Sustainable Selection of Machine Learning Algorithm for Gender-bias Attenuated Prediction

IEEE Open Journal of
Vehicular Technology

Received 11 October 2024; accepted 13 November 2024. Date of publication 20 November 2024;
date of current version 22 January 2025. The review of this article was coordinated by Editor Darren Galpin.

Digital Object Identifier 10.1109/OJVT.2024.3502921

Sustainable Selection of Machine Learning Algorithm for Gender-Bias Attenuated Prediction

RAIK ORBAY [1,2] AND EVELINA WIKNER [2] (Member, IEEE)

[1]Volvo Car Corporation - 97100 Propulsion and Energy - Strategy & Execution, SE-405 31 Gothenburg, Sweden
[2]Department of Electrical Engineering, Chalmers Technology University, SE-412 96 Gothenburg, Sweden

CORRESPONDING AUTHOR: RAIK ORBAY (e-mail: raik.orbay@volvocars.com).

ABSTRACT Research into novel approaches like Machine Learning (ML) promotes a new set of opportunities for sustainable development of applications through automation. However, there are certain ML tasks which are prone to spurious classification, mainly due to the bias in legacy data. One well-known and highly actual misclassification case concerns gender. As the vast dataset for engineering rules, standards and experiments are based on men, a bias towards women is the subject of research. Accordingly, any bias should be contained before the algorithms are deployed to the service of the sustainable society. There is a substantial amount of data on ML gender-bias in the literature. In these, the majority of the investigated cases are for ML branches like image or sound processing and text recognition. However, utilizing ML for driving style investigations is not an extensively researched area. In this work, a novel application for gender-based classification with bias-attenuation using anonymized driving data will be presented. Using data devoid of biometric and geographic information, the proposed pipeline distinguishes manifested binary genders with 80% accuracy for the drivers in the holdout data set. In addition, a method for sustainable algorithm selection and its extension to embedded applications, is proposed. An investigation into the environmental burden of seven different types of ML algorithms was conducted and the popular neural network algorithm had the highest environmental burden.

INDEX TERMS Battery electric vehicles, driving style classification, energy consumption, feature engineering, machine learning, unsupervised/supervised learning.

I. INTRODUCTION

Conforming with the United Nations Sustainable Development goals [1], automation will support sustainable cities and help in reducing inequalities. Machine Learning (ML) is an enabler for automation impacting all aspects of the society. Instead of model based approaches, ML makes use of the available data to discover the characteristics of the system at hand, which may impose a cost on the environment [2], [3], [4], [5], [6].

ML is used extensively for systems including those involving human interaction. Numerous characteristics can be used for identification of individuals using ML. Traits in human face [7], human voice, thumb impression, touch dynamics, gait dynamics, iris morphology, shopping behavior [8] etc. are reported to be used for effective identification incorporating ML. Although ML based systems have proven to reach very efficient classification, occasionally, the automatized systems misidentify individuals [9]. Accordingly, there is research aiming to improve the recognition rate [10]. Specifically, as there is a general mismatch between amount of training data for men and women, a gender-bias may affect classification score for ML algorithms. ML empowers the society to come with sustainable products and services, however, being a computation-intense undertake, ML itself may cost a CO_2 burden. Accordingly, a trade-off may be established between the ML delivery and the environmental burden due to the computations by the ML algorithms.

There is a substantial amount of data on ML gender-bias in the open literature. The bias in training may lead ML to fail a benign case of touchscreen lock failure or may lead to dire consequences like accommodation credit application rejection [11]. Research demonstrates that ML has in many cases discriminated based on gender [12], [13]. Despite its power, even the state-of-the-art Neural Network (NN) based algorithms are not spared from committing erroneous classification [14]. There is also research showing that deep learning occasionally learns uninterpretable solutions that could have counterintuitive properties [15]. Additionally, it is reported that one of the major suppliers to the vehicle industry may have gender-bias in speech recognition software [16]. Furthermore, recent research aiming at vehicle industry may also contain gender bias [17]. Another common shortcoming is assuring the quality for the training data. For the situations as the one reported in this work, when the drive data is recorded on roads with the presence of calculated (tunnels on the road), or uncalculated (GPS errors) artifacts in the training set. For the automotive industry, gender-balanced data is a reoccurring issue as the actual data is mismatched in favor of men. Most dummies used in automotive crash tests are designed to represent an average man [18]. It was first in 2011 that women's crash test dummies were to be used [19]. Also for determining thermal comfort in vehicles, the standard values of metabolic rate for an average male are used and may overestimate female metabolic rate by up to 35% [20]. Despite a substantial partition of vehicles in the market is purchased by women, their needs and preferences have been ignored throughout history until but recently [21]. Women and men use their vehicles differently, as men tend to drive more and longer distances, take more risks and thus be involved in more traffic accidents per driven km [22], [23]. By recognizing the minute differences in driving dynamics of women and men, further improvements can be achieved in developing tailored vehicles suitable for both men and women [24]. It is also well known that the diving style impacts the energy consumption of a vehicle [25], [26]. Research points out that women in average are driving slower and are also choosing more environmentally friendly vehicle models [27].

Most driving style studies typically rely on characteristics such as the driver's somatic, behavioral and emotional conditions as complements to the recorded drive data [28]. Exploiting ML algorithms for driving styles can thus be problematic, unintentionally introducing a gender bias. Hence, utilizing ML for investigating driving styles and differences in driving habits between the genders should be conducted with consideration. Accordingly, this paper will detail the steps taken to efficiently classify non-biometric identification cursors for driver gender, i.e., a ML pipeline with gender-bias attenuated gender prediction. A noteworthy effort in this study has been to attenuate the gender bias by keeping the amount of input data for both genders in balance. In addition, all data features which may indicate affluence via residential address, identification of profession, etc., are eliminated. Further, tactical feature engineering steps to improve the classification are presented. The final contribution is devoted to a rarely considered problem concerning ML algorithms, a comparison of the environmental impact of seven selected ML algorithms

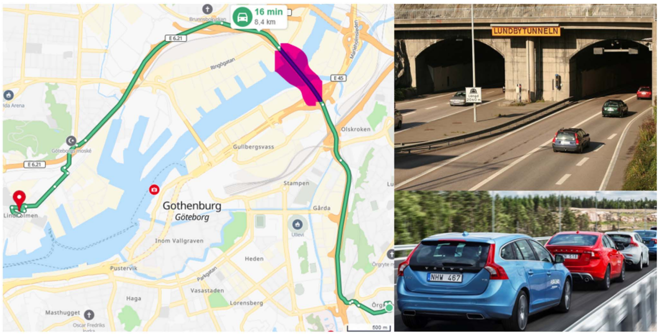

FIGURE 1. A representative driving trajectory of an assumed individual living in Gothenburg, a Swedish town, and commuting to work through a tunnel (in magenta), which is shown in the upper right infrastructure image from Wikipedia. Representative vehicle models from Volvo Cars archives shown in the lower left image. Map modified from qwant.com.

applied to the same classification problem. Conforming with the sustainability emphasis of the paper, the complete work is done using open-source code.

