University of Amsterdam

Bachelor thesis

Detecting heavy drinking episodes with ELM

Author:

Nick Wortel

Student ID's:

11688866

Supervisor:

Samaneh Khoshrou

June 27, 2021

Universiteit van Amsterdam

Abstract

Heavy drinking causes many kinds of disease and even death it is especially prevalent among young adults. This paper aims to make a model that is fast and precise in detecting heavy drinking episodes, such that there is time to intervene with for example an app on your phone. To achieve an improved model, this paper builds on the work of Killian et al. (2019) in the effort to predict heavy drinking episodes by accelerometer data using machine learning. The established random forest method can give high accuracy but is extremely slow. To improve on this, a new method is used that is named extreme learning machine. This method has already shown its strength in many stylized situations especially in speed. This paper search for the optimal subset based on different feature selection methods and the influence of different activation functions. It shows that ELM performs marginally better in accuracy but is extremely fast compared to random forest.

Statement of Originality This document is written by Student Nick Wortel who declares to take full responsibility for the contents of this document.

I declare that the text and the work presented in this document are original and that no sources other than those mentioned in the text and its references have been used in creating it.

The Faculty of Economics and Business is responsible solely for the supervision of completion of the work, not for the contents.

1 Introduction

Heavy drinking is the cause of more than 5% of the deaths worldwide in 2016 (WHO, 2018). It can also lead to all kinds of chronicle ailments including liver disease, cardiovascular disease and cancer to just name a few (Gutierrez, 2016). Heavy alcohol consumption is still prevalent especially on college campuses (SAMHSA, 2015).

To avoid this unnecessary health risk, several researchers tried to detect heavy drinking episodes by students. Such that an app can be developed to detect if the user is inebriated and could then for example call a cab or notify a health professional. Such an application would be able to save lives not only of the alcohol consumers but also reduce the mortality of drunken drivers.

To develop such an application it is necessary to gather reliable and large datasets. Due to the rise of smartphones, it has become possible to generate those datasets. With the field of machine learning and econometrics rapidly evolving, new methods arise quickly, which leads to possible new improvements on the model in performance or speed. So we strive closer to a model that can detect intoxication in real-time.

Extreme learning machine is a new method of machine learning that is efficient and extremely fast because it does not use normal gradient descent. The question this paper will answer is: can an extreme learning machine improve the established method of random forest in accuracy and speed? And does the activation function and feature selection influence the performance? This opens the possibility of using real-time data. The activation functions are crucial in machine learning as they transform the data in the model. Feature selection is an interesting factor because it can cause or counter

overfitting but also influences the speed of the model fitting.

The next part will look at prior results in the form of a literature review and will lay the groundwork of the theory. After that, the methodology where the data and the transformation that are done to the data of this paper are explained, as well as, the feature extraction and the models. Then the results where all the performance of the models are stated and explained. Followed by the discussion and future research this will look critically at the results and talk about future improvements. Finally, there is the conclusion with a summary of all findings of this paper.

2 Literature review

This literature review will first compare five papers on machine learning and heavy drinking episodes. This will be done in a parallel manner because the papers have large similarities in data gathering and preprocessing. But differ in feature selection and modeling. After that, there will be an explanation of the extreme learning machine (ELM), its origin and why it is applicable for this paper.

Most often surveys are used to sample the amount of alcohol consumed (Bae et al., 2017, Arnold, LaRose, and Agu, 2015, Phan et al., 2020, Gharani et al., 2017). This is not the most reliable source, as Bae et al. (2017) noted that most heavy drinkers underestimate their consumption when they are surveyed afterward. The interesting differences between the papers are in what other data they collected and the methods used. For example, Gharani et al. (2017) and Arnold, LaRose, and Agu (2015) only used the location with

the accelerometer, while Bae et al. (2017) and Phan et al. (2020) also used the length of the text input, device usage, keystroke speed and luminosity to name a few. Contrary Killian et al. (2019) used an ankle bracelet sensor named SCRAM that calculates the alcohol consumption and only used the accelerometer as another input. The advantage of measuring alcohol consumption by a SCRAM ankle bracelet instead of a survey is that there is no self-reporting bias. However Killian et al. (2019) only has thirteen usable participants.

