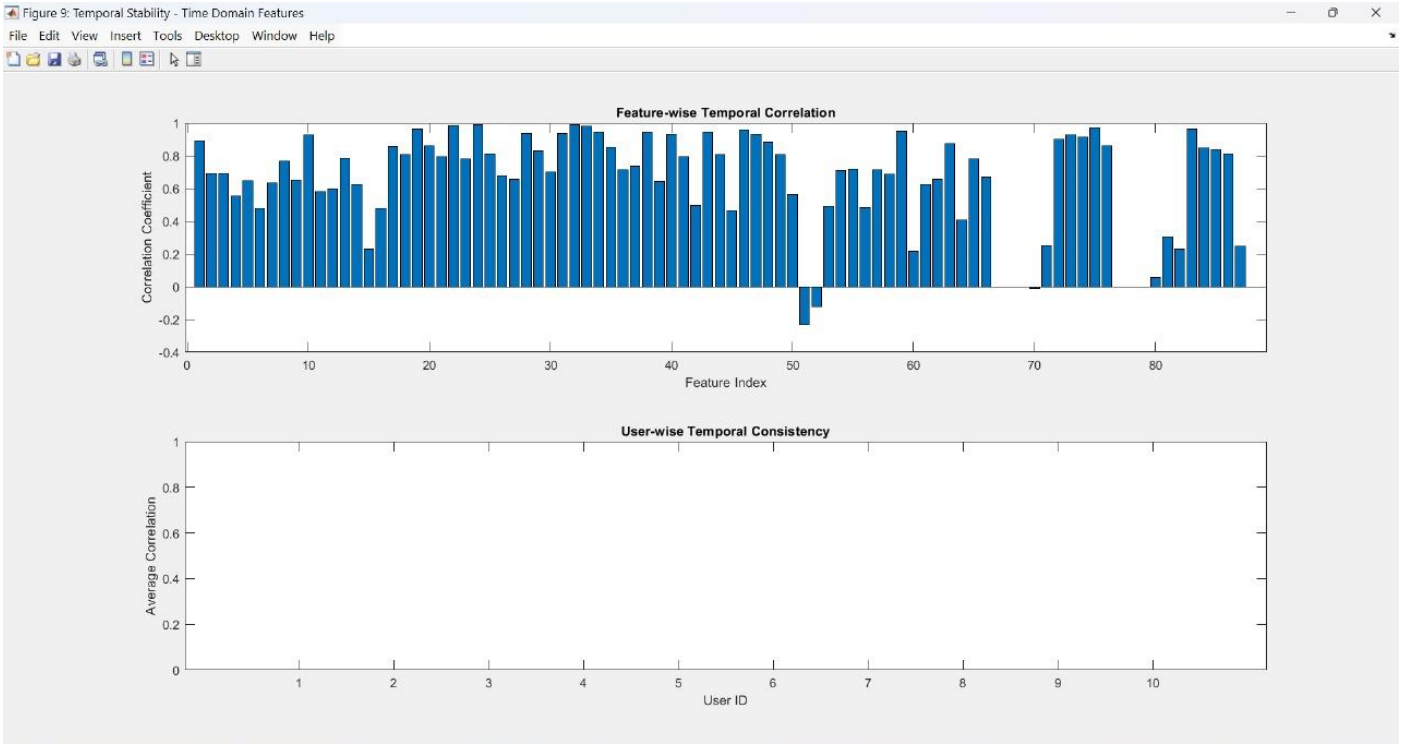

Figure 9 : Temporal Stability TimeD Features

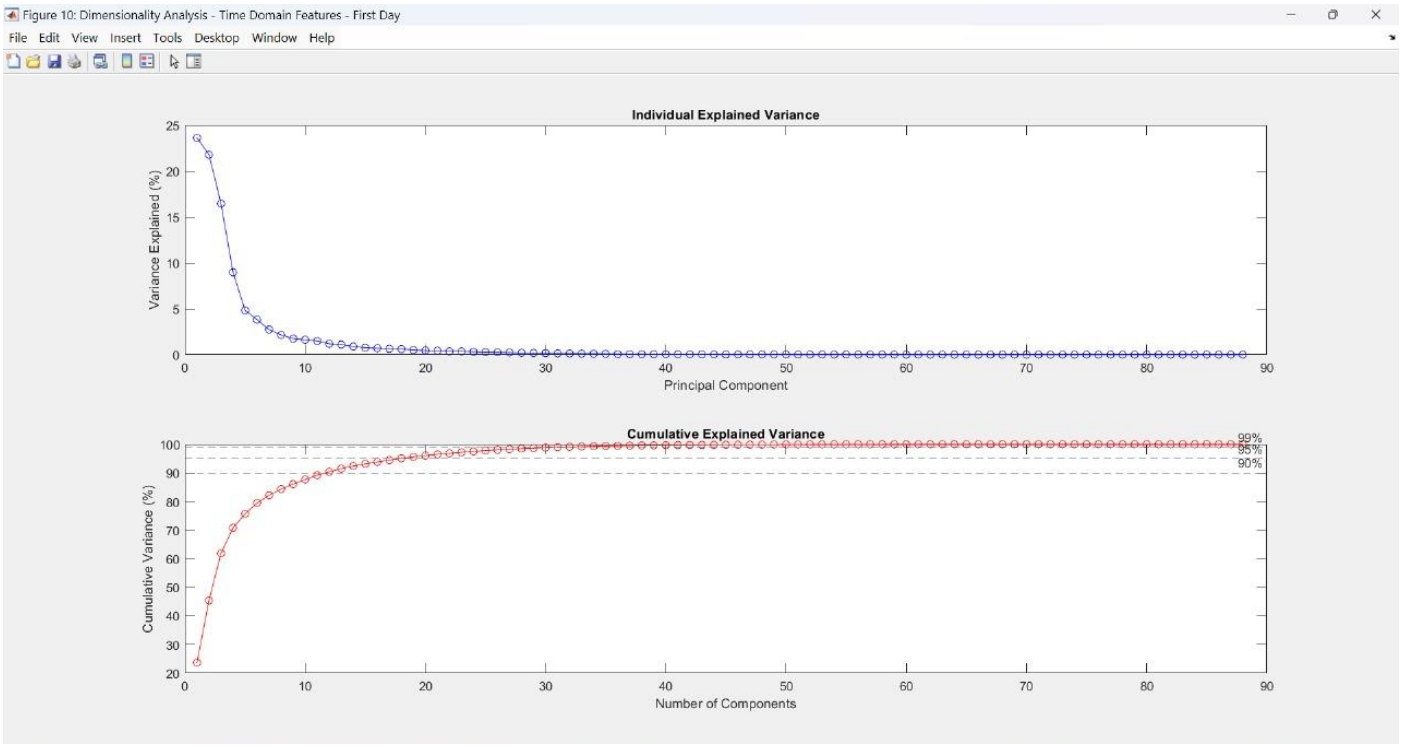

Figure 10 : Dimensionality Analysis TimeD Features FDay

The charts collectively analyze time-domain features for inter- and intra-variance. Distribution Analysis (Figure 6) shows diverse feature distributions, with some features (e.g., features 20, 30, and 40) having skewed or distinct profiles, reflecting variability across the dataset. Fisher's Discriminant Ratio (Figure 7) highlights features 20, 30, and 40 with the highest scores (>200), indicating they are key discriminators between groups. Feature Importance (Figure 8) corroborates this with these features showing the highest F-scores (up to 8000) and significant -log10(p-values), confirming their statistical relevance. Temporal Stability (Figure 9) reveals high intra-group consistency, with most features maintaining strong temporal correlations (>0.8), ensuring stability over time. Lastly, Dimensionality Analysis (Figure 10) demonstrates that the first 20 components explain over 95% of the cumulative variance, supporting dimensionality reduction while retaining essential information. These insights highlight critical features for classification and temporal consistency for reliable modeling.

