Title: Predicting Mental Health and Mood Swings Based on Demographic, Lifestyle, and Emotional Factors Using Deep Learning and Neural Networks

Abstract

The COVID-19 pandemic has exacerbated mental health challenges, with anxiety and depression surging globally. This study explores mood swings as an early indicator of mental health issues, utilizing a deep neural network to predict mood variability. The independent variables (predictors) we used to identify mood swings and mental health issues include demographic data such as age, gender, and employment status, which are known to influence mental health lifestyle factors such as daily habits, work-life balance, and social support levels and emotional states such as self-reported stress, anxiety, and depression levels. The Kaggle Mental Health Dataset was preprocessed by handling missing values, removing duplicates, encoding categorical features, and standardizing variables.
A ReLU-based neural network model was developed, utilizing demographic, lifestyle, and emotional state variables to uncover complex, non-linear relationships affecting mood swings. The study demonstrates strong predictive performance, achieving an accuracy of 83%. This research underscores the potential of deep learning and neural networks to enhance early diagnosis and personalized mental health interventions. Quantitative results and trends support the model's robustness, offering a data-driven approach to addressing mental health crises.

Introduction

In today's world, while remarkable advancements in science and technology mark human progress, pressing issues like mental health remain a growing concern. The COVID-19 pandemic has significantly worsened this situation. For instance, the World Health Organization (WHO) reported a 25% increase in the global prevalence of anxiety and depression during the pandemic's first year ("COVID-19 Pandemic Triggers 25% Spike"). This surge can be attributed to social isolation, economic insecurity, and healthcare disruptions (Panchal et al.). This stark reality underscores the urgent need to prioritize mental health awareness, access to care, and societal resilience.

Mood swings, a common and significant feature of various psychiatric disorders, serve as an early warning sign for potential mental health problems ("Adult Psychiatric Morbidity Survey"). Studies suggest that mood instability, particularly pronounced during adolescence, is associated with a variety of mental health conditions, including depression, anxiety, and bipolar disorder, and is linked to increased health service utilization and suicidal ideation (Marwaha et al.). Mood instability is reported in 40–60% of those with depression, anxiety disorder, post-traumatic stress disorder, and obsessive-compulsive disorder. It is associated with increased health service use and suicidal ideation, independent of neurotic symptoms, alcohol misuse, borderline personality disorder, and other confounders.

Building on this understanding, this paper leverages advancements in deep learning and neural networks to predict and quantify mood swings as a step toward addressing broader mental health concerns. Utilizing self-reported patient data, machine learning models are designed to identify predictive factors for mood instability, which has transdiagnostic potential as both an investigational and therapeutic target. By analyzing neurobiological correlates, prevalence data, and clinical characteristics, these models aim to improve early diagnosis and personalized intervention strategies. Such tools not only enhance our ability to monitor mental health conditions but also hold promise in mitigating the long-term impacts of psychiatric disorders, including bipolar disorder, borderline personality disorder, and psychotic disorders.

Integrating AI technology into mental health research exemplifies how data-driven approaches can bridge the gap between clinical understanding and proactive care. By focusing on mood instability as a key metric, we can advance therapeutic outcomes and contribute to a deeper understanding of mental health dynamics.

Materials and Methods

In this study, we utilized the Mental Health Dataset from Kaggle, which provides comprehensive data on mental health indicators, including self-reported emotional states, demographic information, and related lifestyle factors. This dataset offers a valuable foundation for exploring mental health patterns and predictive modeling, particularly in identifying mood swings. The Kaggle Mental Health Dataset contains survey responses on mental health issues, including emotional states, lifestyle habits, and personal well-being factors. Key insights from the dataset show that a significant portion of respondents report experiencing stress, anxiety, and depression. Additionally, factors such as age, employment status, and social support influence mental health outcomes. This data is ideal for machine learning models that predict mood swings and identify mental health patterns across diverse demographic groups. Below is a detailed description of the independent variables that predict mood swings, including their data characteristics. These variables were preprocessed (e.g., standardized or encoded) to enhance compatibility with the neural network and ensure robust predictions.