All the aforementioned literature starts with basic pre-processing and then used the acquired data to generate features. Because the data is large and quite noisy, some papers smoothened the data to make it more usable (Arnold, LaRose, and Agu, 2015, Killian et al., 2019). All of them used the mean and standard deviation of all three axes of the accelerometer, as well as, other basic statistical values. Some generated more complex new features, such as the energy feature of Gharani et al. (2017) that calculates the squared length of the movement vector over a given time interval. Bae et al. (2017) used SMOTE, which oversamples the minorities, in the case of this paper drinking and heavy drinking. The paper by Kao et al. (2012) explains the methods of calculating gait stretch and step time based on accelerometer data. This kind of feature is used in all the aforementioned papers, it is useful because it quantifies movement.

Phan et al. (2020), Killian et al. (2019) and Arnold, LaRose, and Agu (2015) all used random forest and support vector machine and all found that random forest performed best with an accuracy of 0.76, 0.775 and 0.56, respectively. Bae et al. (2017) only used random forest and achieved an accu-

racy of 0.966. Gharani et al. (2017) tried three different methods of training. The first was, Bayesian regularization, this method uses the Bayesian approach which treats the weights as random variables and will be iteratively updated based on observations. The other methods used are the Levenberg-Marquardt algorithm and conjugate gradient to find the optimal weights. The conjugate gradient performed significantly worse with an R-value of 0.88 while the Levennberg-Marquardt and Bayesian regularization both had a comparable performance with R values of 0.994 and 0.998 respectively. Gharani et al. (2017) also tried three different regression methods: multilayer perceptron, support vector machine and linear regression. The multilayer perceptron outperformed the other two by a large margin (the respective correlation coefficients are 0.90, 0.39 and 0.44). So in general random forest has been proven an effective model that shows high performance over several different datasets, but newer alternatives such as the multilayer perceptron also show potential.

A new machine learning method that is getting more traction in recent times is the ELM. This was first introduced by Huang, Zhu, and Siew (2004) as a fast and accurate method on some stylized classification problems. ELM has shown its strength in many different cases for stylized data. Such as in the paper by De Campos Souza et al. (2020), where it outperformed more established methods such as random forest and support vector machine both in performance and speed. It is also used in a lot of different fields to make fast classification and accurate models for real-life problems. One example of this is the paper by Wang, Yang, and Kalivas (2020) where they tried to predict the level of octane released by the production of gasoline and get

R-squared over 0.93. Another example is the paper by Shariati et al. (2020) where they predicted the strength of cement. They compared some neural networks, support vector machine and ELM. The ELM-GWO an augmented ELM performed best overall.

To summarize machine learning is a vibrant and rapidly evolving field with ELM being one of the forefronts of this field. ELM is still mainly used in stylized classification problems and only a few times on real-life examples. This paper intends on showing that the ELM can also stand its ground in real-life classification problems.

3 Methodology

3.1 Dataset

In this paper, the same data is used as in Killian et al. (2019). This consists out of transdermal alcohol content (TAC), or in other words the measure of drunkenness, and phone accelerometer data. This data was acquired during a bar crawl of students. The TAC was registered by a SCRAM ankle bracelet, which estimates the intoxication based on the amount of alcohol in the participants' sweat. The TAC was noisy due to the inaccuracy of registering on the body of the participant so it was cleaned and shifted 45 minutes because of the delay of the alcohol absorption by the blood and later to the sweat. The TAC data has a mean of 0.065 and a standard deviation of 0.182. Participants were classified as drunk if their TAC was higher than 0.08. Accelerometer data is the acceleration of the phone and is given in three axes x, y and z and is registered with a frequency of 40 Hertz. There

are a total of over 14 million observations for the accelerometer data and 715 for the TAC spread over the thirteen participants.

3.2 Feature extraction

The accelerometer data still needs some pre-processing. Because of the high frequency of the data a Chebyshev type 2 filter is used, also known as inverse Chebyshev filter. This makes the data better to use and smoother because of its steep roll-off, especially at higher order.

Basic features

Then some basic transformations are done over all the axes of the accelerometer data and the data that was filtered using Chebyshev. These include: first difference, the absolute value, and square value of each dimension. Some features do not only use one axis but all three axes. First energy, this is the euclidean length of the three variables ($\sqrt{x^2 + y^2 + z^2}$) and second the product of the x, y and z axes.

