# Student Life

COMP9417 project

Zhu qinshi z5184477

Yang Keyu z5177443

Binbin Xu z5240523

Yueyang Li z5240322

Xinshu Wang z5174787

Semester 3 2019

## I. INTRODUCTION

With rapid development of modern technology, more information is accessible through portable and wearable devices. By analyzing these data can we have a better view of how daily habits affect people's performance and mental health.

Thanks to Studentlife Dataset, many questions like 'why do some students do better than others', 'why do some individuals excel while others fail under similar environment' may have a solution. In this paper, several types of data of 46 students in Ivy League college recorded automatically by smart phone will be used to predict their Flourishing scale and PANAS score by implementing 3 learning algorithms: KNN, decision tree and SVM.

## II. DATASET

**Dataset and Data Grouping:** The dataset we used is divided into input section and output section.

For input part, the first one is Physical Activity Inferences which reflects the status of students as time goes on. And there are four types of student's status which respectively are Stationary, Walking, Running and Unknown. And after calculating the amount of these four types of all the students, we found that the majority part is the stationary status.

The second one is Audio dataset which indicates the participant's physical audio inferences in 10 weeks, and the audio inferences also divided into four types, Silence, Voice, Noise and Unknown. The majority section of students' phone audio is Silence, and then followed by Noise proportion。

Conversation dataset includes each communication's start time and end time, and we could use it to calculate the length of whole conversation of one student.

GPS Location give us the statistic of students' location, which is reflected by latitude, longitude, altitude these three fields. And the network and provider and network type are another important field which can be expressed by one-hot encoding.

The next one is WIFI-location. And it provides us the location of mac address. And WIFI-location calculate each participant's on-campus rough location and we could infer the GPS coordinates of each building [1]. For example, in timestamp 1364357009 and the content in location field is in[kemeny], then we could know the location of the students.

The last three datasets respectively are Light, Phone lock, Phone charge, which has the same data format as the Conversation dataset and reflects the time duration of phone information of students.

For output section, there are two scores we have to predict which are flourish score and PANAS scores. Flourish score is a single psychological well-being score and its dataset provide us the student's life feeling in 8 aspects [2]. And in PANAS datasets, it has two different score metrics which are positive affect score and negative affect score, and for missing value, a median value is add instead of adding zero. The mean value of positive score is 28.6 and negative score is 20.2. This dataset indicates the feeling of each students in past week. According the reference, the flourishing scale is associated with conversation duration and number of co-locations. Similarly, the PANAS scores is related to conversation, activity, and co-locations too.

We have used the combinations method which aims to combine all the features from the input file. And we will discuss about how we group the data explicitly in feature importance criteria of our paper,

## III. METHOD

**Methods Applied:** As described above, three algorithms: decision tree, KNN and SVM are used in the experiment. Firstly, decision tree is a machine learning method which is a tree structure, and each internal node of the decision tree represents the judgment on attribute, different judgment corresponds to different conditions, and each branch represents the output of one judgment result, and finally each leaf node represents only one classification result. Decision tree is a very common classification method, but it can only be established by supervised learning. A bunch of samples and their labels are given to us in supervised learning, and each sample has many attributes and one known classification result. We create this decision tree through learning these samples by the labels, and then this tree can give the correct classification of new data. However, decision tree is easy to have the overfitting problem if there are too many leaf nodes or the maximum depth is high. Secondly, the core idea of the

KNN algorithm is that if most of the k nearest samples in the feature space belongs to one certain category, then the sample also belongs to this category and has the same features which the majority class has in this category. In the decision-making of classification, KNN method only determines the category of the samples which can be classified according to the category of the nearest one or several samples. In other words, the KNN method is only related to a small number of adjacent samples. This is because the KNN method mainly depends on the limited adjacent samples nearby, rather than on the method of identifying the class domain to determine the category. It is better when the sample set to be divided with more overlapping or overlapping class domain compared to other methods. lastly, Support vector machine (SVM) is a kind of generalized linear classifier which classifies data according to the supervised learning method. the decision boundary of SVM is the maximum margin hyperplane for learning samples

**Pre-processing and Feature Extraction:** Several significant features have been introduced in [3], which is attached in the given dataset. Some features are easy to extract while others are not. Here are the examples about pre-processing and extraction of some most important features:

