Technical Sciences

УДК 004.89

Hvozdiev Valerii
Master Student of the
Odessa I.I. Mechnikov National University
Гвоздєв Валерій Дмитрович
студент
Одеського національного університету імені І.І. Мечникова
Гвоздев Валерий Дмитриевич
студент
Одесского национального университета имени И. И. Мечникова

Hvozdieva Tetiana
Master Student of the
Odessa I.I. Mechnikov National University
Гвоздєва Тетяна Юріївна
студентка
Одеського національного університету імені І.І. Мечникова
Гвоздева Татьяна Юрьевна
студентка
Одесского национального университета имени И. И. Мечникова

Strakhov Yevhen
PhD in Physics and Mathematics, Associate Professor
Odessa I.I. Mechnikov National University
Страхов Євген Михайлович
кандидат фізико-математичних наук, доцент
Одеський національний університет імені І.І. Мечникова
Страхов Евгений Михайлович
кандидат физико-математических наук, доцент
Одесский национальный университет имени И. И. Мечникова

UNCERTAINTY ESTIMATION AND USAGE FOR DEEP LEARNING MODELS
ВИМІРЮВАННЯ ТА ВИКОРИСТАННЯ НЕПЕВНОСТІ ДЛЯ МОДЕЛЕЙ ГЛИБИННОГО НАВЧАННЯ
ИЗМЕРЕНИЕ И ИСПОЛЬЗОВАНИЕ НЕУВЕРЕННОСТИ ДЛЯ МОДЕЛЕЙ ГЛУБОКОГО ОБУЧЕНИЯ

Summary. *The default DL approaches in ML tend to output only prediction, but not an uncertainty measure alongside the prediction. There are several approaches to DL model modification that allow deciding if the model can be trusted. The approaches vary by computational load and performance considering given constraints. In the real world project, it is often not possible to modify the model or perform retraining to apply common uncertainty estimation techniques (black box problem).*

In the first part of the paper, we aim to measure the uncertainty of the model in practice. We have researched tolerable perturbations as a way to enforce noise in the input data. A framework was built that acts as a compound for a prediction model in image classification tasks and allows output uncertainty for given samples. For test purposes, a CNN model will be used over the CIFAR-10 dataset to showcase uncertainty evaluation. We also show how to use uncertainty values to get data insights into a real-world task.

In the second part, we discuss how to get a model to know when prediction is uncertain. We built a selective classifier to increase the performance of the model by narrowing the confidence interval on the input data and used the aforementioned uncertainty estimations in the rejection classifier. To showcase classifier features, we made an experiment with a softmax-based uncertainty classifier (vanilla) and Dirichlet distribution based value. To measure the performance of the predictor, we took the Brain Tumor Classification (MRI)[6] dataset as an example. For the received predictors we measured coverage and selective risks. We have shown that one could get

significant accuracy gains by using selective models given accurate uncertainty measure.

Key words: *machine learning, deep learning, uncertainty estimation, selective predictor, image classification.*

Аннотація. *Зазвичай, техніки глибокого навчання на виході дають тільки прогнозоване значення, але не міру непевності в прогнозі. Існує безліч методів модифікувати модель так, що існуватиме можливість оцінити, наскільки моделі можна довіряти. Ці методи відрізняються вимогами до обчислювальних потужностей і точності з урахуванням заданих обмежень. На практиці, часто не існує можливості модифікувати модель або перетренувати з використанням технік оцінки невпевненості (моделі типу чорний ящик).*

У першій частині статті, ми визначимо і дослідимо толерантні перетворення як засіб додавання шуму у вхідні дані з метою виміряти непевність моделі. Ми побудуємо фреймворк, який буде визначати невпевненість для задач пов'язаних з класифікацією зображень. Для тесту, був узятий датасет CIFAR-10, щоб порахувати невпевність на зображеннях. Показано, як метрика невпевненості дозволяє отримати додаткову інформацію про набір даних на практиці.

Іноді важливо розуміти, коли модель не впевнена в своєму прогнозі. У другій частині статті, буде побудований вибірковий класифікатор з метою поліпшення точності моделі шляхом звуження довірчого інтервалу. Класифікатор буде отримувати на вхід метрику невпевненості в прогнозі. Щоб показати як працює класифікатор, був проведений експеримент з використанням softmax-значення невпевненості і невпевненістю на підставі розподілу Діріхле. Щоб оцінити якість предиктора, був використаний набір даних Brain Tumor Classification (MRI)[6]. Для отриманих предикторов виміряні покриття і вибірковий

ризик. Продемонстровано, що можна отримати значний приріст точності моделі, якщо використовувати хороші дані про невпевність.

