JOURNAL OF
EMERGING INVESTIGATORS

Article

# Identifying anxiety and burnout from students facial expressions and demographics using machine learning

**Tanishka Shahoo[1], Sophia Barton[2]**
[1] Randolph High School, Randolph, New Jersey
[2] Stanford University, Stanford, California

## SUMMARY

**Anxiety is growing among students today and can interfere with a student's performance or wellbeing, negatively impacting their academic career or their overall quality of life. Thus, it is a pressing issue to distinguish and mitigate this growing mental health crisis, especially among younger students. The first step in doing so is identifying and predicting anxiety before it can significantly impact their lives. Thus, we aimed to determine how we could identify and predict anxiety and burnout in students using both images of their facial expressions and their demographic information. We utilized two classification and two regression machine learning models trained on two different datasets: one dataset containing images of facial expressions and the other containing demographic information and self-reported metrics. We hypothesized that working with image data using a CNN would more accurately identify anxiety than models trained on demographic data, including classification models like the KNN and regression models. Our convolutional neural network (CNN) model was able to identify anxiety in facial expressions correctly with about 81% accuracy. However, our highest performing model trained on the demographic information was our K-nearest neighbors (KNN) model, achieving about 71% accuracy. Thus, although both approaches of analyzing their facial expressions and demographic information can be utilized, the greater accuracy of the CNN confirms our hypothesis that models trained on image data can identify anxiety more accurately than those trained on demographic data.**

## INTRODUCTION

Symptoms of anxiety and depression have increased significantly as of late, especially in young adults. Anxiety among adults in the United States has increased by 1.56% from 2008 to 2018, with 18 to 25 year-olds having experienced the most rapid increase (1). Furthermore, the growth of anxiety in higher education students in particular has only been accelerated by the COVID-19 pandemic, which has introduced numerous novel stressors. This notion holds especially true for college and university students, who exhibited significant levels of anxiety symptoms, depression symptoms, and sleep disturbances during the spread of COVID-19 (2). Therefore, all educational environments should pay heed to this rising issue. It is imperative we identify signs of these mental health issues to then enact initiatives to mitigate them. Thus, we must explore the possible applications of artificial intelligence (AI), to assist in being proactive about unearthing these signs, by monitoring and analyzing student's verbal and nonverbal dispositions.

One approach to identifying a student's nonverbal dispositions is through analyzing and identifying the presence of anxiety in their facial expressions. Existing research has applied AI, in particular the use of convolutional neural networks (CNNs), to perform facial emotion recognition with image data. For example, the Thapar Institute of Engineering and Technology created an engagement index to detect if an online student was engaged, using various deep learning models (3). Their best performing model was ResNet-50, a CNN with 50 layers that is trained on the ImageNet database, which achieved an accuracy of 92.32% (3). Although the researchers used a similar CNN model to ours, they sought to achieve a different goal: identifying engagement in a classroom setting, rather than specific emotions or conditions like anxiety among students.

However, to investigate the verbal dispositions of students to identify their levels of anxiety, other AI models - such as large language models - have been utilized (4). For example, a study of English classrooms explored how ChatGPT could adapt to the anxiety levels of students throughout the writing process (4). The students guided by chatbot as opposed to the human professor reported having less anxiety, leading the authors to conclude that the interactive and personalized style of large language models can also act as an effective tool in identifying anxiety in students and taking action to mitigate it before it worsens, similar to the aims of implementing our machine learning models (4).

On the other hand, traditional methods such as clinical diagnostic tests attempt to identify anxiety through reports from parents, psychiatric screenings, strengths and difficulties tests, and several symptom tests (5). However, the machine learning approach is ideal for classroom settings since our aim is to identify the presence of anxiety purely in the classroom as soon as possible. This focus ensures that teachers can better evaluate their classroom environments and teaching style before a student's anxiety worsens to the point of negatively impacting their wellbeing and academic performance. In a classroom setting machine learning tasks can be performed faster and with more convenience than traditional methods that require longitudinal data collection. For example, machine learning models ensure that after every class, teachers can attain constant insights while traditional assessments would be difficult and time-consuming to complete after every class for each student.

https://doi.org/10.59720/24-195

Thus, in our research, we sought to apply AI algorithms to predict anxiety in students, using a variety of models and two datasets. We first used image data of faces to train a CNN model to make classifications of fear, as a proxy for anxiety. Past studies have corroborated that certain fears such as that of the unknown and pain have been predictors of anxiety in undergraduate students, cementing our justification for making classifications of fear to identify early signs of anxiety (6). We also trained one other classification and two regression models to make predictions about self-reported anxiety and burnout scores of students, using other demographic information such as age, the year of study the student was in, the average number of hours a student studied a week, and self-reported anxiety and burnout scores. We hypothesized that working with image data and using a corresponding CNN model would more accurately identify anxiety than models trained on numerical and categorical data, like the K-nearest neighbors (KNN), support vector regression (SVR), and random forest regression (RF). Our hypothesis was supported as the CNN model trained on facial image data achieved an accuracy of 81%, while our highest-performing model trained on demographic information achieved an accuracy of 71%. Overall, the KNN model outperformed all regression model experiments trained on this demographic information; this is because the task of classification is simpler compared to regression, since regression includes predicting specific values rather than general categories. However, the KNN experiments still performed worse than the CNN model trained on image data. Therefore, although the CNN and KNN are both perform classification, our hypothesis is corroborated by our optimal accuracy being achieved by the CNN, trained on image data, as opposed to the KNN, trained on demographic data.

