Principle Component Analysis and Multiclass Classification of Stress

Siva Tejaswini Unnam, 40164034

Github link: https://github.com/TejaswiniU/INSE6220

***Abstract*—People have been disregarding their sleep as a result of their hectic lifestyles. Being an insomniac has a significant negative impact on one's health. The quality of one's nighttime sleep has an impact on one's productivity during the day. The principal component analysis (PCA) is used to determine the association between stress levels and sleeping hours. There are five stress levels in this, ranging from normal/low to intense. The 1.0 accuracy was achieved using the Logistic Regression, K Neighbors Classifier, Naive Bayes, Random Forest Classifier, Linear Discriminant Analysis, and Extra Trees Classifier algorithms.**

***Index Terms*—Principle component analysis(PCA), Machine learning algorithms, stress level**

I. Introduction

The Smart-Yoga Pillow (SaYoPillow) is a proposed edge gadget to aid in the knowledge of the relationship between stress and sleep and to completely materialize the concept of "Smart-Sleeping." It is proposed to use an edge processor with a model that analyses the physiological changes that occur during sleep as well as sleeping habits. Stress prediction for the next day is proposed based on these changes during sleep. The processed stress data, as well as the average physiological changes, are securely transferred to the IoT cloud for storage. There is also a proposal for a secure transfer of any data from the cloud to any third-party applications. The user is supplied with a user interface that allows them to control the data's accessibility and visibility. Including an accuracy of up to 96 percent, SaYoPillow is unique, with security elements as well as consideration of sleeping habits for stress reduction.The relationship between the parameters snoring range of the user, respiration rate, body temperature, limb movement rate, blood oxygen levels, eye movement, number of hours of sleep, heart rate, and Stress Levels (0- low/normal, 1 – medium low, 2-medium, 3-medium high, 4 -high) that was generated from Literature Review is shown in data.csv.

Fig. 1. SaYoPillow purpose

The multi-level classification of stress is first subjected to Principal Component Analysis (PCA).Second, the original, altered, and dimension-reduced data sets are subjected to three common machine learning techniques.The following is the remainder of the report: The PCA algorithm is described in Section I, the applied ML algorithms are described in Section II, and the data set is introduced in Section IV. Section V presents the PCA results, Section VI presents the categorization results, and Section VI closes with a summary of the findings.

II. Principal Component Analysis

In the actual world, data sets are usually large and multifaceted.These features make data processing and storage prohibitively expensive, as well as a data display.On the other side, the manifold hypothesis proposes that real-world high-dimensional data can be discovered on low-dimensional manifolds. As a result, to make interpretation easier, methodologies to transform real-world data sets to low-dimensional spaces are applied.PCA is an example of a technique that aids in the reduction of dimensionality.

Principle component analysis is a multivariate technique for reducing a large number of associated variables into a small number of independent variables called principal components, which capture as much variability in the original variables as feasible. It is a statistical technique for extracting features. By generating new variables that are linear combinations of the original features, the technique preserves the data's variability. The new variables, known as Principal Components (PCs), can be thought of as new data coordinates. By decreasing the distance between the data and its projection onto the PC, the first PC captures the highest variance in the data. The other PCs minimise this distance as well, and they are unrelated (or orthogonal) to the preceding ones.

A. Steps for solving PCA

PCA is used on a data set of n rows and p columns (nxp), which is referred to as a data matrix.

1) **Center data**:
   To begin, we must calculate the average of each column. The average is then deducted from each data input in that column. The centering matrix with nxp dimensionality is

$$Y = HX \tag{1}$$

2) **Covariance matrix**:
   From the centered matrix(Y), the covariance matrix is calculated as

$$S = 1/n\ Y^{T}\ Y \tag{2}$$

   The S matrix is of pxp dimensions.

3) **Eigen decomposition**:
   Using eigendecomposition, calculate the eigenvectors and eigenvalues. Here A* is

$$S = A\Lambda A^{T} \tag{3}$$

4) **Principal component**
   Calculate the converted data using n x p matrix (Z). The observations are represented by the rows of Z,

while the PC is represented by the columns of Z. The number of PCs is the same as the algorithm's dimension in the original data matrix.

$$Z = (Z1', Z2', Z3', \ldots ZI', \ldots\ldots ZP') = YA \quad (4)$$

III. CLASSIFICATION ALGORITHMS

Classification is a method of breaking our data into a manageable number of distinct classes, with its own labels. Machine learning classifiers can be used in one of two ways: supervised learning or unsupervised learning. We utilize well-labeled data to train the computer in supervised learning, whereas unsupervised learning is a machine learning approach that does not require supervision. Regression and classification are two types of supervised machine learning algorithms. For this project, we've used the labeled data. So, using this data set, we'll use two classification algorithms: logistic regression, k closest neighbors, and random forest.

1. **Logistic regression**:
   Logistic regression is a way to figure out how likely it is that a certain outcome will happen based on an input variable. The most common type of logistic regression models a binary outcome, like true/false, yes/no, and so on. This type of outcome can be either true or false. Using multinomial logistic regression, you can figure out what will happen in situations where there are more than two possible outcomes. Logistic regression is a good way to figure out if a new sample fits best into a group.

