CLOUD x LAB

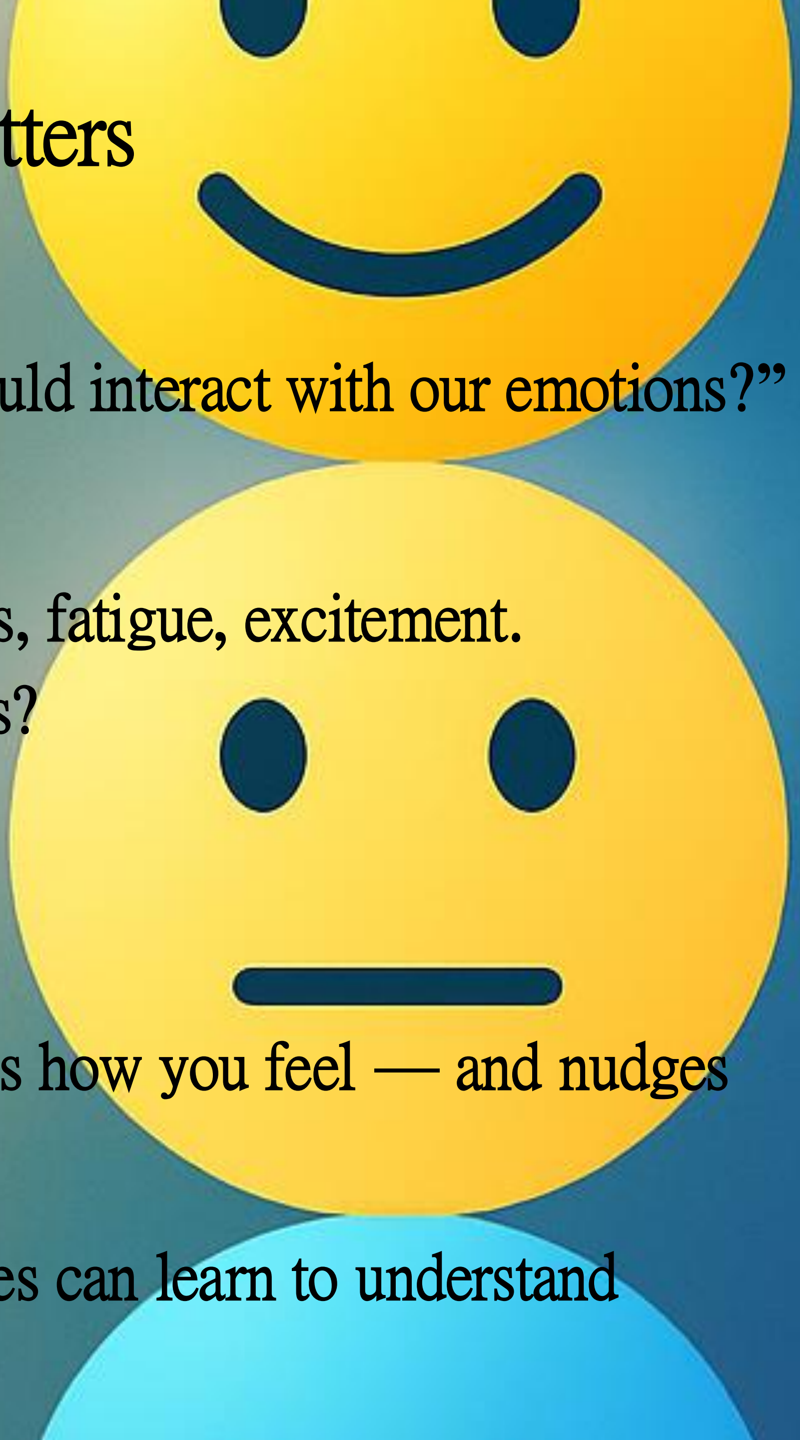
EMOTIONS IN MOTION

MOOD PREDICTION ENGINE

Prince Raj

August 2025

The Story Begins – Why This Project Matters



- “We interact with technology daily. But what if tech could interact with our emotions?”
- Every day, we go through different moods — joy, stress, fatigue, excitement.
  But what if our digital world could *sense* those changes?
- I imagined a world where your smart device understands how you feel — and nudges you toward better habits.
- That’s how this project was born: To explore if machines can learn to understand human mood through daily behavior.

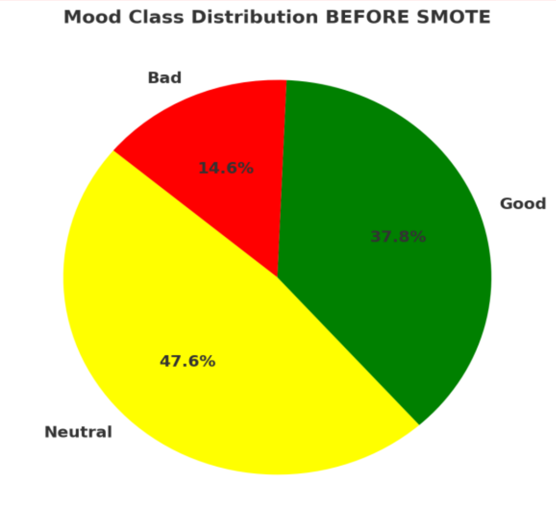
What This Project Aims to Do

I built a machine learning system that classifies human mood into Bad, Neutral, or Good using activity patterns and emotional signals like stress, energy, and intensity.

The vision: A future where mood-aware AI supports wellness, mental health, and productivity. Embed mood intelligence into smart devices, apps, wearables.

