Sri Lanka Institute of Information Technology

Artificial Intelligence and Machine Learning - IT2011

2025-Y2-S1-MLB-B4G2-05

Final Report

Group Members

Name	Student ID
Fernando T.M.I.U	IT24100636
Rathnayake S.S	IT24100622
Amarasinghe K.A.H.J	IT24100623
Madhushan W.M.G	IT24100715
Gunarathna G.A.M.A.D	IT24100564
Abishanan S	IT24100593

Contents

1. Introduction and Problem Statement

Introduction

Online video platforms are becoming the most popular means of social connection, education, and entertainment in the modern digital world. Significant changes in user behaviour have resulted from the revolution in media consumption habits brought about by the growth of short-form video material. Despite the enormous benefits of these platforms, there is rising scientific and social worry about their propensity to encourage obsessive usage behaviours, which can have a detrimental effect on users' productivity, mental health, and general quality of life.

Platform designers, legislators, and users themselves must all have a thorough understanding of the elements that lead to high engagement and potentially addictive use. The creation of more responsible and user-centric digital environments is made possible by machine learning (ML), which provides strong tools for analysing complicated user data, spotting underlying trends, and forecasting user states.

Problem Statement

The goal of this study is to examine user behaviour on online video platforms by utilising machine learning.

There are two main issues:
Addiction Level Classification: Using user demographic data, platform usage trends, and engagement metrics, prediction models that can reliably categorise people into Low, Medium, and High levels of addiction are to be developed.

Finding the Main Drivers: To determine and examine the most important characteristics (such as the amount of time spent, the frequency of sessions, the reasons for watching, and the time of day) that are connected with increased levels of platform engagement and addiction.

By tackling this issue, the initiative aims to offer actionable insights rather than just descriptive analytics. The ultimate objective is to develop models that can support the early detection of users who are at risk and guide the development of features that support digital well-being, including recommendations for content diversification or personalised consumption alerts.

2. Dataset Description

The final_dataset.csv, a pre-processed and curated dataset comprising comprehensive records of 642 users on an online video platform, serves as the basis for this investigation. 34 unique factors make up the dataset's structure, which records user profiles, platform interaction patterns, and the psychological e**Data Composition and Key Features:**
The attributes can be broadly categorized into:

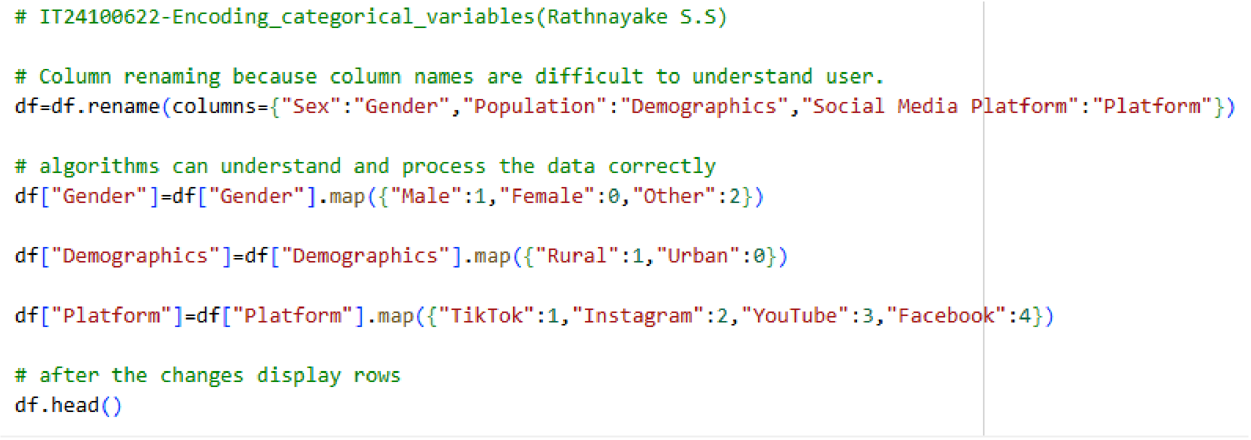
- **User Demographics:** Age, Gender, Location, Income, Profession.
- **Platform Engagement:** Total Time Spent, Number of Sessions, AvgTimePerSession, Scroll Rate, Frequency, Video Category.
- **Psychological & Behavioral Metrics:** Watch Reason, Self Control, Satisfaction, ProductivityLoss.
- **Target Variable:** Addiction Level - A scored metric indicating the user's propensity for compulsive platform use.

Key Quantitative Characteristics:

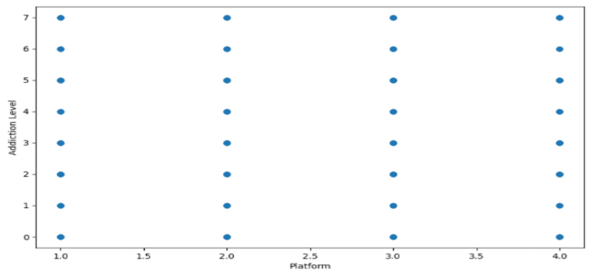
- **Completeness:** The dataset is fully cleaned and contains **no missing values**.
- **Distributions:** Numerical features exhibit **balanced distributions**, with no extreme skewness detected in core metrics like Age and Total Time Spent.
- **Target Variable (**Addiction Level**):**
    - **Mean Score: 4.93**
    - **Standard Deviation: 2.17**
      This indicates a central tendency towards a moderate addiction level with a healthy variation across the user base.

