Exploring the Impact of Context and Personality on Student Mood: A machine Learning Approach

Kumar Swaviman
swaviman.kumar@tudenti.unitn.it
Masters in Data Science
Department of Sociology & Social Research

***Abstract* - This study aims at understanding the relation between various factors and a student's mood and how their personality traits affect this relation.Data used for this study was collected through a smartphone app called i-Log, using questionnaires as well as smartphone sensors. A GPS sensor was used in this case. Data was collected over a month period The datasets were cleaned, pre-processed and merged together to form a master dataset. Statistical tests were performed to check if there exists any difference between the mood over weekdays and over weekends and also between students of various personalities. Machine Learning methods such as Random Forest and Gradient Boosting were used to predict mood. Results showed that the most contributing predictors were varying based on personalities of the students and the time of the week. These findings provide insights into how contextual factors, sensor data and personality differences interact to shape a student's mood.**

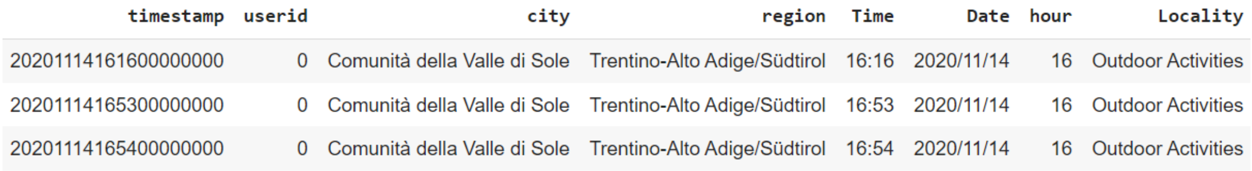
Keywords - Random Forest, Gradient Boosting, Feature Importance, i-Log, GPS, ANOVA, t-test, One-Hot Encoding

1. Introduction

Online data collection methods have transformed research methodologies. Through smartphone apps a larger and diverse population can be reached in order to collect data anytime anywhere which has never been possible through traditional methods. One such application is i-Log which researchers at the University of Trento have developed in order to collect data through questionnaires as well as mobile sensors. Our study made use of the data collected through i-Log application over a month period between November 2022 and December 2022. Data was collected by asking student participants contextual questions like where they are at the moment, who they are with, what they were doing at that moment and how they feel (mood). Simultaneously the app sends sensor data such as light sensors, proximity, gyro, gps coordinates and so on. Certain socio-demographic and personality trait information was collected from participants in the beginning such as, which department they study in, what degree they are enrolled in, nationality and region they live in and estimations were also made on where they stand in terms of big five personality traits. For my specific study I made use of the gps sensor data along with other datasets mentioned above. The purpose was to study the mood of respondents, what are the predictors of mood among students with different personality types and how do these predictors vary between different time diaries.

2. Data Cleaning and Preprocessing

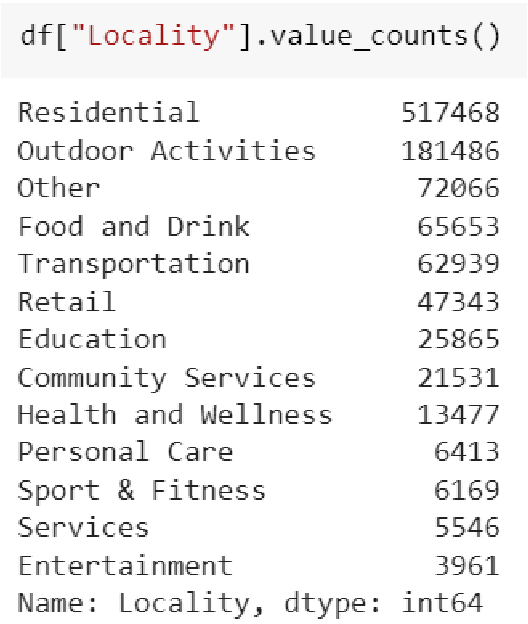
The different datasets collected needed cleaning and pre-processing. The gps sensor data has features such as `timestamp`, `userid`, `suburb`, `city`, `region`, `moving`, `fclass0`, `fclass1`, `flcass2` up to `fclass19`, `code0` up to `code19`, `name0` up to `name19`, `speed`, `experiment` and `bearing`. Clearly I didn't need all these columns so I dropped the unnecessary columns and kept only `timestamp, userid, region, fclass0 to fclass19`. The `timestamp` was in string format so it needed to be separated out into date and time and then each of these two new columns were to be typecasted to datetime format. I extracted `date`, `time` and `hour` from the timestamp column. The major location information was within columns `fclass0` to `fclass19`. However there were a lot of missing entries. Those rows didn't provide any location information so it was wise to drop those rows. Within these location features there were values such as fountain, city center, parking, university and so on. We needed to label these terms so that we can capture the context of the location and form categories. For example `"flcass"` values stating university or school or college should be termed as "Education", values such as parks, grass, playground etc should be under the category of "Outdoor Activities". We manually did the labeling as there were only 119 values and we could easily segregate them into 15 broad categories. Once the category creation is done, we used the `fclass` entries for each row to assign one locality category based on context majority. For example, if an observation has 5 `fclass` column entries at one timestamp as University, Parking, Library, parking_bicycle, college then this row would be assigned a category as Education. Based on the majority of the values, I got an impression of the locality context of the person being at University. Similarly I assigned locality contexts to each rows so that I can sum up the various sparse fclass entries into one column. Finally I had the features I needed such as `userid`, `region`, `date`, `time`, `hour` and `locality`. The data looked like in figure 1.



timestamp	userid	city	region	Time	Date	hour	Locality
20201114161600000000	0	Comunità della Valle di Sole	Trentino-Alto Adige/Südtirol	16:16	2020/11/14	16	Outdoor Activities
20201114165300000000	0	Comunità della Valle di Sole	Trentino-Alto Adige/Südtirol	16:53	2020/11/14	16	Outdoor Activities
20201114165400000000	0	Comunità della Valle di Sole	Trentino-Alto Adige/Südtirol	16:54	2020/11/14	16	Outdoor Activities

Fig 1

The localities into which we segregated all the gps locations, are shown in figure 2.



```
df["Locality"].value_counts()
```

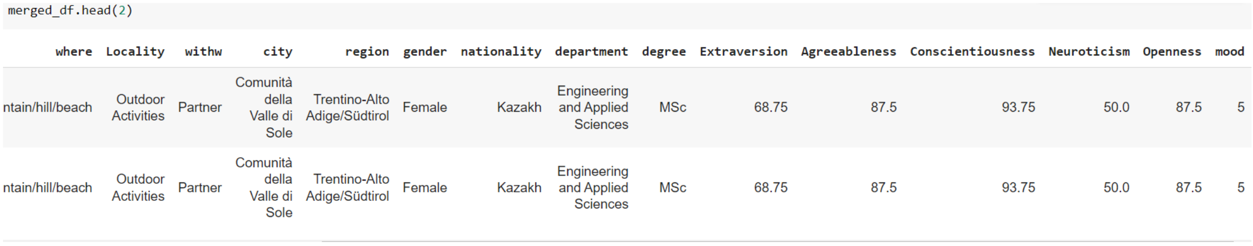
```
Residential             517468
Outdoor Activities      181486
Other                    72066
Food and Drink           65653
Transportation           62939
Retail                   47343
Education                25865
Community Services       21531
Health and Wellness      13477
Personal Care             6413
Sport & Fitness           6169
Services                  5546
Entertainment             3961
Name: Locality, dtype: int64
```

Fig 2

I had the context information in a dataset named "*td_ida.dta*". This contained features such as `date`, `what1`, which tells us what the individual was doing at that moment, `where`, which gives us his/her/their location detail, `withw` feature tells us with whom the person is, `mood` tells us how the person is feeling right now, This dataset didn't have many missing values. The outcome variable for our project, "mood" was in this dataset. So I checked the entries for mood. It seemed there were erratic entries for this feature. There were certain entries such as "travel", "No Information" and "Expired". As the mood can not be "travel", I dropped the corresponding observations. The rows with values "No Information" and "Expired" were there when the respondent either skipped or didn't respond to the push notification. These were of no use to us, so I dropped these entries as well. The mood had 5 categorical entries which I encoded from 1 to 5 with 1 being very sad and 5 being very happy.

The last dataset I used was called *"sociopscicodemo ITA.dta"*. This dataset contained information such as `userid`, `gender`, `nationality`, `department`, `cohort`, `degree`, personality traits scored between 0 to 100. This dataset didn't require cleaning as there were no missing values or outliers. Now that I had all the three datasets cleaned and pre-processed, I merged them all together using inner join based on time and userid. This gave us a master dataframe which had the context info collected through questionnaires, gps sensor data as well as the personality information.