The Dataset – Fit-Life Simulation

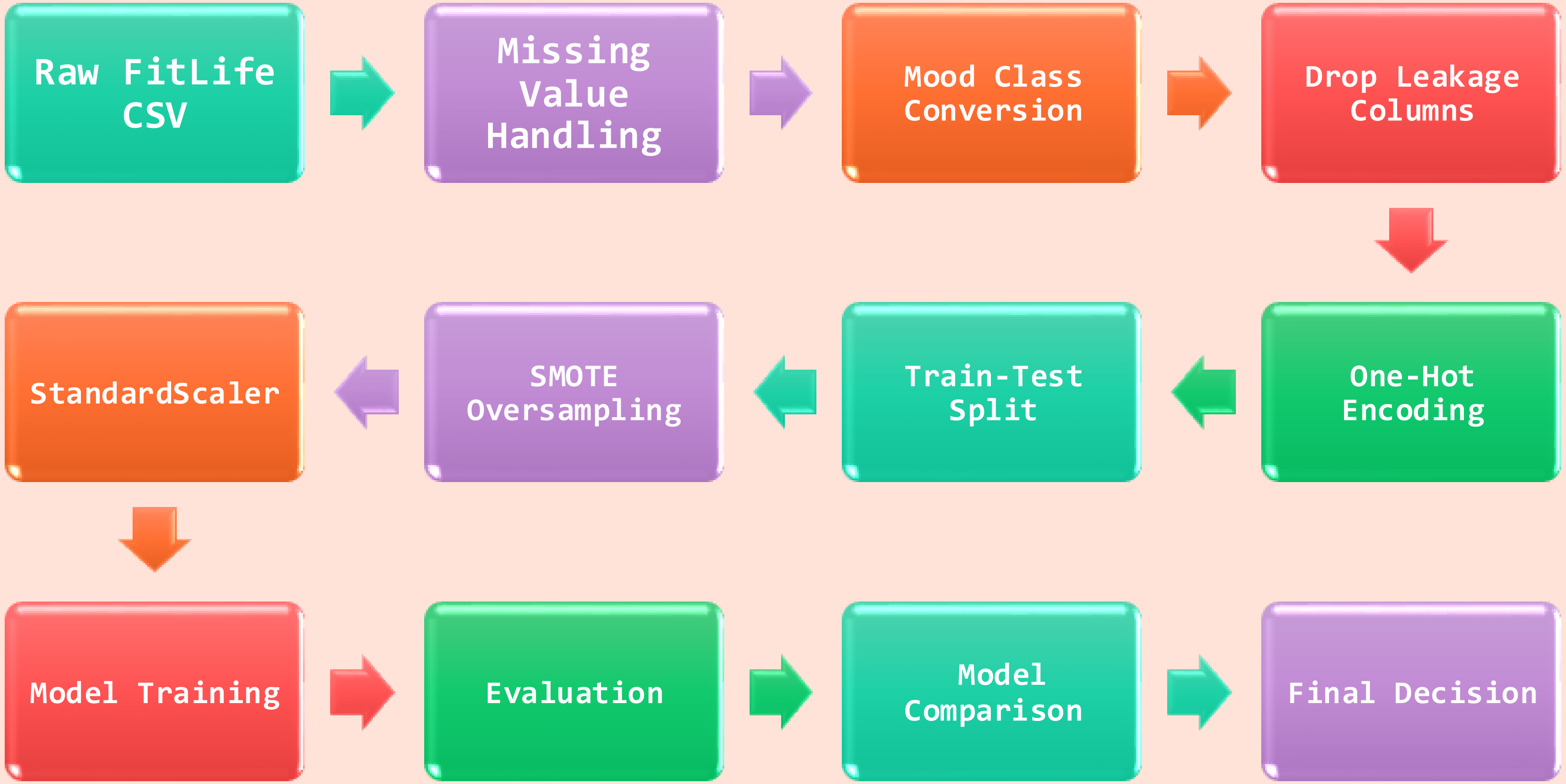
- I worked with a synthetic behavioral dataset of over 100,000 rows capturing how people live, move, and feel throughout their day.
- Key features included:
- Age, Gender, Employment
- Type of Activity, Duration, Intensity
- Mood Before/After, Energy, Stress
- I converted Mood (1–10 scale) into 3 classes: Bad (1–3), Neutral (4–6), Good (7–10)



Age	Gender	Employment Status	Time of Day	Activity Category	Sub-Category	Activity	Duration (minutes)	Intensity	Primary Emotion	Secondary Emotion	Mood Before (1-10)	Mood After (1-10)	Energy Level (1-10)
34	Male	Retired	Afternoon	Physical - Exercise	Mind-Body	Stretching	79	High	Energized	Challenged	3	3	6
30	Prefer not to say	Retired	Night	Physical - Sports	Extreme Sports	Bungee Jumping	72	High	Accomplished	Accomplished	6	7	4
75	Female	Employed	Night	Relaxation	Entertainment	Music Listening	117	High	Energized	Recharged	7	8	7

Project Pipeline

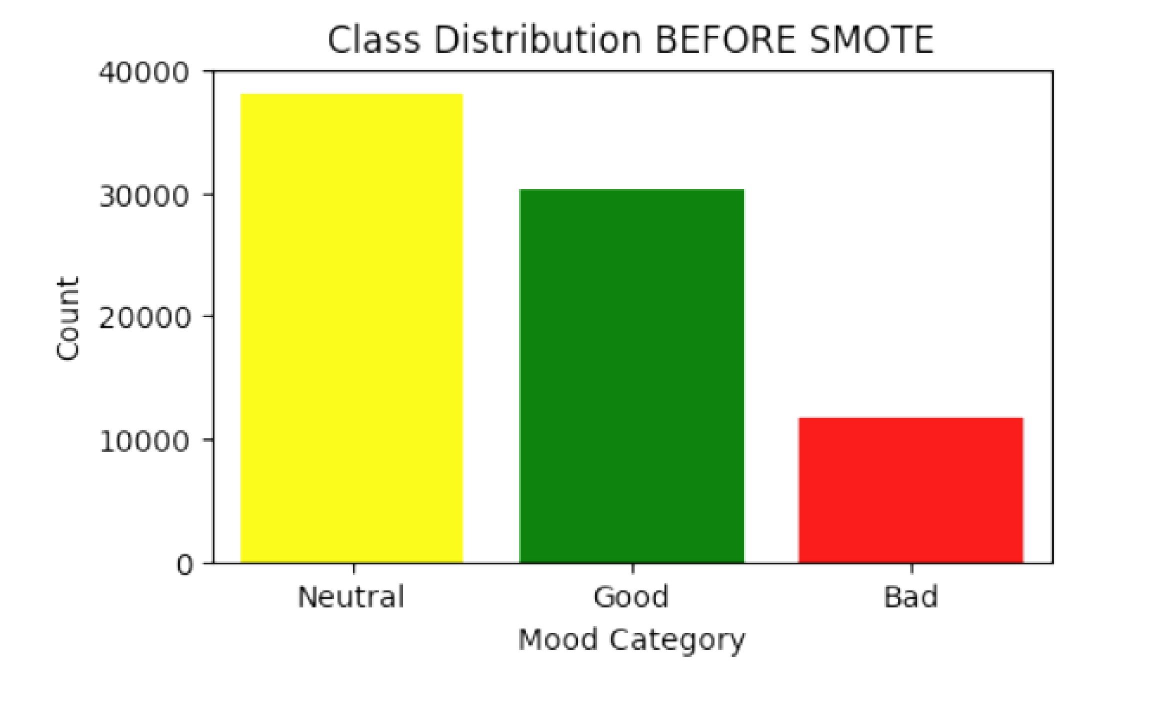
- Let me show you the complete pipeline — from raw data to trained model.