II. EXPERIMENTAL DATA

A set of 30 vehicles with individual drivers, who are distributed 50% women and 50% men, are investigated. The data set comprises individuals living near the Swedish town Gothenburg as representative owners of the contemporary vehicle park in Europe. The GPS-based data was collected by the Swedish car movement data project, of which the equipment and data acquisition procedure are described in [29]. All vehicles used by the participants are light duty class privately owned cars not older than the 2001 model year with no restrictions on power, torque, travelled distance, frequency of usage, or number of occupants.

Most of the drivers in the set use the vehicles to commute to work. Traffic in the Gothenburg region includes several tunnels, as shown in Fig. 1, but no restrictions on driven paths and trajectories are imposed for the study. The characteristics of each driver's driving style are logged using time-resolved, velocity, trip, and position variables. The data from these individuals is logged as time-stamps with a frequency of 2.5 Hz. The GPS data is sanitized into cumulative distance travelled per trip and comprises no biometric information, sound recordings or any images. In this way, the data is completely anonymized, and each automobilist's discernible characteristics are only their individual driving styles. To further generalize the reported approach, logged data is not cleaned or imputed by a human operator before being fed into the ML pipeline. Experimental data is saved as a comma-separated value (csv) file for further processing and automated imputation for spurious samples.

In the questionnaire answered by the participating drivers, only participants who positioned themselves as belonging to one of the binary gender sets and who were the sole users of their vehicles were included. In the original dataset men are overrepresented. A usual solution to unbalanced datasets are through synthetically over-sampling the minority class [30],

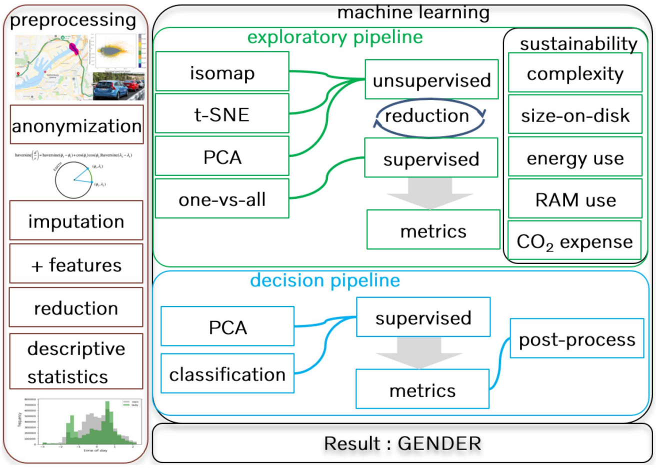

FIGURE 2. **Overall ML pipeline created using Python tools. In this flowchart the pipeline components are color coded in red for preprocessing, in green for exploratory component and in blue for final decision component.**

however, synthetic manipulation may lead to spurious balance as minority class may be over-generalized [31]. Aiming to curtail the gender bias robustly, the partition of each gender sample in the input data is equal. Following the preprocessing and exploratory analysis, an arbitrary selection of 30 vehicles driven by 50% women and 50% men are used for training and validation of the prescribed algorithms. For the final delivery, holdout data of 10 drivers comprising 50% women and 50% men, initially kept out of the training and validation data sets, are used. Being effectively unseen by the pipeline, they are then used to quantify the merit of the prescribed machine learning pipeline.

III. COMPUTATIONAL APPROACH

The study aims to minimize human interventions. Accordingly, a pipeline is created to read csv files to extract data as tensors. The experimental data is parsed and processed in a modular fashion. The overall pipeline is depicted in Fig. 2. In the following, the csv parser that searches through any arbitrary folder tree in any online repository is scripted using Python. This tree structure comprises each driver's data as a csv file. This data out of the logger is flattened; thus, it is transformed into driver class, trip and features as a tensor for later processing. The described parser can handle csv files but allows for alternative file protocols as input for versatility. All the computational steps are parallelized on High Performance Computing (HPC) resources. In this section, each individual module will be described.

A. PREPROCESSING

It is known that minute differences in driving related to unknown parameters such as the sentiments of the drivers, the number of passengers in the car, available power and torque which would impact the data set as additional entropy; could not be accounted for. Five preprocessing steps were taken before the data was fed into the pipeline. The data was prepared through anonymization, imputation, feature engineering, data reduction and descriptive statistic steps, as depicted in the Fig. 2.

First, the data had to be anonymized. Secondly, erroneous data points had to be detected and replaced. Lastly, feature engineering steps for increasing the accuracy were scripted into the pipeline.

1) ANONYMIZATION

Recent research reports a measurable bias towards women in housing [32]. Valid also in EU, more women than men suffer financial hardship in countries with high housing costs [33]. This fact introduces a bias, not correlated to drivestyles, but affluence in GPS positions registered for the drivers. More men will be living in affluent regions compared to ladies. Accordingly, effort put in place to minimize the over-representation of men in data set through casting the GPS data into covered distance.

The experimental location data is anonymized using Haversine formulation,

$$\text{havs}\left(\frac{d}{r}\right)=\text{havs}(\phi_2-\phi_1)+\cos(\phi_1)\cos(\phi_2)\,\text{havs}(\lambda_2-\lambda_1) \tag{1}$$

which includes the radius of earth, r, for calculating the distance, d, between two points. ϕ_1, ϕ_2 is latitude of the two points and λ_1, λ_2 is longitude of the two points respectively [34].

2) IMPUTATION

Due to GPS errors and loss of connection, i.e. driving through tunnel passages where the exposure to satellite signals is lacking, experimental data comprises sparsely distributed NaN cells. Three different approaches were investigated to apply imputation when invalid data is encountered. Traditional imputation using column/row means or other descriptive statistics, ML imputation with inferential statistics, and filtering (Kalman), was explored to provide the missing data in the input tensor. For the final pipeline the invalid cells of the input tensor is substituted with descriptive statistics imputation [35].

3) FEATURE ENGINEERING

The driver classification problems can exhibit a large imbalance in the distribution of the target classes. For instance, acceleration data may have several times more negative samples than positive ones. In such cases, it is recommended to use stratified sampling as implemented in StratifiedKFold and StratifiedShuffleSplit. This ensures that the relative class frequencies are preserved in each training and validation fold, i.e., the same percentage of genders for each target class are kept as in the complete dataset [35]. For this, the StratifiedK-Fold with 5-folds was implemented in the pipeline.

For supervised learning, additional preprocessing tasks were scripted. The experimental data was distributed into several training and validation data set distributions as well as swept k-fold grid searches, aiming at a good distribution in the training and validation set. A specific distribution for the training and validation populations was achieved for the investigated dataset. For the supervised learning task, the pipeline is informed on the genders of individual drivers by provided labels. As no restrictions on power or torque, as well as on the number of occupants, etc. are applied to the experimental data, this yields a challenge for the automatic preprocessing. The minute difference among dynamic driving styles due to the spread in available vehicle power compared to aggressive acceleration events is an example of this challenge. To increase the accuracy, several feature engineering practices are followed in the exploratory analysis.