2. **K closest neighbors**:
   The K-NN algorithm is known as a lazy learner since it is non-parametric and does not require training [8].The k closest points are identified by calculating the distance between each point in the data set and the new input [8]. Distance is calculated using the euclidian or manhattan formula.

$$p(y = c|x, k) = \frac{1}{k} \sum_{i \epsilon N_{k(x,D)}} I(y_i = c) \quad (5)$$

3. **Random forest**:
   Random forests, also known as random choice forests, are an ensemble learning approach for classification, regression, and other tasks that works by building a large number of decision trees during training. For classification tasks, the random forest's output is the class chosen by the majority of trees. The mean or average forecast of the individual trees is returned for regression tasks. Random decision forests address the problem of decision trees overfitting their training set. Random forests outperform decision trees in most cases, but they are less accurate than gradient-enhanced trees. Data features, on the otherhand might have an impact on their performance.

4. **Naive Bayes**
   It's a classification technique based on Bayes' Theorem and the predictor independence assumption. The presence of one feature in a class is unrelated to the presence of any other feature, according to a Naive Bayes classifier. It is straightforward and quick to predict the class of the test data set. It's also good at predicting many classes.

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n \mid y)}{P(x_1, \ldots, x_n)}$$

IV. DATASET DESCRIPTION

This data was taken from 6441 individuals between 1995 and 2010. Out of these, raw polysomnography data is collected from 5804 individuals [4], [5]. Figure 2, The pie chart explains the stress levels population in the population. Level 2 has the highest count and least number in level 4.

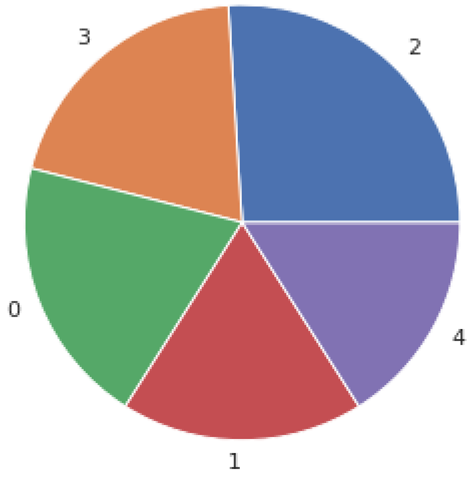


Figure 2. Stress Level

Figure 3 illustrates a box plot that uses boxes and lines to describe the distributions of all normalized attributes. Box boundaries represent the range of the data's central 50%, whereas a central line indicates the median value. Lines extend from each box to indicate the remaining data, with dots added outside the line borders to denote outliers.

Figure 4 presents all of the normalized observations in a strip plot, which is an expansion of the box plot.It represents all the data points and the spread of the points.There are no outliers in this dataset i.e all the points are in the control limits.

Here in Figure 5, the pairs plot shows the relationship between all of the dataset's numerical attributes individually and in relation to one another.

