TILBURG UNIVERSITY

Understanding Society

Predicting Mood from Smartphone Application Usage

Examining differences among algorithms and blocks of measurements

---

Maartje (M.E.) Verhoeven
STUDENT NUMBER: 1273860

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY
DEPARTMENT OF COGNITIVE SCIENCE & ARTIFICIAL INTELLIGENCE
SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
TILBURG UNIVERSITY

Thesis committee:
Dr. A.T. Hendrickson
Dr. A. Alishahi

Tilburg University School of Humanities and Digital Sciences
Department of Cognitive Science & Artificial Intelligence
Tilburg, The Netherlands
May 2020

Preface

Over the last few months, I wrote this research paper in an attempt to provide further insights on the relationship between smartphone usage and mood. I have in the last years seen many friends and colleagues struggle with their energy and mental health. I strongly believe that smartphone usage is, at least to some extent, linked to these problems. I am grateful for having the opportunity to further investigate this topic for the fulfilment of my master thesis.

I would like to thank my supervisor and first reader Dr. Drew Hendrickson for his guidance during this process. I also would like to express my appreciation to Dr. Afra Alishahi for providing a second opinion on this thesis and to my friends and family for their support. My dear friend Caitlin van Mil deserves a particular note of thanks for her support and feedback. I believe to have strongly benefitted from debating issues with her.

Abstract

Insight into how smartphone usage affects mood is important in order to enable people to use their smartphones in a manner that improves rather than deteriorates their wellbeing as an increasing amount of people is struggling with their smartphone usage. This study investigates the extent to which smartphone application usage can predict mood among blocks of measurement in panel studies. Panel conditioning and panel attrition have been widely discussed in the social sciences to affect the quality of the results but this has, to our knowledge, never been taken into account for predictive models in the field of data science. Data from a population of 124 first year Psychology students at Tilburg University, measured in period of 34 days, were used to train and tune several learning algorithms and compare models from different blocks of measurements. Results indicate the Random Forest (RF) classifier to best predict mood from application usage and the model containing data from the first half of the study to score highest in comparison to the other defined models. However, the achieved accuracy scores were only slightly above the baseline and the predictive performance is therefore considered to be low. It is recommended for future research to use more frequent mood measurements as it was not possible to capture the experienced mood at the moment that the smartphone was used with the limited measurements from this study.

Keywords: smartphone application usage, mood prediction, panel studies, predictive modelling, multiclassification model

Contents

1. Introduction

1.1 Context

Smartphones have become very popular in a relatively short amount of time. It was only 10 years ago that smartphones were seen as pure luxury. Nowadays, the devices are practically needed as they have been fully integrated in society. 92% of Dutch citizens used smartphones in 2019 and an average of 61 hours per month is spent on these mobile devices (CBS, 2020; SIDN, 2018). Research shows how people use smartphones in manners that suit their individual needs. The devices have been blended into people's life-style (Barkhuus & Polichar, 2010). However, 79% of Dutch people has made deliberate attempts to go offline. Internet-free vacations and internet-free restaurants have gained popularity as they help people to unplug from the increasing demands of smartphones (SIDN, 2018). The popularity of smartphones on the one hand and the attempts to go offline on the other hand suggests that the current smartphone usage is not perceived as satisfying. Previous research indicated that different applications have different effects on people's mood, suggesting mood differs depending on the applications people use. For example, time spent on Email has been found to positively relate to stress (Kushlev & Dunn, 2015) whereas playing games has been found to reduce stress (Reinecke, 2009). The current smartphone usage can become more satisfying when people are aware of how application usage affects their mood. Additional information on application usage and mood is needed in order to enable people to use their smartphones in a manner that improves rather than deteriorates their wellbeing.

The current study investigates the predictability of mood from application usage, using a topical, big, real-life dataset, in order to obtain a better insight into the relationship between application usage and mood. Several studies in the area of data science have investigated the relationship between application usage and mood. Amongst others, the relationship between mobile application usage and satisfaction with life (Linnhoff & Smith, 2017), the predictability of negative emotions from smartphone usage (Hung, Yang, Chang, Chang & Chen, 2016) and the predictability of mood from phone usage (LiKamWA et al., 2013) have been investigated. However, contradicting relationships have been found and researchers stress how the relationship between application usage and wellbeing is still unclear (Alibasa, Calvo & Yacef, 2019). In addition, previous work mostly used relatively small datasets and recommended further investigation with larger datasets in order to ensure validity of the results (LiKamWA et al., 2013).

Furthermore, the current study focuses on differences in predictive performance among models from different blocks of measurement in the study. The time of measurement in panel studies is expected to affect the results of the measurements. This is amongst others grounded in the effects of previously taken measures, social desirability and participants dropping out during longitudinal studies (Lugtig, 2014; Warren & Halpern-Manners, 2012). These biases in panel studies have been widely discussed in social sciences

but have, to our knowledge, never been taken into account for predictive modelling in the field of Data science.

1.2 Problem statement and research questions

Data on smartphone application usage and mood experiences from 124 psychology students at Tilburg University is used to answer the problem statement of this study: '*To what extent can smartphone application usage predict mood among blocks of measurement in panel studies?*'. The problem statement is divided into two research questions. The research questions are stated as follows:

1. *What learning algorithm best predicts mood from smartphone application usage?*
2. *Are there any differences in predictive performance between the different blocks of measurement when predicting mood from smartphone application usage in a panel study?*

Participants' application usage and mood, constructed according to the quartiles of the Circumplex model of affect (Russel, 1980), are measured in the morning, evening and afternoon. The data is split into a training set and a test set and the k-Nearest Neighbor, Support Vector Machine, Random Forest and Logistic Regression classifier are trained and tuned in order to answer the first research question. The accuracy and macro-F1 scores are used for model comparison and evaluation.

For the second research question, differences in predictive performance for several blocks of measurements are investigated. Five different models are used, corresponding to all measurements, measurements from the first half of the study, measurements from the last half of the study, measurements from the first three days of the study and measurements from the last three days of the study. The best performing learning algorithm from research question 1 is trained and tuned again for each model. Accuracy scores and macro-F1 scores will be examined and used for model comparison.

1.3 Thesis outline

First, section 2 of this research paper describes related work that is relevant for the topics and research design of the current study. Sections 3 and 4 elaborate on the methods, the datasets and the experimental procedure. Then, the results are explained and presented in tables and figures in section 5. The discussion in section 6 further elaborates these results in relation to previous research and mentions drawbacks and suggestions for future research. Finally, the conclusion in section 7 provides answers to the research questions.

2. Related work

This section elaborates on relevant scientific literature. Previous work, theories and the way in which the current research builds further on this will be elaborated on below. First, literature regarding the predictability of (aspects of) mood from smartphone usage is discussed. Second, influences from different blocks of measurements in panel studies on quality of the study are examined.

2.1 Predicting mood from phone usage

Several studies have investigated the relationship between (aspects of) mood and smartphone usage. To start with, Alibasa, Calvo and Yacef (2019) used sequence pattern mining to extract features which would best predict mood from behavioural patterns on smartphones (Alibasa et al., 2019). The researchers stress how the relationship between the usage of different digital technologies and wellbeing is still unclear and how previous studies' findings are contradicting (Alibasa et al, 2019). For example, games have been found to reduce stress (Reinecke, 2009), while frequent use of Email has been found to increase stress (Kushlev & Dunn, 2015). Alibasa, Calvo and Yacef (2019) gathered data from 72 participants for their study. Sequences were generated based on buckets of activities, which were included in mood arrays. These arrays were used as input for generalized sequential pattern (GSP) algorithms. The most frequent patterns found by the GSP results were used as features for mood detection. Results indicate this method to be useful for predicting mood from digital technology usage. An accuracy of 80% was achieved, which was above the defined baseline. Results amongst others indicated networking and games to be correlated with more positive mood reports, while the search category was correlated with more negative mood reports. Furthermore, the researchers used application categories for analysis instead of application names as categories would result in less granular data. For example, *Gmail* and *Outlook* were transformed to the category *Email*.

Similarly, Ferdous, Osmani and Mayora (2015) redesigned applications into broader categories in order to obtain a more generic level understanding of the applications used (Ferdous et al., 2015). The researchers divided the diverse applications in their dataset into 5 main application categories: *entertainment, utility, social networking*, *games* and *browser*. For example, the *utility* category amongst others included *applications calendar, map, clock, weather* and *calculator applications.* Each of these application categories were used as one of the input features for investigating the relationship between patterns in application usage and stress in the workplace (Ferdous, et al., 2015). Data was collected from 28 participants over a 6-week period, including monitored smartphone usage and daily questionnaires measuring experienced stress. A Support Vector Machine classifier was trained in order to develop a user-centric model of application usage. Perceived stress reports and individual application usage behaviour were implemented as ground truth for the classifier. The researchers achieved an average accuracy of 75% which was above

the majority baseline. Results suggest self-reported stress levels and application usage patterns to be highly correlated (Ferdous et al., 2015).