## RESULTS

### Facial emotion recognition dataset

In order to classify fear as a proxy for anxiety among students, we first trained a CNN using labeled images of different facial expressions. We were unable to find labeled facial data that had anxiety as a label, so we opted to use fear as the closest emotion. We also chose a CNN as the model since that is the most common AI model for image data (7).
To measure the accuracy of our model, we tracked both the validation accuracy, which is the accuracy on the training data, and the testing accuracy, the accuracy on the testing data. We modified several hyperparameters of the model such as the number of epochs (the number of times the CNN works through the training dataset), convolutional layers (layers of the CNN that use filters to detect patterns in parts of the face in the images of the input data), and dropout layers (layers of the CNN that randomly deactivate some of the artificial neurons during training to ensure the CNN does not train on overly specific details and thus minimize overfitting) to try to increase the accuracy. Our most accurate version of this model achieved an accuracy of 81%, using three convolutional layers, two max pooling layers, and was trained for 15 epochs **(Table 1)**.

At first, we faced overfitting with our model, which means that the validation accuracies were consistently higher than testing accuracies, indicating that the model has not learned to generalize to new data well. Therefore, we tried increasing the number of max pooling layers, which prevent the CNN from learning overly specific details and instead allow it to focus on more robust, generalized features across different parts of the image to combat this. However, we ultimately saw a 10% reduction in the testing accuracy when we created a more complex model with a higher number of both convolutional and max pooling layers, as seen in the discrepancy of results between experiment 2 and experiment 1 **(Table 1)**. Experiment 3 was identical to the simpler model of experiment 2 with fewer layers overall but was trained for more epochs; this resulted in a sweet spot in terms of complexity: not overfitting, but also not worsening the testing accuracy **(Table 1)**.

After maximizing the accuracy of the model, we created a confusion matrix to analyze how many correct and incorrect predictions the model made across each of the labels or emotions. The emotion that the CNN model most often wrongly predicted in place of fear was sadness (196 times or 19.6%), whereas the model correctly predicted fear 36.5% of the time **(Figure 1)**.

We performed qualitative analysis on a random subsample of misclassified images to be able to make hypotheses about the errors made by the model. Overall, we found that in the sample of misclassified images, most contained hands covering a portion of the face or widened eyes **(Table 2)**.

### Medical student mental health dataset

To predict the degree to which a student might experience anxiety or burnout, we ran several experiments using a mixture of classification and regression models, trained on numerically represented demographic and self-reported information, such as the age, year of study, average number of hours studied per week, and the self-reported anxiety and burnout scores from medical students of various ethnic origins studying in Switzerland.

We first experimented with a KNN classification model because it is a simple and straightforward model, and classification tends to be an easier task to perform well on compared to regression. We also believed that reducing the self-reported anxiety and burnout values to three groups – “low”, “medium”, and “high” - made more sense in terms of real-world application as well; predicting a student’s exact score can be less helpful in real life than having a more holistic idea of the severity of their mental health struggles.

| Hyperparameter | Experiment 1 | Experiment 2 | Experiment 3 |
|---|---|---|---|
| *Number of Convolutional Layers* | 6 | 3 | 3 |
| *Number of Max Pooling Layers* | 5 | 2 | 2 |
| *Number of Epochs* | 10 | 10 | 15 |
| *Validation Accuracy* | 41% | 44% | 51% |
| *Accuracy* | 69% | 79% | 81% |

**Table 1: Convolutional Neural Network (CNN) Model Results.** Validation and testing accuracies for experiments 1 through 3. All experiments were run with a CNN model with varied values for the number of convolutional layers, max pooling layers, and epochs in the models as described in the table. Convolutional layers, extract spatial data from images, max pooling layers aim to reduce overfitting, and epochs is the number of passes through the training set the model experiences. The validation accuracy is from the performance on the training data while the accuracy, at the bottom of the table, is from the performance on the testing data. These results were obtained from the CNN model we developed in python using the libraries TensorFlow and Keras.