Statistical features

After this, all basic features were used to calculate the following basic statistical properties for a sliding window equivalent to 1 second. These include: sum, minimum, maximum, mean, median, standard deviation, variance, skewness and kurtosis. So now there are 252 different features.

Frequency features

Most papers also use features from the frequency domain. These can also be useful to add because of the high frequency accelerometer data. So the following spectral features are also added for all basic features: centroid, spread, flux, entropy, roll-off, flatness and the zero-crossing rate. So the total size of the features is 644187 by 448.

3.3 Feature selection

It is important to select the optimal subset of features because too few features will lead to a small amount of information being used, but too many features lead to the curse of dimensionality and slow model fitting. To find the optimal selection three methods are used in this paper, namely: Principle Component Analysis (PCA), correlation and sequential forward floating selection (sffs).

Principle Component Analysis

PCA generates a smaller number of new features based on the original features such that the first feature of the PCA subset contains as much as possible variance from the other features. This is done using the following maximisation:

$$w_k = \underset{||w||=1}{\operatorname{argmax}} ||\hat{X}_k w||^2$$

$$\hat{X}_k = X - \sum_{s=1}^{k-1} X w_s w_s^T$$

This reduces the number of features, which can be useful in reducing the fitting time of the model, but has the disadvantage of losing some information.

Correlation

A fundamental way to look at the importance of a feature is to look at how much it is correlated to the target. This has been shown to be a powerful and easy method in the paper by Paskaleva et al. (2008). To do this the Pearson product-moment correlation coefficient for every feature. Then sort the absolute values and then select the k highest valued features.

Sequential forward floating selection

Sffs has shown its strength in many different situations for example in the paper of Olvera-López, Martínez-Trinidad, and Carrasco-Ochoa (2007). The sffs uses an algorithm to select a subspace of the k best features that works as follows.

Algorithm 1: sffs

```
1  selected ← ∅
2  while # features selected < k do
3      for x ∈ X do
4          fit and predict using X̂ ∪ x
5          select x with highest performance
6          selected = selected ∪ x
7      end
8      for x̂ ∈ selected do
9          fit and predict using selected\x̂
10         if lowest performing model is for x̂ ≠ x then
11             selected= selected\x̂
12         end
13     end
14 end
```

So to summarise this method it selects the best feature and adds it to the selected subset. Then checks if the model performs better excluding one of the earlier selected features, if this is the case the earlier added feature is removed from the selected subset. It repeats this process until it reaches the given number of features.

3.4 Models

Random forest

Random forest is chosen as a baseline due to its good performance in Killian et al. (2019). A random forest model is used consisting out of a large number of decision trees. A decision tree is a binary operator that decides what goes into the next activation function and then into the next decision tree. It is a random forest because it randomly selects variables and thresholds for the trees. To do this the sklearn package in Python is used.

ELM

A possible alternative to the random forest is the ELM. The ELM was introduced by Huang, Zhu, and Siew (2004), who showed that it was a fast and accurate method. This method consists of a single hidden layer feed-forward network (SLFN). An SLFN uses the initial weights w to multiply the features and put them in an activation function $\phi(\Sigma_{i=1}^{n} x_i w_{i,j}) = O_j$ and then does the same but with O instead of x and β instead of w. The values that this gives are the predicted target. The speed is achieved because ELM does not use gradient-based backpropagation but the Moore-Penrose generalization instead. De Campos Souza et al. (2020) built on this principle but robustly addressed the problem of overfitting. To achieve this they bootstrapped the data to artificially increase its size and then used Lasso. To find the optimal lambda in the Lasso equation they used a grid search and showed that this method outperforms all alternative ELM methods for a large group of stylized datasets.

