The criterion validity of smartphone sensor data for predicting AUDIT scores

Metadata

Study Information

Hypotheses

4.1. Our first research aim is to quantify to what extent a machine learning model fit on passive sensing features from the AWARE smartphone app is able to accurately predict AUDIT scores.
4.2. Our second research aim is to quantify to what extent a machine learning model fit on passive sensing features from the AWARE smartphone app is able to accurately predict AUDIT scores at six months following the initial baseline measurement, as well as change in AUDIT scores from baseline.
4.3. Our third research aim is to quantify each feature's relative role in accurately predicting AUDIT scores and AUDIT change.
4.4. Our fourth research aim is to interpret the effect of each feature on AUDIT scores and AUDIT change.
4.5. Our fifth research aim is to quantify how many weeks of passive sensing data (p to eight) are needed until the predictive accuracy of the machine learning model stabilizes.

Design Plan

Study type

Observational Study - Data is collected from study subjects that are not randomly assigned to a treatment. This includes surveys, "natural experiments," and regression discontinuity designs.

Blinding

No blinding is involved in this study.

Is there any additional blinding in this study?

No response

Study design

Participants completed a baseline questionnaire followed by 56 days of passive sensing, during which the AWARE app was installed on their smartphone. Then participants completed a follow-up questionnaire similar to the baseline six months after the beginning of the passive sensing period.

No files selected

Randomization

No response

Sampling Plan

Existing Data

Registration prior to any human observation of the data

Explanation of existing data

11.1. At the time of preregistration, all participants have completed the baseline questionnaire, the passive sensing period, and the 6-months follow-up questionnaire. We have not yet accessed, observed, or analyzed any of the data involved in analysis, and are not aware of any summary statistics or patterns in the data. We provided 10 cases of data to Co-Authors Han Zhang and YiYi Ren to facilitate writing data processing scripts. They were not otherwise involved in data collection and had no access to the survey data prior to pre-registration.

Data collection procedures

Participants were young adults at baseline (n = 497, age 18 – 22, Mage = 20.3, SD = 1.3, 45% cisgender female, 42% cisgender male, assigned male at birth), with the remaining participants identifying as nonbinary/gender queer/gender nonconforming (8.5%), transgender male or female (4.0%), or nongendered (0.2%). Participants were recruited from King, Pierce, and Snohomish Counties in Washington State from both college and non-college sources to ensure a representative sample of young adults in Washington State. We recruited using internet (Facebook, Instagram, TikTok, YouTube, Twitter, Craigslist, Reddit, and emails to university registrar lists and high school list-servs) and non-internet (newspaper advertisements and flyers) sources. Participants were required to be between the ages of 18 and 22 at study screening, own a smartphone, be fluent in English, and reported drinking or using marijuana "about once per week" or more over the past three months. Participants were excluded if they were not fluent in English or if they moved to the United States after age 12. Participants endorsed a variety of race/ethnicities: 54% solely non-Hispanic White, 28.5% Asian, 6.6% African American, 8.37% Hispanic/Latino, and 22.7% who endorsed more

than one ethnicity. Most participants identified as heterosexual (52%), with the remaining participants either identifying as LGBQ+ (47.6%) or declining to respond (n = 2). Finally, 9.8% of the sample was born outside the U.S. Racial/ethnic proportions broadly reflected Washington census data from counties in which participants were recruited. Approximately 67% of the sample were attending a 4-year college at recruitment.

Participants completed a baseline survey, an EMA period coupled with smartphone monitoring for the next 8 weeks, and completed a second survey 6 months following the baseline.

No files selected

Sample size

497 participants completed the larger study, of which 283 agreed to install the AWARE smartphone app. Thus, the final sample size for this study will be 283.