1. **Age**:
   - **Data Type**: Numerical, continuous.
   - **Example Values**: 18, 25, 45, 60.
   - **Description**: Captures the respondent's age. Young adults and older populations often show differing patterns of mood instability.
2. **Gender**:
   - **Data Type**: Categorical, encoded numerically (e.g., 0 = Male, 1 = Female, 2 = Other).
   - **Example Values**: 0, 1, 2.
   - **Description**: Identifies gender-based trends in mental health.
3. **Employment Status**:
   - **Data Type**: Categorical, encoded numerically (e.g., 0 = Unemployed, 1 = Employed, 2 = Self-employed).
   - **Example Values**: 0, 1, 2.
   - **Description**: Assesses how professional stability or lack thereof impacts mood.
4. **Emotional States (Stress, Anxiety, Depression)**:
   - **Data Type**: Numerical, scaled from self-reported surveys (e.g., 0–10).
   - **Example Values**: Stress = 7, Anxiety = 5, Depression = 8.
   - **Description**: Quantifies individual mental health conditions directly tied to mood variability.
5. **Lifestyle Factors**:
   - **Data Type**: Numerical, based on surveys or encoded metrics (e.g., hours of sleep, physical activity levels).
   - **Example Values**: Sleep = 6 hours, Physical Activity = 3 days/week.
   - **Description**: Measures behaviors influencing mental health.
6. **Occupation**:
   - **Data Type**: Categorical, encoded numerically (e.g., 0 = IT, 1 = Healthcare, 2 = Retail).

  - **Example Values**: 0, 1, 2.
  - **Description**: Accounts for work-related stress and its impact.
7. **Relationship Status**:
  - **Data Type**: Categorical, encoded numerically (e.g., 0 = Single, 1 = Married, 2 = Divorced).
  - **Example Values**: 0, 1, 2.
  - **Description**: Represents social support or isolation factors.

This research bridges deep learning methodologies with mental health analytics, demonstrating the potential for data-driven approaches to improve diagnostic precision and support mental health interventions.

Neural Network Architecture

Neural networks mimic how the human brain processes information to recognize patterns and make predictions. It consists of layers of interconnected nodes (neurons), starting with the input layer, which takes raw data (like pixels of an image). The data is passed through hidden layers, where each neuron uses mathematical functions to focus on specific features, such as edges or shapes. ReLU (Rectified Linear Unit) is commonly used here to make the network capable of capturing complex patterns by activating only relevant signals and ignoring the rest. Finally, the output layer provides the prediction, and feedback helps the network adjust its internal parameters to improve accuracy over time.

More specifically, a neural network works like a digital brain that learns through trial and error. It starts with random "weights" to determine how strong the connections between its "neurons" should be. In hidden layers, weighted sums of inputs are calculated, and activation functions like ReLU highlight patterns by introducing non-linearity. Input data flows through these neurons in layers, where calculations combine the inputs, apply functions like ReLU to highlight patterns, and generate a prediction. Data flows through input, hidden, and output layers and then compares this prediction to the actual value using a loss function to measure errors. Adjustments (backpropagation) refine the weights repeatedly through training cycles until the network learns to make accurate predictions. Training iterates over epochs, with data divided into batches for efficiency, refining predictions until the loss stabilizes.