B. EXPLORATORY ANALYSIS

During the initial phases of the study, an effort to directly classify drivers' gender using unsupervised learning was undertaken. This would align the data along available principle components for a straight forward classification using linear classifiers. However, a low-bias decision plane with reasonable variance was impossible to prescribe, given the entropy of the initial dataset. Hence, following the preprocessing of the data, an exploratory step was taken before starting the ML pipeline design.

The exploratory data analysis makes use of correlations, scatter plots, pairplots, histograms, probability density functions, reduction and embedding and simpler linear ML algorithms to pinpoint explicit-implicit relationships among the data series and target classes.

Being relatively light in computations, exploratory analysis can be applied to whole or subsampled input tensors. Aligned with best practices, the data is initially studied using descriptive statistics. Correlation tables are initially employed to discern superfluous features. Other visual measures like scatterplots and pairplots were also used, leading to detection of outliers. As many ML algorithms favor a Gaussian character, histograms for several features were visualized to see the distribution type the data obeyed.

In this study, the exploratory analysis effort is foremost for the programmer to comprehend the data at hand, thus not required for the ML agent to function. However, also in this work the exploratory analysis is extended to include CO_2 burden per ML approach, which is not commonly studied for exploratory reported in the literature. Specifically, the approach developed is a trial-and-error method to choose the highest scoring pipeline topology, while limiting the CO_2 burden. This step started using the nearly complete data sets exceeding 100 drivers. As this much data lead to extended computation durations, the subsets of the initial large dataset were found to be more appropriate.

Employing the principles of unsupervised learning by using Principal Component Analysis (PCA) to prescribe a relatively simple decision boundary among drivers, a first exploratory analysis is undertaken. When this approach caught low variances per feature, a more advanced unsupervised learning method as embedding is also applied. However, as embedding observed to consume more hardware resources compared to PCA, the final pipeline does not include embedding.

FIGURE 3. Exploratory analysis of data with a pairplot, where the correlations among features are depicted. In the inset, there is a discernable difference in the driving speeds.

An example for the prescribed exploratory data analysis is visualized in Fig. 3, where a difference in speed can be observed for the two gender groups. This analysis is used as a basis for the feature design and specifically to decide on feature engineering steps, folding practices and scaling.

1) LOW DIMENSIONAL STATISTICS THROUGH LINEAR FEATURE EXTRACTION

Linear feature extraction via Singular Value Decomposition (SVD) is used to discern features and to sort them in importance, so that classifiers could prescribe linear decision boundaries, using relatively less resources. PCA is a linear dimensionality reduction using SVD. In the example histogram depicted in the inset in Fig. 3, a difference between genders for the feature "speed" is observable. This difference can be quantified using PCA, yielding the "speed" as the feature with a high entropy. As an exploratory step, unsupervised learning approaches like lightweight clustering (k-means++) were also investigated for feature engineering. K-means++ makes use of mini-batches to limit the stochastic noise while assuring a decreased computational cost for large data sets [36]. Exploratory data analysis has shown that relatively low component count was sufficient for good entropy capture. A PCA process with just 5 components is programmed to generate a new input tensor. In this step, the initial tensor with $\sim$1 million rows and 17 columns is reduced to $\sim$1 million rows with 5 columns while keeping the explained variance larger than $\sim$70 %.

2) STATISTICS IN LOW DIMENSIONAL SPACE THROUGH NON-LINEAR FEATURE EXTRACTION

The non-linear feature extraction has the same purpose as the linear feature extraction presented in Section III-B1. For

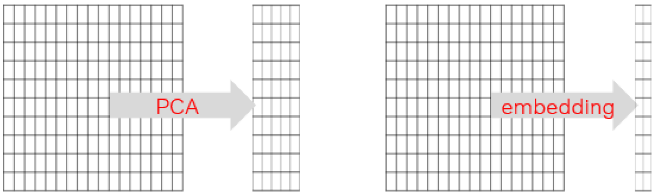

FIGURE 4. Feature engineering using unsupervised learning. PCA is reducing the feature tensor shape from a $10^6 \times 17$ to a $10^6 \times 5$ tensor and embedding from a $10^6 \times 17$ to a $10^6 \times 2$ tensor.

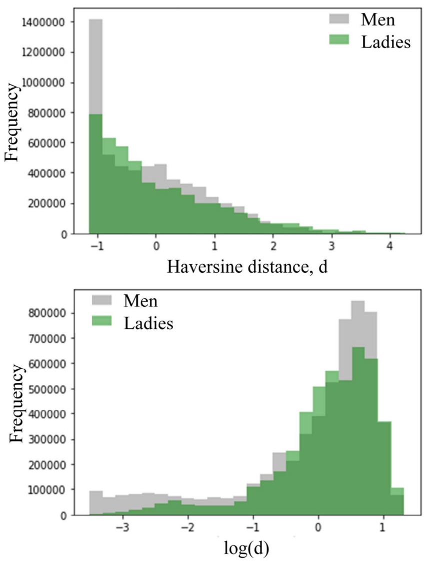

FIGURE 5. An example of feature engineering using simple arithmetic operation on the Haversine distance for 20 drivers.

the non-linear feature extraction, embedding is used and as expected, the tensor size is the limiting factor on HPC clusters. After several trials using isomap grid searches, t-distributed Stochastic Neighbor Embedding (t-SNE) was chosen to be used for the exploratory analysis. By using unsupervised learning as embedding, the feature tensor can be reduced from a $10^6 \times 17$ to a $10^6 \times 2$ tensor as elucidated in Fig. 4. However, as it is computationally heavy, embedding was not used in the final pipeline.

C. TRANSFORMATION AS FEATURE ENGINEERING

Feature engineering is a proven method to increase the feature space and improve the accuracy of ML algorithms [37]. Accordingly, several studies were launched using grid searches comprising alternative feature engineering formulation. In these approaches, it was found that the pipeline reached a higher score when additional features were introduced. In the literature, polynomial and interaction features are reported [35]; accordingly, simple arithmetic operations on the driving data gave the best results. In Fig. 5, the impact of a logarithm operation is illustrated as the logarithm of Haversine distance per driver gives a more Gaussian type of distribution correlation of the feature and thus increases the gender classification.

D. UNSUPERVISED LEARNING FOR FEATURE ENGINEERING

Already at the start of the project, as a binary characteristic, gender is eclipsed by the variance from punctual, economical, cautious driving styles, plus the spread in the car type each of the drivers possessed. The initial expectations to use unsupervised learning to discern one of the prominent binary characteristics of the drivers did not lead to a conclusive result. However, it turned out that there are certain traits unsupervised clustering algorithms can capture. The usage of unsupervised learning to augment the feature set is a known practice. To increase the classification accuracy, new features are discovered in the data set via k-means clustering [38]. As the dataset contains several millions of measurement points, a purpose-built fast k-means clustering algorithm [36] is employed in the final pipeline.