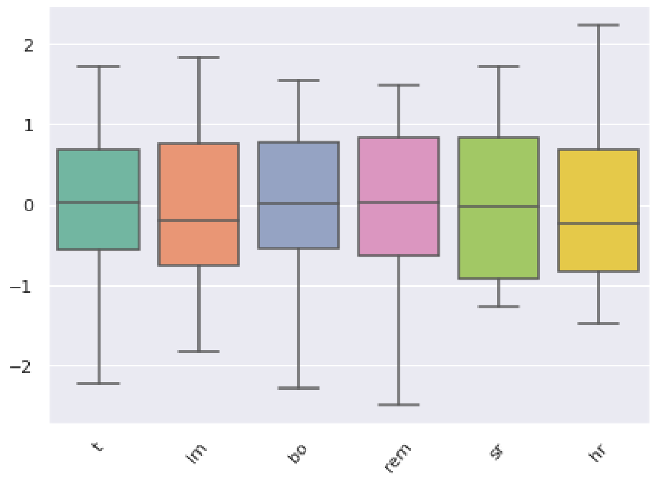


Figure 3. Blox plot

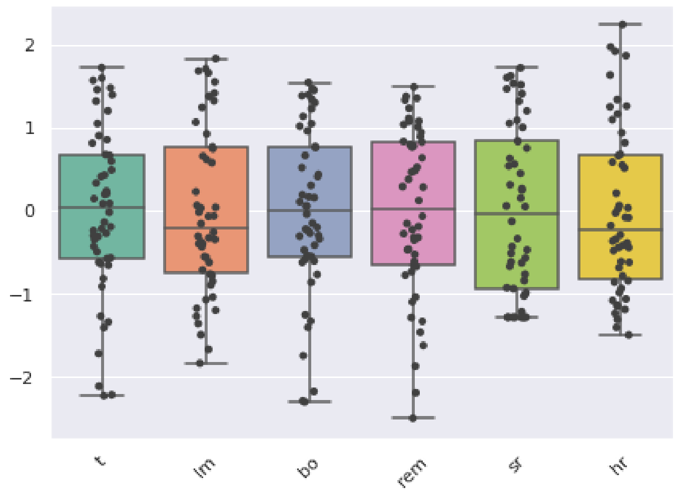


Figure 4. Strip plot

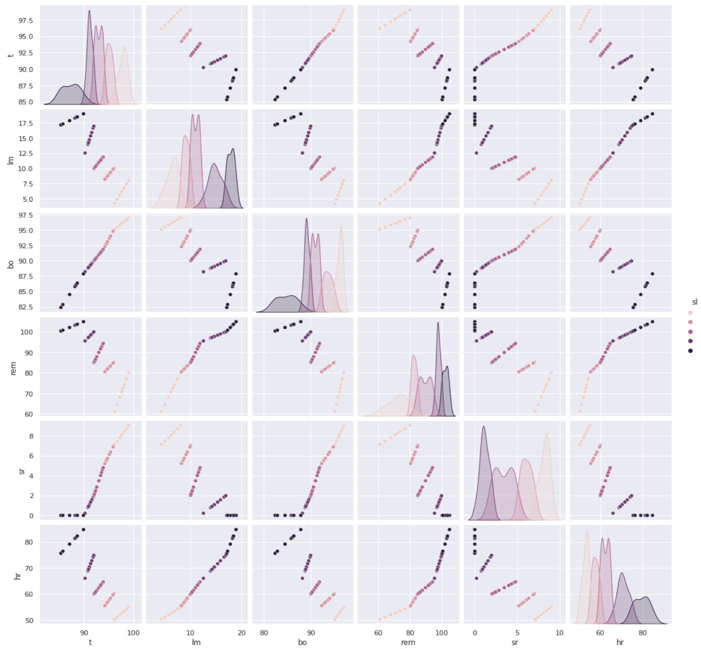
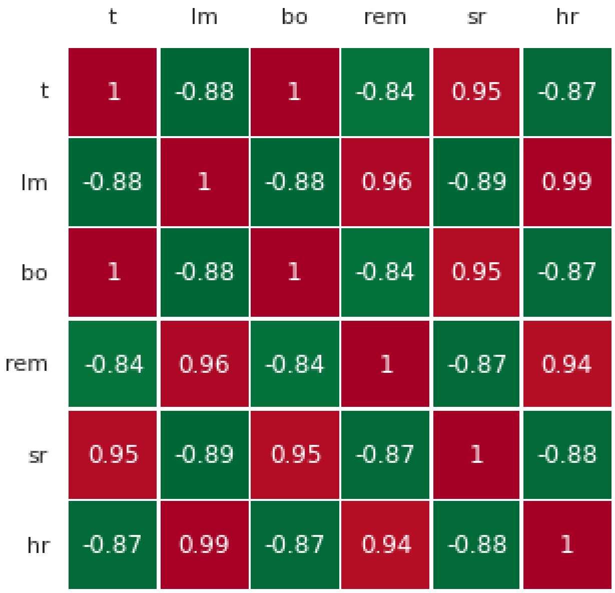


Figure 5. Pair plot

The covariance matrix is a square matrix that indicates the degree of correlation between any two elements of a random vector. A covariance matrix is symmetric and semi-definite in the positive direction. Using equation 6, we try to calculate how the two variables vary with each other.

$$\sigma(x, y) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \tag{6}$$

Figure 6 represents the correlation between the columns in the dataset. The temperature has a direct relationship with body temperature and sleeping hours, limb moment is positively correlated with eye moment and heart rate.



	t	lm	bo	rem	sr	hr
t	1	-0.88	1	-0.84	0.95	-0.87
lm	-0.88	1	-0.88	0.96	-0.89	0.99
bo	1	-0.88	1	-0.84	0.95	-0.87
rem	-0.84	0.96	-0.84	1	-0.87	0.94
sr	0.95	-0.89	0.95	-0.87	1	-0.88
hr	-0.87	0.99	-0.87	0.94	-0.88	1

Figure 6. Covariance Matrix

V. PCA RESULT

PCA was used to minimize the data set's dimensions. The procedures in Section Il were used to decrease the data set from seven features (p = 6) to r features with r < 6. The n x p data set is reduced to a smaller size using the eigenvector matrix (A). The eigenvector matrix obtained after applying PCA is

$$\begin{bmatrix} -0.4085,-0.4568,-0.2270,0.2076,0.0873,-0.7225 \\ 0.4130,-0.3838,0.2234,0.0856,0.7882,0.0587 \\ -0.4084,-0.4465,-0.2493,0.3245,-0.0188,0.6826 \\ 0.4025,-0.4468,-0.5133,-0.5694,-0.2230,0.0256 \\ -0.4076,-0.3111,0.6250,-0.5830,-0.0559,0.0564 \\ 0.4091,-0.3847,0.4266,0.4242,-0.5637,-0.0683 \end{bmatrix}$$

The eigenvalues are the variations in the data that each PC records. The amount of variance that each PC accounts for is visually shown using a Scree plot and a Pareto plot. The following equation [6] is used to calculate the percentage of variation accounted for by the jth PC:

$$\ell_j = \frac{\lambda_j}{\sum_j^p \lambda_j} \times 100\%, \quad for\ j = 1, \ldots, p \tag{7}$$

here λj is the jth element in the PC

The eigenvalue matrix for the dataset is:

$$\begin{vmatrix} 5.652 \\ 0.333 \\ 0.0819 \\ 0.0494 \\ 0.0037 \\ 0.0018 \end{vmatrix}$$

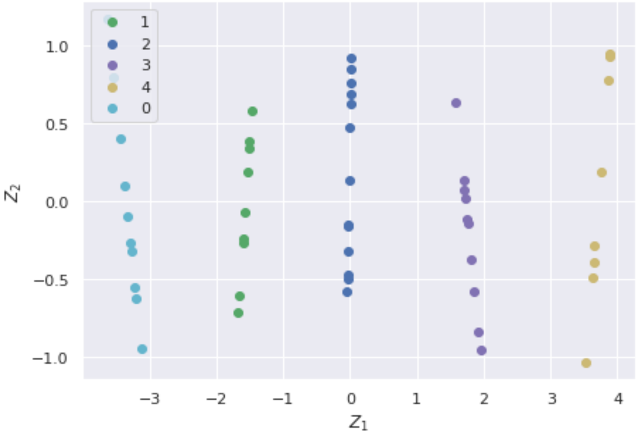


Figure 7. Scatter plot for components

Figure 7 shows the distribution of the components in a scatter plot.