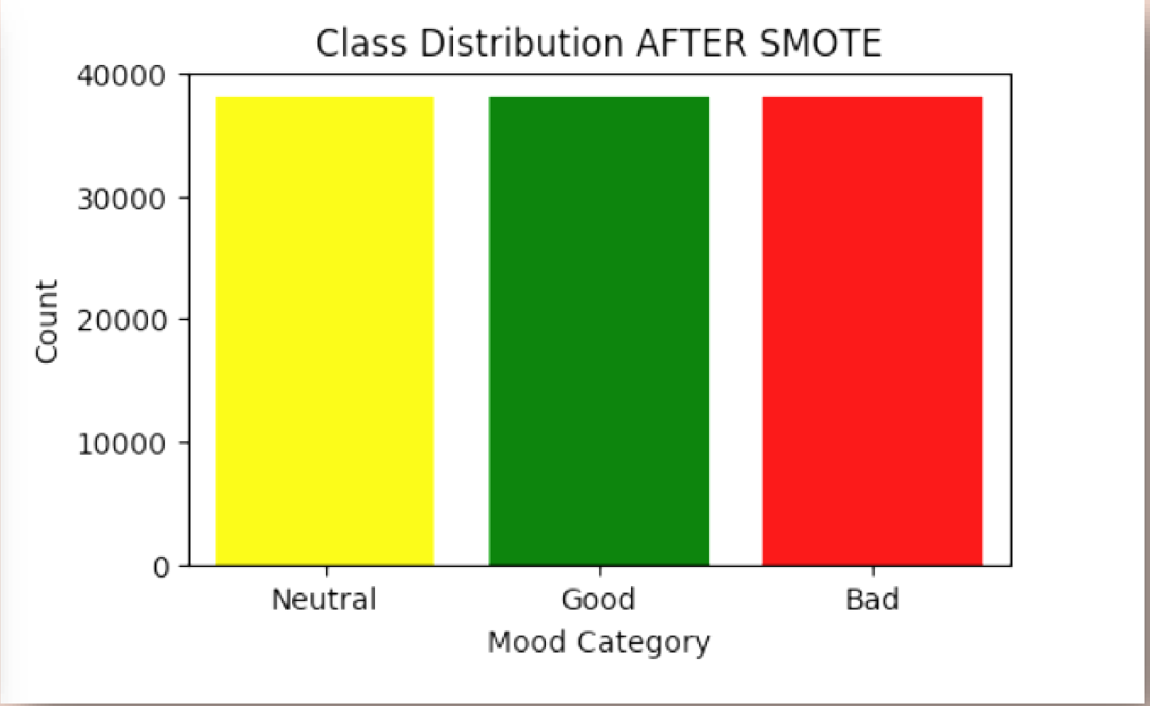
SMOTE Handling

- The original dataset had very skewed mood distribution — Neutral was dominant. So, I used SMOTE (Synthetic Minority Over-sampling Technique) to balance the training data.

Before SMOTE



After SMOTE



Feature Engineering

I dropped unnecessary columns like Date and Mood After to avoid leakage.

Then applied One-Hot Encoding → ended with 317 features.

For numeric fields like Age, Mood Before, Stress, I applied Standard Scaler - essential for deep learning models to converge properly.

Feature Engineering

```
[10]: df.head()
```

[10]:

	Age	Gender	Employment Status	Time of Day	Activity Category	Sub-Category	Activity	Duration (minutes)	Intensity	Primary Emotion	Secondary Emotion	Mood Before (1-10)	Energy Level (1-10)	Stress Level (1-10)
0	34	Male	Retired	Afternoon	Physical - Exercise	Mind-Body	Stretching	79	High	Energized	Challenged	3	6	9
1	30	Prefer not to say	Retired	Night	Physical - Sports	Extreme Sports	Bungee Jumping	72	High	Accomplished	Accomplished	6	4	9
2	75	Female	Employed	Night	Relaxation	Entertainment	Music Listening	117	High	Energized	Recharged	7	7	6
3	28	Prefer not to say	Retired	Night	Relaxation	Leisure	Baking	78	High	Accomplished	Serene	7	8	9
4	75	Female	Student	Afternoon	Physical - Sports	Team Sports	Basketball	55	Medium	Flexible	Challenged	7	6	2

```
[11]: print(f"Total Rows: {df.shape[0]}\n Total Columns: {df.shape[1]}")
```

```
Total Rows: 100000
 Total Columns: 15
```

Before encoding

```
[23]: X_train_resampled[:5]
```

[23]:

	Age	Duration (minutes)	Mood Before (1-10)	Energy Level (1-10)	Stress Level (1-10)	Gender_Female	Gender_Male	Gender_Non-binary	Gender_Prefer not to say	Employment Status_Employed	...	Secondai Emotion_Resilier
0	0.791086	0.388994	-0.566027	-0.318153	-1.237173	False	False	True	False	False	...	Fals
1	1.506768	-1.618360	1.622729	-0.915556	-0.307327	False	False	False	True	False	...	Fals
2	-0.695330	-0.878809	-0.566027	1.474056	1.552367	False	True	False	False	False	...	Fals
3	-1.631221	-0.825984	1.622729	0.279250	0.157597	True	False	False	False	True	...	Fals
4	1.286558	-0.667508	1.622729	-1.512959	-1.237173	False	False	True	False	False	...	Fals

