Risk-Controlled Selective Prediction for Regression Deep Neural Network Models

Wenming Jiang
Tsinghua University
Beijing, China
jwm17@mails.tsinghua.edu.cn

Ying Zhao*
Tsinghua University
Beijing, China
yingz@tsinghua.edu.cn

Zehan Wang
Tsinghua University
Beijing, China
zw2457@columbia.edu

***Abstract*—Regression deep neural network (DNN) models have been successfully utilized in numerous fields. In real-world applications, large regression errors on individual samples may result in severe consequences. Selective techniques, also known as reject options, have been used to reject predictions with high uncertainty. However, they have yet been mainly considered in classification neural networks (NNs), in comparison to the limited work in regression NNs. In this paper, we considered the selective regression problem from a risk-coverage point of view, and proposed a method to construct a selective regression model given a trained regression DNN model and a desired regression error risk. Then, we proposed to utilize blending variance to quantify uncertainty in regression NNs. We evaluated both the proposed uncertainty function and selective regression models for two real-world applications, the tropical cyclone (TC) intensity estimation problem and the apparent age estimation problem. Our proposed methods achieved promising results. For example, for the TC intensity estimation problem, our selective regression model guaranteed a risk bound (in terms of the root mean squared error (RMSE)) of 9.5 knots for 75% test coverage with a guided confidence level of 0.05, whereas the RMSE value achieved by the state-of-the-art model without selection was 10.5 knots.**

***Index Terms*—selective prediction, regression, deep neural networks, blending variance**

I. Introduction

Deep neural networks (DNNs) have been widely used for various regression problems, such as tropical cyclone (TC) intensity estimation [1]–[3], age estimation [4], [5], wind power prediction [6], pain intensity estimation [7], and so on. Applying such models to real-world applications often requires a control on regression errors on individual samples. For example, TC intensity estimation uses satellite images of TCs to estimate their intensities. When using regression models for weather forecasting, a large regression error on a single satellite image may result in underestimating the rank of a TC and causing significant casualties and economic losses. Hence, measuring model uncertainty is important and we do not want to accept all prediction results without taking the uncertainty of each prediction into account. Instead, we would like to reject the predictions with high uncertainty (and consult with domain experts). This method is called predicting with a reject option or selective prediction [8].

Supported by the National Key Research and Development Program of China (2017YFA0604500) and Tsinghua University Initiative Scientific Research Program.

* Ying Zhao is the corresponding author.

Most of the existing works on selective prediction focused on selective classification problems and various uncertainty functions for classifiers to construct selective models [9]–[12]. Given a classification model and a function measuring model uncertainty, selective classification models trade-off between misclassification and rejection rates to achieve higher classification accuracy on as many input samples as possible. In particular, references [11] and [12] put forward a risk-coverage framework, under which selective classification models were constructed to maximize the selective coverage with a guaranteed risk bound.

However, the existing selective prediction methods for classification problems cannot be used directly to solve regression problems. We should first acknowledge that a regression problem can be transferred to a classification problem and solved by DNN models. For example, in the TC intensity estimation problem, we can divide TC intensity into a number of categories (*e.g.*, 18 TC ranks) and use classification DNN models to solve it [13]. However, the evaluation of classification performance is different from that of regression performance. For classification problems, we commonly use classification accuracy indicators, such as the misclassification rate, to evaluate classification results, whereas for regression tasks, we usually employ regression risk functions to measure the magnitude of errors, such as the mean squared error (MSE) or the mean absolute error (MAE). Since the existing selective classification methods and theoretical bounds were derived for optimizing classification accuracy, they could not be used directly for optimizing regression risk functions.

In this paper, we focused on the selective prediction problem for regression DNNs, and made the following contributions:

- For a given regression model f, a confidence level δ, and a desired regression risk target r^*, we proposed a method to construct a selective function g, such that the selective regression model (f, g) can achieve maximum coverage while keeping expected regression risk no larger than r^* with probability $1 - \delta$.
- We proposed a new uncertainty function for rejection, *Blend-Var*, which measures the variance of multiple predictions of a single input sample (such as an image) when blending with rotation, reflection, shifting and so on.
- We evaluated our selective regression models with the proposed uncertainty function on two real-world applica-

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

tions and achieved promising results. For example, for the TC intensity estimation problem, our selective regression model guaranteed a risk bound (in terms of the root mean square error (RMSE)) of 9.5 knots (1 knots $\approx$ 0.514 meter per second) for 75% test coverage with a guided confidence level of 0.05, whereas the RMSE value achieved by the state-of-the-art model without selection was 10.5 knots.

