Drink Detection from Quantac Prototype Transdermal Alcohol Sensing Wearables Using Random Forests

Benjamin C. Lieber, Domingo J. Martinez T., Lukas Schulte

***Abstract* — Instrumenting and reporting alcohol drink consumption through a passive and continuous monitoring system has been a longstanding goal in the alcohol research community. This capability would provide key insights into drinking behavior, empower modern alcohol intervention techniques, and provide counselors and rehabilitation specialists with a new tool to encourage healthier drinking habits. In an empirical study using Quantac prototype transdermal alcohol sensing devices coupled with self-reported alcohol drink-event information, we apply random forest classification and regression models toward automatic drink detection and classification. Data from multiple users under real-life conditions is used to train and validate predictive models. Moreover, we demonstrate that incorporating temperature and relative humidity sensors data with transdermal alcohol series data improves accuracy of drinking classification.**

***Index Terms* — transdermal alcohol biosensor, transdermal alcohol concentration, machine learning, random forests, time series prediction**

I. Introduction

Accurately monitoring alcohol consumption is challenging in most real-world scenarios. Studies where objective measurement of alcohol consumption is desired are often conducted under the watchful eye of an administrator and/or in a controlled lab environment. Some drawbacks of this approach are that it is harder to gauge real-life social dynamics, how a subject would drink in an unsupervised environment, and it has been reported that even asking about drinking behavior can alter drinking behavior. [1] Moreover, due to high cost and effort per participant, studies are limited in sample size.

In both research and clinical applications, alcohol consumption is commonly measured using retrospective self-report measures such as the Time Line Follow Back (TLFB) [2]. These measures depend on the ability and willingness of a participant to provide an accurate record of their drinking. Breath based systems are also commonly employed. These require compromise to naturalistic drinking as the subject must stop drinking to take the measure and in cases where an accurate BAC measurement is desired, the subject must abstain from food or drink (including alcohol) for many minutes.

The appeal of a transdermal alcohol measurement system is that it is noninvasive, passive, and relatively simple to use [3]. In the past, two prevailing transdermal alcohol concentration sensing devices have been tested, the Alcohol Monitoring Systems Inc. Secure Continuous Remote Alcohol Monitor (SCRAM) and the Giner, Inc. WrisTAS™ [3, 4, 5]. The Giner device is no longer manufactured. SCRAM remains a commercially available device and is primarily employed for abstinence compliance applications. Some drawbacks of SCRAM currently, however, are that it provides limited data for analysis (e.g. one raw datapoint every 30 minutes), it is expensive (estimated $8-10 per subject per day), and it carries stigma as it is a large device worn on the ankle commonly used in legally mandated scenarios.

A significant challenge with transdermal devices remains calibration and/or error compensation. While breath alcohol analyzers can be used in practice without calibration to each individual subject, similar patient and device independent calibration involving a single linear gain, simply will not work in the case of transdermal measurements. [3] From our initial results testing devices in real-world scenarios, we've observed many contributing factors to variation including:

- Device manufacturing differences
- Physiology of user
- Activity of user
- Positioning of device/sensor on skin
- Ambient conditions

Given high variance, the challenge can be viewed as a data collection problem – first to collect sufficient alcohol sensor data across devices, users, environments, etc., but also to collect supplemental data such as user physiology in hopes of capturing the factors contributing to different sensor output series. Machine learning offers promise at least as a first-pass approach to develop predictive models using data and training from our platform.

We posit that with recent progress in wearable technology, mobile and cloud platforms, and data analysis tools, it is possible to create a continuous alcohol-monitoring and feedback system which performs:

- In the real-world, during subject's daily lives

- At population scale
- With low administrative and infrastructure cost versus current best techniques
- With competitive accuracy versus current best techniques

This paper introduces the Quantac technical stack developed for such a system (sensors hardware, electrical, mechanical, and software), the data captured with the stack (physiology, environmental, TAC, self-reported drink events), and ultimately explores machine learning applications to detect drink consumption.

II. Alcohol Data Collection and Reporting

We developed and used an end-to-end custom platform for collecting, processing, and analyzing transdermal alcohol data, Figure 1.

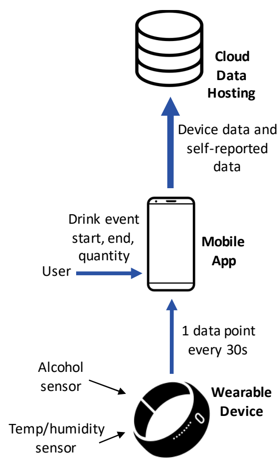


Figure 1 High-level flow diagram representing Quantac's technology stack. The wearable device samples data from alcohol, temperature, and relative humidity sensors, then aggregates, packages, and sends this data to a mobile app. The mobile app then relays device data to a cloud database for persistance and analysis. The mobile app also provides a user interface to collect drink event data from a user when testing a wearable device and similarly relays this data to the cloud once reported.

A. Custom Transdermal Alcohol Sensor Design

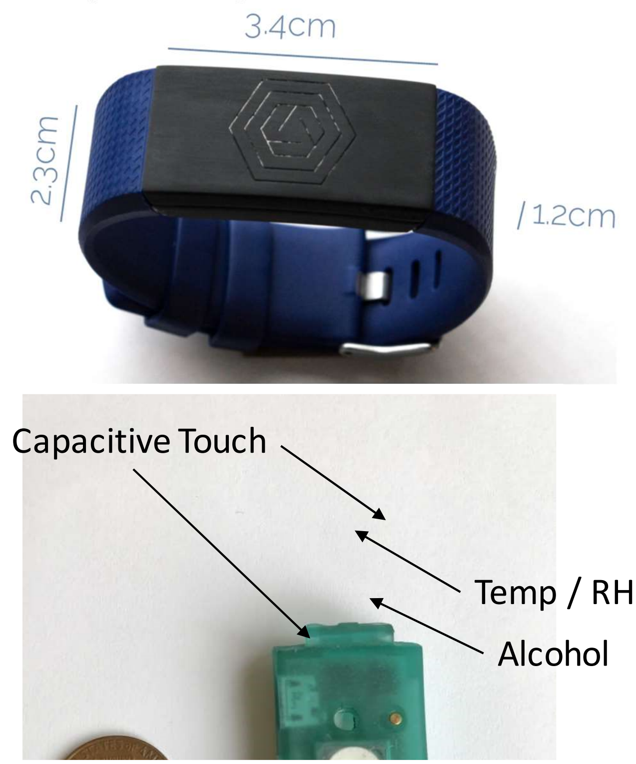
Quantac has developed a custom alcohol biosensor. It is a two-lead amperometric cell that sits on a user's skin and detects transdermal alcohol content (TAC). The sensor package is 8.7mm by 8.7 mm wide by 3mm tall. The sensing orifice is protected by a polyimide membrane that rejects moisture and debris while allowing gaseous ethanol to pass through.

