جامعة الطائف
TAIF UNIVERSITY

MACHINE LEARNING PROJECT

Mental Health and Lifestyle

Prepared for

Al-Batool Mohammed Abu Al-Nada
ID : 4151371
CLASS : 4232

Prepared by

Dr. NADA Al-twarqi
May 19, 2025

What is your name of Data ?

Mental Health and Lifestyle Dataset

The source of the data (which database)?

The dataset was published on Zenodo – a reputable research data repository.

Link to the original data?

https://zenodo.org/records/14838680

Explain the data in words:

This dataset contains mental health survey responses from 50,000 individuals. It includes demographic features such as age, gender, and country, as well as lifestyle indicators like sleep hours, work hours, physical activity, diet quality, and more. The target variable " Mental_Health_Condition " indicates whether an individual reported having a mental health condition (Yes/No).

Is it a regression or classification problem?

It is a binary classification problem – the model predicts either " Yes " or " No" for the mental health condition.

How many attributes?

There are 16 usable attributes after removing unnecessary columns.

How many samples?

The dataset contains 50,000 samples.

What are the properties of the data? (statistics)

Most numerical features (e.g., Age, Sleep_Hours, Work_Hours) follow a roughly normal distribution. Categorical variables like Gender and Country contain multiple levels. The dataset is mostly balanced in terms of the target variable.

Are there any missing data? How did you fill in the missing values?

Yes , the " Severity " column had 50% missing values and was dropped . All other columns had complete data .

Visualize the data :

To better understand the dataset, we used Seaborn and Matplotlib to generate several visualizations :

- **Bar Plot** (Mental Health Risk Count) :

A colored bar plot was created to show the number of individuals labeled as at risk or not at risk of mental health issues , the distribution appeared to be relatively balanced , indicating that the dataset is suitable for binary classification without severe class imbalance as shown figure

(1) .

- Correlation Heatmap :

We generated a heatmap to visualize the correlation between all numeric features . A strong negative correlation was observed between " depression score " and " productivity score " (-0.94) , which is logical - higher depression is associated with lower productivity also , " mental_health_risk " , showed moderate correlation with " productivity score " and " depression score " look figure (2) .

- Histogram (Stress Level Distribution) :

A histogram displayed the distribution of stress levels among participants . The values range from 1 to 10 and are almost evenly distributed , meaning that stress levels are diverse and not skewed as shown figure (3) .

These visualizations helped us confirm the quality and diversity of the data and guided the feature selection for model training .

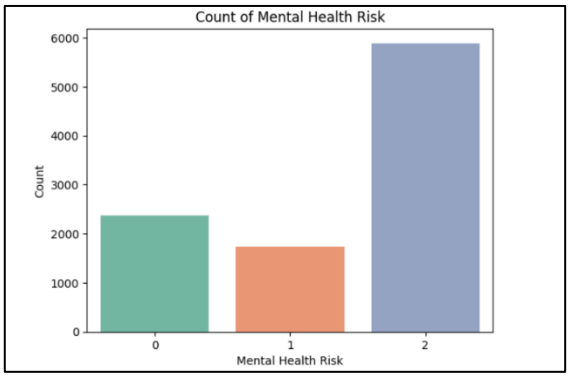

Figure (1) Count of Mental Health Risk

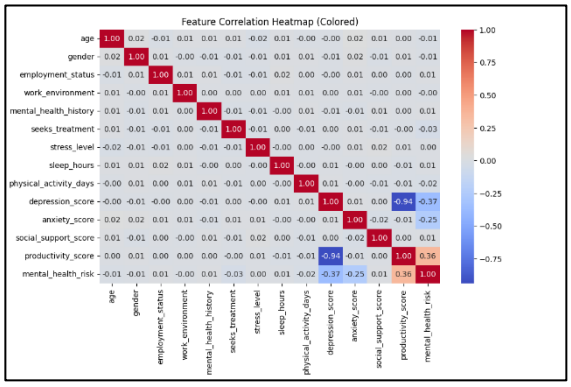

Figure (2) : Feature Correlation Heatmap

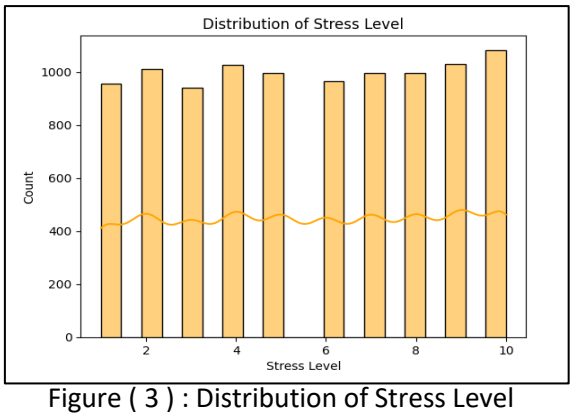

Figure (3) : Distribution of Stress Level

Did you normalize or standardize any of your data? Why?

No normalization or standardization was applied because the models used (like decision trees and random forest) do not require feature scaling . Additionally, categorical features were encoded and numerical values were already in interpretable ranges.

What type of preprocessing did you apply to your data? List everything and explain why :

- **Dropped Columns** : We dropped the " User_ID " and " Severity " columns , the first was an identifier and the second had more than 50% missing values.