The rest of the paper is organized as follows. Section 2 reviews related works. Section 3 introduces the background of the general selective model and Hoeffding's Inequality. Section 4 proposes the risk-controlled selective regression model and the novel uncertainty function *Blend-Var*. Section 5 presents the experimental results on two real-world applications. Section 6 concludes the paper.

II. Related Work

Selective prediction, or prediction with a reject option, has been studied for more than a half century. Early works tackled the problem based on statistical decision theory to trade-off between error and rejection rates in the recognition problem [8], [14]. Later on, selective prediction models were proposed for various hypothesis classes and learning algorithms, among which selective models for neural networks (NNs) [9], [10] and DNNs [11], [12] drew people's attention lately. Most of the existing works focused on selective classification problems and various uncertainty functions for classifiers to construct selective models. Recently, a DNN architecture called SelectiveNet [15] with an integrated reject option was put forward, which is trained to optimize both classification (or regression) performance and rejection rate simultaneously.

There are basically two types of selection classification models: from a cost-based point of view and from a risk-coverage point of view. References [9] and [10] tackled the problem from a cost-based point of view, which defines a selection cost function including both misclassification and rejection rates and searches for the selective classification model that optimizes the cost function. The risk-coverage point of view, applied in [12] and [11], also aims to trade-off between the selective risk and coverage. However, in this framework, selective classification models are constructed (usually through constructing selection functions of classifiers) to maximize the selective coverage, under the control of a selective risk target.

The uncertainty functions of classifiers are used to reject samples for which the risk of inaccurate prediction is judged too high. In [9] and [10], a reject threshold was set on the maximal neuronal response in the softmax layer. This mechanism is known as *softmax response* (SR). In [12] and [16], *Monte-Carlo dropout* (MC-dropout) was used to estimate the predictive uncertainty in neural networks with dropout layers [17]. Dropout could be interpreted as an ensemble technique, approximately combining different networks with shared weights. Reference [16] showed that a neural network with dropout applied before every weight layer is approximately equivalent to the probabilistic deep Gaussian process. The predictive uncertainty can then be seen as the variance of sample predictions of multiple stochastic forward passes through the network.

In recent years, DNNs have been widely used for regression problems as well, such as TC intensity estimation [1], age estimation [4], wind power prediction [6], remaining lifetime estimation [18], and pain intensity estimation [7]. However, the selective prediction problem for regression neural networks was just discussed by [15] lately. Although SelectiveNet [15] is a DNN architecture with an integrated reject option, it is challenging to apply such architecture to real-world applications. Instead, we tackle this problem from a risk-coverage point of view using the given regression DNNs and some uncertainty functions associated with them to achieve the desired selective risk and coverage.

III. Background

Let $\mathcal{X}$ be some feature space (e.g., raw image data or d-dimensional vectors in $\mathbb{R}^d$) and $\mathcal{Y}$, the output space. We have a prediction function f, $f : \mathcal{X} \to \mathcal{Y}$. Although in the literature, uncertainty driven selective models were mainly studied and derived for classification problems, here we introduce the selective model from a more general point of view, *i.e.*, letting f be either a classification function or a regression function, and $\mathcal{Y}$ be either a set of categorical labels or a real-valued set.

A. General Selective Model

A selective model is a pair (f, g) [11], where f is a prediction function, and $g : \mathcal{X} \to \{0, 1\}$ is a *selection function*, which serves as a binary qualifier as follows. For a given sample $x \in \mathcal{X}$, its output is:

$$(f, g)(x) \triangleq \begin{cases} f(x) & if\ g(x) = 1 \\ reject & if\ g(x) = 0 \end{cases} \tag{1}$$

Note that $(f, g)(x) \triangleq f(x)$, if $\forall x, g(x) = 1$, *i.e.*, no sample is rejected and the selective model is the function f itself. We usually utilize an uncertainty function $\kappa_f : \mathcal{X} \to \mathbb{R}$ for f, to measure how well a prediction fits the corresponding ground truth [19]. Using an uncertainty function κ_f and a threshold θ, we can form a selection function $g_\theta(x)$ as follow,

$$g_\theta(x) = g_\theta(x|\kappa_f) \triangleq \begin{cases} 1 & if\ \kappa_f(x) \le \theta \\ 0 & if\ \kappa_f(x) > \theta \end{cases} \tag{2}$$

The idea of a selective model is to reject some badly predicted samples so that f can achieve better performance on the remaining ones. The performance of a selective model can be considered from a *risk-coverage* point of view [11]. Formally, let $P(X, Y)$ be a distribution over $\mathcal{X} \times \mathcal{Y}$, also shorted as P. The *selective coverage* of (f, g), defined to be $\Phi(f, g) \triangleq E_P[g(x)]$, is the expectation of $g(x)$ or the no-reject-region-rate in $\mathcal{X}$. The *selective risk* of (f, g) is defined as

$$R(f, g) \triangleq \frac{E_P[\ell(f(x), y)g(x)]}{\Phi(f, g)}, \tag{3}$$

where ℓ is a loss function measuring the loss between $f(x)$ and the ground truth output y of x, $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$.