04.01.03) Time FreqD Analysis

Figure 11: Distribution Analysis Time FreqD FDay

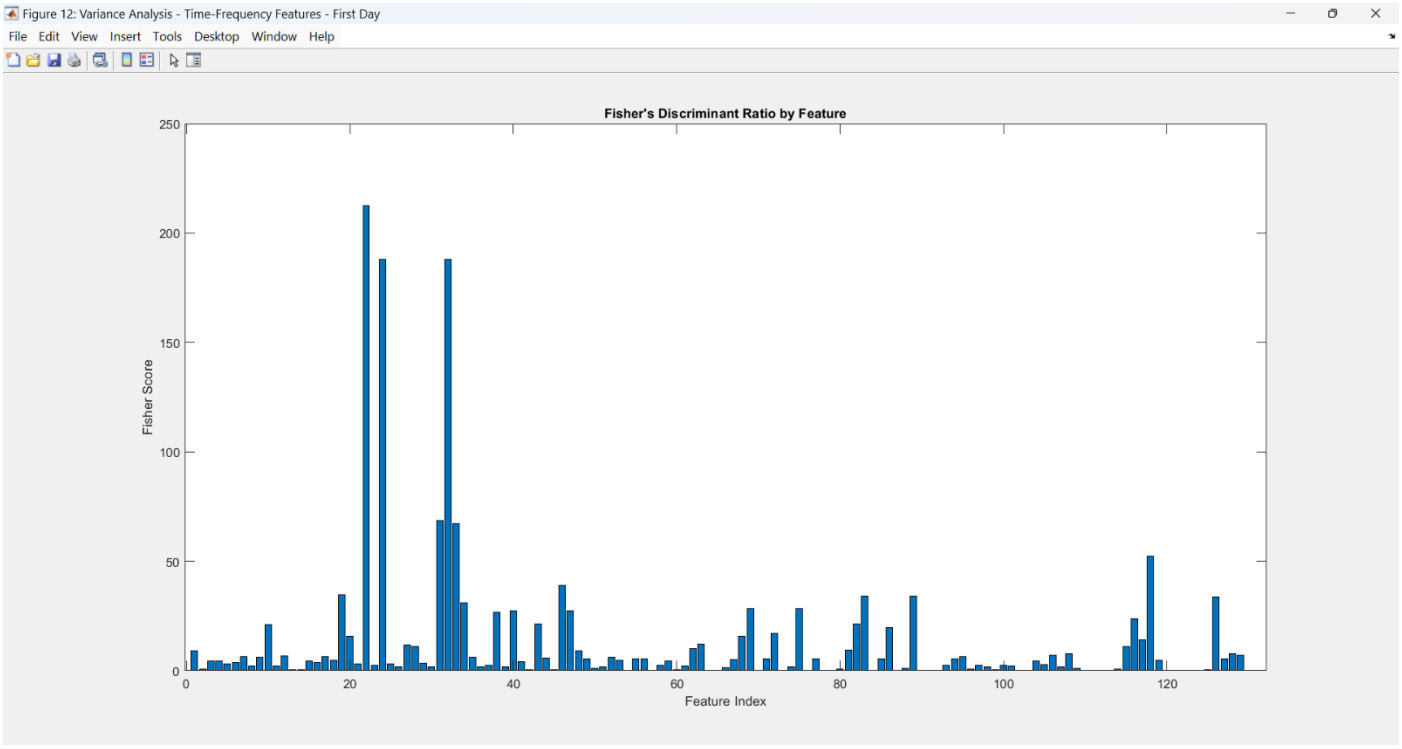

Figure 12 : Variance Analysis Time FreqD FDay

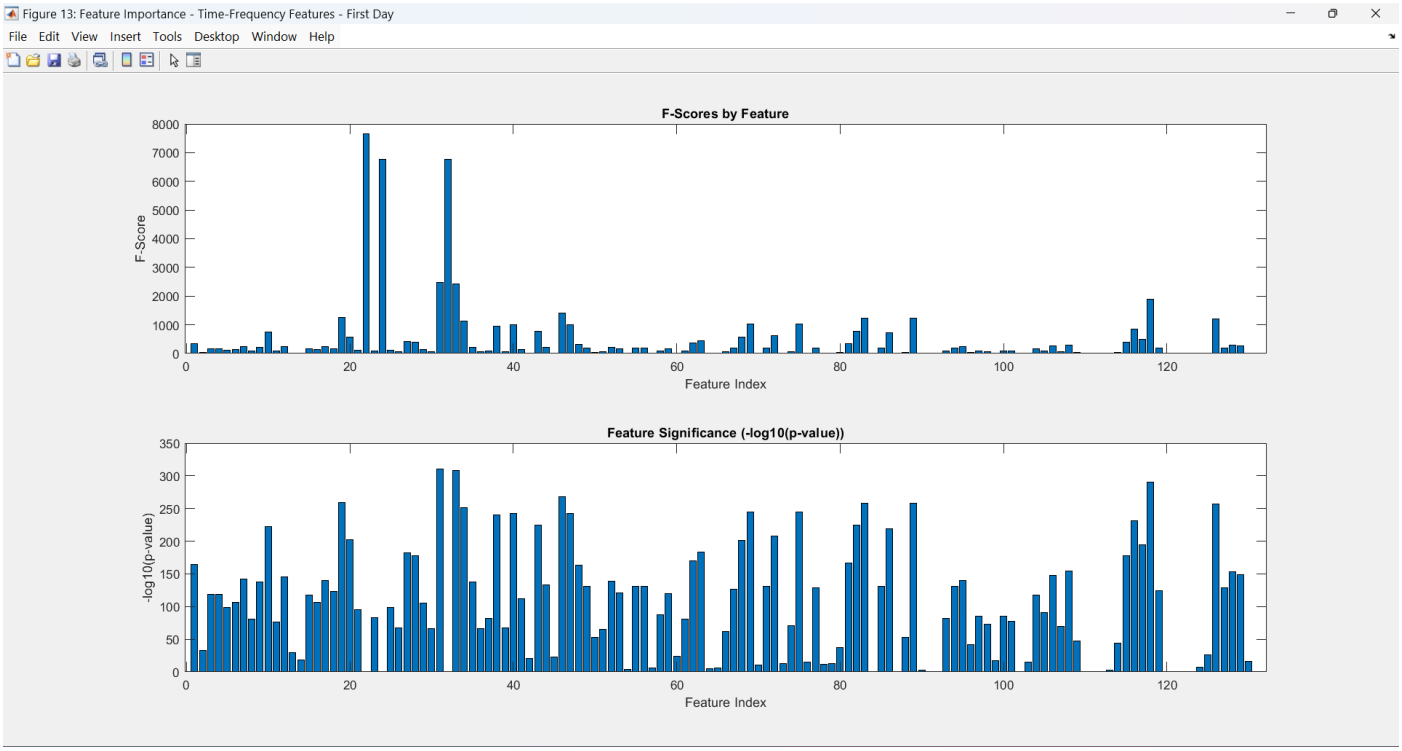

Figure 13 : Feature Importance Time FreqD Features FDay

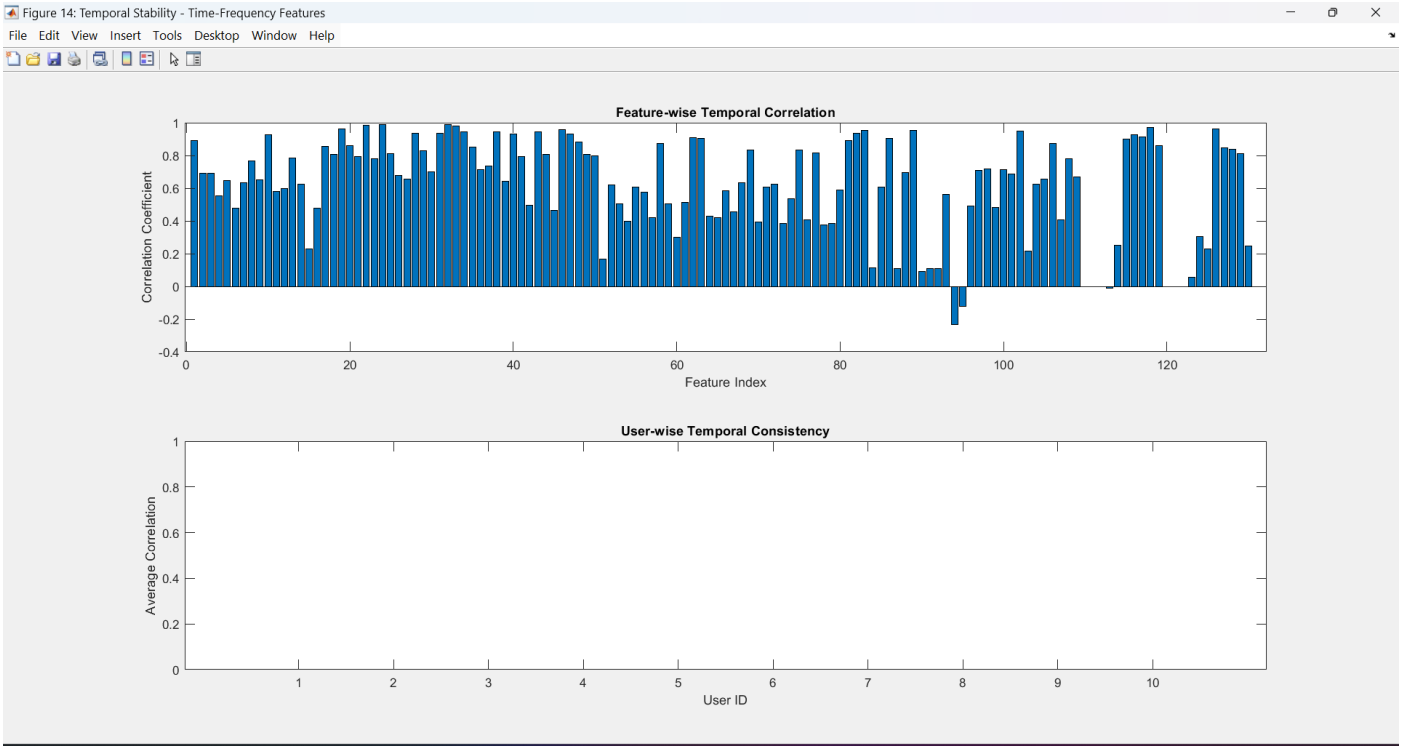

Figure 14 : Temporal Stability Time FreqD Features

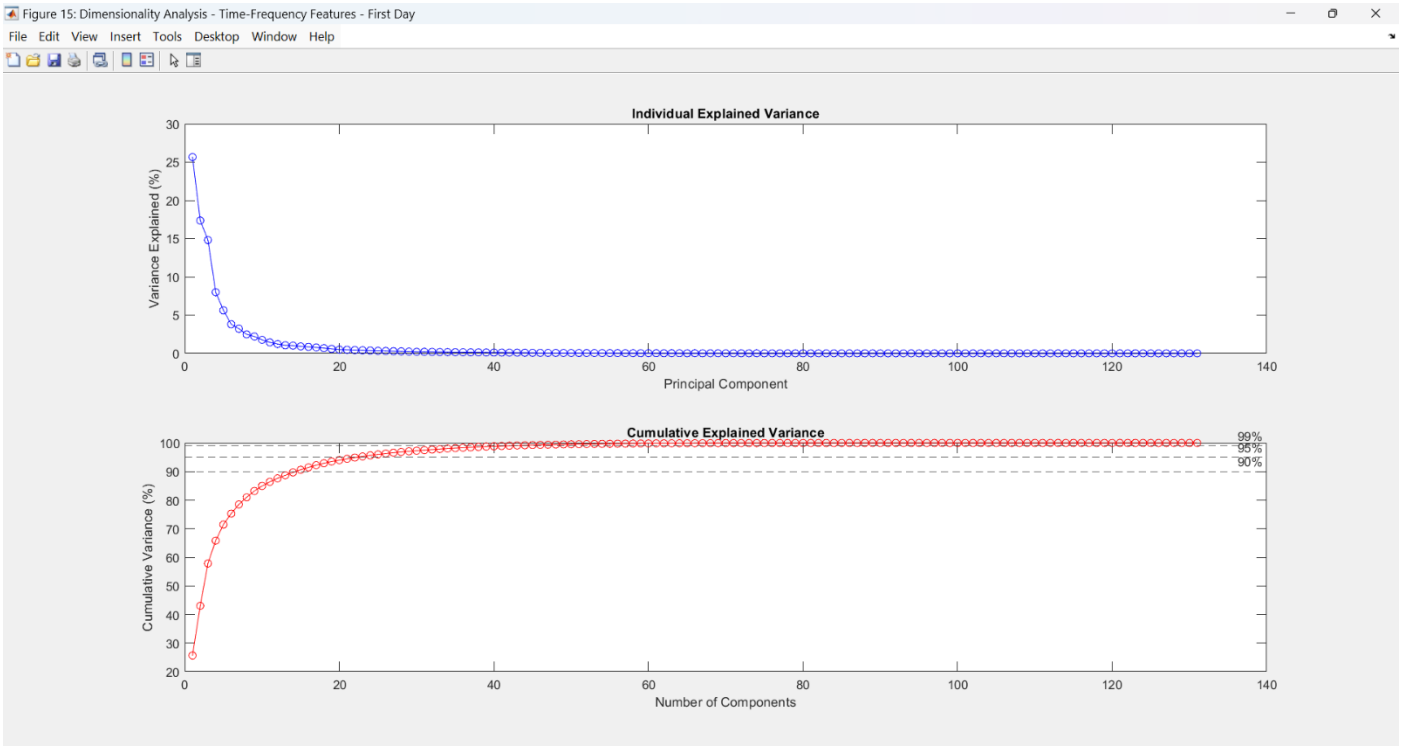

Figure 15 : Dimensionality Analysis Time FreqD Features FDay

The analysis of time-frequency domain features reveals significant insights into inter- and intra-variance. The Distribution Analysis indicates diverse and skewed patterns in features 20, 30, and 120, suggesting variability across groups. Fisher's Discriminant Ratio identifies these features as key discriminators, with scores reaching up to 250. Furthermore, Feature Importance demonstrates their exceptional significance, with F-scores up to 8000 and highly significant p-values. Temporal Stability illustrates strong intra-group reliability, with most features exhibiting correlations above 0.8. Lastly, Dimensionality Analysis confirms that the first 30 principal components account for over 95% of the cumulative variance, effectively preserving critical information while streamlining the model. Overall, these findings underscore the importance of specific features for effective modeling and classification.

04.02) Pre-Processing the Data Before Training Process

Data reshaping is an essential step while preparing data for a machine learning model. For frequency domain features, the reshaped matrices are 7x7 and 10x10 with 6 padding, for time domain features it is 12x12 and 13 padding, and for time-frequency domain features it is also 12x12 and 13 padding.

For the binary classification task, the data were labeled into two classes: positive and negative. There are 324 samples in the positive class and 36 samples in the negative class, which gives a distribution of binary labels equal to [324, 36].

The shaped and labeled data is split into training and testing sets by an 80-20 split to prepare the data for model training and evaluation. This grants most of the data, 288 samples, to be allocated for training to allow the model to learn patterns effectively, while the remaining 72 samples are reserved for testing. Thus, this approach helps assess the model's performance on unseen data, which ensures a robust evaluation of its generalization capabilities.

Once the reshaping, labeling, and splitting tasks were completed for each domain - Frequency Domain, Time Domain, and Time-Frequency Domain - we finalized the datasets for the "First Day" and "Multi-Day" periods.

Before the training process, specify the authorized user ID must be in the range of 1,2,3,4,5,6,7,8,9,10. For this instance, user 1 is assumed to be the requested authorized user. The subsequent training process is user 1.

04.02.01) Training Process

The binary classification model's training began with the **Frequency Domain - First Day** dataset, with a network input size of [7 7 1], corresponding to the reshaped 7x7 matrix for each sample. The Convolutional Neural Network (CNN) was trained to distinguish positive and negative classes by learning frequency domain features.

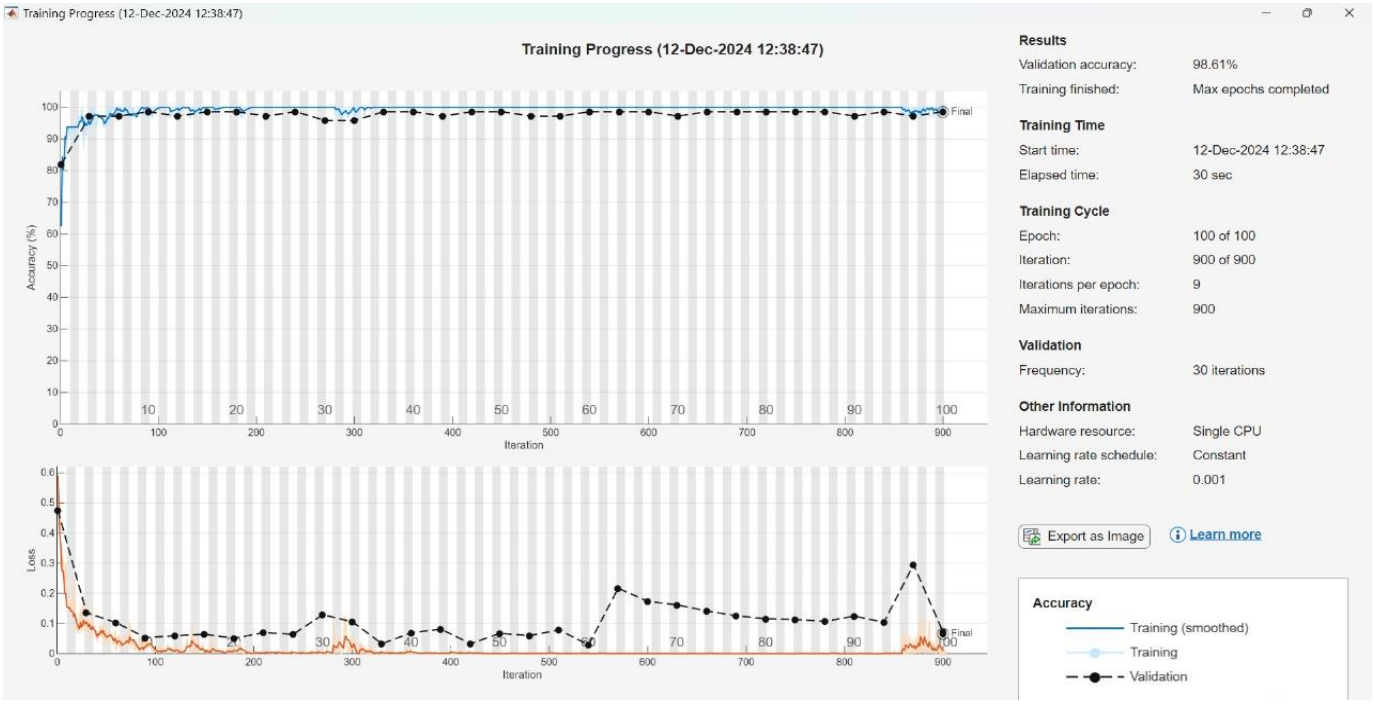

Figure 16 : Result of the Training FreqD FDay

Table 1 : Table of the FreqD FDay Training Process

Command Window

```
Initializing input data normalization.
|=========================================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |  Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |      Rate       |
|=========================================================================================================|
|       1 |           1 |       00:00:04 |      62.50% |      81.94% |       0.5917 |       0.4753 |          0.0010 |
|       4 |          30 |       00:00:06 |      93.75% |      97.22% |       0.1175 |       0.1359 |          0.0010 |
|       6 |          50 |       00:00:07 |     100.00% |             |       0.0443 |              |          0.0010 |
|       7 |          60 |       00:00:07 |     100.00% |      97.22% |       0.0141 |       0.1031 |          0.0010 |
|      10 |          90 |       00:00:08 |      96.88% |      98.61% |       0.0381 |       0.0531 |          0.0010 |
|      12 |         100 |       00:00:08 |     100.00% |             |       0.0075 |              |          0.0010 |
|      14 |         120 |       00:00:09 |     100.00% |      97.22% |       0.0015 |       0.0596 |          0.0010 |
|      17 |         150 |       00:00:10 |     100.00% |      98.61% |       0.0092 |       0.0643 |          0.0010 |
|      20 |         180 |       00:00:10 |     100.00% |      98.61% |       0.0239 |       0.0511 |          0.0010 |
|      23 |         200 |       00:00:11 |     100.00% |             |       0.0048 |              |          0.0010 |
|      24 |         210 |       00:00:11 |     100.00% |      97.22% |       0.0013 |       0.0696 |          0.0010 |
|      27 |         240 |       00:00:12 |     100.00% |      98.61% |       0.0022 |       0.0645 |          0.0010 |
|      28 |         250 |       00:00:12 |     100.00% |             |       0.0013 |              |          0.0010 |
|      30 |         270 |       00:00:13 |     100.00% |      95.83% |       0.0058 |       0.1292 |          0.0010 |
|      34 |         300 |       00:00:14 |      96.88% |      95.83% |       0.0269 |       0.1057 |          0.0010 |
|      37 |         330 |       00:00:15 |     100.00% |      98.61% |       0.0036 |       0.0331 |          0.0010 |
|      39 |         350 |       00:00:15 |     100.00% |             |       0.0050 |              |          0.0010 |
|      40 |         360 |       00:00:15 |     100.00% |      98.61% |       0.0005 |       0.0692 |          0.0010 |
|      44 |         390 |       00:00:16 |     100.00% |      97.22% |       0.0005 |       0.0815 |          0.0010 |
|      45 |         400 |       00:00:17 |     100.00% |             |       0.0173 |              |          0.0010 |
|      47 |         420 |       00:00:17 |     100.00% |      98.61% |       0.0014 |       0.0331 |          0.0010 |
|      50 |         450 |       00:00:18 |     100.00% |      98.61% |       0.0009 |       0.0674 |          0.0010 |
|      54 |         480 |       00:00:19 |     100.00% |      97.22% |       0.0001 |       0.0598 |          0.0010 |
|      56 |         500 |       00:00:19 |     100.00% |             |       0.0002 |              |          0.0010 |
|      57 |         510 |       00:00:20 |     100.00% |      97.22% |       0.0004 |       0.0790 |          0.0010 |
|      60 |         540 |       00:00:20 |     100.00% |      98.61% |       0.0003 |       0.0289 |          0.0010 |
|      62 |         550 |       00:00:21 |     100.00% |             |       0.0001 |              |          0.0010 |
|      64 |         570 |       00:00:21 |     100.00% |      98.61% |   4.4227e-05 |       0.2170 |          0.0010 |
|      67 |         600 |       00:00:22 |     100.00% |      98.61% |       0.0012 |       0.1737 |          0.0010 |
|      70 |         630 |       00:00:23 |     100.00% |      97.22% |       0.0003 |       0.1616 |          0.0010 |
|      73 |         650 |       00:00:23 |     100.00% |             |       0.0001 |              |          0.0010 |
|      74 |         660 |       00:00:24 |     100.00% |      98.61% |   7.3681e-05 |       0.1417 |          0.0010 |
|      77 |         690 |       00:00:25 |     100.00% |      98.61% |   9.6578e-05 |       0.1256 |          0.0010 |
|      78 |         700 |       00:00:25 |     100.00% |             |       0.0002 |              |          0.0010 |
|      80 |         720 |       00:00:25 |     100.00% |      98.61% |   9.3450e-05 |       0.1157 |          0.0010 |
|      84 |         750 |       00:00:26 |     100.00% |      98.61% |   3.4903e-05 |       0.1135 |          0.0010 |
|      87 |         780 |       00:00:27 |     100.00% |      98.61% |       0.0002 |       0.1071 |          0.0010 |
|      89 |         800 |       00:00:27 |     100.00% |             |   6.8528e-05 |              |          0.0010 |
|      90 |         810 |       00:00:28 |     100.00% |      97.22% |       0.0005 |       0.1246 |          0.0010 |
|      94 |         840 |       00:00:29 |     100.00% |      98.61% |   5.8427e-05 |       0.1041 |          0.0010 |
|      95 |         850 |       00:00:29 |     100.00% |             |       0.0065 |              |          0.0010 |
|      97 |         870 |       00:00:29 |      96.88% |      97.22% |       0.0473 |       0.2953 |          0.0010 |
|     100 |         900 |       00:00:30 |     100.00% |      98.61% |       0.0048 |       0.0715 |          0.0010 |
|=========================================================================================================|
Training finished: Max epochs completed.
[2024-12-12 12:39:20] Training completed in 0.70 minutes
```