(1) **conversation**. The feature conversation is divided into two parts that are duration and frequency. The sensing data of conversation has two start and end timestamps which is easy to implement. The gap between the end and start timestamps indicate the time of a single conversation. The amount of all conversation represents the duration and the length of all rows in the csv file shows the frequency. Our design is to divide conversation data into different duration of one day to reflect the student's communication information. And the duration is respectively day (9am - 6pm), evening (6pm - 12pm), and night (12pm - 9am). We also split the timestamps into these three sections in other features. (2) **activity duration**. the method of calculating activity duration is similar as that of conversation frequency. The accelerometer records the state of motion of each phone every 2-3 seconds whether is sedentary or moving. As the probability of 2 or 3 seconds is close, we count all records as one time if the state is not sedentary (type 0). (3) **co-location**. the sensing data of Bluetooth is used to calculate the number of co-locations. In order to remove the phones passed by, we should record how many times the certain MAC address has appeared in the sensing data and set a threshold to eliminate these distributions. As for this project, the threshold is set to 6. After that, we calculate the total times of the MAC addresses have appeared. (4) **sleep duration**. instead of using wearable devices, [4] introduce

completely unobtrusive way to predict sleep duration. As the result of their paper, 4 features are used including stationary, silence phone-lock and phone-charge and their ecoefficiencies are 0.5445, 0.3484, 0.0512 and 0.0469 respectively. At last, the feature will be classified into different time period depends on the requirement and the time zone should be set to America/New_York. (5) **PANAS score**. the questions of **PANAS** score can be divided into two parts, the first one is positive score and another is negative score, and we calculate the total score of each students for these two choices. Because the value of students is not completed, and some students features value are missing. For the fairness and calculation, we use mean value instead of zero to replace these values. (6) flourish score: similar as the PANAS score, the flourish score are also missing some values, so we also compare the performance then using these two values.

**Level of Method Complexity:** For KNN algorithm, the time complexity is O(n*m) which n is the amount of training example and m is the number of dimensions. For example, in flourish output model, the feature number is 5 and example training amount is 46. For decision tree algorithm, the time complexity O(n*m*d) which n is the amount of training example, m is the number of the features and d is the depth of the tree. For example, in flourish output model, n =5, m = 46 and d = 8. The kernel of our SVM algorithm is linear. So, the time complexity of the model is O(n*m) which n is the amount of training example, m is the number of the features.

About the time cost of each model, a table is made for each model. The time includes the time of using gridSearchCv to find the best model and the time each model fit the training data.

| Model/Output | Flourish | PANAS positive | PANAS negative |
|---|---|---|---|
| KNN | 0.192 | 0.395 | 0.382 |
| Decision Tree | 0.055 | 0.059 | 0.489 |
| SVM | 0.059 | 0.075 | 0.060 |

**Method Choices and Reasonable Design Choices:** Our group considered some aspects to decide which model and algorithm to use, for example, the accuracy, the ease of use and training time of each model. As the project specification descripted, the aim is to predict by training the input training data. Supervised learning is a kind of algorithm which is based on sample dataset and make a prediction. By analyzing the output such as the flourishing score and PANAS score, we defined this project is a binary classification problem. The extracted features are simple, continues and have linear relation with output. In addition, the total number of students is 46, a small number that can easily overfit for some sophisticated learning algorithms like NN. Therefore, we chose three classic but effective supervised learning algorithms to solve the classification problem. They are KNN, decision tree and Support Vector Machine.

About the design choices, there are several steps for a problem design. There are three outputs: Flourish score, positive PANAS score and negative PANAS score. Each output should be trained by three different algorithms generate nine models. The output data need preprocessing liking dealing with the data 'nan' which indicate a missing data, convert output into binary classification depending on median score and so on. As for the model parameters, by using GridSearchCv in scikit-learn, we can choose appropriate parameters for each model separately depending on the different dataset. however, there are many features but some may not fit specific output, so try to find suitable input can be necessary. First, we will try the features suggested in the reference [3] to create a general view. Then we will try to find some other useful features which may improve the accuracy by calculating the cross-correlation between features and output and implement training. Last, we may use exhaustive method that calculate all possible combinations to find a best and meaningful input for each model, this can take a huge amount of time. Last but not least, final review and conclusion can make the whole experiment better.

**Evaluation Metrics**: For classification problem, the confusion matrix is always useful and intuitive. And since we use median value to decide the whether a score is high or low, the difference in the number of two classes is little, the precision and recall rate can represent the capability of the model. For the better presentation, we use classification_report in model sklearn.metrics to show the result.

