Probabilistic calibration of a binary classifier, applied to detecting sleeping state in a car drive

Paolo Giudici[1] and Giulia Villone[2]

[1] Corresponding author, Department of Economics and Management, University of Pavia, Via San Felice 7, 27100, Pavia, Italy. Email paolo.giudici@unipv.it

[2] FotoNation Ltd, Xperi Business Park, Block 5, Brockagh, Parkmore East, Co. Galway, H91 V0TX, Ireland. Email giulia.villone@xperi.com

Abstract. Predictions arising from deep neural networks may be very accurate but not very robust, leading to uncertainty in their outcome. This critical problem is receiving growing attention from the Machine Learning (ML) community. A practical solution that is increasingly applied is calculating confidence bounds for the ML predictions. Most confidence bounds available in the literature are theoretically sound but unfeasible from a practical viewpoint. In this paper, we contribute to the literature with probabilistic confidence bounds based on conditional probabilities, and we demonstrate their operational validity by means of a real-world application that concerns the prediction of the sleeping states of car drivers.

Keywords: Confidence measure, confidence calibration, neural networks, driver's drowsiness

1 Introduction

Modelling the uncertainty of the predictions of deep neural networks is a critical problem that is receiving growing attention from the Machine Learning (ML) community. Deep learning architectures have been proven to achieve outstanding results in various domains. However, their application in high-risk fields, such as autonomous driving, where an erroneous prediction can dramatically impact people's lives, demands that neural networks be accurate and indicate when their predictions are likely incorrect due to unexpected inputs or out-of-distribution data.

The paper by (27), provides a recent literature review on on the topic of driving and the importance of improving predictions and robustness in such a problem.

For instance, a self-driving car that uses a neural network to detect obstructions should count more on the output of other sensors for braking if the network cannot confidently predict the presence of immediate obstructions. Uncertainty modelling can help identify when the model faces such situations, enabling more robust and reliable performance.

Incorrect predictions can also lead to biased or unfair outcomes, mainly when ML models are used in sensitive domains like hiring or lending. Uncertainty modelling can help identify cases where the model may be biased or uncertain, prompting further review and intervention. Overall, modelling the uncertainty of deep neural network predictions is crucial for improving the trust, safety, fairness, reliability and interpretation of AI systems, as well as for enhancing human-AI collaboration and allowing for safer decision-making (5; 9). Uncertainty modelling is also essential in the research and development of ML models. It can guide the development of more robust architectures and training procedures, help identify weaknesses and limitations in existing models and make informed choices about which models to use. Specifically, a neural network should provide a calibrated measure of its confidence that its predictions are correct, meaning that they correspond to the ground truth (8). Estimating the reliability of networks' predictions is still an open research quest.

Probabilistic methods can be applied to obtain confidence bounds in predictions. This paper focuses on probabilistic confidence values for neural networks applied to classification problems, particularly in

detecting whether a car driver is alert or has a microsleep. The proposed calibration method presents a novel approach for complementing predictions with valid confidence measures. The advantage of the proposed methods is that confidence bounds can be calculated for individual predictions without requiring heavy computations.

2 Literature review

The goal of confidence calibration is to estimate uncertainty via matching the confidence level of a set of samples with their prediction accuracy (8; 18). For instance, a model, like a neural network, should correctly classify 90 out of 100 samples if its confidence level on such predictions is 0.9. More formally, given the input $X \in \mathcal{X}$ and label $Y \in \mathcal{Y} = \{1, \cdots, K\}$ both random variables following a ground truth joint distribution $\pi(X, Y) = \pi(Y|X)\pi(X)$, a neural network f with $f(X) = (\hat{Y}, \hat{P})$ where $\hat{Y} \in \{1, \cdots, K\}$ is a predicted class and $\hat{P}$ is its associated confidence level, *perfect calibration* can be defined as (8):

