Predictive analysis of depression in students using Machine Learning Algorithms

---

Master's degree in engineering and computer science

Machine Learning & Deep Learning – A. Y. 2024/2025

Filippo Ridolfi VR502263

Summary

Problem Definition and Activities **3**
State of Art 3
Main Objective 3
Methodology 4
Tasks definition 4
Dataset and description **5**
Features distribution 5
Possible correlations between features 6
Features extraction from data **7**
Dimensionality reduction 8
Models **9**
Evaluation metrics 9
K-fold cross-validation 9
K-Nearest Neighbors 10
Random Forest 11
SVM (Support Vector Machine) 12
Performance Summary 13
Results discussion and conclusions **14**
Reference **15**

Problem Definition and Activities

Depression among students is an increasingly prevalent phenomenon, negatively impacting their psychological well-being, academic performance, and social relationships. However, the signs of depression are often overlooked due to a lack of strategies for early identification and prevention. Automating the prediction of depression may help develop more effective support strategies. Traditionally, the diagnosis of depression occurs through psychometric questionnaires and clinical assessments, which have significant limitations, such as the subjectivity of responses and the risk of underreporting by students. In recent years, artificial intelligence and Machine Learning (ML) have shown great potential in mental health, allowing for the automatic analysis of complex data and early identification of signs of depression through predictive models based on structured and unstructured data.

State of Art

The application of Machine Learning in mental health represents an increasingly active and promising research area. The potential of these techniques in analyzing large amounts of data, identifying patterns, and predicting clinical outcomes offers new perspectives for the diagnosis, treatment, and prevention of mental disorders. Not only can diagnoses be made and prognoses predicted, but treatments can also be personalized to identify the most effective therapies for individual patients based on their characteristics. For instance, Machine Learning and Deep Learning techniques are widely used to classify electroencephalographic (EEG) data to analyze brain signals and identify useful patterns in applications such as brain-computer interfaces (BCI) (controlling devices with thought) and monitoring mental states (sleep, stress, attention). Other examples include:

- SimSensei: a therapeutic AI chatbot that provides psychological support for individuals with anxiety and depression.
- DeepMind Health: developing advanced Machine Learning techniques to create predictive models for conditions such as acute kidney injury up to 48 hours in advance. Another interesting creation is AlphaFold, an AI model capable of predicting protein structures and accelerating the development of new drugs.

Despite the excellent applications, several challenges remain, such as data quality, privacy, interpretability, and ethics. It is also essential to ensure patient autonomy and correctly manage their sensitive data.

Main Objective

The study aims to achieve comprehensive analysis using a dataset that contains both academic information and psychological and behavioral variables. Additionally, various Machine Learning models will be compared to evaluate not only their performance in predicting depression but also the best compromise between accuracy, interpretability, and scalability.

The main objective of the project is to develop predictive models that can identify potential students at risk of depression. It will be important to ensure as low a few false negatives as possible. To avoid students with possible future depression may not be counted by the model.

This type of analysis can be useful to support educational institutions in designing targeted interventions and prevention strategies against depression among students.

Methodology

The three Machine Learning models that will be used in the study will be: K-Nearest Neighbor (KNN), Random Forest, and Support Vector Machine.

K-Nearest Neighbor is a classification algorithm that assigns to a new piece of data the class of its K most similar neighbors in the training dataset. The distance between points is calculated using metrics such as Euclidean distance or Mahalanobis distance. It's a simple algorithm able to manage non-linear relations.

Random Forest is an ensemble algorithm that combines many decision trees to improve performance and robustness. Each decision tree in the forest is built on a random subset of data and features. Able to capture complex interactions between variables.

Support Vector Machine is a very powerful supervised learning algorithm used for classification problems. The fundamental idea is to find the hyperplane that best separates the data into two classes. Very efficiently in high dimensional space.

Tasks definition

The following procedures were followed for the analysis of the dataset:

1. **Data pre-processing**: understanding the dataset, analyzing the structure of the data, the distribution of features, and how the records are stored in the table. Cleanup of missing values and categorical data conversions via one-hot encoding or label encoding.

2. **Dimensionality reduction**: using techniques such as PCA (principal component analysis) to build a new set of features by creating new ones starting from existing ones. To represent samples using a smaller number of features, trying to preserve the structure of the original data as much as possible.