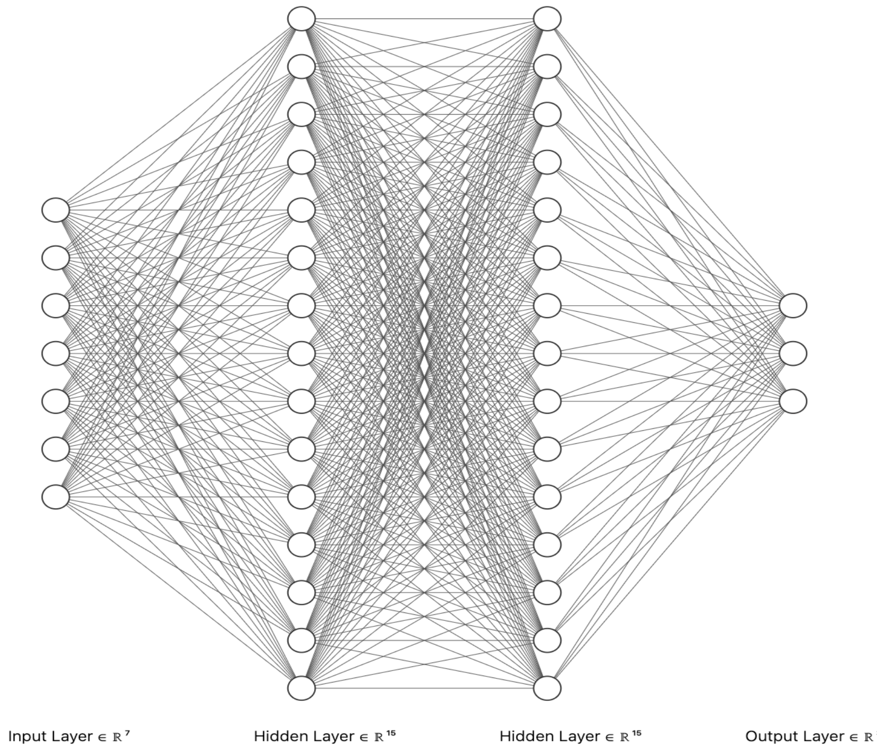


Figure 1: Design of a Neural Network

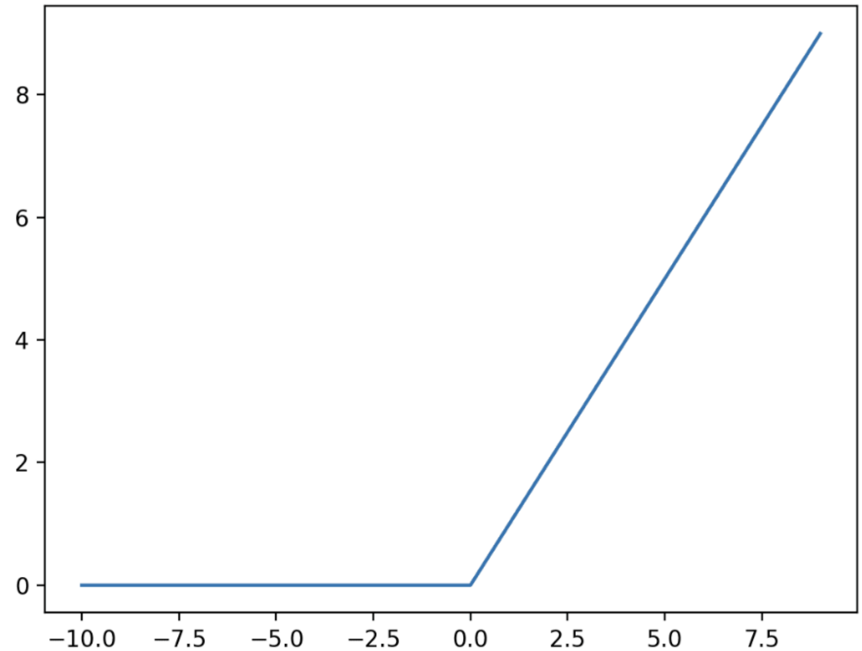


Figure 2: ReLU Activation Function

Below is a detailed summary of steps describing the mathematical intuition behind neural networks:

1. **Weight Initialization**: At the start, all connections (edges) between neurons are assigned small, random weights. These weights determine how much influence one neuron has on the next. Random initialization ensures that neurons learn different features.
2. **Feedforward Propagation**:
   - **Input Layer**: The raw input data is introduced to the network. Each feature is represented as a numerical value and fed into the input nodes.
   - **Hidden Layers**: Data flows sequentially through hidden layers. At each neuron, a weighted sum of the inputs is computed:

     $$z = \sum w_i x_i + b$$

     where w is the weight, x is the input, and b is the bias term.
   - **Activation Functions**: Non-linear activation functions, like $ReLU\ max(0, z)$, are applied to the weighted sum, introducing non-linearity. This enables the network to learn complex patterns.

- **Output Layer**: The final layer produces the network's predictions (y_{pred}) which might represent probabilities (for classification) or continuous values (for regression).

3. **Loss Function**: The loss function measures the difference between the network's prediction (y_{pred}) and the actual output (y_i). For instance, Mean Squared Error is common in regression tasks, while cross-entropy loss is typical in classification tasks. The goal is to minimize this loss.

4. **Backward Propagation (Backprop)**:
   - **Gradient Computation**: Gradients (partial derivatives) of the loss with respect to weights and biases are calculated using the chain rule. This determines how each weight contributes to the error.
   - **Weight Updates**: Using an optimization algorithm (e.g., stochastic gradient descent or Adam), weights and biases are updated to minimize the loss:
     $w_{new} = w_{old} - \eta . \frac{\partial L}{\partial w}$
     where , η is the learning rate, which controls the step size for the update, and $\frac{\partial L}{\partial w}$ is the gradient of the loss L with respect to the weight w is the learning rate.
5. **Training Iterations**:
   - Feedforward propagation, loss calculation, and backpropagation are repeated over multiple iterations (epochs). During each epoch, the entire dataset is passed through the network.
   - Data is divided into smaller subsets (batches) to manage computational efficiency. Each batch updates the model weights, ensuring faster convergence and less memory usage.
6. **Convergence**: Training continues until the loss stabilizes or validation metrics (e.g., accuracy or validation loss) stop improving, indicating that the model has learned as much as possible from the data.