It is hard to obtain $\Phi(f,g)$ and $R(f,g)$ from an unknown distribution. Instead, we can construct a validation set consisting of m labeled samples $S_m = \{(x_i, y_i)\}_{i=1}^m$, assumed to be sampled i.i.d. from $P(X,Y)$, and estimate $\Phi(f,g)$ and $R(f,g)$ on S_m.

Now, given a confidence parameter $\delta > 0$ and a desired risk target $r^* > 0$, the goal is to find a selection function g that maximizes $\Phi(f,g)$ while its selective risk satisfies

$$Pr\{R(f,g) > r^*\} < \delta. \tag{4}$$

B. Measuring Model Risk with Hoeffding Bounds

Hoeffding's inequality [20], Chernoff bound [21], and Azuma's inequality [22] are the main analytic tools to bound the probability of a large discrepancy between sample and population means. In machine learning, Hoeffding's inequality is often used to ensure the generalization of a prediction function f by bounding the probability of the gap between the expected risk over the distribution $P(X,Y)$ and the empirical risk on a validation set of f.

For each data point (x,y) sampled i.i.d. from $P(X,Y)$, we consider $\ell(f(x),y)$ as a random variable, and use Hoeffding's inequality [20] directly as stated in the following lemma.

Lemma 1: Given a prediction function f, a distribution $P(X,Y)$, and a loss function ℓ, assume that $b = max_{P(X,Y)}(\ell(f(x),y))$ and $a = min_{P(X,Y)}(\ell(f(x),y))$ are finite and real-valued. If we sample n data points i.i.d. from $P(X,Y)$ (*i.e.,* $(x_i,y_i) \sim P(X,Y)$, for each $1 \le i \le n$), then for $t \ge 0$,

$$Pr\{R(f) - R_n(f) \ge t\} \le e^{\frac{-2nt^2}{(b-a)^2}}, \tag{5}$$

where $R(f) = E_P(\ell(f(x),y))$ is the expected model risk and $R_n(f) = \frac{1}{n}\sum_{i=1}^n \ell(f(x_i),y_i)$ is the empirical risk over n samples.

Hoeffding's inequality provides loose bounds for estimating model risks. In practice, $b-a$ are often approximated by using the maximum and minimum error observed on the validation set [23], or by adding a few standard deviations to the average error of f on the validation set to avoid the affection of prediction outliers of f [24].

IV. Method

We formulate the selective regression problem following the general selective model framework and propose a learning algorithm to obtain selective regression models that are likely to produce better regression performances for a large portion of samples from the feature space $\mathcal{X}$.

A. Problem Setting

For a regression model f, the output space $\mathcal{Y}$ is assumed to be real-valued, $\mathbb{R}$. The expected risk of f w.r.t. the distribution over $\mathcal{X} \times \mathcal{Y}$ (*i.e.,* $P(X,Y)$) is $E_P(\ell(f(x),y))$, where the loss function $\ell(f(x),y)$ is usually the squared error loss $\ell(f(x),y) = (f(x)-y)^2$ or the absolute error loss $\ell(f(x),y) = |f(x)-y|$. Thus, $E_P(\ell(f(x),y))$ measures MSE or MAE of the regression model f, assumed to be real-valued and finite.

A *selective regression model* is a pair (f,g), where f is a regression model and g is a selection function as defined in (1). Similarly, we form a validation set consisting m labeled samples $S_m = \{(x_i,y_i)\}_{i=1}^m$, assumed to be sampled i.i.d. from $P(X,Y)$. We formulate the selective regression problem as follows.

Definition 1: **Selective Regression Problem**. Given a feature space $\mathcal{X}$, a real-valued output space $\mathcal{Y}$, a distribution over $\mathcal{X} \times \mathcal{Y}$ $P(X,Y)$, a regression model $f : \mathcal{X} \to \mathcal{Y}$, a validation dataset S_m, a confidence parameter $\delta > 0$, and a desired risk target $r^* > 0$, the **selective regression problem** is to find a selective regression model (f,g) that maximizes $\Phi(f,g)$ while its selective risk satisfies

$$Pr\{R(f,g) > r^*\} < \delta, \tag{6}$$

where $\Phi(f,g) = E_P[g(x)]$ is the coverage of (f,g), $R(f,g) = \frac{E_P[\ell(f(x),y)g(x)]}{\Phi(f,g)}$ is the selective risk of (f,g) and evaluated using regression risk functions, such as MSE or MAE.