Another study compared various classifiers for predicting happiness from phone usage (Bogomolov, Lepri & Pianesi, 2013). Phone usage data was used as input for the learning algorithms and included features from SMS and call logs, Bluetooth hits, amount and diversity of calls and SMS. Individual happiness was measured with three classes and used as output. The classification task was performed by a Neural Network, Support Vector Machine and Random Forest. The Random Forest classifier achieved the highest performance with an accuracy of 80%. Based on the presented confusion matrix, the majority baseline was approximately 60%. The random forest classifier was thus able to outperform the majority baseline by roughly 20% as it calculated the average decrease in Gini index. The SVM only did a good job predicting the majority class. The results indicate that the Random Forest classifier can quite accurately predict individual happiness from smartphone usage data (Bogomolov et al., 2013).

The study by Preotiuc-Pietro, Schartz Park, Eichstaedt, Kern, Ungar and Schulman (2016) used the Circumplex model of affect (Russel, 1980) to model sentiment for the classification task of predicting sentiment from Facebook posts (Preotiuc-Pietro, Schartz Park, Eichstaedt, Kern, Ungar & Schulman, 2016). The Circumplex model is a well-established system for describing emotional states, which assumes any affective experience to be a linear combination of two independent values for valence and arousal (Russel, 1980). The Circumplex model has been widely validated and used in scientific studies (LiKamWa et al., 2013; Preotiuc-Pietr, 2016). The model can be described in the dimensions of pleasure and activeness. Pleasure refers to the extent of positive or negative feelings. Activeness refers to the likeliness to take an action (LiKamWa et al., 2013). The linear combination between these two values is interpreted as one value for the state of affect. The study measured sentiment from ratings on two separate nine-point ordinal sales, representing valence and arousal, which were placed on the Circumplex model. The researchers trained a bag-of-words linear regression model on the data to predict sentiment ratings for new Facebook posts. The final model achieved higher correlations with ratings on the Circumplex model in comparison to other studies in which both dimensions were predicted with standard sentiment analysis lexicons (Pretiuc-Pietro et al., 2016). Other studies have tested the theoretical consideration on which the Circumplex model is build, using principal-component analyses (PCA). PCA is used to convert samples to lower dimensional spaces by linearly transforming the data into a new coordination system. Wooyeon (2020) reviewed the reasonableness of classification by the Circumplex model. The correlations from the PCA among different mood variables were found to be similar to the suggested correlations in the Circumplex model (Wooyeon, 2020). Similarly, results from a study by Pukrop (2000) indicate that PCA strongly confirms the Circumplex model. The theoretical considerations on which the Circumplex model was build are thus confirmed by PCA (Pukrop, 2000).

To summarize, several studies achieved good performance when predicting (aspects of) mood from phone usage. Dividing applications into categories and transforming mood values into ratings on the Circumplex model are expected to yield better predictive performance. Even though various learning algorithms were able to achieve good results when predicting (aspects of) mood from smartphone usage, researchers admit the outcomes to rely on human judgement and design of the analysis (Alibasa et al., 2019). In addition to this, previous studies have mainly included small and homogeneous groups of participants, thereby highlighting a need for research on the relationship between smartphone usage and mood with larger and more heterogeneous samples is recommended (LiKamWa et al., 2013; Alibasa et al., 2019). The current study builds on this by further investigating the predictability of mood from smartphone application usage, taking into account various learning algorithms and using a large, topical dataset.

2.2 Predictive performance among blocks of measurements

The current study used data from a panel study in which the same participants were repeatedly measured over a period of time. Panel studies in the social sciences have been widely discussed to bias results as the responses can be influenced by several factors (Warren & Halpern-Manners, 2012). To begin with, the phenomenon of panel conditional suggests responses to measurements to be affected by the previously taken measurements (Halpern-Manners et al., 2012). Several studies have investigated this phenomenon. To illustrate, previous research shows how attitudes differ among participants who were asked for their attitudes multiple times and those in a control group who were only asked once (Wilson & Kraft, 1993). Waterton and Lievesley (1989) argued that conditioning is grounded in participants becoming more honest while others suggested the differences in repeated measurements to be grounded in the multiple moments of reflection when filling in questionnaires (Wilson & Kraft, 1993). A third argument for panel conditioning is that participants pay less attention to follow-up measurements as they remember fairly well what was asked during previous measurements. This particularly is the case when the measurements take place relatively frequently (Wilson & Kraft, 1993). Overall, it is suggested that participants show less socially desirable behaviour but pay less attention to the measurements as the measurements continue (Warren & Halpern-Manners, 2012). This would imply the following two implications for the current study. First, measurements from later in the study are suggested to correspond better to reality as the participants showed less socially desirable behaviour. For example, participants might not have opened certain applications or might have spent less time on their phone in the beginning of the study depending on their beliefs about appropriate smartphone usage (Warren & Halpern-Manners, 2012). Second, the measurements might become more biased later in the study as the participants might pay less attention to the measurements. For example, participants might not read the mood questionnaires as thoroughly as in the beginning of the study, because they remember the questions fairly well.

Another factor that influences the quality of the panel study is the phenomenon of attrition. Attrition refers to participants dropping out during the study (Lugtig, 2014). Damen et al. (2015) suggest that features differ between the participants that drop out during the study and the participants that continue to participate in the study. The researchers suggest that dropout would therefore pollute the measurements (Damen et al., 2015). This implies that the quality of the panel study differs among various blocks of measurements in the study.

To summarize, panel conditioning and panel attrition have in the social sciences been stated to affect the quality of the results. These effects of panel studies on the quality of the study have, to our knowledge, never been taken into account for the predictive models in the field of data science. The current work is the first to investigate differences in predictive performance among various blocks of measurements from the panel study.

3. Methods

This section elaborates on the methods used in this study to build and analyze the predictive models. Further description on the datasets and the procedure is provided in the experimental setup section.

3.1 Programming in R

Exploring, cleaning, preprocessing and feature engineering was executed in R (version 3.4.1.). The used packages are displayed in Table 1. The initial separate datasets were combined, variables of interests were created and the data were split according to the defined blocks of measurements. Finally, the data was used for model building and analyses in Python as described in the following section.

Table 1

Packages used in R

Package	Source
Dplyr (version 0.8.3)	*Wickham & Francois, 2017*
Data.table (version 1.12.6)	*Dowle & Srinivasan, 2017*
Lubridate (version 1.7.4)	*Spinu, Grolemund & Wickham, 2017*
Ggplot2 (version 3.2.1)	*Wickham & Winston, 2009*
Tidyverse (version 1.3.0)	*Wickham & Hadley, 2017*
Chron (version 2.3-55)	*James & Hornik, 2010*

3.2 Programming in Python

Models were build and analyzed in Python (version 3.6.9.). The used packages are displayed in Table 2. Input arrays and output arrays were created which were then split into a training set and a test set. Learning algorithms were trained and tuned as further explained in the section on the experimental setup. The best performing learning algorithm, according to the accuracy scores and macro-F1 scores, was trained and tuned again for each of the defined models from research question two.

Table 2

Packages used in Python

Package and version	Source
Scikit-learn (version 0.21.3)	*Pedregosa et al., 2011*
Numpy (version 1.16.5)	*Oliphant, 2006*
Pandas (version 0.25.1)	*McKinney, 2010*

4. Experimental Setup

This section of the paper elaborates on the datasets and the experimental procedure. The experimental setup consists of two major parts: sections 4.1 up to and including 4.4 describe the used data, and sections 4.5 up to and including 4.7 outline the analyses and algorithms. The corresponding codes can be found on GitHub (Appendix A).

4.1 Dataset

Three different datasets were used for this study, containing information on a population of first year Psychology students at Tilburg University. The students voluntarily participated in a research on smartphone usage and mood for a period of 34 days. During this period, the smartphone usage was logged with the MobileDNA application and mood was measured with daily questionnaires. The questionnaires were distributed four times per day. Participants were able to respond to the questionnaires within two hours. However, not all questionnaires were completed as this was not mandatory. The three datasets were received as anonymized Comma Separated Value (CSV) files after a form for Data Protection Rules for Master's Thesis was signed. The first dataset contained information on smartphone usage from the 124 students. This data was gathered in 2019 from February 21 until March 26 , and amongst others included the variables *session*, *start time* and *application*. The second dataset contained information on 1,748 applications, corresponding to 59 different application categories, including the variables *application name* and *application category*. The third dataset contained data on mood from students, measured during three periods in time from June 2018 until May 2019. However, only the data from the third period in time, namely from February 2019 until March 2019, corresponds to the participants from the smartphone usage dataset and can therefore be related to the smartphone usage dataset. Moreover, the mood questionnaires were sent out by error after the phone tracking had finished, so the mood dataset was filtered so that it was aligned with dates in the smartphone usage dataset. The filtered mood dataset contained information from 136 students and amongst others include the variables *stressed*, *cheerful* and *tired*. The three datasets were combined as explained in section 4.3.