Raw output is reported from Quantac's sensor in custom units "full scale response percentage." This is a value from -50% to 100% and represents the maximum deflection capable of the underlying alcohol sensing circuitry. Ideally the operating values of the sensor should be from 0 to 100% with 0% representing no alcohol present and 100% representing the maximum threshold of alcohol the sensor system can detect. However, in part due to baseline drift and in part due to manufacturing differences between sensors and devices[1], baseline is not always maintained at exactly 0%.

B. Wearable Design, Auxiliary Sensors, and Sampling Behavior

All sensor data was collected from Quantac prototype devices, Figure 2. A primary hypothesis of Quantac is that drink prediction can be improved by including data from auxiliary sensors. In addition to sampling data from the custom alcohol sensor, the device collects data from temperature and relative humidity sensors[2], as well as a pair of capacitive touch sensors to help determine if the device was being worn[3]. Limited data processing is performed on the device itself; the device acts as a data relay, sampling from the device's sensors and storing this data, then opportunistically transmitting the data to a paired mobile device via Bluetooth Low Energy (BLE). For the data collected and analyzed for this paper, the device records a data point for upload every 30s[4].

The prototype is not designed to meet IP67 water resistance standards although it is splash resistant. Consequently, all device users were advised to remove their devices when showering or swimming.



[1] Methods to calibrate sensor output given manufacturing differences and/or baseline drift are not investigated further here. Anomalous negative values have also been observed that are often correlated with environmental shock, e.g. a sharp change in relative humidity; these are also not investigated further.

[2] For collecting temperature and relative humidity the Si7020-A20 I2C humidity and temperature sensor is used. The datasheet can be found at https://www.silabs.com/documents/public/data-sheets/Si7020-A20.pdf.

[3] Due to limitations in the prototype design and reliability of the capacitive touch sensors, data from these sensors is not discussed in further detail.

[4] Internally the device samples from each sensor every 3s and averages every 10 samples resulting in one data point every 30s. Both the sample and aggregation rate are software configurable.

Figure 2 Quantac prototype device. All hardware components are contained within a small acrylic puck which provides contacts at each end to attach a wearable band.

C. Drink Self-Report Mobile Application

Drink events detailing amount of alcohol consumed (in std. drink units), drink consumption start time, and drink consumption completion time are self-reported by the device wearer via Quantac's mobile application. (Figure 3) The application is supported for iOS and allows a smartphone to connect to the device, manages user authentication, enables notifications, and transmits device and user data to the cloud-backend. Physiological information (gender, height, weight) is collected for each user upon sign-up.

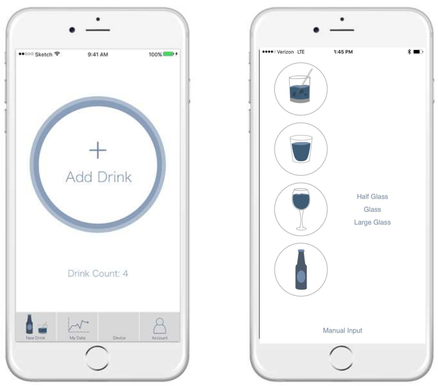
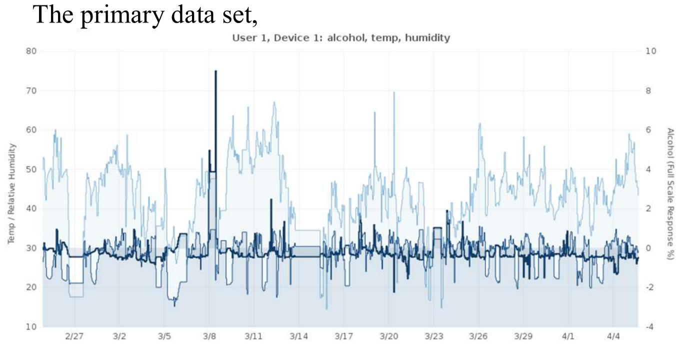


Figure 3 The "New Drink" page prompts users to manually input their drinks. Preset drink types (mixed drink, liquor, wine, beer) and amounts (single, double, half glass, glass, large glass, 12oz, 16oz, 22oz) appear after a user hits the "Add Drink" button to easily enter information, or the user can choose "Manual Input" to enter the specific alcohol percentage, ounces, and description of his or her drink.

D. Data Selection and Characterization

The amount of alcohol consumed, when and where a subject would drink and under what environmental conditions were not controlled. In gathering test data, precision was traded for convenience where a mobile app was developed so the user could self-report his or her drinking rather than use a breathalyzer which is more invasive to use, especially on a spontaneous day-to-day basis. All data presented here was collected on individuals who self-identified as being in good health, with moderate to heavy drinking history.