Ключові слова: *машинне навчання, глибинне навчання, визначення непевності, вибірковий предиктор.*

Аннотация. *Обычно, техники глубокого обучения на выходе дают только предсказание, но не меру неуверенности в предсказании. Существует множество методов модифицировать модель так, что будет возможность оценить, насколько модели можно доверять. Эти методы отличаются требованиями к вычислительным мощностям и точности с учетом заданных ограничений. На практике, часто нет возможности модифицировать модель либо перетренировать с использованием техник оценки неуверенности (модели типа черный ящик).*

В первой части статьи, мы определим и исследуем толерантные преобразования как средство добавления шума во входные данные с целью измерить неуверенность модели. Мы построим фреймворк, который будет поверх предсказывающей модели определять неуверенность для задач связанных с классификацией изображений. Для теста, был взят датасет CIFAR-10, чтобы посчитать неуверенность на изображениях для примера. Будет показано, как метрика неуверенности позволяет получить дополнительную информацию о наборе данных на практике.

Иногда важно понимать, когда модель не уверена в своем предсказании. Во второй части статьи, будет построен выборочный классификатор с целью улучшения точности модели путем сужения доверительного интервала. Классификатор будет получать на вход метрику неуверенности в предсказании. Чтобы показать как работает классификатор, был проведен эксперимент с использованием softmax-значения неуверенности и неуверенностью на основании распределения Дирихле. Чтобы оценить качество предиктора, был использован набор данных Brain Tumor Classification (MRI). Для полученных предикторов

измерены покрытие и выборочные риски. Продемонстрировано, что можно получить значительный прирост точности модели, если использовать хорошие данные о неуверенности.

Ключевые слова: *машинное обучение, глубокое обучение, определение неуверенности, выборочный предиктор.*

Part I. Background. Uncertainty is the state of having limited knowledge where it is impossible to exactly describe the existing state. Understanding if a model is under-confident or falsely overconfident can help reason about the model and dataset. There are several types of uncertainty, but aleatoric and epistemic are most widely used. Aleatoric uncertainty is important in cases where parts of the observation space have higher noise levels than others. Concrete examples of the aleatoric uncertainty in stereo imagery are occlusions (parts of the scene a camera can't see), lack of visual features (i.e a blank wall), or overexposed (underexposed) areas. Epistemic uncertainty measures the influence of a lack of training data over model false predictions. A possible way to observe the epistemic uncertainty in action is to train one model on part of a dataset and to train a second model on the entire. The model trained on part of the dataset will have higher average epistemic uncertainty. It is important because it identifies situations the model was never trained to understand (lack of training data). It is also helpful in dataset exploration, e.g. it shows whether a model is using primary base parameters instead of unwanted secondary features of the dataset.

From the implementation perspective, it matters if the model is open for modification. Generally speaking, there are three cases of model modification availability: black-box, grey-box and white-box. The black-box stands for a closed model, where there is no access to the internal module structure. Grey-box is a case when the internal structure of the model is accessible, while parameters are not. White-box case describes a model fully available for

modification. The uncertainty estimation of the black-box model is particularly challenging but appears to be a more universal solution.

Problem statement. We will be looking to solve the aleatoric uncertainty estimation problem for the black-box model. We have model F(x) that is trained to produce class affiliation prediction. We know the shape and the type of the input x, the classes that model supposed to infer, and meaning of the model output y. Since the model is black-box, there is no possibility to either read or write model structure and weights. We have to propose a framework that could be built on top of the model to estimate the uncertainty of the model for a given input.

Related work. A neural network is usually composed of a large number of parameters and activation functions, which makes the posterior distribution of a network prediction hard to interact with. To approximate the posterior, existing methods deploy different techniques, mainly based on Bayesian inference and Monte-Carlo sampling[5]. There are other options besides essentials, including modifications of the algorithms.

The suggested approach. To derive the softmax distribution we will have to perform multiple inferences of the same sample with different noise factors (according to the subject area). To force the data uncertainty, we will use tolerable perturbations. Tolerable perturbation is a method of causing small changes in the model input, that exploits model dependence on the input transformations. Let T denote a transformation. Then T'denotes an inverse transformation and $T \cdot T'$cancels the transformation. If input x is an $H * W$matrix, the variance ϵt estimated through transformation is also an $H * W$ matrix.