3. **Classification**: we proceed to predict the target variable Depression using three different algorithms: Random Forest to capture the complex interactions between the variables; KNN (K-nearest neighbor), excellent for non-linear relationships between the variables, and SVM (support vector machine). The dataset will be divided into training, validation, and testing sets. The models on the training data will be validated via K-fold cross-validation

4. **Performance evaluation**: evaluation of the above-mentioned models with metrics such as Accuracy, Precision, and Recall also changing the key parameters (k for KNN and n for the number of Random Forest components). Greater importance will be given to Recall over Precision to obtain the least number of false negatives.

Dataset and description

The “Student Depression Analysis” dataset is a two-dimensional dataset used to analyze the mental well-being of students, with the aim of identifying factors that influence depression. It includes information on variables such as age, gender, sleep habits, academic satisfaction, and study pressure. The target variable is the students' depression status (yes/no).

It is composed of 27,898 observations and 18 columns representing the main characteristics of the individuals analyzed.

Main features

1. Id (integer): unique identifier for each person.
2. Gender (object): sex of the person.
3. Age (float): age of the person.
4. City (object): city of residence.
5. Profession (object): profession.
6. Academic Pressure (float): perceived level of academic pressure (from 0 to 5).
7. Work Pressure (float): perceived level of work pressure (from 0 to 5).
8. CGPA (float): academic average (0 to 10).
9. Study Satisfaction (float): satisfaction with the study (from 0 to 5).
10. Job Satisfaction (float): job satisfaction (from 0 to 4).
11. Sleep Duration: average sleep duration.
12. Dietary Habits (object): Eating habits.
13. Degree (object): type of degree or qualification.
14. Have you ever had suicidal thoughts?: if you have ever had suicidal thoughts (yes/no).
15. Work/Study hours (float): weekly hours dedicated to work or study.
16. Financial Stress (float): level of perceived financial stress (from 1 to 5).
17. Family History of Mental Illness (object): the presence of a family history of mental illnesses (yes/no).
18. Depression (int): indicates the individual suffers from depression.

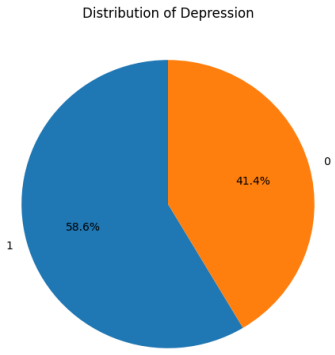
Features distribution

Through the analysis of the distribution of features, we can understand, clean, and prepare the data before using it in predictive models. This phase allows you to identify the variability of the samples (whether the data is balanced or unbalanced) and choose the most appropriate transformations (one-hot encoding or label encoder).

The analysis was also carried out through descriptive statistics which help better understand the distribution of the data: mean for the average value of the distribution and standard deviation which indicates how scattered or concentrated the data are around the mean.

Some interesting points found during the analysis:

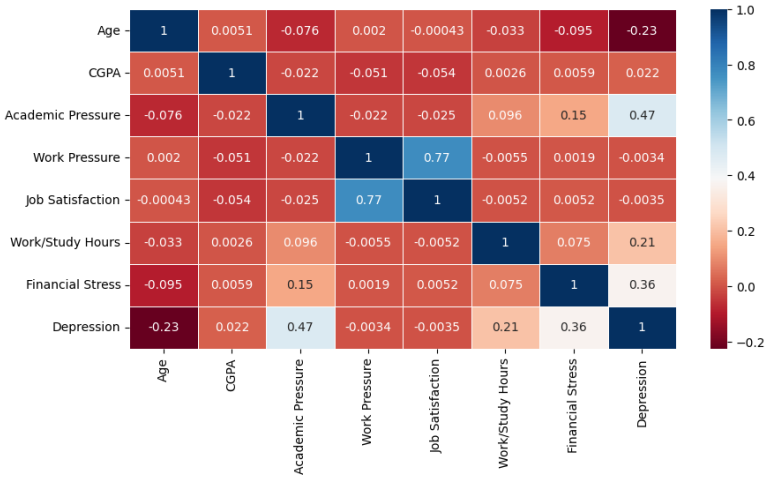
- Job Satisfaction, Work Pressure, and Profession have low or zero values for almost all participants, suggesting that many students are jobless. The tables are therefore not useful for practical analysis purposes.
- The target variable, Depression, in our dataset is imbalanced, but not extremely so. A dataset is considered highly unbalanced when one class has much less data than the other.