E. SUPERVISED LEARNING FOR DATA CHARACTERIZATION

Supervised learning is an efficient classification approach when the data is labelled. In this study the data is labelled using the gender information. In order to understand the characteristics of the data at hand, simple ML cases were launched for a small subset of the data as the exploratory approach. One-vs-rest classifier was used to separate one driver from a group of 8 drivers [35] to discern the inner workings of the data. Extra emphasis is put on the bias as the sample size is limited due to available HPC at a given time and a general objective to scale the developed pipeline for embedded solutions. For all the learning task reported, the data is 50/50 balanced among targets of interest, i.e., genders. Furthermore, scaling is used to ensure SVC, LR, kNN algorithms will not add bias due to the heterogeneity in the input data.

For the current study, the ML programmer was kept agnostic to the genders of the drivers; i.e., the ML programmer did not know any genders attached to the drivers. Still, each of the drivers has a unique alpha-numeric identification tag, which was used for supervised learning tasks.

F. ALGORITHM SELECTION

The large dataset at hand required HPC resources. Therefore, as discovered in Section III-B1 a PCA process with 5 components was programmed as a preprocessing step for all the ML classification cases for the final pipeline. This allowed the initial tensor with $\sim$1 million rows and 17 columns to be reduced to $\sim$1 million rows with 5 columns while keeping the explained variance larger than $\sim$70 %. To further reduce the computational burden, initial inferential statistics studies were launched to find the appropriate ML algorithm and hyperparameters tuned for the driving styles. These exploratory computations used a subset of the driving style data (8 drivers out of the 30 in the training data set) while scores, confusion matrices and receiver operation characteristics as well as energy consumption, memory occupation, size-on-disk and computation duration were logged.

Initial modeling using Logistic Regression (LR) aimed at fast results, due to the linear character of the decision surface.

In LR, the probabilities describing the possible outcomes of a single trial are modeled using an S-shaped curve (logistic function). For LR, preprocessing is important and two types of scaling were investigated: standard scaler which standardizes features by removing the mean and scaling to unit variance (LR+StdScaler), and minmax scaler which standardizes features by scaling each feature to a given range (LR+MinMax) [35].

Decision Trees (DT) aim to predict the value of a target variable by learning simple decision rules inferred from the data features. As an advantage, this algorithm and its ensembles do not need any scaling prior to usage. As a precaution to minimize DT from overfitting, Random Forests (RF), which are ensembles of DT, were also applied to the driving style dataset. RF combines a given number of DT classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control minimize-fitting [35].

Next ML model tested is based on Support Vector Machine classification (SVC). SVC constructs hyperplane(s) in a high dimensional space while maximizing the distance to the nearest training data points of any class to diminish the generalization error. Also here, the standard scaler that standardizes features by removing the mean and scaling to unit variance was used (SVC+StdScaler).

Following SVC, k-Nearest Neighbor (k-NN) classifiers were tested. k-NN finds a predefined number of training samples closest in distance to the new point, and predicts the label from these [35]. The crucial parameter distance (Euclidean/Minkowski) was investigated using a gridsearch. The k-NN classification was implemented with 5-folds and the standard scaler that standardizes features by removing the mean and scaling to unit variance (k-NN+StdScaler).

Finally, a deep feedforward NN algorithm was also run in a gridsearch setup. Multi-layer Perceptrons (MLP) are classifiers iteratively trained on the driving style data since at each time step the partial derivatives of the loss function with respect to the model parameters are computed to update the parameters [35]. Also here, the standard scaler was used (MLP+StdScaler).

After the initial exploratory ML runs were finalized, yielding tuned hyperparameters, the full test data set of 30 drivers was prescribed to the pipeline in turn for all seven investigated ML algorithms.

G. PERFORMANCE METRICS

Classification algorithms have metrics based on a comparison of predicted classes versus ground truth. During the hyperparameter tuning, the gridsearch algorithm quantified impact in classification accuracy while changing the parameters for the ML algorithms. The accuracy metric is used for both unsupervised and supervised learning. Accuracy is defined as the number of correct predictions divided by the total number of predictions.

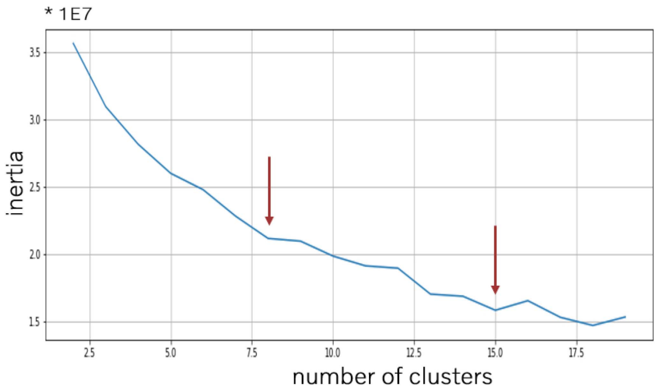

FIGURE 6. SC for feature engineering using unsupervised learning on the 30 drivers. 8 and 15 clusters were selected to be tested for their impact.

1) METRICS FOR UNSUPERVISED LEARNING

The accuracy of the unsupervised learning is determined by computing the mean Silhouette Coefficient (SC) and elbow model; an example of which is shown in Fig. 6. The SC is calculated using the mean cluster distance among cluster centers and the mean nearest-cluster distance for each sample [35]. 1 being best, 0 means overlapping clusters and minus values are cursor to failed classification. The inflation points of the elbow plot, Fig. 6, indicates the appropriate cluster counts. 8 and 15 clusters were selected for further investigations.

2) METRICS FOR SUPERVISED LEARNING

Supervised classification algorithms have additional metrics based on a comparison of predicted classes versus the ground truth. In this work, precision, recall, F1-score, Confusion Matrix (CM) and Area Under Curve - Receiver Operating Characteristics (AUC-ROC) are used as metrics.

Precision score is the ratio of true positives over the total positives predicted and assures that the classifier does not label negative data as positive. Recall is a metric which quantifies the ability of the algorithm in detecting all the true positives. The F1-score is the harmonic mean of the precision and recall.

The accuracy of the supervised learning is also determined by computing the CM and ROC for all samples. ROC curves plot the true positive count on the ordinate and false positive rate on the abscissa, see Fig. 10. As a cursor, the AUC is important to distinguish good classifiers. A 50% classifier is a random number generator; thus larger areas trespassing beyond the 50% curve are required. An ideal classifier would provide no false positives and 1 in true positive rate, positioning to top leftmost in a ROC plot. The steepness of the ROC curve is another important metric, since it is ideal to maximize the true positive rate while minimizing the false positive rate [35].

IV. ENVIRONMENTAL BURDEN

All computations use energy in the form of electricity, sourced using an energy mix comprising several renewable/non-renewable sources. The Swedish electricity generation CO_2

emission intensity of 13.3g CO_2/kWh [39] is the figure which quantifies the overall environmental burden of electricity use while using HPC resources. An ability to predict this as greenhouse gas emission intensity (gCO_2 e/kWh) before the pipeline execution is preferable, however, challenging. Due to its sustainability focus, the energy consumption while performing the computations has been of high interest for this work. Aligned with UN Sustainable Development goals [1], the environmental burden from various ML algorithms is estimated.

For computational clusters and data centers, the energy consumption is not only related to the Information Technology (IT) equipment, but also their auxiliaries like cooling systems [2], [4], [5], [40]. However, for this work, the same cluster computer has been used for all ML algorithms, and only the IT equipment energy consumption has been of interest for the comparison.