The primary data set,



	mean	min	max	stddev
Alcohol	-0.24	-4.74	9.00	0.57
Humidity	44.12	13.98	73.80	8.16
Temperature	28.62	14.97	35.80	4.17

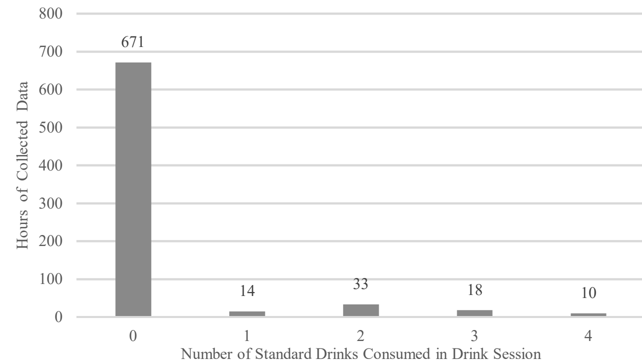
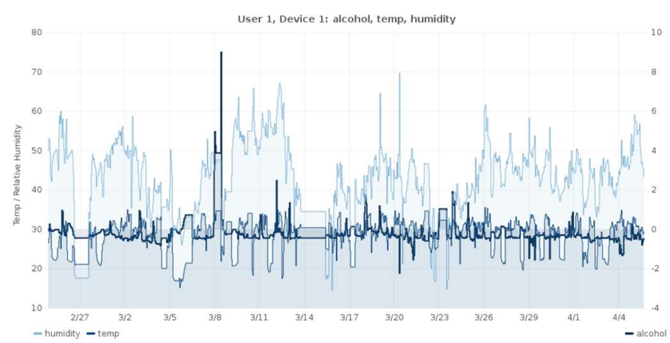


Figure 4 is from a single user and device. However, a larger data set is also provided and includes data across multiple users, devices with the former data set as a subset, Figure 5. Physiology of each user is detailed in Table 1. All data was collected between February and April of 2017 primarily in New York, New York.

User ID	Height (in.)	Weight (lbs.)	gender
user1	74	170	Male
user2	68	140	Female
user3	74	172	Male

Table 1 User physiology information.



	mean	min	max	stddev
Alcohol	-0.24	-4.74	9.00	0.57
Humidity	44.12	13.98	73.80	8.16
Temperature	28.62	14.97	35.80	4.17

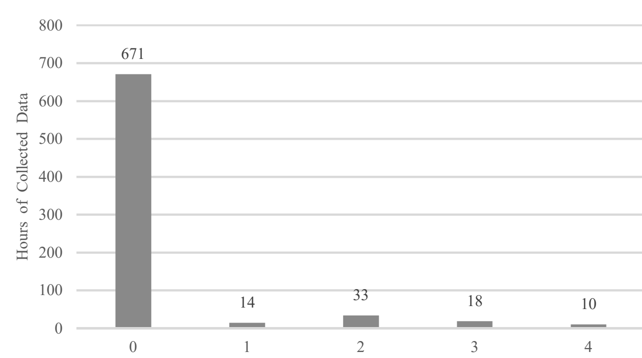
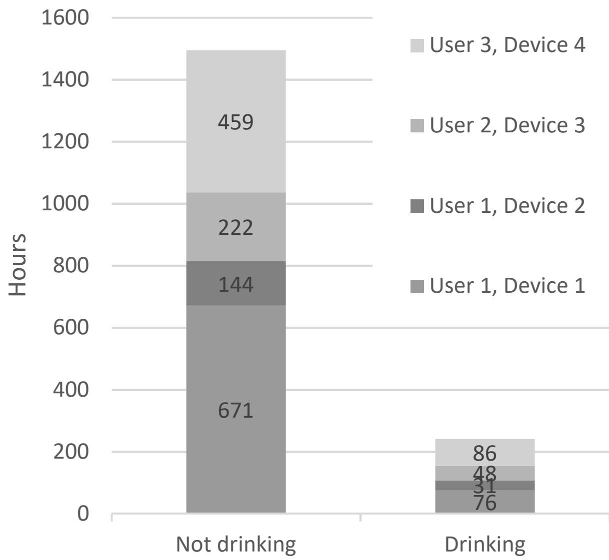


Figure 4 Summary of primary data set from User1, Device1. (Top) Raw data plotting full scale response percentage of alcohol series against relative humidity and temperature series. (Center) Summary features of series'. (Bottom) Time recorded drinking versus not drinking, and the amount of alcohol consumed (in rounded standard drink units) per drink session.



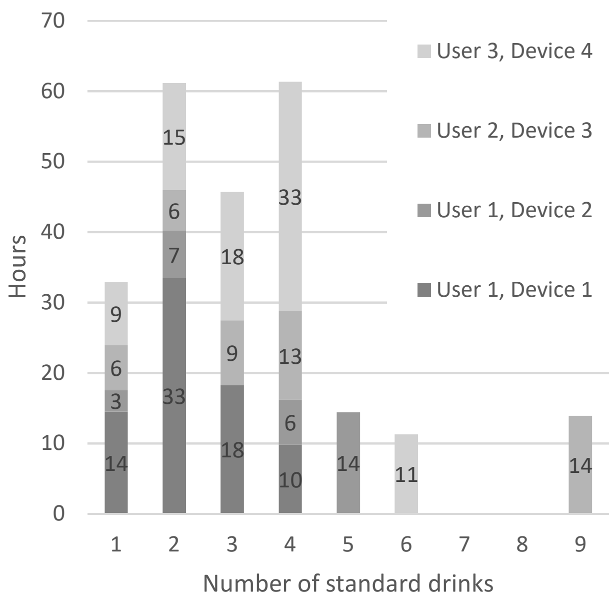


Figure 5 Summary of aggregate data set that includes primary data set from *Figure 4*. (Top) Time recorded drinking versus not drinking in hours. (Bottom) Time recorded drinking given the rounded number of std. drinks in a drink session. Most data is collected from drink sessions of 2, 3, or 4 drinks.

Hours drinking at a given number of standard drinks are defined by the total number of drinks consumed within a drinking session. A drinking session is defined by the time from when a user starts drinking until when the user is completely sober again. This time window is approximated based on the number of std. drinks reported by the user and a simplified version of Widmark's equation [6] which accounts for a user's gender and weight:

$$t_{drink\_session\_hours} = \frac{3.084 * n}{W * r * .015}$$

Where n is the amount of alcohol consumed in standard drink units, W is the weight in lbs., and r is the gender coefficient (.73 if male and .66 if female).

This computed window is similarly used to approximate output labels for drinking classification, Figure 6. For the binary classification case, labels with timestamps within a drinking session window are assigned a value of '1' and all others are assigned a value of '0'. Otherwise, drinking session labels are assigned instead a value corresponding to the number of std. drinks for that session[5].