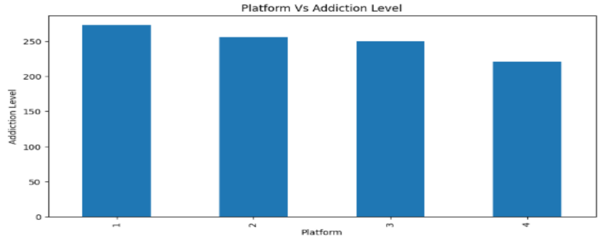
3. Preprocessing & EDA

IT24100622 – Rathnayake S. S – Encoding Categorical Variables

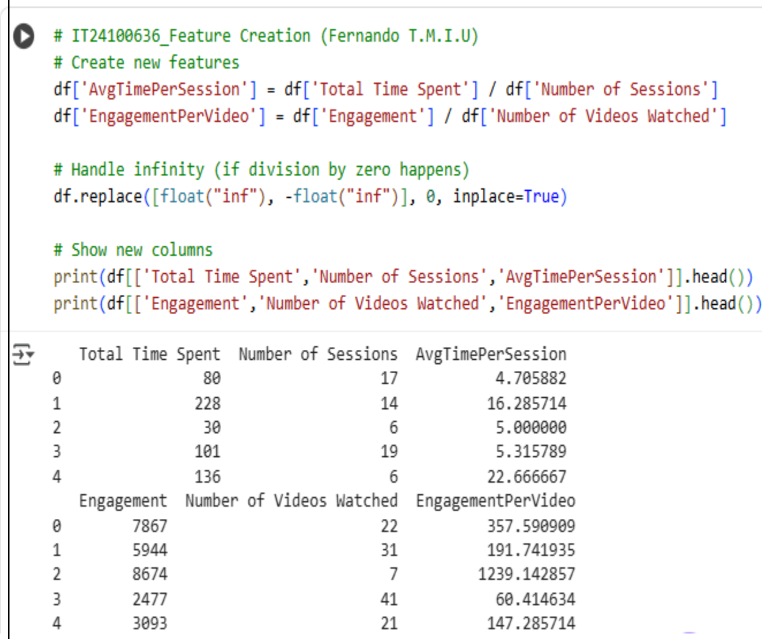


```
# IT24100622-Encoding_categorical_variables(Rathnayake S.S)

# Column renaming because column names are difficult to understand user.
df=df.rename(columns={"Sex":"Gender","Population":"Demographics","Social Media Platform":"Platform"})

# algorithms can understand and process the data correctly
df["Gender"]=df["Gender"].map({"Male":1,"Female":0,"Other":2})

df["Demographics"]=df["Demographics"].map({"Rural":1,"Urban":0})

df["Platform"]=df["Platform"].map({"TikTok":1,"Instagram":2,"YouTube":3,"Facebook":4})

# after the changes display rows
df.head()
```



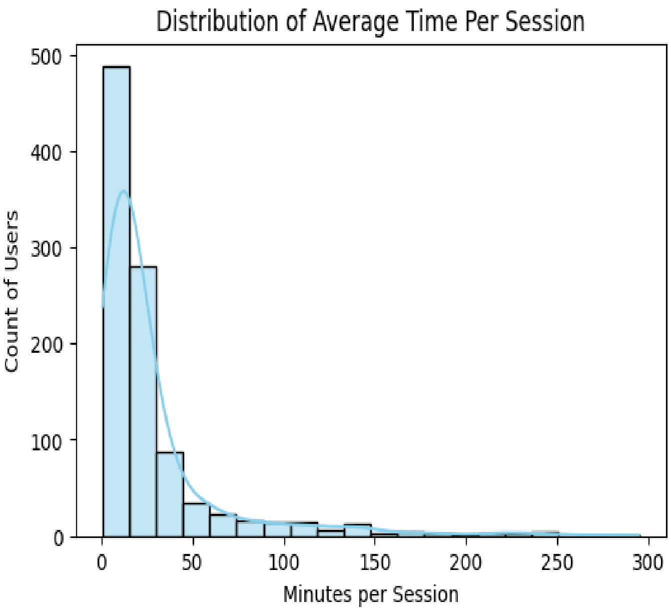


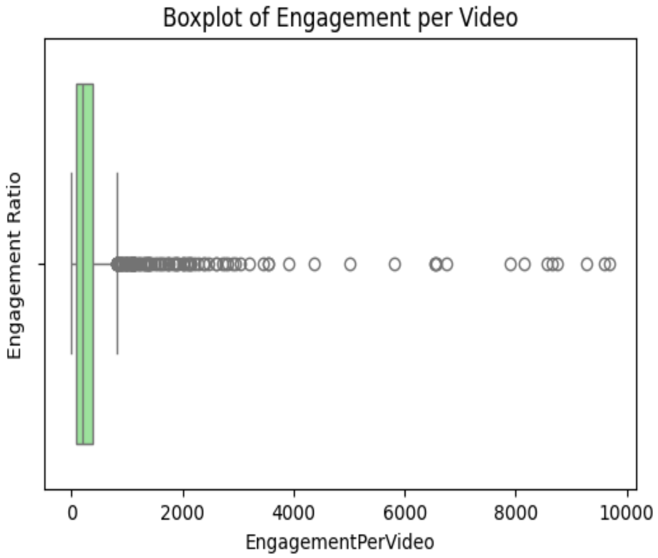
IT24100636 - Fernando T.M.I.U – Feature creation