5 rows × 317 columns

After One-hot encoding / StandardScaler

Model 1 – Random Forest

- First, I trained a Random Forest to get quick, interpretable results.
- Accuracy: 72.4%
- Best at predicting Neutral / Good mood

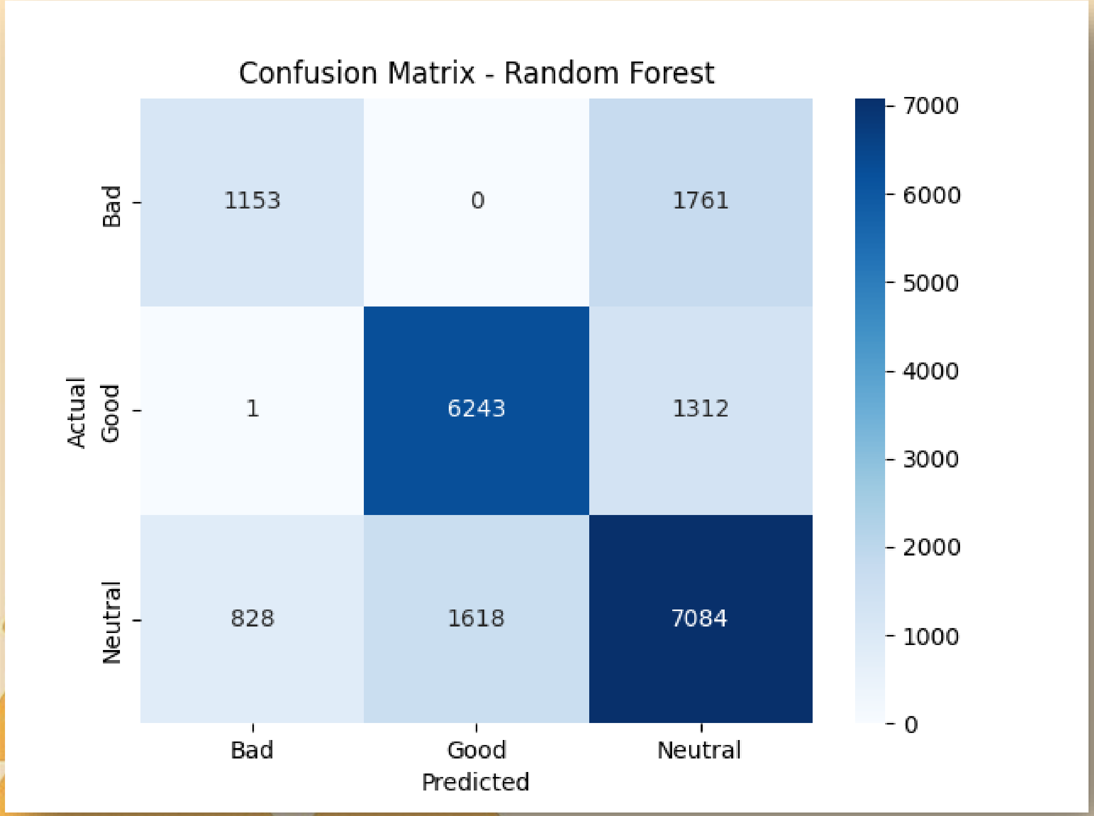
Classification Report

Random Forest Classification Report:

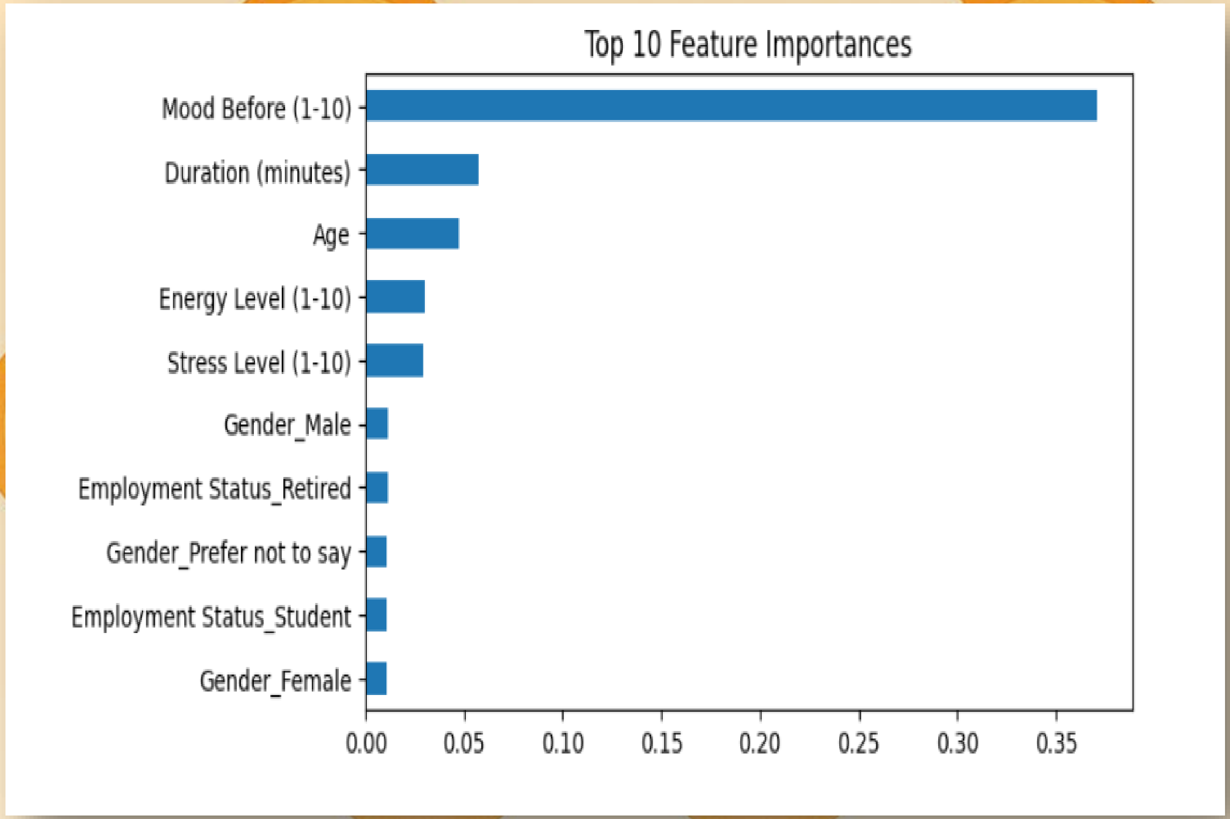
	precision	recall	f1-score	support
Bad	0.58	0.40	0.47	2914
Good	0.79	0.83	0.81	7556
Neutral	0.70	0.74	0.72	9530
accuracy			0.72	20000
macro avg	0.69	0.66	0.67	20000
weighted avg	0.72	0.72	0.72	20000