I made certain pre-processing to the master dataframe again. I intended to study the behavior over weekdays and over weekends. For this reason a new feature was derived from the date column, called `"day_type"` having entries as either weekdays or weekends. The dataframe looked like in fig 3.



```
merged_df.head(2)
```

where	Locality	withw	city	region	gender	nationality	department	degree	Extraversion	Agreeableness	Conscientiousness	Neuroticism	Openness	mood
ntain/hill/beach	Outdoor Activities	Partner	Comunità della Valle di Sole	Trentino-Alto Adige/Südtirol	Female	Kazakh	Engineering and Applied Sciences	MSc	68.75	87.5	93.75	50.0	87.5	5
ntain/hill/beach	Outdoor Activities	Partner	Comunità della Valle di Sole	Trentino-Alto Adige/Südtirol	Female	Kazakh	Engineering and Applied Sciences	MSc	68.75	87.5	93.75	50.0	87.5	5

Fig 3

3. Data Visualization

Various data visualizations are performed on the master dataset. This helped us derive useful insights. I begin with the distribution of the outcome variable “mood”. The idea behind plotting the distribution was to check if there exists any severe class imbalance. The figure 4 below shows the distribution of the mood feature. It seems the values aren’t severely out of balance.

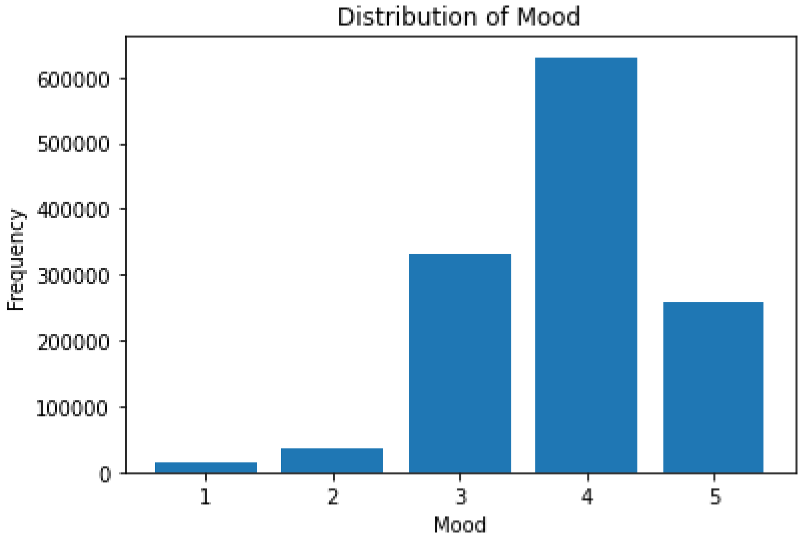


Fig 4

Through extensive Exploratory Analysis I did line plots to see the distribution of average mood over a complete day (24 hour period) for all respondents on weekdays and weekends. Figure 5 shows us that the overall mood dips after 10 AM and spikes back up in the evening.

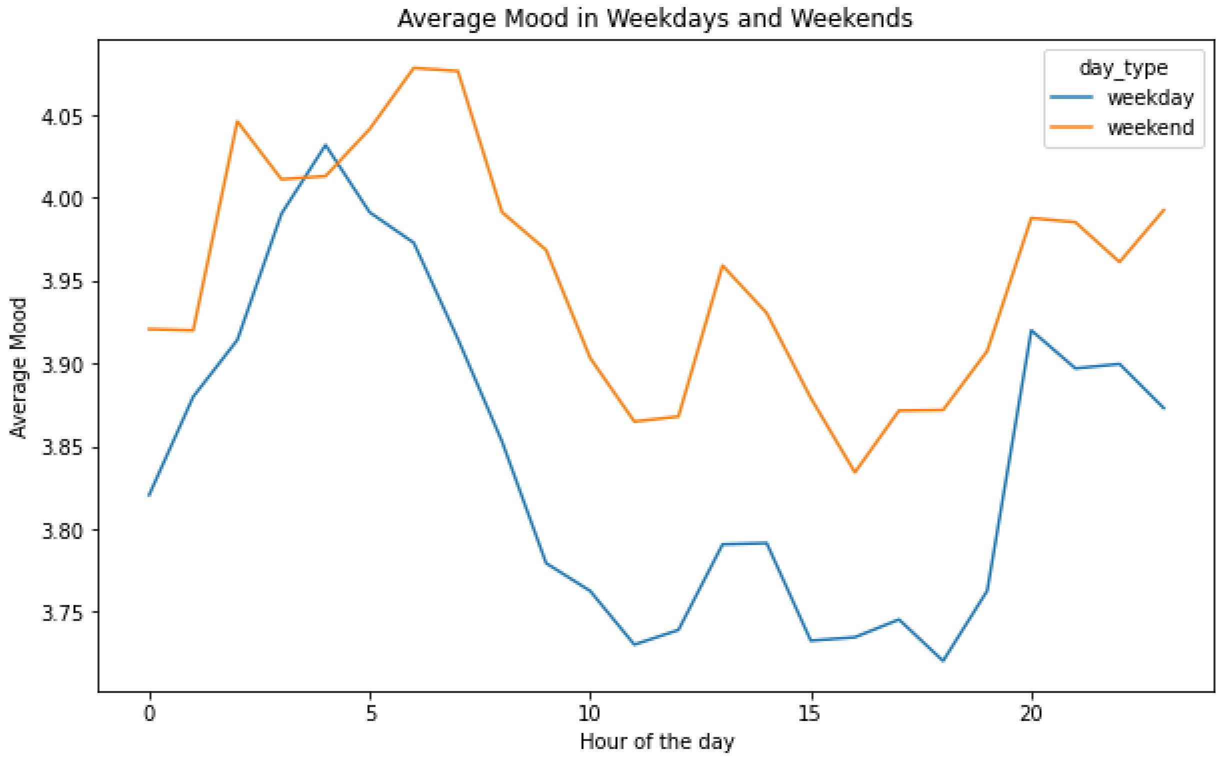


Fig 5

In order to understand the mood of people with certain types of dominant personalities I plotted the mood over a day for Neurotic people (those with neuroticism score greater than 70 out of 100), Extroverts and Conscientious people.We observe very different characteristics for each one of them. The plots are shown in figure 6, 7 and 8.