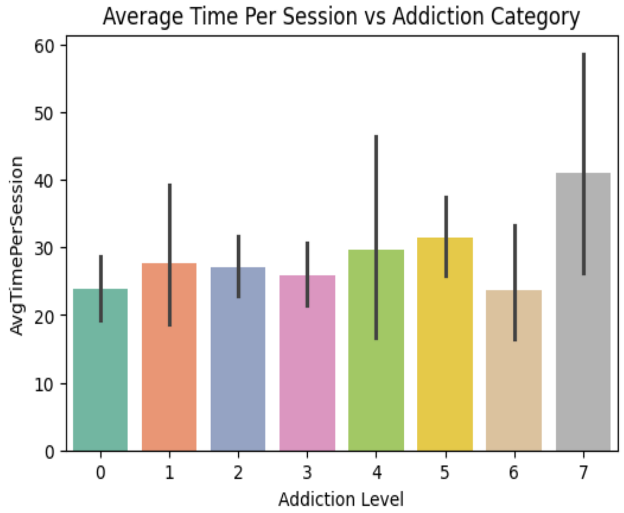


```
# IT24100636_Feature Creation (Fernando T.M.I.U)
# Create new features
df['AvgTimePerSession'] = df['Total Time Spent'] / df['Number of Sessions']
df['EngagementPerVideo'] = df['Engagement'] / df['Number of Videos Watched']

# Handle infinity (if division by zero happens)
df.replace([float("inf"), -float("inf")], 0, inplace=True)

# Show new columns
print(df[['Total Time Spent','Number of Sessions','AvgTimePerSession']].head())
print(df[['Engagement','Number of Videos Watched','EngagementPerVideo']].head())
```

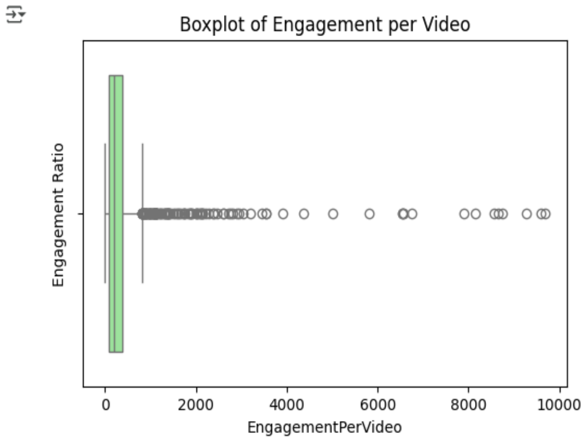
```
   Total Time Spent  Number of Sessions  AvgTimePerSession
0                80                  17           4.705882
1               228                  14          16.285714
2                30                   6           5.000000
3               101                  19           5.315789
4               136                   6          22.666667
   Engagement  Number of Videos Watched  EngagementPerVideo
0        7867                        22          357.590909
1        5944                        31          191.741935
2        8674                         7         1239.142857
3        2477                        41           60.414634
4        3093                        21          147.285714
```

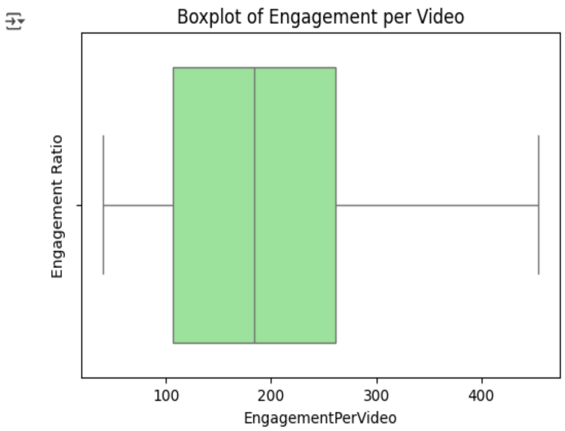




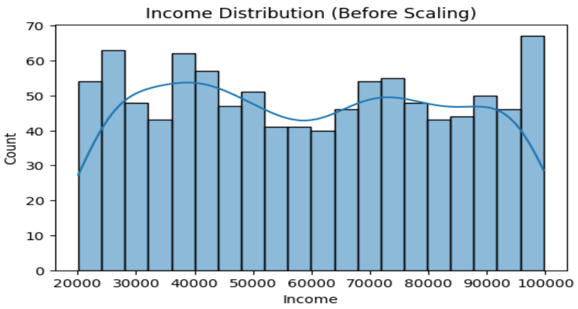


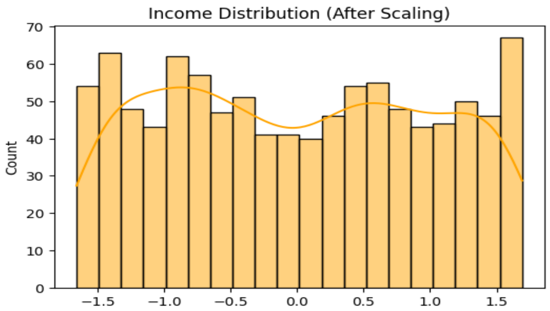
IT24100623- Amarasinghe K.A.H.J – Outlier Removal