https://doi.org/10.59720/24-195

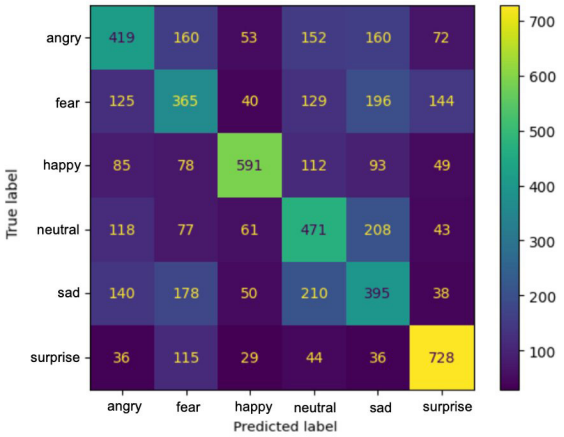



**Figure 1: CNN Confusion Matrix.** Confusion matrix showing the number of times each true emotion label (y-axis) was predicted as each of the six possible emotion labels (x-axis) by the convolutional neural network (CNN). Each cell indicates the count of predictions, with darker colors representing lower values and lighter colors representing higher values. The highest value in the matrix is 728 (true label: "surprise", predicted as "surprise"), while the lowest is 29 (true label: "surprise", incorrectly predicted as "happy"). Accurate predictions are located along the diagonal from top-left to bottom-right. This visualization was generated using the Scikit-learn Python library. A well-performing model will have high numbers along the diagonal (correct predictions) and low numbers elsewhere (incorrect predictions).

The main metric used to evaluate the performance of our KNN classifier was accuracy, which is determined by dividing the number of instances where the KNN model predicted the correct class of the data points in the testing dataset by the total number of data points in the testing dataset, resulting in a percentage value that represents the model's accuracy. We ran experiments to predict both anxiety and burnout levels with the KNN by programming it to predict the range of self-reported anxiety and burnout scores the data points in the testing dataset would belong to. The accuracy of our model in predicting the anxiety class peaked at around 64%, whereas our model peaked at around 71% accuracy when predicting the burnout class. For both these experiments, we observed typical behavior in which there was a sweet spot to the hyperparameter value of *k* or the number of neighbors, which is the number of nearby data points considered when classifying a point or making a prediction. The optimal *k* value for our model predicting anxiety was 5, and it was 20 for burnout, meaning less data points needed to be considered when predicting self-reported anxiety scores than self-reported burnout scores **(Figure 2)**.

We also ran several experiments with varying input features using two different types of regression models to predict both anxiety and burnout. We evaluated our regression models using the $R^2$ value, or the coefficient of determination. This value tends to be between 0 and 1, and the closer the value is to 1, the more accurate a model is and the less residuals, or the more exact the prediction is to the actual data point. When the $R^2$ value is closer to 0, it means there are more residual data points not captured by the regression delineation, meaning the model is less accurate.

In our first set of experiments using an RF model, we predicted anxiety scores. We trained the model using

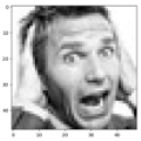
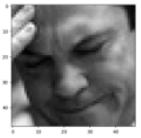
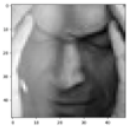
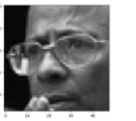
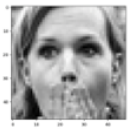
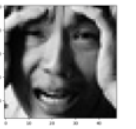
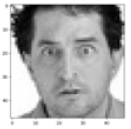
| Image of Fear | CNN Prediction | Image of Fear | CNN Prediction |
|---|---|---|---|
|  | Surprise |  | Angry |
|  | Sadness |  | Surprise |
|  | Sadness |  | Angry |
|  | Sadness |  | Angry |
|  | Sadness |  | Surprise |

**Table 2: Images of fear that were misclassified by the CNN model.** Ten sample images from the FER-2013 image dataset that represented fear but were predicted as a different emotion by our CNN model. We obtained these results by testing the CNN model we made in python using libraries like TensorFlow and Keras.

principal component analysis (PCA) components in place of the raw data as input features. PCA reduces data dimensions by transforming the training data into a set of generalized components that still aim to capture a portion of the variance of the data. The highest variance we were able to capture was 66.32% using two principal components **(Figure 3)**. This is done to reduce the computational power needed to run the models. When running the RF model trained on these generalized principal components, the highest $R^2$ value achieved was only around 0.4, with seven PCA components as the optimal hyperparameter **(Figure 4)**. However, when using the original features from the dataset as input, our RF model consistently performed better, regardless of any hyperparameter values. The best $R^2$ value achieved for this experiment was around 0.62 with a max depth hyperparameter of around five **(Figure 4)**. Although training the model on the raw data as opposed to the principal components yielded higher accuracy, it required more computational power to do so, highlighting a possible justification for using either principal components or the original training data to implement an RF model.

