main
final
midterm

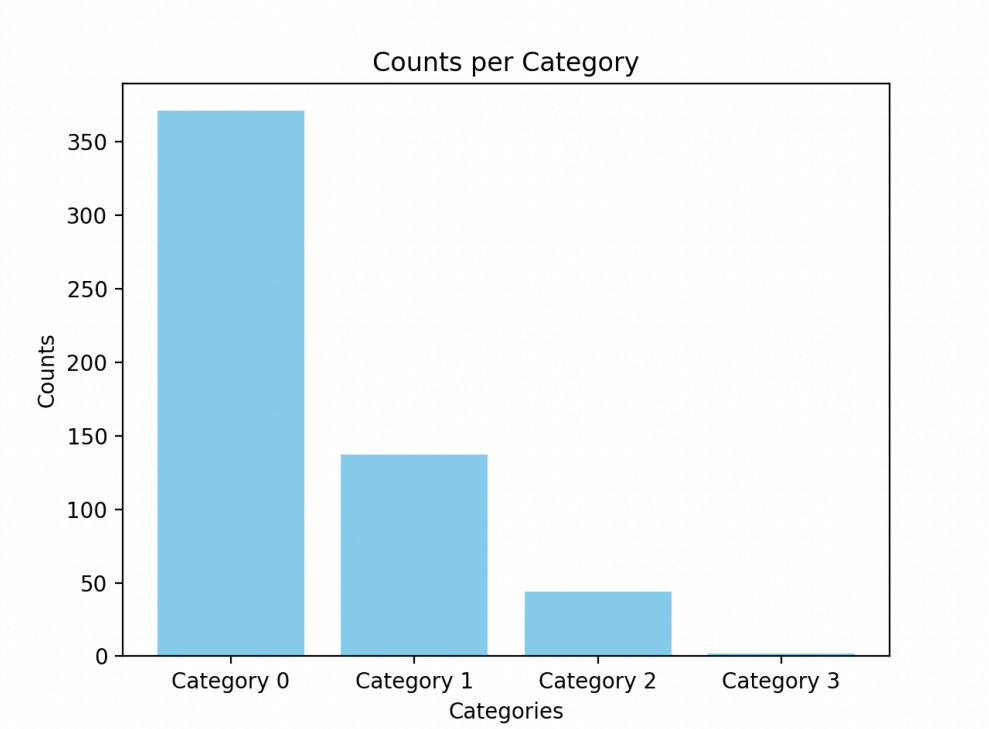
Physical Activity's Influence on Internet Addiction - Group 20 Final Report

GitHub link

Data Preprocessing

Data Exploration

We first examined the data we were working with to get a general sense of what features should be used for training and what preprocessing needed to be done. We noticed that the target variable SII value has a very uneven distribution, which we must keep in mind for the interpretability of our models. Other things of note were that our dataset contained many null values, which necessitates imputation. Finally, there was a significant portion of the data containing the exact values of the answers to the PCIAT questionnaire, which directly contributes to the mathematical calculation of the SII values. Thus, we decided to manually remove these columns so that our models will not be influenced by those direct values and instead will be trained more intelligently on more biometric data.



SII value category distribution in training dataset.

Manual Feature Selection/Hand-Engineered Features

The dataset contains many repetitive columns, such as weight, height, and BMI. By manually deleting redundant features (like height and weight due to having a BMI column), we created a more streamlined and useful dataset.

Filling Missing Data

Since our dataset had a lot of missing values, we needed to fill them in strategically. First, we removed columns with more than 50% missing values. Next, for most columns, such as PCIAT-PCIAT and BIA BIA set data, we filled missing values with the average of the existing data. For some columns, like the CGAS-CGAS score, we used linear regression to predict the missing values based on other related features, including BIA-BIA_BMI, BIA-BIA_BMR, BIA-BIA_DEE, and BIA-BIA_TBW.

SMOTE for Balancing Dataset

Because the original dataset was very skewed towards the 0 range of SII values, we used SMOTE to resample the dataset after value imputation. This allowed for much better predictions for less represented classes.

PCA for Dimensionality Reduction

PCA was used to reduce the dimensionality of our original dataset. By setting an appropriate threshold of 90% accuracy, we retained 13 principal components (PCs) on the training dataset. We then fit the training and testing data to these loadings and generated new training and testing datasets. This significantly reduced the number of features to focus on, thus resulting in more manageable data.

ML Algorithms/Models Implemented

Supervised Learning: Logistic Regression

In our data set, the goal is to categorize each person into ranges of SII values, which is a summary of the total scores each participant received indicating their believed problematic internet usage. In particular, there are 4 score ranges (0 for None, 1 for Mild, 2 for Moderate, and 3 for Severe). Because this is directly translated into a classification problem, we believed logistic regression would be a quick and simple starting point we could use. Moreover, for any other classification model, this simple logistic regression could be a strong benchmark to compare against.

Unsupervised Learning: Fuzzy K-means Clustering

Since the target variable SII value is categorical, the prediction task is well-suited to a k-means clustering method, where the clusters represent SII value score ranges. To categorize SII values into four score ranges (0-3), we applied fuzzy K-means clustering using k = 4. After preprocessing the data using missing value imputation and feature scaling, we chose the five most relevant features based on the preprocessing (BIA-BIA_ICW, BIA-BIA_TBW, BIA-BIA_FFM, BIA-BIA_BMR, and BIA-BIA_LST). Then, we calculated the membership probabilities for each sample and assigned each data point to the cluster it had the highest probability of being in based on those features.