## IV. RESULT

**Result Presentation:** Due to space constrain, we only show the result of pre-survey here. The rest of result could be found in Appendix. The following table shows the average accuracy of a 5-flod grid search and the features are those recommended in the reference [3]. For Flourishing scale, the features are conversation duration, conversation duration during evening and number of co-location and for

PANAS score, the features used are sleep duration, conversation duration during day evening, conversation frequency during day and evening, number of co-locations, activity during day and eve, traveled distance and indoor mobility.

| Model/Output | Flourish | PANAS positive | PANAS negative |
|---|---|---|---|
| KNN | 0.696 | 0.696 | 0.739 |
| Decision Tree | 0.652 | 0.587 | 0.739 |

By using different combinations of data to fit model and setting optimal parameters, we get three models which have the highest accuracy on the test data set. Here is the accuracy table of three models when using selected features and when using all the features.

| Model/Output | Flourish | PANAS positive | PANAS negative |
|---|---|---|---|
| KNN | 0.849 | 0.848 | 0.760 |
| Decision Tree | 0.826 | 0.826 | 0.739 |
| SVM | 0.804 | 0.761 | 0.696 |

The cross-correlation numbers and the features used will be discussed in the evaluation part of this chapter.

**Metrics:** the dataset is divided into training set (80% of total dataset) and test set (20% of total dataset). however, since the total number of students is 46, the amount of test set is only 10. The result can vary dramatically according to different random state. As previous discussed, we use precision, recall and f1-score to analyses the result and macro average is used. The result is as follow.

| KNN | precision | recall | f1-score |
|---|---|---|---|
| Flourish | 0.88 | 0.93 | 0.89 |
| PANAS positive | 0.92 | 0.90 | 0.90 |
| PANAS negative | 0.80 | 0.80 | 0.80 |
| Decision Tree | precision | recall | f1-score |
| Flourish | 0.75 | 0.94 | 0.80 |
| PANAS positive | 0.93 | 0.88 | 0.89 |

| | | | |
|---|---|---|---|
| PANAS negative | 0.72 | 0.70 | 0.70 |
| SVM | precision | recall | f1-score |
| Flourish | 0.94 | 0.83 | 0.87 |
| PANAS positive | 0.80 | 0.86 | 0.79 |
| PANAS negative | 0.35 | 0.50 | 0.41 |

As can been seen, the result is decent. KNN and decision tree preforms well in all three scores and SVM preform well in the first two scores but is not good in PANAS negative score.

**Evaluation:** before starting, we calculate the cross- correlations between the feature and the output to remove some features, which can benefit the efficiency of calculation in the future steps. Cross-correlations indication the relation between two sets of data. The higher the score, the greater the relation between them and the positive and negative show the positive and negative affect between two sets. Simply, the NumPy provides a easy way to calculate the cross- correlation in np.corrcoef model. Top three higher values are shown as follow and the specific figure can be found in the Appendix.

| Flourish | feature name | cross- correlations |
|---|---|---|
| | conversation duration day | 0.309 |
| | conversation duration eve | 0.288 |
| | conversation frequency day | 0.265 |
| PANAS POS | activity night | 0.450 |
| | activity day | 0.360 |
| | indoor mobility day | 0.352 |
| PANAS POS | activity night | 0.358 |
| | indoor mobility night | 0.200 |
| | conversation duration eve | 0.207 |

Then we will analyses the evaluation separately.

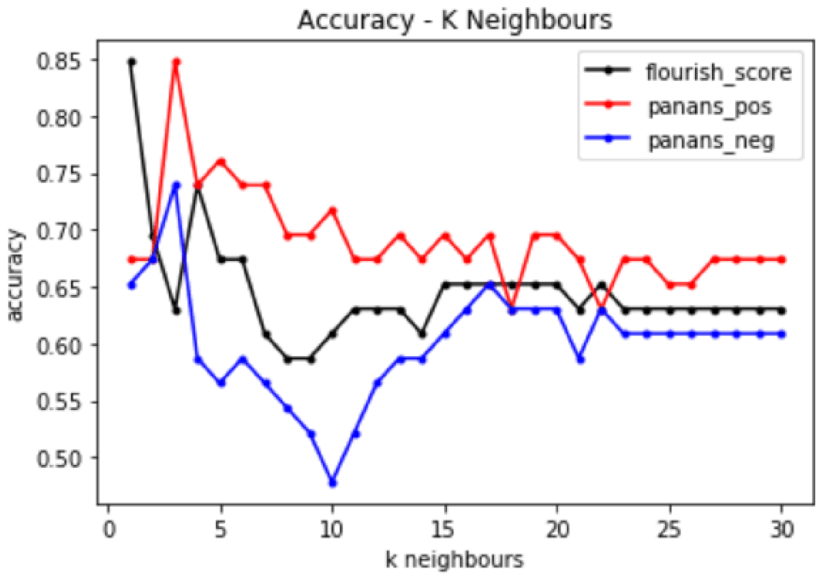
1) KNN:

KNN classifier has 2 important parameters: n_neighbors and weights. The first one indicates how many neighbor points is used and the second one decides how to calculate distance of which the default setting is Euclidean distance. In this project, Euclidean distance is not suitable because the features are using different measurements. The distance will be dominated by the features of larger numbers, so we should do further preprocessing before we train the model.