Graphs of the performance for different number of hidden nodes

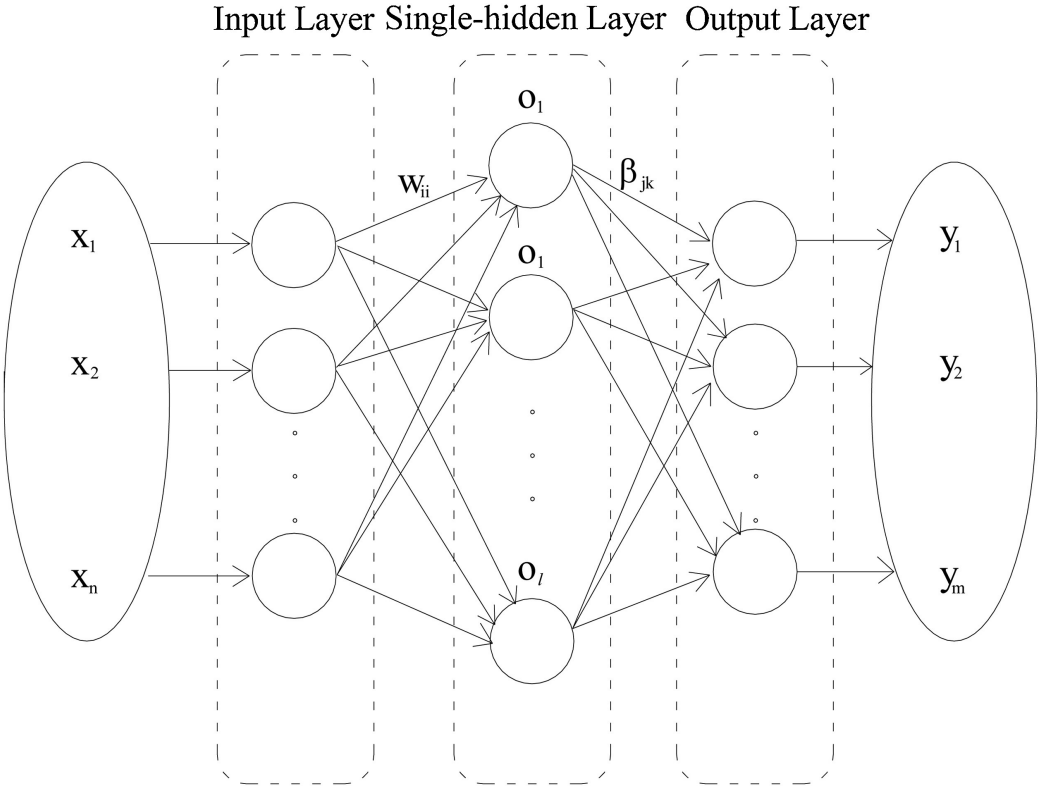


Figure 1: The topological model structure of ELM (Wang, Yang, & Kalivas, 2020)

The only disadvantage is that it is slower than the more basic ELM procedures. Another important decision for the ELM is which activation function is used. This paper will look at three common activation functions: ReLu, sine and sigmoid. The ReLu is a very simple activation function it is $\phi(x) = max(0, x)$. The sine function is also straight forward with $\phi(x) = sin(x)$. Finally the sigmoid is as follows $\phi(x) = \frac{1}{1+e^{-x}}$

3.5 evaluation measures

To compare the performances of the different models the Sklearn package is used the measures it gives are as follows.

Precision

$$\frac{TP}{TP + FP}$$

With TP being the true positive or in other words, the number of correctly predicted drunks and FP being the false positive or in other words the number of incorrectly predicted drunk.

Recall

$$\frac{TP}{TP + FN}$$

Very similar to the precision but now divide by the number of observed drunk because FN is false negative what means the number of observations that were drunk but were predicted as sober.

F1-score

$$2 * \frac{precision \ * \ recall}{precision \ + \ recall}$$

All these measures were calculated for drunk, sober, the macro and weighted average. The time it took to fit the model and predict are also calculated.

3.6 Experimental setup

To be able to validate the performance of the models later on the data is split into two groups the train set containing 70% of the data and the test set containing the rest. The optimal number of features, activation function and number of hidden nodes or trees will be estimated using a grid search. For this grid search a large list of possible values are selected then they are all tried and the values that perform best on the macro F1-score will be selected.

4 Results

All results are gathered with Jupyter notebook and can be found on Github[1]. The script is based on Gara (2020).

Feature selection

When we compare the first 75 features that are selected we find that correlation and sffs select quite different features. With only sixteen features from the selection being the same. It is impossible to compare this to PCA because it does not select features but transforms all of them into a new space. When we compare the time to run of the different methods PCA and correlation both only take thirty seconds for this data set to be calculated. The sffs based on a random forest with 30 trees for 75 features takes a hundred hours.