Our second set of experiments used a RF model to predict burnout scores. These exhibited the same pattern as above, in which PCA components worsened the performance of the model. The highest variance we were able to capture for burnout was 69.11% using two principal components **(Figure 5)**. When these PCA components were used, the best $R^2$ value achieved was only 0.35 - with three PCA components and a max depth hyperparameter of four. However, without

https://doi.org/10.59720/24-195

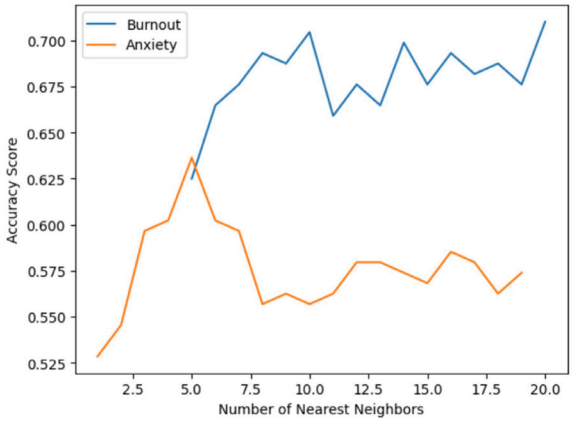



**Figure 2: Accuracy of the K-nearest neighbors (KNN) model when predicting burnout and anxiety.** Accuracy scores for different values of the nearest neighbors, the number of nearest data points considered when making predictions, used in our KNN model when predicting burnout and anxiety. The closer the accuracy score, or the $R^2$ value, is to 1.0, the more accurate the model was. We plotted this graph using the Matplotlib python library.

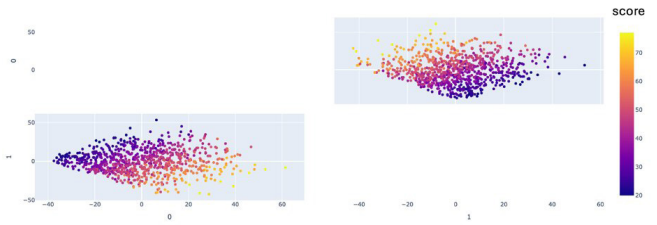



**Figure 3: Visualization of anxiety levels in two principal components of the Medical Student Mental Health dataset.** Each point represents one individual from the dataset. The x- and y-axes correspond to the first and second principal components, which capture the greatest sources of variation in the dataset after PCA transformation. The color of each point represents the individual's anxiety score from the State-Trait Anxiety Inventory (STAI), with yellow indicating higher anxiety (scores closer to 70) and purple indicating lower anxiety (scores closer to 20). Together, these components explain 66.32% of the total variance, meaning they preserve a substantial portion of the dataset's structure while reducing dimensionality. This plot was generated using the Matplotlib Python library.

the use of PCA components, the best $R^2$ value was 0.44 - using a max depth hyperparameter of around two **(Figure 6)**. Our final set of experiments with this medical student dataset utilized the SVR model, which predicts values by finding a line or curve within a margin that best fits the data as opposed to averaging results from multiple decision trees trained on different subsets of the data like RF. We did so as we wanted to see if a different regression model could perform any differently. Similar to our experiments using the RF model, we trained versions of different SVR models with and without PCA components as the input features.

When we predicted anxiety, the highest $R^2$ value achieved using PCA data as input was only 0.32, using seven PCA components. Our SVR model that used the input features directly instead of PCA components only achieved a slightly higher $R^2$ value of 0.35. These results exhibited the same pattern, but on a smaller scale, as our RF model experiments, which showed that using PCA components consistently worsened the performance of the model.

We also predicted burnout levels with the SVR model. The performance of these experiments was worse than predicting anxiety, similar to our RF experiments. Using a kernel value of *poly,* which maps data into a higher-dimensional polynomial space to capture complex relationships, and seven PCA components as the input features, the highest $R^2$ value achieved was only 0.15 (compared to 0.32 when predicting anxiety). Using a kernel value of *poly* and no PCA as input, the highest $R^2$ value was 0.25. Lastly, we experimented with changing the kernel hyperparameter to *rbf*, and the $R^2$ value did increase to 0.30, using no PCA components.

## DISCUSSION

We developed a CNN model to classify the emotions demonstrated in images of faces. Overall, the performance of our CNN model was highly accurate, with about 81% accuracy **(Table 1)**, compared to the 71% accuracy of the KNN **(Figure 2)**, the 0.62 $R^2$ value of the RF **(Figure 4)**, and the 0.35 $R^2$ value of the SVR. Thus, the CNN outperformed all other models as predicted by our hypothesis. However, there are a couple areas for improvement. First, our CNN model most commonly mistook the emotions of sadness, anger, and surprise for fear **(Figure 1)**. After we examined specific images that were incorrectly predicted, we hypothesized that the quality of the dataset itself could be one reason for this. The images that our model incorrectly predicted seemed to be of lower quality than the ones correctly predicted **(Table 2)**. Looking at images that were correctly classified, the facial features such as the mouth and eyes were more clearly defined.