Model 1 – Random Forest

Confusion Matrix – RF



Top 10 Features – RF



Model 2 – XGBoost

- Then I trained XGBoost — A smarter learner with better interaction capture.
- Accuracy: 74.1%
- Best at predicting Bad mood

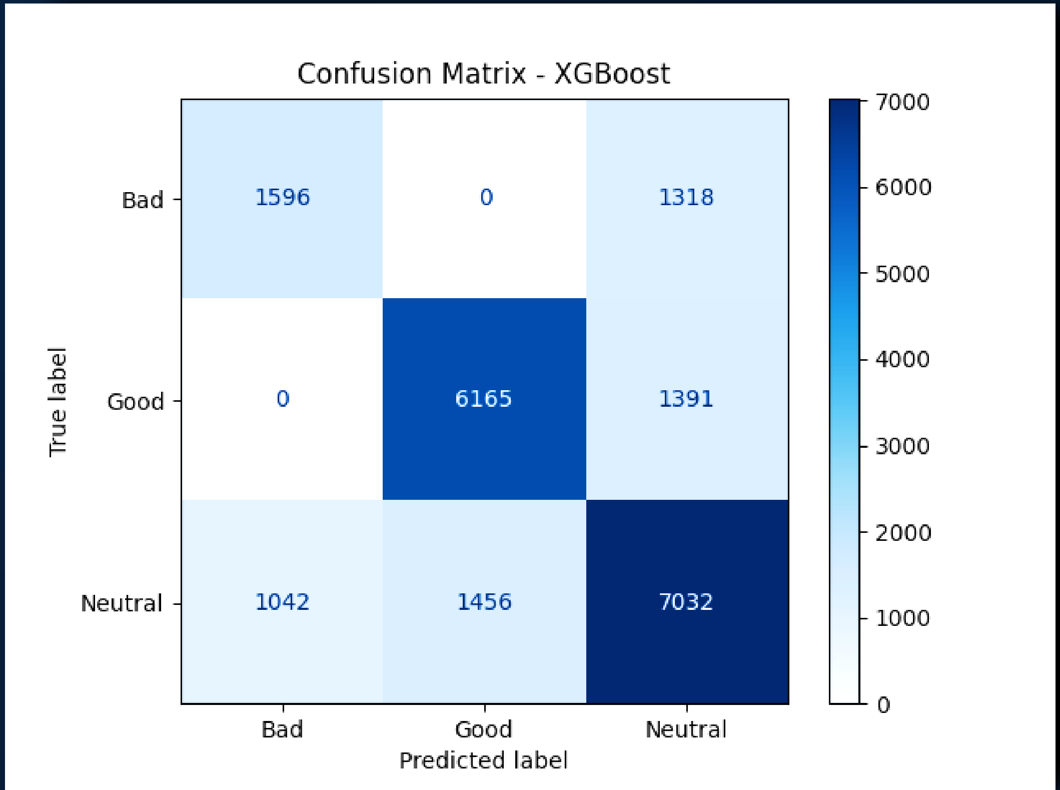
Classification Report

XGBoost Classification Report:

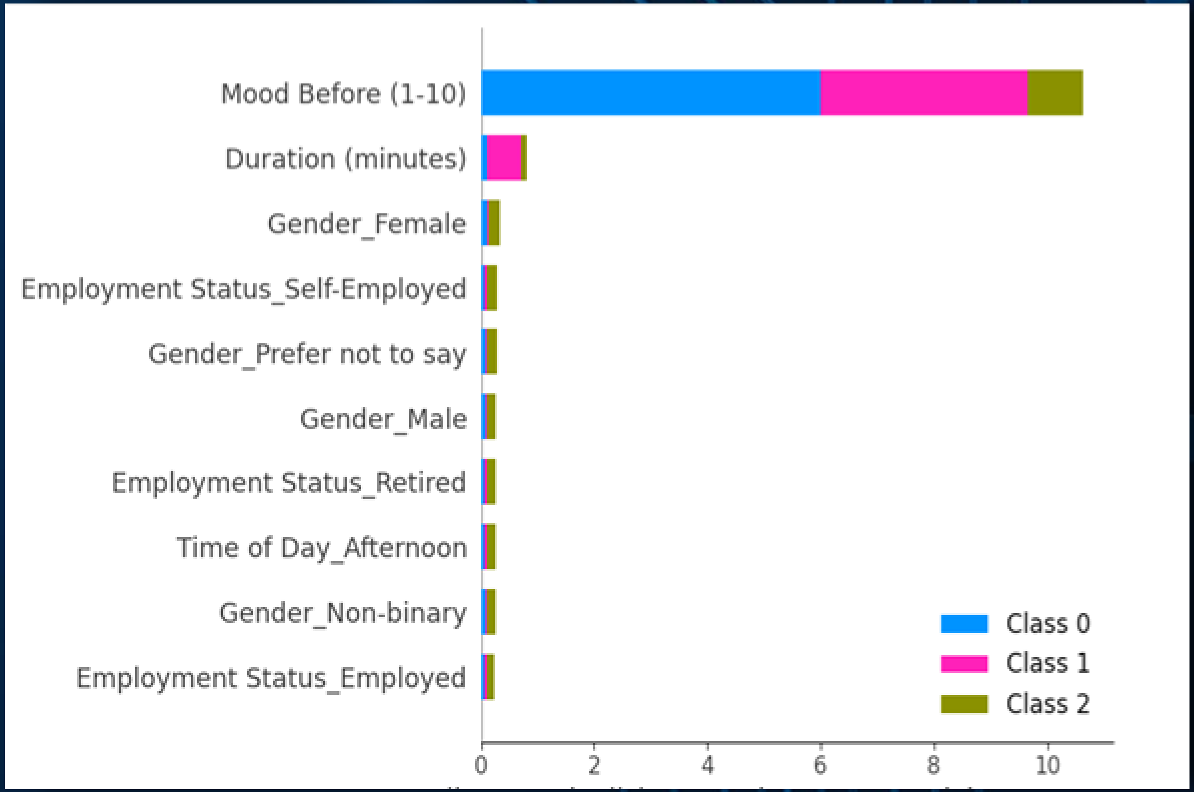
	precision	recall	f1-score	support
0	0.61	0.55	0.57	2914
1	0.81	0.82	0.81	7556
2	0.72	0.74	0.73	9530
accuracy			0.74	20000
macro avg	0.71	0.70	0.71	20000
weighted avg	0.74	0.74	0.74	20000