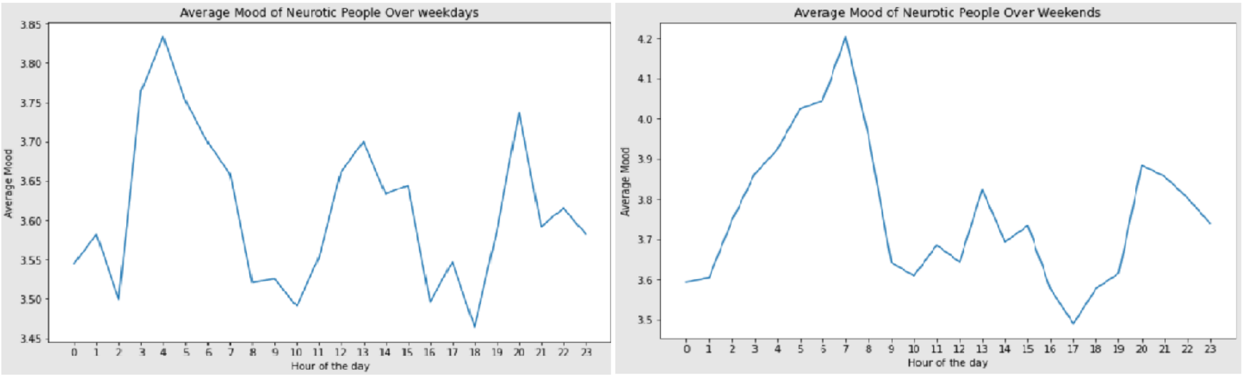


Fig 6

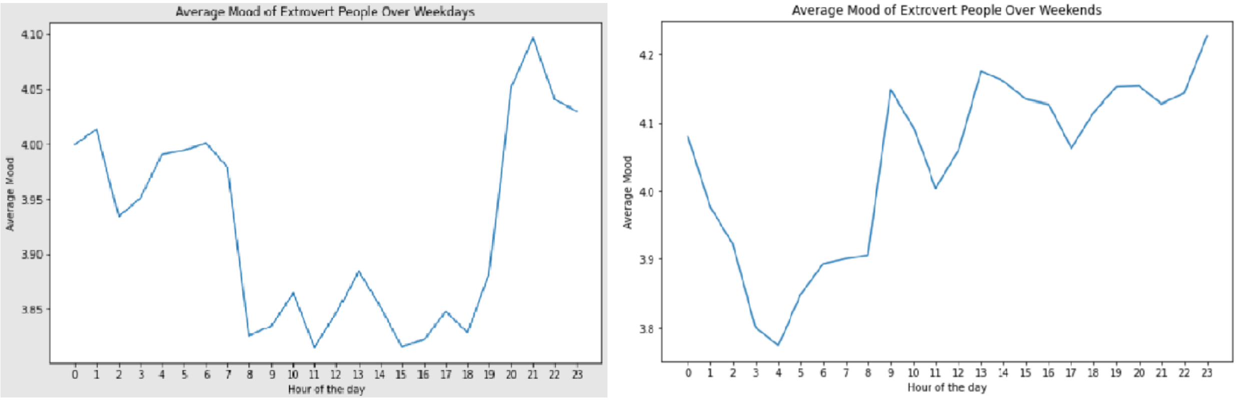


Fig 7

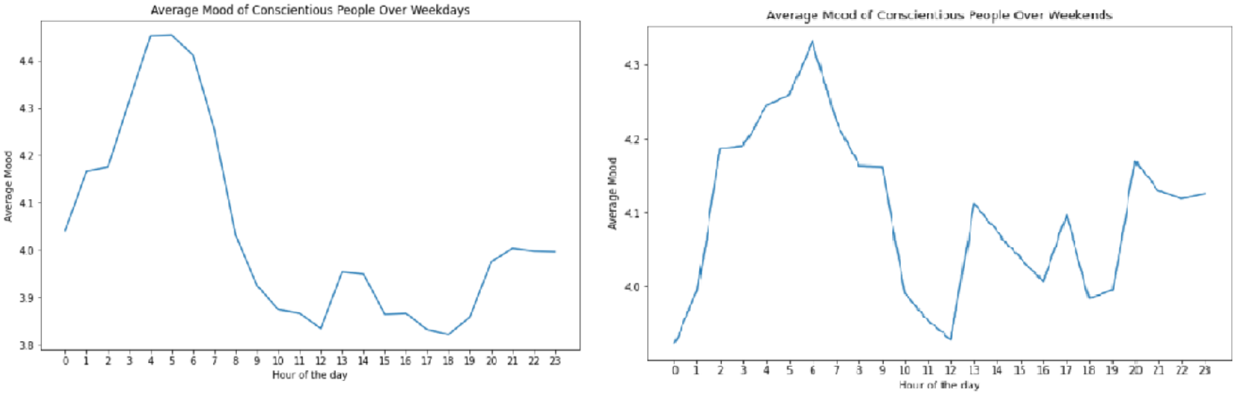


Fig 8

We can observe that for weekdays figure 7 and 8 follow similar patterns. Average mood of people spikes during early morning hours. As the day progresses it sinks and reaches bottom at about noon. Around lunch break, the mood goes up and during the evening when they go back from work, the mood spikes back up again. Though neurotic people also follow the same behavior, their variation in mood is higher compared to the other personality people. On the contrary to the weekdays plots, if we look at the mood over weekends, conscientious and neurotic people show similar behavior with mood at its peak during morning hours and with the progression of the day it gradually sinks. However, for extroverts, as the weekend progresses their mood takes an upward curve until late night perhaps because of their socializing and outgoing nature, quite predictably so.

If we look at figure 9 it tells us mean mood variation by gender of the participants. Though during weekdays both males and females exhibit very similar behavior, during weekends, females tend to be less happier compared to men. This observation is quite striking. Figure 10 tells us the variation in mood over weekdays and weekends by students of different departments. Though most of them follow more or less the same pattern, students from International Relations & Public Administration tend to be significantly unhappy during morning hours whereas students from Agriculture tend to be considerably happier during the weekend evening hours.

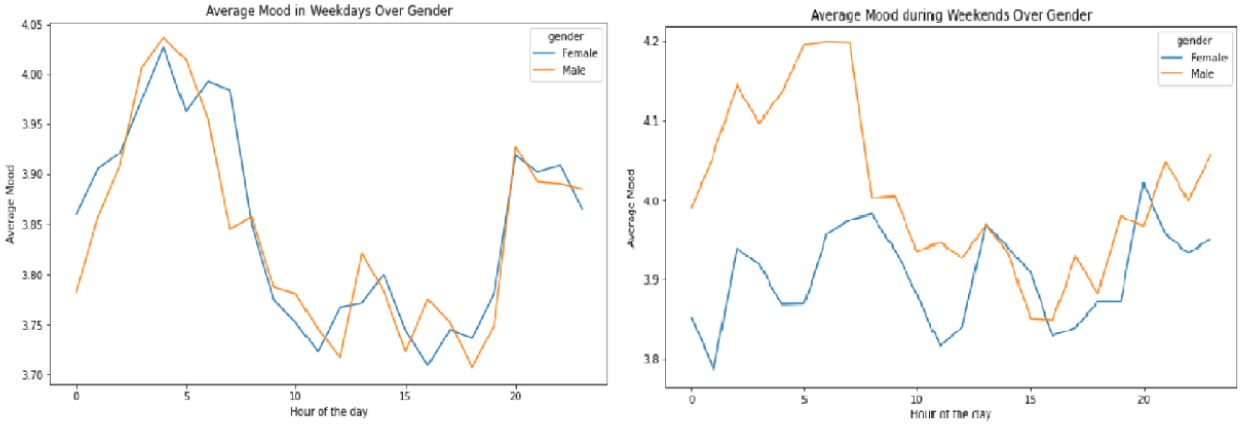


Fig 9

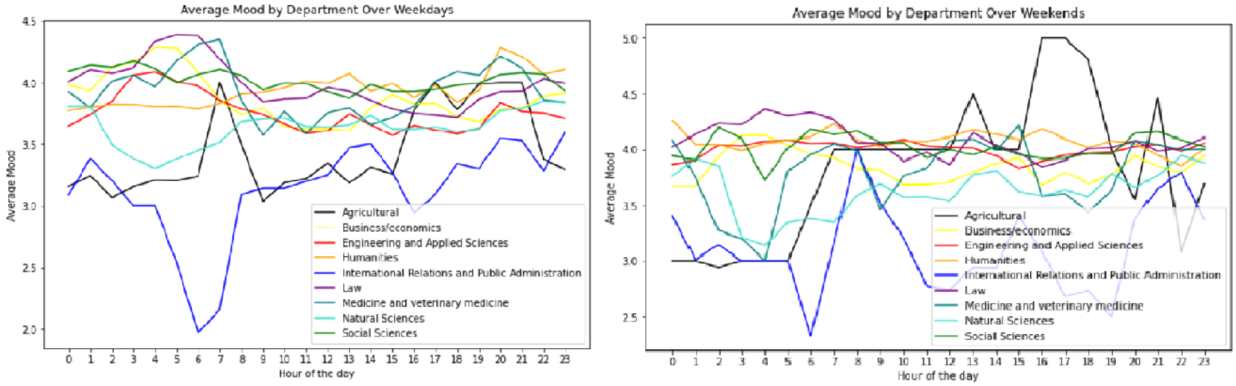


Fig 10

I moved on to observe how activities are varying over a time diary. As explained in figure 11, extroverts tend to be the happiest in socializing and partying events and as they go into solitude their mood goes down as well. This behavior supports their personality traits and hence are quite obvious. On the contrary, introverts tend to be invested in more solo and creative endeavors which is also observed in the plot in figure 12.