Our Methodology

Neural networks are computational models inspired by the structure of the human brain, consisting of layers of interconnected neurons that process and learn from input data. Each neuron performs a mathematical operation on the input, and the network adjusts its internal parameters through training to minimize prediction errors. **ReLU (Rectified Linear Unit)** is a commonly used activation function in neural networks, introducing non-linearity and helping the model capture complex patterns by transforming negative values to zero while passing positive values unchanged. This makes the network capable of learning intricate relationships in data.

As part of our research, we leveraged a **ReLU neural network** model for predicting mood swings based on a mental health dataset. We followed the steps listed below:

1. Data Preprocessing: In the data preprocessing step, missing values in the 'self_employed' column are addressed by removing rows with null values. Duplicated rows are identified and removed to maintain the integrity of the dataset. Categorical features are then transformed into numerical values using **LabelEncoder**, allowing the neural network to process them. Finally, **StandardScaler** is applied to normalize the feature values, ensuring all input variables are on a similar scale, which improves model performance and convergence during training. These preprocessing steps help clean and prepare the data for effective model training.

1. **Handling Missing Values**: The dropna() function is used to remove rows containing missing values. Specifically, rows with null values in the 'self_employed' column are dropped to ensure the dataset remains complete and valid for analysis.
2. **Removing Duplicates**: The duplicated() function identifies duplicate rows, and drop_duplicates() removes these rows to prevent redundant data that could skew model performance.
3. **Label Encoding**: The LabelEncoder() function converts categorical variables (e.g., 'Gender', 'Occupation') into the numerical format by assigning each category a unique integer. This step is essential for neural networks, which require numeric input.
4. **Standard Scaling**: The StandardScaler() standardizes the features by removing the mean and scaling them to unit variance, ensuring that all features contribute equally during training. This prevents the model from being biased towards variables with larger ranges.

2. Model Development: We built a **sequential neural network with three hidden layers**. A sequential neural network is designed by stacking multiple layers of neurons, each performing a computation on the input data. In this model, three hidden layers with 50 neurons each allow the network to learn increasingly complex data representations. This architecture enables the model to capture deep patterns in the data that may not be obvious with simpler structures.

The **input features** in our dataset are: **Gender, Occupation, Family History, Days Indoors, Changes in Habits, Mental Health History,** and **Social Weakness.** The main target variable is Mood Swings. Therefore, the model consists of **7 input features**, **3 hidden layers (each with 50 neurons)**, and **1 output neuron**.

a. **Input Layer:** The neural network starts with the input layer, which takes in the 7 features. This forms a vector of size 7, representing the input to the first layer of the network.

$$X = \left[x_1, x_2, x_3, x_4, x_5, x_6, x_7\right]$$

Where:

- x_1=Gender
- x_2=Occupation
- x_3=Family History
- x_4=Days Indoors
- x_5=Changes in Habits
- x_6=Mental Health History
- x_7=Social Weakness

b. **Hidden Layers:** The network contains **three hidden layers**. **The ReLU** (Rectified Linear Unit) activation function (Figure 2 above) is used in the hidden layers to introduce non-linearity. This helps the model learn complex, non-linear relationships between the input features and the output, making it more powerful in capturing patterns that simpler linear models might miss. The first hidden layer is defined as below. It has 50 neurons and uses the ReLU activation function.

"layers.Dense(units=50, activation='relu', input_shape=[7])"

Each neuron in the first hidden layer calculates a weighted sum of the inputs, adds a bias, and applies the ReLU activation function. For the j-th neuron in the first hidden layer:

$$z_1^{(1)} = \sum_{i=1}^{7} w_{ij}^{(1)} . x_i + b_j^{(1)}$$

$$a_j^{(1)} = ReLU\left(z_j^{(1)}\right) = max\left(0, z_j^{(1)}\right)$$

Where:

- $w_{ij}^{(1)}$ is the weight from input i to neuron j in the first hidden layer.
- x_i is the input value for the i-th feature.
- $b_j^{(1)}$ is the bias for neuron j in the first hidden layer.
- $z_1^{(1)}$ is the pre-activation value for neuron j in the first hidden layer
- $a_j^{(1)}$ is the activation (output) of neuron j in the first hidden layer

This is repeated for all 50 neurons in the first hidden layer, forming an output vector $a_j^{(1)}$ of size 50. The second and third hidden layers are defined as below. Each has 50 neurons and uses the ReLU activation function

“layers.Dense(units=50, activation='relu')”

For the second and third hidden layers (with 50 neurons each), the process is similar:

$$z_k^{(l)} = \sum_{j=1}^{50} w_{jk}^{(l)} a_j^{(l-1)} + b_k^{(l)}$$

$$a_k^{(l)} = ReLU\left(z_k^{(l)}\right) = max\left(0, z_k^{(l)}\right)$$

Where

- $w_{jk}^{(l)}$ is the weight between the j-th unit in the previous layer and the k-th unit in the current layer.
- $a_j^{(l-1)}$ is the activation from the j-th unit in the previous layer (layer l - 1).