Sample size rationale

Sample size justification for machine learning analyses is not straightforward, as inference is not a goal. Instead, the predictive performance of the model is evaluated in a test sample. If the sample size is too small to reliably detect patterns in the data, predictive performance in the test sample will be low. Thus, there are no concerns regarding sample size beforehand. The sample size of the larger study was based on a power analysis for a different set of analyses than the ones reported here. As we added the passive sensing to this study, we simply aimed to get passive sensing data from as many participants as possible. 283 participants ultimately agreed to download the AWARE app for the duration of the study.

Stopping rule

No response

Variables

Manipulated variables

No response

No files selected

Measured variables

Alcohol Use Disorder Identification Test: As part of the baseline and six-month follow-up questionnaire, participants completed the 10-item AUDIT (Saunders et al., 1993). Items include ‘How often do you have a drink containing alcohol?’ with answer options being 0 = ‘Never’, 1 = ‘Monthly or less’, 2 = ‘Two to four times a month’, 3 = ‘Two to three times a week’, and 4 = ‘Four or more times a week’, and ‘How often during the last year have you found that you were not able to stop drinking once you had started’ with answer options being 0 = ‘Never’, 1 = ‘Less than monthly’, 2 = ‘Monthly’, 3 = ‘Weekly’, and 4 = ‘Daily or almost daily’.

Passive smartphone data: Participants installed the AWARE app on their smartphone (Ferreira et al., 2015), which unobtrusively collects sensor data from the smartphone directly. The AWARE app collects data on battery, Bluetooth, calls, conversations, location, messages, screen, and WiFi. These data were used to create the feature sets (described under indices below). Data were collected continuously with AWARE for 8 weeks following the baseline survey.

No files selected

Indices

18.1. AUDIT score: The ten AUDIT items will be summed into one AUDIT score at both baseline and 6-month follow up. As described by Saunders et al. (1993), the first eight items are scored from 0-4 while the last two items have just three answer options that are scored 0,2,4 respectively. We measured AUDIT at the Baseline interview and at the 6 month follow up. To predict change at the 6 month follow-up, we will compute a change score for the AUDIT by subtracting the baseline AUDIT score from participants’ AUDIT scores at the 6 month follow up. Although there are disadvantages to using change scores in this manner, this allows us to predict changes in AUDIT scores without incorporating baseline AUDIT scores into the model, which would dramatically inflate model performance.
18.2. In case participants skipped individual items on the AUDIT, we will impute these items using predictive mean matching with the mice package in R, as missing items invalidate sum scores. To facilitate their use in data analysis, we will conduct single imputation for missing data.

imp = mice(baseline, m = 1, maxit = 100, method = “pmm”)

18.3. Feature extraction: Except for sleep, each feature was created for 5 different daily epochs (all day, 6am-12pm, 12pm-6pm, 6pm-12am, 12am-6am). We chose these epochs based on existing pipelines for processing AWARE data. Sleep will be computed at the day level only.
18.3.1. Referring to the feature extraction method proposed by Doryab et al. [1], we extracted the following features used in this paper.
18.3.1.1. Physical Activity. For a given day/ epoch, we calculated four physical activity-related features, including the number of transitions between various participant activity types (e.g., transitioning from a state of 'still' to 'walking'), number of unique activity types observed, total duration of each activity type, and identification of the prevailing activity type recorded with the highest frequency.
18.3.1.2. Application Usage. Prior to analysis, we undertook data pre-processing steps, which involved the exclusion of system applications to center our feature computation predominantly around user-installed applications (UIA). For a given day/ epoch, we proceeded to compute six features, including the number of unique apps used, the number of apps used per minute for each day/epoch, the most commonly used package category from the apps, the most commonly used app, total app usage duration in seconds, and app transition frequency.
18.3.1.3. Battery. We calculated the number of times users charge their phones and the total battery charging time to indicate how often and how long users charge their phones.
18.3.1.4. Bluetooth. We applied the K-means clustering algorithm to scanned Bluetooth addresses based on their frequency in the data set, and grouped the devices into 2 or 3 clusters depending on which can better separate the data points with more concentrated clusters, to differentiate the person’s own devices (labeled as “self”) and other people’s devices (labeled as “others”) [1]. We then calculated 16 Bluetooth-related features for each given day/epoch. These included the overall count of Bluetooth samples amassed, the frequency of scans for the most and least prevalent Bluetooth devices, and the analogous frequency calculations for devices associated with “self” and “others”. Moreover, the total number of distinct scanned devices, along with those unique to “self”’ and “others”, were considered. Additionally, sum, mean, and standard deviation values were computed for the scan counts of all devices, delineated further between “self” and

"others".