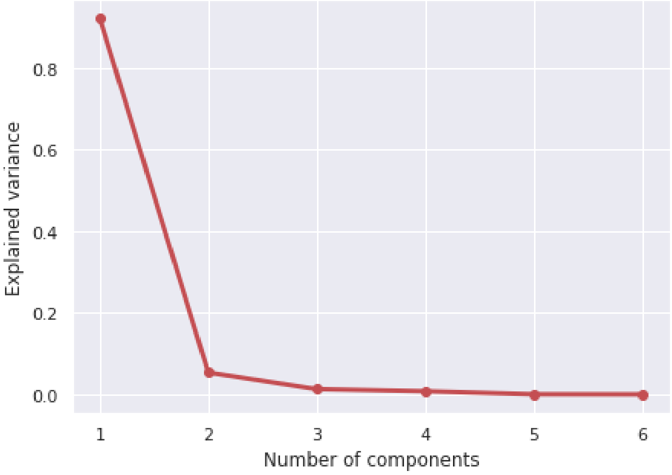


Figure 8. Scree plot

Figure 8 is the scree plot of the principal component vs the variance and the elbow curve started at the second component. The first two principal components account for 99.10% of the total variance.

So for the rest of the analysis, only PC1 and PC2 are considered. The reduced matrix r value equals two(r=2).

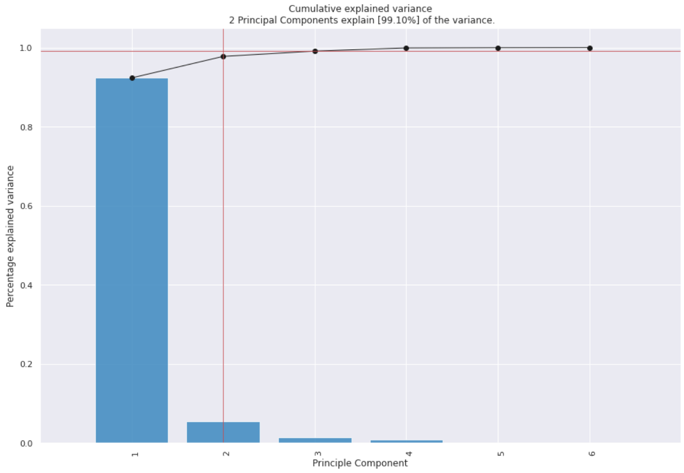


Figure 9. Pareto plot

The first principal component Z1 is

Z1 = -0.4085 **X1** + 0.4130 **X2** - 0.4084 **X3**+0.4025 **X4** - -0.4076 **X5** +0.4091 **X6**
(8)

The second principal component Z2 is

Z2 = -0.4568 **X1**- 0.3838 **X2** -0.4465 **X3** -0.4468 **X4** -0.3111 **X5** - 0.3847 **X6** **(10)**

In both the PC, its evident that there are no negligible factors that need to be excluded from the equation. In PC1, X2,X4 and X6 has positive values and the rest factors are negative.

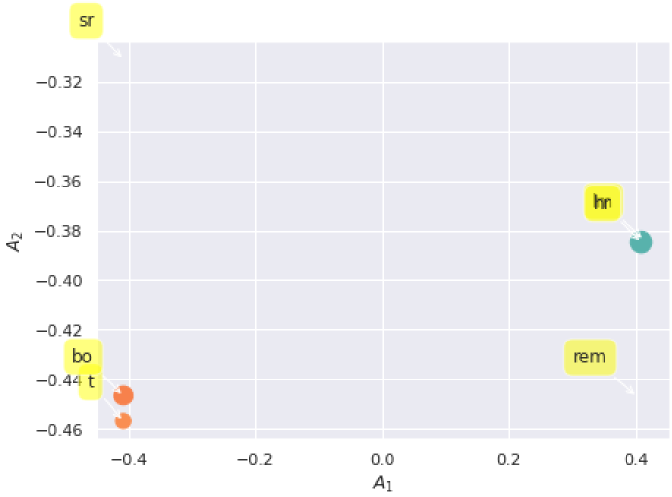


Figure 10. PC coefficient plot

Figure 10 shows the contribution of each variable to the first two PCs using a PC coefficient plot. In other words, it reveals which factors are involved in the first two PCs in a similar way. The graphic depicts the coefficients in Eqs. (9) and (10) in a visual manner (11).