Note that the nature of the MSE/MAE-based selective risk reflects the essential difference between regression and classification problems, which also makes the method proposed in [12] unsuitable for solving this problem.

Although f can be any kind of regression models, in this paper we focus on DNNs, where existing techniques (such as softmax response [9], dropout layers [17], and ensemble methods [1]) provide promising ways of measuring uncertainty of f.

B. Selective Regression with Controlled Risk

Given a selective regression model (f,g), let $P_g(X,Y)$ be the projection of $P(X,Y)$ over g, *i.e.,* $P_g(X,Y) \triangleq P(X,Y|g(X)=1)$. The selective risk of (f,g) can be written as

$$R(f,g) = \frac{E_P[\ell(f(x),y)g(x)]}{E_P[g(x)]} = E_{P_g}[\ell(f(x),y)].$$

We can use Hoeffding's inequality [20] to establish the risk bound of a selective regression model (f,g) using a validation set S_m sampled i.i.d. from $P(X,Y)$.

Lemma 2: Given a selective regression model (f,g), a projection distribution $P_g(X,Y)$, and a loss function ℓ, assume that $b = max_{P_g(X,Y)}(\ell(f(x),y))$ and $a = min_{P_g(X,Y)}(\ell(f(x),y))$ are finite and real-valued. If we sample n data points i.i.d. from $P_g(X,Y)$ (*i.e.,* $(x_i,y_i) \sim P_g(X,Y)$, for each $1 \le i \le n$), then for $t \ge 0$,

$$Pr\{E_{P_g}(\ell(f(x),y)) \ge \frac{1}{n}\sum_{i=1}^n \ell(f(x_i),y_i) + t\} \le e^{\frac{-2nt^2}{(b-a)^2}}. \tag{7}$$

The assumption that $b = max_{P_g(X,Y)}(\ell(f(x),y))$ and $a = min_{P_g(X,Y)}(\ell(f(x),y))$ are finite is true for many real-world applications, such as tropical cyclone intensity estimation and human age estimation, where the output space $\mathcal{Y}$ has natural bounds and $f(x)$ can follow the same bounds.

Suppose an uncertainty function κ_f is available for constructing selective regression models using (2). For a certain

selection function g_θ and a validation set S_m, we can filter S_m using g_θ, *i.e.*, keep samples with $\kappa_f(x) \leq \theta$. Note that sampling from $P_{g_\theta}(X,Y)$ is equivalent to filtering S_m using g_θ, since S_m was drawn i.i.d. from $P(X,Y)$. Hence, we can estimate the selective risk of (f, g_θ), $E_{P_{g_\theta}}[\ell(f(x), y)]$. The following theorem ensures this estimation with a guaranteed bound.

Theorem 1: Given a selective regression model (f, g_θ), a projection distribution $P_{g_\theta}(X,Y)$, a loss function ℓ, a validation set S_m, assume that $b = max_{P_{g_\theta}(X,Y)}(\ell(f(x), y))$ and $a = min_{P_{g_\theta}(X,Y)}(\ell(f(x), y))$ are finite and real-valued. Let S_θ be the filtered sample set of S_m using g_θ and z is the size of S_θ. Then, for $t \geq 0$,

$$Pr\{E_{P_{g_\theta}}(\ell(f(x), y)) \geq \frac{1}{z} \sum_{(x,y)\in S_\theta} \ell(f(x), y) + t\} \leq e^{\frac{-2zt^2}{(b-a)^2}}. \tag{8}$$

Now, given a regression risk target r^*, we need to search for a uncertainty threshold θ for $\kappa_f(x)$, such that the selection function g_θ maximizes the coverage while its selective risk satisfies $Pr(R(f, g_\theta) > r^*) < \delta$. Theorem 1 suggests that the selective risk $E_{P_{g_\theta}}(\ell(f(x), y))$ can be estimated by the empirical risk $\frac{1}{z}\sum_{(x,y)\in S_\theta} \ell(f(x), y)$ plus a gap t. Given a confidence level δ, we can obtain an analytic solution of t from the right-hand side of (8) by setting $e^{\frac{-2zt^2}{(b-a)^2}} = \delta$, which gives us $t = \sqrt{-\frac{(b-a)^2 \ln \delta}{2z}}$.

If we have an ideal uncertainty function κ_f, *i.e.*, for $(x_1, y_1) \sim P(X,Y)$ and $(x_2, y_2) \sim P(X,Y)$, $\kappa_f(x_1) \leq \kappa_f(x_2)$ if and only if $\ell(f(x_1), y_1) \leq \ell(f(x_2), y_2)$, sorting all samples in S_m w.r.t. κ_f also results in a monotonically increasing sequence of $\ell(f(x), y)$. Hence, by searching along this sequence we can find the maximum number of samples (thus the corresponding θ to make the split) whose empirical risk is less than the desired risk target.