Possible correlations between features

By analyzing the heatmap of the correlation matrix the following observations were made:

- Academic pressure and financial stress appear to be key factors associated with depression.
- Age appears to have a negative correlation with depression, suggesting that younger students may be more vulnerable.
- Job satisfaction is strongly linked to work pressure.



Features extraction from data

After the distribution analysis, new features were selected and transformed starting from the raw data. These data transformations will subsequently allow the most important features to be used. The aim is that they directly influence the performance of the various ML models used.

Two different types of processes were followed in the extraction phase:

- **Manual process**: observing the distribution data, it was noted that the Job Satisfaction, Work Pressure, and Profession features do not provide any relevant information for the purpose, given their highly unbalanced distribution. The columns were removed manually using the drop method, thus bringing the features from the initial 18 to 15.

Profession		Job Satisfaction		Work Pressure	
Student	27867	0.0	27890	0.0	27895
Architect	8	2.0	3	5.0	2
Teacher	6	4.0	2	2.0	1
Digital Marketer	3	1.0	2		
Content Writer	2	3.0	1		
Chef	2				
Doctor	2				
Pharmacist	2				
Civil Engineer	1				
UX/UI Designer	1				
Educational Consultant	1				
Manager	1				
Lawyer	1				
Entrepreneur	1				

- **Automatic process**: I transformed the categorical variables into a numeric format to be able to use them in the models through both One-hot encoding and Label encoding.

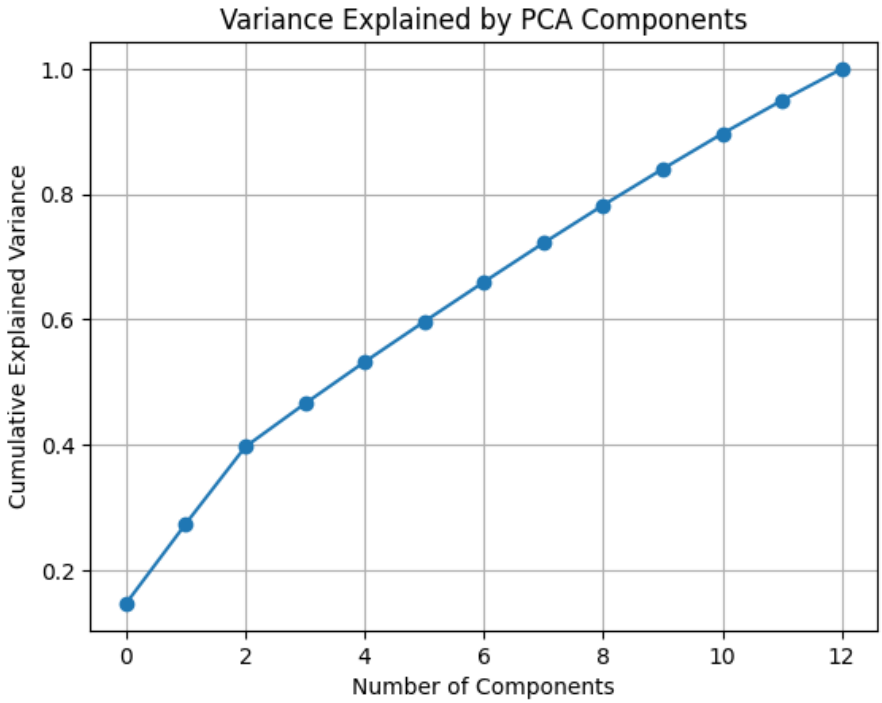
    - One-hot encoding: used to create a binary column for each category present in a variable. The variables chosen were: Gender, have you ever had suicidal thoughts? and Family History of Mental Illness.
    - Label encoder: used to assign an internal number to each category. This method was useful to avoid the excessive creation of columns within the dataset. The variables chosen were Degree, Sleep Duration, and Dietary Habits. City.

Although categorical variables such as City and Degree do not have a natural order, we chose to use Label Encoding for them. Tests have been carried out both regarding their conversion via Label Encoding and One-Hot encoding, observing that the changes are not significant. Using One-Hot encoding the dimensionality of the dataset increased greatly and this led to the slowing down of the models without leading to any significant improvement. Using Label Encoding for City and Degree it was possible to contain the number of features (17) while maintaining the same level of detail.