$$P(\hat{Y} = Y|\hat{P} = p) = p, \quad \forall p \in [0, 1] \tag{1}$$

Unlike those from a decade ago, the most recent neural networks are poorly calibrated (8). Depth, width, weight decay, and batch normalisation influence calibration. Hence, in practical settings, it is impossible to achieve perfect calibration. To improve calibration, scholars have proposed different solutions that can be clustered into scaling-based, binning-based, similarity-based and Bayesian-based methods. *Scaling-based methods* adjust the probability returned by a model that an input belongs to an output class by learning one or more scalar parameters so that this probability accurately represents the likelihood of that particular class. Standard methods for confidence calibration in the classification domain are Platt scaling (24), beta calibration (12) and temperature scaling methods (8). *Binning-based methods* divide samples into multiple bins based on confidence and calibrate each bin. Popular binning-based methods include Bayesian Binning into Quantiles (BBQ) (19), histogram binning (31) and an Ensemble of Near Isotonic Regression (20). However, existing calibration methods fail to see the proximity bias issue, the tendency of models to be overly confident in low proximity samples (samples lying in sparse density regions of the input space) than high proximity ones. Thus, models suffer from inconsistent miscalibration, limiting the capabilities of calibration methods to deliver reliable and interpretable uncertainty estimates (30). *Similarity-based methods* estimate the confidence level based on the output class of the instances in the input dataset that are closer (or more similar) to the test sample. For instance, the method proposed in (1) estimates confidence levels using a non-conformity measure, calculated as the average k-neighbour proximity for all the samples in the same class predicted by the model for a given sample under analysis, to indicate how 'atypical' this sample is relative to the other samples.

Bayesian-based methods quantify the uncertainty related to inputs and parameters' calibration via a posterior distribution of the model's parameters, which balances the prior probability of the parameters with the likelihood function learned from the available data and also enables accurate uncertainty quantification on the Bayesian methods are a common tool to provide a mathematical framework for uncertainty estimation (6; 10). However, exact Bayesian inference is not tractable in deep neural networks due to its sophisticated implementation and high computational cost (26). Furthermore, these methods are often harder to scale and can suffer from sub-optimal performance (2). Scholars have proposed many techniques to approximate the intractable posterior distributions derived by Bayesian inference for neural networks (3; 7; 15; 29), a popular one being Markov Chain Monte Carlo (MCMC) (21). Monte Carlo simulations were exploited in (11) to estimate data uncertainty, the amount of noise inherent in the input data distribution. The estimated uncertainty is fed into the loss function of the neural network. The authors demonstrated that this uncertainty-based loss function improves the model's calibration. More recently, stochastic gradient versions of MCMC were also proposed to allow scalability (4; 16; 28). There have also been efforts to approximate the posterior with Laplace approximations (25) and with related approaches, such as the Stochastic Weight Averaging (SWA)-Gaussian, which performs Gaussian posterior

approximation using the SWA algorithm. However, all these methods are often computationally expensive and sensitive to the choice of hyperparameters.

On a different line of research, (14) presented a framework to measure and calibrate biased (or miscalibrated) confidence estimates of object detection methods by including additional bounding box information from the detector. This was followed by Bayesian confidence calibration, a framework to obtain calibrated confidence estimates in conjunction with the uncertainty of the calibration method (13) by treating each model's parameter in a Bayesian way. Bayesian neural networks are neural networks that use distributions instead of weights for inference, thus indicating the network's uncertainty about a specific prediction. The authors transferred this idea of using distributions to confidence calibration. For this purpose, each model's parameter, or weight in the case of a neural network, is replaced by a normal distribution. This allows obtaining a single calibrated estimate for a single prediction and a sample distribution indicating the epistemic uncertainty about the current prediction. Similarly, (26) proposed calibrating a single sample's prediction accuracy and confidence in deep neural networks through stochastic inferences. They interpreted stochastic regularization using a Bayesian model. They showed that the network's predictive uncertainty and the variance of the prediction scores obtained by stochastic inferences for that sample are highly correlated with the variance of multiple inferences given by stochastic depth or dropout. Motivated by the findings, the authors designed a variance-weighted and confidence-integrated loss function composed of two cross-entropy loss terms w.r.t. ground truth and uniform distribution balanced by the variance of stochastic prediction scores. Deep neural networks trained with this loss function predict confidence-calibrated scores using a single inference.

Finally, Conformal Prediction (CP) is a framework for assessing the uncertainties of AI systems. Given a sample, CP returns a prediction interval in regression problems and a set of classes in classification problems guaranteed to cover the true value with high probability. However, CP is computationally inefficient as it requires retraining a model over a calibration set containing $n + 1$ samples w.r.t. the previous iteration (22). CP becomes unfeasible when coupled with neural networks that require long training times.

2.1 Gaps in the existing knowledge

Existing methods are computationally burdensome as they require to retrain the neural networks. Some real-world applications, such as autonomous-driving cars, need lightweight neural networks as the computational resources are required to process the signals received from various sensors, including radars and cameras. Furthermore, most of these methods are complex, thus hard to understand, explain, and debug. Explanations for model-based predictions should be supplemented for process control and quality certification by auditors.