18.3.1.5. Calls. Similar to call features extracted by Doryab et al. [1], we also calculated 19 call features using call logs from the smartphone for each given day/epoch. These included the total number and duration of incoming, outgoing calls, and missed calls to everyone, family members, friends off-campus, and friends on-campus, and the number and identification of the most frequently engaged correspondents within the aforementioned categories.

18.3.1.6. Locations. We extracted location variance, radius of gyration, total distance traveled, and circadian movement features described by Doryab et al. [1]. We used DBSCAN [2] to group static location samples into clusters, and calculated the statistical features (e.g., sum, mean, standard deviation, maximum, and minimum) on the duration of stay at each cluster. In addition, we calculated the entropy of the duration of stay at each cluster to evaluate how students distributed their time. We inferred students' home locations by clustering their location data at night (12am to 6am). We considered a potential cluster to be a home location if the student stays there for more than 3 days in a row, and the dwelling time at the cluster is at least 80% of each night. We then calculated the total time spent at home (within 10 meters from home) and near home (within 100 meters from home) accordingly based on their home locations. In total, we calculated 38 location-related features for each given day/epoch.

18.3.1.7. Screen. We used screen data to define a phone interaction session as a time series with a screen status of "on" at the beginning and a screen status of "off" or "locked" at the end of the session. Similarly, we defined a screen unlock session to be a time series with a screen status of "unlocked" at the beginning and a screen status of "locked" at the end. We then computed statistical features (e.g., sum, max, min, mean, standard deviation) on the duration of interaction and unlock sessions. In addition, we extracted the time information of the first and last occurrence of different types of screen events (i.e., on, off, unlock and lock), and calculated the average number of unlocks per minute to indicate the frequency of a user initiating a phone interaction. In total, we calculated 17 screen features for each given day/epoch.

18.3.1.8. WiFi. We calculated three wifi features for each given day/epoch. These included the number of total collected wifi samples, the number of unique WiFi access points sensed by the phone, and the most frequently detected access point.

18.3.1.8.1. [1] Afsaneh Doryab, Prerna Chikarsel, Xinwen Liu, and Anind K Dey. "Extraction of behavioral features from smartphone and wearable data". In: arXiv preprint arXiv:1812.10394 (2018).

18.3.1.8.2. [2] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." In: kdd. Vol. 96. 34. 1996, pp. 226–231.