Furthermore, some of the misclassified images depicted widened eyes or contained hands covering certain parts of the face, such as the forehead or mouth, which are human reactions elicited by surprise, anger, sadness, as well as fear, and one of these emotions could often underlie another **(Table 2)**. Humans are complex: there is great variation in how an individual may express their own emotions, and several different emotions can elicit the same or similar reactions in facial expressions. For example, behaviors like crying and smiling can be used to mask one's emotions or can be interpreted in several ways, making it difficult to discern which expressions correlate to which emotions (8). Overall, because of this even the fact that *fear* had to be used as a proxy for anxiety in the first place is a limitation in our progress towards developing machine learning models that aim to predict and identify the preliminary signs of *anxiety*. Therefore, future work towards this endeavor can still be done, specifically with a dataset curated specifically with images that exhibit anxiety. In our KNN, RF, and SVR models we aimed to predict anxiety and burnout levels in students using numerical and categorical demographic data. Our KNN model consistently performed well with about 71% accuracy when using 20 neighbors as the optimal hyperparameter when predicting burnout **(Figure 2)**. The same model only peaked at 64% accuracy when using five neighbors to predict anxiety **(Figure 2)**. The anxiety scores in the dataset were more evenly distributed, which made it harder to form clusters that represented as much data as possible. We believe 15 and 5 neighbors, respectively, worked optimally as both values did not cover too large of a sample of the dataset for classification, which could

https://doi.org/10.59720/24-195

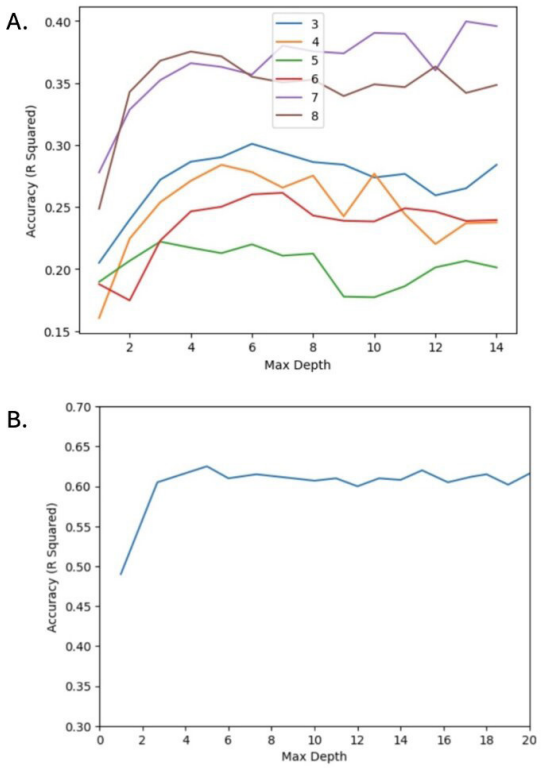



**Figure 4: PCA improves model accuracy. T**he $R^2$ score of the random forest (RF) model predicting anxiety is shown for varying values of max depth. **A)** Displays results using PCA-transformed input, color-coded by the number of PCA components used. **B)** Displays results using untransformed data. Higher $R^2$ values (closer to 1.0) indicate better model performance. PCA generally led to higher accuracy across most max depth values. Graphs were generated using the Matplotlib Python library.

introduce too much variation, yet did not include too small a sample size that might not be representative. The KNN classification model had higher accuracy rates compared to all of the regression experiments. One reason for this is that classification tasks are typically easier to perform well on, in general, compared to regression tasks, because predicting categories is easier than predicting specific values. Predicting the overall presence and severity of anxiety, rather than specific values of self-reported scores could be less helpful in real life as well.

This is why we believed our RF and SVR models performed worse than our KNN model. When predicting anxiety, our RF experiments peaked at a 0.62 $R^2$ value (which is only slightly lower than our KNN model), and our SVR experiments peaked at 0.35 **(Figure 4)**. SVR models are more sensitive to outliers, which can significantly affect the regression boundary and may struggle with complex non-linear patterns more so than RF, limiting the accuracy of our SVR models. These were the highest accuracies we obtained while experimenting with input features and hyperparameters.

Interestingly, in our SVR experiments, we found the *poly* kernel value to be optimal when using the PCA components as input, but the *rbf* kernel value to be optimal when using the regular input data. The *rbf* kernel is required when data is unevenly distributed since it can create different types of dilations depending on the type of data, which is difficult for

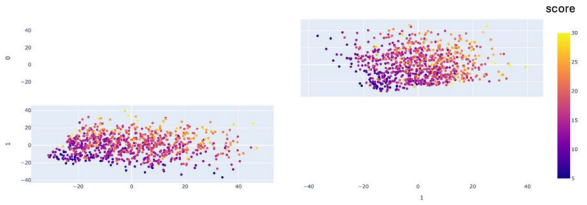



**Figure 5: Visualization of burnout levels in two principal components of the Medical Student Mental Health dataset.** Each point represents one individual from the dataset, plotted using two principal components. The x- and y-axes represent the first and second principal components, which together capture 69.11% of the total variance. Each point is colored according to the individual's burnout score from the Maslach Burnout Inventory–Exhaustion subscale, with yellow indicating higher burnout (scores near 30) and purple indicating lower burnout (scores near 5). These visualizations help show how well PCA separates individuals by burnout level, while reducing the complexity of the input features. The plot was created using the Matplotlib Python library.

the *poly* kernel to achieve. So, when PCA is used and the dimensionality is reduced, there is lower variance compared to the raw input features, meaning the data is more evenly distributed, so *rbf* would not have as much benefit.