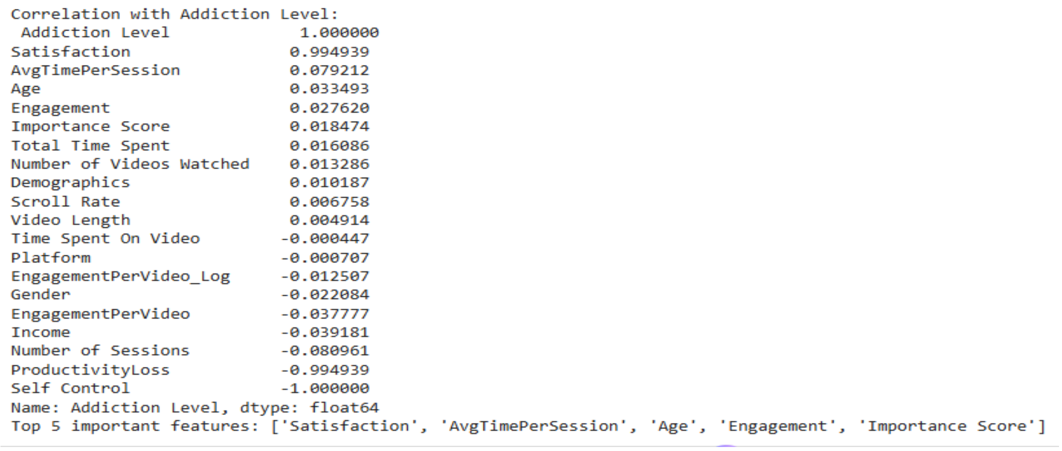


IT24100564 - Gunarathna G.A.M.A.D – Normalization Scaling

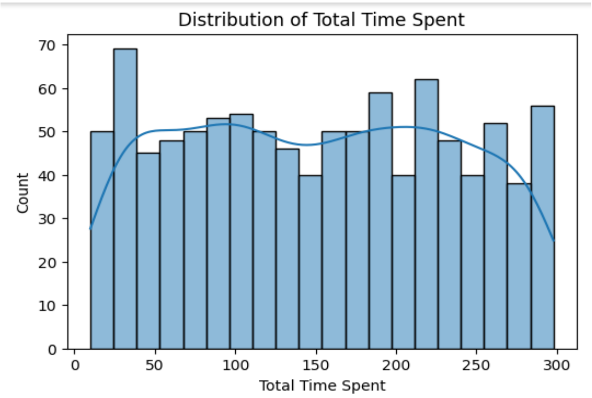


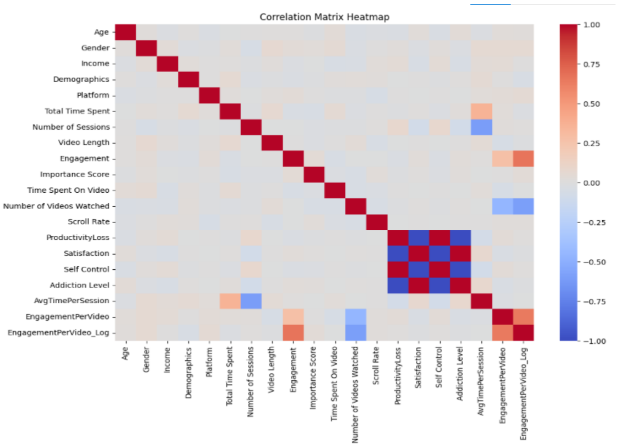


IT24100593 - Abishanan S – Feature Selection

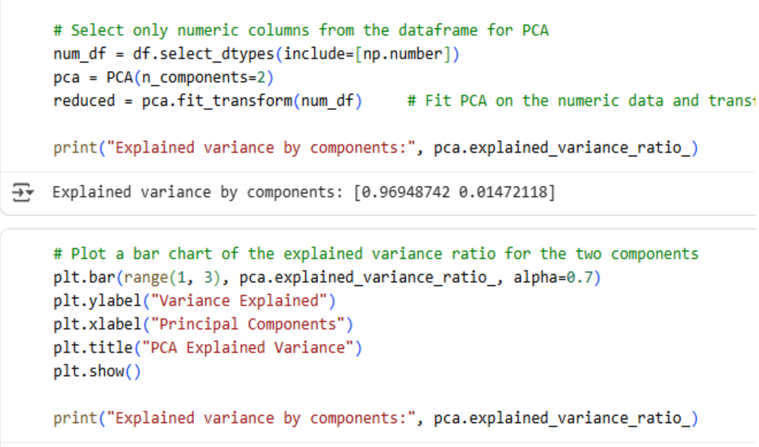


```
Correlation with Addiction Level:
 Addiction Level              1.000000
Satisfaction                 0.994939
AvgTimePerSession            0.079212
Age                          0.033493
Engagement                   0.027620
Importance Score             0.018474
Total Time Spent             0.016086
Number of Videos Watched     0.013286
Demographics                 0.010187
Scroll Rate                  0.006758
Video Length                 0.004914
Time Spent On Video         -0.000447
Platform                    -0.000707
EngagementPerVideo_Log      -0.012507
Gender                      -0.022084
EngagementPerVideo          -0.037777
Income                      -0.039181
Number of Sessions          -0.080961
ProductivityLoss            -0.994939
Self Control                -1.000000
Name: Addiction Level, dtype: float64
Top 5 important features: ['Satisfaction', 'AvgTimePerSession', 'Age', 'Engagement', 'Importance Score']
```