18.3.1.9. Following is a list of specific variable names, grouped by feature:

18.3.1.9.1. Physical Activity:

18.3.1.9.1.1. count_changes

18.3.1.9.1.2. number_of_activities

18.3.1.9.1.3. most_common_activity

18.3.1.9.1.4. activity_duration_minutes

18.3.1.9.2. App Usage:

18.3.1.9.2.1. number_of_unique_apps

18.3.1.9.2.2. apps_per_minute

18.3.1.9.2.3. app_use_time_seconds

18.3.1.9.2.4. number_of_app_changes

18.3.1.9.2.5. most_common_category

18.3.1.9.2.6. most_common_app

18.3.1.9.3. Bluetooth:

18.3.1.9.3.1. number_samples_bluetooth

18.3.1.9.3.2. num_scans_of_most_frequent_device

18.3.1.9.3.3. num_scans_of_least_frequent_device

18.3.1.9.3.4. number_unique_devices

18.3.1.9.3.5. num_scans_of_most_frequent_device_of_others

18.3.1.9.3.6. num_scans_of_least_frequent_device_of_others

18.3.1.9.3.7. number_unique_devices_of_others

18.3.1.9.3.8. num_scans_of_most_frequent_device_of_self

18.3.1.9.3.9. num_scans_of_least_frequent_device_of_self

18.3.1.9.3.10. number_unique_devices_of_self

18.3.1.9.3.11. sum_num_scans_of_all_devices_of_self

18.3.1.9.3.12. sum_num_scans_of_all_devices_of_others

18.3.1.9.3.13. avg_num_scans_of_all_devices_of_self

18.3.1.9.3.14. avg_num_scans_of_all_devices_of_others

18.3.1.9.3.15. std_num_scans_of_all_devices_of_self

18.3.1.9.3.16. std_num_scans_of_all_devices_of_others

18.3.1.9.4. Call features

18.3.1.9.4.1. number_rows_calls

18.3.1.9.4.2. number_incoming_calls

18.3.1.9.4.3. number_outgoing_calls

18.3.1.9.4.4. number_missed_calls

18.3.1.9.4.5. duration_incoming_calls_seconds

18.3.1.9.4.6. duration_outgoing_calls_seconds

18.3.1.9.4.7. most_frequent_correspondent_phone

18.3.1.9.4.8. number_of_correspondents_phone

18.3.1.9.4.9. number_incoming_calls_family

18.3.1.9.4.10. number_outgoing_calls_family

18.3.1.9.4.11. number_missed_calls_family

18.3.1.9.4.12. duration_incoming_calls_seconds_family

18.3.1.9.4.13. duration_outgoing_calls_seconds_family

18.3.1.9.4.14. most_frequent_correspondent_phone_family

18.3.1.9.4.15. number_of_correspondents_phone_family

18.3.1.9.4.16. number_incoming_calls_friends_on_campus

18.3.1.9.4.17. number_outgoing_calls_friends_on_campus

18.3.1.9.4.18. number_missed_calls_friends_on_campus

18.3.1.9.4.19. duration_incoming_calls_seconds_friends_on_campus

18.3.1.9.4.20. duration_outgoing_calls_seconds_friends_on_campus
18.3.1.9.4.21. most_frequent_correspondent_phone_friends_on_campus
18.3.1.9.4.22. number_of_correspondents_phone_friends_on_campus
18.3.1.9.4.23. number_incoming_calls_friends_outside_of_campus
18.3.1.9.4.24. number_outgoing_calls_friends_outside_of_campus
18.3.1.9.4.25. number_missed_calls_friends_outside_of_campus
18.3.1.9.4.26. duration_incoming_calls_seconds_friends_outside_of_campus
18.3.1.9.4.27. duration_outgoing_calls_seconds_friends_outside_of_campus
18.3.1.9.4.28. most_frequent_correspondent_phone_friends_outside_of_campus
18.3.1.9.4.29. number_of_correspondents_phone_friends_outside_of_campus
18.3.1.9.5. Locations
18.3.1.9.5.1. number_samples_location
18.3.1.9.5.2. circadian_movement
18.3.1.9.5.3. location_entropy
18.3.1.9.5.4. location_entropy_normalized
18.3.1.9.5.5. location_variance
18.3.1.9.5.6. location_variance_log
18.3.1.9.5.7. max_len_stay_at_clusters_in_minutes
18.3.1.9.5.8. mean_len_stay_at_clusters_in_minutes
18.3.1.9.5.9. min_len_stay_at_clusters_in_minutes
18.3.1.9.5.10. std_len_stay_at_clusters_in_minutes