[1] https://github.com/NickW1999/ELM

Table 1: Number of selected features

	ELM ReLu	ELM sine	ELM sigmoid	Random forest
PCA	10	75	75	75
Correlation	10	80	90	75
sffs	75	75	75	75

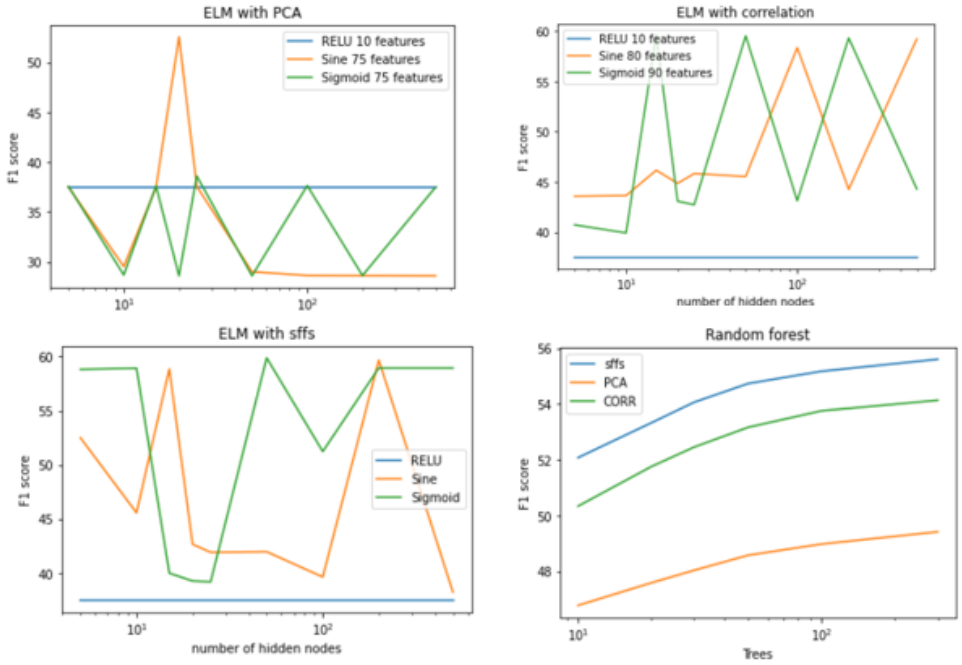
When we look what the optimal number of features are from each method of selection the grid search shows that it is similar with around 75 for all methods except for ReLu. Note that for the sffs always 75 features are selected.

Activation functions

In the first three graphs the number of features used is such that they include the highest-performing F1-score observation in the grid search. The first three graphs show how a given activation function performs for different numbers of hidden nodes in the ELM. They all have the blue line for ReLu the orange line for the sine and green for sigmoid. First of all, note that ReLu performs constantly poorly this can be caused by very fast convergence for only a few nodes. second of all note that the results look noisy this can be caused by the randomness of the starting values of the ELM. The optimal number of hidden nodes is not conclusive but has to be at least ten to be able to fit properly. Graph 4 looks at the number of trees in the random forest model and shows that an increasing number of trees leads to better performance but that the gains of adding new trees are decreasing for higher amounts. It is also optimal to select 75 features under all three feature selection methods. Sffs performs best for random forest followed closely by

the correlation and PCA is in last by a significant amount.

Graphs of the performance for different number of hidden nodes



accuracy

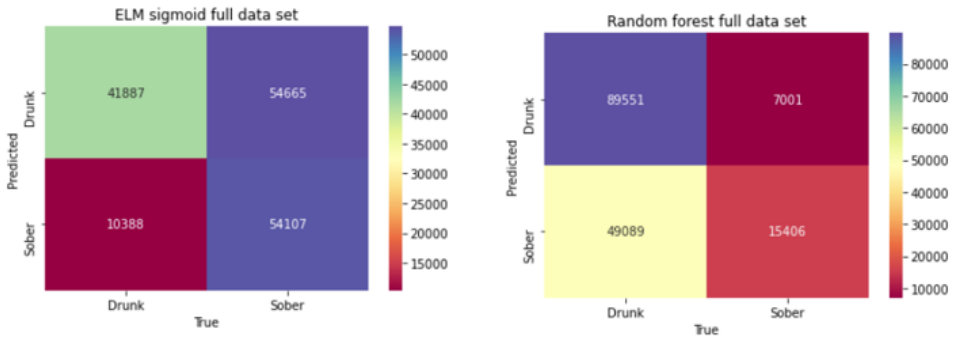
In this table the two best performing models are highlighted. Both are ELM models the best performing using a sigmoid as activation function and the whole data set, and the second using the sine and sffs. These perform marginally better than the best performing random forest models with an F1 score of four points more.