The **Time Domain - First Day** dataset has undergone binary training with a network input size of [10 10 1], corresponding to a 10x10 matrix for each sample. The Convolutional Neural Network (CNN) is trained to differentiate between positive and negative classes based on temporal patterns.

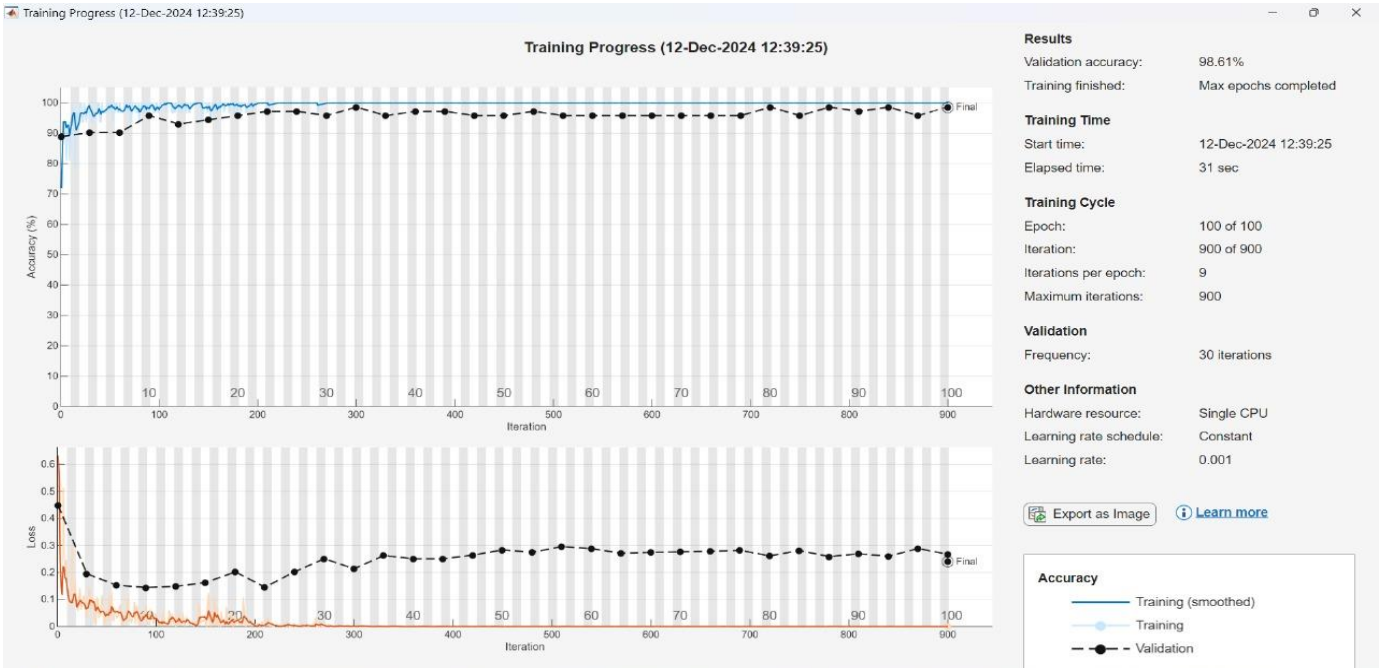

Figure 17 : Result of the Training TimeD FDay

Table 2 : Table of the TimeD FDay Training Process

Command Window

```
Initializing input data normalization.
|=========================================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |  Base Learning  |
|         |             |    (hh:mm:ss)  |   Accuracy   |   Accuracy   |     Loss     |     Loss     |      Rate       |
|=========================================================================================================|
|       1 |           1 |       00:00:02 |       71.88% |       88.89% |       0.6326 |       0.4484 |          0.0010 |
|       4 |          30 |       00:00:04 |      100.00% |       90.28% |       0.0315 |       0.1951 |          0.0010 |
|       6 |          50 |       00:00:04 |       96.88% |              |       0.0622 |              |          0.0010 |
|       7 |          60 |       00:00:05 |      100.00% |       90.28% |       0.0312 |       0.1530 |          0.0010 |
|      10 |          90 |       00:00:05 |       96.88% |       95.83% |       0.0764 |       0.1445 |          0.0010 |
|      12 |         100 |       00:00:06 |       96.88% |              |       0.0372 |              |          0.0010 |
|      14 |         120 |       00:00:07 |      100.00% |       93.06% |       0.0007 |       0.1486 |          0.0010 |
|      17 |         150 |       00:00:08 |      100.00% |       94.44% |       0.0058 |       0.1630 |          0.0010 |
|      20 |         180 |       00:00:09 |       96.88% |       95.83% |       0.0436 |       0.2020 |          0.0010 |
|      23 |         200 |       00:00:09 |      100.00% |              |       0.0069 |              |          0.0010 |
|      24 |         210 |       00:00:10 |      100.00% |       97.22% |       0.0028 |       0.1456 |          0.0010 |
|      27 |         240 |       00:00:11 |      100.00% |       97.22% |       0.0038 |       0.2029 |          0.0010 |
|      28 |         250 |       00:00:11 |      100.00% |              |       0.0005 |              |          0.0010 |
|      30 |         270 |       00:00:12 |      100.00% |       95.83% |       0.0047 |       0.2510 |          0.0010 |
|      34 |         300 |       00:00:13 |      100.00% |       98.61% |       0.0001 |       0.2138 |          0.0010 |
|      37 |         330 |       00:00:14 |      100.00% |       95.83% |       0.0005 |       0.2629 |          0.0010 |
|      39 |         350 |       00:00:14 |      100.00% |              |       0.0003 |              |          0.0010 |
|      40 |         360 |       00:00:15 |      100.00% |       97.22% |       0.0007 |       0.2509 |          0.0010 |
|      44 |         390 |       00:00:16 |      100.00% |       97.22% |       0.0002 |       0.2510 |          0.0010 |
|      45 |         400 |       00:00:16 |      100.00% |              |       0.0002 |              |          0.0010 |
|      47 |         420 |       00:00:17 |      100.00% |       95.83% |       0.0003 |       0.2642 |          0.0010 |
|      50 |         450 |       00:00:17 |      100.00% |       95.83% |       0.0005 |       0.2838 |          0.0010 |
|      54 |         480 |       00:00:18 |      100.00% |       97.22% |   1.1581e-05 |       0.2748 |          0.0010 |
|      56 |         500 |       00:00:19 |      100.00% |              |       0.0002 |              |          0.0010 |
|      57 |         510 |       00:00:19 |      100.00% |       95.83% |       0.0011 |       0.2960 |          0.0010 |
|      60 |         540 |       00:00:20 |      100.00% |       95.83% |   2.3124e-05 |       0.2883 |          0.0010 |
|      62 |         550 |       00:00:20 |      100.00% |              |       0.0006 |              |          0.0010 |
|      64 |         570 |       00:00:21 |      100.00% |       95.83% |       0.0001 |       0.2713 |          0.0010 |
|      67 |         600 |       00:00:22 |      100.00% |       95.83% |   3.0780e-05 |       0.2747 |          0.0010 |
|      70 |         630 |       00:00:23 |      100.00% |       95.83% |       0.0001 |       0.2766 |          0.0010 |
|      73 |         650 |       00:00:24 |      100.00% |              |   6.4443e-05 |              |          0.0010 |
|      74 |         660 |       00:00:24 |      100.00% |       95.83% |   5.3605e-06 |       0.2786 |          0.0010 |
|      77 |         690 |       00:00:25 |      100.00% |       95.83% |       0.0003 |       0.2820 |          0.0010 |
|      78 |         700 |       00:00:25 |      100.00% |              |   4.7686e-05 |              |          0.0010 |
|      80 |         720 |       00:00:26 |      100.00% |       98.61% |       0.0003 |       0.2617 |          0.0010 |
|      84 |         750 |       00:00:26 |      100.00% |       95.83% |   1.0821e-05 |       0.2815 |          0.0010 |
|      87 |         780 |       00:00:27 |      100.00% |       98.61% |   2.3998e-05 |       0.2589 |          0.0010 |
|      89 |         800 |       00:00:28 |      100.00% |              |       0.0002 |              |          0.0010 |
|      90 |         810 |       00:00:28 |      100.00% |       97.22% |       0.0003 |       0.2694 |          0.0010 |
|      94 |         840 |       00:00:29 |      100.00% |       98.61% |   1.0918e-05 |       0.2601 |          0.0010 |
|      95 |         850 |       00:00:29 |      100.00% |              |   8.5664e-05 |              |          0.0010 |
|      97 |         870 |       00:00:30 |      100.00% |       95.83% |   8.3247e-05 |       0.2888 |          0.0010 |
|     100 |         900 |       00:00:31 |      100.00% |       98.61% |   1.0937e-05 |       0.2676 |          0.0010 |
|=========================================================================================================|
Training finished: Max epochs completed.
[2024-12-12 12:39:57] Training completed in 0.62 minutes
```

The Time-Frequency Domain - First Day dataset underwent binary training with a network input size of [12 12 1], corresponding to a reshaped 12x12 matrix for each sample. This initiated the Convolutional Neural Network (CNN) training process, where the model learned to distinguish positive and negative classes.