- $b_k^{(l)}$ is the bias term for the k-th unit in layer l.
- The summation runs over all 50 previous units.
- $a_k^{(l)}$ is the activation of the k-th unit in layer l, l is the layer index (l=2,3)
- **ReLU** (Rectified Linear Unit) applies the function $max\left(0,\ z_k^{(l)}\right)$, where values less than 0 are set to 0.

c. **The output layer** (with Softmax activation) calculates a weighted sum of the activations from the last hidden layer:

$$z_k^{(L)} = w_k^{(L)} a_k^{(L-1)} + b_k^{(L)}$$ for each output unit K = 1, 2 3

Where:

- $w_k^{(L)}$ is the weight connecting neuron j in the third hidden layer to the output neuron. $z_k^{(L)}$ is the sum of weighted inputs $w_k^{(L)}$ and $a_k^{(L-1)}$, with a bias $b_k^{(L)}$

The **Softmax** function is then applied to convert these logits into class probabilities

$$p_k = \frac{e^{z_k^{(L)}}}{\sum_{j=1}^{3} e^{z_j^{(L)}})}$$

- p_k is the probability of class k
- $z_k^{(L)}$ is the raw score for class k

The predicted class is the one with the highest probability:

$$\hat{y}_i = arg\ max\ (p1,\ p2,\ p3)$$

Where $\widehat{y_i}$ is the predicted class for the i^{th} sample.

3. Training & Validation: At a high level, in this step, the dataset is split into training and validation sets using **train_test_split**, typically allocating 80-90% of data for training and the remaining 10-20% for validation. **10 epochs** allow the model to iterate over the dataset multiple times, each epoch refining the model's weights. A **batch size of 40** means the model processes 40 samples at once before updating weights, balancing computational efficiency and effective learning. Monitoring **validation loss** ensures the model generalizes well, preventing overfitting if loss increases on the validation set. Below is the stepwise summary.

a. **Loss function:** The Adam optimizer adjusts the learning rate dynamically during training, improving model convergence and efficiency. This is combined with the Categorical Cross-Entropy (CCE) loss function, which penalizes the model for making significant errors in predictions. The loss function is a crucial component of the training process, as it quantifies how well the model's predictions match the actual labels. In this classification task, we use CCE since the model predicts three different classes.

$$\text{Loss}=\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C}\left(y_{ij}log\,(\widehat{y_{ij}})\right)$$

Where:

1. N is the number of samples in a batch.
2. C= Number of classes (3 in this case)
3. y_{ij} is the actual class label.
4. $(\widehat{y_{ij}})$ is the predicted value for class j.

b. **Optimization:** The Adam optimizer updates the weights and biases based on the gradients of the loss function:

$$\theta \leftarrow \theta - \eta \cdot \nabla_{\theta}\text{Loss}$$

Where:

- θ represents the parameters (weights and biases).

- η is the learning rate.
- ∇_θLoss is the gradient of the loss with respect to θ.

The choice of **10 epochs** is based on balancing adequate learning without overfitting. If training continues beyond a point where the validation loss stops improving, it could lead to diminishing returns, suggesting the model has learned as much as it can from the data.

4. Model Evaluation: After training the model, predictions $\widehat{y_i}$ on the validation set were compared to the actual values y_i. Key metrics used for evaluation include **accuracy**, **precision**, **recall**, and **F1-score**. These metrics are relevant because we are dealing with a classification problem and using **Categorical Cross-Entropy** as the loss function.

Accuracy is the most important metric in this case, as it measures how well the model predicts the correct class:

$$Accuracy = \frac{\sum_{i=1}^{N} 1\,(\widehat{y_i} = y_i)}{N}$$

Where:

- N is the total number of samples,
- $1\,(\widehat{y_i} = y_i)$ is an indicator function that equals 1 if the predicted class $\widehat{y_i}$ matches the true class y_i, and 0 otherwise.

Precision, Recall, and F1-Score are also computed for each class to understand how the model performs across all classes, especially if the class distribution is imbalanced.