Best performing models

		ELM RELU	ELM sine	ELM Sigmoid	Random forest
PCA	**F1 score**	37.48	37.48	37.67	48.20
	Time	0.07	0.07	0.08	18.37
	#hidden nodes/trees	50.00	20.00	25.00	300.00
	#features	10.00	75.00	75.00	75.00
Correlation	**F1 score**	37.48	47.03	45.74	50.91
	Time	0.06	0.07	0.07	12.21
	#hidden nodes/trees	20.00	100.00	15.00	300.00
	#features	10.00	80.00	90.00	75.00
sffs	**F1 score**	37.48	59.48	38.97	55.61
	Time	0.07	0.07	0.07	12.86
	#hidden nodes/trees	100.00	200.00	50.00	300.00
	#features	75.00	75.00	75.00	75.00
full	**F1 score**	37.48	53.98	59.71	55.80
	Time	0.12	0.11	0.11	30.95
	#hidden nodes/trees	100.00	55.00	30.00	300.00

Table 1: An overview of the best perfroming models found by the grids search for all activation functions and feature selection methods (times is in minutes and full means the full data set of features is used).

Confusion matrices



However, it is important to note that the ELM has more false positives while the random forest has more false negatives as shown in the confusion matrices.

Time

The other important aspect of ELM is its speed to predict and fit the model. There is a clear difference between the random forest models that take between 12 and 31 minutes to fit and predict and the ELM model that only takes between 4 and 7 seconds. The difference between using the full data set or a subset of only 75 is also larger for random forest where it takes around twice as long, while for the ELM it is closer to one and a half times slower. The activation function does not show a clear influence on the speed of the model.

Summary

So when we compare all results we find that first of all the ELM with ReLu performs very poorly just as shown in the graphs the same goes for the feature selection method of PCA with all their F1-scores below forthy. All the best performing models have a score just under 60 and are both ELM one being with the sffs and sine as activation function and the second being including all features and using sigmoid as the activation function. When we look at the time it took to fit and predict for the models we also see that all ELM models outperform the random forest with the time of ELM being around six seconds and the random forest took on average around fifteen minutes. This shows that the hypothesis that ELM could be an improvement

is true. ELM performs better in both speed and F1-score.

5 Discussion

The results show that ELM has higher power to identify if someone is drunk or sober compared with the well-established random forest. In terms of speed, the ELM model is extremely fast in fitting and predicting, 150 times faster than random forest. One large disadvantage of ELM is that it can be noisy due to the randomness of the initial values. It is also clear that the activation function that is used influences the results with especially the ReLu severely decreasing the performance of the model, but the sigmoid and sine perform similarly. It is important to note that the feature selection method influences the model performance in the case of random forest significantly with the time it took halved and similar performance. However, for ELM the full feature space gives the best result and even though it takes 50% longer it is still extremely fast. In general, the best method in these results is sffs however the downside of this method is that it takes a long time to run so the assumption needs to be made that the features selected by sffs are universal and thus only need to be calculated once.

6 Conclusion

ELM has proven its usefulness in many stylized situations, but this paper shows that it also is applicable for real-life classification problems. The research question consists of several different parts these are can ELM out-

perform random forest in prediction, speed, with the right activation function and right feature selection method.

Even though the performance of Killian et al. (2019) is not reached, does the ELM outperform the random forest in performance. The F1 scores of ELM are 59.71 while the random forest has 55.80. However, the random forest has a higher power to identify drunk compared to the ELM, but has more false negatives.

ELM is especially effective in increasing the speed of fitting and predicting. It takes ELM around 6 seconds while it takes the random forest fifteen minutes. This can be done so fast that it will be possible to do this in real life.

The activation function has a large influence on the performance of the model with ReLu performing particularly poorly and the sine and sigmoid performing similarly and both well.

