2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
17-20 October, 2021. Melbourne, Australia

2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC) | 978-1-6654-4207-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/SMC52423.2021.9658662

Exploring Conventional, Automated and Deep Machine Learning for Electrodermal Activity-Based Drivers' Stress Recognition

Katharina Lingelbach[1,2*], Michael Bui[1], Frederik Diederichs[1], and Mathias Vukelić[1]

***Abstract*—Stress and cognitive overload during driving are associated with decreased performance potentially leading to serious mistakes and even fatal incidents. Therefore, research on drivers' mental states recognition is promising to reduce these traffic accidents caused by human error (e.g., in combination with driver assistance systems and automated driving functions). Easy-to-use, unobtrusive wearables allow convenient measurement of electrodermal activity (EDA) which is an informative measure for the experienced stress level. In this article, we explore the potential of various conventional machine learning (ML) models with hand-crafted features, automated pipeline optimization (AutoML), and deep learning (DL) to recognize drivers' stress states from EDA recordings in a driving simulator. Three different stress states (low, mid, and high stress) were induced via (a) the complexity level of the driving task (manual and automated driving) and (b) simultaneous secondary cognitive tasks. Our results reveal that a k-nearest neighbors (KNN) classifier with handcrafted features of the phasic and tonic EDA response as well as a pipeline suggested by AutoML via Tree-Based Pipeline Optimization (TPOT) are particularly suited with a high classification performance above an empirical chance level estimated via a dummy classifier. Predicting the three stress levels, they achieved a bootstrapped balanced accuracy of 63.7 % (KNN; 95 % confidence interval (CI): [36.7, 86.7]) and 71.6 % (stacking classifier of the AutoML; 95 % CI: [50.0, 90.0]), respectively. Interestingly, the DL model architecture was not superior in performance compared to the conventional and AutoML models with handcrafted features. Our results propose that AutoML might be beneficial to find optimal ML pipelines for EDA-based state recognition. In the future, we aim to evaluate the here proposed models regarding their generalization ability by applying them (a) on new datasets collected during realistic driving scenarios and (b) within a subject-independent approach (i.e., training one model for all subjects).**

This work was supported by a grant from the Baden-Wuerttemberg Ministry for Economic Affairs, Labour and Housing (Project »KI-Fortschrittszentrum LERNENDE SYSTEME UND KOGNITIVE ROBOTIK«). Data acquisition was funded via the European Union's Horizon 2020 research and innovation program under grant agreement ID. 688900 (ADAS&ME www.adasandme.com).

*Corresponding author: K. Lingelbach; phone: +49 711 970 5342; fax: +49 711 970 2299: katharina.lingelbach@iao.fraunhofer.de).

[1]K. Lingelbach, M. Bui, F. Diederichs, and M. Vukelić are with the Fraunhofer Institute for Industrial Engineering IAO, 70569 Stuttgart, Germany.

(e-mails: michael.bui@iao.fraunhofer.de, frederik.diederichs@iao.fraunhofer.de, mathias.vukelic@iao.fraunhofer.de)

[2]K. Lingelbach is with the Applied Neurocognitive Psychology Lab, Department of Psychology, University of Oldenburg, 26129 Oldenburg, Germany.

I. Introduction

Distracted driving and reduced attentional resources due to stressful events or cognitive overload are a major concern for traffic safety causing numerous traffic accidents annually [1]. A driver's ability to process all relevant signs and perform the driving task adequately, that is, without errors, is dependent on the ratio of available and required (cognitive) processing resources [2]. Available cognitive resources are claimed by (a) the driving task itself, and (b) further simultaneous secondary tasks (e.g., answering a call, entering the destination in a route planner, or being engaged in a conversation). The complexity of the driving task varies depending on road and traffic conditions as well as the related amount of information which needs to be continuously integrated and updated [3]. The secondary task represents a distraction from the primary driving task and reduces driving performance [4-5]. By monitoring the driver's mental states, safety-critical distractions, stress, or cognitive overload can be identified, and potentially life-threatening situations and errors prevented; for instance, via an assistive driving system [6]. Various sensor recordings can be used to monitor drivers' mental states: behavioral data (e.g., speed and acceleration of the vehicle) [7], facial and body expressions [8], eye gaze [9-10], neurophysiological signals (e.g., electroencephalography, EEG [12], and functional near-infrared spectroscopy, fNIRS, [13]), or (combined) physiological signals (e.g., respiration, skin temperature, electrocardiography, ECG, and electrodermal activity, EDA) [14-21]. Especially, easy-to-use and unobtrusive sensor wearables (e.g., smart watches or wrist bands) measuring physiological signals such as EDA reveal great potential for the real-life application of driver state recognition [20-21]. Arousing or stressful events can be measured from the physiological signals as correlates of decreased parasympathetic and increased sympathetic activity in the autonomous nervous system.