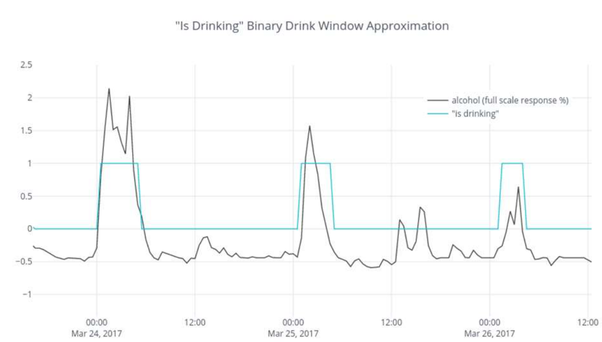
A standard scalar preprocessing is applied only over the input feature vector. This step contributes to performance of different machine learning algorithms allowing them to converge on optimal solutions faster. It also helps to visualize the data and clustering analysis following dimensional reduction.

Finally, windowing is employed around the neighborhood of each input observation to create an expanded feature set that includes future and/or historical observations as additional features, Figure 7. Overlap of windows between observations is maximized and is equal to:

$$\frac{num\_observations\_in\_window - 1}{num\_observations\_in\_window}$$

Downsampling of the original data, window size (number of observations per window), and offset to shift how many auxiliary future versus historical observations given the window size are investigated as input series parameters.

Labels are joined with the respective features only after feature sets are fully preprocessed. Both the features and labels are dropped at timestamps where device data is missing.



[5] Alternate windowing techniques e.g. using quantized steps instead of a simple square window are not discussed in this paper. In an absence of more precise training data (e.g. from a breathalyzer), this could be an interesting topic for future investigation.

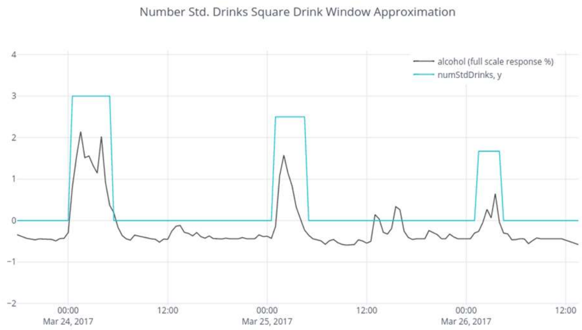


Figure 6 Example fabricated drink window training output. (Top) Binary window describes when user is drinking versus not drinking. (Bottom) A square window is constructed over the estimated duration of the drinking session with magnitude equal to the total of the number of drinks consumed in std. drink units.

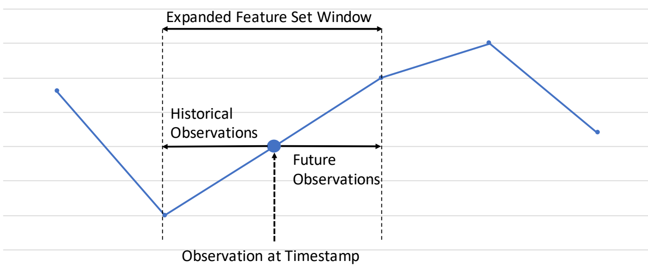


Figure 7 Example time series expanded feature window. In this example the feature space of the input series is tripled by including 1 future and 1 historical data point at each observation.

An out-of-box 2-dimensional PCA [7, 8] was used as an initial means to gain intuition for data clustering after processing via downsampling and expanding the time window feature set, Figure 8. Principal Component Analysis (PCA) is an unsupervised linear transformation technique that is used for dimensionality reduction as well exploratory data analysis. [9] PCA helps to identify patterns based on the correlation between features [9] and is a common technique for finding patterns in data of high dimension [8]. Figure 8 shows the PCA output from 0 to 4 standard drinks for user 1 and device 1. Observations for the higher standard drink values cluster upwards as the number of standard drinks grows, although for smaller values of std. drinks there is more overlap with the observations of 0 drinks. Intuitively this means that if we train a model with this data, observations with a higher number of standard drinks should be easier to classify.

PCA output from all users and devices shows more overlap at all values of standard drinks; this is potentially caused by different sensor, device and/or user calibrations, and intuitively means the data may be harder to classify generally than in the single user, device case.