By applying grid search method, we can easily get the model of optimal accuracy. As described above, we use 80% for training and 20% for testing. As for grid search, we use 5-fold cross validation to prevent overfitting. The best accuracy is 0.695 for original features and the corresponding parameters are {'n_neighbors': 16, 'weights': 'uniform'}. The accuracy is not satisfied.

After this we tried to find a better model by adding or reducing features. It would take huge amount of time to compute when using combinations, but we have a lot. As is described above, the KNN is sensitive to number size. If we use ‘distance’to train model, the best result is 0.783 and the features used are traveled distance and traveled distance during night. As can be expected, the traveled distance is relatively large numbers.

After applying min-max scaler, the result is better. The average accuracy return by the grid search is 0.870 and the 6 features used is conversation duration during night, conversation frequency during evening, co-location, traveled distance during day, indoor mobility during day, and night. We did Min-Max scale before training and use ‘distance’ as weight. The result of n_neighbors shown as follow.

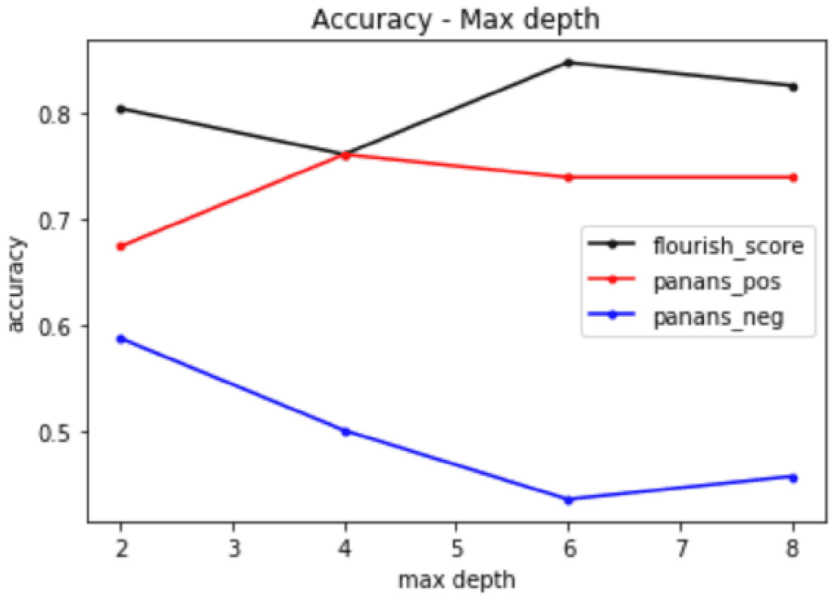


2) Decision tree:

Decision tree classifier has three main parameters the first one is max_depth which limit the depth of tree models and the second one is min_sample_split which control the minimum number of samples required for internal node subdivision． And the third one is min_samples_leaf.

For flourish score, The using features are conversation duration during day, conversation frequency during day, activity during day, traveled distance day, indoor mobility during night respectively represent the total conversation time in one day, conversation frequency, activity time and indoor time at night. And we could obtain the max_depth is 4 and min_sample_split is 2 when we use the grid search and cross validation to prevent overfitting and find the optimal solution for decision tree, which is 0.826.

For panas negative, and if we add the 'min_samples_leaf' parameters, the accuracy get decreased which is 0.717. And the best parameters is {'max_depth': 2, 'min_samples_leaf': 2, 'min_samples_split': 4}



3) SVC

There are three important parameters which respectively is kernel,gamma and C.

kernel: we choose the default value ‘rbf’ gamma: Kernel coefficient for ‘rbf’. C: the penalty coefficient C of the objective function, using the balanced classification interval margin and the wrong sample, the default C is 1.0. We expand the value of C (1, 10, 100, 1000) to seek the optimal solution. After getting lists value of a hyper parameters, we pass these lists to gridSearchCV model.