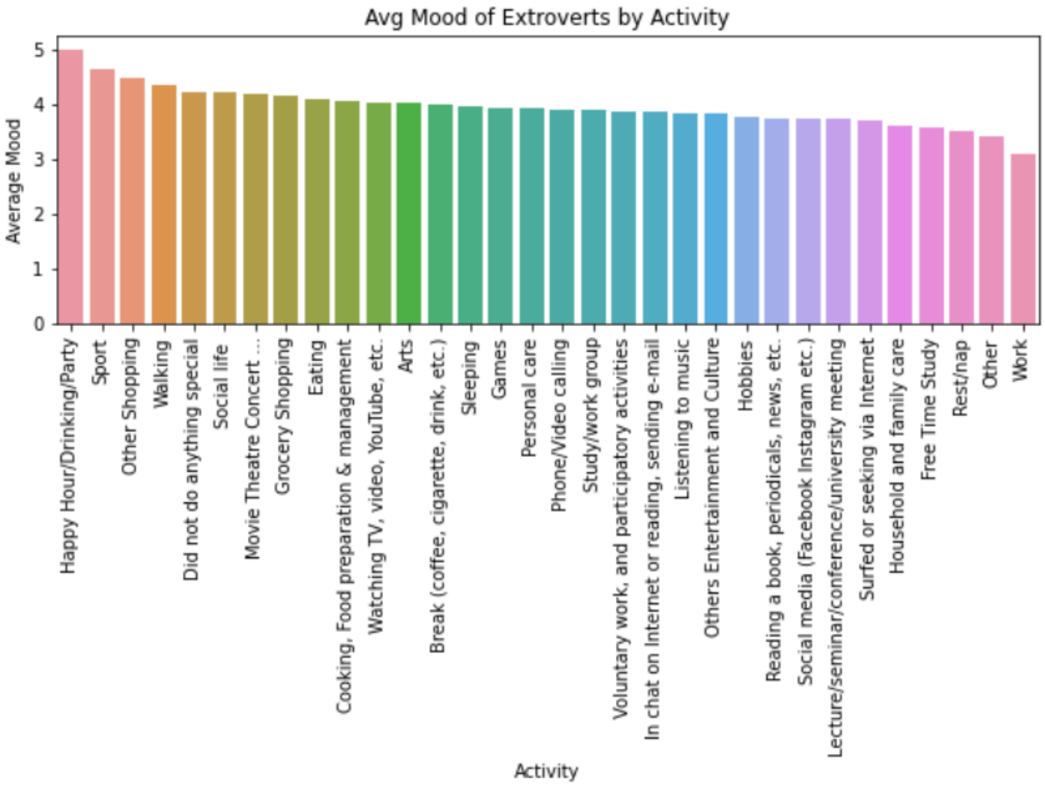

Avg Mood of Extroverts by Activity
Average Mood
0
1
2
3
4
5
Happy Hour/Drinking/Party
Sport
Other Shopping
Walking
Did not do anything special
Social life
Movie Theatre Concert ...
Grocery Shopping
Eating
Cooking, Food preparation & management
Watching TV, video, YouTube, etc.
Arts
Break (coffee, cigarette, drink, etc.)
Sleeping
Games
Personal care
Phone/Video calling
Study/work group
Voluntary work, and participatory activities
In chat on Internet or reading, sending e-mail
Listening to music
Others Entertainment and Culture
Hobbies
Reading a book, periodicals, news, etc.
Social media (Facebook Instagram etc.)
Lecture/seminar/conference/university meeting
Surfed or seeking via Internet
Household and family care
Free Time Study
Rest/nap
Other
Work
Activity

Fig 11

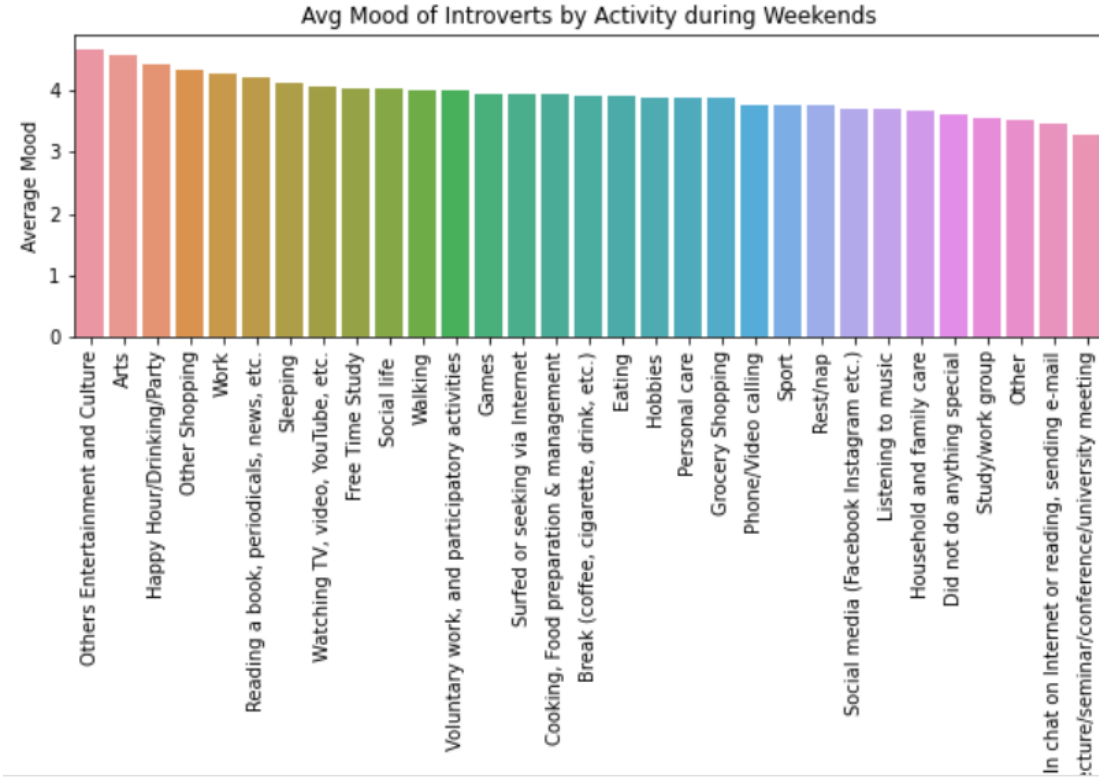

Avg Mood of Introverts by Activity during Weekends
Average Mood
0
1
2
3
4
Others Entertainment and Culture
Arts
Happy Hour/Drinking/Party
Other Shopping
Work
Reading a book, periodicals, news, etc.
Sleeping
Watching TV, video, YouTube, etc.
Free Time Study
Social life
Walking
Voluntary work, and participatory activities
Games
Surfed or seeking via Internet
Cooking, Food preparation & management
Break (coffee, cigarette, drink, etc.)
Eating
Hobbies
Personal care
Grocery Shopping
Phone/Video calling
Sport
Rest/nap
Social media (Facebook Instagram etc.)
Listening to music
Household and family care
Did not do anything special
Study/work group
Other
In chat on Internet or reading, sending e-mail
cture/seminar/conference/university meeting

Fig 12

Another fascinating observation is found with how people feel when they are with somebody. Based on my Exploratory Analysis, the average mood of female participants remained high when they were with their partners, but the average mood of male participants was comparatively lower when they stayed with their partners This behavior is observed by the line plots in figure 13.

