

Bachelor's Thesis

Literature search

March 14, 2023

Error perception classification in Brain-Computer Interfaces using CNN [2]

The main goal of this paper was to implement a CNN to predict error potentials which could be used for error-correction in Brain Computer Interfaces (BCI's) Error-related Potentials (ErrP) have been used to include the perception of subjects on errors caused by the BCIs ErrP originate from the anterior cingulate cortex and is elicited as a response to error perception after a feedback presentation It used the publicly available dataset from the BNCI Horizon 2020 project website under the name Monitoring error-related potentials. The data was generated by subjects who monitored a moving cursor without controls. There was a 20Only preprocessing was a 1 to 10hz bandpass filter All 64 channels were used. This gave better results than 2 selected channels (FCZ and FZ)

Estimating the Mean and Variance of the Target Probability Distribution [7]

The main aim of this paper is to develop a feed-forward network which can also predict it's uncertainty in a regression task. To achieve this, they created a network with two heads, each head having its individual hidden layers. They only share the input layer, without any shared hidden layers. For this particular task, it showed better performance. First, only the weights of the y-headed side (the actual regression task) were trained, and the σ^2 (Variance) head was ignored. The reasoning behind this is to reduce the computing load, since training the variance predictor does not make sense, if the actual regression task is nowhere near accurate. This paper was written in '91, so computing power was quite scarce. In this age, this step might not be necessary, since computing power is (usually) not extremely scarce anymore. Unless it has a serious effect on performance, which is unknown.

After the regression task has trained for a sufficient amount, and it progress is starting to flatten off, all weights are trained. Including the variance predictor weights. The weights are updated using the following equations:

Where η is the learning rate. D_i is the true pattern, $y(x)$ is the prediction, and s_2 is the predicted variance. The aim of to adjust the weights of the this predictor to also predict this variance. This paper is aimed for regression tasks, where there is a clear difference between the predicted continuous value and the actual continuous value. In a classification environment this becomes more challenging, since this clear difference is no more.

What uncertainties do we need in bayesian deep learning for computer vision [5]

Deep learning models have two major types of uncertainty. The first one being epistemic uncertainty. This is the uncertainty of the model parameters. This can often be explained away given enough data. The other type is aleatoric uncertainty. This is noise inherent to the observations, such as faulty sensors. The latter can be further categorized into homoscedastic uncertainty, which is constant regardless of the

input, and heteroscedastic uncertainty, which depends on the inputs. The latter sparks the most interest for my research topic. These uncertainties can be modeled using Bayesian deep learning approaches. To capture epistemic uncertainty in NNs a prior distribution is put over all weights, such as a Gaussian distribution. These are called Bayesian Networks. To model this uncertainty, they try to capture how much these weights vary on the data. For the aleatoric uncertainty, a distribution is placed over the output of the model, and then try to learn the noise's variance based on the inputs.

The paper suggests to use a single network with its head split to predict both \hat{y} and $\hat{\sigma}^2$. The model does need 'uncertainty labels' to learn the uncertainty. Instead, it infers it from the loss function during the regression task. To make the model work for classification tasks there needs to be some adjustments. In their example, the model predicts a vector of unaries, which, when passed through a softmax function, forms a probability vector. They place a Gaussian distribution over this vector. Then they approximate through Monte Carlo integration, and sample unaries through the softmax function.

All formulas and a better description can be found in the paper in section 3. This section is almost too dense to properly summarize.

On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks [9]

Further possible readings

- Uncertainty quantification for multilabel text classification [1]
- Uncertainty estimation in medical image classification: systematic review [6]
- BayesNetCNN: incorporating uncertainty in neural networks for image-based classification tasks [3]
- Confidence estimation methods for neural networks: A practical comparison [8]
- Practical uncertainty quantification for brain tumor segmentation [4]

References

- [1] Wenshi Chen, Bowen Zhang, and Mingyu Lu. Uncertainty quantification for multilabel text classification. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(6):e1384, 2020.
- [2] J Rafael Correia, J Miguel Sanches, and Luca Mainardi. Error perception classification in brain-computer interfaces using cnn. In *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 204–207. IEEE, 2021.
- [3] Matteo Ferrante, Tommaso Boccatto, and Nicola Toschi. Bayesnetcnn: incorporating uncertainty in neural networks for image-based classification tasks. *arXiv preprint arXiv:2209.13096*, 2022.
- [4] Moritz Fuchs, Camila Gonzalez, and Anirban Mukhopadhyay. Practical uncertainty quantification for brain tumor segmentation. In *Medical Imaging with Deep Learning*, 2021.
- [5] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

- [6] Alexander Kurz, Katja Hauser, Hendrik Alexander Mehrtens, Eva Krieghoff-Henning, Achim Hekler, Jakob Nikolas Kather, Stefan Fröhling, Christof von Kalle, and Titus Josef Brinker. Uncertainty estimation in medical image classification: systematic review. *JMIR Medical Informatics*, 10(8):e36427, 2022.
- [7] David A Nix and Andreas S Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 ieee international conference on neural networks (ICNN'94)*, volume 1, pages 55–60. IEEE, 1994.
- [8] Georgios Papadopoulos, Peter J Edwards, and Alan F Murray. Confidence estimation methods for neural networks: A practical comparison. *IEEE transactions on neural networks*, 12(6):1278–1287, 2001.
- [9] Maximilian Seitzer, Arash Tavakoli, Dimitrije Antic, and Georg Martius. On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks. *arXiv preprint arXiv:2203.09168*, 2022.