**Naive Bayes Classifier**

**Justification for Choosing Naive Bayes**

For this project, I chose to use the Naive Bayes classifier because it’s a simple and powerful algorithm, especially for binary classification problems like this project’s objective, predicting whether a user is addicted or not. The dataset included several numerical features such daily screen time, app sessions, gaming time, which are fit for the Gaussian Naive Bayes variant since it handles continuous data under the assumption of normal distribution.

**Understanding the Dataset**

The dataset contained variables such as daily screen time, social media usage, gaming time, notifications, and age, among others. After cleaning and preprocessing, I encoded the binary target (addicted vs. not addicted) as 1 and 0. The features remained numerical, aligning with Naïve Bayes' assumption of Gaussian-distributed data.

**Model Performance**

**1.** Accuracy

* **Training Accuracy**: 98%
* **Testing Accuracy**: 98%

This indicates high performance on both training and unseen data, suggesting that the model generalizes well.

**2.** F1-Score

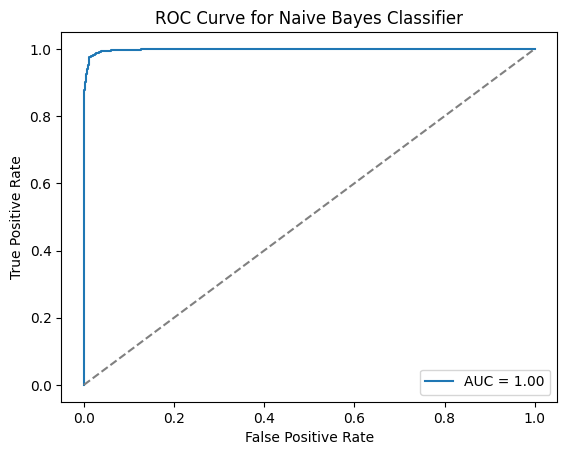
* **Not Addicted (0)**: 0.98
* **Addicted (1)**: 0.98

The F1-score balances precision and recall, confirming the model performs well even if the classes are slightly imbalanced.

**3.** AUC-ROC

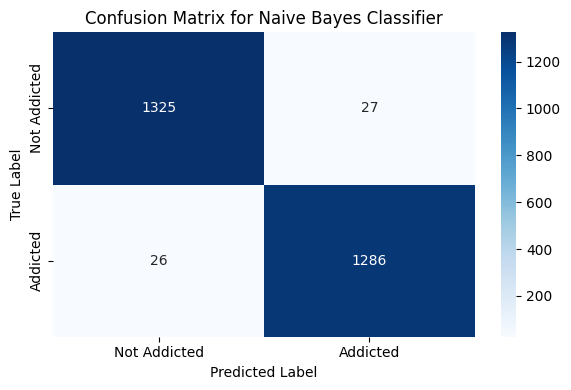
* **AUC Score**: 1.0

The AUC of 1.0 indicates perfect separation of classes on the test set (fig 1.1). This could mean that the data is highly separable using current features or it could be a sign of overly clean data that doesn't reflect real-world noise.



**Fig 1.1** This plot shows how well the model distinguishes between the two classes. A curve close to the diagonal indicates weak performance.

**Confusion Matrix Analysis**

To further evaluate the performance of the Naive Bayes classifier, I generated a confusion matrix (fig 1.2). This matrix provides a detailed breakdown of how well the model predicted each class

**Fig 1.2** Confusion Matrix

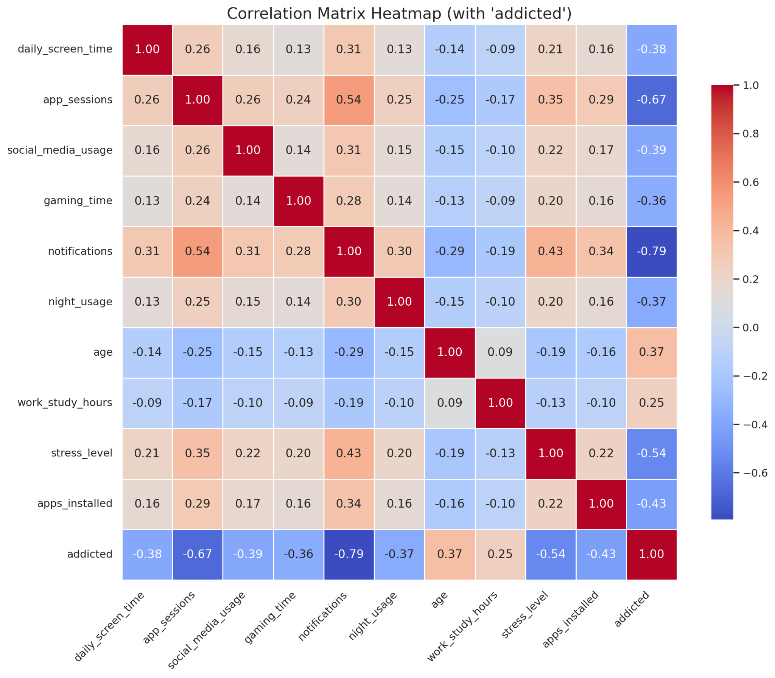
These values confirm the high precision and recall observed in the F1 scores. The low number of false positives and false negatives indicates that the classifier is making very few mistakes, which is consistent with the strong overall accuracy and AUC score.