Two main machine learning (ML) techniques for mental states recognition can be distinguished: (1) conventional ML with hand-crafted feature extraction and (2) deep learning (DL) with automated feature extraction. Conventional ML requires domain knowledge to extract informative features in the time, frequency, and time-frequency domain from the physiological signals. Several methods are available to perform a supervised classification on those extracted features including support vector machines (SVM), k-nearest neighbors (KNN), logistic regression (LR), or ensemble learners such as random forest classifiers (RFC) or gradient boosting classifiers (GBC) (e.g., [7]). In contrast, DL models such as convolutional (CNN) or recurrent neuronal networks

(RNN) circumvent the necessity of domain knowledge and manual feature engineering by automatically extracting meaningful and complex features from the recorded signals during the training phase. Recent literature suggested DL architectures (CNN, RNN, or combined CNN+RNN) to be particularly suitable for EDA-based state recognition [22-24]. In addition to the process of elaborated feature engineering, choices regarding the ML pipeline including model selection and hyperparameter optimization have a major impact on the classification performance. Recently, along with the exponential increase in computing power, various algorithms for automated pipeline optimizations were developed [25].

However, to our knowledge, such AutoML approaches have received rather little attention so far in the research field of drivers' stress recognition. In addition, model comparisons are mostly reported without providing confidence intervals of the classifiers' performance as well as empirical chance levels (e.g., by using a dummy classifier) as baseline reducing the robustness and validity of the comparisons. Hence, we systematically compared (1) conventional ML models with handcrafted features, that are LR, linear discriminant analysis (LDA), SVM, KNN, RFC, GBC, (2) AutoML with handcrafted features using the Tree-Based Pipeline Optimization (TPOT) toolbox (v.0.11.7) [26-27] to identify the optimal ML pipeline for the classification (model selection and combination as well as hyperparameter settings), and (3) DL using a CNN-based DeepResNet with residual blocks similar to [22-23, 28]. We aimed to identify a suitable ML approach to robustly recognize drivers' stress states from EDA recordings within close-to-realistic driving scenarios. Especially in complex driving scenarios, it is important to recognize stress and cognitive overload at an early stage in order to have enough time for immediate countermeasures. Therefore, we were interested whether it is possible to robustly detect stress from rather short time windows of 10 s.

II. Material and Methods

A. Participants

Nine healthy volunteers (two male, age: $M = 35.33 \pm SD = 13.70$ years) participated in the study. One participant had to be excluded due to an insufficient number of samples in one class (< 7 samples) remaining with eight participants for the ML analysis (two male, age: $M = 36.13 \pm SD = 14.14$ years). Table 1 summarizes the number of class samples per participant. Variation in the number of samples and length of the conditions among participants can be explained due to the different driving speed during manual driving. In addition to the electrodermal activity, we recorded electrocardiographic activity (ECG) during a close-to-realistic driving scenario. However, we here focus solely on the EDA recordings and exclude the latter in this work. The study was approved by the ethics committee of the Technical University of Darmstadt, Germany (ID EK33/2018). Participants signed a written informed consent according to the recommendations of the declaration of Helsinki before the experiment and received monetary compensation.

B. Experimental Procedure

A driving simulator at the Fraunhofer IAO was chosen as experimental environment with a vehicle mock-up of a Porsche Macan and the SILAB driving simulator software (v. 5.0; Figure 1). Participants had to drive on a two-lane highway alternating between manual and automated driving phases as well as transitioning between these conditions. During some driving phases, participants were asked to perform secondary tasks comprising either (1) a visual-cognitive in-tray task on a tablet or (2) an auditory n-task (i.e., a 1-back task). The in-tray task required reading and remembering emails and reflects rather low (during automated driving) to medium (during manual driving) stress induction, whereas the auditory 1-back task reflects rather high stress induction independent of the driving style (i.e., manual or automated). For the driving task, difficult weather conditions (i.e., heavy raining) were chosen as medium stress induction. In total, each of the three stress levels (low, medium, high) were induced twice. Figure 2 provides an overview of the driving phases in the experiment. EDA was recorded using the wearable Shimmer3 GSR Unit and iMotions software with a sampling rate at 128 Hz. We mounted electrodes on the fingertips of the left index finger and middle finger as suggested by the Shimmer manual. The left hand was placed on the armrest of the door to reduce movement artifacts.

Figure 1. Immersive driving simulator as experimental environment at the Fraunhofer IAO.

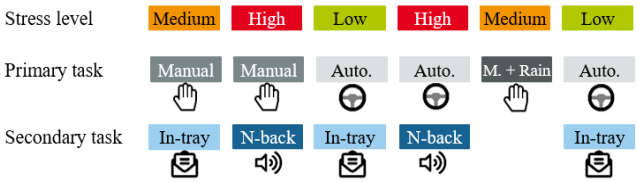

Figure 2. Overview of the driving phases with alternating driving complexities (primary task) and simultaneous secondary tasks.