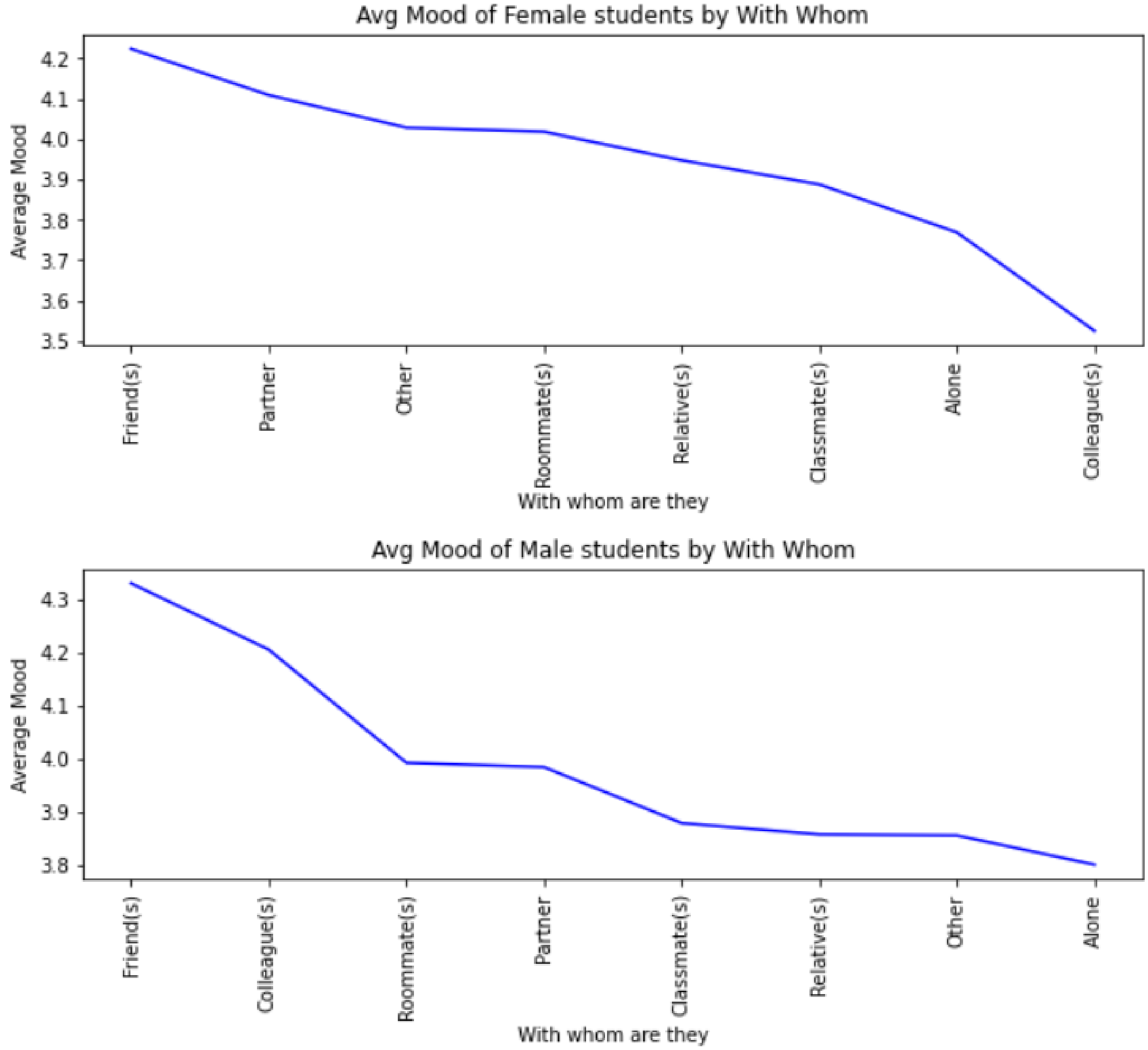
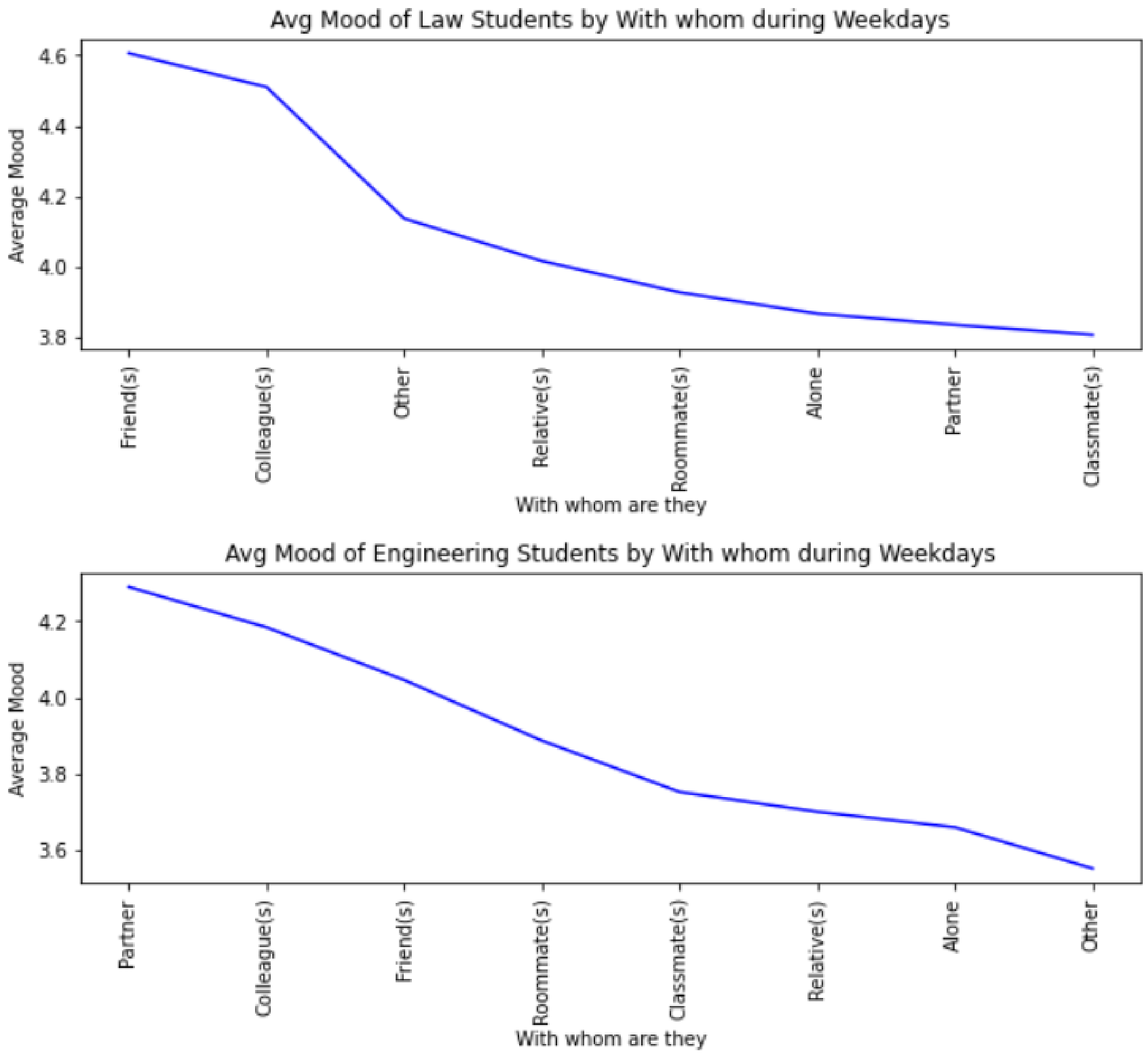


Fig 13

In case of mood across different departments, students from most departments stay quite happy when they are with their partners, whereas, for Law students that is not the case. They are rather less happy when they are around their partners. Figure 14 gives us this fascinating observation.


Avg Mood of Law Students by With whom during Weekdays

Avg Mood of Engineering Students by With whom during Weekdays

Fig 14

Many such analyses were conducted and can be found in the code listed under the appendix section below. I only listed a few interesting observations made.

4. Hypothesis Testing

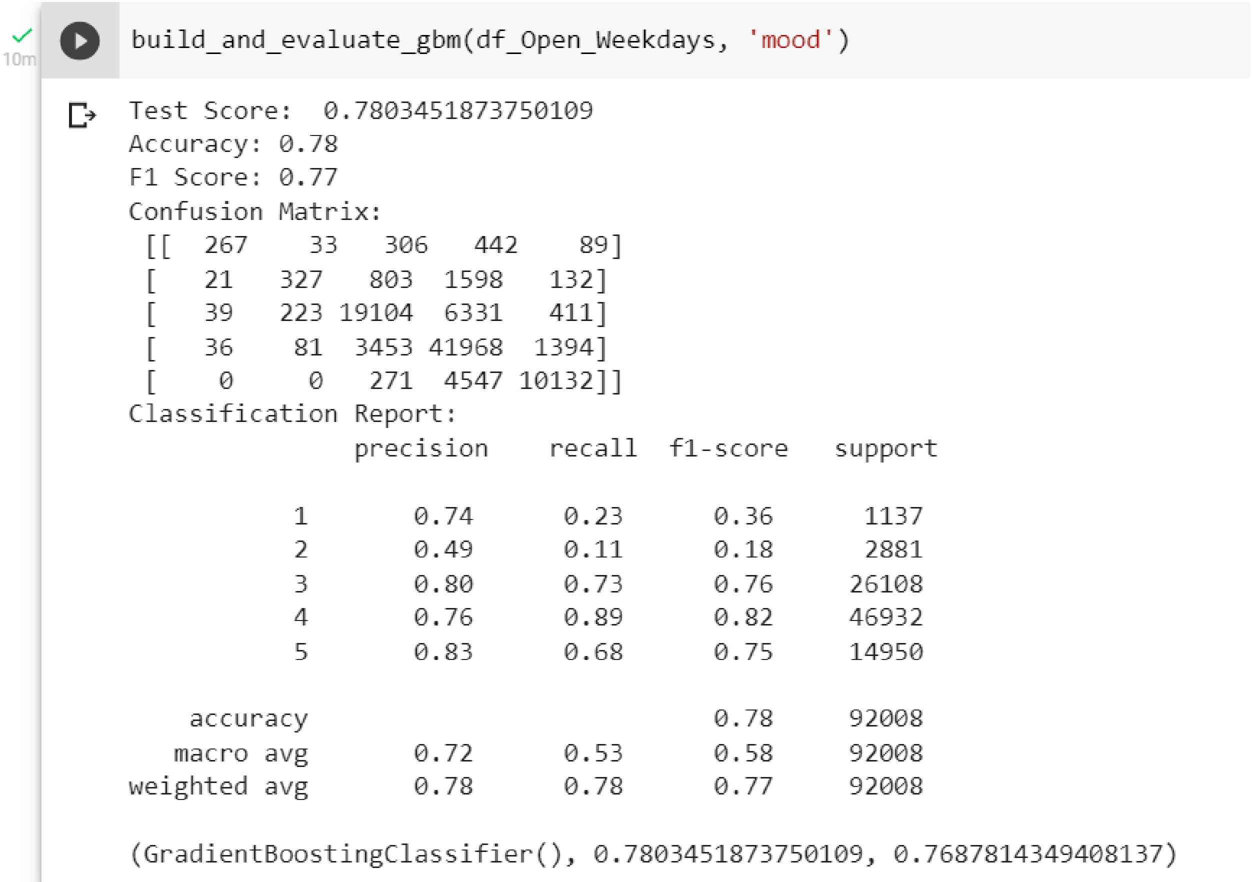
The purpose of the study was to investigate the associations between mood and various contextual variables, including day of the week (weekday vs. weekend) and personality characteristics. The t-test and one-way ANOVA were used in a hypothesis test to accomplish this.The first hypothesis test was performed to determine if there is a significant difference in mood scores between weekdays and weekends. With a p-value of lesser than 0.05, the results of the t-test showed that there is a significant difference in mood scores between the two day types. This indicates that mood may be influenced by the type of day, with participants reporting higher or lower mood scores on weekdays or weekends, respectively. To ascertain whether there is a significant difference in mood scores between various personalities, the second hypothesis test was conducted. To achieve this, one-way ANOVA was used to group the data according to the Big Five Personality traits. The findings demonstrated that the various personalities' mood scores differed significantly from one another. This suggests that personality traits may affect mood because participants' mood scores varied according to their personality types.