Noticing that $b - a$ is bounded by the difference between the maximum regression error and minimum regression error of f over the data distribution, so t decreases rapidly as z increases and becomes stable after z reaches a number (we call it m_0, the *minimum number of samples needed*), such that (a) the change in z above this threshold would not bring any significant change in t and (b) t would be small enough compared with both the expected risk and empirical risk.

When $z \geq m_0$, the sequence sorted by κ_f is also a monotonically increasing sequence of $\ell(f(x), y)$ plus t, and we can find the maximum θ whose empirical risk is less than the desired risk target. We speed up this process by a binary search strategy shown in Algorithm 1. In line 2, z_{min} is the starting index to begin the search with. We set z_{min} to be m_0. Lines 4 and 5 define a selection function g_θ, and the first z samples of S_m form the set S_θ for g_θ. With $t = \sqrt{-\frac{(b-a)^2 \ln \delta}{2z}}$ and $\hat{r}_z = \frac{1}{z}\sum_{i=1}^{z} \ell(f(x_i), y_i)$, by Theorem 1 we have $Pr\{R(f, g_\theta) > (\hat{r}_z + t)\} \leq \delta$. Furthermore, we require $\hat{r}_z + t \leq r^*$ to lead the binary search shown in Lines 9 to 12. When the search terminates and a solution indeed exists, the algorithm finds the maximum θ such that $\hat{r}_z + t \leq r^*$, which also guarantees $Pr\{R(f, g_\theta) > r^*\} \leq \delta$.

Algorithm 1 Selection with Guided Regression Risk (SGRR)

Require: $f, \kappa_f, \delta, r^*, S_m$
Ensure: g_θ, $\hat{r}_z + t$

1: Sort S_m according to $\kappa_f(x), x \in S_m$;
2: $z_{min} = m_0$; $z_{max} = m$;
3: **while** $z_{min} \leq z_{max}$ **do**
4: $\quad z = \lfloor (z_{max} + z_{min})/2 \rfloor$;
5: $\quad \theta = \kappa_f(x_z)$; $S_\theta = \{(x_i, y_i)\}_{i=1}^{z}$;
6: $\quad b - a = Approx(S_\theta, f)$;
7: $\quad t = \sqrt{\frac{-ln\delta(b-a)^2}{2z}}$;
8: $\quad \hat{r}_z = \frac{1}{z}\sum_{i=1}^{z} \ell(f(x_i), y_i)$;
9: $\quad$ **if** $\hat{r}_z + t \leq r^*$ **then**
10: $\quad\quad z_{min} = z + 1$;
11: $\quad$ **else**
12: $\quad\quad z_{max} = z - 1$;
13: $\quad$ **end if**
14: **end while**
15: **if** $z_{max} \geq m_0$ **then**
16: $\quad z = z_{max}$;
17: $\quad$ Update g_θ and $\hat{r}_z + t$ and Output;
18: **end if**
19: Return;

Note that not all input values of δ and r^* lead to a feasible solution. However, when such feasible solution does exist, our SGRR algorithm guarantees to find the selection function g_θ that maximizes the coverage and satisfies the selective risk requirement. In practice, an ideal uncertainty function κ_f may not exist. However, with a proper choice of uncertainty functions, SGRR still can satisfy desirable risk bounds well as shown in our experiments. We also approximated $b - a$ by adding two standard deviations to the average error $(f(x) - y)$ of f on S_θ (as shown in Line 6 of Algorithm 1) as suggested in [24] to avoid the affection of prediction outliers of f. Although in this case, we cannot use Theorem 1 to guarantee the risk bound strictly, it still can be used as a guideline for risk control purposes. We call δ a *guided* confidence level, which was also verified in our experiments.

C. Uncertainty Functions

In this section we discuss two uncertainty functions, *Blend-Var* and *MC-dropout* for measuring the predictive uncertainty of a regression function f.

1) Blend-Var: Blending is a wildly used technique in data augmentation and ensemble models [1]. Unlike MC-dropout, which changes a DNN model slightly and gets different prediction results each time, blending transforms input x_0 by rotation, reflection, shifting and so on, and makes multiple predictions using these transformations. Fig. 1 demonstrates the regression prediction through blending for TC intensity estimation.