4.2 Data cleaning

Data cleaning was done in R (version 3.4.1.), using the packages displayed in Table 1. To start, techniques for cleaning and exploring were applied. Empty values were mutated to not available (NA) values and rows which only consisted of NA-values were removed. The classes from all variables were checked and several variables had to be transformed to integer values like *cheerful* which was seen as character values in R. The

unique labels for the categorical variables were checked. Various variables in the mood dataset were measured on a 5-point Likert scale. The values above 5 were mutated to NA-values for these variables. The total amount of NA-values per variables was checked. The variables *envious, inferior* and *social* contained the most NA-values in the mood dataset with approximately 600 NA-values out of 9,640 observations. The variable *category* had the most NA-values in the smartphone usage dataset with approximately 1,000 NA-values out of the 472,659 observations. The mood dataset was filtered on dates corresponding to the smartphone dataset. Lastly, some plots and graphs were created in order to explore the relationships between the cleaned variables and final check for outlying values.

4.3 Feature engineering

After cleaning, the datasets had to be merged and the variables of interest were created. The application data were merged with the smartphone usage data by application name. Due to the fact that there were inconsistencies in the time of day at which the questionnaires were distributed and the fact that most participants did not fill in all questionnaires, the smartphone usage data and the mood data could not be combined directly and merging the datasets resulted in difficulties. Table 3 displays the counts of completed questionnaires for each hour of the day. Most questionnaires were completed between 9:00 and 22:00, with peaks at 9:00, 10:00, 12:00, 15:00, 16:00, 19:00 and 20:00. In this study, smartphone usage per participant for each part of the day was summarized as peaks in the mood data can quite fairly be distributed among these parts of the day and not too much data would be lost. The new variable *daypart* was constructed for both datasets, including the classes *morning* (6:00 – 12:00), *afternoon* (12:00 – 18:00) and *evening* (18:00 – 00:00). An average for mood was constructed in case the participant completed more than 1 questionnaire per daypart. 2,165 from the 7,487 smartphone usage datapoints had to be removed as they corresponded to times between 00:00 and 6:00. The datasets were merged based on date, daypart and user id. The merged dataset consisted of 5,322 observations. The features were constructed as explained below:

Table 3

Mood measurements per hour

Hour	0	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Count	2	3	1079	1106	287	1620	329	109	986	1036	110	88	982	991	129	17	4

Day_part: Part of the day during which is measured, including the classes *morning*, *afternoon* and *evening*. For the smartphone usage data, this column was derived from the variable *start time* for which the format had to be transformed to Universal Coordinated Time Zone (UTC). Daypart was derived from the

variable *sent time* for the mood data. Datapoints from 6:00 until 12:00 are categorized into morning, datapoints from 12:00 until 18:00 are categorized into afternoon and datapoints from 18:00 until 00:00 are categorized into evening.

Game_count, Util_count, Entert_count, Social_count, Brow_count: The number of times an application of this specific category was opened. Each observation from the smartphone usage dataset was linked to the application dataset which contained the corresponding application category. A total of 44 different application categories were used by the participants with different frequencies. For example *Art & Designed* appeared 9 times and *Communication* appeared 207,861 times in the dataset. The 44 categories were redesigned into a smaller amount of categories in order to obtain a more generic level understanding of the used applications (Ferdous et al., 2015). The categories were transformed into five categories as suggested by Ferdous, Osmani and Mayora (2015): *Entertainment, Utility, Social networking, Games* and *Browser*. Table 4 provides an overview of the redefined categories. The times each participant opened one of these categories was counted for each daypart.

Table 4

Redefined categorisation of applications

New category	Old categories
Entertainment	Auto & Vehicles; Art & Design; Music & Audio; Sports; Books & Reference; Video Players & Editors; Entertainment; Food & Drink; News & Magazines; Health & Fitness; Others.
Utility	Background Process; Maps & Navigation; Finance; House & Home; Medical; Personalization; Photography; Shopping; Tools; Travel & Local; Weather; Word.
Social networking	Communication; Dating; Social; Lifestyle.
Games	Action; Adventure; Board; Arcade; Card; Casino; Casual; Strategy; Trivia; Puzzle; Racing; Racing, Action & Adventure; Simulation.
Browser	Business; Education; Productivity.

Game_dur, Util_dur, Entert_dur, Social_dur, Brow_dur: The total amount of milliseconds spend per application category. The column '*duration*' was constructed with the difference between the time that the application was initiated and the time the application was ended for each observation of smartphone usage. The durations per category were added together for all the applications that the participants used during each daypart.

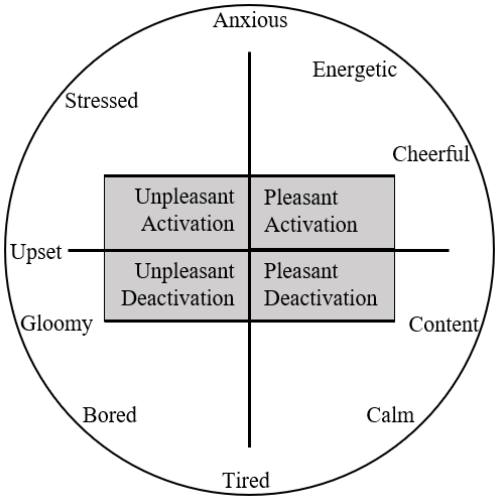
Max_dur: The longest amount of milliseconds spent in one session, derived from the maximum value from the duration column as discussed above.

Min_dur: The shortest amount of milliseconds spent in one session, derived from the minimum value from the duration column as discussed above.

Mood: Mood corresponding to the quartiles on the Circumplex model of affect: *Unpleasant Activation, Pleasant Activation, Pleasant Deactivation* and *Unpleasant Deactivation*. The variables *anxious, energetic, cheerful, content, calm, tired, bored, gloom, upset* and *stressed* from the mood dataset were placed on the Circumplex model of affect (Figure 1) based on the scale coordinates assigned to them in the scientific literature (Russel, 1980; Posner, 2005). The corresponding X-values and Y-values are calculated based on the corresponding circular coordinates, for which the X-value corresponds to level of pleasure and the Y-value corresponds to the level of activation. For example, *content* was placed at 315 degrees which correspond to X-coordinates of 0.69 and Y-coordinates of -0.69. The overall values for pleasure and activation for all variables on the model are calculated for each observation by summing all values, using the X-coordinates as weights for pleasure and the Y-coordinates as weights for activation. The scores per observation for pleasure and activation are placed onto the Circumplex model of affect. For example: an observation with a positive value on activation and a negative value on pleasure corresponds to the dimension *Unpleasant Activation.*

Figure 1

Variables placed onto the Circumplex model of affects



Notif_count: Number of times applications were opened because of a phone notification, counted per daypart.

Total_count: The total number of times applications were opened during the daypart, derived from the sum of the counts from all application categories.

Total_dur: The total number of milliseconds spent on applications during the daypart, derived from the sum of durations from all application categories.

Lastly, four models were derived from the full model which contained all measurements. The five models as displayed in Table 5 were created in order to investigate differences in predictability among blocks of measurements in panel studies. Table 5 displays the model name, the dates to which the observations in the model correspond and the amount of observations in each model.

Table 5

Blocks of measurement models

Model	Corresponding dates	Observation count
All measurements (AM)	February 21 - March 26	5,322
First half of measurement (FH)	February 21 - March 9	2,969
Last half of measurements (LH)	March 10 - March 26	2,353
First three days of measurement (F3D)	February 21 – February 23	275
Last three days of measurements (L3D)	March 24 - March 26	75

4.4 Exploratory data analysis

This section of the paper provides descriptive and visual representations of the features in the dataset. To start, Figure 2 shows the distribution of application categories used in the dataset. Social network applications were most frequently used - more than half of the measurements correspond to this application category. Games applications were least used and this class is barely visible in the figure. Figure 3 shows the distribution of the mood classes in the dataset. Participants most often experienced the mood *Pleasant Deactivation* - almost half of the mood measurements correspond to this class. *Unpleasant Activation* is least experienced in the sample.