The outcomes of the hypothesis tests indicate that mood is significantly influenced by personality traits as well as day type. These results emphasize the significance of taking these contextual factors into account when analyzing and comprehending mood.

5. Machine Learning

After the hypothesis tests, the next logical step was to apply machine learning models to be able to predict participants' mood using the factors we have taken into account. For this I first split the entire data into a train set and a test set. The conventional 80:20 ratio is followed here. There were many categorical features in our data. These needed to be encoded to be fed to the machine learning algorithms. As all our categorical features such as "Locality", "withw", "region", "department" etc are nominal values, I had to go with the One-Hot encoding method. The `"get_dummies()"` method from python pandas library was used to achieve that. This resulted in a wide, sparse matrix.

Gradient Boosting was applied and we got an accuracy of 78%. The figure 15 shows the output from the gradient boosting algorithm.

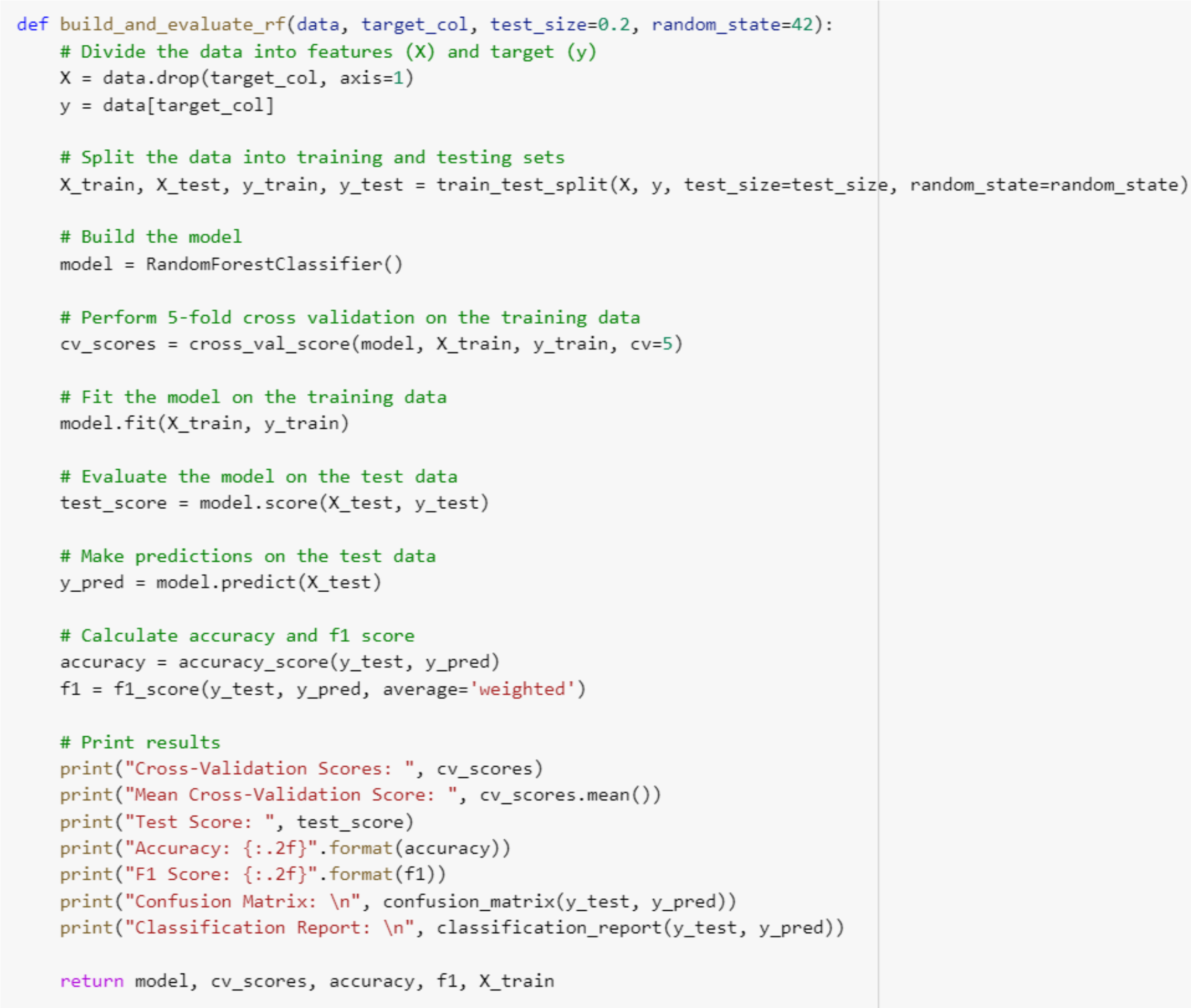


```
build_and_evaluate_gbm(df_Open_Weekdays, 'mood')
```

```
Test Score:  0.7803451873750109
Accuracy: 0.78
F1 Score: 0.77
Confusion Matrix:
 [[  267    33   306   442    89]
 [   21   327   803  1598   132]
 [   39   223 19104  6331   411]
 [   36    81  3453 41968  1394]
 [    0     0   271  4547 10132]]
Classification Report:
               precision    recall  f1-score   support

           1       0.74      0.23      0.36      1137
           2       0.49      0.11      0.18      2881
           3       0.80      0.73      0.76     26108
           4       0.76      0.89      0.82     46932
           5       0.83      0.68      0.75     14950

    accuracy                           0.78     92008
   macro avg       0.72      0.53      0.58     92008
weighted avg       0.78      0.78      0.77     92008

(GradientBoostingClassifier(), 0.7803451873750109, 0.7687814349408137)
```

Fig 15

When compared with the next model i.e. the Random Forest Classifier, Gradient Boosting wasn't giving comparatively better results and hence i decided to stick to Random Forest henceforth. To go a bit granular into the data, I ran the model for different subset of samples, with each subset data containing students pertaining to one dominant personality category. In other words, I built 10 different models with 5 models for weekdays and 5 models for weekends. This division I made, considering the fact that predictors which exist during weekdays may not be present during weekends. Hence with different predictors we could also observe different patterns of mood variations as explained in the Exploratory Analysis segment. Within weekdays or weekends also I decided to go with 5 separate models for 5 types of personalities, one model for people high on Extraversion, one model for Neurotic people and so on. The reason behind such separation is also similar. Since the mood variation was quite different for different personalities, I wanted to see if the predictors play the same role for each of these 5 cases or do they vary. We implemented Feature Importance at the end in order to understand how predictors change according to the type of personality people have. This is explained in detail in the evaluation part of the report. Coming back to the random forest models, in order to optimize the models, I implemented 5 fold cross validation. A method named, `"build_and_evaluate_rf()"` was written which would take dataframe and outcome variable as input parameters and perform cross validation, modeling and return the f1 score, Cross validation Score, Accuracy and Confusion Matrix as output. Figure 16 shows one example of such a model.