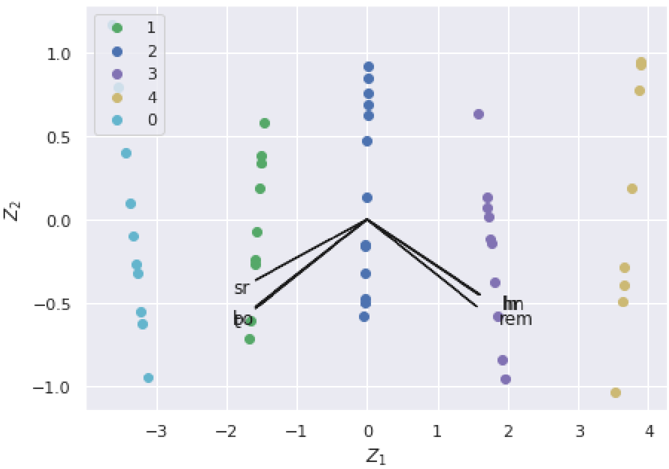
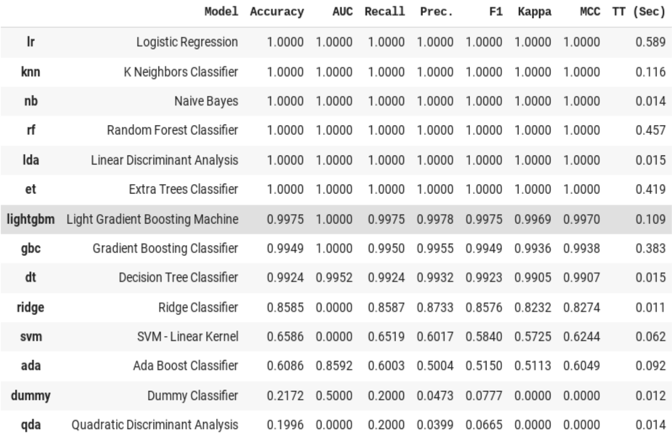


Figure 11. Biplot

Figure 11 is a biplot where the x and y-axis are PC1 and PC2 respectively. All the observations are marked as dots, and rows of the eigenvector matrix are represented as vectors.The angle between the vector and the axis tells about the contribution of the variable. If the angle is small it has a huge contribution, as the angle increases the contribution decreases.

VI. CLASSIFICATION RESULT

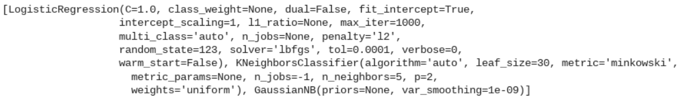
For the classification and graphical representation of the results used by Pycaret and shap." Training data" refers to data used to train the algorithm. This will aid the algorithm in label formation. We used 70% of the data for training. The remaining data is utilized to assess the algorithm's accuracy and efficiency. Stress levels are selected as the target. Using “compare_model” an inbuilt method available in pycaret. It compared all of the models in the library and returned the averaged cross-validated performance metrics. Then the result of the best model with the inbuilt method gave as “Logistic Regression”.



	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
lr	Logistic Regression	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.589
knn	K Neighbors Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.116
nb	Naive Bayes	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.014
rf	Random Forest Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.457
lda	Linear Discriminant Analysis	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.015
et	Extra Trees Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.419
lightgbm	Light Gradient Boosting Machine	0.9975	1.0000	0.9975	0.9978	0.9975	0.9969	0.9970	0.109
gbc	Gradient Boosting Classifier	0.9949	1.0000	0.9950	0.9955	0.9949	0.9936	0.9938	0.383
dt	Decision Tree Classifier	0.9924	0.9952	0.9924	0.9932	0.9923	0.9905	0.9907	0.015
ridge	Ridge Classifier	0.8585	0.0000	0.8587	0.8733	0.8576	0.8232	0.8274	0.011
svm	SVM - Linear Kernel	0.6586	0.0000	0.6519	0.6017	0.5840	0.5725	0.6244	0.062
ada	Ada Boost Classifier	0.6086	0.8592	0.6003	0.5004	0.5150	0.5113	0.6049	0.092
dummy	Dummy Classifier	0.2172	0.5000	0.2000	0.0473	0.0777	0.0000	0.0000	0.012
qda	Quadratic Discriminant Analysis	0.1996	0.0000	0.2000	0.0399	0.0665	0.0000	0.0000	0.014

Figure 12. Compare models

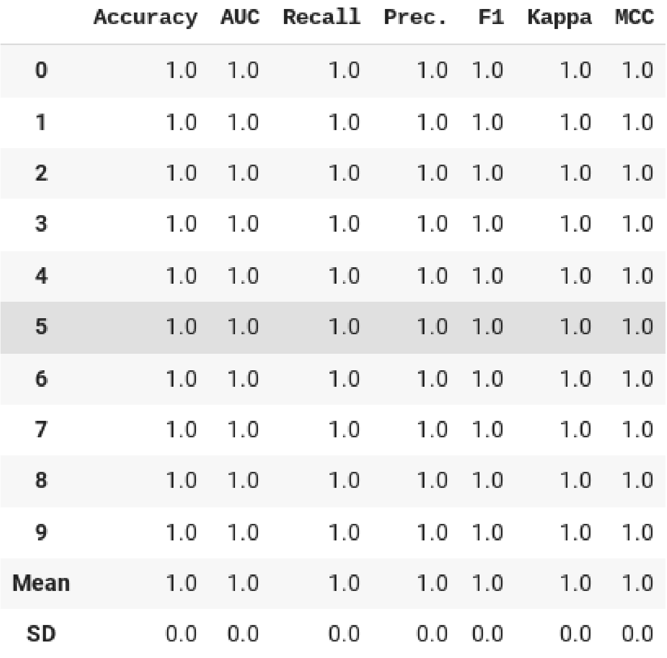
Since the algorithms are showing similar values and the best model has default ‘n_select’ as 1. To know the top algorithms, the factor can be changed as per the requirement. n_select=3



```
[LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=1000,
                   multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=123, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False), KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                     metric_params=None, n_jobs=-1, n_neighbors=5, p=2,
                     weights='uniform'), GaussianNB(priors=None, var_smoothing=1e-09)]
```