3 The proposed calibration method

The proposed calibration method was inspired by the request of a car manufacturer to show an indicator of a driver's state (either alert or microsleep) augmented by confidence levels in the predictions made by a neural network.

We uderline that the request of the car manufacturer involved two main aspects, which condtion the type of methodology to be employed: a) to develop a methodology that does not require heavy computations; b)to develop a methodology that is understandable, especially in terms of incremental innovations with respect to existing practices. These conditions led us to develop a methdology that, while mathematically sound, is also simmple to understand and implement.

More precisely, we assume that, to be effective and usable in an automotive electronic system, the method generating this indicator had to meet the following requirements:

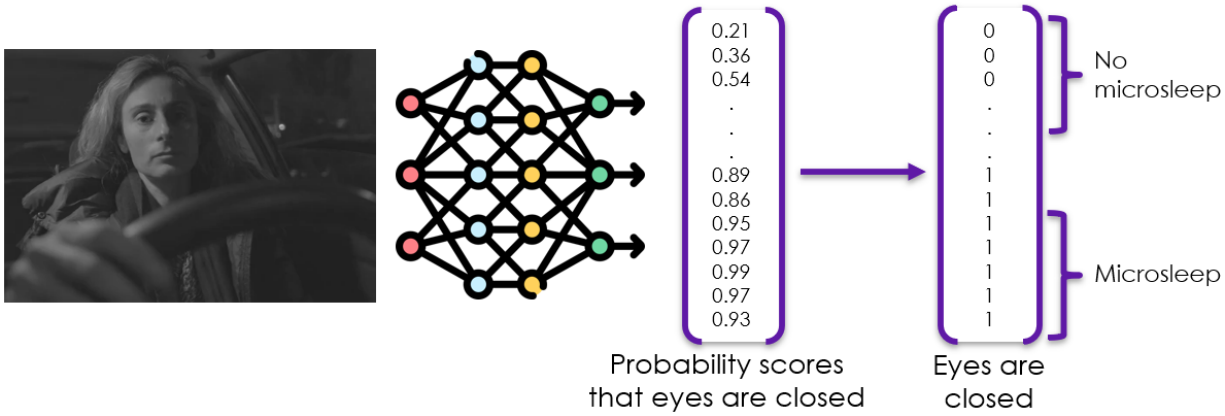
– Be inspectable, comprehensible and explainable to ensure that it meets safety and quality standards.

– Return two confidence levels, depending on whether the driver is/is not in a microsleep state.
– Be calculated and displayed in real-time; hence, its computation cannot be resource greedy.

The neural network does not directly predict microsleep states, but it processes one video frame at a time and returns a probability score on whether the eyes of the driver are open or closed. The eyes are considered open if the probability score is lower than 80%; above this threshold, eyes are considered closed. This threshold was determined by maximising the prediction accuracy of the network on an Xperi proprietary dataset containing several videos of people driving at a simulator at different times of the day and night.

When the eyes are predicted as open, the driver is in a no-microsleep, or alert, state. A microsleep starts after 15 consecutive frames are labelled as "eyes closed" and ends when either 1) there occurs a frame labelled as "open eyes", or 2) 3,000 consecutive frames are labelled as "closed eyes". In the latest case, the microsleep state is updated to "sleep"; the sleep state is not considered in this paper, but the proposed method can be easily extended to return the network's confidence levels of the sleep state. Figure 1 shows diagrammatically how the system works, from an input frame to its output, which consists of an "alert" or "microsleep" label for each frame.

Fig. 1: Diagrammatic view of the AI system structure developed to predict a driver's alert/microsleep states based on a neural network that returns the probability of whether the driver's eyes are closed. If this probability is higher than 80%, eyes are considered closed. A microsleep consists of 15 consecutive frames with eyes closed.



Noticeably, the proposed calibration method is not a neural network approach but a statistical method based on the behaviour of a neural network that was trained to assess if the eyes of a driver are open or closed. This method considers the confusion matrices to report how many instances are wrongly labelled as open or closed eyes and, subsequently, are assigned to the alert or microsleep state. Confusion matrices represent a performance measurement for machine learning classification problems. They consist of four cells, organised into two rows and two columns, reporting the combinations of predicted and actual values. The false positive and negative rates reported in the confusion matrices provide an insight into how often the network confuses the two output classes. The assumption underlying the proposed method is that these rates suffice to assess the confidence level of a network's prediction without running Montecarlo simulations or retraining the network on a new calibration dataset. In this case, the positive cases correspond to eyes

open and no microsleep, and an example of the two confusion matrices calculated over the network's predictions is reported in Tables 1 and 2.

