# Machine Learning Assignment

## Child Mind Institute — Problematic Internet Use

Nguyễn Đức Khánh - 22028196
Ngô Lê Hoàng - 22028042
Hoàng Duy Hưng - 22028115

# 1 Overview

## 1.1 Dataset Description

The Healthy Brain Network (HBN) dataset includes clinical data from approximately 5,000 participants aged 5 to 22 years. This dataset is derived from the HBN study, which seeks to identify biological markers that could improve the diagnosis and treatment of mental health and learning disorders from an objective biological perspective.
For this competition, two main elements are used:

- Physical Activity Data: The dataset encompasses time-series information collected from wrist-worn accelerometers, which were continuously worn by participants for up to 30 days. Furthermore, the data set includes fitness assessments and questionnaires that provide additional health metrics and self-reported data, offering valuable insights into the physical well-being of participants.
- Internet Usage Behavior Data: Information related to the internet usage patterns of participants, which is crucial for predicting their Severity Impairment Index (sii).

The dataset is organized into:

- Parquet files containing the accelerometer data.
- CSV files with other tabular data, supplemented by a `data_dictionary.csv` file that provides detailed descriptions of the fields and instruments used in the dataset.

A significant challenge in working with this dataset is the prevalence of missing data, especially for the target sii in the training set. This missing information necessitates the use of various techniques such as imputation and feature engineering to build effective predictive models.
The competition aims to predict the sii based on the available data, employing both supervised learning methods and strategies to handle missing values. The test set includes 3,800 instances with the sii value present for all, allowing for direct evaluation of prediction models.

### 1.2 Evaluation

Submissions are evaluated using the quadratic weighted kappa, which quantifies the agreement between two outcomes. This metric ranges from 0 (random agreement) to 1 (complete agreement), and can fall below 0 if there is less agreement than expected by chance.

## 2 Baseline (Version 0)

### 2.1 Data Processing

First, we drop rows that are missing the `sii` value.
Next, we convert all columns with categorical types into numerical representations (specifically, columns related to the participated season).
Since the test data does not include columns corresponding to PCIAT questions, we drop these columns from the train data as well.
After preprocessing, the train data consists of 2736 rows and 59 columns (excluding the `id` column), and the test data contains 20 rows and 58 columns.

### 2.2 Model

As a baseline, we decided to use the `CatBoostClassifier` without tuning its hyperparameters.

### 2.3 Result

The result of the baseline model is a score of 0.259, which is relatively low because no optimization techniques have been applied yet.

## 3 EDA

### 3.1 Tabular Data

The structured dataset contains demographic information, behavioral survey scores, and related features, including the Parent-Child Internet Addiction Test (PCIAT) scores. These variables form the foundation for understanding patterns and trends in Problematic Internet Use (PIU).

Exploration of PCIAT Scores and sii (target variables)

- The PCIAT dataset consists of 20 questions scored on a scale, capturing the frequency of problematic behaviors related to Internet use. Aggregate metrics like `PCIAT-PCIAT_Total` provide a composite view of a child's Internet addiction risk.
- From the visualizations, we observed the following: higher SII scores are more common in older age groups, though the number of severe cases is very small. The average `PCIAT-PCIAT_Total` score is highest among adolescents. Therefore, issues related to Problematic Internet Use (PIU) often peak during adolescence and then decline in

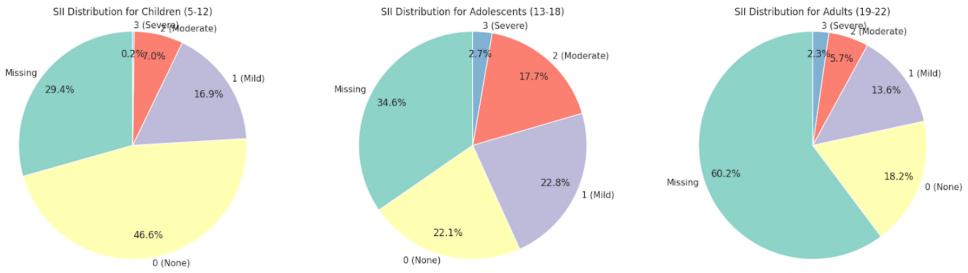


Figure 1: sii distribution within each age group

adulthood. The number of adult participants is very low, which may result in findings that are not fully representative. b

Exploration of Internet Use

- We analyzed the distribution of Internet usage time by age, both for age groups in general and for specific age groups individually.
- We also examined the correlation between `sii` and children's Internet usage time.