We will be using rotate, mirror, white noise, and translate perturbations with random parameters. If we train a model using a data augmentation with a similar idea, we might expect a reduction in variance when applying

perturbations to test samples. The understanding of tolerable and intolerable perturbation might be inferred based on the knowledge of the subject area.

The first primary aim would be to implement a few algorithms and research its behavior given certain synthetic cases. We will try a few datasets within our research but begin with cifar10[1]. For the experiment purposes, the whole airplane class images were cut off the testing set. We expect to find that the undertrained class should stand out from the others in terms of uncertainty measure. Ideally, the measurements should provide formal criteria for the class that lacks data samples. After the initial train with 40 epochs, the model scored 73% of accuracy, given it did not encounter any images of airplane class.

To derive uncertainty estimation, the model should be tuned to provide softmax distribution rather than softmax output. As we are focused on the black-box estimation, we should get the distribution first. Considering there is no access to the model structure, we have to get the different outputs for the single sample. To achieve that, tolerable perturbations were used[2] (fig 1).

Fig. 1. Perturbation example. Left to right: original image, skew, mirror, white noise, rotate, translate

Results interpretation. Expectedly, the model produces different softmax outputs for each perturbation of the same sample, which forms certain distribution of softmax. Given the distribution, several scales of uncertainty might be calculated.

$$u_i = \sum p_i * log_{10}(p_i)$$

The formula gives a per-class entropy for a given sample. This data gives insights into the model performance. The following observations were received. For each sample, we perform 50 tolerable perturbations and calculate entropy with the formula given above.

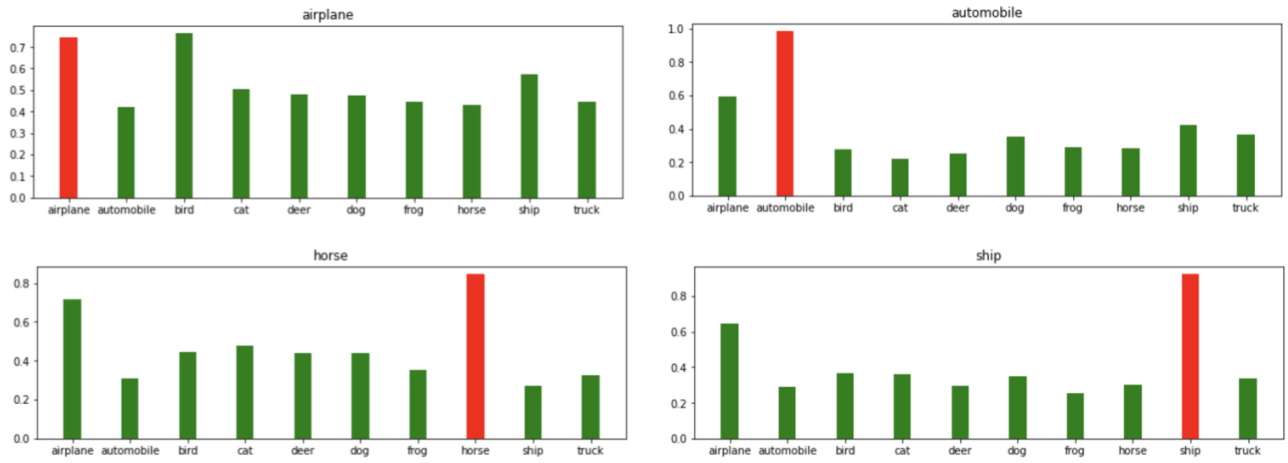

Fig. 2. Softmax distributions per samples of certain classes (red bars)

The chart (fig. 2) is built the following way:

1. 500 samples randomly picked from the test set
2. Per-class entropy is calculated for each sample
3. The results are gathered to the data frame
4. Uncertainties grouped by class and averaged

Intuitively, such distributions suggest that the higher entropy within a class in relation to the other classes, the more model ready to correctly predict that class. For all classes except airplane, we observe that target class entropy is much higher than the entropy of other classes within the sample. The other explanation is that the model distributes the noise among the invalid classes thus ending up with a correct prediction. Also, the general entropy for non-airplane outputs is higher for airplane samples. The airplane variance is most often the second high after the primary class. The model was unable to infer the unseen class feature set, therefore its softmax output reacts heavily to any data change. This fact is deeply researched in this publication [3]. Next, we produced the same calculations using variance uncertainty calculation instead of entropy.