18.3.1.9.5.11. moving_time_percent
18.3.1.9.5.12. number_location_transitions
18.3.1.9.5.13. number_of_clusters
18.3.1.9.5.14. radius_of_gyration
18.3.1.9.5.15. speed_mean_meters_per_sec
18.3.1.9.5.16. speed_var_meters_per_sec
18.3.1.9.5.17. time_at_cluster_1
18.3.1.9.5.18. time_at_cluster_2
18.3.1.9.5.19. time_at_cluster_3
18.3.1.9.5.20. time_at_cluster_1_in_group
18.3.1.9.5.21. time_at_cluster_2_in_group
18.3.1.9.5.22. time_at_cluster_3_in_group
18.3.1.9.5.23. home_stay_time_percent_10m
18.3.1.9.5.24. total_distance_meters
18.3.1.9.5.25. home_stay_time_percent_100m
18.3.1.9.5.26. outliers_time_percent
18.3.1.9.5.27. location_entropy_local
18.3.1.9.5.28. location_entropy_normalized_local
18.3.1.9.5.29. number_of_clusters_local
18.3.1.9.5.30. moving_time_percent_local
18.3.1.9.5.31. time_at_cluster_1_local
18.3.1.9.5.32. time_at_cluster_2_local
18.3.1.9.5.33. time_at_cluster_3_local
18.3.1.9.5.34. max_len_stay_at_clusters_in_minutes_local
18.3.1.9.5.35. mean_len_stay_at_clusters_in_minutes_local
18.3.1.9.5.36. min_len_stay_at_clusters_in_minutes_local
18.3.1.9.5.37. std_len_stay_at_clusters_in_minutes_local
18.3.1.9.5.38. outliers_time_percent_local
18.3.1.9.6. Screen
18.3.1.9.6.1. number_samples_screen
18.3.1.9.6.2. unlocks_per_minute
18.3.1.9.6.3. number_of_minutes_interaction
18.3.1.9.6.4. max_len_minute_interaction_bout
18.3.1.9.6.5. min_len_minute_interaction_bout
18.3.1.9.6.6. std_len_minute_interaction_bout
18.3.1.9.6.7. mean_len_minute_interaction_bout
18.3.1.9.6.8. number_of_minutes_unlock
18.3.1.9.6.9. max_len_minute_unlock_bout
18.3.1.9.6.10. min_len_minute_unlock_bout
18.3.1.9.6.11. std_len_minute_unlock_bout
18.3.1.9.6.12. mean_len_minute_unlock_bout
18.3.1.9.6.13. first_unlock_for_grpbyday
18.3.1.9.6.14. first_on_for_grpbyday
18.3.1.9.6.15. last_unlock_for_grpbyday
18.3.1.9.6.16. last_lock_for_grpbyday
18.3.1.9.6.17. last_on_for_grpbyday
18.3.1.9.7. Wifi
18.3.1.9.7.1. number_samples_wifi
18.3.1.9.7.2. number_unique_wifi_hotspots
18.3.1.9.7.3. most_frequent_wifi
18.3.1.10. There are 103 total features that will be considered in predictive models.
18.4. Feature aggregation. Features will be aggregated to the person level within each of the five daily epochs within two weekly epochs: weekdays (Monday –

Thursday) and weekends (Friday – Sunday) respectively. "We will create a person level aggregate of features from 12am-6am, 6am-12pm, 12pm-6pm, 6pm-12am, and 12am (previous day)-12am (next day) for weekdays, and a separate person level aggregate of features from 12am-6am, 6am-12pm, 12pm-6pm, 6pm-12am, and 12am (previous day)-12am (next day) for weekends." We will create the following person level aggregates:
18.4.1. Mean
18.4.2. Median
18.4.3. Mode
18.4.4. Within-person standard deviation
18.4.5. Within-person auto-correlation
18.4.6. Within-person mean squared successive difference
18.5. There are 103 total numerical features that will be considered in models. Each of the 103 features will be aggregated into person level scores: 5 daily epochs within the 2 weekly time frames, aggregated 6 different ways (18.4.1 – 18.4.6, above). This will produce a total of 6,180 person level features considered in modeling.
18.6. Finally, prior to modeling, we will estimate correlations among features to identify any features that are perfectly correlated (r = 1.0). We will select one feature of each perfectly correlated pair for modeling.