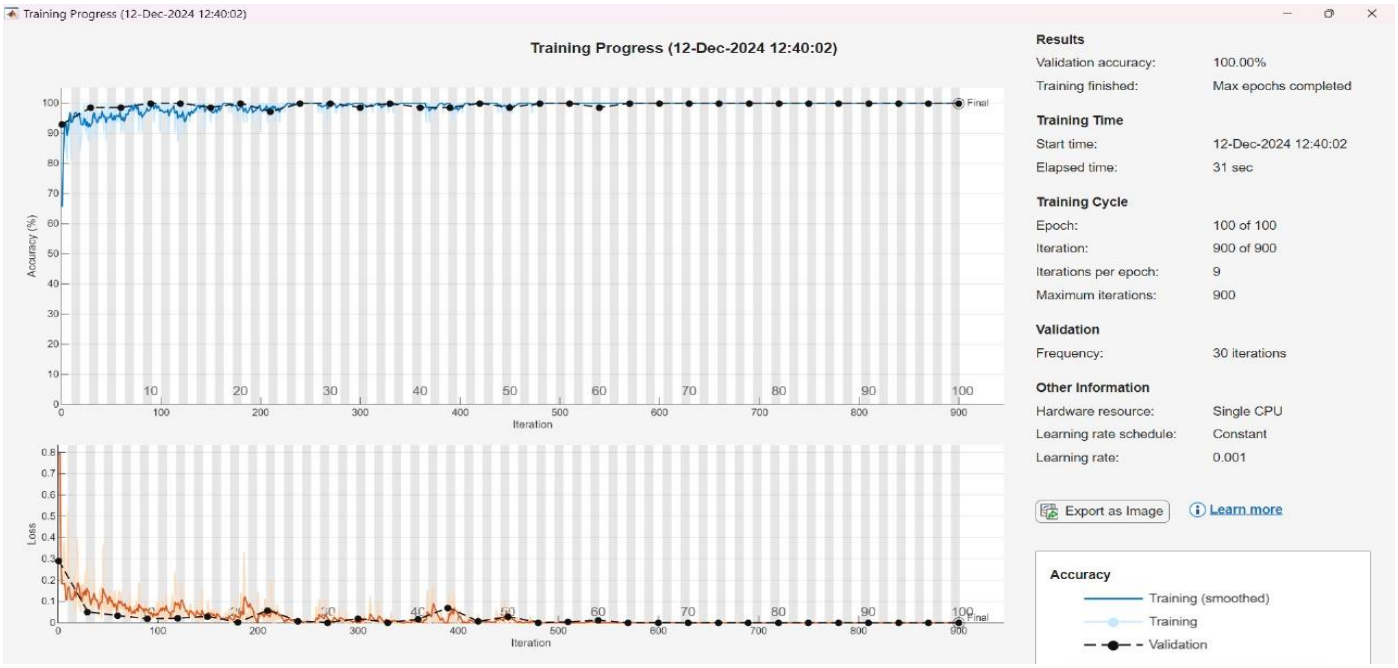

Figure 18 : Result of the Time FreqD FDay

Table 3 : Table of the Time FreqD FDay Training Process

Command Window

```
Initializing input data normalization.
|=========================================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |  Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |      Rate       |
|=========================================================================================================|
|       1 |           1 |       00:00:02 |      65.62% |      93.06% |       0.7878 |       0.2915 |          0.0010 |
|       4 |          30 |       00:00:03 |      93.75% |      98.61% |       0.1234 |       0.0512 |          0.0010 |
|       6 |          50 |       00:00:04 |     100.00% |             |       0.0465 |              |          0.0010 |
|       7 |          60 |       00:00:04 |      90.62% |      98.61% |       0.1188 |       0.0340 |          0.0010 |
|      10 |          90 |       00:00:05 |     100.00% |     100.00% |       0.0507 |       0.0192 |          0.0010 |
|      12 |         100 |       00:00:05 |     100.00% |             |       0.0153 |              |          0.0010 |
|      14 |         120 |       00:00:06 |      93.75% |     100.00% |       0.0847 |       0.0221 |          0.0010 |
|      17 |         150 |       00:00:07 |      93.75% |      98.61% |       0.1045 |       0.0300 |          0.0010 |
|      20 |         180 |       00:00:08 |     100.00% |     100.00% |       0.0419 |       0.0026 |          0.0010 |
|      23 |         200 |       00:00:09 |      90.62% |             |       0.1427 |              |          0.0010 |
|      24 |         210 |       00:00:09 |     100.00% |      97.22% |       0.0175 |       0.0594 |          0.0010 |
|      27 |         240 |       00:00:10 |     100.00% |     100.00% |       0.0264 |       0.0087 |          0.0010 |
|      28 |         250 |       00:00:10 |     100.00% |             |       0.0083 |              |          0.0010 |
|      30 |         270 |       00:00:11 |     100.00% |     100.00% |       0.0088 |       0.0022 |          0.0010 |
|      34 |         300 |       00:00:12 |     100.00% |      98.61% |       0.0024 |       0.0198 |          0.0010 |
|      37 |         330 |       00:00:13 |     100.00% |     100.00% |       0.0163 |       0.0018 |          0.0010 |
|      39 |         350 |       00:00:13 |     100.00% |             |       0.0005 |              |          0.0010 |
|      40 |         360 |       00:00:14 |     100.00% |      98.61% |       0.0009 |       0.0170 |          0.0010 |
|      44 |         390 |       00:00:15 |     100.00% |      98.61% |       0.0150 |       0.0709 |          0.0010 |
|      45 |         400 |       00:00:15 |     100.00% |             |       0.0008 |              |          0.0010 |
|      47 |         420 |       00:00:16 |     100.00% |     100.00% |       0.0402 |       0.0072 |          0.0010 |
|      50 |         450 |       00:00:17 |     100.00% |      98.61% |       0.0008 |       0.0294 |          0.0010 |
|      54 |         480 |       00:00:17 |     100.00% |     100.00% |       0.0016 |       0.0018 |          0.0010 |
|      56 |         500 |       00:00:18 |     100.00% |             |       0.0042 |              |          0.0010 |
|      57 |         510 |       00:00:18 |     100.00% |     100.00% |       0.0017 |       0.0048 |          0.0010 |
|      60 |         540 |       00:00:19 |     100.00% |      98.61% |       0.0003 |       0.0125 |          0.0010 |
|      62 |         550 |       00:00:20 |     100.00% |             |   6.6628e-05 |              |          0.0010 |
|      64 |         570 |       00:00:20 |     100.00% |     100.00% |       0.0007 |       0.0005 |          0.0010 |
|      67 |         600 |       00:00:21 |     100.00% |     100.00% |       0.0004 |       0.0013 |          0.0010 |
|      70 |         630 |       00:00:22 |     100.00% |     100.00% |       0.0002 |       0.0003 |          0.0010 |
|      73 |         650 |       00:00:23 |     100.00% |             |       0.0001 |              |          0.0010 |
|      74 |         660 |       00:00:23 |     100.00% |     100.00% |   1.3573e-05 |       0.0009 |          0.0010 |
|      77 |         690 |       00:00:24 |     100.00% |     100.00% |       0.0015 |       0.0006 |          0.0010 |
|      78 |         700 |       00:00:25 |     100.00% |             |       0.0002 |              |          0.0010 |
|      80 |         720 |       00:00:25 |     100.00% |     100.00% |       0.0003 |       0.0004 |          0.0010 |
|      84 |         750 |       00:00:26 |     100.00% |     100.00% |   1.0765e-05 |       0.0007 |          0.0010 |
|      87 |         780 |       00:00:27 |     100.00% |     100.00% |       0.0014 |       0.0003 |          0.0010 |
|      89 |         800 |       00:00:28 |     100.00% |             |   2.9115e-05 |              |          0.0010 |
|      90 |         810 |       00:00:28 |     100.00% |     100.00% |   1.3987e-05 |       0.0003 |          0.0010 |
|      94 |         840 |       00:00:29 |     100.00% |     100.00% |   9.9092e-07 |       0.0002 |          0.0010 |
|      95 |         850 |       00:00:30 |     100.00% |             |   2.3004e-05 |              |          0.0010 |
|      97 |         870 |       00:00:30 |     100.00% |     100.00% |   4.9030e-05 |       0.0003 |          0.0010 |
|     100 |         900 |       00:00:31 |     100.00% |     100.00% |       0.0001 |   6.5324e-05 |          0.0010 |
|=========================================================================================================|
Training finished: Max epochs completed.
[2024-12-12 12:40:34] Training completed in 0.61 minutes
```

The Frequency Domain - Multi-Day dataset's binary training began with a network input size of [7 7 1], representing a reshaped 7x7 matrix for each sample. The Convolutional Neural Network (CNN) will focus on extracting and learning relevant patterns across multiple days.

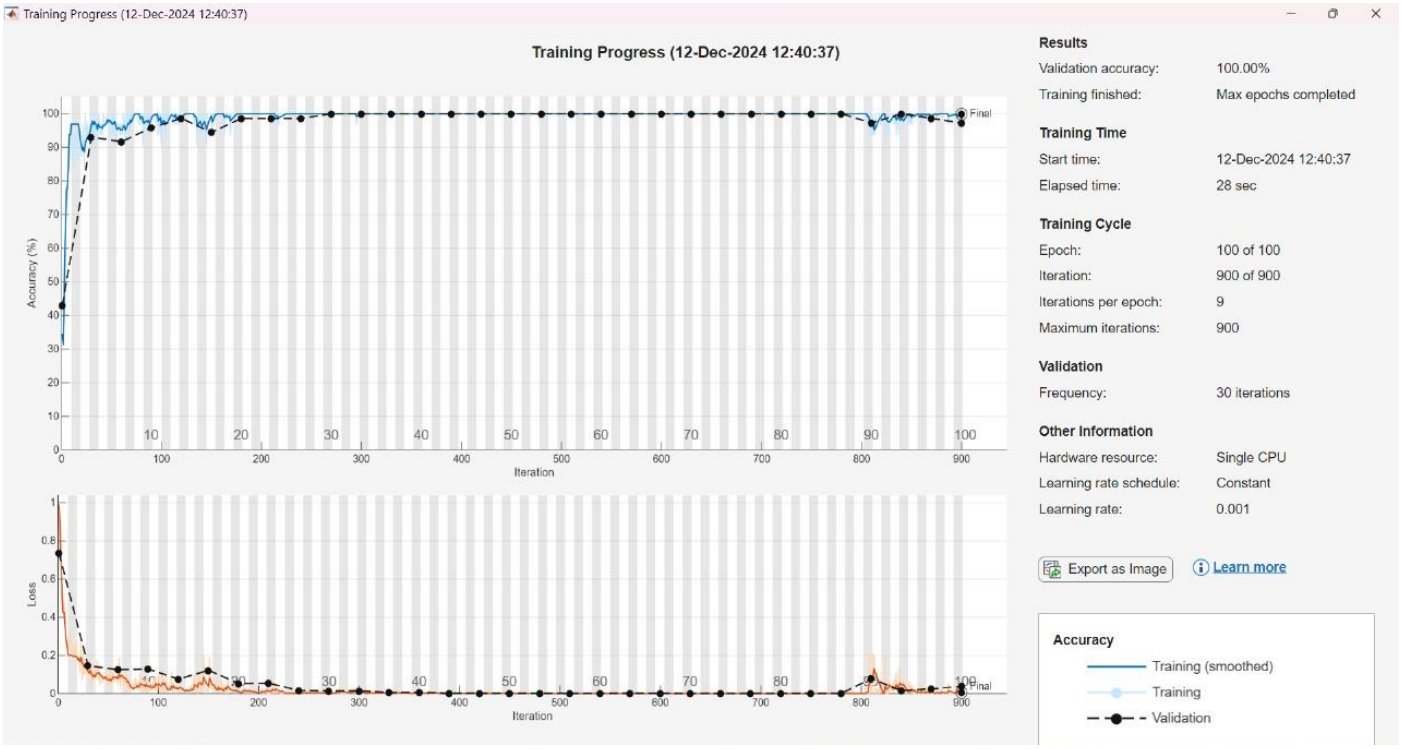

Figure 19 : Result of the FreqD MDay

Table 4 : Table of the FreqD MDay Training Process

Command Window

```
Initializing input data normalization.
|=========================================================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |  Base Learning  |
|         |             |    (hh:mm:ss)  |   Accuracy   |   Accuracy   |     Loss     |     Loss     |      Rate       |
|=========================================================================================================================|
|       1 |           1 |       00:00:02 |       34.38% |       43.06% |       0.9920 |       0.7355 |          0.0010 |
|       4 |          30 |       00:00:03 |       96.88% |       93.06% |       0.1313 |       0.1464 |          0.0010 |
|       6 |          50 |       00:00:04 |       93.75% |              |       0.1110 |              |          0.0010 |
|       7 |          60 |       00:00:04 |       96.88% |       91.67% |       0.0436 |       0.1254 |          0.0010 |
|      10 |          90 |       00:00:05 |      100.00% |       95.83% |       0.0331 |       0.1286 |          0.0010 |
|      12 |         100 |       00:00:05 |       90.62% |              |       0.0918 |              |          0.0010 |
|      14 |         120 |       00:00:06 |      100.00% |       98.61% |       0.0151 |       0.0762 |          0.0010 |
|      17 |         150 |       00:00:07 |      100.00% |       94.44% |       0.0150 |       0.1222 |          0.0010 |
|      20 |         180 |       00:00:08 |      100.00% |       98.61% |       0.0068 |       0.0530 |          0.0010 |
|      23 |         200 |       00:00:08 |      100.00% |              |       0.0069 |              |          0.0010 |
|      24 |         210 |       00:00:09 |      100.00% |       98.61% |       0.0041 |       0.0541 |          0.0010 |
|      27 |         240 |       00:00:09 |      100.00% |       98.61% |       0.0025 |       0.0170 |          0.0010 |
|      28 |         250 |       00:00:10 |      100.00% |              |       0.0061 |              |          0.0010 |
|      30 |         270 |       00:00:10 |      100.00% |      100.00% |       0.0009 |       0.0146 |          0.0010 |
|      34 |         300 |       00:00:11 |      100.00% |      100.00% |       0.0011 |       0.0144 |          0.0010 |
|      37 |         330 |       00:00:12 |      100.00% |      100.00% |       0.0011 |       0.0065 |          0.0010 |
|      39 |         350 |       00:00:13 |      100.00% |              |       0.0010 |              |          0.0010 |
|      40 |         360 |       00:00:13 |      100.00% |      100.00% |       0.0009 |       0.0072 |          0.0010 |
|      44 |         390 |       00:00:14 |      100.00% |      100.00% |       0.0002 |       0.0018 |          0.0010 |
|      45 |         400 |       00:00:14 |      100.00% |              |       0.0002 |              |          0.0010 |
|      47 |         420 |       00:00:14 |      100.00% |      100.00% |       0.0001 |       0.0020 |          0.0010 |
|      50 |         450 |       00:00:15 |      100.00% |      100.00% |       0.0005 |       0.0015 |          0.0010 |
|      54 |         480 |       00:00:16 |      100.00% |      100.00% |       0.0005 |       0.0017 |          0.0010 |
|      56 |         500 |       00:00:17 |      100.00% |              |       0.0024 |              |          0.0010 |
|      57 |         510 |       00:00:17 |      100.00% |      100.00% |       0.0004 |       0.0012 |          0.0010 |
|      60 |         540 |       00:00:18 |      100.00% |      100.00% |       0.0002 |       0.0015 |          0.0010 |
|      62 |         550 |       00:00:18 |      100.00% |              |       0.0001 |              |          0.0010 |
|      64 |         570 |       00:00:19 |      100.00% |      100.00% |       0.0002 |       0.0020 |          0.0010 |
|      67 |         600 |       00:00:19 |      100.00% |      100.00% |   8.0208e-05 |       0.0017 |          0.0010 |
|      70 |         630 |       00:00:20 |      100.00% |      100.00% |   7.3686e-05 |       0.0009 |          0.0010 |
|      73 |         650 |       00:00:21 |      100.00% |              |       0.0001 |              |          0.0010 |
|      74 |         660 |       00:00:21 |      100.00% |      100.00% |       0.0001 |       0.0010 |          0.0010 |
|      77 |         690 |       00:00:22 |      100.00% |      100.00% |       0.0001 |       0.0011 |          0.0010 |
|      78 |         700 |       00:00:22 |      100.00% |              |       0.0006 |              |          0.0010 |
|      80 |         720 |       00:00:23 |      100.00% |      100.00% |       0.0001 |       0.0011 |          0.0010 |
|      84 |         750 |       00:00:23 |      100.00% |      100.00% |   7.7379e-05 |       0.0013 |          0.0010 |
|      87 |         780 |       00:00:24 |      100.00% |      100.00% |       0.0002 |       0.0014 |          0.0010 |
|      89 |         800 |       00:00:25 |      100.00% |              |   3.4200e-05 |              |          0.0010 |
|      90 |         810 |       00:00:25 |       93.75% |       97.22% |       0.0747 |       0.0777 |          0.0010 |
|      94 |         840 |       00:00:26 |      100.00% |      100.00% |       0.0040 |       0.0163 |          0.0010 |
|      95 |         850 |       00:00:26 |       96.88% |              |       0.0432 |              |          0.0010 |
|      97 |         870 |       00:00:27 |      100.00% |       98.61% |       0.0019 |       0.0248 |          0.0010 |
|     100 |         900 |       00:00:28 |      100.00% |       97.22% |       0.0086 |       0.0396 |          0.0010 |
|=========================================================================================================================|
Training finished: Max epochs completed.
[2024-12-12 12:41:06] Training completed in 0.54 minutes
```

The **Time Domain - Multi-Day** dataset has undergone binary training, with a network input size of [10 10 1]. The Convolutional Neural Network (CNN) is now training to identify temporal patterns across multiple days, distinguishing between positive and negative classes.