Figure 2. Application distribution

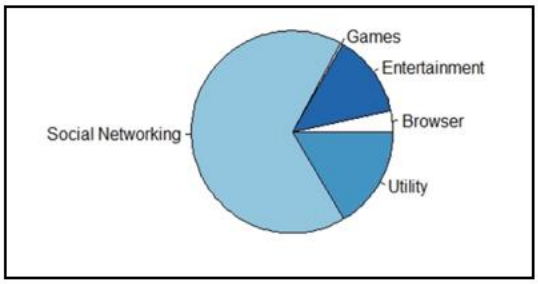
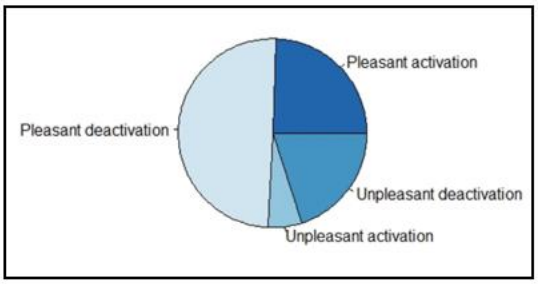


Figure 3. Mood distribution



Further, Figure 4 displays the distribution of application usage among the moods for each of the five models: all measurements (AM), measurements from the first half of the study (FH), measurements from the last half of the study (LH), measurements from the first three days (F3D) and measurements from the last three days (L3D). Social networking applications are most often used, among each mood class and among all models. Model AM, FH and LH are almost similar to each other in terms of the distribution of the four affects of the Circumplex model. Model F3D contains relatively more *Pleasant Activation* and less *Unpleasant Deactivation* than these aforementioned models. Model L3D differs most from these three models as the majority class shifted from *Pleasant Deactivation* to *Pleasant Activation* and the classes *Unpleasant Activation* and *Unpleasant Deactivation* were almost equally large. The distribution of applications among the mood classes is similar in all models and all application distributions correspond to the distribution depicted in Figure 2.

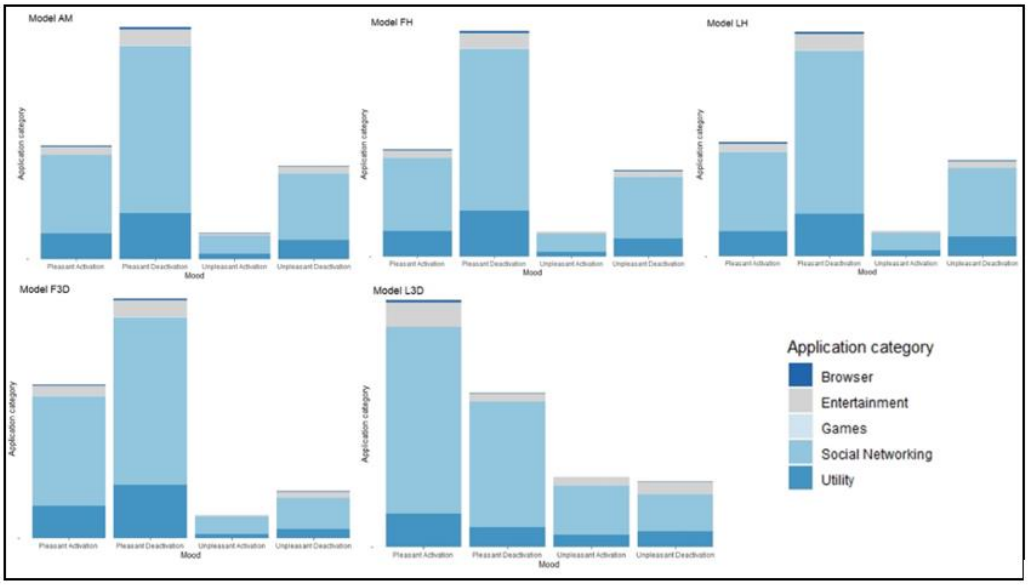


Figure 4. Application counts per mood category among the models

Lastly, the Pearson's correlation between the continuous features from the dataset are displayed in Figure 5. The Pearson's correlation measures the linear dependency between two features. Positive correlations are displayed in blue, varying from lighter shades for weaker correlations to darker shades for stronger correlations. Positive correlations indicate that an increase in one feature corresponds to an increases in the other feature. For example, *Social_count* is highly positively correlated with *total_count* (r = 0.94), indicating that participants who often open social network applications can be expected to overall open many applications. Negative correlations are displayed in red, varying from lighter shades for weaker

correlations to darker shades for stronger correlations. Negative Pearson's correlations suggest that an increase in one feature corresponds to a decrease in the other feature. For example, *total_count* is negatively correlated with *min_dur* (r = -0.07), indicating that participants who spend a long minimum time in applications can be expected to open less applications in total. The white cells in the figure indicate no correlation between the features, suggesting that an increase in one feature does not correspond to an increase or decrease in the other feature. For example, *Brow_dur* and *entert_dur* are not correlated with each other (r = 0.00). More time spent on browser applications does not indicate more or less time spent on entertainment applications.

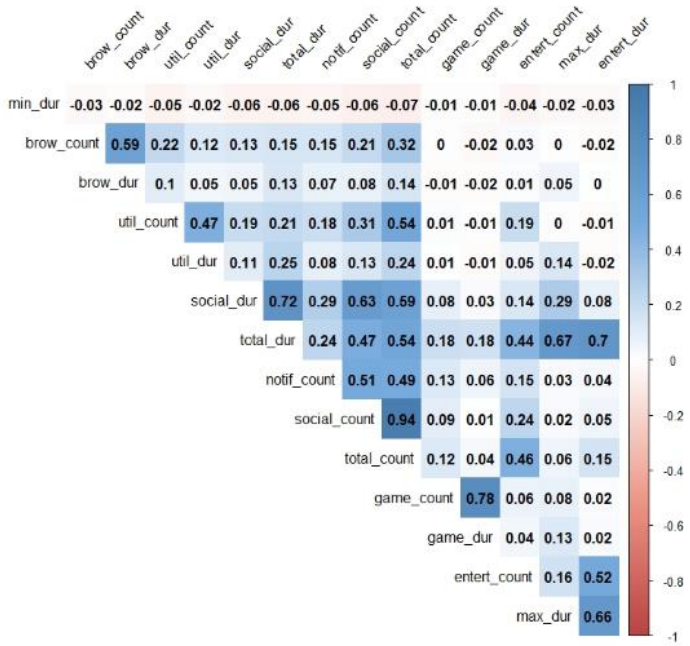


Figure 5. Correlation matrix continuous features

4.5 Data splitting

A trained machine learning model is tested on new, unseen data in order to evaluate the performance of the model. By comparing the performance of the model on the training data to the performance of the model on the test data, inferences can be made about whether the model is over-fitted, under-fitted or well generalized. This study used Cross Validation (CV) to test the performance of the designed models. CV is a resampling procedure which sets aside a part of the data when training the model and later uses this part for testing. The Train_Test Split approach from the scikit_learn library in Python was used for CV in this study (Pedregosa et al., 2011). A random split was created in which 70 percent of the data was used for the training set and 30 percent for the test set. The models were trained on the training set and the test set was used for evaluating the performance of the model.

4.6 Algorithms

This section elaborates on the learning algorithms used in this study. Several learning algorithms were trained and tuned in order to optimize the performance for predicting mood from application usage. The learning algorithms were evaluated with their accuracies and macro-F1 scores. The best performing algorithm was used again to compare the defined models for research question two.

4.6.1 Learning Algorithms

k-Nearest Neighbor Classifier (k-NN): K-NN is described as a rote learning method for classification. The classifier searches for the most similar example in the feature space for each newly given example and provides the same label to the new example. All computations are deferred until classification is required (Zualkernen, Aloul, Shapsough, Hesham & El-Khorzaty, 2017). In other words, this classifier measures the distance from the new example to the other training points in the feature space and then selects the nearest one for classification. The classification is based on similarity in the feature space. This instance based algorithm is considered to be a lazy learner as it does not actually learn a discriminative function but instead memorizes examples from the training data. Parameter *K* indicates the number of neighbors in the feature space that are considered when determining the class label of a new datapoint. Uniform or distance weights can be applied. Uniform weights provide equal weights to all neighbors whereas distance weight gives different weights to the neighbors based on their distance from the example that is being classified. The Euclidean distance function is used according to the following formula:

$$d_{Euclidean}(p,q) = \sqrt{\sum_{i=1}^{n}(p_i - q_1)^2}$$

The values p and q in the formula correspond to the feature values of the two examples between which the distance is calculated. This Euclidean distance function is used as the default distance function, referring to the straight-line distance between two examples in the feature space (Mabayoje et al., 2019). Advantages of this classifier include its simple implementation and its fairly robustness to outliers (Zualkernen et al., 2017).

Random Forest Classifier (RF): The RF classifies based on the average outcome from multiple decision trees. Decision trees contain different nodes which each test a binary condition and subsequently add a decision boundary. Each node relates to another node or to a leaf node. The leaf nodes refer to class labels and can be seen as stopping criteria (Liu & Salvendy, 2007). RF classifiers are created by introducing randomness into multiple decision trees and classifies based on the average predictions of these individual trees (Coeurjolly & Leclercq-Samson, 2018). The complexity depends on the number of trees and the depth

of the trees (Belgiu & Dragu, 2016). A main advantage of this classifier is that it is able to select relevant features from noisy environments. The RF contains several parameters, including the number of preselected directions for splitting (mtry), the tree levels (nodesize) and the number of trees. The number of trees in a forest should in principle be as large as possible so that each example feature is likely to be used (Couronné, Probst, & Boulesteix, 2018). The default values for these parameters are commonly believed to yield good predictive performance, which is one of the reasons why this classifier is popular. The default classification setting for the preselected splitting direction is mtry = √d, with d as the number of features in the dataset (Coeurjolly & Leclercq-Samson, 2018). The Gini index is seen as the measure of node purity and the default setting to split nodes is according to the following formula:

$$G = 1 - \sum_j p_j^2$$

In the formula, the sum of squared probabilities for each class is subtracted from one. Larger partitions are favored when splitting nodes (Coeurjolly et al., 2018).