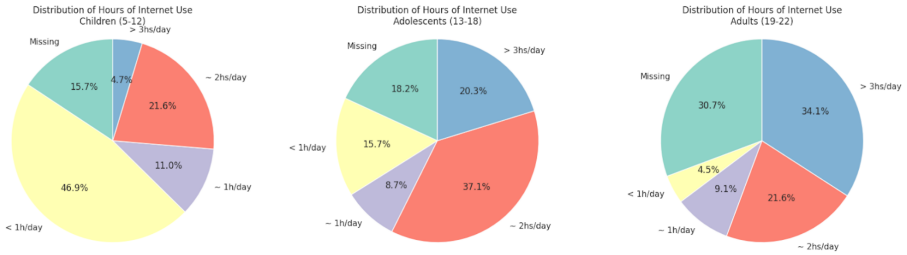


Figure 2: Distribution of Hours of Internet Use within each age group

- From the visualizations, we derived the following insights: individuals who spend more time online daily tend to be older. However, there is overlap in age ranges across different Internet usage time groups. A linear trend exists between `sii` and Internet usage time, with individuals scoring higher on `sii` spending more time online. A non-linear relationship was observed between Internet usage time and PCIAT scores across age groups, with adolescents being the most affected age group. Interestingly, regardless of whether adolescents use the Internet a lot or a little, their PCIAT scores remain high.

Exploration of Physical Measures

- We analyzed the correlation between Physical-Weight by Age, Physical-Height by Age, and Physical-Waist_Circumference vs. Physical-Weight. From this, we observed that both weight and height increase with age, while waist circumference and weight are strongly correlated.