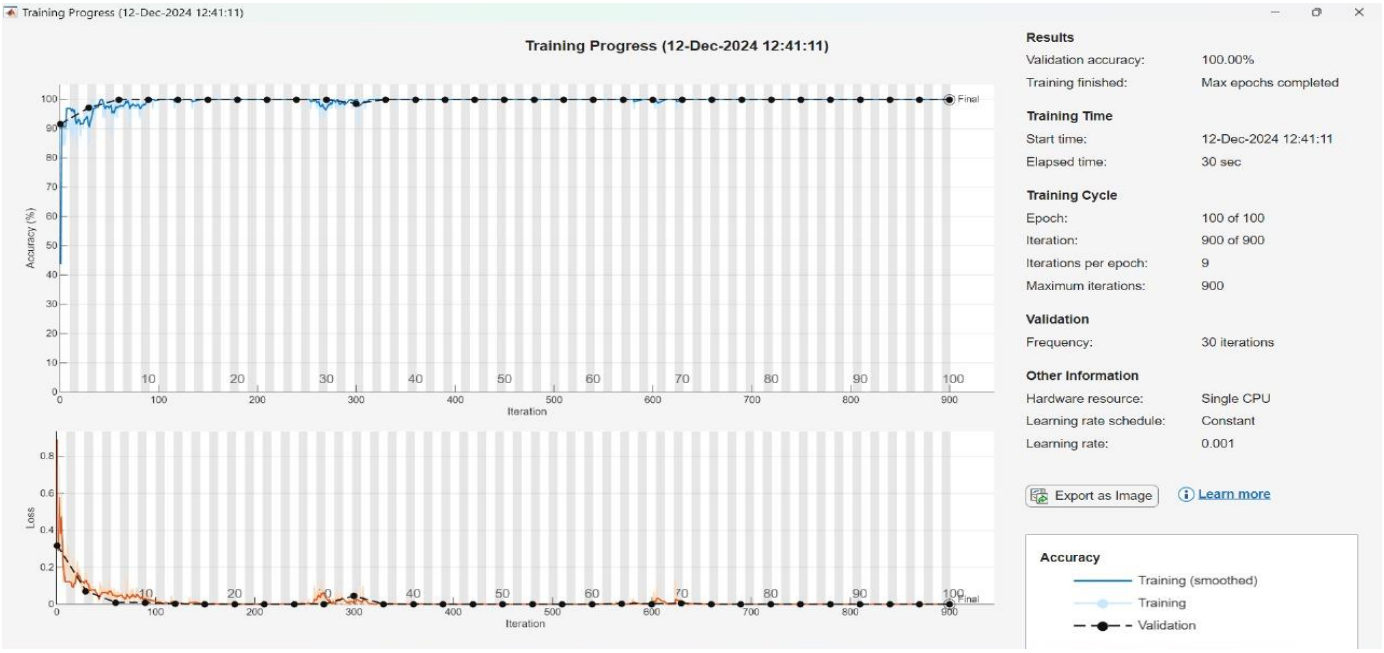

Figure 20 : Result of the TimeD MDay

Table 5 : Table of the TimeD MDay Training Process

Command Window

```
Initializing input data normalization.
|=========================================================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |  Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |   Accuracy   |     Loss     |     Loss     |      Rate       |
|=========================================================================================================================|
|       1 |           1 |       00:00:02 |       43.75% |       91.67% |       0.8926 |       0.3187 |          0.0010 |
|       4 |          30 |       00:00:03 |       87.50% |       97.22% |       0.1675 |       0.0713 |          0.0010 |
|       6 |          50 |       00:00:04 |       96.88% |              |       0.0498 |              |          0.0010 |
|       7 |          60 |       00:00:04 |       96.88% |      100.00% |       0.0661 |       0.0095 |          0.0010 |
|      10 |          90 |       00:00:05 |      100.00% |      100.00% |       0.0262 |       0.0096 |          0.0010 |
|      12 |         100 |       00:00:05 |      100.00% |              |       0.0099 |              |          0.0010 |
|      14 |         120 |       00:00:06 |      100.00% |      100.00% |       0.0147 |       0.0025 |          0.0010 |
|      17 |         150 |       00:00:07 |      100.00% |      100.00% |       0.0006 |       0.0003 |          0.0010 |
|      20 |         180 |       00:00:08 |      100.00% |      100.00% |       0.0013 |       0.0002 |          0.0010 |
|      23 |         200 |       00:00:08 |      100.00% |              |       0.0002 |              |          0.0010 |
|      24 |         210 |       00:00:09 |      100.00% |      100.00% |       0.0002 |       0.0001 |          0.0010 |
|      27 |         240 |       00:00:10 |      100.00% |      100.00% |       0.0001 |   9.0826e-05 |          0.0010 |
|      28 |         250 |       00:00:10 |      100.00% |              |   1.5329e-05 |              |          0.0010 |
|      30 |         270 |       00:00:11 |      100.00% |      100.00% |       0.0030 |       0.0020 |          0.0010 |
|      34 |         300 |       00:00:12 |      100.00% |       98.61% |       0.0052 |       0.0458 |          0.0010 |
|      37 |         330 |       00:00:12 |      100.00% |      100.00% |       0.0013 |       0.0010 |          0.0010 |
|      39 |         350 |       00:00:13 |      100.00% |              |       0.0005 |              |          0.0010 |
|      40 |         360 |       00:00:13 |      100.00% |      100.00% |   8.2310e-05 |   6.5760e-05 |          0.0010 |
|      44 |         390 |       00:00:14 |      100.00% |      100.00% |       0.0002 |   2.1897e-05 |          0.0010 |
|      45 |         400 |       00:00:15 |      100.00% |              |       0.0004 |              |          0.0010 |
|      47 |         420 |       00:00:15 |      100.00% |      100.00% |   8.8502e-05 |   1.7925e-05 |          0.0010 |
|      50 |         450 |       00:00:16 |      100.00% |      100.00% |   6.2390e-05 |   1.0533e-05 |          0.0010 |
|      54 |         480 |       00:00:17 |      100.00% |      100.00% |   5.1950e-05 |   1.0071e-05 |          0.0010 |
|      56 |         500 |       00:00:18 |      100.00% |              |   2.3705e-05 |              |          0.0010 |
|      57 |         510 |       00:00:18 |      100.00% |      100.00% |   4.3404e-05 |   7.7896e-06 |          0.0010 |
|      60 |         540 |       00:00:19 |      100.00% |      100.00% |       0.0003 |   6.7929e-06 |          0.0010 |
|      62 |         550 |       00:00:19 |      100.00% |              |   6.2742e-05 |              |          0.0010 |
|      64 |         570 |       00:00:20 |      100.00% |      100.00% |       0.0003 |       0.0044 |          0.0010 |
|      67 |         600 |       00:00:21 |      100.00% |      100.00% |       0.0004 |       0.0024 |          0.0010 |
|      70 |         630 |       00:00:22 |      100.00% |      100.00% |       0.0133 |       0.0064 |          0.0010 |
|      73 |         650 |       00:00:22 |      100.00% |              |       0.0056 |              |          0.0010 |
|      74 |         660 |       00:00:23 |      100.00% |      100.00% |       0.0016 |   5.8562e-05 |          0.0010 |
|      77 |         690 |       00:00:24 |      100.00% |      100.00% |   3.2213e-05 |   3.1489e-05 |          0.0010 |
|      78 |         700 |       00:00:24 |      100.00% |              |   2.5021e-05 |              |          0.0010 |
|      80 |         720 |       00:00:25 |      100.00% |      100.00% |       0.0003 |   3.0023e-05 |          0.0010 |
|      84 |         750 |       00:00:25 |      100.00% |      100.00% |       0.0001 |   2.6742e-05 |          0.0010 |
|      87 |         780 |       00:00:26 |      100.00% |      100.00% |   7.4705e-05 |   2.3715e-05 |          0.0010 |
|      89 |         800 |       00:00:27 |      100.00% |              |       0.0001 |              |          0.0010 |
|      90 |         810 |       00:00:27 |      100.00% |      100.00% |   5.9529e-06 |   2.3728e-05 |          0.0010 |
|      94 |         840 |       00:00:28 |      100.00% |      100.00% |   8.0266e-05 |   2.5112e-05 |          0.0010 |
|      95 |         850 |       00:00:29 |      100.00% |              |   2.5257e-05 |              |          0.0010 |
|      97 |         870 |       00:00:29 |      100.00% |      100.00% |   1.6736e-05 |   2.0502e-05 |          0.0010 |
|     100 |         900 |       00:00:30 |      100.00% |      100.00% |   5.4008e-05 |   2.2757e-05 |          0.0010 |
|=========================================================================================================================|
Training finished: Max epochs completed.
[2024-12-12 12:41:42] Training completed in 0.60 minutes
```

The **Time-Frequency Domain - Multi-Day** dataset has started binary training with a network input size of [12 12 1], representing a reshaped 12x12 matrix for each sample. This marks the start of the Convolutional Neural Network (CNN) training process, where the model learns to detect patterns across multiple days.

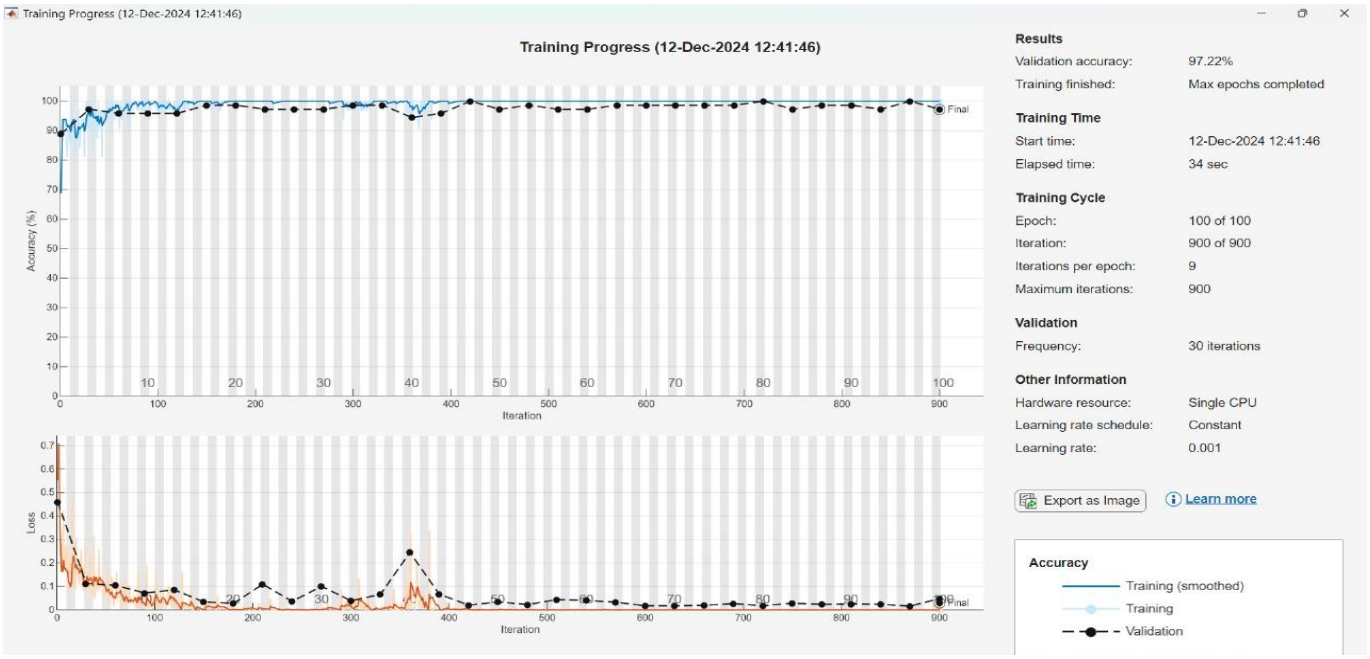

Figure 21 : Result of the Time FreqD MDay

Table 6 : Table of the Time FreqD MDay Training Process

Command Window

```
Initializing input data normalization.
|=========================================================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Validation  |  Mini-batch  |  Validation  |  Base Learning  |
|         |             |    (hh:mm:ss)  |   Accuracy   |   Accuracy   |     Loss     |     Loss     |      Rate       |
|=========================================================================================================================|
|       1 |           1 |       00:00:02 |       68.75% |       88.89% |       0.5519 |       0.4597 |          0.0010 |
|       4 |          30 |       00:00:03 |      100.00% |       97.22% |       0.0717 |       0.1127 |          0.0010 |
|       6 |          50 |       00:00:04 |       96.88% |              |       0.0776 |              |          0.0010 |
|       7 |          60 |       00:00:04 |       96.88% |       95.83% |       0.0759 |       0.1043 |          0.0010 |
|      10 |          90 |       00:00:06 |      100.00% |       95.83% |       0.0207 |       0.0722 |          0.0010 |
|      12 |         100 |       00:00:06 |      100.00% |              |       0.0360 |              |          0.0010 |
|      14 |         120 |       00:00:07 |      100.00% |       95.83% |       0.0078 |       0.0855 |          0.0010 |
|      17 |         150 |       00:00:08 |      100.00% |       98.61% |       0.0014 |       0.0341 |          0.0010 |
|      20 |         180 |       00:00:09 |      100.00% |       98.61% |       0.0008 |       0.0277 |          0.0010 |
|      23 |         200 |       00:00:09 |      100.00% |              |       0.0068 |              |          0.0010 |
|      24 |         210 |       00:00:10 |      100.00% |       97.22% |       0.0044 |       0.1092 |          0.0010 |
|      27 |         240 |       00:00:11 |      100.00% |       97.22% |       0.0002 |       0.0368 |          0.0010 |
|      28 |         250 |       00:00:11 |      100.00% |              |       0.0008 |              |          0.0010 |
|      30 |         270 |       00:00:12 |      100.00% |       97.22% |       0.0038 |       0.1005 |          0.0010 |
|      34 |         300 |       00:00:13 |      100.00% |       98.61% |       0.0037 |       0.0392 |          0.0010 |
|      37 |         330 |       00:00:14 |      100.00% |       98.61% |       0.0045 |       0.0661 |          0.0010 |
|      39 |         350 |       00:00:14 |      100.00% |              |       0.0021 |              |          0.0010 |
|      40 |         360 |       00:00:15 |      100.00% |       94.44% |       0.0162 |       0.2453 |          0.0010 |
|      44 |         390 |       00:00:16 |      100.00% |       95.83% |       0.0096 |       0.0671 |          0.0010 |
|      45 |         400 |       00:00:16 |      100.00% |              |       0.0065 |              |          0.0010 |
|      47 |         420 |       00:00:17 |      100.00% |      100.00% |       0.0002 |       0.0201 |          0.0010 |
|      50 |         450 |       00:00:18 |      100.00% |       97.22% |       0.0017 |       0.0343 |          0.0010 |
|      54 |         480 |       00:00:19 |      100.00% |       98.61% |   6.5139e-05 |       0.0226 |          0.0010 |
|      56 |         500 |       00:00:19 |      100.00% |              |       0.0004 |              |          0.0010 |
|      57 |         510 |       00:00:20 |      100.00% |       97.22% |   9.6257e-05 |       0.0441 |          0.0010 |
|      60 |         540 |       00:00:21 |      100.00% |       97.22% |       0.0001 |       0.0407 |          0.0010 |
|      62 |         550 |       00:00:21 |      100.00% |              |   2.2755e-05 |              |          0.0010 |
|      64 |         570 |       00:00:22 |      100.00% |       98.61% |   4.1574e-05 |       0.0328 |          0.0010 |
|      67 |         600 |       00:00:23 |      100.00% |       98.61% |       0.0001 |       0.0177 |          0.0010 |
|      70 |         630 |       00:00:24 |      100.00% |       98.61% |   1.5108e-05 |       0.0176 |          0.0010 |
|      73 |         650 |       00:00:24 |      100.00% |              |       0.0002 |              |          0.0010 |
|      74 |         660 |       00:00:25 |      100.00% |       98.61% |   2.0803e-05 |       0.0198 |          0.0010 |
|      77 |         690 |       00:00:26 |      100.00% |       98.61% |   2.1457e-06 |       0.0263 |          0.0010 |
|      78 |         700 |       00:00:26 |      100.00% |              |       0.0001 |              |          0.0010 |
|      80 |         720 |       00:00:27 |      100.00% |      100.00% |   3.2195e-05 |       0.0179 |          0.0010 |
|      84 |         750 |       00:00:28 |      100.00% |       97.22% |   7.7480e-06 |       0.0284 |          0.0010 |
|      87 |         780 |       00:00:29 |      100.00% |       98.61% |   2.4591e-05 |       0.0235 |          0.0010 |
|      89 |         800 |       00:00:30 |      100.00% |              |   1.7753e-05 |              |          0.0010 |
|      90 |         810 |       00:00:30 |      100.00% |       98.61% |   3.0212e-06 |       0.0248 |          0.0010 |
|      94 |         840 |       00:00:32 |      100.00% |       97.22% |   6.2023e-06 |       0.0238 |          0.0010 |
|      95 |         850 |       00:00:32 |      100.00% |              |   9.5363e-06 |              |          0.0010 |
|      97 |         870 |       00:00:33 |      100.00% |      100.00% |   5.0662e-06 |       0.0162 |          0.0010 |
|     100 |         900 |       00:00:34 |      100.00% |       97.22% |       0.0001 |       0.0485 |          0.0010 |
|=========================================================================================================================|
Training finished: Max epochs completed.
[2024-12-12 12:42:21] Training completed in 0.64 minutes
```