Logistic Regression Classifier (LR): This linear classifier is one of the most popular algorithms for classification tasks. Classification with LG is based on probability and by default uses the Sigmoid function to map the predicted classes to probabilities between 0 and 1. In other words, the classifier uses the conditional probability score to determine the class label (Couronné et al., 2018). The regression coefficients of the decision boundary are estimated by an iterative process of the maximum likelihood estimation in which a tentative solution is repeatedly revised until no further improvement is found (Memisevic, Zach, Hinton, & Pollefeys, 2010). The logistic regression classifier contains the threshold *c:*

$$Y = 1 \; if \; P(Y = 1) > c$$

This threshold is typically used to determine the strength of the regularization, the value $c = 0.5$ is a commonly used threshold. LR includes a regularization penalty which prevents the algorithm from overfitting on the training data. L1 penalty uses Least Absolute Shrinkage and Selection Operator (Lasso) regression, in which the absolute value of magnitude of the coefficients are added to the loss function as penalty scores. Lasso regression shirks the coefficients of less relevant features to zero and thus removes the influence of these features. The L1 penalty works well for samples with a large number of features. L2 penalty uses ridge regression in which the squared magnitude of the coefficients are added to the loss function as penalty scores. The L2 penalty works well for preventing overfitting, but might result in underfitting as it might add to much weight to the features (Pereira, Baso, & Silva, 2016).

Support Vector Machine Classifier (SVM): SVM has been described as one of the most efficient algorithms and it does not need previous defined data assumptions as long as the right kernel function is chosen. The kernel function refers to a class of algorithms for pattern analysis. The SVM constructs hyperplanes as optimal decision boundaries between classes in a multidimensional feature space. Kernels can create linearly separable features by performing mathematical calculations on non-linearly separable features, enabling the SVM to create more complex decision boundaries. Hyperparameter *C* refers to the proportion of misclassification for which the default value $C = 1$ is implemented. Hyperparameter *Gamma* refers to the range of influence of single training examples in the feature space. Lower values of *Gamma* indicate the influence of a single example to reach further whereas higher values indicate the influence to reach closer to the example (Karamizadeh, Abdullah, Halimi, Shayan, & Rajabi, 2014).

4.6.2 Feature scaling

Scaling is used in machine learning to convert all features into a relatively similar scales as features of different length might affect the performance of the classifier. Features in the LR, k-NN, and SVM classifiers are scaled as these algorithms are expected to perform better with scaled features. The RF classifier is expected to perform well with unscaled data (Hale, 2019). However, the RF classifier in this study seemed to perform slightly better with scaled features and therefore in this study scaled features were used for the RF classifier anyways. The MinMaxScaler from the Scikit-learn library (Pedregosa et al., 2011) was used to convert all features to the same relative scale. The MinMaxScaler subtracts the minimum value in the column and divides this value by the difference between the original maximum value and the original minimum value. All datapoints are scaled into values between -1 and 1 while maintaining the relative space between each feature's values (Hale, 2019).

4.6.3 Parameter tuning

The learning algorithms as discussed above all contain parameters that can be tuned in order to optimize the performance of the algorithm. Several parameters are tuned in order to find the values that increase the performance of the algorithm. Others were kept to their default setting, reflecting the best settings for general performance (Mabayoje et al., 2019). GridSearch and CV were used for parameter tuning in this study. This technique runs an exhaustive search through a predefined subset of parameters in the learning algorithm, guided by CV or other performance metrics. GridSearch iterates over the defined hyperparameter values in the grid (Table 6) and searches for the optimal parameter settings (Consoli, Kustra, Vos, Henriks, & Mavroeidis, 2018).

Table 6

Parameter tuning in GridSearch

Learning algorithm	Parameters	Implemented values
k-Nearest Neighbor (k-NN)	*N_neighbors:* number of neighbors	List(range(1,25))
	Weights: weight function	Uniform; Distance
Random Forest (RF):	*N_estimators:* number of trees in the forest	List(range(10, 101, 10))
	Max_depth: maximum depth of the tree	List(range(2, 20, 2))
Logistic Regression (LR):	*Penalty:* norm used in the penalty	L1, L2
	C: inverse of regularization strength	Np.logspace(-3, 3, 7)
Support Vector Machine (SVM):	*Kernel:* kernel type	Rbf; Linear; Sigmoid
	Gamma: kernel coefficients	Scale; Auto
	Decision_function_shape: type of decision function to return	Ovo; Ovr

4.7 Evaluation methods

Performance of the learning algorithms was optimized by maximizing the accuracy scores. Accuracy refers to the proportion of correctly classified true predictions among the total number of predictions. A higher accuracy score would indicate better classification. The following formula is used to calculate accuracy:

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative}$$

The accuracy scores of the different learning algorithms will be compared to each other and to the majority baseline in order to evaluate what algorithm performs best and how much the algorithms have learned. The majority baseline will be calculated by dividing the count of the most frequently occurring class by the total count. For the second research question, the accuracy score will be compared to the majority baseline in order to compare differences in predictability among the defined models. However, accuracy scores can be misleading for imbalanced classes as they might only observe the proportion of correctly classified examples

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

Since figure 3 suggests the classes in the dataset to be imbalanced, the macro-F1 score provides better insights into how well a model with unbalanced classes is performing. The F1 score is the harmonic mean of precision and recall where the relative contribution of precision and recall is equal. Recall indicates what percentage of the positive examples in the dataset was predicted as positive. Precision refers to the percentage of positive class prediction that actually belong to the positive class. This study chose to use the macro-average F1-score, or macro-F1 in short, to gain insight in how well the algorithm is able to predict each class. The macro-average F1-score computes an arithmetic mean of the F1 scores per class providing equal weights to each class (Shmueli, 2019). The formulas are as follows:

$$Macro - F1 = \frac{F1_{Class1} + F1_{Class2} + \cdots + F1_{ClassN}}{N}$$

Furthermore, confusion matrixes are constructed to provide easy to interpret overviews of how well the model is doing and what type of errors are made. The cells in the matrix display the number of True Positives, False Positives, False Negatives and True Negatives of the model.

5. Results

This section of the paper elaborates on the results of the analyses. The predictive performance of the learning algorithms is discussed in section 5.1. Predictive performance among the models from different blocks of measurements is elaborated on in section 5.2.

5.1 Classification models

Parameter tuning in GridSearch suggested the parameter settings as displayed in Table 7 to optimize performance of the algorithms. The k-NN classifier performed best on the tasks using *N_neighbors = 24* and *Weights = Uniform.* The RF classifier showed optimal performance with *N_estimators = 100* and *Max_depth = 12*. LR classified best using *Penalty = L2* and *C = 0.1*. Lastly, the SVM classifier scored the highest accuracy score using *Gamma = Scale, Kernel = Rbf* and *Decision_function_shape = ovo*.

Table 7

Optimal parameters for the learning algorithms

Learning algorithm	Parameters in GridSearch	Optimal settings
k-Nearest Neighbor (k-NN)	N_neighbors: list(range(1,25))	N_neighbors: 24
	Weights: Uniform; Distance	Weights: Uniform
Random Forest Classifier (RF):	N_estimators: list(range(10, 101, 10))	N_estimators: 100
	Max_depth: list(range(2, 20, 2))	Max_depth: 12
Logistic Regression Classifier (LR):	Pentaly : L1, L2	Penalty: L2
	C : np.logspace(-3, 3, 7)	C: 0.1
Support Vector Machine Classifier (SVM):	Kernel: Rbf, Linear; Sigmoid	Kernel: Rbf
	Gamma: Scale; Auto	Gamma: Scale
	Decision_function_shape: Ovo; Ovr	Decision_function_shape: Ovo

Results from the categorical mood predictions of the learning algorithms are displayed in Table 8. The majority baseline before splitting the data is 47.48%. In other words, 47.48% of the data corresponds to the most frequent mood category (*Pleasant Deactivation*). The majority baseline for the training set corresponded to 48.00% and from the test set to 47.80%. The RF, LR and SVM classifier achieved accuracy scores above the majority baseline. The RF classifier achieved the highest accuracy (49.49%) and the highest macro-F1 score (0.27).