Table 1: Confusion matrix of the predictions made by the neural network on whether the eyes of a driver are open or closed.

		Predicted condition	
		Open eyes	Closed eyes
Actual condition	Open eyes	19,501	499
	Closed eyes	580	19,240

Table 2: Confusion matrix of the predictions on whether the driver is having or not a microsleep, based on the number of consecutive frames classified as "eyes closed".

		Predicted condition	
		Microsleep	Alert
Actual condition	Microsleep	18,745	1,255
	Alert	2,137	17,863

The number of instances can be transformed into probabilities of correctly or wrongly classifying frames by dividing each value by the sum of its column. For example, the values in the first row of the open/closed eyes confusion matrix are divided by $19,501 + 580 = 20,081$. The resulting normalised confusion matrices are reported in Tables 3 and 4.

Table 3: Normalised Confusion matrix of the predictions made by the neural network on whether a driver's eyes are open or closed.

		Predicted condition	
		Open eyes	Closed eyes
Actual condition	Open eyes	97.11%	2.53%
	Closed eyes	2.89%	97.47%

Table 4: Normalised confusion matrix of the predictions on whether the driver is having or not a microsleep'.

		Predicted condition	
		Microsleep	Alert
Actual condition	Microsleep	89.77%	6.56%
	Alert	10.23%	93.44%

The confidence levels of the predictions made by the system on the microsleep states are based on the conditional probabilities of the eyes being open or closed when predicted as such and, subsequently, that the driver is or is not in a microsleep state. The calculations are based on conditional probabilities, which assess the probability of an event based on prior knowledge of conditions possibly related to the

event. In this case, the prior conditions are the true and false positive/negative rates derived from the two confusion matrices and the probability scores that the eyes are closed returned by the neural network as output. Continuing on the example of the two above confusion matrices, let us assume that the network has returned a 45% probability that the eyes are closed in an input frame. In this scenario, the eyes are considered open. Thus, the driver is alert. The confidence level is calculated as the sum of the probability that the eyes are open or closed returned by the neural network multiplied by the true and false positive rates, respectively, given that the network has predicted them as such (see equation 2).

$$P(O) = P(\hat{O})P(O|\hat{O}) + P(\hat{C})P(O|\hat{C}) \tag{2}$$

where $P(\hat{O})$ is the probability that the eyes are open estimated by the neural network on a frame and $P(\hat{O}) = 1 - P(\hat{C})$. $P(O|\hat{O})$ and $P(O|\hat{C})$ are the true and false positive rates as per the eyes open/close confusion matrix, respectively. Similarly, the confidence level that the eyes are truly closed $P(C)$ can be calculated per equation 3.

$$P(C) = P(\hat{C})P(C|\hat{C}) + P(\hat{O})P(C|\hat{O}) \tag{3}$$

where $P(C|\hat{C})$ and $P(C|\hat{O})$ are the true and false negative rates as per the eyes open/close confusion matrix, respectively.

The confidence level that the driver is truly alert ($P(A)$) follows the same logic as the confidence level calculated for the eyes open/close. The probability scores correspond to the probability that the eyes are open calculated in the previous step (see equation 4).

$$P(A) = P(O)P(A|\hat{A}) + (1 - P(O))P(A|\hat{M}) \tag{4}$$

where $P(A|\hat{A})$ and $P(A|\hat{M})$ are the true and false positive rates as per the microsleep confusion matrix, respectively.

The confidence level that the driver is truly having a microsleep differs from the previous cases because the microsleep probability scores correspond to the probability that the eyes are closed, calculated in the previous step, raised to the power of the number of frames that are missing to reach the microsleep state (see equations 5 and 6). For instance, if the network has assigned the label "eyes closed" to just three consecutive frames, $P(C)$ must be raised to the power of 12 because the microsleep state starts only after 15 consecutive frames are labelled as "eyes closed". This corresponds to the probability of independently randomly sampling 12 frames labelled as "eyes closed" by the network where $P(O)$ of the last frame is the best estimate of the probability that the following frames will belong to the same class as it is impossible to know what probability scores returned by the network for these frames.

$$P(M) = P(\hat{M})P(M|\hat{M}) + (1 - P(\hat{M}))P(M|\hat{A}) \tag{5}$$

where $P(M|\hat{M})$ and $P(M|\hat{A})$ are the true and false negative rates as per the microsleep confusion matrix, respectively.

$$P(\hat{M}) = P(C)^{(15-F_{CE})} \tag{6}$$

where $F_{CE} = \sum_{0 \leq n \leq 14} 1_{ClosedEyes}$ represents the number of consecutive frames (up to 15) labelled as "eyes closed" by the neural network.