- **Label Encoding :** All categorical variables (Gender , Occupation , etc.) were converted to numerical values using **Label Encoder** to be usable by ML models .

- **Target Selection :** " Mental_Health_Condition " was selected as the target variable

(Yes = 1, No = 0).

- **Data Splitting :** The dataset was split into 70% training and 30% testing using

" train_test_split( ) " .

How did you divide the train and test data? What are the proportions?

- Training Data : 70% of the data (35,000 samples)

- Testing Data :30% of the data (15,000 samples)

Apply all the machine learning models and report the results what is the best/worst performing model? Why?

We applied the following models :

1. Decision Tree

2. Random Forest

3. K-Nearest Neighbors (KNN)

4. Naive Bayes

5. Support Vector Machine (SVM)

6. Artificial Neural Network (ANN)

7. Linear Regression (used as classifier)

- Best Model : **KNN** with 51.31% accuracy , It performed slightly better due to its reliance on distance metrics which suited the structured numeric nature of the data.

- Worst Model : **Random Forest** and **Naive Bayes** were close to 49.6% , indicating they failed to capture strong patterns in this dataset .

The accuracy of all models using tables and figures :

From the results shown in both the table (1) and the bar chart (4) , we observe that the K-Nearest Neighbors (KNN) model achieved the highest accuracy at 51.31%, while the Random Forest model recorded the lowest performance at 49.66%. Despite the close range of accuracy values across all models, none of them showed a significantly high performance, which may indicate a lack of strong patterns in the dataset or a need for further feature engineering and data preprocessing. This also suggests that improvements such as hyperparameter tuning, feature selection, or trying ensemble techniques might be beneficial in future experiments.

Table (1) : Evaluation of Model Performance Based on Accuracy

Model	Accurcy
Decision Tree	**50.36 %**
Random Forest	**49.66%**
K-Nearest Neighbours (KNN)	**51.31%**
Naive Bayes	**49.95%**
Support Vector Machine (SVM)	**50.31%**
Artificial Neural Network (ANN)	**50.05%**
Linear Regression	**50.07%**

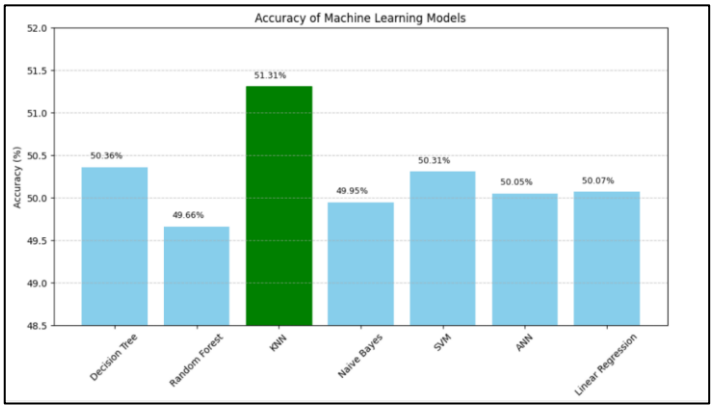

Figure (4) : Bar chart showing the accuracy of all machine learning models.

If your ability to present the result is advanced (using plot libraries such as seaborn etc.) you will get 5 marks bonus

We used advanced data visualization techniques to enhance the understanding of our dataset :

- **Seaborn heatmap** : was used to visualize the correlations between features and detect strong relationships such as depression vs. productivity.

- **Histogram and bar plots** : were applied to show distribution and class balance.

- These visualizations helped in feature analysis and selecting relevant variables.

Therefore, our project includes clear and well-designed visualizations using **Seaborn and Matplotlib.**

Explain in 20 lines , font size 20, Font: Times New Roman :

We chose this dataset because mental health is a major global concern , affecting millions of individuals in both personal and professional settings. The dataset offers a rich collection of features such as sleep hours , stress level , work environment , social support , and depression scores , which are all relevant in assessing mental well-being .

In the modern workplace , stress , burnout , and anxiety are increasing. By applying machine learning to analyse this data , we hoped to discover patterns that indicate which individuals may be at risk . This can help raise awareness and possibly contribute to proactive mental health interventions.

One insight gained was the strong negative correlation between depression and productivity . This means that as depression increases , productivity decreases — a result that aligns with real-world observations.

Among the models tested, KNN (K-Nearest Neighbors) showed the best performance , likely because it can detect localized patterns in structured data. While all models gave similar accuracy , KNN was slightly better.

This project shows how even simple models can provide value when paired with thoughtful data. Understanding the connections between lifestyle and mental health can guide health professionals and employers in supporting well-being.

In conclusion, we selected this dataset for its structure and its importance. The insights gained from the model and data can be a first step toward smarter health analysis.

Link to your code and data , remember the code is in the main folder , the data is in the folder Data, and save the original data + another folder the train test devision(features and targets)?

You can access all the code , data , and results in the following GitHub repository :

- https://github.com/batool1998-hub/ML.project-

in the link there is content :

`main.py` contains the code for all models.

`Data/` contains the original dataset and preprocessed training/testing files.

`results/` folder contains all predicted values from each model.

In the Data folder , create a folder called Result and add test set and the predicted value from all models (to check the accuracy) without the features

A folder named **" Result "** was created inside the **" Data "** directory. It contains the actual test labels and prediction results from all models , including only the **predicted values (Yes / No)** without any feature columns - enabling straightforward accuracy evaluation.