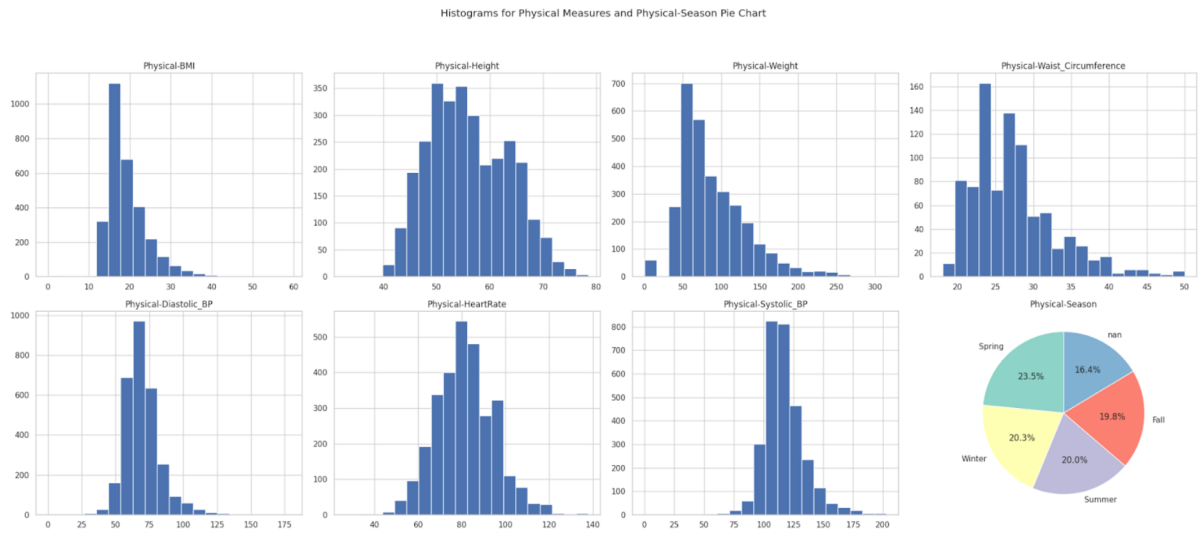


Figure 3: Histograms for Physical Measure and Physical-Season Pie Chart

Exploration of FitnessGram Child

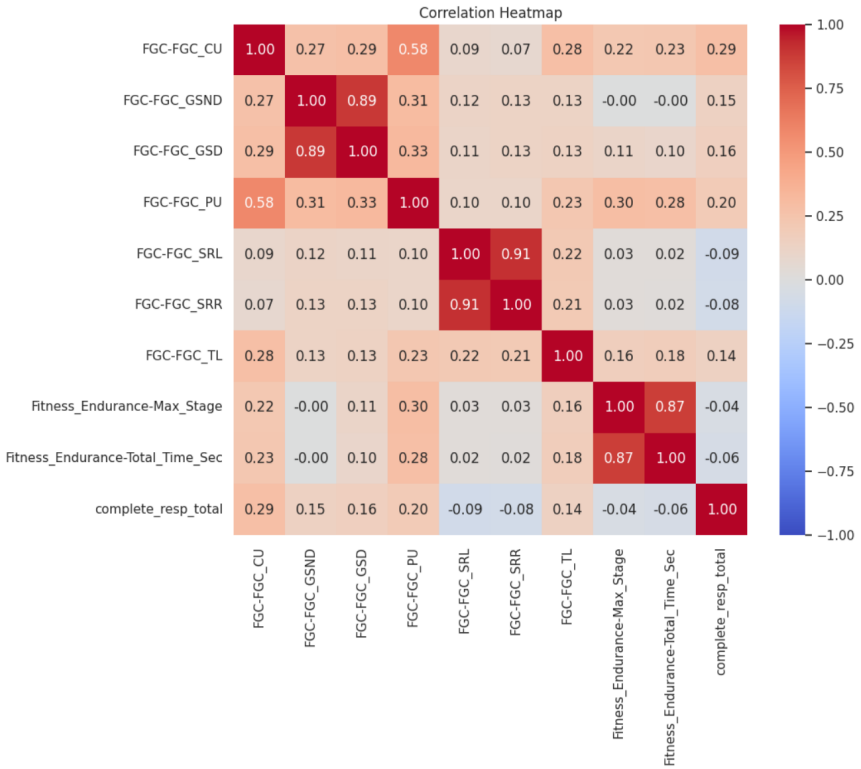


Figure 4: FitnessGram Child Correlation Heatmap

– Through statistics and analysis, combined with the correlation matrix, we derived the following insights: some physical tests show clear correlations, such as hand grip

strength and the sit-and-reach test. Physical performance (sit-ups, push-ups) is positively correlated with higher levels of PIU, which is contrary to expectations. So this correlation is primarily driven by age, and the data is not reliable enough to draw definitive conclusions. Across different age groups, physical fitness correlates well with age, especially in children.

## 3.2 Actigraphy (Time-series Data)

### 3.2.1 Exploration of Time-series data

The actigraphy dataset contains detailed measurements of activity levels over time for each participant. Initial analysis focused on examining the structure and completeness of this data. Missing values and inconsistencies were identified, requiring robust preprocessing steps to ensure accuracy and reliability in subsequent analyses.

### 3.2.2 Preprocessing steps

Feature Engineering

- Aggregated features, such as average activity levels, standard deviations, and activity peaks, were calculated to summarize individual participants' daily and weekly patterns.
- Sleep-related metrics, including average sleep duration, variability, and wake-after-sleep onset (WASO), were extracted using thresholds for detecting sleep and wake periods.

Time alignment

- Time-series data were aligned and normalized to facilitate comparisons across participants, ensuring consistency in daily activity windows.