4 The experiment

The proposed method was applied to the public dataset Night-Time Yawning-Microsleep-Eyeblink-driver Distraction (NITYMED)[3] (23). NITYMED contains 21 videos with a Frame Per Second (FPS) rate of 25,

[3] https://datasets.esdalab.ece.uop.gr/NITYMED

lasting approximately 2 minutes, of drivers in real cars under nighttime conditions. The drivers talk, look around and have microsleeps. The videos have been captured by a camera mounted on the car's dash. The participants were 11 males and eight females with different features such as hair colour, beard, and glasses.

The NITYMED videos had to be pre-processed as follows before being fed into the microsleep model:

1. The videos had to be split into single frames as the model works on one frame at a time.
2. The frames were converted from RGB to grey-scaled, single-channel images.
3. Each image was fed into a face-detection network to crop the driver's face from each frame.
4. The cropped images were fed into the microsleep model to identify which images represent microsleep events.

The NITYMED's videos and their frames are not labelled, so it was necessary to calculate the accuracy and the confusion matrices of the microsleep model. The ground truth labels were created by applying a key point detector to extract the facial landmarks of each frame and calculate the Eyes Aspect Ratio (EAR). It was decided that the driver's eyes are considered closed when EAR is below 20%. The frames classified as "eyes closed" were visually inspected to ensure that this threshold was not too high, thus labelling as such frames where the eyes are evidently open. This was not the case, so the threshold was considered appropriate for this experiment, whose focus is not to design an accurate model for detecting driver's drowsiness but to prove the validity of the proposed calibration method.

We remark that the described labelling procedure provides an "expert-based ground truth, which is not objective. This is the case in many other applications of machine learning, in whcih amodel is assessed not against an "objective" truth, but a subjective one. This does not alter the generality of the proposed method.

Fig. 2: Examples of a female and male driver of the NITYMED dataset with eyes open and closed.

The processed frames of the NITYMED videos were fed into a neural network based on a Shufflenet (17) backbone, trained to asses whether the driver's eyes are open or closed. The confidence levels of the microsleep model were calculated on the predictions made on each frame of the NITYMED dataset. The method returns two numbers representing the confidence levels of the model on the alert and microsleep state. A video[4] shows examples of microsleeps detected by the model and its levels of confidence in predicting either the alert or microsleep states.

The microsleep model reaches a prediction accuracy of 88.1% in classifying the frames of the NITYMED videos as eyes open/closed, where the prediction accuracy of the alert or microsleep states is 95.2%. Tables 5 and 6 report the confusion matrices computed by comparing the number of NITYMED frames classified

[4] https://vimeo.com/870309458https://vimeo.com/870309458

as eyes open/closed and, subsequently, alert/microsleep by the neural network and with the EAR that was considered as ground truth. Noticeably, the true negative rates of the neural network are 78% and 74% in predicting eyes open/close and the alert/microsleep stats, which are quite low and are expected to significantly impact the resulting confidence levels on the alert or microsleep states. It is interesting to see whether the results meet such expectations.

Table 5: Confusion matrix of the network's predictions on whether the eyes of a NITYMED driver are open or closed.

		Predicted condition	
		Open eyes	Closed eyes
Actual condition	Open eyes	46,682 (89%)	1,289 (22%)
	Closed eyes	5,674 (11%)	4,683 (78%)

Table 6: Confusion matrix of the network's predictions on whether a driver of the NITYMED dataset is having or not a microsleep'.

		Predicted condition	
		Alert	Microsleep
Actual condition	Alert	54,498 (96%)	367 (26%)
	Microsleep	2,433 (4%)	1,030 (74%)

5 Results

The confidence levels of the microsleep model in predicting whether the NITYMED videos show microsleep events or not were calculated per frame. Table 7 contains the number of microsleep events predicted by the neural network and detected by calculating the EAR (considered as ground truth) with some summary statistics, namely the min, max, average and median length of these events calculated by number of frames. Noticeably, the network predicted slightly less than 50% of the microsleeps detected with the EAR, but this is due to the high threshold (20%) that was used to determine when the driver's eyes are closed, based on their EAR. This gap can be easily closed by reducing the threshold. A further inspection of the frames classified as "eyes closed" with this threshold highlighted a few where the eyes are still partially open (but it is possible to see most of the eyelids). Whether these frames should be classified as eyes closed or eyes open is a subjective opinion. Furthermore, this threshold allowed testing of the proposed calibration method under suboptimal conditions where the network's accuracy is not high. This is the typical situation where the network should not always be trusted, and the confidence levels can support and improve a decision-making process. On the other hand, there are no significant differences in the four summary statistics. Those microsleeps predicted by the network coincide with microsleeps determined by the EAR.