For the IT equipment, the highest energy consumption is generated by the Central Processing Unit (CPU) followed by peripheral slots, conduction losses, memory and motherboard [41]. Throughout the exploratory analysis, the CO_2 footprint of the ML algorithms is calculated based on the Thermal Design Power (TDP) of the CPU used, the time for the computation and the number of CPUs. In computational hardware, the smallest amount of heat dissipation is due to inherent irreversibilities [42]. Above this limit, the hardware consumes energy through electromagnetic processes related to logic gate operations and leakage currents.

The TDP for the CPU is retained from the Original Equipment Manufacturer (OEM) data [43]. When a full CPU is used for the computation, thus, the consumed electric energy in the process is computed as

$$E = TDP \cdot t_{parse} \tag{2}$$

Alternatively, if a single core is used,

$$E = \frac{TDP \cdot t_{parse}}{n_{core}} \tag{3}$$

where n_{core} and t_{parse} represent number of CPU cores and computation duration, respectively. For embedded applications, the Random-Access Memory (RAM) is a limiting commodity in an IT hardware. As the RAM is one of the five components with high energy consumption during operation [41], [44], which adds to the overall CO_2 burden, it is of high interest for this study. For quantification, the Kolmogorov complexity is shown to be correlated with the RAM consumption while training the ML algorithm, see Section V-A. Taking the description based on the one given by Kolmogorov [45], [46]

$$K(x) := \min\{l(p) : U(p) = x\} \tag{4}$$

the complexity of each ML algorithm can be quantified. Subsequently, the greenhouse gas emission intensity (gCO_2 e/kWh) is quantified for the computations. For the computations, an Intel Xeon E5 2620 v3 2 CPUs, 24 cores @85W is used [43]. The complexity, RAM, and space-on-disk used by the ML algorithms the resulting energy consumption and corresponding CO_2 contribution are included in Section V-A.

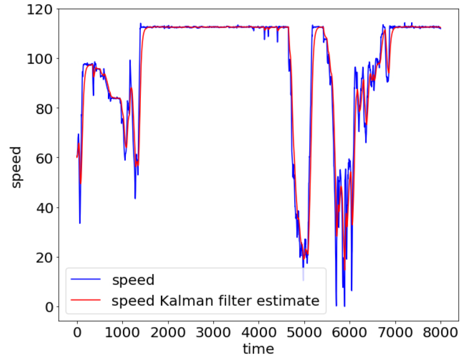

FIGURE 7. An example Kalman filter estimate for the speed and its smoothing effect on the speedprofile.

V. RESULTS

In order to construct a gender-bias attenuated ML pipeline for gender classification of drivers, seven different ML algorithms were investigated to tailor an energy-efficient and accurate pipeline. These algorithms were applied to a subset of the data of 8 drivers where the test scores were quantified. In this exploratory step, the algorithms were tested on how well one driver could be separated from the group as a ML classification task. In an effort to estimate the energy consumption, the full training set with a 50/50 gender distribution was used with the prescribed hyperparameters from the preceding pipeline. After evaluating the energy consumption and performance, the final gender-bias attenuated classification pipeline was deployed.

1) PREPROCESSING FINDINGS

The three preprocessing methods were evaluated before deciding on which of these to include in the final pipeline. It was observed that applying the Kalman filter estimates for the low-quality data resulted in a smoothing effect, as shown in Fig. 7. Due to this, using the Kalman filter estimates as a preprocessing step decreased the classification accuracy for the dataset at hand. Accordingly, the Kalman filter estimates were not implemented in the final pipeline. Instead, descriptive statistics imputation gave the best performance and was selected as a preprocessing step in the final ML pipeline.

2) EXPLORATORY DESCRIPTIVE STATISTIC FINDINGS

For the purpose, small scripts are written to visualize histograms, pairplots, logarithm plots and correlation maps. Meanwhile, the descriptive statistics, including frequencies and percentiles, of the data was computed continuing a previous study [24] where differences in the driving styles and genders were investigated.

An example of this is depicted in the histogram plot in Fig. 8 where the time of the day when a vehicle was used with the difference between genders are observable. For this

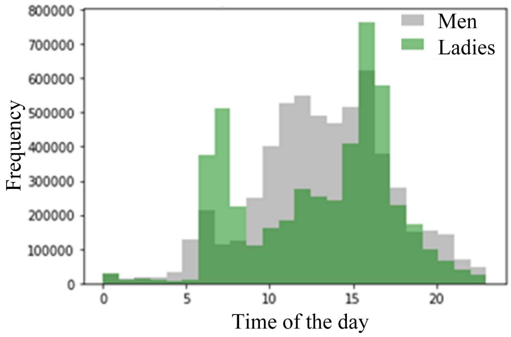

FIGURE 8. Exploratory analysis of data. It is a discernable difference among the clock time individuals are driving their vehicles for the investigated population.

snapshot of data, women have two distinctive peaks around rush hours. Men have a similar drive pattern, albeit with less distinctive peaks, but instead manifesting a preference for driving at mid-day and later during the day.

3) EXPLORATORY ML FINDINGS

Preliminary ML approaches were studied as described in Section III-B. As an unsupervised ML algorithm, PCA was investigated. PCA captures the entropy content of the distinct structures in the data of the drivers as shown in the following cumulative summation of entropies per 8 components: [0.208 0.345 0.466 0.580 0.691 0.790 0.880 0.953]. In the preceding vector, the data in each column represents the captured entropy per principal component. The more components used the higher the accuracy, however, an increased number of components also increases the HPC resources needed. Based on this, five components were found to be the optimal compromise between accuracy and resource usage.

Following the shortcoming of PCA, non-linear embedding (t-SNE) were investigated. In embedding approach, t-SNE maps two distinct regions. As can be seen in Fig. 9, a generalized decision boundary was impossible to prescribe for a binary or an eight-car subset of the investigated drivers. This behavior iterated itself for various driver pairs and embedding was not used for the final classification pipeline.

To continue, a supervised learning module for the pipeline was designed for the exploratory testing. Supervised learning was applied with different scaling, as shown in Table 2. A smaller dataset, of 8 drivers, were used for this classification step. A one-vs-all computation was made to see how well the different drivers could be separated from the group of drivers. Due to the high entropy in the input data, scaling was used, which improved the results as included in Table 2. For three of the eight drivers, the binary classification could separate the driver with classification scores well above 90%. The highest score is achieved by the MLP + StdScaler, however, RF based classifiers reach a comparable classification score. This step enabled the project to fine-tune the supervised learning hyperparameter set. Still, for two drivers, namely VCC1000 and VCC658, all ML algorithms fail to reach relatively high classification scores.