# 4 Model

## 4.1 One model

After conducting Exploratory Data Analysis (EDA) and running some initial experiments to test how well the models performed, we chose LightGBM (LGBM) as our main model. This model is efficient and works well with tabular data while capturing nonlinear relationships effectively. It became the starting point for shaping our approach to the problem.

We approached the problem in two ways: classification and regression. To find out which method worked best, we tested the performance of the LGBM Classifier and LGBM Regressor using cross-validation. The results were as follows:

- The LGBM Classifier achieved a QWK score of 0.3311, accompanied by the confusion matrix shown in Figure 5.
- The LGBM Regressor achieved a QWK score of 0.3895, along with the confusion matrix shown in Figure 6.

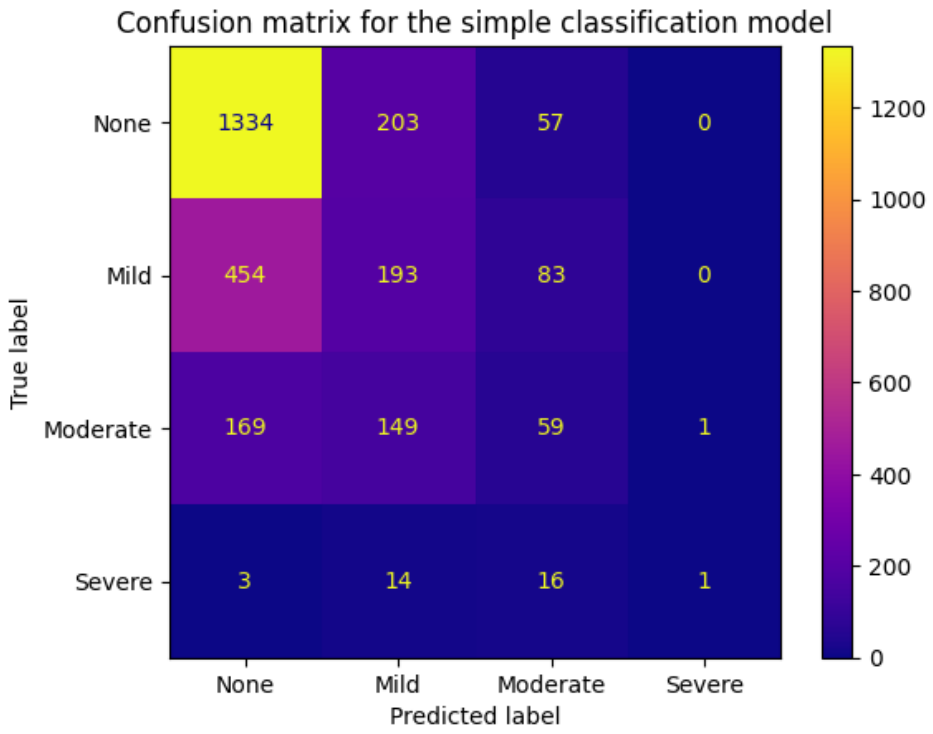


Figure 5: Confusion matrix for classification model

The evaluation indicated that the regression model outperformed the classification model, achieving a higher cross-validation (CV) score. Consequently, we decided to adopt a regression approach, specifically utilizing the LGBM Regressor.

However, when using the LGBM Regressor, the predicted sii values were not integers. Initially, during the evaluation process, we rounded these predictions using thresholds of 0.5, 1.5, and 2.5, but this approach was not optimal. Therefore, our primary objective was to find the optimal thresholds based on model evaluation using the QWK score. Additionally, tuning the model's parameters to optimize the resulting performance was also a key focus for us.

## 4.2 Ensemble (Voting, Boosting)

We chose to utilize the Voting Regressor method to combine predictions from three models: LightGBM (LGBM), XGBoost, and CatBoost. The goal of this approach is to leverage the unique strengths of each model to enhance the accuracy and generalizability of the predictions.