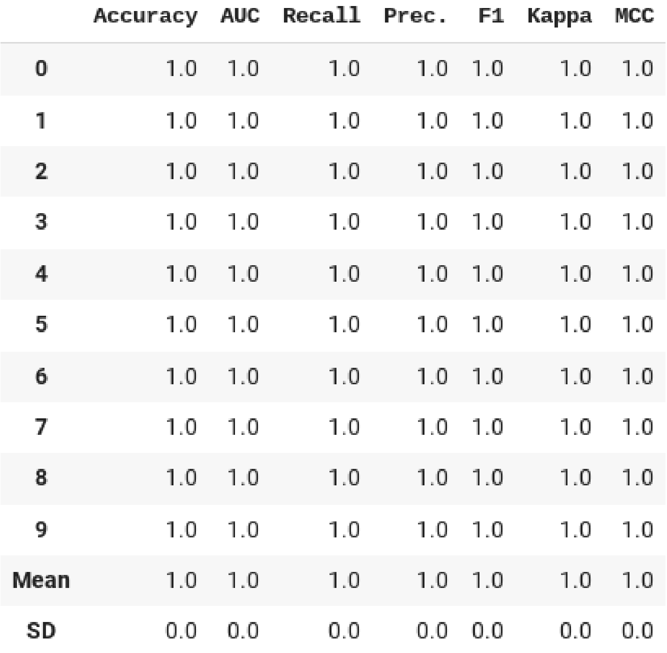
Figure 13.Top3 algorithms

Create model would then be used to train the highest performing model and obtain the trained model output, which you could then use to make predictions. Models are created for the logistic regression, KNN and Random forest. After creating the model for each type, we try to tune them.When the classification is combined with PCA, logistic regression resulted as the best model.



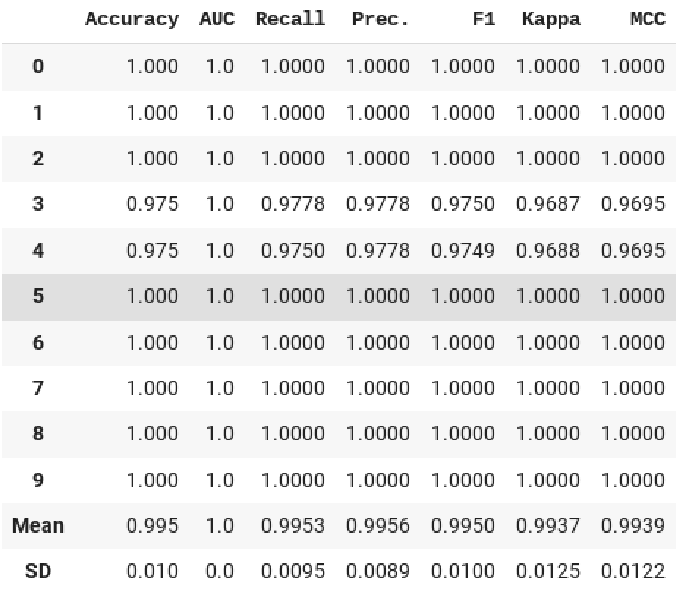
	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
1	1.0	1.0	1.0	1.0	1.0	1.0	1.0
2	1.0	1.0	1.0	1.0	1.0	1.0	1.0
3	1.0	1.0	1.0	1.0	1.0	1.0	1.0
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0
5	1.0	1.0	1.0	1.0	1.0	1.0	1.0
6	1.0	1.0	1.0	1.0	1.0	1.0	1.0
7	1.0	1.0	1.0	1.0	1.0	1.0	1.0
8	1.0	1.0	1.0	1.0	1.0	1.0	1.0
9	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Mean	1.0	1.0	1.0	1.0	1.0	1.0	1.0
SD	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Figure 14. Tuned Logistic regression



	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
1	1.0	1.0	1.0	1.0	1.0	1.0	1.0
2	1.0	1.0	1.0	1.0	1.0	1.0	1.0
3	1.0	1.0	1.0	1.0	1.0	1.0	1.0
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0
5	1.0	1.0	1.0	1.0	1.0	1.0	1.0
6	1.0	1.0	1.0	1.0	1.0	1.0	1.0
7	1.0	1.0	1.0	1.0	1.0	1.0	1.0
8	1.0	1.0	1.0	1.0	1.0	1.0	1.0
9	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Mean	1.0	1.0	1.0	1.0	1.0	1.0	1.0
SD	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Figure 15. Tuned KNN



	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	1.000	1.0	1.0000	1.0000	1.0000	1.0000	1.0000
1	1.000	1.0	1.0000	1.0000	1.0000	1.0000	1.0000
2	1.000	1.0	1.0000	1.0000	1.0000	1.0000	1.0000
3	0.975	1.0	0.9778	0.9778	0.9750	0.9687	0.9695
4	0.975	1.0	0.9750	0.9778	0.9749	0.9688	0.9695
5	1.000	1.0	1.0000	1.0000	1.0000	1.0000	1.0000
6	1.000	1.0	1.0000	1.0000	1.0000	1.0000	1.0000
7	1.000	1.0	1.0000	1.0000	1.0000	1.0000	1.0000
8	1.000	1.0	1.0000	1.0000	1.0000	1.0000	1.0000
9	1.000	1.0	1.0000	1.0000	1.0000	1.0000	1.0000
Mean	0.995	1.0	0.9953	0.9956	0.9950	0.9937	0.9939
SD	0.010	0.0	0.0095	0.0089	0.0100	0.0125	0.0122

Figure 16. Tuned Random forest

'evaluate_model' provides a group of graphs to analyze. For this paper, I have considered the Learning curve. Cross-validation is a technique used in applied machine learning to estimate a machine learning model's skill on unknown data. That is, to use a small sample to assess how the model will perform in general when used to generate predictions on data that was not utilized during the model's training.