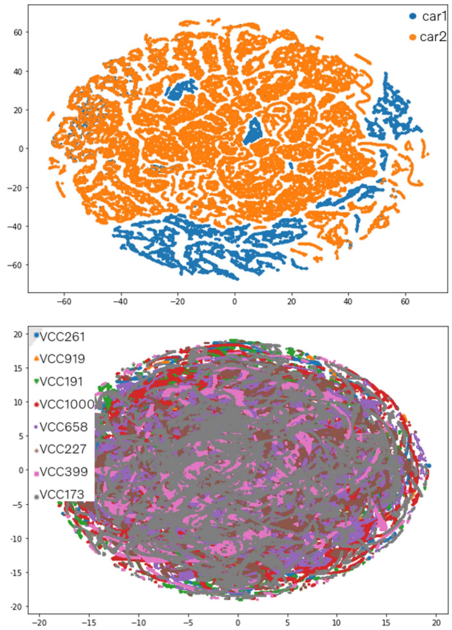

FIGURE 9. Illustration of t-SNE mapping to decrease the data to a lower dimensional space. In the first inset, two drivers with distinct driving styles are investigated. In the inset on the right, eight drivers are investigated.

TABLE 1. List of the Different Features and Their Correlation to Gender

Feature	Correlation
gender	1.00
altitude	0.24
latitude	0.14
longitude	0.14
kmeans	0.13
$log(d)$	0.11
trip	0.10
elapsed	0.05
speed	0.05
d	0.04
day	0.03
time	0.02
hdop	0.02
dist	0.0004
acc	0.0002

A. ENVIRONMENTAL BURDEN

The energy consumption for each of the ML algorithms were estimated based on metrics extracted during the exploratory analysis. The CO_2 footprint was calculated according to (3) where the TDP of the computational chip used for the different ML algorithms was recorded, together with the time for the computation and number of CPUs. Taking the total TDP of 85W distributed evenly on each core for computation of each approach and by calculating the Kolmogorov complexity and

TABLE 2. One-vs-All Classification for Testing How Well the Different Algorithms Can Differentiate One Driver From the Rest of the Group

One v. all	LR+MinMax	LR+StdScaler	SVC+StdScaler	kNN+StdScaler	MLP+StdScaler	DT	RF
VCC261	0.95	0.95	0.95	0.95	0.95	0.95	0.96
VCC919	0.99	0.99	0.99	0.99	0.99	0.98	0.99
VCC191	0.88	0.88	0.88	0.88	0.89	0.86	0.88
VCC1000	0.76	0.76	0.76	0.76	0.79	0.77	0.78
VCC658	0.71	0.71	0.71	0.71	0.75	0.70	0.71
VCC227	0.89	0.89	0.89	0.89	0.89	0.86	0.87
VCC399	0.95	0.95	0.95	0.95	0.95	0.93	0.94
VCC173	0.86	0.86	0.86	0.86	0.86	0.83	0.85

TABLE 3. Performance Cursors During Training, Calculated Kolmogorov Complexity, and CO_2 Footprint

	Complexity	RAM	Size-on-disk	Energy	CO_2	Score
	[-]	[GB]	[kB]	[J]	[g]	[-]
LR+MinMax	0.85	1.28	196	577	0.00213	0.874
LR+StdScaler	0.89	1.25	196	509	0.00188	0.874
DT	0.15	1.33	71	1683	0.00622	0.860
RF	1	1.79	93	7972	0.02950	0.873
SVC+StdScaler	0.72	1.29	165	58735	0.21700	0.874
k-NN+StdScaler	0.39	1.27	127	557	0.00206	0.874
MLP+StdScaler	0.56	1.27	168	214464	0.79200	0.889

TABLE 4. Correlation of Kolmogorov Complexity, Energy Consumption, Memory Usage, Size-on-Disk and Classification Score

	Comp.[-]	Ener.[J]	RAM[GB]	Disk[kB]	Score[-]
Comp.[-]	1.0	-0.1	0.4	0.5	0.3
Energy[J]	-0.1	1.0	-0.2	0.2	0.8
RAM[GB]	0.4	-0.2	1.0	-0.6	-0.2
Disk[kB]	0.5	0.2	-0.6	1.0	0.6
Score[-]	0.3	0.8	-0.2	0.6	1.0

correlating the complexity to the RAM consumption the algorithms were also sorted in their memory consumption, Table 3.

RF has the highest complexity value and the complexity values in Table 3 are normalized based on this. This fact may be particularly important envisioning embedded in-car applications. It can be observed that the complexity is correlated to the RAM memory utilization, as visualized in Table 4, where RF also has the highest value. Even so, the RF consumes the second least space-on-disk, only the DT lies lower. DT also has the lowest complexity, however, not the lowest RAM utilization. Despite the linear approach, the LR consumes the highest space-on-disk. This is due to the application of the algorithm for the data at hand, which requires a scaling process. This scaling step is not needed for DT, and its ensemble version, RF.

When looking at the energy consumption data, two algorithms stand out, the SVC+StdScaler and MLP+StdScaler. The classification score for the ML algorithms is also included, showing a very similar performance among the different algorithms, with the MLP as the clear winner. Similarly, according to the populated data in Table 3, one can observe that the lowest energy consumption is achieved by LR+StdScaler. This is reasonable, as the decision surface in this algorithm is linear. It is interesting to note that, LR+MinMax and k-NN+StdScaler expend the least energy.

A correlation matrix was calculated to establish the correlation among five parameters, as shown in Table 4. Here, the Kolmogorov complexity is correlated to the size-on-disk, the RAM consumption and the classification score. The energy expenditure is strongly correlated to the classification score and mildly correlated to the size-on-disk. The RAM occupation of the investigated classification algorithms is only correlated to complexity. The size-on-disk is correlated to classification score, the Kolmogorov complexity and, albeit mildly, to the energy consumption. Finally, the classification score is strongly correlated to energy consumption, size-on-disk, and Kolmogorov complexity.

The MLP+StdScaler had the highest score and energy consumption leading to the largest CO_2 footprint. The lowest energy consumption is by the LR+StdScaler, with a reasonable classification score achieved, allowing for the lowest overall CO_2 footprint. The lowest classification score is achieved by DT, however DT algorithm has the lowest size-on-disk and Kolmogorov complexity; which may justify its specific implementation for certain embedded solutions.

1) FINALIZED PIPELINE WITH 30 DRIVERS

The features used to classify the behavior of drivers were reduced using PCA to 5 components for easier handling by the computational cluster. Following several training-test data set distributions as well as swept k-fold grid searches, aiming at good distribution in the training set, a 90/10 distribution for the training/test populations was selected for this study, while the training set is folded 5-times. The correlation to

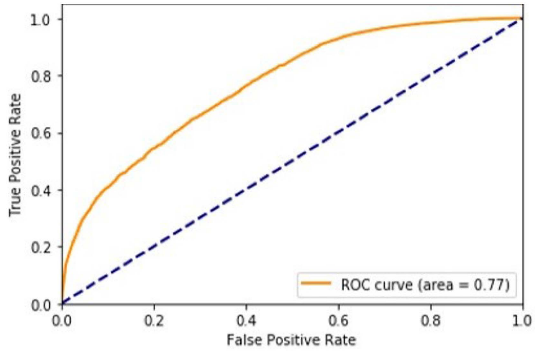

FIGURE 10. Receiver operating characteristic plot for 30 drivers for the final pipeline. These are drivers which were deliberately kept out of training and test sets.

gender for the investigated features is listed in Table 1. It was observed that various features had high correlation to gender which were related to each individual's address. This may introduce bias as affluence effects the living standards. This underscores the significance of usage of engineered features. It can be observed in Table 1 that the engineered feature k-means, which is based on an unsupervised learning step as data preparation, scored as high as address related features. The validity of this is also true for the anonymized distance d, using the Haversine formulation of a spline.