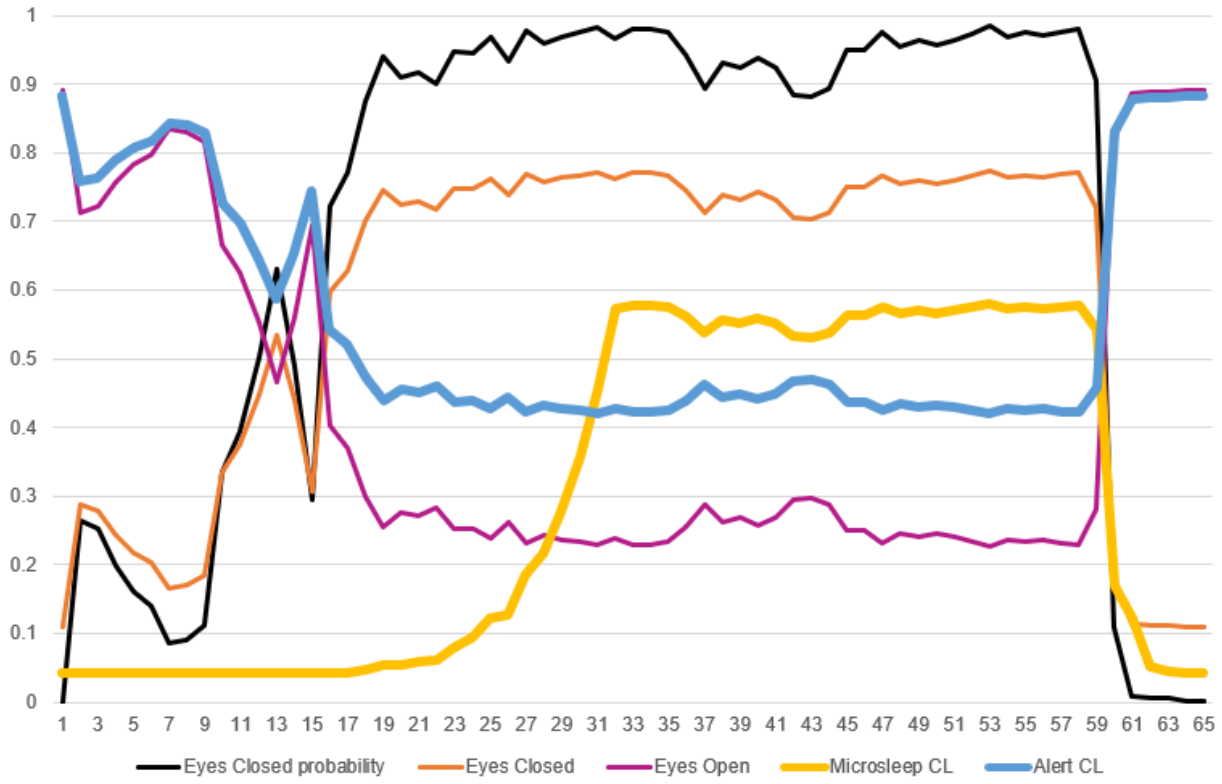
Figure 3 shows the confidence levels of the alert and microsleep statuses of a NITYMED driver as detected by the microsleep model. The confidence level of the microsleep state remains constantly low until the network returns a frame with closed eyes, and it quickly increases as the number of consecutive eyes-closed frames increases. However, this confidence level never goes above 60% even when the number of consecutive eyes-closed frames is far higher than 15, and the network assigns high probability scores to these frames that the eyes are closed, meaning that the chances that the driver is truly having a microsleep

Table 7: Summary statistics about the microsleep events predicted by the neural network and detected by calculating the EAR (considered as ground truth).

Statistics	Ground truth	Predicted
Number of microsleeps	185	87
Min length (# frames)	1	1
Max length (# frames)	117	110
Avg length (# frames)	19	16
Median length (# frames)	11	9

are pretty high. This is, as expected, due to the combined effect of the low eyes open/close and microsleep true negative rates, which are 78.4% and 74%, respectively.

Fig. 3: Example of alert/microsleep confidence levels calculated on the frames representing a microsleep event in a NITYMED video.