Let x_0^i denote the ith transformation of x_0. For T transformations, we can use f to calculate a prediction array

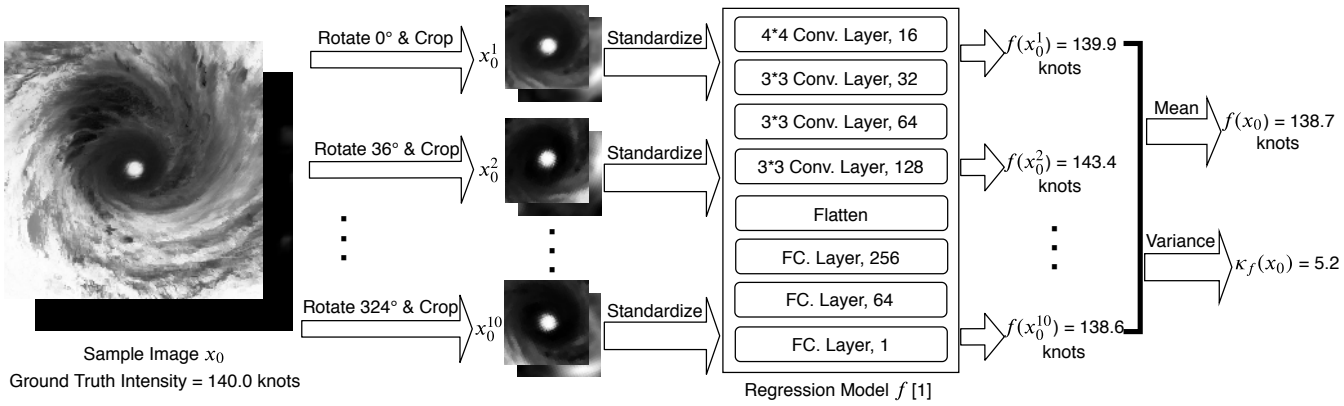


Fig. 1. Regression prediction and Blend-Var calculation through blending for TC intensity estimation.

$X_0 = [f(x_0^1), f(x_0^2), ..., f(x_0^T)]$. We propose to use the variance of X_0 (named as *Blend-Var*) to measure the predictive uncertainty of x_0, *i.e.*, $\kappa_f(x_0) = var(X_0)$. The mean of T predictions is the final prediction, *i.e.*, $f(x_0) = mean(X_0)$. For $T = 10$ as shown in Fig. 1, ten test samples were generated through rotation and cropping from the original input. The final averaged predicted intensity is 138.7 knots, which is very close to the ground truth of 140.0 knots. The Blend-Var uncertainty of x_0, $\kappa_f(x_0)$, is 5.2, which suggests that this prediction is reliable. Blend-Var can be used in NNs with dropout layers or not. To the best of our knowledge, we are the first one to put forward this uncertainty function.

2) MC-dropout: The predictive uncertainty of a prediction by a neural network regression function f with dropout layers can also be measured by the prediction variance of T stochastic forward passes through the network [16]. For a given instance x_0, we make T predictions with the same dropout rate as used in the training step, denoted as $f_1(x_0), f_2(x_0), ..., f_T(x_0)$. The variance of these predictions is used as the uncertainty function, *i.e.*, $\kappa_f(x_0) = var(X_0)$, while $X_0 = [f_1(x_0), f_2(x_0), ..., f_T(x_0)]$. MC-dropout technique could be used in both classification and regression neural networks. It does not work for those neural networks without dropout layers.

V. Experiments

We evaluate the proposed selective regression method, SGRR algorithm, and uncertainty functions on two regression tasks, TC intensity estimation from satellite remote sensing images and apparent age estimation from facial images. We first introduce the network architecture, datasets, and other experimental settings for each regression task, and then present the evaluation results. All the code and datasets used in the evaluation are available at https://github.com/Wenming-Jiang/Selective-regression-model.

A. Two Case Studies

1) Tropical Cyclone Intensity Estimation: TC intensity estimation from satellite imagery is a typical regression problem. We chose the current state-of-the-art regression model for individual TC images presented in [1], which is a convolutional neural network (CNN) model based on AlexNet without dropout layers as shown in Fig. 2. We re-produced this model in TensorFlow using the same data pre-processing procedures, data augmentation (rotating the images by arbitrary degrees before training), and hyper parameters as the ones used in [1] and achieved the same RMSE result on the same test set reported in [1]. We also followed the blending procedure presented in [1]. Given an input image, the image is rotated by evenly-split angles to produce multiple inputs to the model and the resultant predictions are averaged to be the final predicted intensity. We used the the open benchmark dataset released by [1], Tropical Cyclone for Image-to-intensity Regression (TCIR) dataset, which can be downloaded from https://www.csie.ntu.edu.tw/htlin/program/TCIR. We used 39811 satellite images of TCs in 2003 $\sim$ 2014 as the training set for training the regression CNN model. We randomly partitioned 11060 satellite images of TCs in 2015 $\sim$ 2017, and used one half as the validation set for constructing the selective regression model, and the other half as the test set. We used the squared error loss as the loss function and MSE to measure the model regression risk.