Typical hyperparameters for RF classifier following the gridsearch are listed in the following:

bootstrap=True,
class weight=None,
criterion=gini,
max depth=5,
max features=auto,
max leaf nodes=None,
min impurity decrease=0.0,
min impurity split=None,
min samples leaf=45,
min samples split=2,
min weight fraction leaf=0.0,
n estimators=10,
oob score=False,
warm_start=False.

For clustering for feature engineering purposes, representative hyperparameters are as in the following :

batch size=45,
init=k-means++,
random state=42,
n clusters=15,
n init=10,
max no improvement=10

The designed pipeline reached 77% classification accuracy for unseen drivers, which can be observed in Fig. 10.

VI. CONCLUSION

A classification pipeline with a careful distribution of driving data between two classes and usage of feature engineering technologies for driver gender is deployed. The study aimed to assure an attenuation of gender-bias prior to ML, while ensuring a sustainable deployment. Using data devoid of biometric and geographic information, the proposed pipeline will distinguish manifested binary genders with 80% accuracy for drivers out of the holdout data set.

A method for sustainable algorithm selection is proposed by calculating the CO_2 footprint. The environmental burden for seven different ML algorithms were evaluated when using the same pipeline and training data set. It was computed that all investigated classification algorithms reach to a comparable accuracy. However, the algorithms also showed large differences in the Kolmogorov complexity, RAM requirements, size-on-disk and energy consumption. The most environmentally friendly ML algorithm for the problem at hand proved to be LR+StdScaler. The widespread NN approach, MLP+StdScaler, had the highest classification score, but with the burden of the highest CO_2 footprint. In addition, the proposed method for sustainable algorithm selection can also be tuned for embedded application design, where the RAM and disk sizes are limited.

A final remark is that, the environmental burden for application of ML has only recently started to be acknowledged. Being overlooked, limited effort has been deployed for developing resource efficient un-biased algorithms. This should be given larger considerations to meet the sustainable targets of UN.

REFERENCES

[1] United Nations, "United nations document on sustainable development goals," early access: Dec. 25, 2019. [Online]. Available: https://www.un.org/
[2] K. Crawford, "Generative AI's environmental costs are soaring–and mostly secret," *Nat.*, vol. 626, no. 8000, pp. 693–693, Feb. 2024, doi: 10.1038/D41586-024-00478-X. [Online]. Available: https://www.nature.com/articles/d41586-024-00478-x
[3] K. Kirkpatrick, "The carbon footprint of artificial intelligence," *Commun. ACM*, vol. 66, no. 8, pp. 17–19, Jul. 2023, doi: 10.1145/3603746.
[4] K. Hao, "Training a single AI model can emit as much carbon as five cars in their lifetimes| mit technology review,"(n.d). [Online]. Available: https://www.technologyreview.com/2019/06/06/239031/training-a-single-ai-model-can-emit-as-much-carbon-as-five-cars-in-their-lifetimes/
[5] M. Heikkila, "We're getting a better idea of AI's true carbon footprint — mit technology review," Nov. 2022. Accessed: Non. 14, 2024. [Online]. Available: https://www.technologyreview.com/2022/11/14/1063192/were-getting-\\a-better-idea-of-ais-true-carbon-footprint/
[6] J. Gonzalez-Gonzalez, S. Garcia-Mendez, F. De Arriba-Perez, F. J. Gonzalez-Castano, and O. Barba-Seara, "Explainable automatic industrial carbon footprint estimation from bank transaction classification using natural language processing," *IEEE Access*, vol. 10, pp. 126326–126338, 2022, doi: 10.1109/ACCESS.2022.3226324.
[7] A. Dantcheva and F. Brémond, "Gender estimation based on smile-dynamics," *IEEE Trans. Inf. Forensics Secur.*, vol. 12, no. 3, pp. 719–729, Mar. 2017.
[8] Y. An, S. Meng, and H. Wu, "Discover customers' gender from online shopping behavior," *IEEE Access*, vol. 10, pp. 13954–13965, 2022.
[9] J. Buolamwini and T. Gebru, "Gender shades: Intersectional accuracy disparities in commercial gender classification," in *Proc. Conf. Fairness Accountability Transparency*, 2018, pp. 77–91.