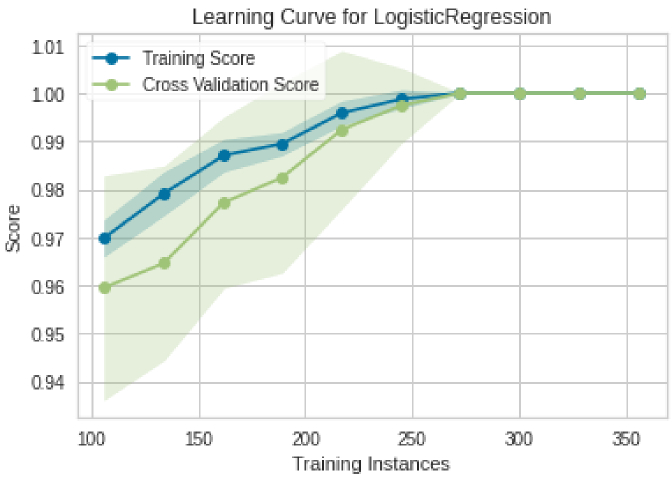


Figure 17. Logestic regression Learning cure

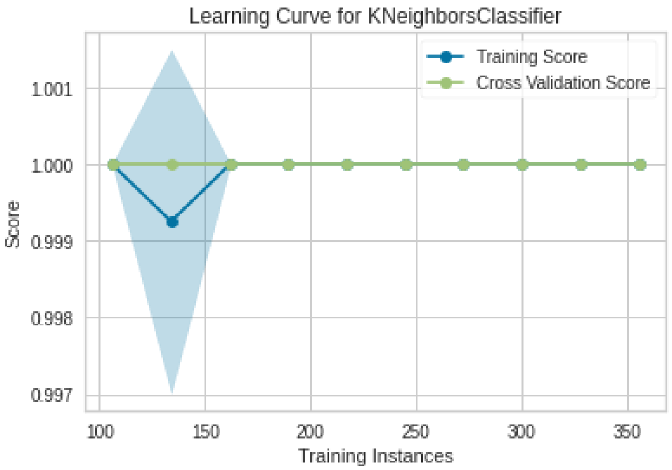


Figure 18. KNN Learning curve

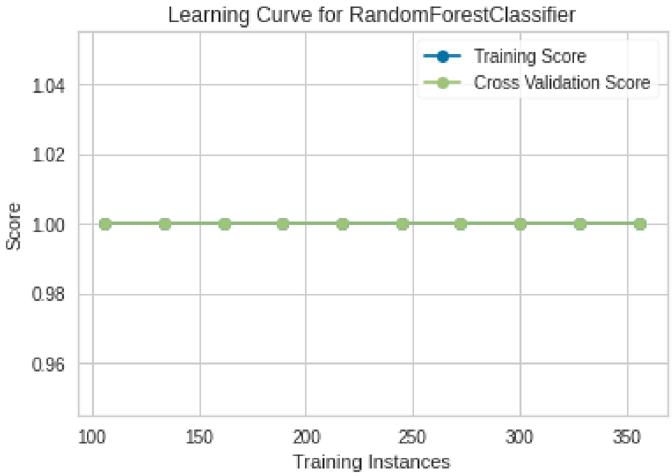


Figure 19. Random forest Learning curve

We utilize a density scatter plot of SHAP values for each feature instead of a traditional feature important bar chart to determine how much impact each feature has on the model output for individuals in the validation dataset. The total of the SHAP value magnitudes across all samples is used to order features. The summary plot gave the overall dependency and the reason plot gave individual impact.