One of the desired requirements for a confidence level assessment method, like the proposed one, is to compute calibrated uncertainty estimates. However, it was not expected that this method would meet this requirement as the confidence levels for the microsleep state are computed using confusion matrices that consider the errors made by the network throughout the entire dataset. This assumption was tested by binning the frames of the NITYMED videos according to their microsleep confidence levels

and checking whether these levels match the prediction accuracy. The results are reported in Table confirm this assumption. When the confidence level for the microsleep state is below 50%, only one frame out of 57,000 was correctly labelled as microsleep by the network. Conversely, the prediction accuracy is higher than the confidence levels when they are in the range 50-58%. Confidence levels and prediction accuracy tend to match on the two extreme tails of the data distribution. The confidence levels cannot be higher than 59%, and, correspondingly, 61% of the frames are correctly labelled as showing a microsleep event. And the model does not assign the microsleep label to any frame with confidence level close to 0%. This issue could be easily overcome by extracting other confusion matrices for the frames with mid-range confidence level (these are the frames where the eyes are not fully open or closed) and calculate the confidence levels with these matrices.

Table 8: Number of NITYMED frames sharing the same confidence level for a microsleep event and the prediction accuracy reached by the AI system on these sets of frames.

Confidence level	# frames	Prediction accuracy
$< 50\%$	56,927	0%
50%	24	63%
51%	35	77%
52%	42	76%
53%	63	79%
54%	96	79%
55%	107	83%
56%	140	84%
57%	184	81%
58%	385	72%
59%	320	61%

We conclude this section with a comparison between our methods and others available in the literature, in the light of the obtained results. Bayesian methods are based on the logic of probability, and are attractive for our calibration problem, especially for their mathematical coherence. However, they are difficult to scale, and their actual implementation has a high computational costs. Our proposed method is indeed related to Bayesian methods, as it also uses the logic of probability, but with a more pragmatic and realistic approach, which can scale. Conformal prediction methods are instead based on simulation and sampling, the "dual" of probability laws. They are also very attractive, especially as they can provide simulation based confidence intervals which can be directly be used for calibration. However, they also have a large computational cost, requiring a multiple retraining of the underlying neural network model, which is not possible in our context.

6 Conclusions

In the paper, we have presented a probabilistic method to calibrate the predictions arising from machine learning methods.

We have demonstrated its operational validity by means of a real-world application that concerns the prediction of the sleeping states of car drivers.

The proposed calibration method has returned a reliable estimate of the confidence level of the predictions made by a neural network that considers the true and false positive/negative rate to assess the network's confidence. This method brings the following advancements compared to other calibration methods:

– It is simple to implement
– It is comprehensible.
– It is not resource-greedy and does not require high computation power.

We remark that our analysis is conditional on the available data. If data allow, further analysis can be entertained. For example, in the paper we have assumed that the cost of type I and type II errors is the same. If a cost function were known we could incorporate it in our model. Similarly, if more data on the drivers were known, we could assess other aspects, such as gender fairness of the proposed algorithm.

Future research directions include testing this method on datasets containing data from other application domains than autonomous driving cars. Theoretically, the proposed calibration method is model agnostic and should be applicable to other learning algorithms, such as support vector machines. From a methodological viewpoint, it would be important to examine how the proposed confidence bounds change when the two prediction errors in the confusion matrix are assigned different costs.

Acknowledgements The paper results from a close collaboration between the two authors. However, while GV mainly processed the data and wrote the paper, PG worked on the methodology and supervised the writing. We acknowledge support from the referees.

Bibliography

[1] S. Bhattacharyya (2011). Confidence in predictions from random tree ensembles. *2011 IEEE 11th International Conference on Data Mining*, 71–80

[2] L. Blier, and Y. Ollivier (2018). The description length of deep learning models. *Advances in Neural Information Processing Systems*, **31**

[3] C. Blundell, J. Cornebise, K. Kavukcuoglu, D. Wierstra (2015). Weight uncertainty in neural network. *International conference on machine learning*, 1613–1622. PMLR

[4] T. Chen, E. Fox, and C. Guestrin (2014). Stochastic gradient Hamiltonian Monte Carlo. *International conference on machine learning*, 1683–1691. PMLR

[5] I. Cortés-Ciriano, and A. Bender (2018). Deep confidence: a computationally efficient framework for calculating reliable prediction errors for deep neural networks *Journal of chemical information and modeling*, **59**(3), 1269–1281

[6] Y. Gal, R. Islam, and Z. Ghahramani (2017). Deep Bayesian active learning with image data. *International conference on machine learning*, 1183–1192. PMLR