When predicting burnout levels with the regression models, the accuracy was lower than predicting the anxiety feature **(Figure 6)**; oddly, this is the opposite pattern we discovered with our KNN classification **(Figure 2)**. Since the anxiety feature was more evenly distributed it was easier to make a split with regression compared to the burnout feature. The highest R2 value achieved when predicting burnout with our RF model was 0.44, and only 0.30 with our SVR model **(Figure 6)**. One hypothesis for this is that the RF model is less sensitive to hyperparameter tuning and can capture nonlinear relationships without requiring feature transformations, unlike SVR.

In the future, we would expand on this work by experimenting with different datasets of demographic information that could have simpler or more evenly distributed features, especially for the anxiety and burnout scores, as these are the labels that we were predicting. Furthermore, the medical student dataset used reflected demographic information and self-reported scores of those of ages 17 to 49. Thus, our work can still be built off of to focus in on the specifically 18 to 25 year-old students, who have experienced the most rapid increase in anxiety as of late (1). Similarly, for the CNN model, we would like to examine if training the model on images of specifically 18 to 25 year-old students, higher quality, and full color would increase the accuracy. Furthermore, we want to utilize a greater variety of regression metrics, such as mean squared error and mean absolute error, to better analyze the performance of our regression experiments.

Overall, the practical use of the CNN model is to utilize images of students' faces to identify their emotions. Our research has trained a fairly effective model to do this. However, implementing this model in classrooms would require the consent of students, teachers, parents, and schools to use images or videos of the students. Therefore, this is a large practical limitation to the efficacy of our research in improving the mental health of students. Furthermore, we were not able to do a longitudinal study, tracking students' anxiety for a year and our models' results (with more diverse participants) due

https://doi.org/10.59720/24-195

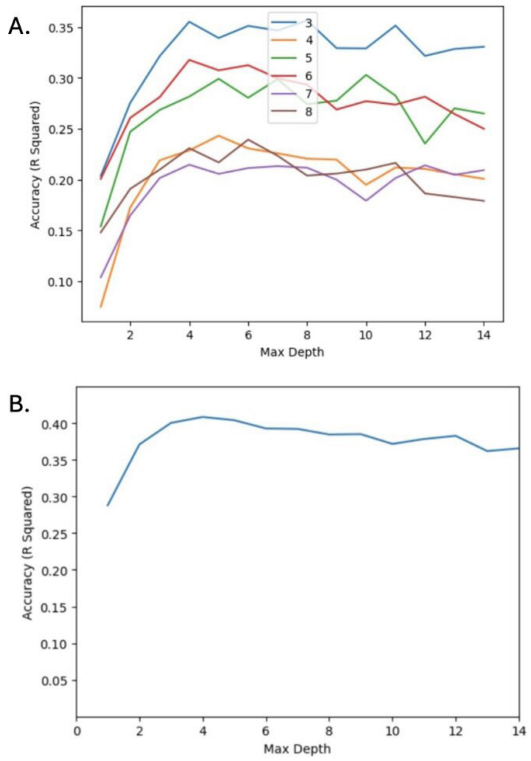



**Figure 6: PCA improves burnout prediction accuracy.** The $R^2$ score of the RF model predicting burnout is shown across different max depth values. **A)** Displays results using PCA-transformed input data, color-coded by the number of PCA components. **B)** Displays results using untransformed input data. Higher $R^2$ values (closer to 1.0) reflect better model performance. Models using PCA generally performed more accurately than those without PCA. Graphs were generated using the Matplotlib Python library.

to a lack of consent from students to do so. To address these limitations, we would like to continue to improve our model's accuracy, and work to gain cooperation and consent from the various parties in educational institutions to approve the use of machine learning models such as ours. If students or adults do not consent to the use of facial recognition AI, our models using demographic information could be used as an alternative to address this limitation as well.

Here, we aimed to make predictions about students' anxiety and burnout levels using a mixture of AI models trained on a variety of image, numerical, and categorical data. In conclusion, our CNN model, which identified anxiety through facial recognition, performed the best of all our experiments, in accordance with our hypothesis, with 81% accuracy **(Table 1)**. Throughout all experiments predicting both anxiety and burnout levels our KNN classification model performed best across the board, followed by our RF model, then the SVR model. Our models aid in taking the first step to recognize the anxieties that students may not communicate or even be aware of, so the education system can work to treat the root causes rather than simply the symptoms.

## MATERIALS AND METHODS

### Facial emotion recognition materials

To perform all the necessary steps of analyzing and preprocessing the data, training and testing the models, and evaluating their accuracies all the code was done in the third version of Python.

To train our CNN, we used a facial emotion recognition dataset, "FER-2013", with image data (7). This dataset contains 28,709 images across seven different emotions: disgust, fear, neutral, sad, surprise, angry, and happy. Since anxiety is not explicitly one of these seven emotions, we decided that "fear" is most similar to anxiety, so it was the emotion of peak interest that we predicted.