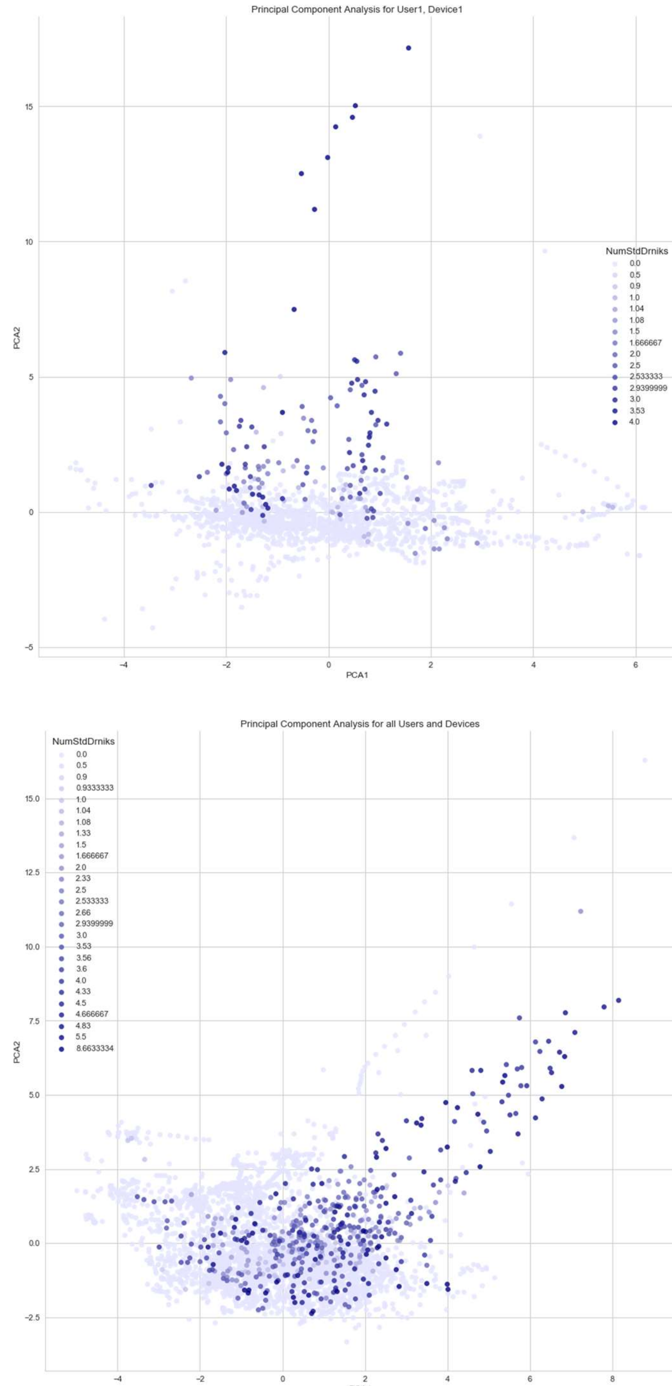


Figure 8 2-dimensional PCA of multi-dimensional input data including alcohol, humidity, temperature, series, 1 future data point and 1 historical data point with each observation, and a downsampling period of 30 minutes. PCA was run on data from User1, Device1 (Top) and run on data from all users and devices (Bottom).

III. Classification and Regression Modelling

While machine learning techniques have been employed at large to classify activity data [10], this is the first example we know of using a similar approach to classify drink behavior from alcohol data, Figure 9.

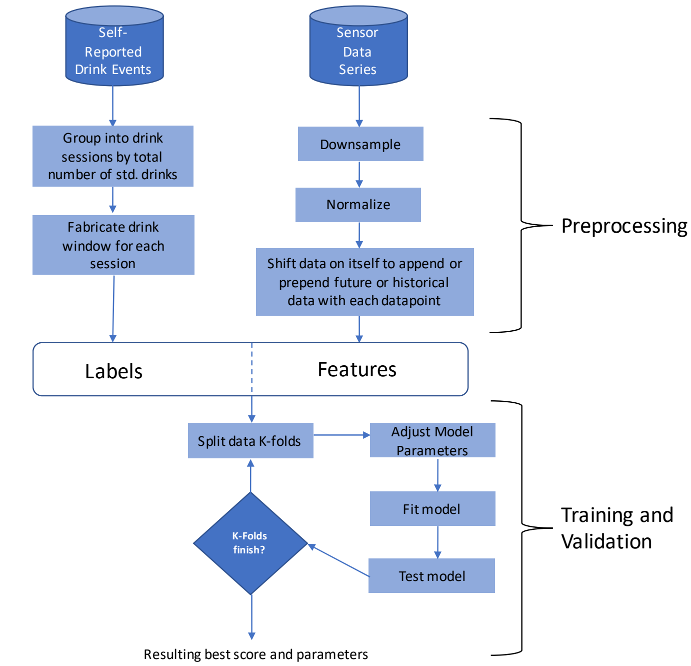


Figure 9 Flow-chart diagram describing end-to-end process for preparing, testing, and validating drink characterization models. Best model parameters are found via iterative fitting and testing under K-Fold cross-validation with 4 folds.

A. Evaluating Drink Prediction

i. F1 Score, Mean Square Error (MSE)

F1 score can be interpreted as a weighted average of precision and recall [11]. F1 ignores true negatives which is desired in this setup since most data is collected when the user isn't drinking. F1 score is best suited for the binary classification case; for classifying quantized number of standard drinks however, it is desired to weigh how close a prediction is to the output class, not just if a prediction matches the test output. Through this process, we modified our approach to instead investigate regression models and validate using MSE when trying to quantify the amount of alcohol consumed. The MSE of an estimator (of a procedure for estimating an unobserved quantity) measures the average of the squares of the errors or deviations—that is, the difference between the estimator and what is estimated [12]. MSE is useful to for comparing different regression models and for tuning their parameters via grid search and cross-validation [9].

ii. Cross-Validation Considerations

Both K-fold and stratified K-fold methods of cross-validation are investigated. In the absence of a large training set, the benefit of stratified K-fold is that the model is given a larger, more random sampling of drink session curves to use for training. However, the benefit of using (not stratified) K-fold is that it's tests how well the model can characterize a curve from a drink session it hasn't seen at all before and helps avoid the risk that a model might over-train on micro-features of each drink curve rather than features that represent generalizable characteristics across drink sessions.

B. Random Forest

Random forests have gained huge popularity in applications of machine learning during the last decade due to their good classification performance, scalability, and ease to use. Additional advantages of random forests are that they are less sensitive to outliers in the dataset and don't require much parameter tuning. [9]

The difference between random forest classification and regression is that it regression uses the MSE criterion to grow the individual decision trees, and the predicted target variable is calculated as the average prediction over all decision trees. [9] Using a random forest, we can measure feature importance as the averaged impurity decrease computed from all decision trees in the forest. Conveniently, the random forest implementation in scikit-learn already collects feature importance for us so that we can access them via the feature_importances_ attribute after fitting the random forest algorithm. [9]

To find the optimal depth of the decision trees and additional hyper parameters in the random forest scikit-learn API we apply the process described in Figure 9 using a grid search tool provided also by scikit-learn. To avoid overfitting we set an optimal limit for the maximal depth of the decision trees. [9] Grid search uses a brute force executive search over a list of different defined hyper parameters. Model performance for a given metric (MSE or F1) for each combination of hyper parameters is evaluated to obtain the optimal set. This way we maximize the generalization of our model to perform well against distinct sets of user and device data.

IV. Results

A. Feature Selection

A critical question in designing Quantac prototypes was whether capturing additional environmental data could be used to improve drink prediction; Figure 10 demonstrates prediction results from random forest binary classification using different combinations of input data series. Including the temperature series improves prediction against alcohol itself, while including the relative humidity series appears to have only a marginal effect on the prediction score. As expected, excluding the alcohol series results in poor prediction results. Investigation of feature selection for a random forest regression model reveals similar insight, Figure 12.