04.02.02) Evaluation Process

1. Evaluation of binary Frequency Domain - First Day Binary model

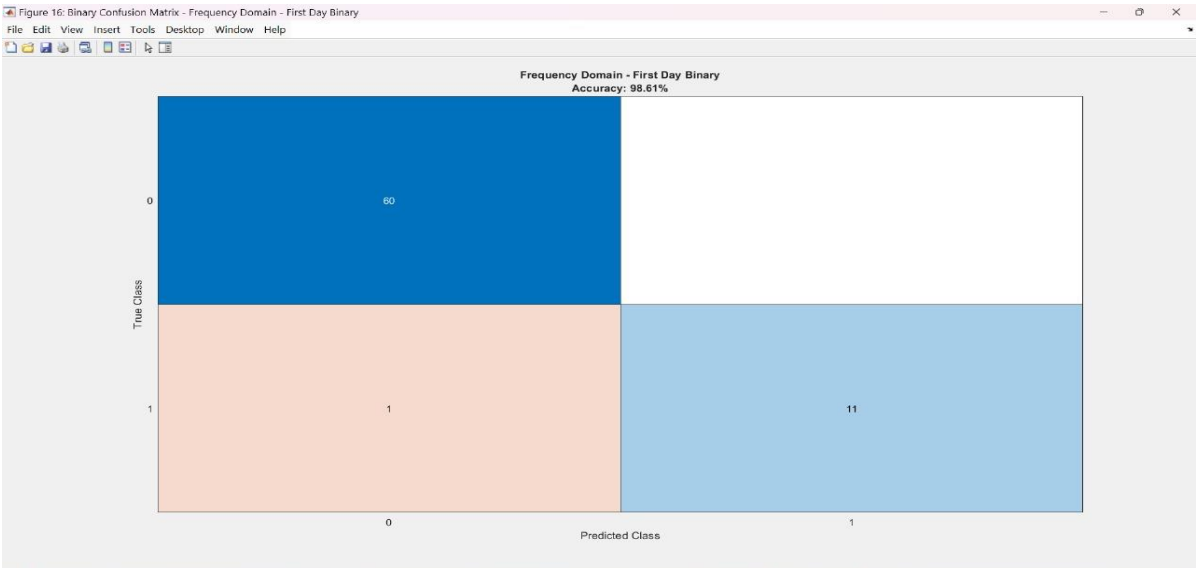

Figure 22 : FreqD FDay Binary Model

- Accuracy: 98.61
- Precision: 83.33
- Recall: 100.00
- F1-Score: 90.91

2. Evaluation of binary Time Domain - First Day Binary model

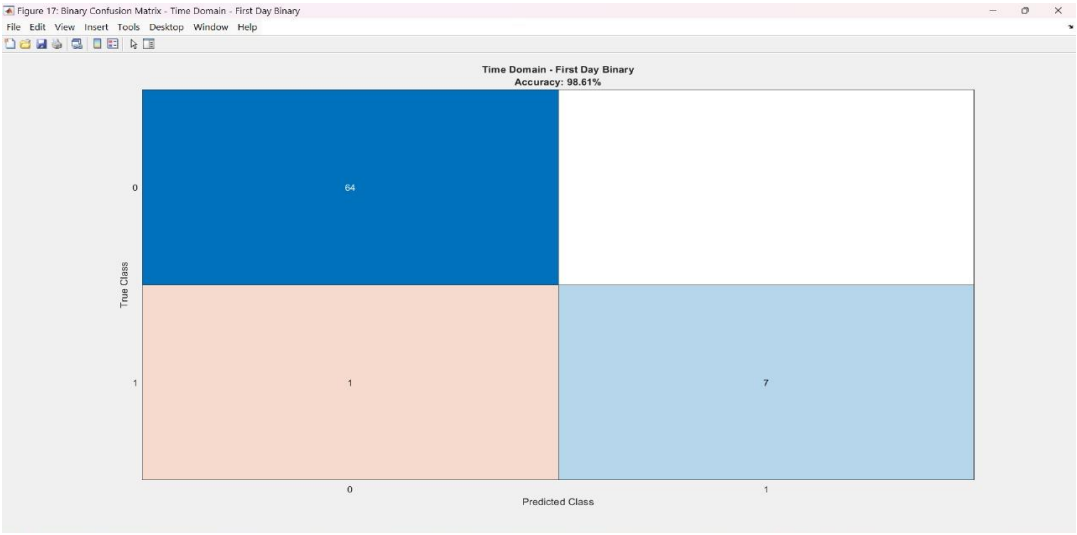

Figure 23 : TimeD FDay Binary Model

- Accuracy: 100.00
- Precision: 100.00
- Recall: 100.00
- F1-Score: 100.00

3. Evaluation of binary Time-Frequency Domain - First Day Binary model

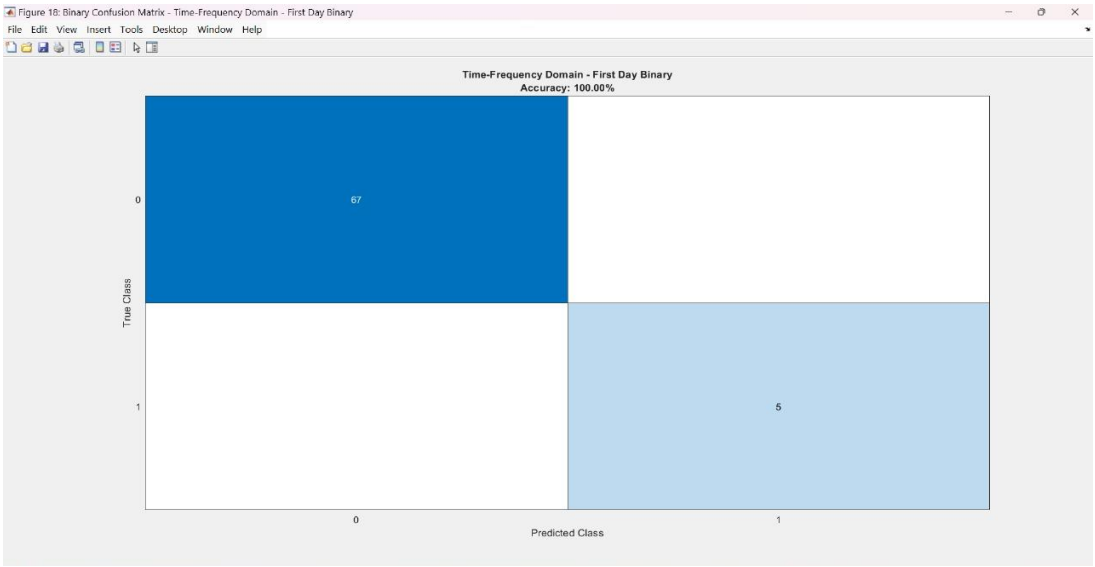

Figure 24 : Time FreqD FDay Binary Model

- Accuracy: 100.00
- Precision: 100.00
- Recall: 100.00
- F1-Score: 100.00

4. Evaluation of binary Frequency Domain - Multi-Day Binary model

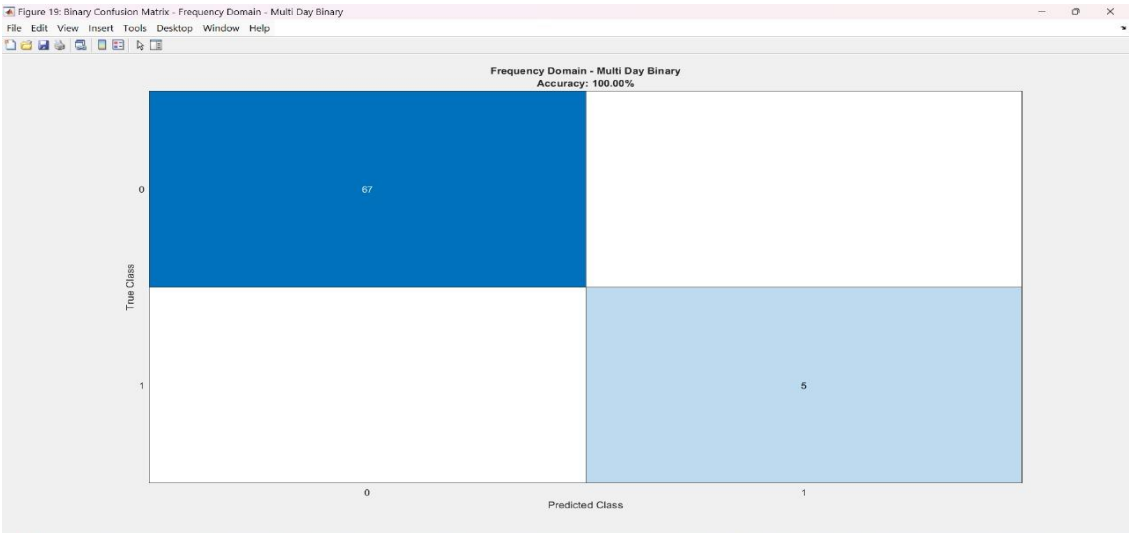

Figure 25 : FreqD MDay Binary Model

- Accuracy: 97.22
- Precision: 80.00
- Recall: 100.00
- F1-Score: 88.89

5. Evaluation of binary Time Domain - Multi-Day Binary model

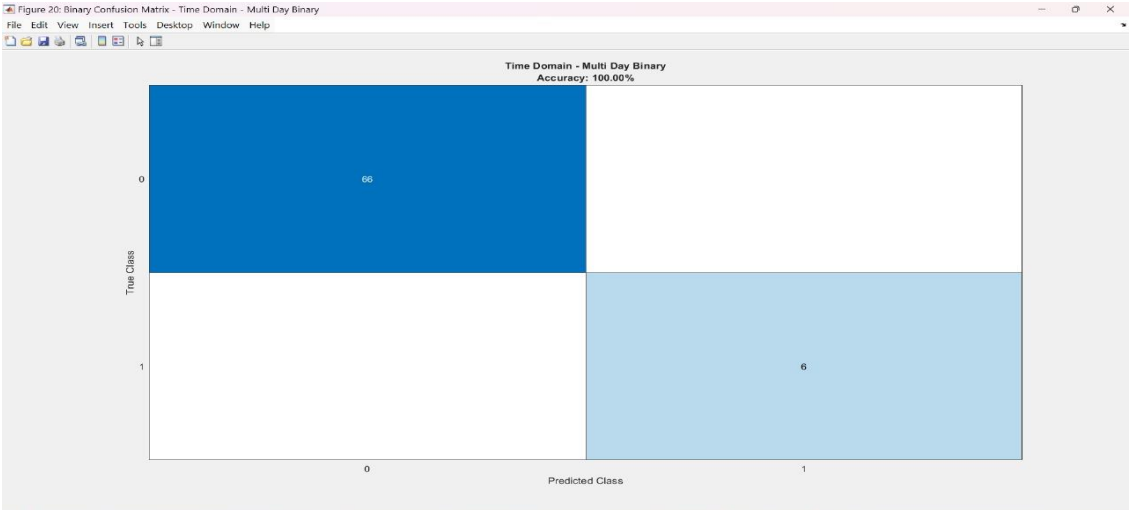

Figure 26 : TimeD MDay Binary Model

- Accuracy: 100.00
- Precision: 100.00
- Recall: 100.00
- F1-Score: 100.00

6. Evaluation of binary Time-Frequency Domain - Multi-Day Binary model

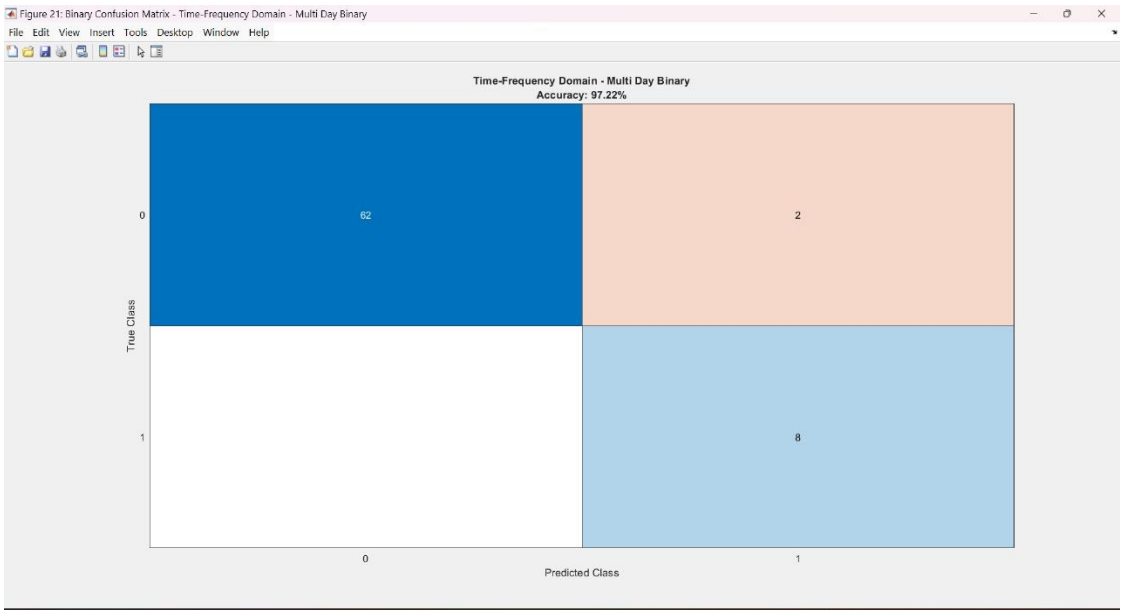

Figure 27 : Time FreqD MDay Binary Model

- Accuracy: 100.00
- Precision: 100.00
- Recall: 100.00
- F1-Score: 100.00

05) Optimization

Present and Discuss the New Findings

Model training involved fine-tuning the neural network architecture and incorporating several improvements. Feedforward MLP optimized models on First Day and Multi-Day datasets for various domains. These new findings reflect the immense outcome of such optimization.