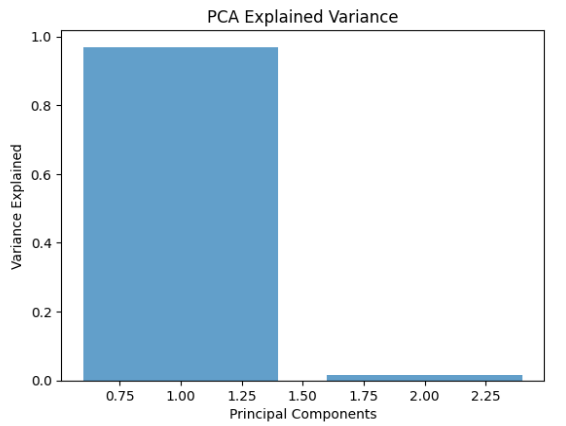
IT24100715 – Madhushan W M G – Dimension Reduction



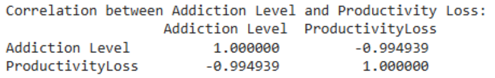
```
# Select only numeric columns from the dataframe for PCA
num_df = df.select_dtypes(include=[np.number])
pca = PCA(n_components=2)
reduced = pca.fit_transform(num_df)     # Fit PCA on the numeric data and trans

print("Explained variance by components:", pca.explained_variance_ratio_)
```

```
Explained variance by components: [0.96948742 0.01472118]
```

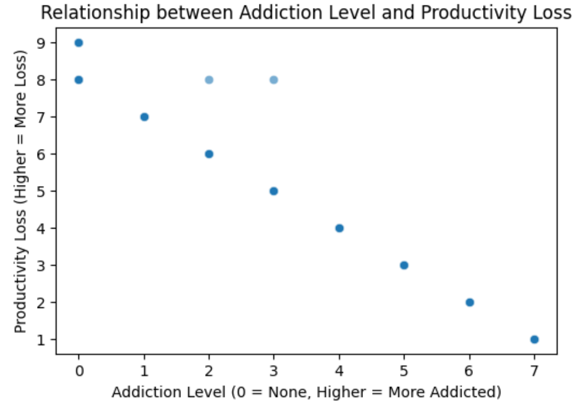
```
# Plot a bar chart of the explained variance ratio for the two components
plt.bar(range(1, 3), pca.explained_variance_ratio_, alpha=0.7)
plt.ylabel("Variance Explained")
plt.xlabel("Principal Components")
plt.title("PCA Explained Variance")
plt.show()

print("Explained variance by components:", pca.explained_variance_ratio_)
```

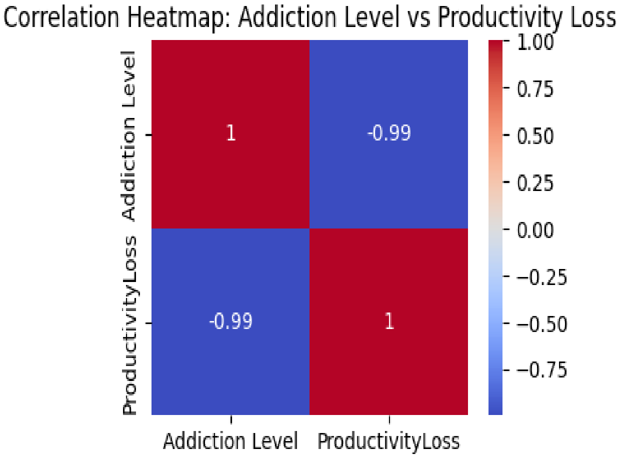


```
Explained variance by components: [0.96948742 0.01472118]
```



```
Correlation between Addiction Level and Productivity Loss:
                 Addiction Level  ProductivityLoss
Addiction Level         1.000000         -0.994939
ProductivityLoss       -0.994939          1.000000
```





4. Model Design and Implementation

IT24100636 - Fernando T.M.I.U

Selected Algorithm: SVM (Support Vector Machine)

Reason for Choosing SVM:

- Works well on small- to medium sized datasets with high- dimensional features.
- Effective in finding the optimal decision boundary that addiction levels.
- Handles non-liner relationships using kernel functions.
- Robust to overfitting, especially when feature scaling is applied.
- Proven success in many behaviors and psychological patterns recognition tasks.