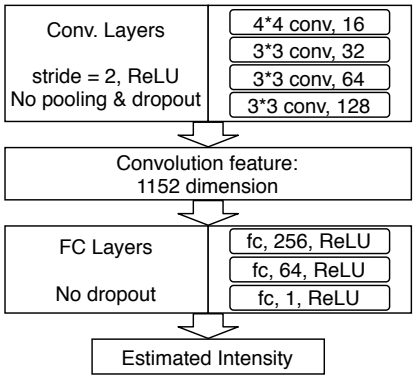


Fig. 2. Network architecture for TC intensity estimation [1].

2) Apparent Age Estimation: Apparent age estimation, which tries to estimate the age as perceived by other humans

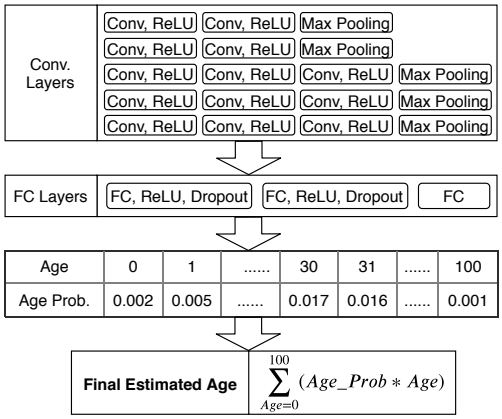


Fig. 3. Network architecture for apparent age estimation [26].

from a facial image, is different from the biological (real) age prediction. References [25] and [26] built CNNs based on VGG-16 and achieved the state-of-the-art results for both real and apparent age estimation on the largest apparent age annotation dataset, ChaLearn Looking At People (LAP) dataset [4]. We downloaded their apparent age estimation model trained already on the LAP dataset from https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/. In their model, apparent age estimation was treated as a multi-class classification of age bins, *i.e.,* 101 age bins from 0 to 100, followed by a softmax layer to output final estimated age, as shown in Fig. 3. The LAP dataset also provided a validation set of 1043 images and a test set of 1003 images. We used the absolute error loss as the loss function and MAE to measure the model regression risk.

B. Choice of κ_f

1) Blend-Var: For the CNN model shown in Fig. 2, we cannot use MC-dropout as the uncertainty function, as it does not have any dropout layers. It was reported in [1] that dropout layers were harmful for prediction performance. We adopt Blend-Var in this case as our choice of the uncertainty function. Suppose we blend predictions from $T = 10$ rotations for each input image, which means we rotate an image by 0, 36, 72, ... , 288, 324 degrees, then treat them as inputs to the model to obtain ten predictions. The Blend-Var uncertainty function calculates the variance of those predictions.

We plot the regression error $f(x) - y$ for each sample in the validation set in the ascending order of κ_f with $T = 10$ in Fig. 4. We can see that as the Blend-Var uncertainty increases, the regression errors indeed tend to increase as well, which means that our choice of κ_f here is a good estimation of the model uncertainty.

2) MC-dropout: The architecture of the apparent age estimation model, as shown in Fig. 3, contains dropout layers. Since no other ensemble method was applied in [25] and [26], we adopt *MC-dropout* as the uncertainty function κ_f and performed $T = 20$ stochastic forward passes through the network to calculate the MC-dropout uncertainty.

We plot the regression error $f(x) - y$ for each sample in the validation set in the ascending order of κ_f in Fig. 5.

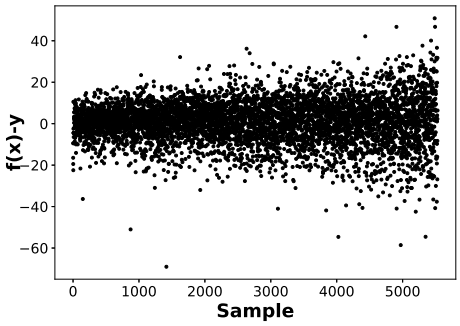


Fig. 4. Regression errors in order of κ_f for TC intensity estimation.

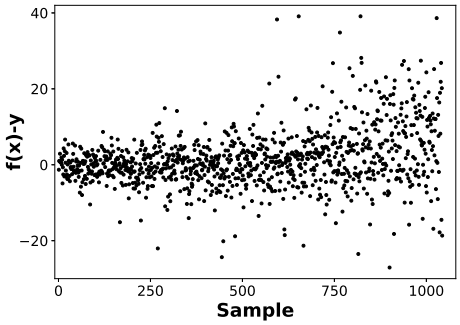


Fig. 5. Regression errors in order of κ_f for apparent age estimation.