Figure 11 demonstrates improvements in prediction by including future and historical data with each input observation. Prediction is best when both future and historical data are included and worst when neither is included. Including only future data also yields greater improvement over including only historical data; our intuition is that this influenced by the time delay from transdermal response from when a user starts drinking and that our methodology does not account for this delay when fabricating test output windows.

One perhaps unexpected result was that a longer downsampling period resulted in higher prediction score. Our best intuition currently albeit informal is that this is somewhat dependent on the way random forest learns; extra observations e.g. every 5 minutes versus 30 minutes may not be more helpful as individual features become less distinguishing.

Additionally, downsampling and predicting more frequently doesn't allow for as much smoothing of noisy and boundary areas which could result in a lower aggregate prediction score.

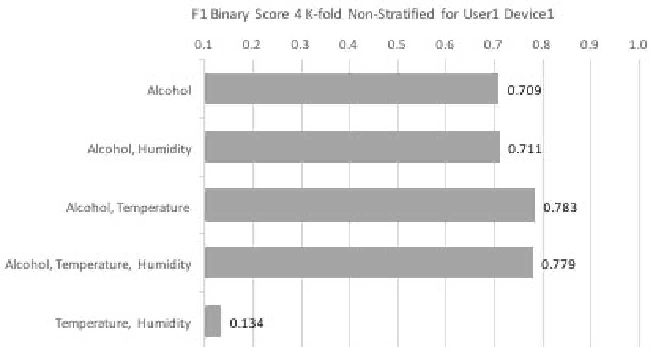


Figure 10 F1 score comparison for random forest binary classification including different combinations of sensor data series as input. Runs were evaluated using an input data set with a downsampling period of 30 minutes and including 1 future data point and 1 historical data point with each observation.

downsampling period (min)	include 30min future data	include 30min historical data	Total # Samples	F1 score
30	TRUE	TRUE	3	0.787
30	TRUE	FALSE	2	0.746
30	FALSE	TRUE	2	0.739
30	FALSE	FALSE	1	0.700
15	TRUE	TRUE	5	0.761
15	TRUE	FALSE	3	0.729
15	FALSE	TRUE	3	0.703
15	FALSE	FALSE	1	0.673
5	TRUE	TRUE	13	0.762
5	TRUE	FALSE	7	0.730
5	FALSE	TRUE	7	0.669
5	FALSE	FALSE	1	0.649

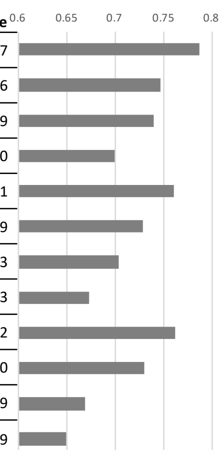
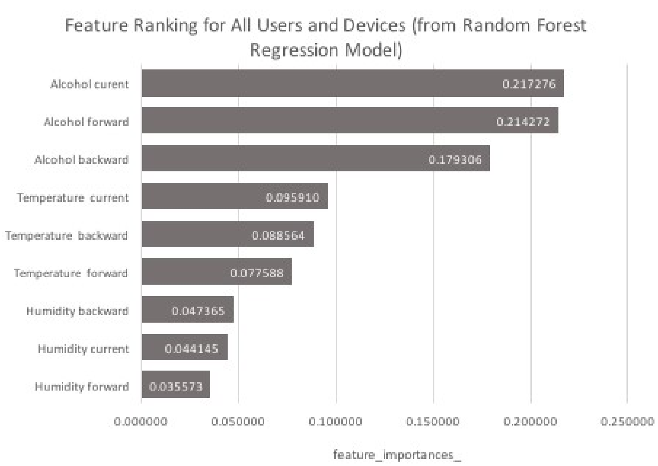


Figure 11 F1 score comparison for random forest binary classification with different time series feature configurations. Runs were evaluated including data series from alcohol, temperature, and humidity.



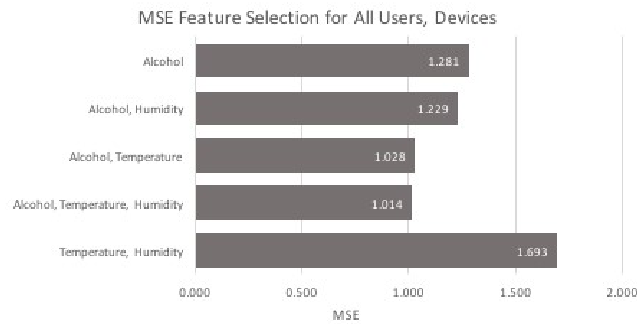


Figure 12 (Top) Feature importance *[13]* comparison between input features. A random forest regression model was trained using an input data set with a downsampling period of 30 minutes and including 1 future (forward) data point and 1 historical (backward) data point with each observation. (Bottom) MSE comparison for random forest regression including different combinations of sensor data series as input.

B. *Model Validation*

Figure 13 demonstrates the results of a single run random forest regression on data from User 1, Device 1; Figure 14 demonstrates the same across all users, devices. These provide illustrative examples for how well model prediction compares to the fabricated square windows of the number of standard drinks used for training. Regression output is also used to classify the binary scenario when a user is drinking versus not drinking by applying a threshold on the predicted output. A confusion matrix is generated for each binary projected prediction output of each data set.

Table 2 shows the MSE for 4-fold cross-validation on both data sets. Stratified performs better than non-stratified in both cases, likely because there is more balanced representation of each drinking amount for each device/user pair. In addition to the fact that there is more variability in the larger data set between users/devices, another reason for the higher MSE could be that there are a few high drink number session outliers in the larger set which do not have as much training data as the lower drink number sessions leading to underestimation of the number of drinks by the model. This is particularly obvious for example in the ~9 std. drinks curve in Figure 14.

K-Fold Method	MSE	
	User1, Device1	All Users, Devices
Non-Stratified 4 folds	0.244	1.014
Stratified 4 folds	0.192	0.885

Table 2 Comparison of MSE for random forest regression using stratified versus non-stratified k-fold cross-validation.