For flourish score, The best five features we test are conversation duration during day, activity during night, traveled distance during day, evening and night. The highest accuracy is 0.8043478260869565 and the optimal parameter is {'C': 1000, 'kernel': 'linear'}. The graph below is the score of using the optimal parameters. '

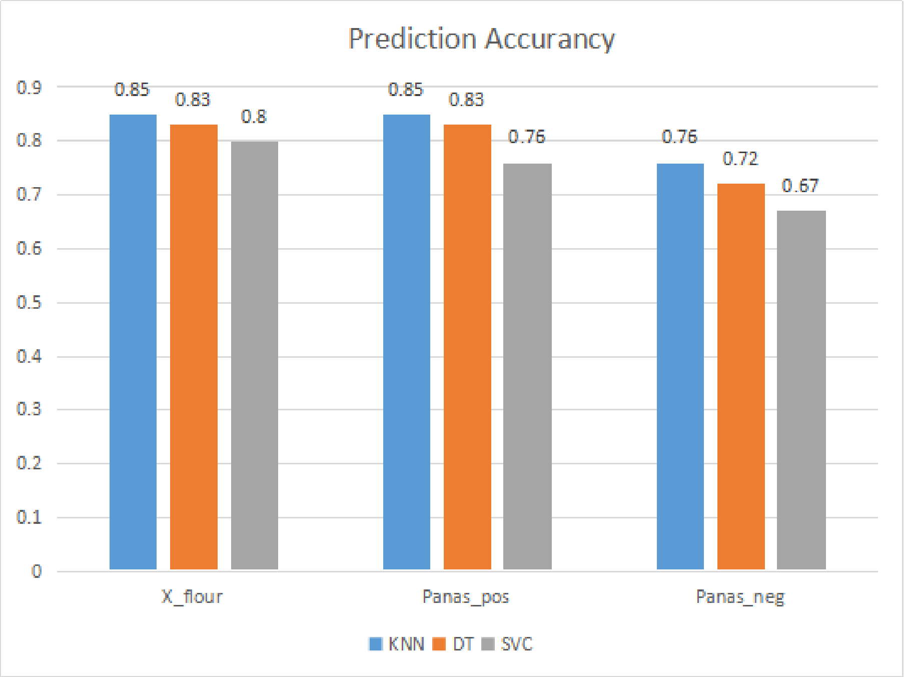
For PANAS positive score, the features we use are by conversation frequency during day and night, activity during night, traveled distance during evening and night, indoor mobility during day from the reference, and then we can obtain the score is 0.760.

For PANAS negative score, we can obtain the score is 0.674. We found that there are two main methods improving the accuracy. The first one is to adjust parameters. Select the appropriate parameter range and use grid search to determine the optimal parameters within this range. Second is adjusting the training set. Different features have different weights, and the full combination of all features are used as training data one by one to obtain the best performing model. This will give us better accuracy than using all feature training. As you can see, the accuracy in table 1 is higher than the accuracy in table2.

**Features:** There are total 8 features we generate from the dataset. They are physical activity inferences, audio dataset, conversation dataset, GPS location, WIFI-location, light, phone lock and phone charge. Different features have different effects on flourish score, PANAS positive score and PANAS negative score. In order to find use which feature to train model can get best performance on test data, we use a combination algorithm to get the full combination of these features and use the generated training set to train the model to find the most accurate model. Finally, we get that the GPS location, WIFI-location and physical activity features influence the flourish score most. The conversation and GPS location feature influence the PANAS positive score most. The conversation and WIFI-location features influence the PANAS negative score most. for example, the coherent value between the conversation duration of day time(9am-6pm) and flourish value is the highest which is about 0.309. The coherent value of activity night features and the PANAS negative score is approximate 0.451 which is the highest relation value. Similarly, there are several relatively high coherent value for PANAS positive score which is the activity of night (0.4507) and the whole day activity of each student(0.4216) and the conversation duration(0.3241).

## V. DISCUSSION

### 1.Comparison of performance among different methods



From the bar chart above, it is clear that the KNN model has a higher accuracy of prediction by using the students' sensing data, while the SVC model shows the worst performance in these three models.

According to the flourishing scale, it is indicated that the KNN model has the highest accuracy, which is 85%. Besides, the DT model comes to the second (83%) and the last one with 80% is the SVC model. The variable Panas positive and negative scores show the similarly patterns in these three models.

On top of that, the KNN model and DT model have the same precision in flourishing scale and Panas positive effect and there is a decrease in Panas negative effect in both models. However, the SVC model have a gradual reduction among flouring scale, Panas negative and positive effect.

### advantages and disadvantages of various methods

Advantage of SVM:

1.The goal of SVM is the optimal hyperplane for feature space partitioning . The idea of maximizing the margin of classification is the core of SVM method.