The dataset was divided into 60% training data, 20% validation data, and 20% testing data. To then be subsequently standardized using the StandardScaler() object. In this way, the subsequent PCA will no longer give weight to variables with larger values but will work with features of the same scale.

Dimensionality reduction

The next step, after finishing the analysis of the dataset and the transformations of the categorical variables into numerical ones, was the reduction of the dimensionality of the features. The idea is to improve the efficiency and performance of the model. The technique chosen was PCA (Principal Computer Analysis), which transforms the data into a new base of smaller dimensions. This technique is adapted to the training data and then transforms the training, testing, and validation data.



The graph shows how the accuracy of the model varies with the number of principal components used in PCA. The variance was set to 95% to reduce the dimensionality of the dataset but still maintain a significant part of the variance of the dataset. As can be observed, around twelve components the accuracy stabilizes. Indicating that adding more principal components does not contribute significantly to the improvement of the model's performance. So PCA has successfully managed to reduce the dimensionality of the data and can improve the accuracy of the models. So, our X will have dimensions (22318, 12).

Models

The problem with which we want to perform the analysis is a classification problem. The goal is to assign one or more labels to observations based on their distinctive features. The discrete variable that we want to predict is 'Depression'. We use the other features of the dataset to make predictions. The output will be discrete, that is, the model must decide which class each example belongs to (whether depressed or not).

Evaluation metrics

The metrics to evaluate whether a model performs effectively and to choose the changes to the pre-established parameters are:

- Accuracy: represents the percentage of correct predictions.
- Precision: represents the ability of the model to not classify as positive an element that is negative.
- Recall: represents the ability of the model to find all the positive examples (useful to avoid false negatives).
- Confusion matrix: represents a table that shows the number of correct and/or incorrect predictions for each class.

To improve the performance of the model, these metrics have been crucial. They have allowed us to update the choice of optimally predefined parameters.

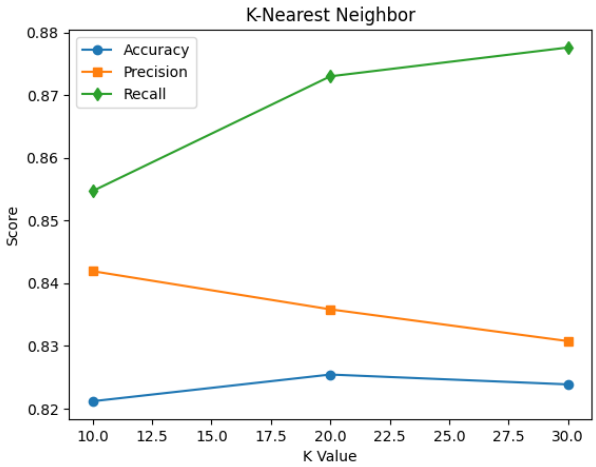
K-fold cross-validation

I chose to use k-fold cross-validation on the training data for each algorithm used in the report. I split the dataset into K subsets (five, to be precise), using each fold in turn as a test set while the others are used to train the model. This process, repeated for each fold, provides an estimate of the model's performance that is less susceptible to data variability and offers a more robust and reliable evaluation compared to a simple split into training, test, and validation sets. It also helps reduce the risk of overfitting because the model is trained on various subsets of the data. This is especially useful for KNN, which can be sensitive to the choice of training data, and for Random Forest and SVM, which tend to overfit the training data. It also aids in assessing accuracy and performance, improving the model's generalization.

K-Nearest Neighbors

The first classification model used was KNN using a dataset selected with the PCA technique. KNN is a data-driven and non-parametric algorithm (Instance-based). Therefore, it relies on the data itself to make predictions.

The main reasons why I chose this model are the simple implementation and interpretation. But above all, the fact that it does not require assumptions on the type of data distribution makes it suitable for any type of distribution. The most critical part of the model is the choice of parameter K (number of neighbors). I tested various values of K using K-fold cross-validation, obtaining these results:



Observing that with K set to twenty neighbors, the model reaches its peak of accuracy with excellent estimates for recall and Precision.