Each model—LightGBM, XGBoost, and CatBoost—possesses distinct advantages. LightGBM excels with tabular data and manages non-linear relationships effectively, XGBoost is robust with large datasets and has strong resistance to overfitting, while CatBoost is particularly adept at handling missing data and managing non-linear features. By integrating

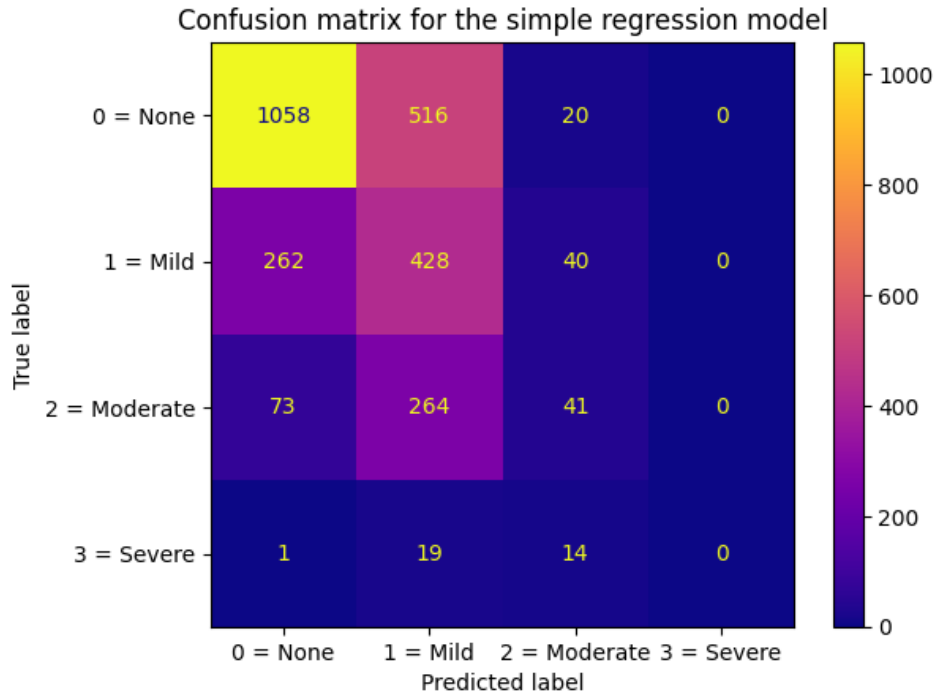


Figure 6: Confusion matrix for regression model

the predictions from all three models, we can harness the complementary strengths of each, thereby improving the overall prediction performance.

Combining the predictions of these models helps mitigate over-reliance on any single model and reduces the risk of overfitting to the training data. This aggregation enhances the model's generalizability on new data by leveraging the diverse capabilities of each model, which improves the robustness and accuracy of the final predictions.

# 5 Notebook Version

## 5.1 Version 1

### 5.1.1 Data processing

Through analysis, we identified that the feature `PCIAT-PCIAT_Total` can predict the `sii` value.
A feature is considered important if it exhibits a high correlation with `PCIAT-PCIAT_Total`.
We retained only the features with an absolute correlation magnitude of $\geq 0.1$.
As a result, the number of features was reduced from 55 to 19.

| | PCIAT-PCIAT_Total |
|---|---|
| Physical-Height | 0.420765 |
| Basic_Demos-Age | 0.409559 |
| PreInt_EduHx-computerinternet_hoursday | 0.374124 |
| Physical-Weight | 0.353048 |
| Physical-Waist_Circumference | 0.327013 |
| FGC-FGC_CU | 0.287494 |
| Physical-BMI | 0.240858 |
| SDS-SDS_Total_T | 0.237718 |
| PAQ_A-Season | 0.219292 |
| FGC-FGC_PU | 0.196006 |
| BIA-BIA_Frame_num | 0.193631 |
| FGC-FGC_GSD | 0.160472 |
| Physical-Systolic_BP | 0.147081 |
| FGC-FGC_GSND | 0.146813 |
| FGC-FGC_TL | 0.136696 |
| PAQ_C-Season | 0.115316 |
| BIA-BIA_FFMI | 0.109694 |
| FGC-FGC_SRR_Zone | -0.109682 |
| FGC-FGC_SRL_Zone | -0.148850 |

Figure 7: Correlation between remain features and `PCIAT-PCIAT_Total`

We also dropped all columns missing more than 70% of their values, as we determined that these columns might not be suitable for further analysis, especially when imputing missing values later. After this step, the number of remaining features decreased to 16.