Since disgust had only 547 images total, which was overwhelmingly fewer compared to the other emotions, it was dropped from the dataset entirely as an outlier class, to prevent any bias or skewing of model results. Furthermore, disgust is quite dissimilar from fear or anxiety, so it was deemed to not hold importance in the model development process.

Then, we decided to limit the number of images per each of the six remaining classes in the dataset, to try to ensure a more even distribution among classes, prior to splitting into training and testing subsections. We capped the number of images at 4,000 per class to help prevent any downstream skewing of our models.

Afterwards, we normalized the image data in the form of NumPy arrays to mitigate data redundancy and errors in manipulating the data. So, the minimum pixel value of the image was transformed to zero, the maximum was transformed to one, and every other value was converted to be between the range of zero to one, as a decimal. The X data of images was reshaped to reduce the extra dimensions we had for the matrices of each image, while still retaining the 3-D shape and not fully flattening; the y data of emotion classes was reshaped as well.

Lastly, we split our data into two groups, with 80% as the training portion and 20% as the testing portion. The X training data had 18,000 images spread roughly evenly across the six classes. The X testing data had 6,000 images. The y training data had 18,000 emotion labels, correlating to each image in the X training data. The y testing data had 6,000 class labels, correlating to each image in the X testing data.

### Facial emotion recognition methodology

Similar to the models trained on the numerical demographic data, to perform all the necessary steps of analyzing and preprocessing the images, training and testing the CNN, and evaluating its accuracy all the code was done in the third version of Python.

In order to classify images of facial expressions as different emotions, a CNN was used. CNNs have convolutional layers, where a filter or kernel is applied to the input data - which is the image at first - to output a convoluted feature. Afterwards, this convoluted feature is fed into the next layer, and this can repeat for any number of layers.

The first convolutional layer tends to identify basic features of an image, and with more layers, greater complexity can be extracted or learned. CNNs also have max pooling layers, used to reduce the dimension of data, by only retaining the maximum value out of certain subsets of the image pixel values. This reduces the computational power needed to run the model and prevents overfitting to training data as well. Furthermore, dropout layers help prevent overfitting by dropping a certain percentage of random neurons in a single training pass of the model. This percentage is another

https://doi.org/10.59720/24-195

hyperparameter than can be specified. Lastly, flatten layers are used to reshape the dimensionality of data from 3-D or 2-D to 1-D, and are used to transform the shape of convoluted data before being passed into a dense - or regular neural network - layer. Activation layers or functions, such as Reli, are interspersed to introduce non-linearity, which ultimately separates neural networks from linear regression models and allows complex patterns to be learned.

Using libraries such as TensorFlow and Keras, we constructed our own CNN from scratch. First, we had two pairs each of a convolutional layer with a max pooling layer. Then we added another convolutional layer followed by a dropout layer and a flatten layer, separated by max pooling layers, with three activation layers at the end. For all of the convolutional layers, a filter size of 3 x 3 and activation function of Relu was used. The max pooling layers used a filter size of 2 x 2, and the dropout layer had a frequency rate of 15%. Afterwards, the model was compiled using an Adam optimizer, and trained for 15 epochs, meaning that 15 passes over the entire dataset were done to train and tweak the learned parameters of the CNN.

### Medical student mental health dataset

The second dataset used was the "Medical Student Mental Health" dataset, containing a mix of demographic and self-reported numeric data (9). The data was collected from a study in Switzerland, documenting information about medical students related to their environment and mental health. The features of this dataset include: "id", "age", "sex", "glang" - the primary language spoken by the participant, "part" - whether the participant had a life partner or not, "stai_t" - the State-Trait Anxiety Inventory scale of the participant, "mbi_ex" - the Maslach Burnout Inventory-Exhaustion scale of the participant, and more. The two features that we predicted were "mbi_ex", i.e. burnout, and "stai_t", i.e. anxiety, as both were most relevant to our broader goal of analyzing mental health in students.

To preprocess the data, we dropped the features "part", "glang", and "id" from the dataset, as they were deemed to be less relevant towards our goal of predicting anxiety and burnout levels. The distribution for the "age" feature was skewed left, or younger, as most of the values were under 40, so we chose to drop all values greater than 40 as outliers. After viewing the distribution for "sex", the non-binary values were outliers because there was significantly less data compared to the values of men and women; so, we dropped the non-binary data points. Lastly, we split the dataset so that 80% was our training subset and 20% was our testing subset. We also needed to transform data specifically for our KNN experiments. To perform this classification, we needed to convert our features of interest from specific values into categories. Thus, for the burnout feature, we decided a burnout score less than 10 would become the class "low", between 10 and 19 inclusive would be "medium", and 20 or higher would be "high". Also, for the anxiety score, we chose to map a score less than 30 to "low", between 30 and 50 to "medium", and above 50 to "high". We determined these ranges based on standard deviation of both features and the mean, representing the majority of the scores under each broader category accordingly.