Testing the model on twenty neighbors the results in the validation phase were:

- The correctly classified instances are on the total 83.42%.
- 85.00% of the examples classified as positive by the model were positive.
- The positive instances were correctly identified for 87.68%

Testing the model on twenty neighbors the results in the testing phase were:

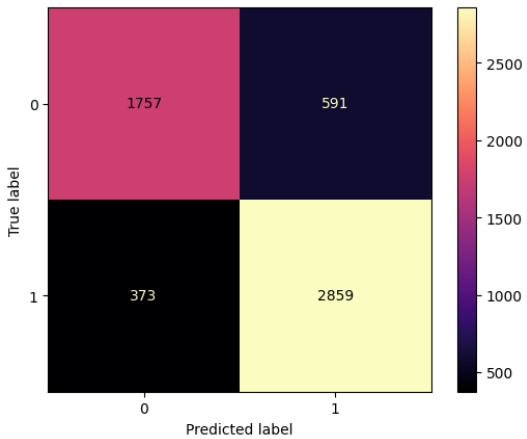
- The correctly classified instances are on the total 82.72%.
- 82.86% of the examples classified as positive by the model were positive.
- The positive instances were correctly identified for 88.46%

The confusion matrix, of the testing set, crosses the real values with the values predicted by the model.

The four cells represent:

- 1757 are the true negatives
- 591 are the false positives
- 373 are the false negatives
- 2859 are the true positives

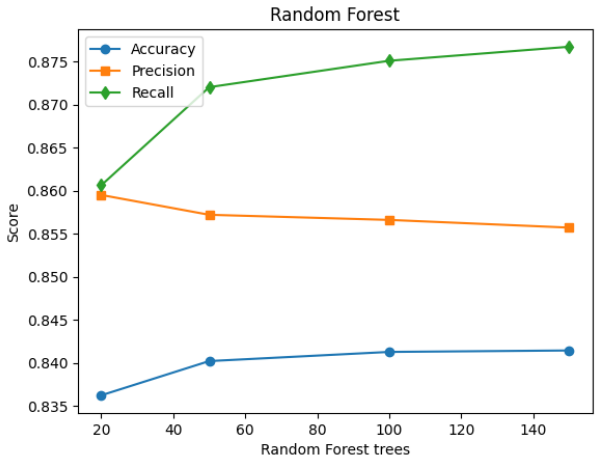
The model is better at correctly identifying class 1 (high recall) than class 0. Great for verifying that students at risk of depression will be correctly flagged.



Random Forest

The next algorithm used was Random Forest. The algorithm that builds a forest of decision trees. Each tree is trained on a random sample of the dataset. The advantage of using this technique compared to KNN is the resistance to outliers thanks to the average of the results between the various trees. It does not require data normalization. However, it is slower than the previous KNN. The goal is to obtain further predictions, robust, and generalizable without having to configure too many hyperparameters.

To verify the effectiveness of Random Forest I used multiple numbers of trees that make up the forest (n_estimators parameter). The values tested, using K-fold cross-validation are: 20, 50, 100, 150. Starting from low values with the risk of a less performing generalization up to high values improves the stability of the model but increases the training time.



Observing that with the number of trees set to fifty, the model reaches its peak of accuracy with excellent estimates also on Recall and Precision.

Testing the model on fifty trees the results in the validation phase were:

- The correctly classified instances are on the total 84.36%.
- 86.42% of the examples classified as positive by the model were positive.
- The positive instances were correctly identified for 87.53%

Testing the model on fifty trees the results in the testing phase were:

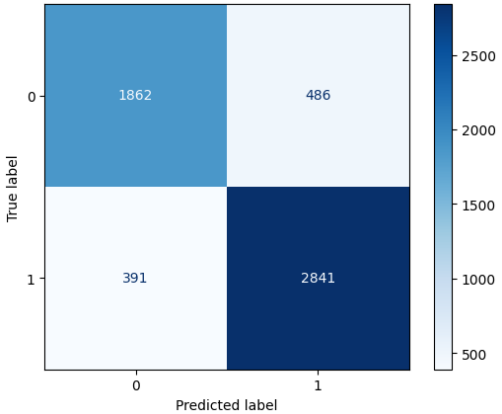
- The correctly classified instances are on the total 84.28%.
- 85.39% of the examples classified as positive by the model were positive.
- The positive instances were correctly identified for 87.90%

The confusion matrix, of the testing set, crosses the real values with the values predicted by the model.