III. Machine Learning for Driver State Detection

All analyses are performed via custom written or adapted scripts in python™ (3.7). The neurokit2 toolbox was used for EDA processing [29].

A. Preprocessing and Feature Extraction

The EDA signal was low-pass filtered using a 5th order Butterworth infinite impulse response (IIR) filter with a cut-off frequency at 1 Hz followed by a moving average smoothing using a linear convolution with a filter kernel size of 0.75 * sampling rate and a boxzen window [30]. Next, the signal of each driving phase was cut into non-overlapping

epochs of 10 s. The 10 s epochs were z-score baseline corrected using the mean and standard deviation of a time window of 500 ms before each epoch. The epoched EDA signal was decomposed in phasic and tonic components via the cvxEDA algorithm using a convex optimization [31]. For the DL approach, the phasic and tonic signals are fed in the CNN-based DeepResNet without further feature extractions as in [23]. In the conventional and AutoML approach, we extracted statistical features (*min, max, mean, sd, kurtosis, skewness)* from the tonic and phasic components as well as additional peak-related features from the phasic response (*sum of peaks of skin conductance response (SCR), mean amplitude of SCR, sum of SCR recovery, average time of SCR recovery*) [32]. Bootstrapped distribution with overall mean of the three stress conditions for each feature are provided in Figure 3.

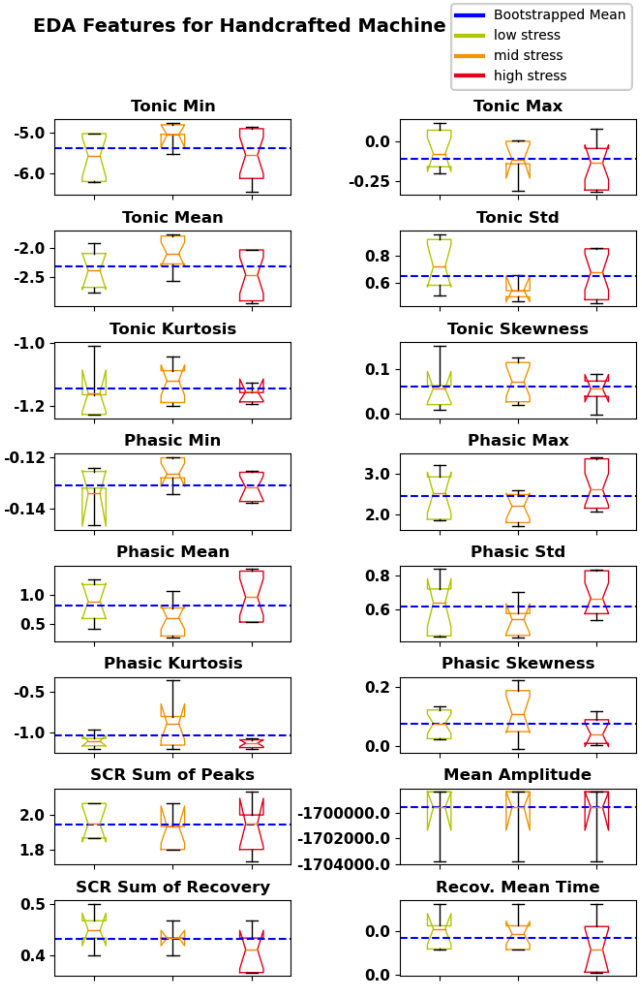

Figure 3. Comparison of bootstrapped means with 5000 repetitions for the stress level conditions (low, mid, high) per feature. Features were extracted from the baseline-corrected epoched tonic and phasic responses. Notches in the boxes of the plot visualize the upper and lower boundary of the mean's 95 % CI. The blue line representing the overall mean. The box comprises 50% of the distribution from the 25th to the 75th quartile. The ends of the whiskers represent the 5th and 95th quantile of the distribution.

The labels for each epoch were assigned according to the respective experimental stress phase (low, mid, high stress; see Figure 2).

TABLE I. SAMPLES PER STRESS LEVEL AND PARTICIPANT

Participant	Samples per Stress Level		
	Low Stress	*Medium Stress*	*High Stress*
1	30	12	30
2	24	36	24
3	12	30	30
4	18	12	24
5	30	12	30
6	24	18	24
7	24	18	24
8	30	24	24
9	12	6	30
M (*sd*)	22.67 (6.80)	18.67 (9.14)	26.67 (2.98)

Note. M: mean, sd: standard deviation. Participant 9 is excluded in the following analysis due to an insufficient number of samples in the medium stress class.