Model 2 – XGBoost

Confusion Matrix



Feature Contributions (Top 10 – SHAP summary)



- SHAP reveals 'Mood Before', 'Duration', and 'Gender' as key signals.

- XGBoost showed slightly better sensitivity to 'Bad' mood than Random Forest.

Model 3 – Deep Neural Network

- Finally, I trained a DNN using batch normalization, dropout, and early stopping.
- Accuracy: 74.2%
- More consistent across all mood classes

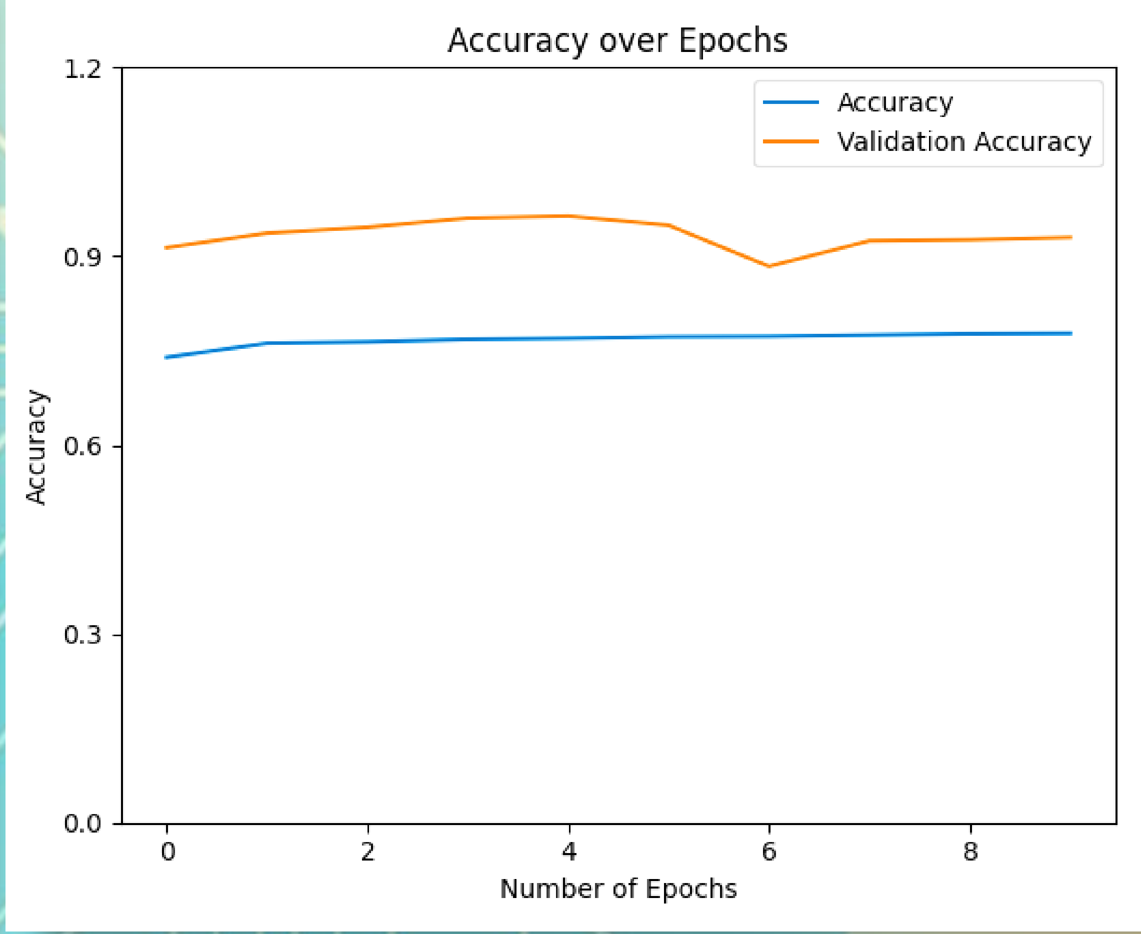
Model: "sequential"

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 256)	81,408
batch_normalization (BatchNormalization)	(None, 256)	1,024
dropout (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 128)	32,896
batch_normalization_1 (BatchNormalization)	(None, 128)	512
dropout_1 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 64)	8,256
batch_normalization_2 (BatchNormalization)	(None, 64)	256
dropout_2 (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 32)	2,080
dense_4 (Dense)	(None, 3)	99

Total params: 126,531 (494.26 KB)
Trainable params: 125,635 (490.76 KB)
Non-trainable params: 896 (3.50 KB)

DNN with 126K trainable parameters across 5 dense layers and regularization components.