The four cells represent:

- 1862 are the true negatives
- 486 are the false positives
- 391 are the false negatives
- 2841 are the true positives

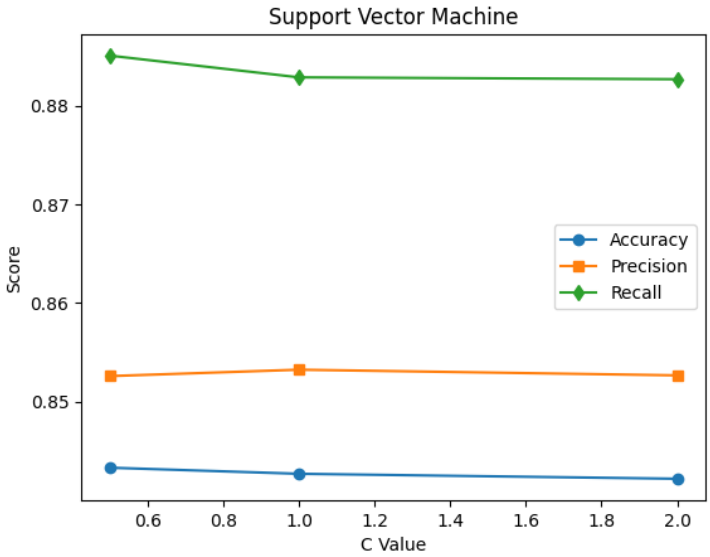
The model is better at correctly identifying class 1 (high recall) than class 0. Great for verifying that students at risk of depression will be correctly flagged.



SVM (Support Vector Machine)

The SVM model is based on the search for a separation frontier (hyperplane) that maximizes the margin between classes. The separation occurs using support points. The reasons why this model was chosen are the good performance on linear and nonlinear problems.

The kernel chosen for the SVM is the Radial Basis Function (RBF). This is a function that maps the data into a higher-dimensional space to make them separable in a nonlinear way. To verify the effectiveness of SVM I used multiple numbers of C value. The values tested, using K-fold cross-validation are: 0.5, 1.0, 2.0. Choosing the less C to improve generalization, reducing overfitting risk. Less margin means less rigidity on error tolerance to find a solid separation margin.



Testing the model on C = 1.0 the results in the validation phase were:

- The correctly classified instances are on the total 84.69%.
- 86.13% of the examples classified as positive by the model were positive.
- The positive instances were correctly identified for 88.62%

Testing the model on C = 1.0 the results in the testing phase were:

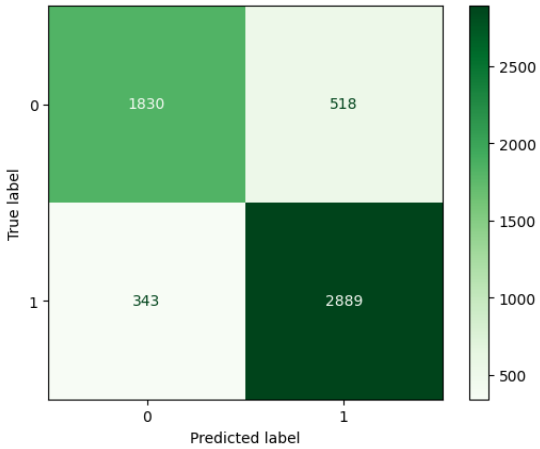
- The correctly classified instances are on the total 84.56%.
- 84.79% of the examples classified as positive by the model were positive.
- The positive instances were correctly identified for 89.38%

The confusion matrix, of the testing set, crosses the real values with the values predicted by the model.

The four cells represent:

- 1830 are the true negatives
- 518 are the false positives
- 343 are the false negatives
- 2889 are the true positives

The model is better at correctly identifying class 1 (high recall) than class 0. Great for verifying that students at risk of depression will be correctly flagged.



Performance Summary

Analyzing the performance of the three models, it is observed that SVM achieved the best results in terms of accuracy, recall, and the number of false negatives, proving to be the most effective model in correctly identifying positive instances while minimizing the loss of relevant cases. However, Random Forest recorded the highest precision and the lowest number of false positives, making it more conservative in classification and particularly useful in scenarios where avoiding incorrect positive classifications is crucial. KNN, on the other hand, showed the least convincing performance, with the lowest overall accuracy and the highest number of false positives.