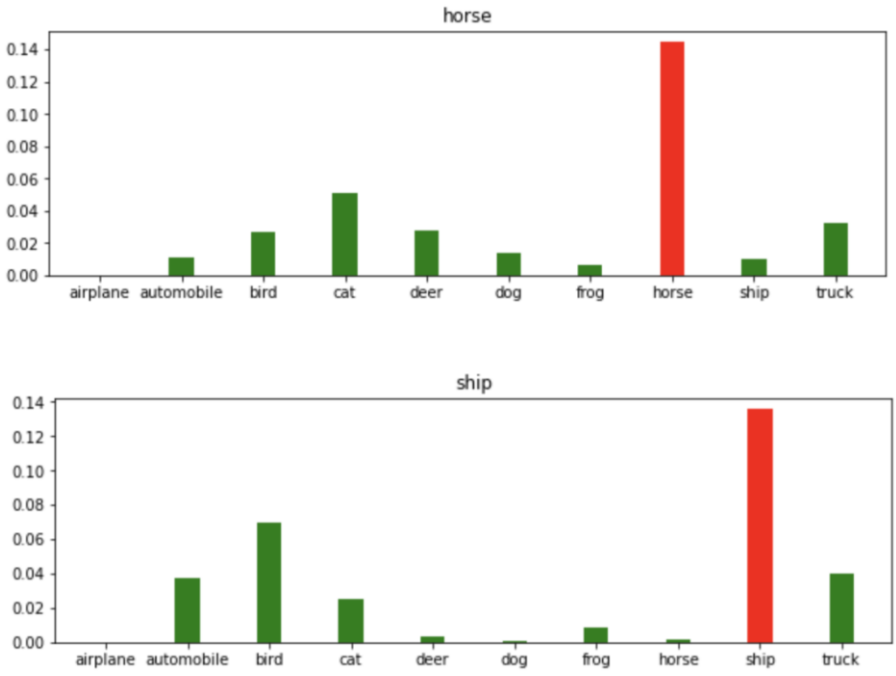

Fig. 3. Per-class variance for certain classes (red bars)

The per-class variance provides a clearer view of the class confidence by softmax output change (fig. 3). The small changes are evaluated closer to zero, which makes a class more outstanding among the others.

Next, we'll calculate a single-value constrained uncertainty for a sample. To achieve that, the softmax should be modelled as a certain distribution. Among the distributions, the Dirichlet distribution was selected because it has a straightforward formula to derive a theoretical uncertainty measurement that falls between 0 and 1. The uncertainty is calculated by the following formula:

$$U(X) = \frac{\sum 1}{\sum 1 + \alpha}$$

Where U is an uncertainty measure, $\sum 1 = N$, N is a number of classes, α is a Dirichlet parameter. To get α, we have to derive the distribution from samples. The Maximum Likelihood Estimation approach was chosen to get the α parameter. After we have per-sample uncertainty calculated with the formula above, we could try to get data insight.

Here are some observations based on the estimator output:

1. The uncertainty varies from 19 to 85 percent.
2. The higher uncertainty usually occurs where the image is dark or a deer looks like a cat or dog.
3. Most of the time, the antlers make the model confident in its predictions.
4. Model considers surroundings green grass and a silhouette is confidently predicted as deer.

Intermediate conclusion. We got 3 values for uncertainty - entropy, variance and single-value uncertainty. For the calculated types of uncertainties, we received meaningful results and the results are pretty decent in comparison to the ones produced by existing approaches. We are able to get some intuition on what those values show and how to use them to analyze flaws in training data / training procedure.

Part 2. Uncertainty application for DL model boosting. It is worth mentioning that there are lots of publications on the uncertainty estimation itself, but there are no user-friendly frameworks, allowing to apply such kinds of instruments to the established ML system. Besides, the complete formal solution to this problem is also missing. Based on the materials found, we will evaluate its further development and usage in the existing models. While working on the materials for the paper, we have encountered multiple ways to make use of uncertainty. However, we will be mostly focused on the Selective Predictor.

Selective predictor. The selective predictor [4] is used when it is critical to work with the prediction that model is very confident in. It is crucial that the model would not give the wrong answer. The essential idea of the approach is that the system should choose if its prediction should be used based on the certain parameters or otherwise returned "I don't know" answer for this input. In our case, such an answer is based on the uncertainty parameter.