C. Conventional Machine Learning

We explored the following conventional ML classifiers: LR, LDA, SVM, KNN, RFC, and GBC as implemented in scikit-learn (v.0.22.2) [30]. Stress recognition was performed on participant-level and the data of each participant was divided into a training and a test set with a stratified 80:20 split. Extracted features were scaled via the StandardScaler and reduced by applying a principal component analysis (PCA). Only components that explain 95% of the variance in their sum when ranked decreasingly in their contribution were selected as features ($M = 6.54 \pm SD = 0.62$ components). Hyperparameters of the classifiers were optimized per iteration and participant using the training data in a 5-fold cross-validated random grid search and evaluated with the performance measure accuracy balanced for the number of samples per class. A 5-fold cross-validation with balanced accuracy as performance metric was chosen to evaluate the prediction on the training and test set.

D. Auto Machine Learning

We used TPOT with 100 generations, a population size of 500 and 5-fold cross-validation to find an optimal ML pipeline via AutoML [26,27]. Data from all participants were used to find an optimal pipeline. We here present the pipelines suggested for the 2nd and 100th generation. The former was included due to strong overfitting in the 100th generation. The suggested pipeline of the 100th generation includes a feature map approximation via radial basis function kernel (gamma=0.601) and a stacking estimator comprising a Bernoulli Naïve Bayes classifier (alpha=0.1, fit_prior=True) and GBC (learning_rate = 0.5, max_depth = 2, max_features = 0.5, min_samples_leaf = 7, min_samples_split = 17, n_estimators = 100, subsample = 0.901). The pipeline of the 2nd generation proposes a feature selector removing all low-variance features (threshold=0.2) and stacking estimator comprising a GBC (learning_rate = 0.5, max_depth = 7, max_features = 0.1, min_samples_leaf = 7, min_samples_split = 5, n_estimators = 100, subsample = 0.851) and RFC (bootstrap = True, criterion = entropy, max_features = 0.3, min_samples_leaf = 2, min_samples_split = 3, n_estimators = 100). We evaluated the pipeline using a stratified 80:20 train-test split and 5-fold cross-validation with balanced accuracy as performance metric for the model evaluation as in the conventional ML approach.

E. Deep Learning – Convolutional Neural Network

For the DL approach, we used a 1D CNN-based DeepResNet architecture that applies kernels along the temporal dimension of the EDA data similar to [23, 28]. The first part of the model extracts features automatically during training even from short time series (e.g., 10 s epochs). As input, we used the phasic and tonic signals per epoch [23]. The CNN-based DeepResNet was implemented via keras (v.2.2.4) with tensorflow backend. The architecture consists of a convolutional layer (Conv) with kernel size 7 and 1270 samples followed by a feed-forward skip connection with one Conv layer (kernel size 1) and three residual blocks with two Conv layers (kernel size 3 and 1) and a subsampling by the factor of 2 per block (see Figure 4). The output of the last residual block is fed into a Conv layer (kernel size 1) followed by a fully connected layer with softmax activation function. We used the sparse categorical cross entropy as loss function and the Adam optimizer with an initial learning rate of 0.01. Each Conv layer is followed by a batch normalization and rectified linear activation unit (ReLu). Within the main branches of the skip connection and residual blocks as well as after the final convolutional layer, a dropout with a ratio of 0.8 addresses overfitting. Within the side branches of the skip connection and residual blocks as well as before the fully connected layer, a MaxPooling layer reduces the dimension of the feature space. Weight initialization was optimized for ReLu nonlinearities as suggested by [34]. Epoch size was set to 100 for training the model.

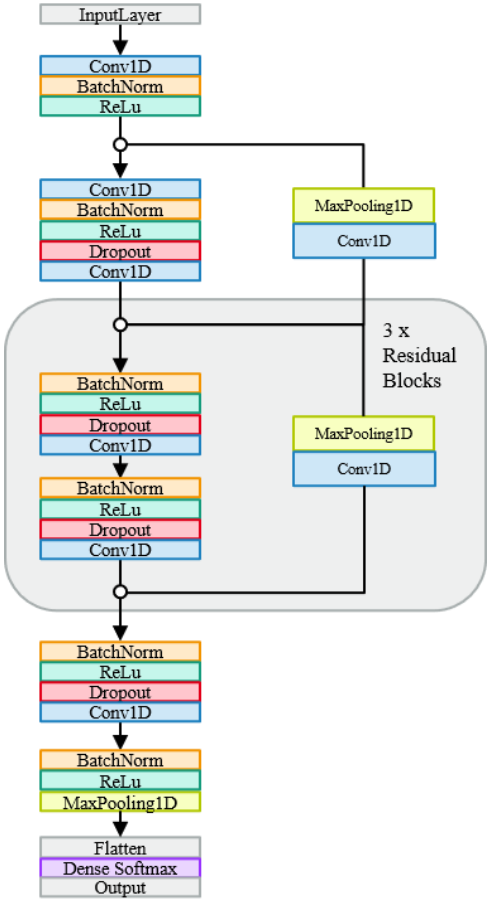

Figure 4. DeepResNet architecture for EDA-based three class stress recognition.