A Loss vs. Validation Loss Plot is generated to visualize the model's learning process. A steady decrease in both training and validation loss is desirable, indicating that the model is effectively learning and generalizing to the validation set.

This methodology, combining robust preprocessing, mathematical rigor, and iterative optimization, allowed us to build a model capable of predicting mood swings from mental health data with high precision.

5. Fine-tuning the model: We also experimented with 14-7-7-3 (Model 1) neuron architecture and 50-50-50-3 (Model 2) neurons architecture to find the model with optimal performance. Overall, Model 2 has higher accuracy (83%) compared to Model 1 with 66% accuracy. The larger architecture (Model 2) allows it to capture more complex patterns, leading to better performance across all metrics. Model 1 is at risk of underfitting due to its smaller architecture, while Model 2 achieves stable, high performance with a good balance of precision, recall, and F1-Score. Model 2 also benefits from higher capacity, which enhances its ability to understand complex relationships, whereas Model 1 remains more basic and struggles to capture complex patterns effectively. Table of Results below.

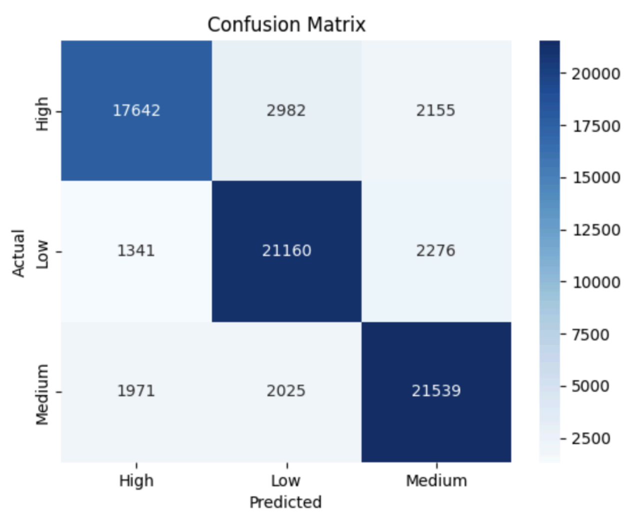
Metric	**Model 1 (14-7-7-3)**	**Model 2 (50-50-50-3)**
Architecture	*(14-7-7-3) 31 neurons*	*(50-50-50-3) 153 neurons*
Accuracy	*66%*	*83%*
Final Training Loss	*0.7191*	*0.3033*
Minimum Validation Loss	*0.7031*	*0.2983*
Validation loss trend	*Fluctuates after epoch 8*	*Steady decline, minor fluctuations*
Risk of Overfitting	*Low (underfitting risk due to lower complexity)*	*Possible (overfitting risk due to model complexity)*
Model Capacity	*Lower capacity, might miss some complex patterns*	*Higher capacity, better at capturing complex patterns*
Performance	*Struggles with complex patterns, lower accuracy*	*Better at capturing complex patterns, higher accuracy*

Results

The model training process showed significant improvement during the first 7 epochs, where training accuracy rose from 58.06% to 79.61%, and validation accuracy improved from 79.19% to 81.14%. After epoch 7, both training and validation accuracy plateaued around 82.28% and 82.56%, respectively, indicating that the model had largely converged. This suggests that while the model continued to improve, the gains were minimal after the 7th epoch, pointing to diminishing returns. The model avoided overfitting as validation accuracy remained stable, and training accuracy increased. Overall, training for 10 epochs is sufficient, with early stopping or learning rate adjustments potentially optimizing performance further without overfitting.

- **Accuracy:** The model achieved an accuracy of **83%** on the validation set, which is a good indication that it can generalize well to unseen data.
- **F1-Score:** The weighted F1-Score is **0.83**, indicating that the model has a balanced performance across the different classes. The F1-score considers both precision and recall, and this value suggests that the model performs consistently across both classes.
- **Precision:** The model achieved a precision of **0.83**, indicating that 83% of the positive predictions are correct. This is important for minimizing false positives.
- **Recall:** The model has a recall of **0.83**, meaning it correctly identifies 83% of the true positives. This is good as it shows that the model does not miss too many true instances.