Selective predictor background. A selective predictor is a pair (f, g), where f is a predictor, and $g: X \rightarrow \{0,1\}$ is a selection function, which serves as a binary qualifier for f as follows,

$$(f, g)(x) = \begin{cases} f(x), & \text{if } g(x) = 1 \\ don't\ know, & \text{if } g(x) = 0 \end{cases}$$

Thus, the selective predictor rejects from prediction at a point x if $g(x) = 0$. The concepts of coverage and risk are introduced for its evaluation.

Coverage and selective risk. The coverage of a selective predictor (f, g) is the mean value of the selection function $g(x)$ taken over the underlying distribution P, $\varphi(f, g) = EP\ [g(x)]$. where EP is expected value for initial dataset distribution P. We calculate the risk of a selective classifier (f, g) as the average loss on the accepted samples, $R(f, g) = EP[\ l(f(x), y)g(x)\ ]\varphi(f, g)$ where $l(f(x), y))$ is a loss function for the selected task. Basically, the risk of a selective classifier can be traded-off for coverage. The entire performance profile of such a classifier can be specified by its risk-coverage(RC) curve, defined to be risk as a function of coverage.

Selective predictor implementation. Since we are working with the image classification task, it is obvious that we have chosen a simple Convolutional Neural Network (CNN) to build a predictor over it. It should be noted that we didn't take into account complex models as the main purpose of the work is to obtain improvements for simple, but promising models based on neural networks.

We have chosen a medical-type dataset of brain tumor classification task. So, our task is to predict the category (type) of brain cancer on the basis of MRI images or choose a category that corresponds to its absence. The dataset is quite small: ~3000 training images and about ~400 test images. Image resolution is 512x512. 4 categories total. After 12 epochs of training, CNN model evaluated to 75% accuracy on the validation dataset. The model was frozen and later used

for tests. Now we can apply the uncertainty estimation techniques over the trained model. We took two uncertainty measures to work with - softmax (vanilla) and dirichlet single-value uncertainty. It is worth mentioning that softmax usage as uncertainty value is technically incorrect. Note, however, that we are not concerned with the standard probabilistic interpretation (which needs to be calibrated to quantify probabilities). The softmax values research has shown that model is probably overfitted. Softmax vectors tend to have values close to 1 in the predicted class position. The softmax outputs suggest that there is a problem with the model.

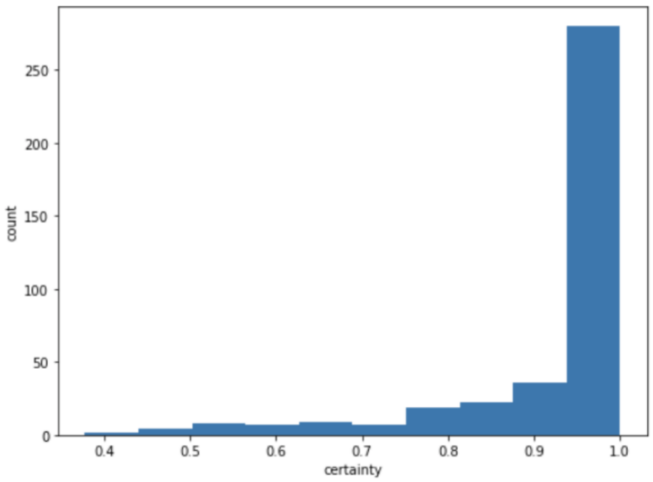

Fig. 4. A certainty distribution for a subset of data

For the single-value uncertainty, the distribution of the uncertainty (fig. 4) suggests that the classifier works well. We can see that it varies from approximately 15 to 90 percent. Estimator does not stick to the same uncertainty level. Overall, it is noticeable that model is rather not confident (depends on the threshold of uncertainty).

Expectedly, after selective classifier implementation, we should have get the increased accuracy over relatively wide coverage. As mentioned earlier, the main characteristics of the classifier are coverage and risk. We have measured them and also implemented functions to compute accuracy considering coverage for the given dataset. We received the following results (fig. 5, 6).