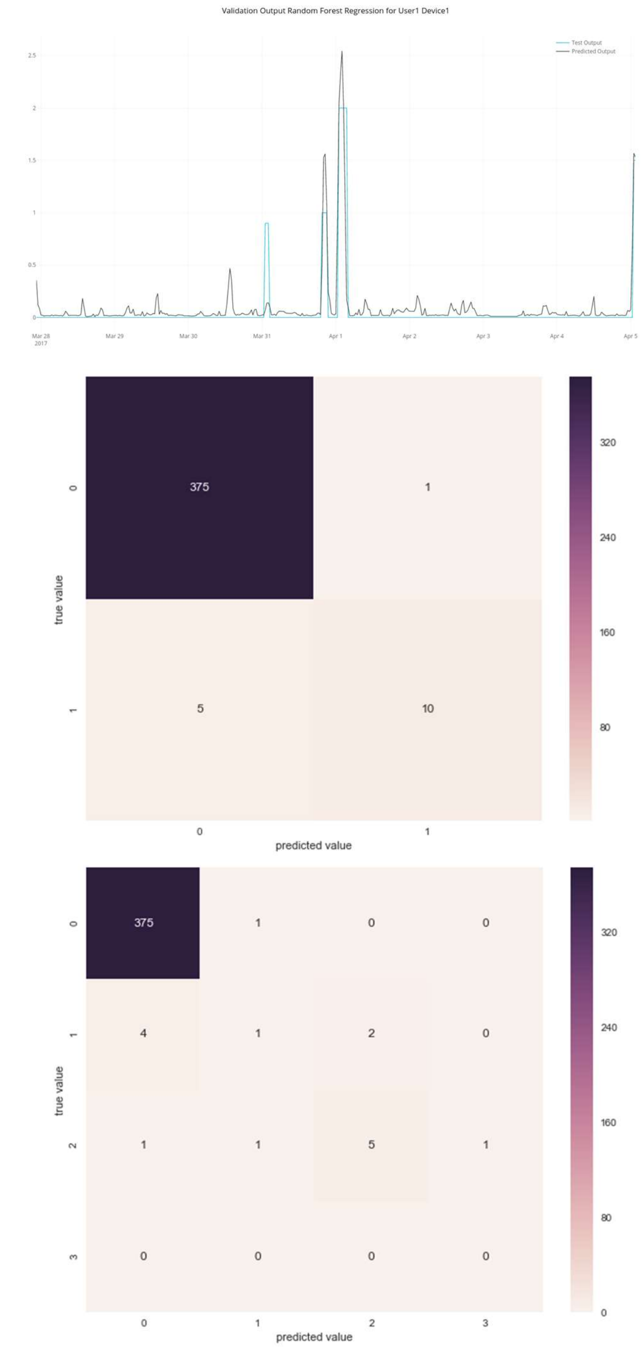


Figure 13 Random forest regression model performance from a run using data from User1, Device1, 75% training / 25% test data split. (Top) Time plot of test output versus output predicted by model, MSE=0.03. (Center) Confusion matrix using a simple threshold at .75 to determine whether user is drinking, F1=0.77. (Bottom) Confusion matrix rounding predicted output and training output.

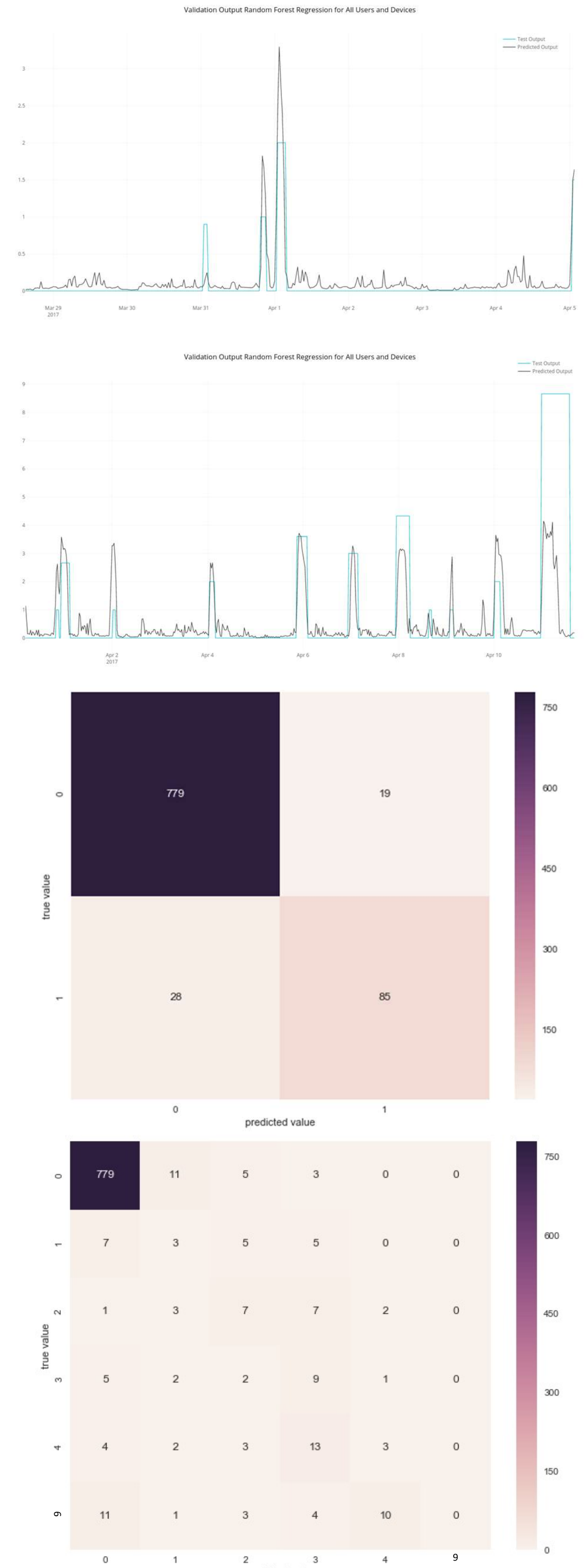


Figure 14 Random forest regression model performance from a run using data from all users, 75% training / 25% test data split, MSE=1.80. (Top) Time plot of test ouptut versus output predicted by

model for subset of test data for User1, Device 1 (Top-Center) Time plot of test ouptut versus output predicted by model for subset of test data for User 3, Device 4 (Bottom-Center) Confusion matrix using a simple threshold at .75 to determine whether user is drinking, F1=0.78 (Bottom) Confusion matrix rounding predicted output and training output.