F. Results

For the comparison of the model performance and to account for imbalanced classes, we used an empirical chance level estimated by a dummy classifier with the method stratified as reference implemented via scikit-learn. To gain a distribution of the classifiers' average performance, we used a Monte Carlo Simulation by retraining the ML pipeline 100 times using the iteration number as random seed for train test split and model initiation. No overlap between the CIs of these distributions with those of the dummy classifier indicates a strong statistical significance and a partial overlap without including the mean a moderate statistical significance of $p < .05$ [35]. Figure 5 and Table 2 provide the mean balanced accuracy averaged over the 100 Monte Carlo iterations and participants per classifier and the 95 % CI of the complete distribution. The empirical chance level was estimated at $M = 0.353\%$ ([0.125, 0.6]). Results reveal similar classification performance for the conventional classifiers, AutoML, and DeepResNet. For the SVM, KNN, RFC, GBC as well as AutoML models of the 2nd and 100th generation, a strong overfit is observable as reflected in a much lower classification performance for the test set compared to the training. The SVM, KNN, and RFC as well as the two AutoML models could predict the stress level significantly better than the empirical chance model (i.e., the dummy classifier) when averaged over participants.

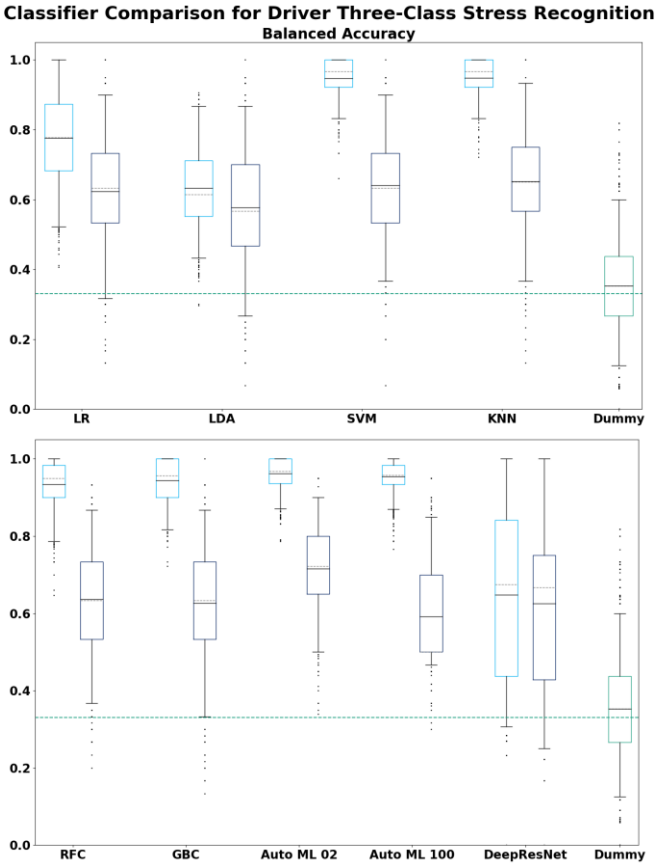

Figure 5. Distribution of balanced accuracy of the train (left box, light blue) and test set (right box, dark blue) from 100 Monte Carlo iterations per

participant and model. Whiskers represent the 5th and 95th CI of the distribution. Solid line within the boxes: mean. Dashed line within the boxes: median. Outer dashed line: theoretical chance level at 0.33. Outer right box: dummy classifier as empirical chance level (green).

TABLE II. STATISTICAL COMPARISON OF THE MODELS.

Classifier	Classification Performance with Balanced Accuracy					
	Training			*Test*		
	Lower CI	*Mean*	*Upper CI*	Lower CI	Mean	*Upper CI*
LR	0.522	0.776	1.0	0.317	0.623	0.9
LDA	0.433	0.633	0.867	0.267	0.577	0.867
SVM	0.833	0.947	1.0	0.367	0.641	0.9
KNN	0.833	0.948	1.0	0.367	0.652	0.933
RFC	0.787	0.934	1.0	0.367	0.637	0.867
GBC	0.817	0.944	1.0	0.333	0.627	0.867
AutoML 02	0.871	0.962	1.0	0.5	0.716	0.9
AutoML 100	0.87	0.955	1.0	0.467	0.591	0.85
DeepResNet	0.307	0.648	1.0	0.25	0.625	1.0
Dummy				0.125	0.353	0.6

Note. No overlap between the CIs with those of the dummy classifier indicate significantly better results (highlighted in yellow).