Model 3 – Deep Neural Network



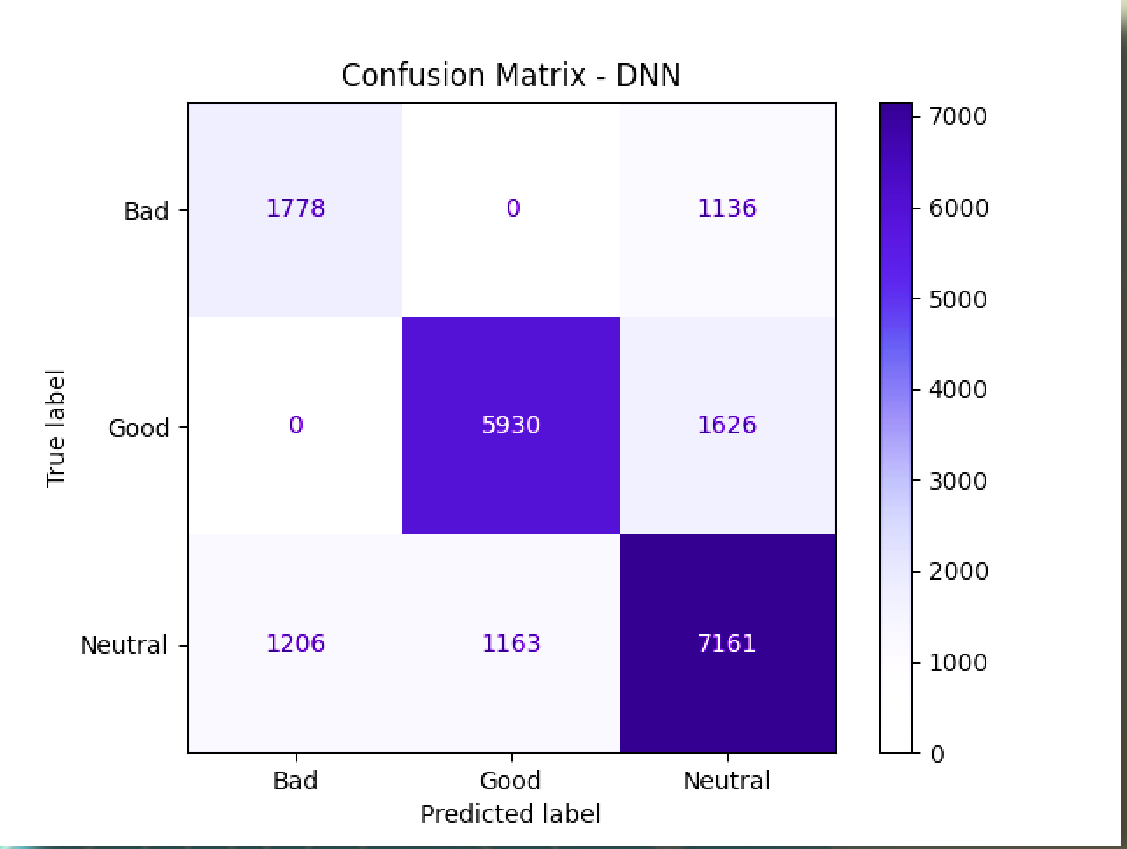
Model showed stable training with minor variance in validation performance across epochs.

Model 3 – Deep Neural Network

Classification Report

DNN Classification Report:

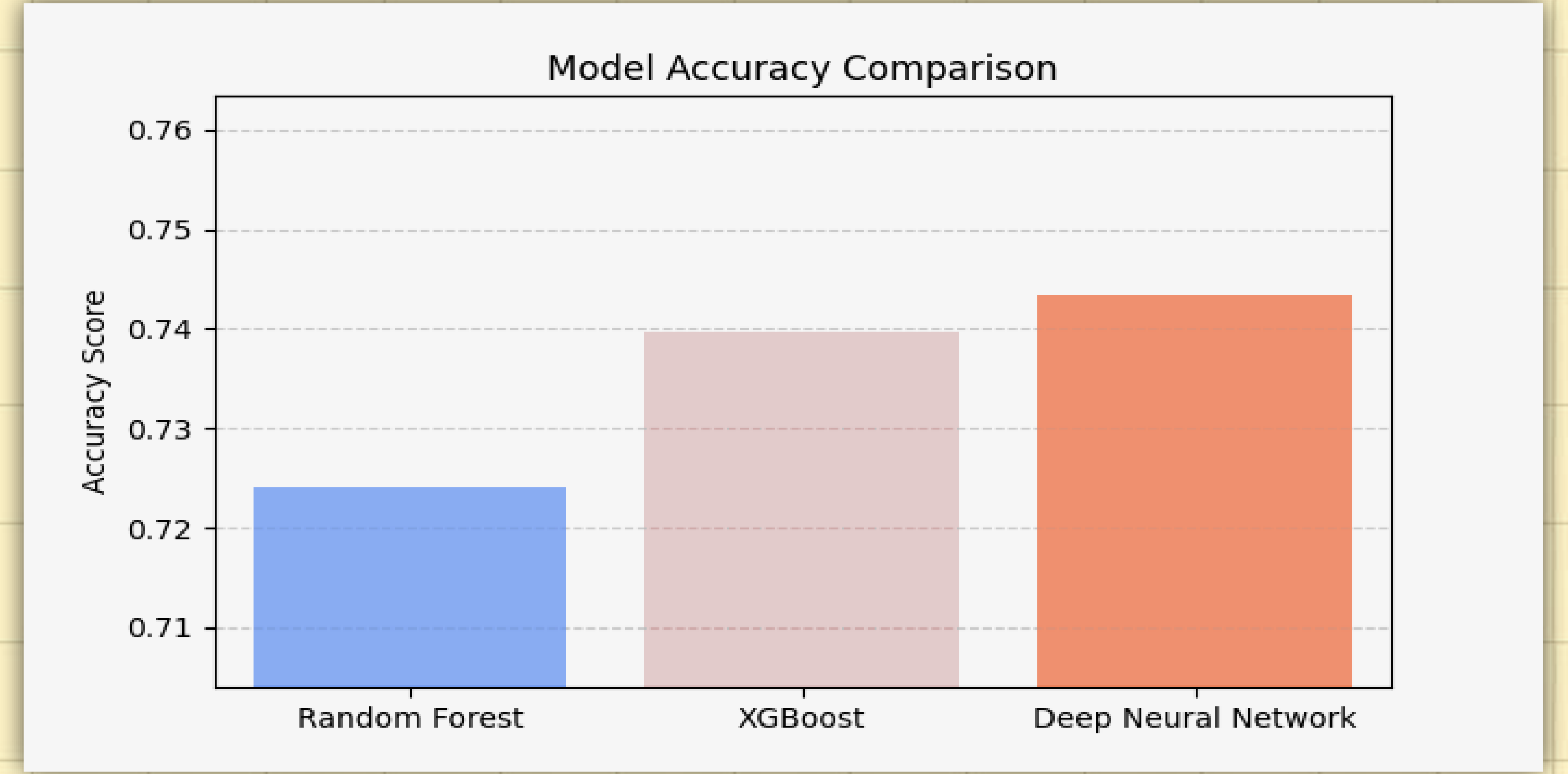
	precision	recall	f1-score	support
Bad	0.60	0.61	0.60	2914
Good	0.84	0.78	0.81	7556
Neutral	0.72	0.75	0.74	9530
accuracy			0.74	20000
macro avg	0.72	0.72	0.72	20000
weighted avg	0.75	0.74	0.74	20000