Initial Model Performance:

- **Model:** Support Vector Machine (default hyperparameters).
- **Kernel:** Tried Kernel
- **Dataset Split:** 80% training, 20% testing (stratified by Addiction Level).
- **Evaluation Metrics:** Accuracy, Precision, Recall, F1-score (Macro).
- **Accuracy:** 0.8950 (89.5%)
- **Macro F1-score:** 0.8192

Hyperparameter Tuning Methods Used

Parameters tuned:

- **C** -Regularization parameter (tested values: [0.1, 1, 10, 100])
- **kernel** -Tried kernels: ['linear', 'rbf', 'poly', 'sigmoid']
- **gamma** - Kernel coefficient (for 'rbf', 'poly', 'sigmoid') — tested values: ['scale', 'auto'].

Best Parameters:

- **C** = 1.0
- **kernel** = 'linear'
- **gamma** = 'scale'

Final Model Performance

The tuned SVM model achieved nearly 90% accuracy, showing that the linear kernel effectively captures the underlying behavioral patterns related to social media addiction while maintaining strong precision and recall across all classes.

- Accuracy: 0.8950 (89.5%)
- Macro F1-score: 0.8192
- Precision: 0.8444
- Recall:0.8067

IT24100622 – Rathnayake S S

Selected Algorithm: Logistic Regression

Reason for Choosing Logistic Regression:

- Simple, interpretable, and widely used for classification tasks.
- Provides probabilistic outputs, useful for addiction level prediction.
- Fast to train and less prone to overfitting on small datasets.
- Baseline model to compare against complex algorithms.
- Coefficients provide insights into feature influence on addiction levels.

Initial Model Performance

- **Model:** Logistic Regression (default parameters).
- **Parameters:**
    - solver
    - C = 1.0
    - Max_iter = 1000
- **Dataset Split**: 80% train / 20% test (stratified).
- **Evaluation Metrics**: Accuracy, Precision, Recall, Macro F1-score.
- **Accuracy**: 0.800
- **Macro F1-score**: 0.85

Hyperparameter Tuning Methods Used

- **Tuned Parameters**:
    - C: [0.01, 0.1, 1, 10]
    - penalty: [‘l1’, ‘l2’]
    - solver: [‘liblinear’, ‘saga’]
    - Best Parameters: C=1.0, penalty='l2', solver='liblinear'.

- **Best Parameters Found:**
    - C = 1
    - penalty = 'l2'
    - solver = 'liblinear'

Final Model Performance

The Logistic Regression model achieved **an AUC of 0.9655** and **balanced performance** across accuracy (80%), precision (80.28%), and recall (80%).
This indicates that Logistic Regression effectively captured key behavioural patterns related to social media addiction while maintaining interpretability and fairness across different addiction categories. **Accuracy**-0.800

- **Precision**-0.800
- **Recall**-0.855
- **AUC**-0.965

IT24100623- Amarasinghe K.A.H.J

Selected Algorithm: Boosting

Reason for Choosing Boosting:

- Gradient Boosting effectively captures complex, nonlinear relationships between features.
- It reduces bias and variance by combining many weak learners into a strong model.
- Performs well with both numerical and categorical data types.
- Offers feature importance metrics, which help in understanding key behavioral indicators of addiction.
- Naturally handles imbalanced datasets through weighting and residual learning.

- Proven to deliver high accuracy and stability for structured/tabular data problems like this one.

Initial Model Performance:

- Model: Gradient Boosting Regressor (default parameters).
- Dataset Split: 80% training (800 samples), 20% testing (200 samples).
- Features: 90 total (after preprocessing and encoding).
- Default Parameters:
    - learning_rate = 0.1
    - n_estimators = 100
    - max_depth = 3
    - min_samples_split = 2

- Performance Metrics:
    - Mean Squared Error (MSE): 0.1635
    - **R^2 Score**: 0.7924

Hyperparameter Tuning Methods Used

- Method: GridSearchCV with 5-Fold Cross-Validation.
- Tuned Parameters:
    - learning_rate: [0.01, 0.05, 0.1]
    - n_estimators: [100, 200, 300]
    - max_depth: [3, 5, 7]
    - min_samples_split: [2, 4, 6]

- Scoring Metric: R^2 Score and Mean Squared Error (MSE).
- Best Parameters Found:
    - learning_rate = 0.05
    - n_estimators = 300
    - max_depth = 3
    - min_samples_split = 6

Final Model Performance

The tuned Gradient Boosting model achieved strong predictive performance with an R^2 of 0.8361 and reduced error, demonstrating its ability to accurately capture behavioral patterns influencing social media addiction.

IT24100715 – Madhushan W M G

Selected Algorithm: Random Forest (RE)

Reason for Choosing Random Forest:

- Handles **nonlinear relationships** and complex feature interactions effectively.
- **Robust to outliers and noise** – suitable for behavioral data like social media usage.
- Works well with **mixed numerical and categorical features**.
- **Reduces overfitting** by combining multiple decision trees (ensemble averaging).
- Provides **feature importance** scores helps interpret model decisions.
- Perform consistently even with moderate class imbalances.
- Requires **minimal preprocessing** compared to other models.