No files selected

Analysis Plan

Statistical models

19.1. Prior to analysis, we will split our data by randomly assigning participants to a training dataset (70%) and test/holdout dataset (30%). Within the training dataset, we will use 10-fold cross-validation for parameter tuning in LASSO and random forest models, using the same folds across methods. The test dataset will be reserved to evaluate the predictive performance of the final models and will not be used for model building to prevent cross-contamination.
19.2. We will compare three methods (Random Forest, LASSO, and a Neural Network) and compare the best performing models across methods.
19.2.1. Random Forest: We will use a random forest model using the ranger package in R including 1,000 trees . We choose a high number of trees in order to assure convergence. The tuning parameters for the model (number of candidate variables to consider at each split of each tree and minimum node size) will be determined with the tuneRanger function from the tuneRanger package. We will report the final number of candidate variables and the minimum node size determined by this procedure. For the optimal model, we will report the relative importance of each feature in the prediction model. For the optimal model, we will report the shape of the association between the feature and the outcome for the top 20 features.
19.2.2. LASSO: We will fit a penalized regression model using the LASSO method with the glmnet package in R. Because we expect the AUDIT to be heavily right skewed, we will model the AUDIT as a sum score and use the family = poisson model to account for the skewed nature of the data. We will select a value of lambda, the tuning parameter, using the lambda.1se option, using 10-fold cross-validation with cv.glmnet. We will then estimate the lasso function on the full training dataset with glmnet, use those model estimates to predict AUDIT values in the test data, and estimate the RMSE for that model.
19.2.3. Neural network: We will fit a neural network model using tensorflow and keras in R. All variables will be normalized prior to training and testing in a normalization layer. We will then fit a sequential model in keras, with three layers (two nonlinear, with “relu” activation, and one linear layer). The learning process will be compiled with the optimizer adam, and the loss function mean squared error. We will then train the model over 100 epochs and an 80/20 validation split. Finally, we will use the model to predict AUDIT values in the test dataset and estimate RMSE for that model.
19.2.4. We will analyze models with two outcomes.
19.2.4.1. H1. First, we will predict AUDIT scores at baseline.
19.2.4.2. H2. Second, we will predict change in AUDIT scores from baseline to Month 6.
19.2.4.3. We will conduct these analytic steps with features aggregated across:
19.2.4.3.1. Week 1 only
19.2.4.3.2. Weeks 1 – 2
19.2.4.3.3. Weeks 1 – 3
19.2.4.3.4. Weeks 1 – 4
19.2.4.3.5. Weeks 1 – 5
19.2.4.3.6. Weeks 1 – 6
19.2.4.3.7. Weeks 1 – 7
19.2.4.3.8. Weeks 1 – 8
19.2.4.4. We will report changes in model predictive performance (e.g. RMSE) across successive weeks of data collection. The best model, which we will report fully, will be selected by RMSE.

See attached code for an analysis template.

- PR Code.R

Transformations

Features will be normalized prior to analysis.

Inference criteria

We will report three indicators of model performance from the results.
First, we will report predictive performance, an indicator of out of sample prediction of the final model from the training dataset. Predictive performance of the final model in both the training and test data will be reported as RMSE and R2. This will be interpreted as an indicator of the generalizability of the model developed in the training data. Concordance in predictive performance across the training and test data will be interpreted as the training model producing a generalizable model that avoided overfitting.

As a descriptive step, we will report the ranking of predictors based on variable importance in the test data, and we will produce partial dependence plots, which describe the direction and shape of predictors marginal associations with the outcome in the test data.

Data exclusion

Participants who are missing more than 80% of data from AWARE will be excluded from analysis on the assumption that their data are unreliable

Missing data

See above.

Exploratory analysis

All analyses are exploratory

Other

Other

NA