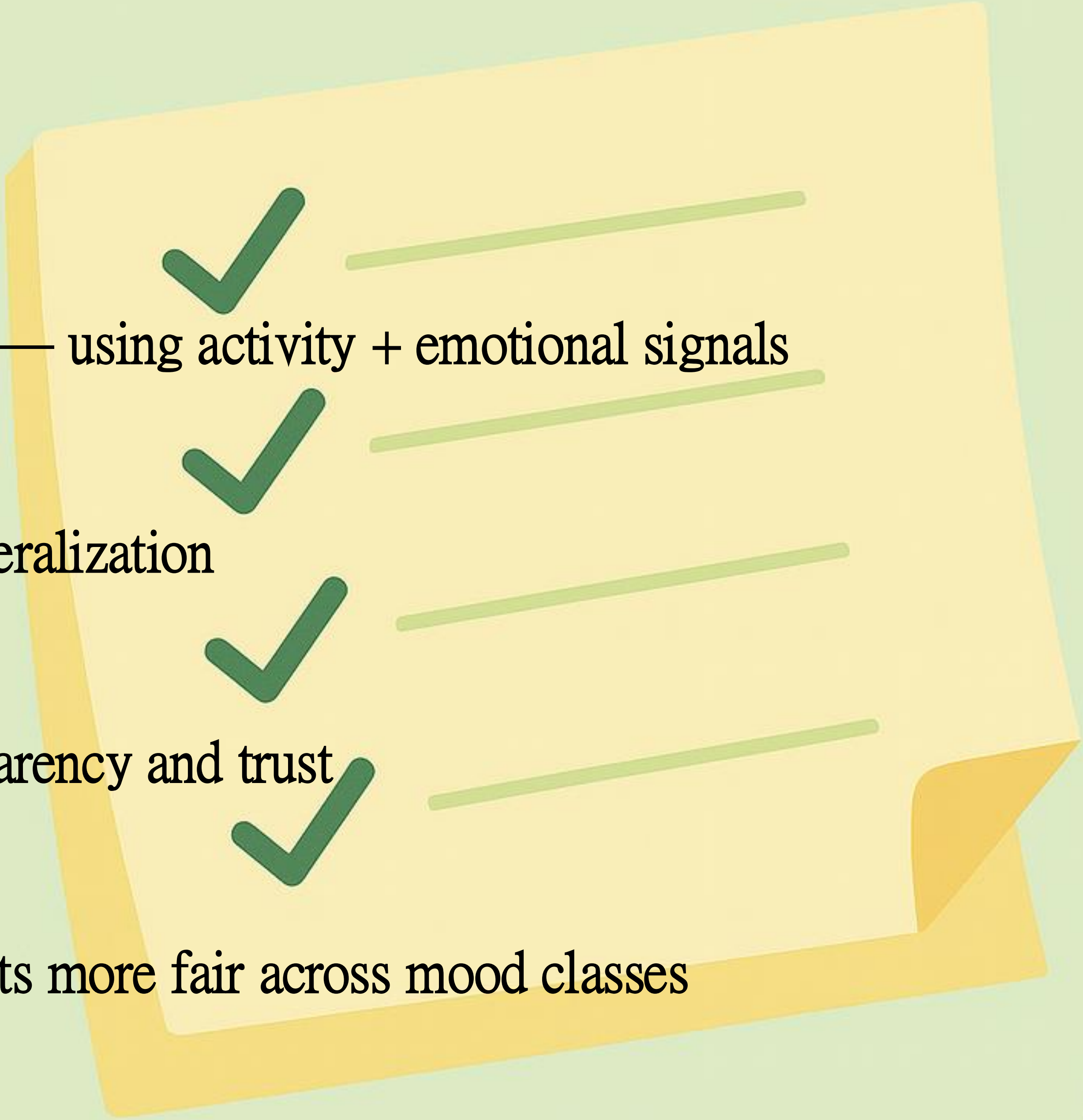
Confusion Matrix

Model Comparison & Summary

- All three models performed well — with DNN slightly ahead in overall accuracy.
- Random Forest: ~72.4%
- XGBoost: ~74%
- DNN: ~74.2%



Final Takeaways



- Mood *is* predictable — using activity + emotional signals
- DNN gave best generalization
- SHAP added transparency and trust
- SMOTE made results more fair across mood classes

Future Vision

- In future versions, this engine could:
  - Predict energy + stress along with mood
  - Be embedded into wearables or wellness apps
  - Adapt to real-time mood shifts using LSTM

Imagine your smartwatch saying:

"Hey , you've been low all morning. Want to stretch or take a walk?"

GitHub & Contact

- Project Notebook, Graphs, and Code:
- 🔗 GitHub

- Connect with me:
- 🔗 LinkedIn

Thank You

Let's build AI that not only understands emotions — but truly cares.