2. Adding or deleting non-support vector samples has no effect on the model

3. Support vector sample set has certain robustness

4. In some certain applications, it is easy to select kernel.

Disadvantage of SVM:

1. The SVM algorithm is difficult to implement for large-scale training samples. Since SVM solves the support vector by means of quadratic programming, solving the quadratic programming involves the calculation of the m-order matrix (m is the number of samples). When the number of m is large, The storage and calculation of the matrix will consume a lot of machine memory and computation time.

2. It is hard to solve multi-classification problems. The classical support vector machine algorithm only gives the algorithm of the second class classification, but in the practical application of data mining, it generally needs to solve the classification problem of many classes.

Advantage of KNN:

1. It can be used easily to do classification problems as well as regression problemsand has a better performance in many models.

2. It can be used for both numeric and discrete data and is insensitive to the outliers.

Disadvantage of KNN:

1. KNN model has high computational complexity and spatial complexity.

2. There can be sample imbalance problem in some conditions.

Advantage of DT:

1. Decision tree is easy to explain and it follows the same approach as humans generally follow when making decisions.

2. Interpretation of a complex Decision Tree model can be simplified by its visualizations.

Disadvantage of DT:

1. There is a high probability of overfitting in Decision Tree.

2. Information gain in a decision tree with categorical variables gives a biased response for attributes with greater no. of categories.

3. The decision tree model always has a lower accuracy in these problem as we can see in this case.

## VI. CONCLUSION

Through analyzing the data from the remote sensing equipment, we could predict the status of the life of students. Three methods we use to predict respectively are decision tree, KNN and svc, which could have a better performance compared to other models. The best accuracy score of our results for flourish score are respectively 0.85,0.83,0.8 ,and three PANAS positive score are 0.85,0.83,0.76 compared to the PANAS negative score(0.76,0.72,0.67).

Therefore, it is obvious that the decision tree model perform better among these three models. But as for each models, we use the grid search and cross validation to prevent overfitting. What we find is that different combination of features may have direct influence to the final prediction.

Thus, we use the combination methods which compute all the possibilities of feature’s combination and calculate the weight of each features. The disadvantages of this method is that it may take a huge amount of time to figure out the best combination . But after that, our models also have better performance.

**Reference**

[1] StudentLife Dataset https://studentlife.cs.dartmouth.edu/dataset.html

[2] Diener, E., Wirtz, D., Tov, W., Kim-Prieto, C., Choi, D., Oishi, S., & Biswas-Diener, R. (2009). New measures of well-being: Flourishing and positive and negative feelings. Social Indicators Research, 39, 247-266.

[3]. Wang, Rui ; Chen, Fanglin ; Chen, Zhenyu ; Li, Tianxing ; Harari, Gabriella ; Tignor, Stefanie ; Zhou, Xia ; Ben-Zeev, Dror ; Campbell, Andrew. StudentLife: assessing mental health, academic performance and behavioral trends of college students using smartphones. Proceedings of the 2014 ACM International Joint Conference on pervasive and ubiquitous computing, 13 September 2014, pp.3-14

[4]. Z. Chen, M. Lin, F. Chen, N. D. Lane, G. Cardone, R. Wang, T. Li, Y. Chen, T. Choudhury, and A. T. Campbell. Unobtrusive sleep monitoring using smartphones. In Proc. Of PervasiveHealth, 2013.

**Appendix**

| Model/Output | Flourish | PANAS postive | PANAS negative |
|---|---|---|---|
| KNN | 0.19185876846313477 | 0.39545392990112305 | 0.3820650577545166 |
| Decision Tree | 0.05481910705566406 | 0.05862116813659668 | 0.48899102210998535 |
| SVM | 0.05942893028259277 | 0.0754239559173584 | 0.05995798110961914 |

Table1: the time table of GridSearchCv

| Model/Output | Flourish | PANAS postive | PANAS negative |
|---|---|---|---|
| KNN | 0.0075719356536865234 | 0.007790088653564453 | 0.006495952606201172 |
| Decision Tree | 0.007158041000366211 | 0.007523775100708008 | 0.009963035583496094 |
| SVM | 0.007582902908325195 | 0.00670170783996582 | 0.007376909255981445 |

Table1: the time table of Fitting dataset

|  | Flourish score | Positive panas score | Negative panas score |
|---|---|---|---|
| Decision tree | 0.8260869565217391 | 0.8260869565217391 | 0.717391304347826 |
| KNN | 0.8478260869565217 | 0.8478260869565217 | 0.7608695652173914 |
| SVC | 0.8043478260869565 | 0.7608695652173914 | 0.6739130434782609 |

Table3: highest accuracy of three models

|  | Flourish score | Positive panans score | Negative panas score |
|---|---|---|---|
| Decision tree | 0.8260869565217391 | 0.7826086956521740 | 0.7173913043478260 |
| KNN | 0.7826086956521740 | 0.7826086956521740 | 0.6739130434782609 |
| SVC | 0.6304347826086957 | 0.7826086956521740 | 0.6956521739130435 |