Feature selection is useful for both ELM and random forest because it reduces the time it takes to predict by one-third and a half respectively. However, this also leads to a small reduction in F1 performance in both cases around 0.2. PCA performs poorly as a feature selection method and sffs performs best. For ELM it might be best to use the full data set because it still only takes a short time.

Future direction

There are several possible solutions for the randomness of ELM that can be explored in future research. Due to the high speed, it is possible to generate more than one ELM prediction and take the average of those as

the final prediction. Another possible solution is as in Shariati et al. (2020) where another fast estimation model is used to generate the initial values of the ELM matrix. There is also the possibility proposed by Killian et al. (2019) that because the predictions are for such a short time frame (1 second) it is possible to take the average of 2 minutes and use this as the prediction.

So to conclude ELM has great potential for classification models with high precision and extremely high-speed modeling and fitting. This makes the possibility of a real-life app closer within reach. There is still room for improvement mainly for the randomness of the initial states.

References

Arnold, Z., D. LaRose, and E. Agu (2015). "Smartphone Inference of Alcohol Consumption Levels from Gait". In: *Healthcare Informatics*, pp. 417–426.

Bae, S. et al. (2017). "Detecting Drinking Episodes in Young Adults Using Smartphone-based Sensors". In: *PACM Interact. Mob. Wearable Ubiquitous Technol* 1.2.

De Campos Souza, P.V. et al. (2020). "An Advanced Pruning Method in the Architecture of Extreme Learning Machines Using L1-Regularization and Bootstrapping". In: *Electronics (Basel)* 9 (5), p. 811.

Gara, G.P. (2020). *ELM tutorial*. `https://github.com/glenngara/tutorials`.

Gharani, P. et al. (2017). "An artificial neural network for movement pattern analysis to estimate blood alcohol content level". In: *Sensors (Basel, Switzerland)* 17.12, pp. 2897–.

Gutierrez, Winston (2016). *Alcohol consumption : patterns, influences and health effects*. eng. Alcohol and Drug Abuse. New York: Nova Publishers. ISBN: 1-63485-546-9.

Huang, G., Q. Zhu, and C. Siew (2004). "Extreme learning machine: a new learning scheme of feedforward neural networks". In: *2004 IEEE International Joint Conference on Neural Networks* 2, pp. 985–990.

Kao, H. et al. (2012). "Phone-based gait analysis to detect alcohol usage". In: UbiComp '12, pp. 661–662.

Killian, J.A. et al. (2019). "Learning to Detect Heavy Drinking Episodes Using Smartphone Accelerometer Data". In: *Proceedings of the 4th International Workshop on Knowledge Discovery in Healthcare Data co-located*

with the 28th International Joint Conference on Artificial Intelligence, pp. 35–42.

Olvera-López, J. Arturo, J. Francisco Martínez-Trinidad, and J. Ariel Carrasco-Ochoa (2007). "Restricted Sequential Floating Search Applied to Object Selection". In: *Machine Learning and Data Mining in Pattern Recognition*. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 694–702. ISBN: 3540734988.

Paskaleva, B et al. (2008). "Canonical Correlation Feature Selection for Sensors With Overlapping Bands: Theory and Application". eng. In: *IEEE transactions on geoscience and remote sensing* 46.10, pp. 3346–3358. ISSN: 0196-2892.

Phan, T. et al. (2020). "Understanding Heavy Drinking at Night through Smartphone Sensing and Active Human Engagement". In: *PervasiveHealth*, pp. 211–222.

SAMHSA (2015). "national survey on drug use and health (nsduh)". In: DOI: `https://www.samhsa.gov/data/sites/default/files/NSDUH-DetTabs-2015/NSDUH-DetTabs-2015/NSDUH-DetTabs-2015.htm#tab6-84b`.

Shariati, Mahdi et al. (2020). *A novel hybrid extreme learning machine–grey wolf optimizer (ELM-GWO) model to predict compressive strength of concrete with partial replacements for cement*. NEW YORK.

Wang, Xiaoyu, Kan Yang, and John H Kalivas (2020). "Comparison of extreme learning machine models for gasoline octane number forecasting by near-infrared spectra analysis". In: *Optik (Stuttgart)* 200, pp. 163325–. ISSN: 0030-4026.

WHO (2018). “Global status report on alcohol and health”. In: *World Health Organization*.