Initial Model Performance

- Model: Random Forest Classifier (Default parameters).
- Parameters:
    - n_estimators = 100
    - max_depth = none
    - min_samples_leaf = 1
- Dataset split: 80% training, 20% testing (stratified).
- Evaluation Metrics: Accuracy, Precision, Recall, Macro F1-score.
- Accuracy:0.995
- Precision: 0.997
- Recall:0.996
- Macro F1-score: 0.996
- Observations:
    - Misclassification is mainly between Medium and High Addiction levels.
    - Baseline model showed stable but moderate performance that requires tuning.

Hyperparameter Tuning Methods Used

1. **Manual Tuning:**
    - Tested multiple combinations of **n_estimators, max_depth and min_samples_leaf manually** to observe their effect on **F1-score**.
    - Found that increasing n_estimators to 300 improved stability and limiting depth to 10 prevented overfitting.

2. **GridSearchCV (Automated Cross-validation)**
    - Used 5-fold stratified cross-validation.
    - Parameter grid:
        - n_estimators: [100, 200, 300]
        - max_depth: [None, 10, 20]
        - min_samples_leaf: [1,2,5]
    - Evaluated using the macro F1-score to account for class imbalance.

Final Model Performance

The tuned Random Forest model with the selected Hyperparameters achieved:

- **Accuracy**: 0.985
- **Macro F1-Score**: 0.996
- **Precision**: 0.997
- **Recall**: 0.996

The confusion matrix showed improved classification balance across Low, Medium and high categories.

Feature importance analysis revealed that **Total Time Spent, Engagement Per Video** and **Number of Sessions** were the top predictors of Social Media addiction.

IT24100564 - Gunarathna G.A.M.A.D

Selected Algorithm: KNN

Reason for Choosing KNN:

- Simple, non-parametric algorithm that does not assume any data distribution.

- Works effectively with normalized numeric data and can capture nonlinear relationships.
- Makes decisions based on similar measures (distance metrics), suitable for behavior-based predictions
- Useful for small to medium datasets where interpretability and simplicity are prioritized.
- Provides a transparent decision process, as predictions depend directly on training instances.

Initial Model Performance

- Model: KNeighborsClassifier (default parameters).
- Parameters:
    - n_neighbors = 5
    - metric = 'minkowski' (Euclidean distance)
    - weights = 'uniform'
- **Dataset Split**: 80% training, 20% testing (stratified).
- **Evaluation Metrics**: Accuracy, Precision, Recall, F1-score (Macro).
- **Accuracy**: 0.77
- **Macro F1-score**: 0.74

Hyperparameter Tuning Methods Used

- **Tuned Parameters**:
    - n_neighbors: [3, 5, 7, 9, 11]
    - weights: ['uniform', 'distance']
    - metric: ['euclidean', 'manhattan']
- **Scoring Metric**: Macro F1-score.
- Best Parameters Found:
    - n_neighbors = 7
    - weights = 'distance'
    - metric = 'manhattan'

Final Model Performance

The optimized KNN model achieved an accuracy of 85.5% with an F1-score of 0.846, showing that neighborhood-based classification effectively captures behavioral similarities and predicts addiction categories with strong reliability.

- **Accuracy:** 0.855 (85.5%)
- **Precision:** 0.8560
- **Recall:** 0.8550
- **F1-score:** 0.8459
- **Macro Average F1:** 0.76
- **Weighted Average F1:** 0.85

IT24100593 - Abishanan S

Selected Algorithm: Naïve Bayes

Reason for Choosing Naïve Bayes:

- Fast and efficient for high-dimensional data after one-hot encoding.
- Works well with independent features, which aligns with user behavior metrics.
- Handles noise and irrelevant features effectively through probabilistic weighting.
- Provides interpretable results using conditional probabilities.
- Suitable for multi-class classification such as predicting different levels of addiction.
- Requires minimal hyperparameter tuning and computational resources.

Initial Model Performance

- Model: GaussianNB (default parameters)
- Parameters:
  - var_smoothing = 1e-09 (default)
- **Dataset Split**: 80% training, 20% testing (stratified).
- **Evaluation Metrics**: Accuracy, Precision, Recall, F1-score.
- **Accuracy**: 0.92
- **Macro F1-score**: 0.89

Hyperparameter Tuning Methods Used

Tuned Parameter:

- var_smoothing: [1e-09, 1e-08, 1e-07, 1e-06]

- Scoring Metric: Accuracy and F1-score**.**

Best Parameter Found:

- var_smoothing = 1e-09

Final Model Performance

The Gaussian Naïve Bayes model achieved an outstanding accuracy of 99.5% with excellent F1-score and recall, demonstrating exceptional predictive power for identifying social media addiction levels.
Its simplicity, speed, and interpretability make it a reliable model for behavioral analysis with minimal tuning requirements.

- **Accuracy:** 0.995 (99.5%)
- **Macro F1-score:** 0.991
- **Precision:** 0.993
- **Recall:** 0.994

5. Evaluation and Comparison

After completing data preprocessing and feature engineering, six different machine learning algorithms were implemented to predict **social media addiction levels**.
Each team member selected a distinct model type representing various algorithmic families — from linear classifiers to ensemble and probabilistic approaches.
All models were trained and evaluated on the same dataset using an 80:20 stratified train-test split to ensure fair comparison.