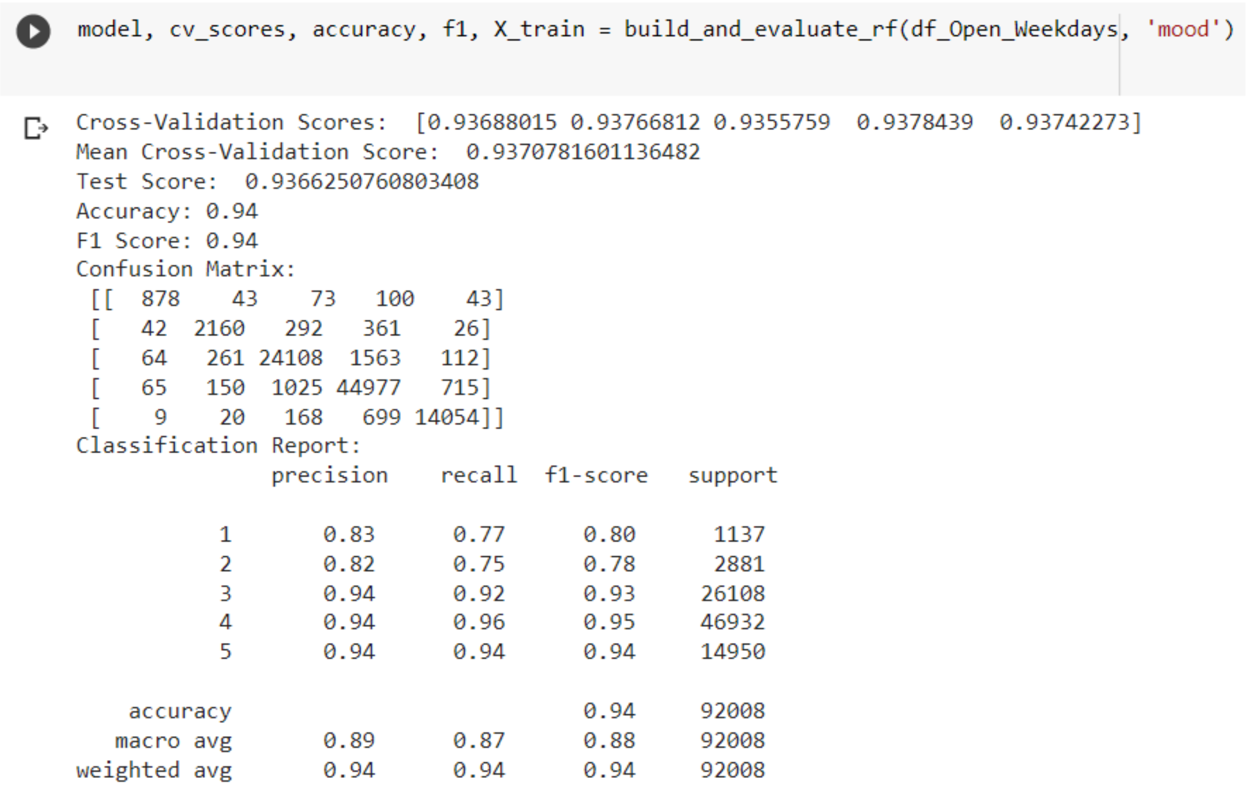
```
def build_and_evaluate_rf(data, target_col, test_size=0.2, random_state=42):
    # Divide the data into features (X) and target (y)
    X = data.drop(target_col, axis=1)
    y = data[target_col]

    # Split the data into training and testing sets
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size, random_state=random_state)

    # Build the model
    model = RandomForestClassifier()

    # Perform 5-fold cross validation on the training data
    cv_scores = cross_val_score(model, X_train, y_train, cv=5)

    # Fit the model on the training data
    model.fit(X_train, y_train)

    # Evaluate the model on the test data
    test_score = model.score(X_test, y_test)

    # Make predictions on the test data
    y_pred = model.predict(X_test)

    # Calculate accuracy and f1 score
    accuracy = accuracy_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred, average='weighted')

    # Print results
    print("Cross-Validation Scores: ", cv_scores)
    print("Mean Cross-Validation Score: ", cv_scores.mean())
    print("Test Score: ", test_score)
    print("Accuracy: {:.2f}".format(accuracy))
    print("F1 Score: {:.2f}".format(f1))
    print("Confusion Matrix: \n", confusion_matrix(y_test, y_pred))
    print("Classification Report: \n", classification_report(y_test, y_pred))

    return model, cv_scores, accuracy, f1, X_train
```

Fig 16

This function is then run over all the subsets of data pertaining to each day_type(weekday or weekend) as well as dominant personality traits. Table 1 shows the accuracy comparisons between all these models for reference.

Personality	day_type	Accuracy Scores
Open People	weekdays	0.94
	weekends	0.94
Extrovert People	weekdays	0.91
	weekends	0.94
Neurotic People	weekdays	0.91
	weekends	0.91
Conscientious People	weekdays	0.94
	weekends	0.93
Agreeable People	weekdays	0.93
	weekends	0.93

Table 1



```
model, cv_scores, accuracy, f1, X_train = build_and_evaluate_rf(df_Open_Weekdays, 'mood')
```

```
Cross-Validation Scores:  [0.93688015 0.93766812 0.9355759  0.9378439  0.93742273]
Mean Cross-Validation Score:  0.9370781601136482
Test Score:  0.9366250760803408
Accuracy: 0.94
F1 Score: 0.94
Confusion Matrix:
 [[  878    43    73   100    43]
 [   42  2160   292   361    26]
 [   64   261 24108  1563   112]
 [   65   150  1025 44977   715]
 [    9    20   168   699 14054]]
Classification Report:
               precision    recall  f1-score   support

           1       0.83      0.77      0.80      1137
           2       0.82      0.75      0.78      2881
           3       0.94      0.92      0.93     26108
           4       0.94      0.96      0.95     46932
           5       0.94      0.94      0.94     14950

    accuracy                           0.94     92008
   macro avg       0.89      0.87      0.88     92008
weighted avg       0.94      0.94      0.94     92008
```

Fig 17

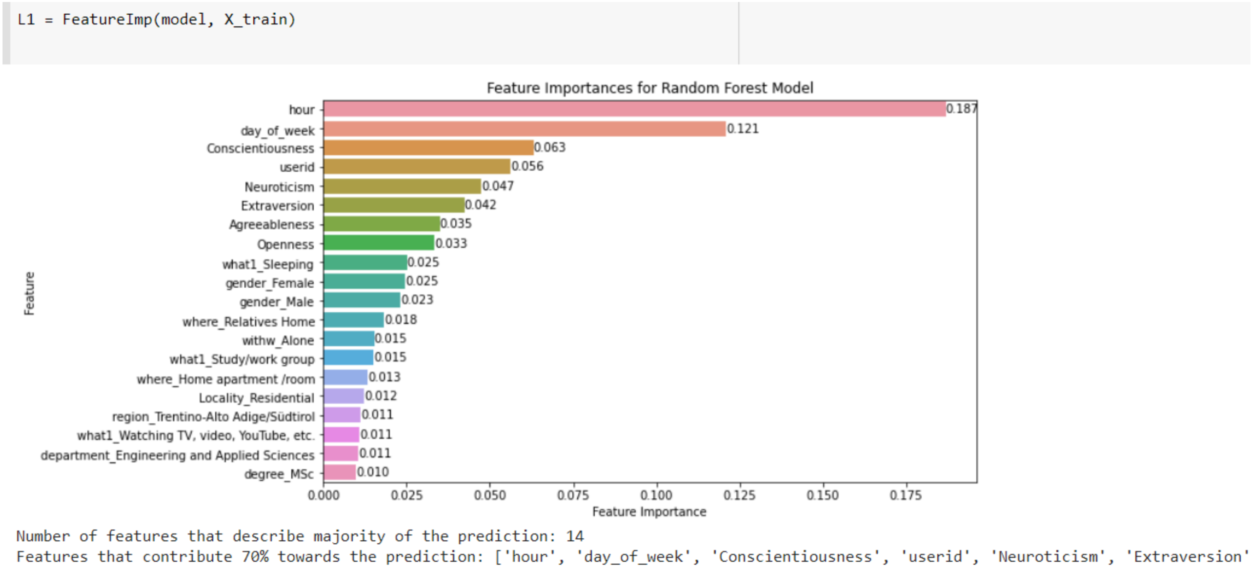
Figure 17 shows the output from a random forest model executed on a subset of open people during weekdays.

As we found out, the random forest model could predict the mood of people of different personalities with good levels of accuracy. Further, we must study whether the predictors which contribute the most towards the outcome variable are same or different for every group. In order to do this, we have to understand the Feature Importance.