Table4: accuracy of three models using all the features

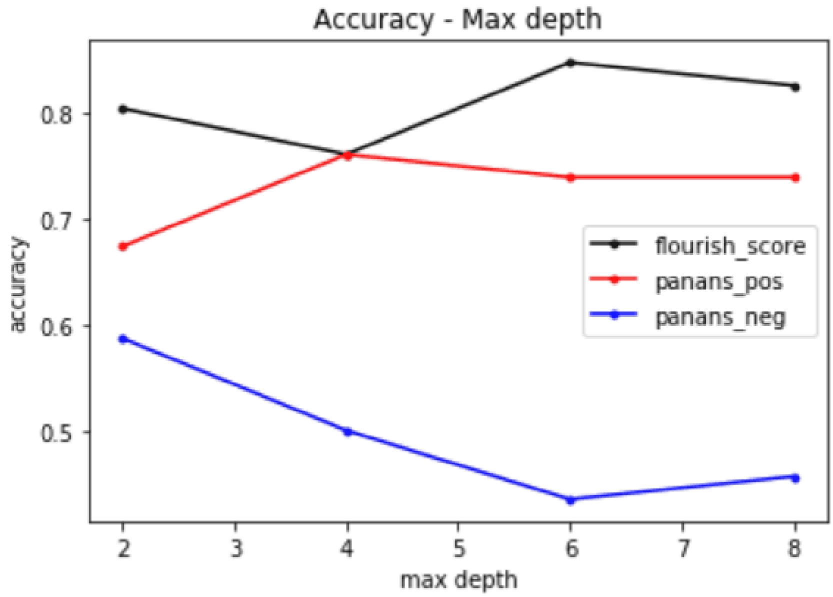


Table5:  adjusting k neighbours in KNN model:

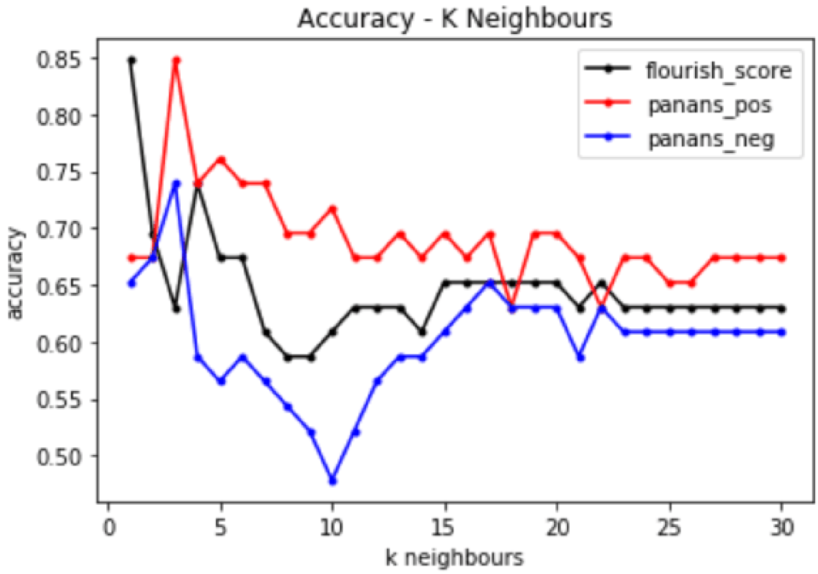
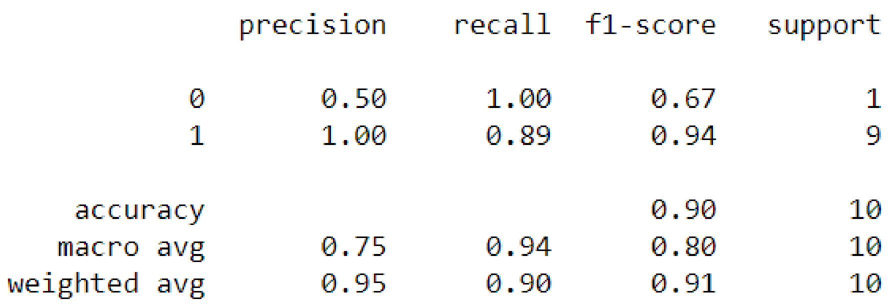