[10] M. Babaee, L. Li, and G. Rigoll, “Person identification from partial gait cycle using fully convolutional neural networks,” *Neurocomputing*, vol. 338, pp. 116–125, 2019.
[11] A. C. B. Garcia, M. G. P. Garcia, and R. Rigobon, “Algorithmic discrimination in the credit domain: What do we know about it?,” *AI Soc.*, vol. 39, no. 4, pp. 2059–2098, 2023, doi: 10.1007/s00146-023-01676-3.
[12] R. R. Fletcher, A. Nakeshimana, and O. Olubeko, “Addressing fairness, bias, and appropriate use of artificial intelligence and machine learning in global health,” *Front. Artif. Intell.*, vol. 3, Apr. 2021, Art. no. 561802, doi: 10.1007/s00146-023-01676-3.
[13] S. Hiba and Y. Keller, “Hierarchical attention-based age estimation and bias analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 12, pp. 14682–14692, Dec. 2023.
[14] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, “Universal adversarial perturbations,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 86–94.
[15] C. Szegedy et al., “Intriguing properties of neural networks,” Dec. 2013. *arXiv:1312.6199*.
[16] R. Tatman, “Gender and dialect bias in youtube’s automatic captions,” in *Proc. 1st Workshop Ethics Natural Lang. Process.*, 2017, pp. 53–59.
[17] M. Evin et al., “Personality trait prediction by machine learning using physiological data and driving behavior,” *Mach. Learn. Appl.*, vol. 9, 2022, Art. no. 100353.
[18] T. Mohn*Forbes*, “Dummies used in motor vehicle crash tests favor men and put women at risk,” *New Report Says*, Forbes, Jun. 29, 2021. [Online]. Available: https://www.forbes.com/sites/tanyamohn/2019/10/28/dummies-used-in-motor-vehicle-crash-tests-favor-men-and-put-women-at-risk-new-report-says/
[19] C. Criado Perez, “The deadly truth about a world built for men–from stab vests to car crashes,” *Guardian*, Feb. 23, 2019. Available: https://www.theguardian.com/lifeandstyle/2019/feb/23/truth-world-built-for-men-car-crashes
[20] B. Kingma and W. van Marken Lichtenbelt, “Energy consumption in buildings and female thermal demand,” in *Proc. Nat. Climate Change*, 2015, pp. 1054–1056.
[21] K. Sehl, *Vice*, “The bizarre history of trying to design cars for women,” *Vice Media*, Apr. 4, 2019. Accessed: Mar. 11, 2024. [Online]. Available: https://www.vice.com/en/article/43jvvb/cars-for-women-design-history-volvo
[22] C. Cook, R. Diamond, J. V. Hall, J. A. List, and P. Oyer, “The gender earnings gap in the gig economy: Evidence from over a million rideshare drivers,” *Rev. Econ. Stud.*, vol. 88, no. 5, pp. 2210–2238, 2021. doi: 10.1093/restud/rdaa081.
[23] R. Aldred, R. Johnson, C. Jackson, and J. Woodcock, “How does mode of travel affect risks posed to other road users? An analysis of English road fatality data, incorporating gender and road type,” *Inj. Prevention*, vol. 27, no. 1, pp. 71–76, 2021. doi: 10.1136/injuryprev-2019-043534. [Online]. Available: https://injuryprevention.bmj.com/content/27/1/71
[24] E. Wikner, R. Orbay, S. Fogelström, and T. Thiringer, “Gender aspects in driving style and its impact on battery ageing,” *Energies*, vol. 15, no. 18, Sep. 2022, Art. no. 6791 [Online]. Available: https://www.mdpi.com/1996-1073/15/18/6791
[25] M. V. Faria, P. C. Baptista, and T. L. Farias, “Identifying driving behavior patterns and their impacts on fuel use,” *Transp. Res. Procedia*, vol. 27, pp. 953–960, Jan. 2017, doi: 10.1016/J.TRPRO.2017.12.038.
[26] J. Rios-Torres, J. Liu, and A. Khattak, “Fuel consumption for various driving styles in conventional and hybrid electric vehicles: Integrating driving cycle predictions with fuel consumption optimization*,” *Int. J. Sustain. Transp.*, vol. 13, no. 2, pp. 123–137, Feb. 2018, doi: 10.1080/15568318.2018.1445321.
[27] M. Hjäldahl, C. Ahlström, and C. Sundström, “Driving style and energy consumption with everyday use of a plug-in hybrid electric vehicle,” *EVS 31 & EVTeC*, 2018. Accessed: Oct. 15, 2021. [Online]. Available: https://www.vehicular.isy.liu.se/Publications/Articles/EVS 18 CS.pdf
[28] L. Eboli, G. Mazzulla, and G. Pungillo, “How drivers’ characteristics can affect driving style,” *Transp. Res. Procedia*, vol. 27, pp. 945–952, 2017, doi: 10.1016/j.trpro.2017.12.024.
[29] S. Karlsson, “The swedish car movement data project,” 2013. [Online]. Available: http://publications.lib.chalmers.se/records/fulltext/187380/local_187380.pdf
[30] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE:synthetic minority over-sampling technique,” *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002.
[31] X. Xiaolong, C. Wen, and S. Yanfei, “Over-sampling algorithm for imbalanced data classification,” *J. Syst. Eng. Electron.*, vol. 30, no. 6, pp. 1182–1191, 2019.
[32] D. Barbieri et al., “Poverty, gender and intersecting inequalities in the EU,” Accessed: Oct. 11, 2024. [Online]. Available: https://eige.europa.eu/sites/default/files/documents/ti_pubpdf_mh0416244enn_pdfweb_20161208181320.pdf
[33] A. B. Azcona, S. D. Valero, and T. P. Uteng, “Harsh realities: Marginalized women in cities of the developing world,” 2020. Accessed: Oct. 11, 2024. [Online]. Available: https://unhabitat.org/harsh-realities-marginalized-women-in-cities-of-the-developing-world
[34] G. W. Evans, “Discussions: Concerning haversines in plane trigonometry,” *Amer. Math. Monthly*, vol. 26, no. 2, pp. 69–72, Feb. 1919, doi: 10.2307/2973146.
[35] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
[36] D. Sculley, “Web-scale k-means clustering,” in *Proc. 19th Int. Conf. World Wide Web*, New York, NY, USA 2010, pp. 1177–1178.
[37] P. S. Panigrahy, D. Santra, and P. Chattopadhyay, “Feature engineering in fault diagnosis of induction motor,” in *Proc. 3rd Int. Conf. Condition Assessment Techn. Elect. Syst.*, 2017, pp. 306–310.
[38] D. Arthur and S. Vassilvitskii, “k-means: The advantages of careful seeding,” in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, 2007, pp. 1027–1035.
[39] European Environmental Agency, early access: Oct. 15, 2021. [Online]. Available: https://www.eea.europa.eu/data-and-maps/daviz/co2-emission-intensity-5
[40] A. S. Luccioni, S. Viguier, and A.-L. Ligozat, “Estimating the carbon footprint of BLOOM, a 176B parameter language model,” *J. Mach. Learn. Res.*, vol. 24, no. 1, Mar. 2024.
[41] K. Main, U. Ahmed, H. J. Math Bollen, and M. Alvarez, “A review of data centers energy consumption and reliability modeling,” *IEEE Access*, vol. 9, pp. 152536–152563, 2021, doi: 10.1109/ACCESS.2021.3125092.
[42] R. Landauer, “Irreversibility and heat generation in the computing process,” *IBM J. Res. Develop.*, vol. 5, no. 3, pp. 183–191, 1961, doi: 10.1147/rd.53.0183.
[43] *Intel Corp.*, early access: Oct. 09, 2021. [Online]. Available: https://ark.intel.com/content/www/us/en/ark/products/83352/intel-xeon-processor-e52620-v3-15m-cache-2-40-ghz.html
[44] N. Rohbani, M. A. Soleimani, and H. Sarbazi-Azad, “CoolDRAM: An energy-efficient and robust DRAM,” in *Proc. 2023 IEEE/ACM Int. Symp. Low Power Electron. Des.*, 2023, pp. 1–6.
[45] E. Stefansson and K. H. Johansson, “Computing complexity-aware plans using kolmogorov complexity,” in *Proc. 60th IEEE Conf. Decis. Control*, 2021, pp. 3420–3427.
[46] A. N. Kolmogorov, “On tables of random numbers,” *Theor. Comput. Sci.*, vol. 207, no. 2, pp. 387–395, 1998.

RAIK ORBAY received the B.Sc. and M.Sc. degrees in aeronautical engineering from Istanbul Technical University, Istanbul, Türkiye, the DEA in energy engineering from SUPAERO, Toulouse, France, and the Ph.D. degree in mechanical engineering from LTH, Lund, Sweden. He is currently the Technical Expert with Advanced Simulation Methods, Volvo Car Corporation - Propulsion and Energy, Gothenburg, Sweden and as a Researcher with the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden. His research interests include experimental / FEM based synthetic data and automation for sustainable engineering.

EVELINA WIKNER (Member, IEEE) receievd the B.Sc. degree in chemical engineering and engineering physics, the M.Sc. degree in nanoscience and nanotechnology, and the Ph.D. degree in electrical engineering from the Chalmers University of Technology, Gothenburg, Sweden. She is currently an Assistant Professor with the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg. Her research interests include Li-ion and Na-ion batteries, optimizing utilization and lifetime in stationary, and electromobility applications with a focus on sustainability. Her main area of expertise is in testing and modeling of Li-ion and Na-ion batteries, from detailed cell modeling to system aspects.