Among the remaining features, some notably important ones include `Physical-Height`, `Basic_Demos-Age`, `PreInt_EduHx-computerinternet_hoursday`, and `Physical-BMI`.

### 5.1.2 Model

Using XGBoost with the objective of predicting the value of `PCIAT-PCIAT_Total` instead of predicting the value of `sii`.

To mitigate overfitting, apply cross-validation, specifically `Stratified Cross Validation`, to maintain the original distribution of the entire dataset across the folds.

### 5.1.3 Result

Score of this version is **0.431**
We made a small adjustment in version 1 by removing "unstable" rows. This change was necessary because `sii` is derived from the sum of NaN values, which could potentially result in incorrect `sii` calculations. To address this, we recalculated rows with missing values in the PCIAT question columns by replacing NaN with 5 (the maximum score for the questions).

After implementing this change, our results decreased from 0.431 to 0.425, which was quite surprising to us.

## 5.2 Version 2

### 5.2.1 Observation

Information loss occurs because QWK requires the model to convert continuous-valued outputs into discrete classification labels (e.g., treating 1.49 and 1.01 as the same label, despite their relatively significant difference). As a result, the performance heavily **depends on the thresholds**, making it essential to **optimize the threshold values**.

### 5.2.2 Data processing

No longer dropping columns based on correlation as in version 1, instead transitioning to using features extracted from **actigraphy data** (the extracted features are summary statistics (e.g., mean, standard deviation) of the numerical columns in the parquet file).

### 5.2.3 Model

Use a single model: `LightGBM` with optimized hyperparameters

Some advantages when using LightGBM

- Support various objectives and handles missing values directly
- Use advanced techniques like leaf-wise splitting and histogram-based learning to improve model precision
- Capable of handling large datasets and high-dimensional data efficiently

### 5.2.4 Result

Score of this version is **0.435**

Variants of version 2 and their corresponding scores:

- Dropping columns with missing data 70% - Score: **0.403**
- Using `MedianImputer` - Score: **0.366**
- Using `KNNImputer` - Score: **0.405**

## 5.3 Version 3

### 5.3.1 Data processing

Adding features extraction from **actigraphy (time-series data)**

### 5.3.2 Model

The improvement in this version lies in our use of the `VotingRegressor` to derive the final result based on three models: `LightGBM`, `XGBoost` and `CatBoost`.

To further refine our model's performance, we utilize `Hyperopt` for **automated hyperparameter optimization** rather than `Optuna`, as it simplifies the code and facilitates easier expansion with additional models when necessary. `Hyperopt` intelligently explores the parameter space by leveraging past evaluation results, enabling informed decisions regarding which hyperparameters to test next. Specifically, it employs algorithms such as Tree of Parzen Estimators (TPE) to effectively model the performance of hyperparameters and guide the search process.

### 5.3.3 Result

This version achieved a score of 0.422

Voting weights: [`LightGBM`: 6.0, `XGBoost`: 5.0, `CatBoost`: 2.0]

Variation of version 3 and their corresponding scores:

- Add Features + Perform Numerical Imputation + Remain Voting Weights: Score=0.442
- Add Features + Perform Numerical Imputation + Adjust Voting Weights [`LightGBM`: 8.0, `XGBoost`: 3.0, `CatBoost`: 1.0]: Score=0.434
- Add Features + Perform Numerical Imputation + Adjust Voting Weights [`LightGBM`: 9.0, `XGBoost`: 4.0, `CatBoost`: 1.0]: Score=0.447

So the best result is **0.447**