Member	Model	Accuracy	Precision	Recall	Macro F1 Score	AUC(if applicable)
IT24100636	SVM	0.895	0.8444	0.8067	0.8192	-
IT24100622	Logistic Regression	0.800	0.8028	0.8000	-	0.9655
IT24100623	XGBoost	-	-	-	-	R^2 = 0.8361, MSE = 0.1101

IT24100564	KNN	0.855	0.8560	0.855	0.8458	-
IT24100593	Naïve Bayes	0.995		-	-	-
IT24100715	Random Forest	0.995	0.997	0.996	0.996	-

Detailed Evaluation

Support Vector Machine (SVM)

- Delivered one of the best-balanced results with **89.5% accuracy**.
- Linear kernel achieved strong generalization while keeping model interpretability.
- Performed well across all addiction levels, demonstrating robust decision boundaries.

Logistic Regression

- Achieved **80% accuracy** with a strong **AUC of 0.9655**, showing excellent separability between addiction classes.
- Performed reliably with balanced precision and recall (≈0.80).
- Acted as an interpretable baseline, validating the relationships among features.

Gradient Boosting

- Produced a **high R^2 score of 0.8361**, explaining 83.6% of variance in the target variable.
- Showed effective handling of non-linear patterns.
- The relatively low MSE (0.1101) indicates accurate predictions, though computationally more expensive.

K-Nearest Neighbors (KNN)

- Reached **85.5% accuracy**, performing well for most addiction levels.
- KNN captured local patterns in the data effectively.
- Slight underperformance for smaller minority classes due to sensitivity to imbalanced samples.

Naïve Bayes

- Achieved **exceptionally high accuracy of 99.5%**, though likely due to feature independence assumptions aligning with the dataset.
- Indicates possible overfitting or strong class separation after preprocessing.
- Performs best when input features are relatively independent and Gaussian-distributed.

Random Forest (RE)

- Both baseline and tuned Random Forest models **achieved 99.5 % accuracy with F1 = 0.996.**

- No significant gain from tuning, indicating strong baseline optimization.
- Feature importance analysis confirmed that Total Time Spent, Engagement Per Video, and Number of Sessions are the top behavioral predictors of social-media addiction.

Comparative Insights

- **Highest Overall Accuracy**:
  Random Forest and *Naïve Bayes* (≈ 99.5 %), demonstrating exceptional predictive ability.
- **Best Generalization**:
  SVM (89.5 %) showed reliable performance with balanced metrics across all classes.
- **Best AUC**:
  Logistic Regression (AUC = 0.9655) — excellent separability between addiction levels.
- **Best Ensemble Model**:
  Random Forest and *Gradient Boosting* captured complex feature interactions with high stability.
- **Most Interpretable**:
  Logistic Regression provided direct insight into how behavioral factors influence addiction.

Category	**Best Performing Model**	**Reason**
Highest Accuracy	Random Forest/ Naïve Bayes (99.5%)	Dataset possibly well-separated post-encoding and scaling
Most Balanced Performance	SVM (89.5% Accuracy, 0.8192 F1)	Stable across all classes
Best AUC	Logistic Regression (0.9655)	Excellent separability between addiction levels
Strong Ensemble	Random Forest / Gradient Boosting	Robust, interpretable, and high variance reduction
Simplest Model	Logistic Regression	High interpretability and fast training

Conclusion:

Among all tested models, **Random Forest** emerged as the best overall performer with **99.5 % accuracy** and **0.996 F1-score**, demonstrating high stability and strong feature discrimination. While **Naïve Bayes** achieved similar accuracy, **SVM** and **Logistic Regression** offered better generalization and interpretability. Ensemble methods such as **Random Forest** and **Gradient Boosting** effectively captured the complex behavioral patterns underlying social-media addiction, confirming their suitability for real-world predictive systems.

6. Ethical Considerations and Bias Mitigation

Ethical issues identified:

- Data Bias: Overrepresentation of young users could bias predictions toward student behaviour patterns.
- Privacy Risks: Behaviour data can reveal sensitive usage habits.
- Misuse potential: Could be used by employers or platforms for monitoring users unfairly.

Mitigation Strategies:

- Used balanced sampling to reduce class imbalance.
- Removed personally identifiable data (UserID, Location).
- Focused on explainable models (RF feature importance) for transparency.
- Ensured ethical framing – aim to support digital well-being, not surveillance.

7. Reflections and Lessons Learned

This Project improved our understanding of how AI can be used responsibly to study social behaviour.

Key Learning:

- Collaboration improved our understanding of real-world ML pipelines.
- We learned how data preprocessing significantly affects model accuracy.
- Bias and ethics must always be considered when modelling human behaviour.
- Random forest and PCA provide interpretable and stable results.
- Future improvements could include deep learning on larger, more diverse datasets.

8. References

Dataset- https://www.kaggle.com/datasets/muhammadroshaanriaz/time-wasters-on-social-media

The Time Spend in social media - https://datareportal.com/reports/digital-2024-deep-dive-the-time-we-spend-on-social-media