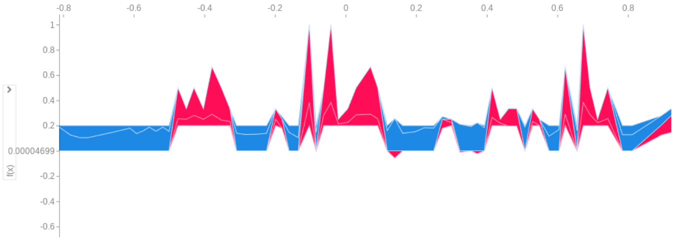


Figure 20. Component 2

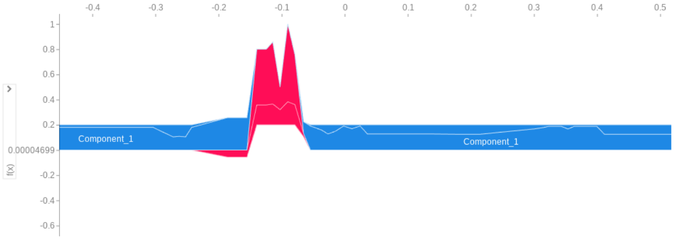


Figure 21. Component 3

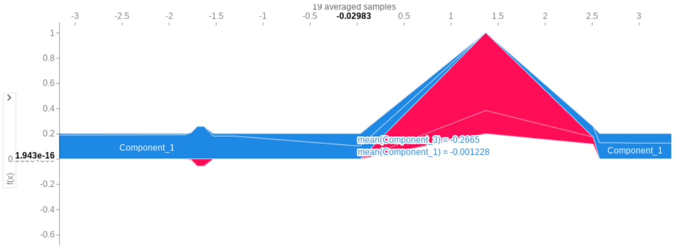


Figure 22. Component 1

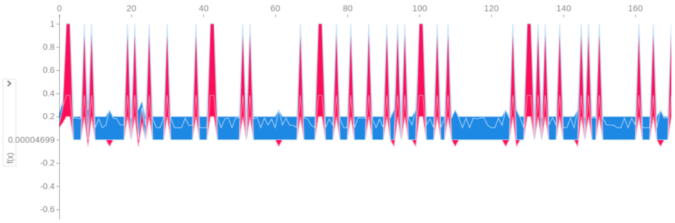


Figure 23. Sampling order

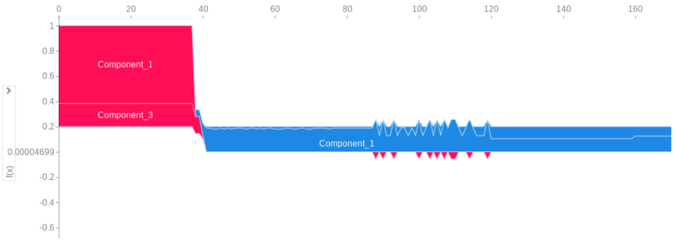


Figure 24. Sampling order by Output

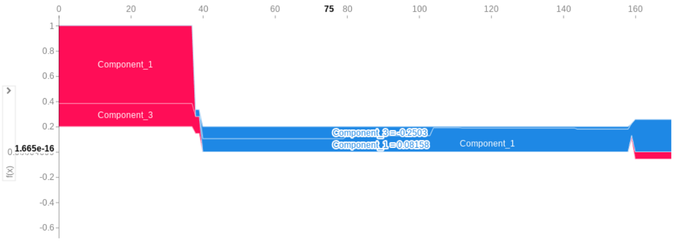


Figure 25. Sampling order by Similarity.

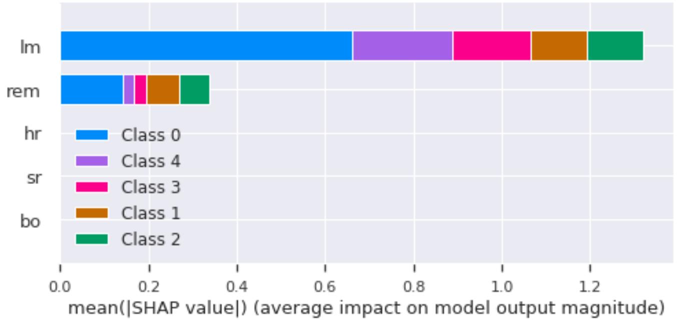


Figure 26. Summary plot

V. CONCLUSION

On five degrees of stress, I used the Principal Component Analysis and two classification algorithms in this study. After using the PCA, I discovered that the first two Principal components account for 92 percent of the variance. As a result, instead of using the entire dataset, only two major components are employed. On the outcome, I used Logistic Regression, KNN, and a Random Classifier. The accuracy of each classifier was used to determine its performance. Based on the results, I believe that Logistic regression outperformed Random forest. SHAP library was used to test the influence of one feature on another. Finally, using Logistic Regression, I was able to get 100% accuracy.

REFERENCES

[1] L. Rachakonda, A. K. Bapatla, S. P. Mohanty, and E. Kougianos, “SaY- oPillow: Blockchain-Integrated Privacy-Assured IoMT Framework for Stress Management Considering Sleeping Habits”, IEEE Transactions on Consumer Electronics (TCE), Vol. 67, No. 1, Feb 2021, pp. 20-29.

[2] L. Rachakonda, S. P. Mohanty, E. Kougianos, K. Karunakaran, and M. Ganapathiraju, “Smart-Pillow: An IoT based Device for Stress Detection Considering Sleeping Habits”, in Proceedings of the 4th IEEE International Symposium on Smart Electronic Systems (iSES), 2018, pp. 161–166.

[3] C. Bishop. Patter Rocognition and Machine Learning 2007.

[4]] S. F. Quan, B. V. Howard, C. Iber, J. P. Kiley, F. J. Nieto, G. T. O’Connor, D. M. Rapoport, S. Redline, J. Robbins, J. M. Samet, and P. W. Wahl, “The Sleep Heart Health Study: Design, Rationale, and Methods.” Sleep., vol. 20, no. 12, pp. 1077–1085, Dec. 1997.

[5] G. Q. Zhang, L. Cui, R. Mueller, S. Tao, M. Kim, M. Rueschman, S. Mariani, D. Mobley, and S. Redline, “The National Sleep Research Resource: Towards a Sleep Data Commons.” J Am Med Inform Assoc., vol. 25, no. (10, pp. 1351–1358., Oct. 2018.

[6] A. Ben Hamza, Advanced Statistical Approaches to Quality, unpublished.