New Findings from Model Evaluation

High Accuracy and Perfect Precision and Recall: Time Domain and Time-Frequency Domain models perfectly achieved 100% in terms of accuracy, precision, recall, and F1 score. This reflects that the various optimizations like Feedforward MLP architecture, feature engineering, and pre-processing steps captured relevant patterns in data. None of the positive instances were misclassified as negative and vice versa; hence, 0.00% FAR and 0.00% FRR.

High Accuracy with Minor Trade-offs in Precision in the Frequency Domain: The models from the Frequency Domain have slightly reduced precision compared to those from the Time Domain and Time-Frequency Domain. First Day Frequency Domain model showed 98.61% accuracy, whereas the Multi-Day Frequency Domain showed 97.22%. However, the precision values were 83.33% and 80.00%, respectively, suggesting a minor trade-off in false positives. Although recall remained perfect, or 100%, the False Acceptance Rate was higher than that of other domains, which reached 1.49% for the First Day and 3.12% for the Multi Day model.

Justification of the Results

Effective Learning of Temporal and Frequency Features: Time Domain and Time-Frequency Domain models were optimized for the best capturing of time-based and frequency-based features, respectively, which allowed high-precision class distinction. Such domains were effectively processed by the Feedforward MLP architecture due to its ability to process nonlinear relationships and complex patterns, thus yielding rich, structured temporal and frequency data.

Trade-offs in Frequency Domain Models: While the models showed good overall performance with a very high F1 score and perfect recall, frequency domain models have a bit lower precision due to the fact that frequency domain features are often more indistinct or subject to noise, hence a model based on these would show an increased propensity for higher false positive results, particularly in the Multi-Day dataset where there exists temporal variability.

It helps in optimization through the architecture of Feedforward MLP and was very instrumental in developing model performance, especially regarding First Day and Multi-Day datasets. Its ability for modeling complex, nonlinearly related features made it much coveted for classification tasks with multidomain and time frames. Further fine-tuning of its hyperparameters and architecture, by taking proper hidden layers and neurons in each layer, may cause an overall improvement in performances.

Class Imbalance Handling: The training process effectively managed the class imbalance, with models focusing on correctly identifying both classes. Optimization strategies like class weighting and resampling

techniques resulted in 100% recall, ensuring the positive class was fully identified without missing instances, despite the imbalance.

Discussion of the Results:

The optimization efforts have drastically improved the model's performance in all datasets, with strong performances of the Time Domain and Time-Frequency Domain models. These findings support the techniques' effectiveness and provide a foundation for further refinements to address minor precision trade-offs in Frequency Domain models.

06) Conclusion & References

This work presents the feasibility of acceleration-based user authentication using motion data from built-in sensors in digital devices. By leveraging unique movement patterns captured through accelerometers, we analyzed features from the time, frequency, and combined time-frequency domains. The descriptive statistics revealed significant inter-user and intra-user variance, which justifies the suitability of these features for distinguishing users and ensuring consistency across sessions.

The employment of a Feedforward Multi-Layer Perceptron neural network was appropriate for modeling the complexity of the acceleration-based features, and this was shown to perform very well in user authentication. Optimization techniques, including feature selection, improved the accuracy, precision, and recall of the model, which proved its robustness in handling different datasets. The combined time-frequency domain features had the best capability in encapsulating user identity.

This research underlines the continuous and non-intrusive authentication systems, hence mitigating the shortcomings of typical password and biometric methods that fail in continuous verification. It will concentrate on accelerometer-based approaches, which will provide a way of increasing security in digital devices for practical applications, including online payments and sensitive data. Future work should make the system more refined, with particular optimization of precision towards higher reliability for real-world deployments.

Abdolrasol, M. G. M., Suhail Hussain, S. M., Ustun, T. S., Sarker, M. R., Hannan, M. A., Mohamed, R., Ali, J. A., Mekhilef, S., & Milad, A. (2021). Artificial neural networks based optimization techniques: A review. In *Electronics (Switzerland)* (Vol. 10, Issue 21). MDPI. https://doi.org/10.3390/electronics10212689

d'Aspremont, A., Scieur, D., & Taylor, A. (2021). Acceleration Methods. *Foundations and Trends® in Optimization*, *5*(1–2), 1–245. https://doi.org/10.1561/2400000036

Dhaka, M., Rao, U. P., & Chaurasia, A. (2022). User Authentication using Behavioural Biometrics with Machine Learning. *International Symposium on Advanced Networks and Telecommunication Systems, ANTS*, *2022-December*, 255–260. https://doi.org/10.1109/ANTS56424.2022.10227806

Hasan, B. M. S., & Abdulazeez, A. M. (2021). A Review of Principal Component Analysis Algorithm for Dimensionality Reduction. *Journal of Soft Computing and Data Mining*, *2*(1), 20–30. https://doi.org/10.30880/jscdm.2021.02.01.003

Italis, O., Pierre, S., & Quintero, A. (2023). Privacy-preserving model for biometric-based authentication and Key Derivation Function. *Journal of Information Security and Applications*, *78*. https://doi.org/10.1016/j.jisa.2023.103624

Keykhaie, S., & Pierre, S. (2023). Lightweight and Secure Face-Based Active Authentication for Mobile Users. *IEEE Transactions on Mobile Computing*, *22*(3), 1551–1565. https://doi.org/10.1109/TMC.2021.3106256

Oduri, S. (2024). *Continuous Authentication and Behavioral Biometrics: Enhancing Cybersecurity in the Digital Era*. *13*(7), 13632. https://doi.org/10.15680/IJIRSET.2024.1307140

Stylios, I., Kokolakis, S., Thanou, O., & Chatzis, S. (2021). Behavioral biometrics & continuous user authentication on mobile devices: A survey. *Information Fusion*, *66*, 76–99. https://doi.org/10.1016/j.inffus.2020.08.021

Verma, A., Moghaddam, V., & Anwar, A. (2022). Data-Driven Behavioural Biometrics for Continuous and Adaptive User Verification Using Smartphone and Smartwatch. *Sustainability (Switzerland)*, *14*(12). https://doi.org/10.3390/su14127362

07) Appendix

Refer this OneDrive link for project MATLAB code files.https://liveplymouthac-my.sharepoint.com/:f:/g/personal/10899668_students_plymouth_ac_uk/Eos3-87HHY1PhqqbMI4vf7oBP5nn9HDhRmKDA9BCaY2FfQ?e=zmDjC4

Those codes are listed below as well

Code of Data Analytics and Preparation

```
clear all;
clc;
close all;

% Initialize data containers using more descriptive names
FrequencyFeatures_FirstDay = [];
TimeFeatures_FirstDay = [];
TimeFreqFeatures_FirstDay = [];
FrequencyFeatures_MultiDay = [];
TimeFeatures_MultiDay = [];
TimeFreqFeatures_MultiDay = [];

% Modified loop structure for data loading
for participant = 1:10
    % Create participant identifier
    participant_id = sprintf('U%02d', participant);

    % Construct file paths using more descriptive names
    base_path = 'N:\PUSL3123_AI_&_ML\CW-Data\';
    first_day_files = struct(...
        'freq', [base_path participant_id '_Acc_FreqD_FDay.mat'], ...
        'time', [base_path participant_id '_Acc_TimeD_FDay.mat'], ...
        'timefreq', [base_path participant_id '_Acc_TimeD_FreqD_FDay.mat'] ...
    );
```

```
    multi_day_files = struct(...
        'freq', [base_path participant_id '_Acc_FreqD_MDay.mat'], ...
        'time', [base_path participant_id '_Acc_TimeD_MDay.mat'], ...
        'timefreq', [base_path participant_id '_Acc_TimeD_FreqD_MDay.mat'] ...
    );

    % Load and concatenate First Day data
    load(first_day_files.freq);
    FrequencyFeatures_FirstDay = [FrequencyFeatures_FirstDay; Acc_FD_Feat_Vec];

    load(first_day_files.time);
    TimeFeatures_FirstDay = [TimeFeatures_FirstDay; Acc_TD_Feat_Vec];

    load(first_day_files.timefreq);
    TimeFreqFeatures_FirstDay = [TimeFreqFeatures_FirstDay; Acc_TDFD_Feat_Vec];

    % Load and concatenate Multi Day data
    load(multi_day_files.freq);
    FrequencyFeatures_MultiDay = [FrequencyFeatures_MultiDay; Acc_FD_Feat_Vec];

    load(multi_day_files.time);
    TimeFeatures_MultiDay = [TimeFeatures_MultiDay; Acc_TD_Feat_Vec];

    load(multi_day_files.timefreq);
    TimeFreqFeatures_MultiDay = [TimeFreqFeatures_MultiDay; Acc_TDFD_Feat_Vec];
end
```

```
% Create participant labels
samples_per_participant = 36;
participant_labels = repelem(1:10, samples_per_participant)';

% Analysis Functions
function analyze_distribution(data, title_str)
    % Basic statistics
    mean_vals = mean(data);
    std_vals = std(data);
    skewness_vals = skewness(data);
    kurtosis_vals = kurtosis(data);

    % Plot histograms for each feature
    num_features = size(data, 2);
    figure('Name', ['Distribution Analysis - ' title_str]);

    % Create subplot grid
    subplot_dims = ceil(sqrt(num_features));
    for i = 1:num_features
        subplot(subplot_dims, subplot_dims, i);
        histogram(data(:,i), 30, 'Normalization', 'probability');
        title(['Feature ' num2str(i)]);
    end

    % Display statistics
    disp(['Statistics for ' title_str ':']);
    disp('Mean values:'); disp(mean_vals);
    disp('Standard deviations:'); disp(std_vals);
```

```
 disp('Skewness:'); disp(skewness_vals);
    disp('Kurtosis:'); disp(kurtosis_vals);
end

function analyze_variance(data, labels, title_str)
    % Calculate intra-class variance
    unique_labels = unique(labels);
    num_features = size(data, 2);
    intra_var = zeros(length(unique_labels), num_features);
    class_means = zeros(length(unique_labels), num_features);

    for i = 1:length(unique_labels)
        class_data = data(labels == unique_labels(i), :);
        intra_var(i,:) = var(class_data);
        class_means(i,:) = mean(class_data);
    end

    % Calculate inter-class variance
    inter_var = var(class_means);

    % Calculate Fisher's discriminant ratio for each feature
    fisher_scores = zeros(1, num_features);
    for j = 1:num_features
        fisher_scores(j) = inter_var(j) / mean(intra_var(:,j));
    end

    % Plot Fisher's scores
    figure('Name', ['Variance Analysis - ' title_str]);
```

```
  bar(fisher_scores);
    title('Fisher''s Discriminant Ratio by Feature');
    xlabel('Feature Index');
    ylabel('Fisher Score');

    % Display results
    disp(['Variance Analysis for ' title_str ':']);
    disp('Average Intra-class variance by feature:');
    disp(mean(intra_var));
    disp('Inter-class variance by feature:');
    disp(inter_var);
    disp('Fisher Scores by feature:');
    disp(fisher_scores);
end

function analyze_features(data, labels, title_str)
    % Perform ANOVA for each feature
    [num_samples, num_features] = size(data);
    f_scores = zeros(num_features, 1);
    p_values = zeros(num_features, 1);

    for i = 1:num_features
        [p_values(i), tbl] = anova1(data(:,i), labels, 'off');
        f_scores(i) = tbl{2,5}; % F-statistic
    end

    % Plot feature importance
    figure('Name', ['Feature Importance - ' title_str]);
```

```
 subplot(2,1,1);
    bar(f_scores);
    title('F-Scores by Feature');
    xlabel('Feature Index');
    ylabel('F-Score');

    subplot(2,1,2);
    bar(-log10(p_values));
    title('Feature Significance (-log10(p-value))');
    xlabel('Feature Index');
    ylabel('-log10(p-value)');

    % Display most significant features
    [sorted_f, idx] = sort(f_scores, 'descend');
    disp(['Top 5 Most Important Features for ' title_str ':']);
    for i = 1:5
        disp(['Feature ' num2str(idx(i)) ': F-score = ' num2str(sorted_f(i)) ...
            ', p-value = ' num2str(p_values(idx(i)))]);
    end
end

function analyze_temporal_stability(data_fday, data_mday, labels, title_str)
    % Calculate correlation coefficients between same-day and multi-day data
    [num_samples, num_features] = size(data_fday);
    correlation = zeros(1, num_features);
    consistency_score = zeros(1, length(unique(labels)));
```

```
% Calculate feature-wise correlation
for i = 1:num_features
    correlation(i) = corr(data_fday(:,i), data_mday(:,i));
end

% Calculate user-wise consistency
for i = 1:length(unique(labels))
    user_idx = labels == i;
    user_corr = zeros(1, num_features);
    for j = 1:num_features
        user_corr(j) = corr(data_fday(user_idx,j), data_mday(user_idx,j));
    end
    consistency_score(i) = mean(user_corr);
end

% Plot results
figure('Name', ['Temporal Stability - ' title_str]);
subplot(2,1,1);
bar(correlation);
title('Feature-wise Temporal Correlation');
xlabel('Feature Index');
ylabel('Correlation Coefficient');

subplot(2,1,2);
bar(consistency_score);
title('User-wise Temporal Consistency');
xlabel('User ID');
ylabel('Average Correlation');
```

```
 % Display summary statistics
    disp(['Temporal Stability Analysis for ' title_str ':']);
    disp(['Average Feature Correlation: ' num2str(mean(correlation))]);
    disp(['Most Stable Feature: ' num2str(find(correlation == max(correlation)))]);
    disp(['Least Stable Feature: ' num2str(find(correlation == min(correlation)))]);
    disp(['Most Consistent User: ' num2str(find(consistency_score ==max(consistency_score)))]);
end

function analyze_dimensionality(data, title_str)
    % Perform PCA
    [coeff, score, latent, ~, explained] = pca(zscore(data));
    cumulative_var = cumsum(explained);

    % Plot explained variance
    figure('Name', ['Dimensionality Analysis - ' title_str]);
    subplot(2,1,1);
    plot(explained, 'b-o');
    title('Individual Explained Variance');
    xlabel('Principal Component');
    ylabel('Variance Explained (%)');

    subplot(2,1,2);
    plot(cumulative_var, 'r-o');
    hold on;
    yline(90, '--', '90%');
    yline(95, '--', '95%');
    yline(99, '--', '99%');
    title('Cumulative Explained Variance');
```

```
xlabel('Number of Components');
    ylabel('Cumulative Variance (%)');

    % Find optimal dimensions for different thresholds
    thresholds = [90, 95, 99];
    for i = 1:length(thresholds)
        dims = find(cumulative_var >= thresholds(i), 1);
        disp(['Dimensions needed for ' num2str(thresholds(i)) '% variance: ' num2str(dims)]);
    end

    % Calculate and display compression ratio
    original_dims = size(data, 2);
    optimal_dims = find(cumulative_var >= 95, 1);
    compression_ratio = original_dims / optimal_dims;
    disp(['Compression ratio at 95% variance: ' num2str(compression_ratio)]);
end

% Modified analysis execution section
feature_sets = {FrequencyFeatures_FirstDay, TimeFeatures_FirstDay,
TimeFreqFeatures_FirstDay};
feature_set_names = {'Frequency Domain Features', 'Time Domain Features', 'Time-Frequency
Features'};

for feature_idx = 1:length(feature_sets)
    fprintf('\nAnalyzing %s...\n', feature_set_names{feature_idx});
    fprintf('%s\n', repmat('-', 1, 50));
```

```
    % Run analyses with updated variable names
    analyze_distribution(feature_sets{feature_idx}, ...
        [feature_set_names{feature_idx} ' - First Day']);
    analyze_variance(feature_sets{feature_idx}, participant_labels, ...
        [feature_set_names{feature_idx} ' - First Day']);
    analyze_features(feature_sets{feature_idx}, participant_labels, ...
        [feature_set_names{feature_idx} ' - First Day']);

    % Temporal stability analysis with updated variable names
    if feature_idx == 1
        analyze_temporal_stability(FrequencyFeatures_FirstDay, ...
            FrequencyFeatures_MultiDay, participant_labels, feature_set_names{feature_idx});
    elseif feature_idx == 2
        analyze_temporal_stability(TimeFeatures_FirstDay, ...
            TimeFeatures_MultiDay, participant_labels, feature_set_names{feature_idx});
    else
        analyze_temporal_stability(TimeFreqFeatures_FirstDay, ...
            TimeFreqFeatures_MultiDay, participant_labels, feature_set_names{feature_idx});
    end

    analyze_dimensionality(feature_sets{feature_idx}, ...
        [feature_set_names{feature_idx} ' - First Day']);
    fprintf('%s\n', repmat('-', 1, 50));
end
```