Supervised Learning: Neural Network

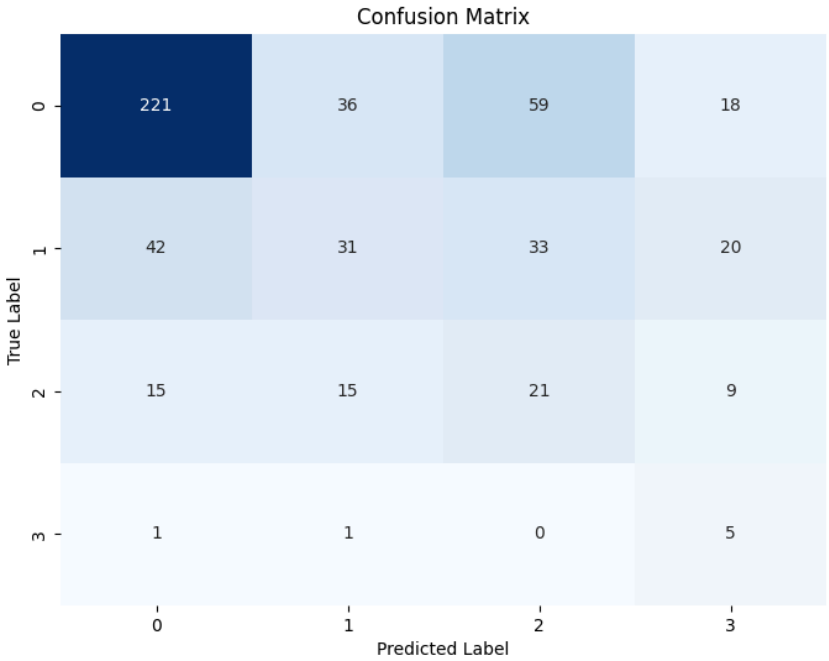
Given that we were trying to categorize into 4 categories as well as the fact that the dataset was pretty complex, we decided a good method would be a neural network. For the actual implementation, our structure was as follows: 31 features (post pca) → Fully connected layer (64 units) → Fully connected layer (32 units) → Dropout layer (20%) → 4 unit output layer.

Note: In our proposal, we originally claimed to create an SVM, but we decided to switch to a neural network out of interest.

Results and Discussion

Logistic Regression | Fuzzy K-means Clustering | Neural Network

Model Results



Logistic regression confusion matrix.

Accuracy: 73%

Precision: macro: 51%, weighted: 70%

Recall: macro: 33%, weighted: 73%

Model Analysis

For logistic regression, while the accuracy was decent for the 4 categories, the recall and precision were not. This was mainly because the model overly preferred choosing category 0. As for why this happened, looking deeper into the data, we see that of the actual labels, 371 of the 554 belonged to range 0. This imbalance of data could explain the issues in precision and recall. As for choosing the features themselves, we chose the features from the principal components that had the highest absolute loading values and thus the most impact on the principal components. Specifically, these features were, SDS-SDS_Total_T, BIA-BIA_FFMI, Physical-Diastolic_BP, and PreInt_EduHx-computerinternet_hoursday. Note, while PAQ_A-PAQ_A_Total had one of the top 5 absolute loading values, we did not use this value as it also had many null values in the dataset.

(change tabs above for other model)

Next Steps

Logistic Regression

Seeing as how logistic regression definitely benefited from the more diverse dataset, we believe that a finer tuned method of sampling would be the next step in improving regression. Specifically, being more careful in choosing which features to be used in which process steps. For example, maybe leaving out gender (a binary value) from things like PCA and value imputation.

Fuzzy K-Means Clustering

Given our lower-than-expected performance for our fuzzy K-means clustering model, we could take several steps to improve them through a couple of ways, notably improving our feature selection, further experimenting with different preprocessing methods like K-nearest neighbors, and also using supervised techniques instead of unsupervised techniques. This time, we did balance the dataset well during training, but still achieved only a 31.4% accuracy. We believe that utilizing supervised learning may be very helpful for Fuzzy K-means clustering in this specific problem.

Neural Network

One thing we might want to experiment with is tuning hyperparameters a bit more. During development, we tuned them a little, but we didn't generate many meaningful improvements beyond the initial parameters we set. The next step might be to tune the preprocessing a bit more. For example, we applied general preprocessing to all features, however, our dataset includes a few binary features like the season, something that might benefit from remaining untouched and not scaled.

Conclusion

In conclusion, we identified our neural network as the best performing overall model, as it had a relatively high accuracy across different classes. We may explore deep learning methods in the future to have more robust predictive power. Finally, we aim to submit to the Kaggle competition, on which the official leaderboard accuracy is still quite low, so who knows what might happen in the future...

Gaant Chart

	Team Member	Proposal Contributions
0	Chen Ye	Value imputation exploration and implementation
1	Matthew Lei	Fuzzy K-means exploration and implementation
2	Eric Ma	Logistic regression exploration and implementation
3	Tiffany Ma	PCA feature reduction exploration and implementation, Streamlit
4	Kevin Song	Fuzzy K-means exploration and implementation