Table5:  adjusting max_depth in decision tree model

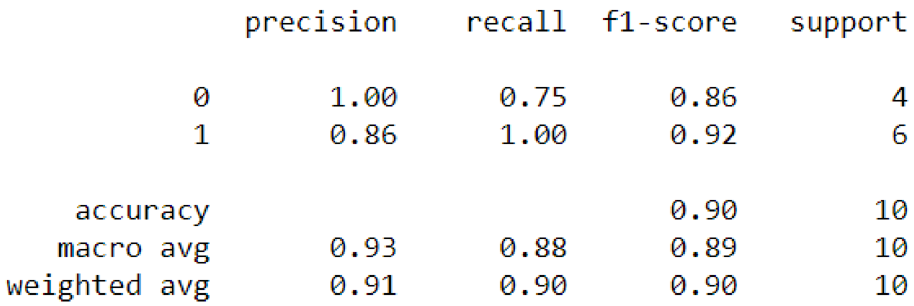


```
              precision    recall  f1-score   support

           0       0.50      1.00      0.67         1
           1       1.00      0.89      0.94         9

    accuracy                           0.90        10
   macro avg       0.75      0.94      0.80        10
weighted avg       0.95      0.90      0.91        10
```

Table6: flourish score accuracy of DT

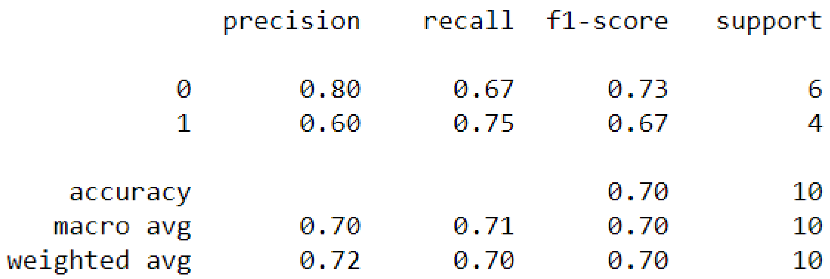


```
              precision    recall  f1-score   support

           0       1.00      0.75      0.86         4
           1       0.86      1.00      0.92         6

    accuracy                           0.90        10
   macro avg       0.93      0.88      0.89        10
weighted avg       0.91      0.90      0.90        10
```

Table7:  panas positive score accuracy of DT

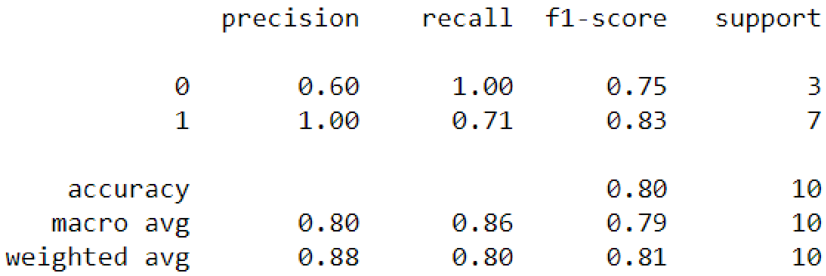


```
              precision    recall  f1-score   support

           0       0.80      0.67      0.73         6
           1       0.60      0.75      0.67         4

    accuracy                           0.70        10
   macro avg       0.70      0.71      0.70        10
weighted avg       0.72      0.70      0.70        10
```

Table8:  panas negative score accuracy of DT

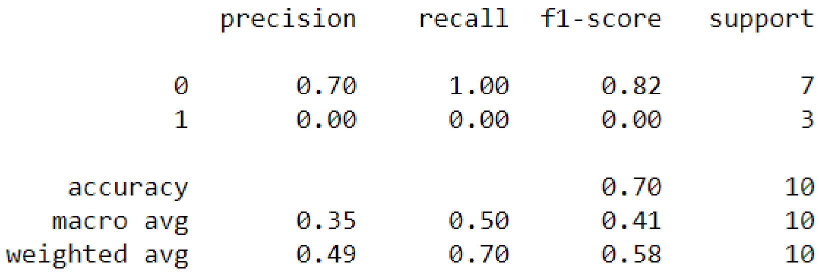


```
              precision    recall  f1-score   support

           0       0.60      1.00      0.75         3
           1       1.00      0.71      0.83         7

    accuracy                           0.80        10
   macro avg       0.80      0.86      0.79        10
weighted avg       0.88      0.80      0.81        10
```

Table9: flourish score accuracy of SVC



```
              precision    recall  f1-score   support

           0       0.70      1.00      0.82         7
           1       0.00      0.00      0.00         3

    accuracy                           0.70        10
   macro avg       0.35      0.50      0.41        10
weighted avg       0.49      0.70      0.58        10
```

Table10: panas positive accuracy of SVC