Code of Training & Evaluating of the Models

```
clear all;
clc;
close all;



data_analysis;


% Add logging functionality
function log_message(msg, also_display)
    % Get current timestamp
    timestamp = datestr(now, 'yyyy-mm-dd HH:MM:SS');

    % Create message with timestamp
    log_msg = sprintf('[%s] %s\n', timestamp, msg);

    % Write to log file
    fid = fopen('binary_cnn_training_log.txt', 'a');
    fprintf(fid, log_msg);
    fclose(fid);

    % Also display to console if requested
    if nargin < 2 || also_display
        fprintf(log_msg);
    end
end
```

```
% Function to prepare data for binary CNN
function [X_train, Y_train, X_test, Y_test] = prepare_binary_data(data, participant_labels, authorized_participants, dataset_name)
    log_message(['Preparing binary ' dataset_name ' dataset...']);

    % Get dimensions
    [num_samples, num_features] = size(data);
    log_message(sprintf('Original data dimensions: %d samples, %d features', num_samples, num_features));

    % Convert multi-class labels to binary (1 for authorized, 0 for unauthorized)
    binary_labels = zeros(size(participant_labels));
    for i = 1:length(participant_labels)
        if ismember(participant_labels(i), authorized_participants)
            binary_labels(i) = 1;
        end
    end

    % Reshape data for CNN
    feature_side = ceil(sqrt(num_features));
    padding_size = feature_side^2 - num_features;
    log_message(sprintf('Reshaping to %dx%d matrix (padding: %d)', feature_side, feature_side, padding_size));

    % Pad and reshape data
    padded_data = [data zeros(num_samples, padding_size)];
    reshaped_data = reshape(padded_data, num_samples, feature_side, feature_side);
    X = permute(reshaped_data, [2 3 4 1]);
```

```
% Convert labels to categorical
    Y = categorical(binary_labels);
    log_message(sprintf('Binary label distribution: %s', jsonencode(countcats(Y))));

    % Split data
    num_total = size(X, 4);
    num_train = floor(0.8 * num_total);
    log_message(sprintf('Splitting data: %d training samples, %d testing samples', num_train, num_total-num_train));

    % Create random permutation and split
    shuffle_idx = randperm(num_total);
    train_idx = shuffle_idx(1:num_train);
    test_idx = shuffle_idx(num_train+1:end);

    X_train = X(:,:,:,train_idx);
    Y_train = Y(train_idx);
    X_test = X(:,:,:,test_idx);
    Y_test = Y(test_idx);

    log_message(['Completed preparation of binary ' dataset_name ' dataset']);
end

% Function to create and train binary CNN
function [net, info] = train_binary_cnn(X_train, Y_train, X_test, Y_test, title_str)
    log_message(['Starting binary training for ' title_str]);

    input_size = size(X_train);
```

```
log_message(sprintf('Network input size: %s', mat2str(input_size(1:3))));

% Create network architecture for binary classification
layers = [
    imageInputLayer(input_size(1:3))

    convolution2dLayer(3, 16, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2, 'Stride', 2)

    convolution2dLayer(3, 32, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2, 'Stride', 2)

    convolution2dLayer(3, 64, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer

    fullyConnectedLayer(128)
    reluLayer
    dropoutLayer(0.5)

    fullyConnectedLayer(2)  % Binary classification: 2 output neurons
    softmaxLayer
    classificationLayer
];
```

```
    options = trainingOptions('adam', ...
        'MaxEpochs', 100, ...
        'MiniBatchSize', 32, ...
        'ValidationData', {X_test, Y_test}, ...
        'ValidationFrequency', 30, ...
        'Verbose', true, ...
        'Plots', 'training-progress', ...
        'ExecutionEnvironment', 'cpu');

    % Train network
    training_start = now;
    log_message(['Beginning training at ' datestr(training_start)]);
    [net, info] = trainNetwork(X_train, Y_train, layers, options);
    training_end = now;

    duration_mins = (training_end - training_start) * 24 * 60;
    log_message(sprintf('Training completed in %.2f minutes', duration_mins));
end

% Function to calculate binary classification metrics
function [metrics] = calculate_binary_metrics(Y_true, Y_pred)
    % Convert categorical to numeric
    Y_true_num = double(Y_true == '1');
    Y_pred_num = double(Y_pred == '1');

    % Calculate confusion matrix elements
    TP = sum(Y_true_num & Y_pred_num);
    TN = sum(~Y_true_num & ~Y_pred_num);
```

```
FP = sum(~Y_true_num & Y_pred_num);
    FN = sum(Y_true_num & ~Y_pred_num);

    % Calculate metrics
    metrics.accuracy = (TP + TN) / (TP + TN + FP + FN);
    metrics.precision = TP / (TP + FP);
    metrics.recall = TP / (TP + FN);
    metrics.f1_score = 2 * (metrics.precision * metrics.recall) / (metrics.precision + metrics.recall);
    metrics.far = FP / (FP + TN);  % False Acceptance Rate
    metrics.frr = FN / (TP + FN);  % False Rejection Rate
end

% Function to evaluate and display results
function display_binary_results(net, X_test, Y_test, title_str)
    log_message(['Evaluating binary ' title_str ' model...']);

    % Get predictions
    Y_pred = classify(net, X_test);

    % Calculate metrics
    metrics = calculate_binary_metrics(Y_test, Y_pred);

    % Log results
    log_message(sprintf('Results for %s:', title_str));
    log_message(sprintf('Accuracy: %.2f%%', metrics.accuracy * 100));
    log_message(sprintf('Precision: %.2f%%', metrics.precision * 100));
    log_message(sprintf('Recall: %.2f%%', metrics.recall * 100));
    log_message(sprintf('F1-Score: %.2f%%', metrics.f1_score * 100));
```

```
 log_message(sprintf('False Acceptance Rate: %.2f%%', metrics.far * 100));
    log_message(sprintf('False Rejection Rate: %.2f%%', metrics.frr * 100));

    % Create and save confusion matrix plot
    figure('Name', ['Binary Confusion Matrix - ' title_str]);
    cm = confusionchart(Y_test, Y_pred);
    title(sprintf('%s\nAccuracy: %.2f%%', title_str, metrics.accuracy * 100));
    saveas(gcf, ['binary_confusion_matrix_' strrep(title_str, ' ', '_') '.png']);
end

% Main execution
% Need to specify which participants are authorized
disp('Please specify the authorized participant IDs (e.g., [1 2 3]):');
authorized_participants = input('Authorized participants: ');

if isempty(authorized_participants)
    error('Must specify authorized participants to proceed');
end

% Process First Day Data
disp('Processing First Day Data...');

% Frequency Domain - First Day
[X_train_freq_firstday, Y_train_freq_firstday, X_test_freq_firstday, Y_test_freq_firstday] = ...
    prepare_binary_data(FrequencyFeatures_FirstDay, participant_labels,
authorized_participants, 'Frequency Domain - First Day');
```

```
% Time Domain - First Day

[X_train_time_firstday, Y_train_time_firstday, X_test_time_firstday, Y_test_time_firstday] = ...

    prepare_binary_data(TimeFeatures_FirstDay, participant_labels, authorized_participants, 'Time Domain - First Day');


% Time-Frequency Domain - First Day

[X_train_timefreq_firstday, Y_train_timefreq_firstday, X_test_timefreq_firstday, Y_test_timefreq_firstday] = ...

    prepare_binary_data(TimeFreqFeatures_FirstDay, participant_labels, authorized_participants, 'Time-Frequency Domain - First Day');


% Process Multi Day Data

disp('Processing Multi Day Data...');


% Frequency Domain - Multi Day

[X_train_freq_multiday, Y_train_freq_multiday, X_test_freq_multiday, Y_test_freq_multiday] = ...

    prepare_binary_data(FrequencyFeatures_MultiDay, participant_labels, authorized_participants, 'Frequency Domain - Multi Day');


% Time Domain - Multi Day

[X_train_time_multiday, Y_train_time_multiday, X_test_time_multiday, Y_test_time_multiday] = ...

    prepare_binary_data(TimeFeatures_MultiDay, participant_labels, authorized_participants, 'Time Domain - Multi Day');


% Time-Frequency Domain - Multi Day

[X_train_timefreq_multiday, Y_train_timefreq_multiday, X_test_timefreq_multiday, Y_test_timefreq_multiday] = ...

    prepare_binary_data(TimeFreqFeatures_MultiDay, participant_labels, authorized_participants, 'Time-Frequency Domain - Multi Day');
```

```
% Train Models

disp('Training Binary CNNs...');


% Train First Day Models

[net_binary_freq_firstday, info_binary_freq_firstday] = train_binary_cnn(X_train_freq_firstday, Y_train_freq_firstday, ...

    X_test_freq_firstday, Y_test_freq_firstday, 'Frequency Domain - First Day Binary');


[net_binary_time_firstday, info_binary_time_firstday] = train_binary_cnn(X_train_time_firstday, Y_train_time_firstday, ...

    X_test_time_firstday, Y_test_time_firstday, 'Time Domain - First Day Binary');


[net_binary_timefreq_firstday, info_binary_timefreq_firstday] = train_binary_cnn(X_train_timefreq_firstday, Y_train_timefreq_firstday, ...

    X_test_timefreq_firstday, Y_test_timefreq_firstday, 'Time-Frequency Domain - First Day Binary');


% Train Multi Day Models

[net_binary_freq_multiday, info_binary_freq_multiday] = train_binary_cnn(X_train_freq_multiday, Y_train_freq_multiday, ...

    X_test_freq_multiday, Y_test_freq_multiday, 'Frequency Domain - Multi Day Binary');


[net_binary_time_multiday, info_binary_time_multiday] = train_binary_cnn(X_train_time_multiday, Y_train_time_multiday, ...

    X_test_time_multiday, Y_test_time_multiday, 'Time Domain - Multi Day Binary');


[net_binary_timefreq_multiday, info_binary_timefreq_multiday] = train_binary_cnn(X_train_timefreq_multiday, Y_train_timefreq_multiday, ...

    X_test_timefreq_multiday, Y_test_timefreq_multiday, 'Time-Frequency Domain - Multi Day Binary');
```

```
% Display Results
disp('Evaluating Models...');

% Evaluate First Day Models
display_binary_results(net_binary_freq_firstday, X_test_freq_firstday, Y_test_freq_firstday, ...
    'Frequency Domain - First Day Binary');
display_binary_results(net_binary_time_firstday, X_test_time_firstday, Y_test_time_firstday, ...
    'Time Domain - First Day Binary');
display_binary_results(net_binary_timefreq_firstday, X_test_timefreq_firstday,
Y_test_timefreq_firstday, ...
    'Time-Frequency Domain - First Day Binary');

% Evaluate Multi Day Models
display_binary_results(net_binary_freq_multiday, X_test_freq_multiday, Y_test_freq_multiday,
...
    'Frequency Domain - Multi Day Binary');
display_binary_results(net_binary_time_multiday, X_test_time_multiday, Y_test_time_multiday,
...
    'Time Domain - Multi Day Binary');
display_binary_results(net_binary_timefreq_multiday, X_test_timefreq_multiday,
Y_test_timefreq_multiday, ...
    'Time-Frequency Domain - Multi Day Binary');

% Save all trained models
save('binary_cnn_models.mat', ...
    'net_binary_freq_firstday', 'net_binary_time_firstday', 'net_binary_timefreq_firstday', ...
    'net_binary_freq_multiday', 'net_binary_time_multiday', 'net_binary_timefreq_multiday', ...
    'authorized_participants');

disp('All models have been trained and saved.');
```