We compared the results per participant for the best conventional ML model, that is the KNN, and best AutoML model, that is the model of the 2nd generation. For the KNN, classification performance was above chance in half of the participants. For the AutoML 2nd generation model, classification performance was above chance in all participants (see Table 3).

TABLE III. STATISTICAL MODEL COMPARISON PER PARTICIPANT.

Participants	Classification Performance with Balanced Accuracy					
	Training			*Test*		
	Lower CI	*Mean*	*Upper CI*	Lower CI	Mean	*Upper CI*
	KNN					
1	0.803	0.899	0.984	0.348	0.608	0.817
2	0.933	0.985	1.0	0.449	0.663	0.884
3	0.833	0.930	1.0	0.333	0.610	0.817
4	0.767	0.907	1.0	0.332	0.626	0.9
5	0.879	0.945	1.0	0.558	0.782	0.942
6	0.872	0.954	1.0	0.316	0.569	0.8
7	0.917	0.977	1.0	0.465	0.708	0.933
8	0.927	0.985	1.0	0.367	0.648	0.884
	AutoML Generation 02					
1	0.848	0.930	0.980	0.5	0.592	0.85
2	0.939	0.989	1.0	0.5	0.618	0.872
3	0.855	0.940	1.0	0.5	0.591	0.851
4	0.844	0.922	0.982	0.416	0.552	0.85
5	0.874	0.939	0.979	0.5	0.594	0.85
6	0.895	0.966	1.0	0.467	0.579	0.800
7	0.904	0.968	1.0	0.467	0.592	0.85
8	0.938	0.986	1.0	0.5	0.612	0.872
	Dummy					
1	-	-	-	0.133	0.372	0.6
2	-	-	-	0.118	0.343	0.588
3	-	-	-	0.133	0.373	0.6
4	-	-	-	0.091	0.365	0.636
5	-	-	-	0.133	0.368	0.6
6	-	-	-	0.071	0.335	0.571
7	-	-	-	0.071	0.335	0.571
8	-	-	-	0.125	0.335	0.563

Note. No overlap between the CIs with those of the dummy classifier: $p < .01$; Partial overlap without including the means: $p < .05$ [30]. Significant results are highlighted yellow.

IV. DISCUSSION

We evaluated several conventional ML models, AutoML, and DL for EDA-based drivers' stress recognition. Therefore, we designed a close-to-realistic driving scenario allowing to induce three different stress levels via the driving task itself (i.e., automated and manual) and simultaneous secondary tasks. Our results reveal that the here chosen DL architecture is not superior to the conventional hand-crafted and automated ML for the EDA-based recognition of the three different stress levels. This might be explained due to the chosen model parameters (e.g., numbers of residual blocks, kernel size, or learning rate) or the rather small data set. When comparing conventional ML models with the manually composed pipeline, the KNN and SVM seem to be particularly well suited for EDA-based stress detection with above chance-level classification performance, which is in line with [24, 36]. However, we could achieve even higher accuracy when using an optimized pipeline via the AutoML TPOT. Since the model of the 100th generation strongly overfitted, which is reflected in the high deviance between the performance of the training and test set, we also explored a second pipeline suggested in one of the earlier generations (here the 2nd generation). The better performance of the 2nd compared to the 100th generation pipeline can be explained by the fact that the pipeline was optimized using the whole dataset with all participants (with the highest performance in the 100th generation) but later evaluated separately for each participant (with higher performance in the 2nd generation). The proposed 2nd generation pipeline was a stacked estimator comprising a GBC and RFC. It showed the highest balanced accuracy with 71.6 %, (CI 95%: lower boundary of 50, upper boundary of 90), for the three-class driver stress recognition. Compared to the conventional ML approach, the AutoML combined multiple classifiers into a stacking estimator. Stacking allows to combine benefits of several estimators by using the output of each individual estimator as input of a final one. This approach might have contributed to the higher classification performance of the AutoML models and should be considered when manually developing ML pipelines. We observed a strong variance not only among the ML classifiers but also in comparison to the dummy classifier which emphasizes the importance of repeated evaluation of model performance to access accuracy differences. Future research could investigate the potential of AutoML including DL architectures, since these approaches would not only optimize the ML pipeline, hyperparameter selection (which we here have neglected within the DL approach) but also feature extraction [25]. Major challenges of the EDA-based drivers' stress recognition in general and in particular when using AutoML to optimize ML pipelines are the small data sets and related low generalization. Longitudinal study designs with multiple measurement times per subject could help to provide a larger number of samples per subject but bear a new challenge regarding generalization due to variability across sessions. When applying mental state recognition in realistic environments, time complexity of the ML models will be a further critical factor which needs to be considered. Although time insensitive and computationally costly methods might be acceptable during pipeline optimization, time-effective methods are necessary for the later real-time state detection in naturalistic driving applications.