V. Discussion

Quantac since inception has been iterating rapidly, trading consistency and stability for growth and learning; many data collection or quality issues are from non-functioning or malfunctioning sensors, faulty electronics, or firmware bugs that would not be representative of a commercial product. Nonetheless, one important limitation in this study is the selection bias in choosing a subset from data collected across multiple users and devices through months of stop-and-go testing. At best, this paper represents the performance of a stable, well-functioning Quantac platform. At worst, this paper shows only positive-biased results and misses some of the meaningful failure modes inherent to using this technology to instrument alcohol consumption.

Limitations in the drink reporting data include:

- Drink start (and end) time reported by each user may not be precise; the worst scenario is when a user forgets to report at the time of starting/finishing a drink.
- The pattern in which people consume a drink is not considered; however, this resolution of detail is not a significant limitation at this stage.
- While the user-interface was designed to minimize effort to self-report drinking, the reported amount of alcohol in each drink may be inexact. Frequently users are unsure of the alcohol percentage or volume of a drink which lead to compounded error as a user continues drinking. Best case is a drinking alcohol from a can or bottle with known alcohol content; worst cases include being served a glass of unknown quantity or concentration of alcohol or refilling a glass (e.g. a server refilling wine at dinner) before a drink is completed properly.

Despite these limitations, we are still able to train learning models from this data then predict drink output reasonably well against this data. Future investigations to collect a larger body of data as well as targeted studies to collect more precise data will be helpful to further improve and validate prediction.

While we considered a few different patterns for fabricating drink windows from drink events, we chose the square window approach as it is the simplest to use and understand and the prediction results were comparable to other windowing approaches (at least given our data set). To assess and/or avoid bias in fabricating training output an alternate approach could be to use interpolated BAC data in addition to the std. drinks window as training output. Were BAC data available and used as training output a similar approach to that described in this paper could also be used to directly predict BAC from Quantac device data.

Not addressed directly here is the real-time capability of a random forest learning model to determine if or how a user is drinking. Using the parameter set most commonly employed in this paper, a downsampling period of 30 minutes and including a future data point 30 minutes in the future with each observation, introduces a lag of 60 minutes to determine if or how much a user is drinking. This might be good enough already for some real-time systems. For less lag, a shorter downsampling period or less future data could be tuned trading for model accuracy.

VI. Conclusion

In summary, we demonstrated that out-of-box machine learning tools (random forest classification and regression) can be effective for predicting if a person is drinking and how much a person has had to drink given our device output series. Our initial findings indicate that alcohol and temp (humidity makes marginal difference) is the best combination of features, outperforming just alcohol. Moreover, using time series feature engineering can further improve prediction when we can afford to increase downsampling period and include additional data, especially future data. The prediction error was higher modeling from and predicting against data from all users, devices, but could also be due to fact that distribution of data in augmented set different from that of set from only one user device. Future work will include collecting more data under more consistent testing conditions.

VII. Acknowledgements

1) Eric Porges and SHARC group for their help and confidence as early partners and for assistance in editing this paper.
2) Daniel Fridberg and Andrea King at the University of Chicago for their support as early partners.
3) Juliette Kang for the design of the Quantac logo and mobile application.
4) The Techstars Connection, in Partnership with AB InBev program, as well as the many mentors including Eamonn Carey, Jon Peachey, Jane Tran, and Kyle Jackson who have been so generous with their time and help.
5) Wolfram Schulte, for his unconditional support of Quantac; we can't say thank you enough.

References

[1] J. McCambridge and K. Kypri, "Can Simply Answering Research Questions Change Behaviour? Systematic Review and Meta Analyses of Brief Alcohol Intervention Trials," vol. 6, no. 10, p. e23748, 5 Oct 2011.

[2] L. C. Sobell and M. B. Sobell, "Timeline Follow-Back In Measuring Alcohol Consumption," *Humana Press,* pp. 41-72, 1992.

[3] M. A. Dumett, G. Rosen, J. Sabat, A. Shaman, L. Tempelman, C. Wang and R. M. Swift, "Deconvolving an Estimate of Breath Measured Blood Alcohol Concentration from Biosensor Collected Transdermal Ethanol Data," *Appl Math Comput.,* vol. 196, no. 2, p. 724–743, 1 March 2008.

[4] P. Marques and S. McKnight, "Evaluation of Transdermal Alcohol Devices," 2005.

[5] N. Hill-Kapturczak, J. D. Roache, Y. Liang, T. E. Karns, S. E. Cates and D. M. Dougherty, "Accounting for sex-related differences in the estimation of breath alcohol concentrations using transdermal alcohol monitoring," *Psychopharmacology,* vol. 232, p. 115–123, 2015.

[6] W. C. Head, "BAC Formula Quick Check," 2015. [Online]. Available: http://www.teamdui.com/bac-widmarks-formula/. [Accessed 20 Oct 2017].

[7] scikit-learn developers, "sklearn.decomposition.PCA," 2007-2017. [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html. [Accessed 21 Oct 2017].

[8] L. I. Smith, "A Tutorial on Principal Components Analysis," 26 Feb 2002. [Online]. Available: http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf. [Accessed 17 23 2017].

[9] S. Raschka, Python Machine Learning, Packt Publishing, 2015.

[10] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou and Y. Amirat,

"Physical Human Activity Recognition Using Wearable Sensors," *Sensors,* vol. 15, no. 12, p. 31314–31338, Dec 2015.

[11] scikit-learn developers, "sklearn.metrics.f1_score," 2007-2017. [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html#sklearn.metrics.f1_score . [Accessed 21 Oct 2017].

[12] "Mean Squared Error," 28 Sept 2017. [Online]. Available: https://en.wikipedia.org/wiki/Mean_squared_error. [Accessed 23 Oct 2017].

[13] scikit-learn developers, "Feature importances with forests of trees," 2007-2017. [Online]. Available: http://scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html. [Accessed 21 Oct 2017].