For our regression models, we experimented with PCA, which is a method used to reduce input data features into a specific number of components, to reduce dimensionality while still capturing the variance of a feature in a dataset. We transformed all the dataset features - excluding the anxiety and burnout features which would bias a model, since those were our labels to predict - into a varying number of resulting components.

We graphed the variance when different numbers of components were used, to identify the optimal number of components that best represents the dataset but still compresses it. When we reduced our dataset down to two components, and color-coded by the anxiety feature, the variance recorded was 66.32% **(Figure 3)**. This means the two components alone captured that percentage of variance of the anxiety feature in the entire dataset. When we performed the same experiment for the burnout instead of anxiety feature, two PCA components were able to capture the data slightly better with a 69% variance. **(Figure 5)**. In general, a greater number of components captures greater variance, and thus summarizes the dataset more accurately. However, a greater number of components does not reduce the dimensionality of data as much, so there is always a tradeoff to using PCA to transform input features for AI models.

### Medical student mental health dataset methodology

First, we decided to use a KNN model because it utilizes distance metrics to measure similarity, which is intuitive for numerical data and can be computationally efficient for small datasets. KNN uses *k* points nearest to a point of interest in the training data to predict its class and does so for new data points in the testing data. The value *k* is a hyper parameter that can be altered, representing the number of neighbors.
Then, we used RF and SVR models to make further predictions with this Medical Student Mental Health dataset. For our regression experiments, we used a split of 15% testing data and 85% training data to maximize the accuracy of our models, because the initial accuracies using 20% testing data and 80% training yielded significantly lower results compared to our KNN experiments.

RF uses the output of numerous decision trees, or the opinions of many separate models, to come to a robust conclusion, taking the individual conclusions of its decision trees into account. One hyperparameter of this model, *max_depth*, adjusts the number of "branches" each decision tree has, controlling how deep each one goes. We performed hyperparameter tuning with several values for *max_depth* to improve this model's performance.

The kernel is a hyperparameter for SVR models that is useful for identifying the hyperline or hyperplane of best fit with computational efficiency, even as dimensions of a dataset might increase. In our SVR model we used kernels such as *poly* and *rbf*, which are both nonlinear and thus useful for higher dimensions of data.

We performed several experiments training various RF and SVR models, to predict both burnout and anxiety. We experimented with using the raw data as input, as well as a varying number of components from running PCA on the original dataset features. This helped us find what number of components yielded the best predictions for these features in comparison to utilizing the original dataset features as is.

https://doi.org/10.59720/24-195

**Received:** June 26, 2024
**Accepted:** November 11, 2024
**Published:** July 06, 2025

## REFERENCES

1. Goodwin, Weinberger, et al. "Trends in Anxiety Among Adults in the United States, 2008–2018: Rapid Increases Among Young Adults." *Journal of Psychiatric Research*, vol. 130, November 2020, pp. 441-446. https://doi.org/10.1016/j.jpsychires.2020.08.014.
2. Deng, Zhou, et al. "The Prevalence of Depressive Symptoms, Anxiety Symptoms and Sleep Disturbance in Higher Education Students During the COVID-19 Pandemic: A Systematic Review and Meta-Analysis." *Psychiatry Research*, vol. 301, no. 113863, July 2021, https://doi.org/10.1016/j.psychres.2021.113863.
3. Gupta, Kumar, et al. "Facial Emotion Recognition Based Real-Time Learner Engagement Detection System in Online Learning Context Using Deep Learning Models - Multimedia Tools and Applications." *Multimedia Tools and Applications*, vol. 82, no. 1226, 9 September 2022, pp. 11365-11394. https://doi.org/10.1007/s11042-022-13558-9
4. Hawanti, Santhy, et al. "AI Chatbot-Based Learning: Alleviating Students' Anxiety in English Writing Classroom." *Bulletin of Social Informatics Theory and Application*, vol. 7, no. 2, 5 December 2023, pp. 182-192. https://doi.org/10.31763/businta.v7i2.659.
5. Walter, Oscar, et al. "Clinical Practice Guideline for the Assessment and Treatment of Children and Adolescents with Anxiety Disorders." *Journal of the American Academy of Child & Adolescent Psychiatry*, vol. 69, no. 10, 2020, pp. 1107-1124. https://doi.org/10.1016/j.jaac.2020.05.005.
6. Papenfuss, Inka, and Brian D. Ostafin. "A Preliminary Comparison of Fundamental Fears Related to Anxiety." *Journal of Experimental Psychopathology*, vol. 12, no. 2, 2021, https://doi.org/10.1177/20438087211007601.
7. "FER-2013." *Kaggle*. www.kaggle.com/datasets/msambare/fer2013. Accessed 4 November 2023.
8. Kinchella, Jade, and Kun Guo. "Facial Expression Ambiguity and Face Image Quality Affect Differently on Expression Interpretation Bias." *Perception*, vol. 50, no. 4, 2021, pp. 328-342. https://doi.org/10.1177/03010066211000270.
9. "Medical Student Mental Health." *Kaggle*. www.kaggle.com/datasets/thedevastator/medical-student-mental-health. Accessed 25 November 2023.