[7] A. Graves (2011). Practical variational inference for neural networks. Advances in neural information processing systems **24**

[8] C. Guo, G. Pleiss, Y. Sun, and K.Q. Weinberger (2017). On calibration of modern neural networks. *International conference on machine learning*, 1321–1330. PMLR

[9] B. Ji, H. Jung, J. Yoon, and K. Kim (2019). Bin-wise temperature scaling (bts): Improvement in confidence calibration performance through simple scaling techniques. *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 4190–4196. IEEE

[10] A. Kendall, and Y. Gal (2017). What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, **30**

[11] D.H. Kim, C. Lee, and K.S. Chung (2020). A confidence-calibrated moba game winner predictor. *2020 IEEE Conference on Games (CoG)*, 622–625 IEEE

[12] M. Kull, T. Silva Filho, and P. Flach (2017). Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. *Artificial Intelligence and Statistics*, 623–631. PMLR

[13] F. Küppers, J. Kronenberger, J. Schneider, and A. Haselhoff (2021). Bayesian confidence calibration for epistemic uncertainty modelling. *2021 IEEE Intelligent Vehicles Symposium (IV)*, 466–472. IEEE

[14] F. Küppers, J. Kronenberger, A. Shantia, and A. Haselhoff (2020). Multivariate confidence calibration for object detection. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 326–327

[15] B. Lakshminarayanan, A. Pritzel, and C. Blundell (2017). Simple and scalable predictive uncertainty estimation using deep ensembles *Advances in neural information processing systems*, **30**

[16] Y.A. Ma, T. Chen, and E. Fox (2015). A complete recipe for stochastic gradient MCMC. *Advances in neural information processing systems*, **28**

[17] N. Ma, X. Zhang, H.T. Zheng, and J. Sun (2018). Shufflenet v2: Practical guidelines for efficient CNN architecture design. *In Proceedings of the European conference on computer vision (ECCV)*, 116–131.

[18] M. Minderer, J. Djolonga, R. Romijnders, F. Hubis, X. Zhai, N. Houlsby, D. Tran, and M. Lucic (2021). Revisiting the calibration of modern neural networks. *Advances in Neural Information Processing Systems*, **34**, 15682–15694

[19] M.P. Naeini, G.F. Cooper, and M. Hauskrecht (2015). Obtaining well calibrated probabilities using Bayesian binning. *Proceedings of the AAAI conference on artificial intelligence*, **29**(1)

[20] M.P. Naeini, and G.F. Cooper (2016). Binary classifier calibration using an ensemble of near isotonic regression models. *2016 IEEE 16th International Conference on Data Mining (ICDM)*, 360–369. IEEE

[21] R.M. Neal (2012). *Bayesian learning for neural networks*, **118** Springer Science & Business Media

[22] H. Papadopoulos, V. Vovk, and A. Gammerman (2007). Conformal prediction with neural networks. *In 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007)*, **2**, 388–395. IEEE

[23] N. Petrellis, S. Zogas, P. Christakos, P. Mousouliotis, G. Keramidas, N. Voros, and C. Antonopoulos (2021). Software acceleration of the deformable shape tracking application: How to eliminate the eigen library overhead. *Proceedings of the 2021 European Symposium on Software Engineering*, 51–57

[24] J. Platt (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers* ,**10**(3), 61–74

[25] H. Ritter, A. Botev, and D. Barber (2018) A scalable Laplace approximation for neural networks. *6th International Conference on Learning Representations, ICLR 2018-Conference Track Proceedings*, **6**.

[26] S. Seo, P.H. Seo, and B. Han (2019). Learning for single-shot confidence calibration in deep neural networks through stochastic inferences. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9030–9038

[27] Vellamattathil, T.B., Sotoudeh, S.M., HomChaudhuri, B. (2023). Data-Driven Prediction and Predictive Control Methods for Eco-Driving in Production Vehicles *IFAC-PapersOnLine*, 55 (37), 633-638.

[27] M. Welling, and Y.W. Teh (2011). Bayesian learning via stochastic gradient Langevin dynamics. *Proceedings of the 28th international conference on machine learning (ICML-11)*, 681–688. Citeseer

[28] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He (2017). Aggregated residual transformations for deep neural networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1492–1500

[29] M. Xiong, A. Deng, P.W. Koh, J. Wu, S. Li, J. Xu, and B. Hooi (2023). Proximity-Informed Calibration for Deep Neural Networks. *arXiv preprint arXiv:2306.04590.*

[30] B. Zadrozny, and C. Elkan (2001). Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. *Icml*, **1**, 609–616