V. Conclusion

A robust classification performance and easy-to-use, unobtrusive sensors pave the way for the stress recognition in naturalistic driving scenarios. Our results identify optimal ML features and algorithms for EDA-based classification of stress levels. In a next step, we aim to evaluate the suitability and performance of the suggested pipeline for drivers' stress recognition using new data sets (potentially involving a real driving task). Furthermore, we plan to combine various physiological sensor signals (e.g., ECG and EDA) within one classification model for multi-modal driver state recognition.

Acknowledgment

We would like to thank Katrin Grillo for her patients and dedication during data acquisition and Yannick Lingelbach for his constructive criticism of the manuscript.

References

[1] N. Lavie, "Attention, Distraction, and Cognitive Control Under Load," *Curr Dir Psychol Sci*, vol. 19, no. 3, pp. 143–148, 2010.

[2] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research," in *Advances in Psychology, Human Mental Workload*: Elsevier, 1988, pp. 139–183.

[3] F. P. Da Silva, "Mental Workload, Task Demand and Driving Performance: What Relation?," *Procedia - Social and Behavioral Sciences*, vol. 162, pp. 310–319, 2014.

[4] A. E. Wester, K. B. E. Böcker, E. R. Volkerts, J. C. Verster, and J. L. Kenemans, "Event-related potentials and secondary task performance during simulated driving," *Accident; analysis and prevention*, vol. 40, no. 1, pp. 1–7, 2008, doi: 10.1016/j.aap.2007.02.014.

[5] P. Papantoniou, E. Papadimitriou, and G. Yannis, "Review of driving performance parameters critical for distracted driving research," *Transportation Research Procedia*, vol. 25, pp. 1796–1805, 2017.

[6] W.-Y. Chung, T.-W. Chong, and B.-G. Lee, "Methods to Detect and Reduce Driver Stress: A Review," *Int. J. Automot. Technol.*, vol. 20, no. 5, pp. 1051–1063, Oct. 2019.

[7] Y. Lu, X. Fu, E. Guo, and F. Tang, "XGBoost Algorithm-Based Monitoring Model for Urban Driving Stress: Combining Driving Behaviour, Driving Environment, and Route Familiarity," *IEEE Access*, vol. 9, pp. 21921–21938, 2021.

[8] M. A. Assari and M. Rahmati, "Driver drowsiness detection using face expression recognition," in *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, Kuala Lumpur, Malaysia, Nov. 2011 - Nov. 2011, pp. 337–341.

[9] L. Fletcher and A. Zelinsky, "Driver Inattention Detection based on Eye Gaze—Road Event Correlation," *The International Journal of Robotics Research*, vol. 28, no. 6, pp. 774–801, 2009.

[10] T. Hirayama, K. Mase, and K. Takeda, "Detection of driver distraction based on temporal relationship between eye-gaze and peripheral vehicle behavior," in *2012 15th International IEEE Conference on Intelligent Transportation Systems*, Anchorage, AK, USA, Sep. 2012 - Sep. 2012, pp. 870–875.

[11] F. de Arriba-Pérez, J. M. Santos-Gago, M. Caeiro-Rodríguez, and M. Ramos-Merino, "Study of stress detection and proposal of stress-related features using commercial-off-the-shelf wrist wearables," *J. Ambient Intell. Humaniz. Comput.*, vol. 10, no. 12, pp. 4925–4945, Dec. 2019.

[12] H. Zeng *et al.,* "A LightGBM-Based EEG Analysis Method for Driver Mental States Classification," *Computational intelligence and neuroscience*, vol. 2019, p. 3761203, 2019.

[13] T. Nguyen, S. Ahn, H. Jang, S. C. Jun, and J. G. Kim, "Utilization of a combined EEG/NIRS system to predict driver drowsiness," *Scientific reports*, vol. 7, p. 43933, 2017, doi: 10.1038/srep43933.

[14] M. Awais, N. Badruddin, and M. Drieberg, "A Hybrid Approach to Detect Driver Drowsiness Utilizing Physiological Signals to Improve System Performance and Wearability," *Sensors (Basel, Switzerland)*, vol. 17, no. 9, 2017, doi: 10.3390/s17091991.

[15] G. Rigas, Y. Goletsis, P. Bougia, and D. I. Fotiadis, "Towards Driver's State Recognition on Real Driving Conditions," *International Journal of Vehicular Technology*, vol. 2011, pp. 1–14, 2011.

[16] J. A. Healey and R. W. Picard, "Detecting Stress During Real-World Driving Tasks Using Physiological Sensors," *IEEE Trans. Intell. Transport. Syst.*, vol. 6, no. 2, pp. 156–166, 2005.

[17] C. D. Katsis, N. Katertsidis, G. Ganiatsas, and D. I. Fotiadis, "Toward Emotion Recognition in Car-Racing Drivers: A Biosignal Processing Approach," *IEEE Trans. Syst., Man, Cybern. A*, vol. 38, no. 3, pp. 502–512, 2008, doi: 10.1109/TSMCA.2008.918624.