6. Evaluation

In python, the model *"RandomForestClassifer"* has an attribute called, *"feature_importance_"* using which I extracted the most important features for each model. In order to achieve this I wrote this function "FeatureImp()". With this we can not only find the top (70%) contributing predictors but also plot them in a barplot for visualization and better understanding. Here is the function and figure 18 shows the output when this function is applied on the subgroup of open people over weekdays.

```
def FeatureImp(model, X_train):
    # Extract feature importances
    importances = model.feature_importances_

    # Plot the feature importances
    feature_importances = pd.DataFrame(
        importances, index=X_train.columns, columns=["importance"]
    ).sort_values("importance", ascending=False)

    # Calculate cumulative sum of feature importances
    cumulative_sum = np.cumsum(feature_importances["importance"])

    # Determine the number of features that contribute 70% towards the prediction
    num_features = np.argmax(cumulative_sum >= 0.7) + 1

    # Plot the feature importances with annotations
    plt.figure(figsize=(10, 6))
    sns.barplot(
        x=feature_importances["importance"][0:20],
y=feature_importances.index[0:20]
    )
    plt.xlabel("Feature Importance")
    plt.ylabel("Feature")
    plt.title("Feature Importances for Random Forest Model")

    for i in range(20):
        plt.text(
            x=feature_importances["importance"][i],
            y=i,
            s="{:.3f}".format(feature_importances["importance"][i]),
            ha="left",
            va="center",
        )

    plt.show()

    L1 = feature_importances.index[:num_features].tolist()
    print(
        f"Number of features that describe majority of the prediction:
{num_features}"
    )
    print(
        f"Features that contribute 70% towards the prediction:
{feature_importances.index[:num_features].tolist()}"
    )
    return L1


L1 = FeatureImp(model, X_train)
```

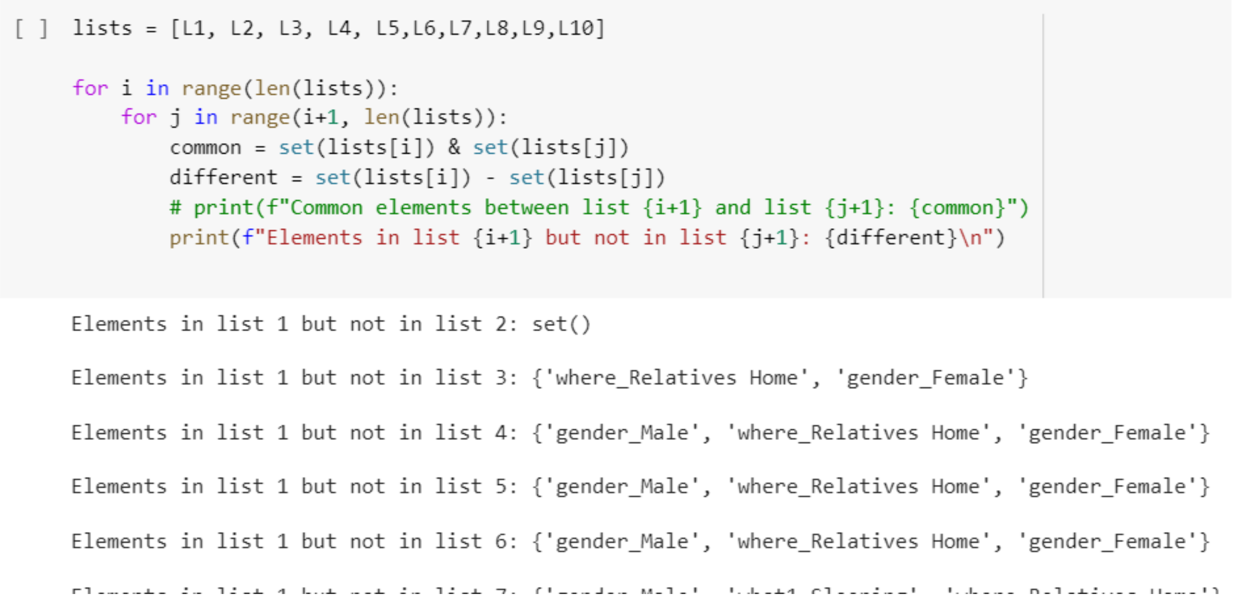


```
L1 = FeatureImp(model, X_train)
```

```
Number of features that describe majority of the prediction: 14
Features that contribute 70% towards the prediction: ['hour', 'day_of_week', 'Conscientiousness', 'userid', 'Neuroticism', 'Extraversion'
```

Fig 18

Similarly we perform Feature Importance for all the 10 models we have. Then we compare the most dominant features we found. Based on the findings, it was evident that the number of dominant features were varying for each and every model. Though most dominant features remained common across all the models, there were certain features such as Location, Gender, Department which were not common for many of these models. In other words, for a few models these 3 features played an important role whereas for the rest these were not as significant a predictor.



```
[ ]  lists = [L1, L2, L3, L4, L5,L6,L7,L8,L9,L10]

     for i in range(len(lists)):
         for j in range(i+1, len(lists)):
             common = set(lists[i]) & set(lists[j])
             different = set(lists[i]) - set(lists[j])
             # print(f"Common elements between list {i+1} and list {j+1}: {common}")
             print(f"Elements in list {i+1} but not in list {j+1}: {different}\n")
```

```
Elements in list 1 but not in list 2: set()

Elements in list 1 but not in list 3: {'where_Relatives Home', 'gender_Female'}

Elements in list 1 but not in list 4: {'gender_Male', 'where_Relatives Home', 'gender_Female'}

Elements in list 1 but not in list 5: {'gender_Male', 'where_Relatives Home', 'gender_Female'}

Elements in list 1 but not in list 6: {'gender_Male', 'where_Relatives Home', 'gender_Female'}

Elements in list 1 but not in list 7: {'gender_Male', 'what1_Sleeping', 'where_Relatives Home'}
```

7. Future Works

Though this analysis gave us an understanding of how different predictors play a role in the mood of individuals in different circumstances, I believe this project suffers from some inherent assumptions and drawbacks which can be addressed in a future extension to this work. The random forest model was developed with 5 fold cross validation, partly because with this huge dataset, iterating over it several times was extremely time consuming. This prevented me from usings methods such as grid search cross validation. Grid Search Cross Validation offers flexibility to try the model with different combinations of hyperparameters, and then gives us the optimum set of hyperparameters which would get us the best model results. Even trying with different numbers of folds was time and resource intensive and hence I stuck to the conventional 5 folds cross validation. With better hardware and more time allowed for the model to execute, the model accuracy could have been improved even further. Other bagging and boosting techniques could also have been implemented in order to compare our random forest model with and observe if any other model with proper tuning of hyperparameters could get us a better result.

Our project works with an inherent assumption of independence. It assumes that all the observations in our dataframe are independent of each other. However in real life, this is not the case. We have multiple observations for each respondent and respondents were also coming from specific groups and sub-groups such as various departments, cohorts and degrees. This respondent heterogeneity was not taken into account. As a part of future works, a multi-level model or mixed effect model can be implemented for a more robust and practical demonstration of its utility.

A user study can also be conducted to understand the perception and feedback of new users on the model's performance and usability.

8. Conclusion

The project aimed to understand the relationship between mood and various predictors in the students of University of Trento. Through Exploratory Data Analysis, several insights were gained on the impact of the environment on mood. For example one insight was that, the law students exhibit relatively lower levels of happiness when they are with their partners. Time of the day, specified as hour, day of the week and level of conscientiousness were found to be some of the most important factors across all groups of students. Additionally the day type was also found to play a significant role in the variation of mood, with distinction made between weekdays and weekends. These findings highlight the importance of considering personality traits as well as contexts in which students experience their mood and provide valuable insights for future research in this area. Use of sensor data has been instrumental in this research as such data have the least chances of error as compared to self reported data. Ultimately, this work has the potential to inform educational policies and strategies aimed at promoting the well-being of university students.

9. Appendix

Link to the original code:
https://colab.research.google.com/drive/1g4WnpW0A96s0LI7QUdruZKv92hPO1RvD?usp=sharing

10. Reference

1. Mattia Zeni, Ivano Bison, Britta Gauckler, Fernando Reis, and Fausto Giunchiglia. "Improving time use measurement with personal big data collection - the experience of the European Big Data Hackathon 2019." *Journal of Official Statistics*, 2020.
2. Zeni M, Zhang W, Bignotti E, et al. *Fixing mislabeling by human annotators leveraging conflict resolution and prior knowledge*[J]. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 2019, 3(1): 1-23.
3. Maddalena E, Ibáñez LD, Simperl E, Gomer R, Zeni M, Song D, Giunchiglia F. *Hybrid Human Machine workflows for mobility management*. Companion Proceedings of The 2019 World Wide Web Conference, 2019.

4. Giunchiglia, F; Zeni, Mattia; Gobbi, Elisa; Bignotti, Enrico; Bison, Ivano. *Mobile social media usage and academic performance*. Computers in Human Behavior, vol. 82, p. 177-185, 2018.
5. Giunchiglia F, Zeni M, Big E. *Personal context recognition via reliable human-machine collaboration*[C]//2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE, 2018: 379-384.
6. Giunchiglia F, Bignotti E, Zeni M. *Personal context modelling and annotation* 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE, 2017: 117-122.
7. Zeni M, Zaihrayeu I, Giunchiglia F. *Multi-device activity logging* Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication. 2014: 299-302.
8. Kim P H, Giunchiglia F. *The open platform for personal lifelogging: the elifelog architecture*[M]//CHI'13 Extended Abstracts on Human Factors in Computing Systems. 2013: 1677-1682.