Table 8

Performance of the Learning Algorithms

Learning algorithm	Majority Baseline y_test	Accuracy	Baseline comparison	Macro-F1
k-Nearest Neighbor (k-NN)	0.47796	0. 47165	-0.00631	0.25
Random Forest Classifier (RF):	0.47796	0.49487	+0.01691	0.27
Logistic Regression Classifier (LR):	0.47796	0.48407	+0.00611	0.20
Support Vector Machine Classifier (SVM):	0.47796	0.48272	+0.00476	0.18

All algorithms scored higher on the accuracy score than on the macro-F1, indicating that the algorithms are better at predicting the True Negatives and True Positives compared to predicting False Positives and False Negatives. Figure 6 shows the confusion matrix of the SVM classifier, which achieved the lowest F1-score. The confusion matrix shows how the algorithm classifies almost all examples as the majority class Pleasant Deactivation.

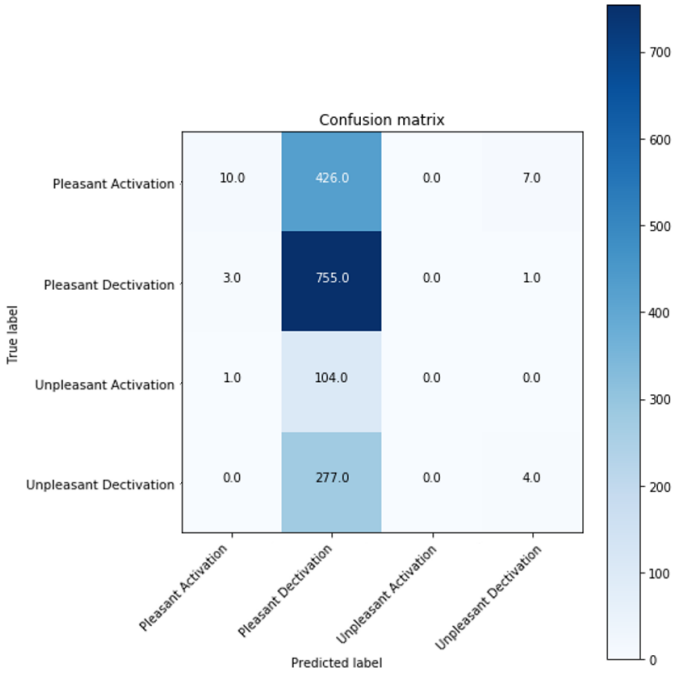


Figure 6. Confusion matrix SVM Classifier

Figure 7 displays the confusion matrix from the RF classifier. The macro-F1 (0.27) has slightly increased compared to the LR classier (0.18). The RF classifier is better able to predict classes other than the majority class.

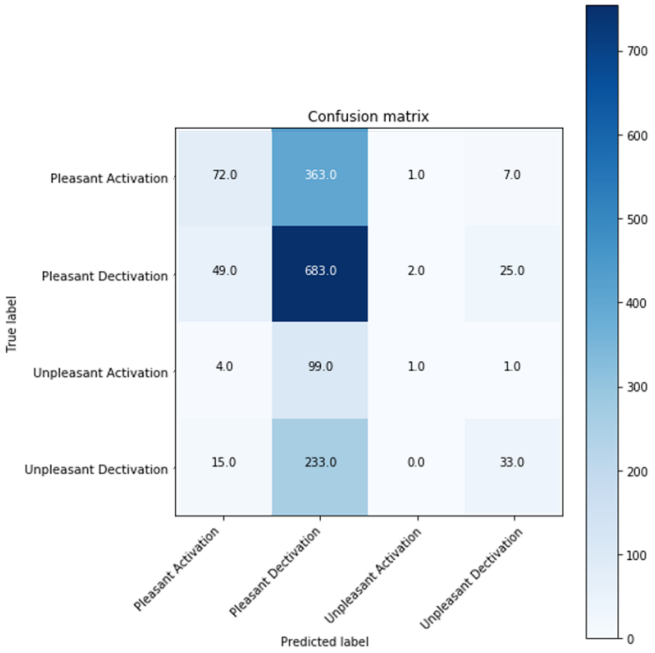


Figure 7. Confusion matrix RF Classifier

5.2 Time measurements models

The RF classifier is trained and tuned again for the defined models from different blocks of measurements during the study. GridSearch is used to tune the parameters for each model. The optimal settings per model are displayed in Table 9. The number of estimators on which the models perform best varies between 40 and 100. The maximum depth of the trees at which the accuracy scores are optimized is in the range of 4 to 16.

Table 9

Optimal parameters per model

Model	N_estimators	Max_depth
All measurements (AM)	80	8
Measurements first half of the study (FH)	90	8
Measurements last half of the study (LH)	50	4
Measurements first three days (F3D)	100	6
Measurements last three days (L3D)	40	16

Table 10 displays the performance of the models. Each model has a different majority baseline as they each contain different observations. Model L3D performed below baseline and achieved the lowest marco-F1 score. The four other models all achieved accuracy scores above baseline. Model F3D achieved the highest accuracy score (56.02%). However, the accuracy score achieved by model FH (51.24%) is highest in comparison to the corresponding baseline (49.04%) and achieves the highest macro-F1 score (0.30). Therefore, model FH is considered to best predict mood from application usage. However, the differences between the models are small and all accuracy scores are close to the baselines.

Table 10

Performance of the models

Model	Majority baseline y_train	Majority baseline y_test	Accuracy	Baseline comparison	Macro-F1
All measures (AM)	0.48002	0.47796	0.49487	+0.01691	0.27
Measures first half of the study (FH)	0.49637	0.49039	0.51235	+0.02196	0.30
Measures last half of the study (LH)	0.45638	0.46942	0.48017	+0.01075	0.27
Measures first three days (F3D)	0.54450	0.54878	0.56021	+0.01143	0.24
Measures last three days (L3D)	0.44231	0.56522	0.53846	-0.02676	0.19

The confusion matrix of model FH (see Figure 8) displays how the predictions are relatively more distributed among the different classes in comparison to the algorithms from Figure 6 and 7. However, the FH model still classifies most examples as the majority class, Pleasant Deactivation.

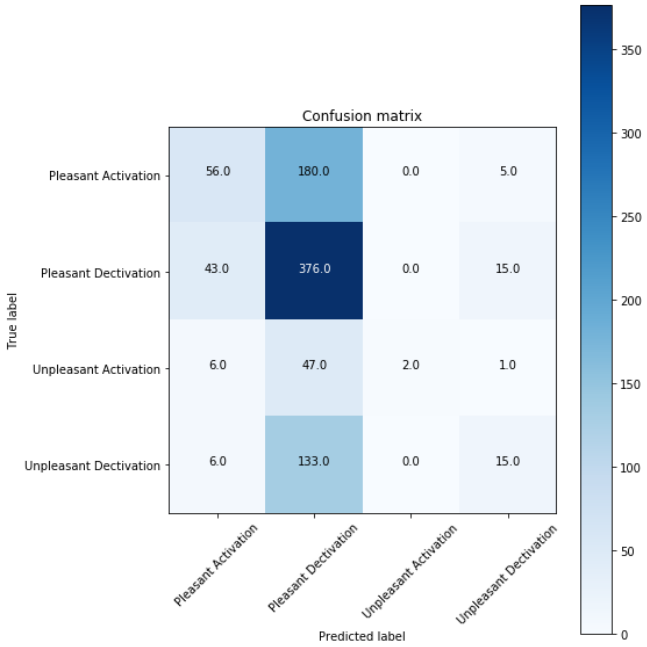


*Figure 8.*Confusion matrix model FH

6. Discussion

This section of the paper discusses the results with regard to the research questions. Moreover, the limitations of this study are elaborated on and suggestions for further research and practical implications are provided.

6.1 Findings

The goal of this study was to find the extent to which smartphone application usage can predict mood. This goal was achieved through two research questions, focusing on what learning algorithm performs best and whether the predictive performance of this algorithm differs among various times of measurement in the panel study. The results of both research questions are discussed below.

6.1.1 Classification models

This first research question focused on the performance of different learning algorithms on predicting mood from application usage. The following algorithms were built and tuned using GridSearch: k-Nearest Neighbor (k-NN), Random Forest Classifier (RF), Logistic Regression Classifier (LR) and Support Vector Machine Classifier (SVM). The performance of the models was evaluated using the accuracy scores and the macro-F1 scores. The RF classifier achieved both the highest accuracy score (49.49%) against the majority baseline (47.79%) and the highest macro-F1 score (0.27), suggesting that the RF classifier makes the most correct predictions and best predicts each class. Previous studies suggested the RF and SVM to often outperform other well-established techniques like k-NN and LR when performing multiclass classification tasks, like in the current study (Kremic & Subasi, 2016). However, the low marco-F1 score reflects how the SVM did a poor job predicting the minority classes. This can be explained by the tendency of parameter *K* to take the observed categories' frequencies as given (Bogolomov et al., 2013). The RF classifier has been suggested to overall be the best algorithm for multiclass classification (Statnikov & Aliferis, 2007) and has been discussed to be the best predictor in cases with many redundant features and when there are complex interactions to be captured and computed in simpler spaces (Kremic & Subasi, 2016). More specifically, the RF classifier has been found to best predict happiness on application usage in a three-class prediction task in comparison to other learning algorithms (Bolomolov et al., 2013). It was therefore not unexpected that the RF outperformed the other learning algorithms on this multiclass classification task. Furthermore, all algorithms achieved lower macro-F1 scores than accuracy scores. The models often predicted the most occurring class (*Pleasant Deactivation*) as they needed to predict four unbalanced categories. In other words, the models did a poor job predicting the least occurring classes, resulting in lower F1-scores in comparison to the accuracy scores. The k-NN classifier achieved an accuracy score below baseline.