Confusion Matrix



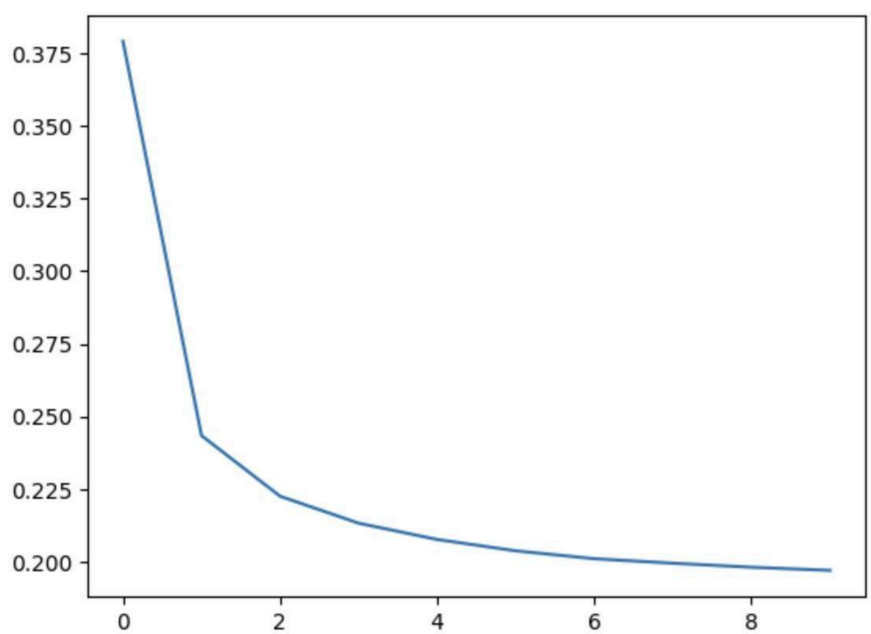
The confusion matrix reveals that the model performs well across all three classes, with the highest performance observed for **Class 2 (Mood Swing = 2)**, where both precision (0.83) and recall (0.84) are well-balanced. **Class 1 (Mood Swing = 1)** shows a strong recall of 0.85, capturing most true positives, but with slightly lower precision (0.81), indicating occasional misclassifications as class 1. **Class 0 (Mood Swing = 0)** has a precision of 0.84 and a recall of 0.77, suggesting that while the model is fairly accurate, it misses some instances of this class (higher false negatives). Overall, the model shows solid performance but could be fine-tuned to improve recall for class 0 without compromising performance for the other classes.

Classification Report:

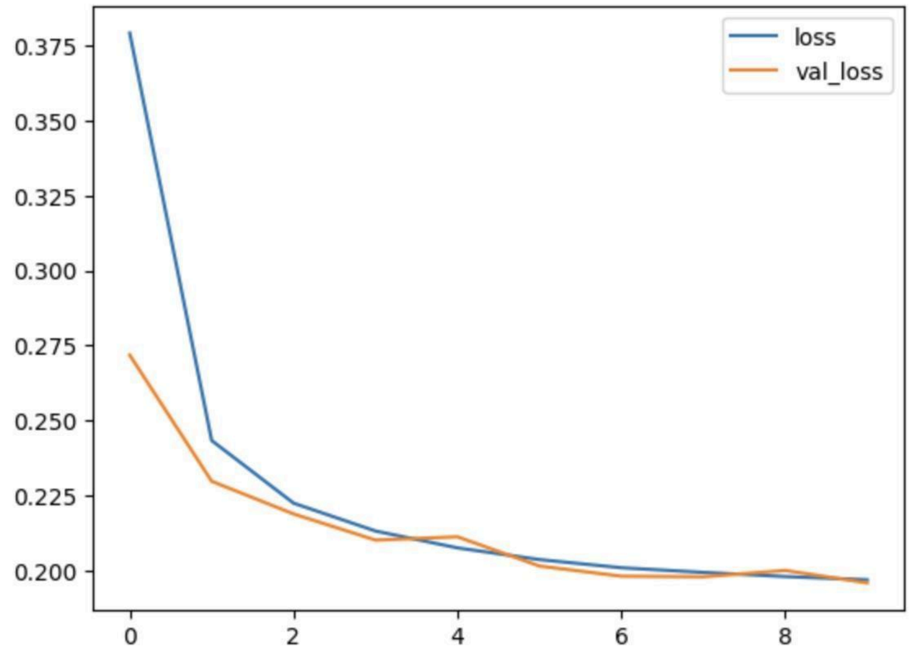
The classification report indicates that the model performs well across all three classes with a slight bias towards class 2.

- For class 0 (Mood Swing = 0), precision is **0.84** and recall is **0.77**.
- For class 1 (Mood Swing = 1), precision is **0.81** and recall is **0.85**.
- For class 2 (Mood Swing = 2), precision is **0.83** and recall is **0.84**. This shows the model is most accurate in predicting class 2.

The **training** and **validation loss curves** show a generally decreasing trend, indicating effective learning. Here are the key points:



Training Loss Curve: The training loss consistently decreases, indicating that the model is progressively learning and fitting the data.