This balanced performance across both classes shows that the model is not biased toward one class over the other, and that it handles both addicted and not addicted users equally well.

**Correlation Matrix Analysis**

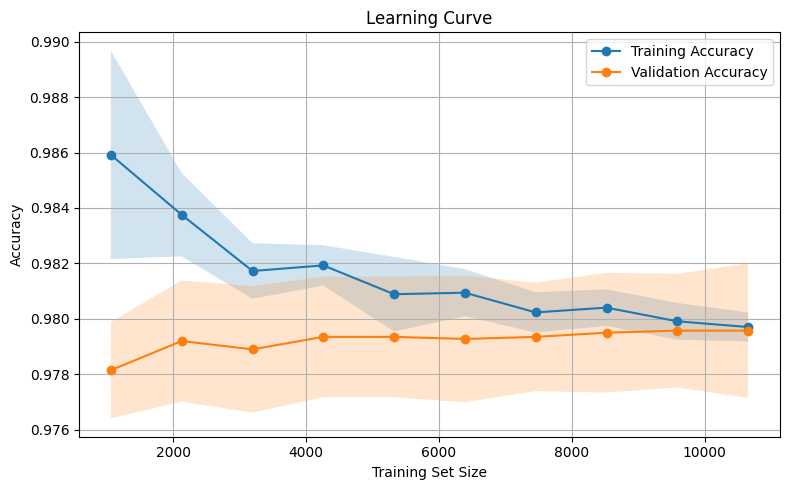
To explore the relationships between the features and the target variable (addicted), I examined the correlation matrix (fig 1.3). The matrix shows how strongly each feature is linearly related to others and to the target.

* **Strongest correlations with addicted:**
  + notifications showed the highest negative correlation with addiction (-0.79), suggesting that as notification frequency increases, the likelihood of being classified as addicted also increases.
  + app\_sessions (-0.67), stress\_level (-0.54), and apps\_installed (-0.43) also had moderately strong negative correlations with addiction.
* **Moderate correlations:**
  + Interestingly, **age** had a positive correlation (0.37), indicating that younger users were slightly more likely to be classified as addicted.
* **Low correlations between features:**
  + Most feature-to-feature correlations remained below 0.5, with only a few exceptions (e.g., notifications and app\_sessions at 0.54). This suggests that multicollinearity is limited, which is beneficial for models like Naive Bayes that assume feature independence.

****Overall, this correlation structure supports the assumptions of the Gaussian Naive Bayes classifier to a reasonable degree, particularly the low to moderate correlations between input features.

**Fig 1.3** Correlation Matrix (Heat Map)

**Is the Model Overfitting or Underfitting?**

The learning curve (fig 1.3) shows that the training accuracy starts off high at around 98.6% and gradually decreases as the training set size increases. At the same time, the validation accuracy remains slightly lower, between 97.8% and 98.0%, and stays quite stable. The small and consistent gap between the two curves indicates that the model is not overfitting, as both training and validation perform similarly across different dataset sizes. Moreover, because both curves are stable and maintain high accuracy, the model doesn’t appear to be underfitting either. This suggests that Naive Bayes is achieving a good balance between bias and variance for this problem, fitting the data well without too much complexity.

**Fig 1.3** Learning Curve

**Why the Accuracy is High**

The high accuracy and strong performance metrics are likely because the classes are well separated in the feature space, which is supported by the high AUC and F1 scores. Also, since all the features are numerical and seem to follow a Gaussian distribution (refer to data visualization), they match the assumptions of the Gaussian Naive Bayes model really well. Another factor is that most of the feature correlations are moderate (below 0.3), which means the independence assumption is mostly holding up and not hurting the model.

**Conclusion**

The Naive Bayes classifier performed really well on this dataset, and there were no clear signs of overfitting or underfitting. The strong results seem to be helped by the fact that, even though there are some small dependencies between features, the data mostly fits the independence assumption that Naive Bayes relies on. Also, the features are likely close to being normally distributed, which works well with the Gaussian version of Naive Bayes. Finally, the features themselves seem to be informative enough to separate the addicted and not addicted classes clearly, which makes the model’s job easier.