Modello	Accuracy	Precision	Recall	False positive	False Negative
KNN (k=20)	82.72%	82.86%	88.45%	591	373
Random Forest (n=50)	84.28%	85.39%	87.90%	486	391
SVM (C=1.0, RBF)	84.56%	84.79%	89.38%	518	343

Results discussion and conclusions

The study addressed the problem of predicting depression among students using machine learning algorithms, to develop predictive models that can identify individuals at risk early on. To achieve this goal, a dataset containing academic, psychological, and behavioral information was used and three classification models were tested: K-Nearest Neighbors (KNN), Random Forest, and Support Vector Machine (SVM). Each of these models demonstrated good predictive performance, indicating that machine learning can be a great tool in mental health assessment and prevention of possible risks.

Among the models tested, SVM emerged as the most suitable choice, mainly due to its low number of false negatives. In this context, it is much more acceptable for a student who is not at risk of depression to be falsely flagged as at risk than for a student who is struggling to remain undetected. The consequences of failing to identify a vulnerable individual could be severe, potentially leading to worsening mental health conditions. It is therefore essential to prioritize minimizing false negatives, ensuring that every student with a possible future case of Depression is identified. Consequently, a model with a higher Recall is preferred, as it maximizes the probability of capturing individuals at risk.

Despite the promising results, this study has some limitations that must be acknowledged. The sample size may not be large enough for the proper evaluation of the obtained features and, consequently, of the model. Furthermore, there is a possibility that some relevant psychological or environmental factors that influence the risk of depression were not included in the dataset, potentially limiting the predictive power of the models. Furthermore, the results have not yet been externally validated on independent datasets, which is a crucial step to ensure their reliability in real-world applications.

Future research could address these limitations by expanding the dataset to include more students with even different characteristics and by increasing the number of variables considered, especially those related to social, psychological, and geographical factors that may contribute to depression. Beyond mere prediction, these models could be integrated into decision support systems for schools and mental health professionals, helping with early identification and intervention for at-risk students.

In conclusion, this study highlights the potential of machine learning to detect early signs of depression among students, demonstrating how these technologies could be leveraged to develop more proactive and data-driven approaches to mental health care. By refining these models and implementing them in real-world settings, machine learning could play a crucial role in improving psychological well-being and providing timely support to those who need it most.

Reference

- **SimSensei Project (USC Institute for Creative Technologies)**
  USC Institute for Creative Technologies. (n.d.). *SimSensei*. Retrieved February 6, 2025, from https://ict.usc.edu/?s=SimSensei
- **YouTube Video**
  Amiri, H. (2021, September 27). *[Title of the video]* [Video]. YouTube. https://www.youtube.com/watch?si=guJTwTPpcNWZczW_&v=ejczMs6b1Q4&feature=youtu.be
- **Nature Article on Artificial Intelligence for Mental Health**
  Chen, M., Song, J., & Thirumuruganathan, S. (2021). *Towards better mental health: Artificial intelligence and emotion recognition*. *Nature*, 592(7853), 501-505. https://www.nature.com/articles/s41586-021-03819-2
- **International Journal of Interactive Mobile Technologies Article**
  Zhang, S. (2019). *Developing AI for education: Case study and insights*. *International Journal of Interactive Mobile Technologies (iJIM)*, 13(10), 48-56. https://online-journals.org/index.php/i-jim/article/view/48669/15185
- **Scikit-learn: KNeighborsClassifier Documentation**
  scikit-learn developers. (n.d.). *KNeighborsClassifier*. Retrieved February 6, 2025, from https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html
- **Scikit-learn: RandomForestClassifier Documentation**
  scikit-learn developers. (n.d.). *RandomForestClassifier*. Retrieved February 6, 2025, from https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
- **Scikit-learn: Support Vector Machines (SVM)**
  scikit-learn developers. (n.d.). *Support vector machines (SVM)*. Retrieved February 6, 2025, from https://scikit-learn.org/stable/modules/svm.html
- **Scikit-learn: Model Evaluation Documentation**
  scikit-learn developers. (n.d.). *Model evaluation*. Retrieved February 6, 2025, from https://scikit-learn.org/stable/modules/model_evaluation.html
- **Kaggle: Student Depression Analysis**
  Amiri, H. (2021). *Student Depression Analysis*. Kaggle. https://www.kaggle.com/code/hajraamir21/student-depression-analysis