[18] P. Zontone, A. Affanni, R. Bernardini, A. Piras, and R. Rinaldo, "Stress Detection Through Electrodermal Activity (EDA) and Electrocardiogram (ECG) Analysis in Car Drivers," in *2019 27th European Signal Processing Conference (EUSIPCO)*, A Coruna, Spain, 2019, pp. 1–5.

[19] F. de Arriba Pérez, J. M. Santos-Gago, M. Caeiro-Rodríguez, and M. J. Fernández Iglesias, "Evaluation of Commercial-Off-The-Shelf Wrist Wearables to Estimate Stress on Students," *Journal of visualized experiments : JoVE*, no. 136, 2018, doi: 10.3791/57590.

[20] T. Kundinger, N. Sofra, and A. Riener, "Assessment of the Potential of Wrist-Worn Wearable Sensors for Driver Drowsiness Detection," *Sensors (Basel, Switzerland)*, vol. 20, no. 4, 2020.

[21] N. El Haouij, R. Ghozi, J.-M. Poggi, S. Sevestre-Ghalila, and M. Jaïdane, "Self-similarity analysis of vehicle driver's electrodermal activity," *Qual Reliab Engng Int*, vol. 35, no. 5, pp. 1502–1513, 2019.

[22] F. Al Machot, A. Elmachot, M. Ali, E. Al Machot, and K. Kyamakya, "A Deep-Learning Model for Subject-Independent Human Emotion Recognition Using Electrodermal Activity Sensors," *Sensors (Basel, Switzerland)*, vol. 19, no. 7, 2019, doi: 10.3390/s19071659.

[23] D. Yu and S. Sun, "A Systematic Exploration of Deep Neural Networks for EDA-Based Emotion Recognition," *Information*, vol. 11, no. 4, p. 212, 2020, doi: 10.3390/info11040212.

[24] R. Sánchez-Reolid, M. T. López, and A. Fernández-Caballero, *Machine Learning for Stress Detection from Electrodermal Activity: A Scoping Review*, 2020.

[25] X. He, K. Zhao, and X. Chu, "AutoML: A survey of the state-of-the-art," *Knowledge-Based Systems*, vol. 212, p. 106622, 2021, doi: 10.1016/j.knosys.2020.106622.

[26] R. S. Olson, R. J. Urbanowicz, P. C. Andrews, N. A. Lavender, C. La Kidd, and J. H. Moore, "Automating Biomedical Data Science Through Tree-Based Pipeline Optimization," in *Applications of Evolutionary Computation: 19th European Conference, EvoApplications 2016, Porto, Portugal, March 30 - April 1, 2016, Proceedings, Part I*, pp. 123–137.

[27] T. T. Le, W. Fu, and J. H. Moore, "Scaling tree-based automated machine learning to biomedical big data with a feature set selector," *Bioinformatics*, vol. 36, no. 1, pp. 250–256, 2020.

[28] A. H. Ribeiro *et al.,* "Automatic diagnosis of the 12-lead ECG using a deep neural network," *Nature communications*, vol. 11, no. 1, p. 1760, 2020, doi: 10.1038/s41467-020-15432-4.

[29] D. Makowski *et al., NeuroKit2: A Python Toolbox for Neurophysiological Signal Processing*, 2020.

[30] S. W. Smith, The scientist and engineer's guide to digital signal processing, 1st ed. San Diego, Calif.: California Technical Publ, 1997.

[31] A. Greco, G. Valenza, A. Lanata, E. P. Scilingo, and L. Citi, "cvxEDA: A Convex Optimization Approach to Electrodermal Activity Processing," IEEE transactions on bio-medical engineering, vol. 63, no. 4, pp. 797–804, 2016, doi: 10.1109/TBME.2015.2474131.

[32] H. Gamboa, "Multi-modal behavioral biometrics based on hci and electrophysiology," PhD Thesis, Universidade Técnica de Lisboa, 2008.

[33] F. Pedregosa *et al.,* "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, no. 12, pp. 2825–2830, 2011.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," Feb. 2015. [Online]. Available: http://arxiv.org/pdf/1502.01852v1

[35] G. Cumming and S. Finch, "Inference by eye: confidence intervals and how to read pictures of data," The American psychologist, vol. 60, no. 2, pp. 170–180, 2005, doi: 10.1037/0003-066X.60.2.170.

[36] Y. S. Can, N. Chalabianloo, D. Ekiz, and C. Ersoy, "Continuous Stress Detection Using Wearable Sensors in Real Life: Algorithmic Programming Contest Case Study," *Sensors (Basel, Switzerland)*, vol. 19, no. 8, 2019, doi: 10.3390/s19081849.