The other three algorithms achieved accuracy scores slightly above the baseline. However, their performance is considered to be low.

6.1.2 Time measurement models

The aim of the second research question was to observe differences in macro-F1 scores and accuracy scores in order to observe whether predictive performance differs among blocks of measurements in panel studies. The following models were created: all measurements (AM), measurements from the first half (FH), measurements from the last half (LH), measurements from the first three days (F3D) and measurements from last three days (L3D). Model L3D was the least accurate in comparison to the baseline (-2.68%) and reflected the lowest macro F1-score (0.19). The other models achieved accuracy scores above baseline, ranging from 1.11% to 2.20% above baseline. Regardless of the baselines, the highest accuracy scores were achieved by model F3D (56.02%) and model L3D (53.85%). However, these scores were not (-2.68%) or slightly (+1.11%) above baseline and both models achieved lower macro-F1 scores in comparison to the other models. This suggests that the amount of data from three days of measurements might be too limited, resulting in poor predictions for minority classes. Model FH achieved the highest accuracy score (51.24 %) above baseline (+2.20%) and the highest macro-F1 score (0.30), suggesting model FH does the best job making correct predictions in comparison to the baseline and best predicts each class. This is in line with the literature which argued panel conditioning and panel attrition to affect the quality of the study (Halpern-Manners et al., 2012; Damen, et al., 2015). Regarding panel conditioning, participants can be expected to pay better attention to measurements in the beginning of the study. This attention decreases as the study continues (Wilson & Kraft, 1993). As a result, the measurements from the beginning of the study are expected to contain less bias and in turn are suggested to yield better predictive performance when used in machine learning. Regarding panel attrition, participants can be expected to increasingly drop out during the study, especially since the questionnaires in this study were not mandatory. Participants that dropout might differ from those that participate in the full study meaning that measurements become increasingly polluted during the study. The relatively cleaner data at the beginning of the study is suggested to yield better predictive performance when used in machine learning. However, model FH did not score high above the majority baseline nor did it achieve a good macro-F1 score. All defined models are considered to perform poorly on the task of predicting mood from application usage.

6.2. Limitations and future research

The current study contains various limitations. The first and foremost limitation is that the current study only had access to a limited amount of mood measurements. The measurements correspond to day-parts whereas more frequent mood measurements might better reflect the mood experienced during application usage. As a result, it was not possible to capture the experienced mood at the moment that the smartphone was used. For example, a participant feels stressed for half an hour in the morning and as a coping mechanism open relatively many social networking applications during that time. The mood measurement might not have captured this moment of stress as the stress has disappeared by the time the participant fills in the questionnaire. It is recommended for future research to use an unsupervised process which measures mood while using smartphones.

Second, the exhaustive search through the defined subset of parameter settings in GridSearch has several limitations. The approach for setting the search interval is ad-hoc and no optimal setting can be guaranteed as it is chosen by aliasing around this set. Moreover, GridSearch is vulnerable to local minima and maxima in the feature space. The hyperparameter can chose the local maximum or minimum value as best parameter setting in case the algorithm gets stuck at this local points in the feature space (Consoli, Kustra, Vos, Henriks, & Mavroeidis, 2018), which affects model performance. It is therefore proposed that further research compares different optimization techniques for the optimal parameter settings, like random search.

Third, the sample in this study was fairly homogeneous as characteristics of the participants were similar in terms of age, study and geographical scope as the sample was derived from a population of first year psychology students from Tilburg University. Data was gathered in the relatively short period of 6 weeks. The generalizability of this study is therefore limited and the results from this study are not representative for other user groups. Future research is recommended to use a more heterogeneous sample in order to have a more representative research for the entire population of smartphone users.

Fourth, conclusions on the effects of panel studies on predictive modeling cannot be drawn with great certainty. This study is the first to combine the insights from social sciences on panel studies with the field of data science. Support has been found for panel attrition and panel conditioning to affect the predictive performance of machine learning models. However, the topic of this study is quite specific and results from a single study are always affected by the design and human judgement (Alibasa et al., 2019). Conclusions must therefore be drawn with caution and should not be generalized beyond this sample. Further research on the effects of panel studies on predictive modelling is needed to draw more certain conclusions.

7. Conclusion

Insight into how smartphone usage affects mood is important in order to enable people to use their smartphones in manners that support rather than hinder their wellbeing. More information about the relationship between smartphone usage and mood is needed in order to reconsider application usage. This study aimed to further investigate the relationship between application usage and mood as previous research found contradicting relationships and recommended further investigation (Alibasa, Calvo & Yacef; LiKamWA et al., 2013). Data from 124 participants, measured during a period of 34 days, were used to test learning algorithms and compare models from different blocks of measurements.

The first research question focused on what learning algorithm best predicts mood from application usage. Based on the findings, the RF classifier is the best at predicting unbalances mood classes from application usage. This result was not unexpected as previous studies suggested the RF to outperform SVM, k-NN and LR classifiers in multiclassification tasks (Kremic & Subasi, 2016; Bolomolov et al., 2013). The second research question focused on differences in predictive performance among blocks of measurement. Findings suggest there are small differences in predictive performance among different blocks of measurements in panel studies. The model containing data from the first half of the study scored highest in comparison to the other defined models. This is in line with panel attrition and panel conditioning as widely discussed in social sciences (Lugtig, 2014; Warren & Halpern-Manners, 2012).

The current study was the first to combine the insights from social sciences on panel studies with the field of data science. However, the topic of this study was quite specific and the outcomes could have been affected by human judgement and design of the analysis. Further research on the effects of panel studies on predictive modelling is recommended in order to obtain more confidence with regards to the results. Further, it is recommended for future research to use more frequent mood measurements as it was not possible to capture the experienced mood at the moment that the smartphone was used with the limited measurements from this study.

The overall problem statement in this study was to investigate the extent to which application usage can predict mood among blocks of measurement in panel studies. The performance among all learning algorithms and among all models was considered to be poor as most models only slightly outperformed the majority baseline. Results suggest application usage not to contain enough useful information to make good predictions about mood. Based on the research results, it is not possible to make any recommendation on how to use smartphones in a different manner to support wellbeing.

References

Alibasa, M. J., Calvo, R. A. & Yacef, K. (2019). Sequential Pattern Mining Suggests Wellbeing Supportive Behaviors. *IEEE Access*, 7, 130133-130143. Retrieved from: https://ieeexplore.ieee.org/abstract/document/8826286

Barkhuus, L. & Polichar, V.E. (2011). Empowerment through seamfulness: smart phones in everyday life. Pers Ubiquit Comput 15, 629–639. Doi :10.1007/s00779-010-0342-4

Beliu, M. & Dragu, L. (2016). Random forests in remote sensing: a review of applications and future directions. Photogramm Remote Sens, 114, 23-31. Retrieved from: https://doi.org/10.1016/j.isprsjprs.2016.01.011

Bogomolov, A., Lepri, B. & Pianesi, F. (2013). Happiness Recognition from Mobile Phone Data.*201*3 International Conference on Social Computing, Alexandria. 790-795. Retrieved from: https://ieeexplore.ieee.org/abstract/document/6693415

Böhmer, M., Hecht, B., Schöning, J., Krüger, A., & Bauer, G. (2011). Falling Asleep with Angry Birds, Facebook and Kindle - A Large Scale Study on Mobile Application Usage. Proceeding MobileHCI, 47-56. doi:10.1145/2037373.2037383

CBS (2020). Internet; toegang, gebruik en faciliteit. Retrieved from: https://opendata.cbs.nl/statline/#/CBS/nl/dataset/83429NED/table?dl=91F4

Coeurjolly, J. & Leclercq-Samson, A. (2018). Tuning parameters in random forests. ESAIM: proceeding and sutrvey, 60, 144-162. doi: 10.1051/proc/201760144

Couronné, R., Probst, P. & Boulesteix, A. (2018). Random forest versus logistic regression: a large-scale benchmark experiment. BMC Bioinformatics, 19, 270. doi: 10.1186/s12859-018-2264-5

Consoli, S., Kustra, J., Vos, P., Hendriks, M., Mavroeidis, D. (2017). Towards an automated method based on Iterated Local Search optimization for tuning the parameters of Support Vector Machines. Benelearn 2017 conference, 1–3. Retrieved from: https://arxiv.org/abs/1707.03191

Damen, N.L., Versteeg, H., Serrys, P.W., Geuns, R.M., Domburg, R.T., Pedersen, S.S. & Boersma, E. (2015). Cardiac patients who completed a longitudinal psychosocial study had a different clinical and psychosocial baseline profile than patients who dropped out prematurely. European Journal of Preventive Cardiology, 22(2), 196-199. doi: 10.1177/2047487313506548

Dimitriadou, E., Hornik, K., Leisch, F., Meyer, D. & Weingessel, A. (2008). e1071: Misc Functions of the Department of Statistics (e1071), TU Wien. Retrieved from: http: //CRAN.R-project.org/package=e1071.