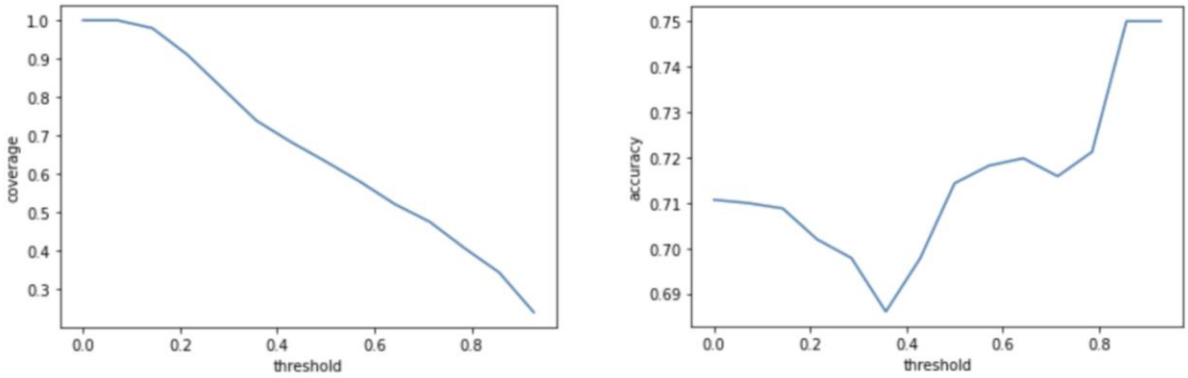

Fig. 5. Coverage and accuracy charts for Dirichlet-based uncertainty

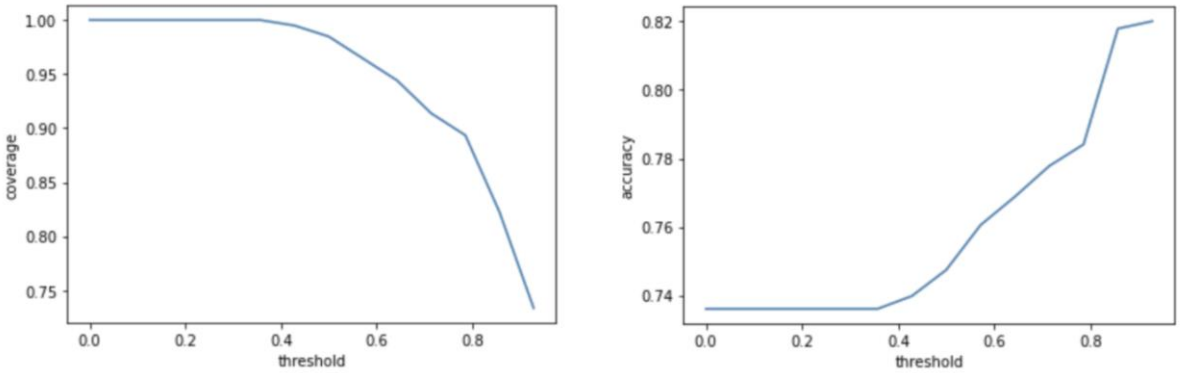

Fig. 6. Coverage and accuracy charts for softmax-based uncertainty

Conclusion. Within this work we have applied the uncertainty estimation theory and implemented uncertainty estimator for the backbox DL model for image classification task. We have analyzed the output of different uncertainty estimators and given some intuition on how to interpret those results considering task and dataset.

In the second part of the work, we have researched ways to apply uncertainty estimation to improve model accuracy and to give model a possibility to be used in risk-sensitive areas.

The experiment results have clearly shown that uncertainty estimator selection is an important subtask for classifier construction. Intuitively, it seems that the following points should be considered:

1. Model architecture

2. Dataset content and labelling quality
3. Subject area.

We have confirmed that vanilla uncertainty estimations (raw softmax) proven to be a good option for certain tasks, despite that practically softmax is not an uncertainty measure. The monotonic increasing function of accuracy of threshold shows how solid the result selective predictor would be. The results suggest that selection of the uncertainty estimation is important. The selected method is shown rather model and dataset imperfection but did not provide for selective classifier construction task.

References

1. CIFAR-10 and CIFAR-100 datasets. URL: https://www.cs.toronto.edu/~kriz/cifar.html11.
2. Mi L., Wang H., Tian Y., and Shavit N. Training-Free Uncertainty Estimation for Neural Networks. arXiv preprint arXiv:1910.04858, 2019.
3. Ovadia Y. Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift", arXiv preprint arXiv:1910.04858, 2019.
4. Geifman Y. and El-Yaniv R. Selective Classification for Deep Neural Networks. arXiv preprint arXiv:1705.08500, 2017.
5. Ian O. "Risk versus uncertainty in deep learning: Bayes, bootstrap and the dangers of dropout", in Advances in Neural Information Processing Systems Workshops, 2016.
6. Brain Tumor Classification (MRI) dataset. URL: https://www.kaggle.com/sartajbhuvaji/brain-tumor-classification-mri