Validation Loss Curve: The validation loss decreases and reaches a minimum of **0.2983** at epoch 9. After this, it fluctuates slightly, indicating that the model has reached near-optimal performance.

Sensitivity Analysis: The sensitivity analysis shows how the model's predicted probabilities for mood swing categories change as the input features are adjusted. As the 'mental health history' feature increases, the probability of mood swing category 0 (no mood swings) rises, indicating that a positive mental health history reduces the likelihood of mood swings, indicating a relationship between improved mental health history and the likelihood of mood swings. However, this relationship is more complex than a simple linear decrease and depends on how the other features influence the model.

Mental Health History	Mood Swing 0 Probability	Mood Swing 1 Probability	Mood Swing 2 Probability	Observation
0.0	0.7018	0.2434	0.0549	Lower probability for mood swing 0, higher for 1

0.1	0.8644	0.0869	0.0488	Increased probability for mood swing 0, decreased for 1
0.2	0.9464	0.0418	0.0118	Significant rise in mood swing 0 probability
0.3	0.9785	0.0204	0.0011	Further increase in mood swing 0 probability, almost no probability for 2
0.4	0.9811	0.0182	0.0008	Steady high probability for mood swing 0, low for 1 and 2

Discussion and Conclusion

In conclusion, this study highlights the potential of integrating AI-driven neural networks into mental health research to predict and address mood swings, a significant marker of psychiatric conditions. The neural network model demonstrates strong performance, with an accuracy of 83% and high scores for precision, recall, and F1-score. It shows balanced performance across all classes, though it slightly favors class 2. The model is successful at predicting mood swings based on the features provided, and the sensitivity analysis confirms that changes in inputs influence the model's predictions appropriately. Further improvements could focus on enhancing the model's ability to distinguish between classes 0 and 1, possibly by exploring advanced techniques such as hyperparameter tuning or adding more features.

These results underscore the promise of leveraging machine learning to advance mental health diagnostics and foster personalized care approaches to improve societal well-being.

References

- **Marwaha, Steven, et al.** "Mood instability as a transdiagnostic marker of mental health problems." *Journal of Clinical Psychiatry*, vol. 75, no. 3, 2014, pp. 305–312.
- **Panchal, Nirmita, et al.** "The Implications of COVID-19 for Mental Health and Substance Use." *Kaiser Family Foundation*, 21 Feb. 2021, www.kff.org/mental-health.
- "COVID-19 Pandemic Triggers 25% Spike in Prevalence of Anxiety and Depression Worldwide." ***World Health Organization***, 2 Mar. 2022, www.who.int/news-room.
- **"Adult Psychiatric Morbidity Survey."** *NHS Digital*, 2014, digital.nhs.uk/data-and-information.
- **Kaggle**. "Mental Health Dataset." *Kaggle*, www.kaggle.com/datasets.
- **Vahratian, Anjel, et al.** *"Symptoms of Anxiety or Depressive Disorder and Use of Mental Health Care Among Adults During the COVID-19 Pandemic — United States, August 2020–February 2021."* Morbidity and Mortality Weekly Report (MMWR), vol. 70, no. 13, 2021, pp. 490–494. Centers for Disease Control and Prevention (CDC), www.cdc.gov/mmwr.
- **Patel, Vikram, et al.** *"Mental Health of Young People: A Global Public-Health Challenge."* The Lancet, vol. 369, no. 9569, 2007, pp. 1302–1313.
- **Torjesen, Ingrid.** *"COVID-19: Mental Health Services Face 'Tsunami' of Cases as Lockdown Eases."* BMJ, vol. 369, 2020, m1994, www.bmj.com/content/369/bmj.m1994.
- **American Psychological Association (APA).** *Stress in America: The State of Our Nation.* American Psychological Association, 2017, www.apa.org/news/press/releases/stress/2017/state-nation.
- **World Health Organization (WHO).** *Depression and Other Common Mental Disorders: Global Health Estimates.* World Health Organization, 2017, apps.who.int/iris/handle/10665/254610.
- **Mental Health Foundation.** *The COVID-19 Pandemic and Mental Health: Impact and Coping Strategies.* Mental Health Foundation, 2021, www.mentalhealth.org.uk.
- **National Institute of Mental Health (NIMH).** *Mental Illness Statistics and Prevalence.* National Institutes of Health, 2021, www.nimh.nih.gov.
- **Office for National Statistics (ONS).** *Coronavirus and Depression in Adults, Great Britain: June 2020.* Office for National Statistics, 2020, www.ons.gov.uk.