Dowle, M., Srinivasan, A .(2017). Data.table: Extension of 'data.frame'. Retrieved from: https://CRAN.R-project.org/package=data.table.

Ferdous, R., Osmani, V., & Mayora, O. (2015). Smartphone app usage as predictor of perceived stress levels at workplace. 9th International Conference on Pervasive Computing Technologies for Healthcare. Retrieved from: https://venetosmani.com/publications/Smartphone_app_usage_as_a_predictor_of_perceived_stress_levels_at_workplace_PH15.pdf

Georgoula, I., Pournarakis, D., Bilanokos, C., Sotiropoulos, D.N. & Giaglis, G.M. (2015). Using time-series and sentiment analysis to detect the determinants of bitcoin prices. MCIS 2015 Proceedings. 20. Retrieved from: http://aisel.aisnet.org/mcis2015/20

Hale, J. (2019). Scale, Standardize, or Normalize with Scikit-Learn. Towards Data Science, March 4, 2019. Retrieved from: https://towardsdatascience.com/scale-standardize-or-normalize-with-scikit-learn-6ccc7d176a02

Halpern-Manners, A., Warren, J.R. & Torche, F. (2014). Panel Conditioning in a Longitudinal Study of Illicit Behaviors, *Public Opinion Quarterly*, 78(3), 565–590. doi: 10.1093/poq/nfu029

Hung, G.C., Yang, P., Chang, C., Chang, J. & Chen, Y. (2016). Predicting negative emotions based on mobile phone usage patterns: an exploratory study. doi: 10.2196/resprot.5551

James, D. & Hornik, K. (2010). Chron: Chronological objects which can handle dates and times. R port by Kurt Hornik. Retrieved from: http: //CRAN.R-project.org/package=chron.

Kanj, S., Abdallah, F., Denœux, T. et al. (2016). Editing training data for multi-label classification with the k-nearest neighbor rule. Pattern Anal Applications, 19, 145–161. doi: 1007/s10044-015-0452-8

Karamizadeh, S., Abdullah, S., Halimi, M., Shayan, J., & Rajabi, M. (2014). Advantage and Drawback of Support Vector Machine Functionality. International Conference on Computer, Communication, and Control Technology, 63-65. doi:10.1109/I4CT.2014.6914146

Kremic, E., & Subasi, A. (2016). Performance of Random Forest and SVM in Face Recognition. The International Arab Journal of Information Technology, 13(2), 287-293. Retrieved from: iajit.org

Liu, Y., & Salvendy, G. (2007). Interactive visual decision tree classification. Proceedings of the 12th international conference on Human-computer interaction: interaction platforms and techniques, 92-105. Retrieved from: https://link.springer.com/chapter/10.1007/978-3-540-73107-8_11

LiKamWA, R. Liu, Y., Lane, N.D., Zhong, L. (2013). Can your smartphone infer your mood? In PhoneSense Workshop. Retrieved from: https://yecl.org/publications/likamwa11phonesense.pdf

Linnhoff, S. & Smith, K.T. (2017) An examination of mobile app usage and the user's life satisfaction. Journal of Strategic Marketing, 25(7), 581-617. doi: 10.1080/0965254X.2016.1195857

Lutgig, P. (2014) - Panel Attrition: Separating Stayers, Fast Attriters, Gradual Attriters, and Lurkers. Sociological Methods & Research, 43(4), 699-723. doi: 10.1177/0049124113520305

McKinney, W. (2011). Data structes for statistical computing in Pyhton. Python in Science Conference, 9, 51-56.

Mabayoje, M.A, Balogun, .A. O. , Jibril, H. A., Atoyebi, J. O. , Mojeed, H. A. & Adeyemo, V. E. (2019). Parameter tuning in KNN for software defect prediction: an empirical analysis. Jurnal Teknologi dan Sistem Komputer, 7(4), 121-126. doi: 10.14710/jtsiskom.7.4.2019.121-126

Memisevic, R., Zach, C., Hinton, G., & Pollefeys, M. (2010). Gated Softmax Classification. Advances in Neural Information Processing Systems, 23, 1603-1611. Retrieved from: http://papers.nips.cc/paper/3895-gated-softmax-classification

Oliphant, T.E. (2006). A guide to NumPy. Trelgol Publishing USA, 1. Retrieved from: https://ecs.wgtn.ac.nz/

Pereira, M.J., Basto, M., da Silva, A.M. (2016). The logistic lasso and ridge regression in predicting corporate failure. Procedia Econics and Finance, 39, 634-641. doi: 10.1016/S2212-5671(16)30310-0

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vandeplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830. Retrieved from: http://scikit-learn.sourceforge.net.

Posner, J., Russell, J.A. & Peterson, B.S. (2005). The Circumplex Model of Affect: An Integrative Approach to Affective Neuroscience, Cognitive Development, and Psychopathology. Development and Psychopathology, 17(3), 715–734. doi:10.1017/S0954579405050340

Preotiuc-Pietro, D., Schartz, H.A., Park, G., Eichstaedt, J.C., Kern, M., Ungar, L., Schulman, E.P. (2016). Modelling valence and arousal in Facebook posts. NAACL-HLT 2016, 9-15. Retrieved from: https://www.aclweb.org/anthology/W16-0404.pdf

Pukrop, R. (2000). Circumplex models for the similarity relationships between higher-order factors of personality and personality disorders: An empirical analysis. Comprehensive Psychiatry 41(6), 438. Retrieved from: https://www.sciencedirect.com/science/article/pii/S0010440X0067952X

Sharar, S.R., Alamdari A., Hoffer, C., Hoffman, H.G., Jensen, M.P. & Patterson, D.R. (2016). Circumplex model of affect: a measure of pleasure and arousal during virtual reality distraction analgesia. Games for health journal, 5(3), 197-202. Retrieved from: http://doi.org/10.1089/g4h.2015.0046

SIDN (2018). Smartphone: spin in het Nederlandse web. Onderzoek trends in internetgebruik 2018. Retrieved from: https://assets.ctfassets.net/qhzu5l2wyxby/68qEO2uhSxmnSd9aLQY1uw/cd39d1f09f9f6763ccb6961025acd6a7/SIDN_Trends_in_internetgebruik_2018.pdf

Shmueli, B. (2019). Multi-Class Metrics Made Simple, Part II: the F1-score. Towards Data Science, July 3. Retrieved from: https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-ebe8b2c2ca1

Spinu, V., Grolemund, G. and Wickham, H. (2017). Lubridate: Make Dealing with Dates a Little Easier. Retrieved from: https://cran.r-project.org/package=lubridate.

Warren, J.R. & Halpern-Manners, A. (2012). Panel conditioning in longitudinal social science surveys. Sociological Methods & Research, 41(4), 491-534. doi: 10.1177/0049124112460374

Waterton, J. & Lievesley, D. (1989). Evidence of Conditioning Effects in the British Social Attitudes Panel Survey. Panel Surveys, 319(39).

Wickham, H. & Francois, R. (2017). Dplyr: a grammar of data manipulation. Retrieved from: http://cran.r-project.org /web/packages/dplyr/index.html.

Wickham, H. & Hadley (2017). Tidyverse: Easily Install and Load the 'Tidyverse'." Retrieved from: https://CRAN.R-project.org/package=tidyverse.

Wickham, H. & Winston, C. (2009). Ggplot2: create elegant data visualizations using the grammar of graphics. Retrieved from: http://cran.r-project.org/pac kage=ggplot2

Wilson, T. D. &Kraft, D. (1993). Why Do I Love Thee: Effects of Repeated Introspections about a Dating Relationship on Attitudes toward the Relationship. Personality and Social Psychology Bulletin, 19(409), 18. Retrieved from: https://journals.sagepub.com/doi/pdf/10.1177/0146167293194006

Wooyeon, K. (2020). Musemo: Express musical emotion based on neural network. Graduate school of UNIST. Retrieved from: https://scholarworks.unist.ac.kr/handle/201301/31791

Zualkernen, I., Aloul, F., Shapsough, S., Hesham, A. & El-Khorzaty, Y. (2017). Emotion recognition using mobile phones. Computers & Electrical engineering, 60, 1-13. Retrieved from: https://www.sciencedirect.com/science/article/pii/S0045790617312752

Appendix A

The programming codes used to conduct this research are visible in GitHub via the following link: *https://github.com/MVerhoeven96/MasterThesis2020*