Similarly, the MC-dropout uncertainty function estimates the model uncertainty well in this case.

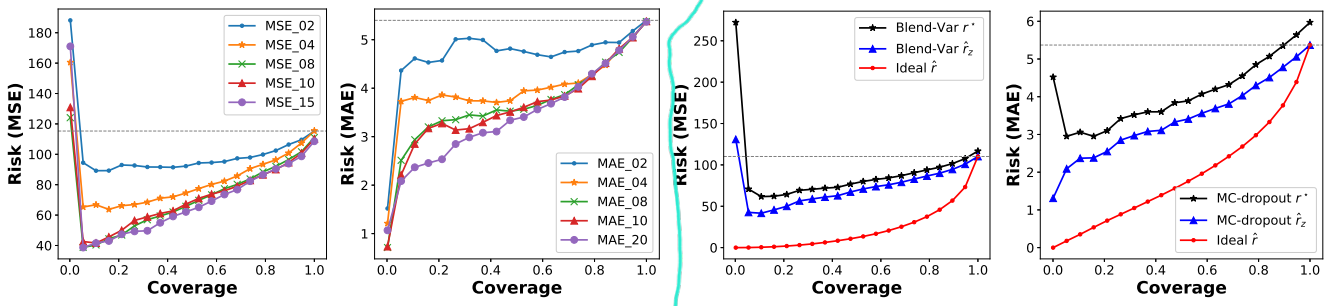
C. Varying T

Noticing that the number of predictions T used in calculating both the Blend-Var and MC-dropout uncertainty functions is a parameter of the regression model. We tried various T values to obtain various regression models and examined their regression performance and the effectiveness of our SGRR algorithm. Using the Blend-Var and MC-dropout uncertainty functions discussed above, we applied our SGRR algorithm on those models and showed their risk-coverage performance in Fig. 6(a) and (b), where the selective risk was evaluated using MSE and MAE for TCIR and LAP, respectively.

As shown in Fig. 6(a) and Fig. 6(b), the dashed line is the original regression risk without blending and selection. When T increases, the performance shows remarkable improvements for both Blend-Var and MC-dropout. The improvements become subtle after $T \geq 10$. Meanwhile, with a larger T value, we need to calculate more predictions, which takes more computing time. Hence, in the rest of the experiments, we set $T = 10$ for TC intensity estimation and $T = 20$ for apparent age estimation, respectively.

D. Confidence Level

Since we used an approximated $b - a$ in our SGRR algorithm, in this set of experiments we verify whether the empirical risk $\hat{r}_z$ is indeed less than the desired risk bound r^* on the test set with the given confidence level. For both TCIR



(a) The influence of T on TCIR. (b) The influence of T on LAP. (c) Risk-Coverage curve on TCIR. (d) Risk-Coverage curve on LAP.

Fig. 6. Selection results.

TABLE I
PERCENTAGE OF EMPIRICAL RISK $\leq$ RISK BOUND ON THE TEST SET OVER 1,000 RUNS.

TCIR	**MSE* (r*)**	73	81	91	101	109
	Percentage	97.6%	98.3%	98.3%	97.7%	98.1%
LAP	**MAE* (r*)**	4	4.5	5	5.2	5.5
	Percentage	98.7%	98.2%	98.9%	99.2%	99.7%

and LAP, we split the dataset into two random halves, one for validation and one for testing, for 1,000 times. We then applied the SGRR algorithm on each validation set with a range of desired risk bound r^* values and a confidence level $\delta = 0.05$. For each r^* value, we constructed a selective regression model g_θ on each validation set, applied it to the corresponding test set, and calculated the percentage of $\hat{r}_z \leq r^*$ on the test set over the 1,000 runs. The percentages are shown in Table I. We can see that although we used an approximated $b - a$, the actual percentages are greater than 95% for all desired risk bound values for both TCIR and LAP, which suggests that the Hoeffding bound with a confidence level can provide a good guidance for constructing selective regression models.

E. Selection Results

In this set of experiments, we examined the selection performance of our SGRR algorithm in more details. For TCIR, we chose one random split of the validation and test sets and evaluated the risk-coverage curve on the validation set and the selection performance on the test set. For LAP, we used the original split of the validation and test sets from the LAP website. Our SGRR algorithm employed both the proposed uncertainty functions and an ideal uncertainty function $\kappa_f(x) = \ell(f(x), y)$, which ranks each sample in the order of its regression error.

1) TC Intensity Estimation: We first show the risk-coverage curve obtained by the selective regression model on the validation set for both the Blend-Var and ideal uncertainty functions in Fig. 6(c). For the selective regression model using the Blend-Var uncertainty function, we drew two curves of the risk bound r^* and the empirical risk $\hat{r}_z$ in terms of MSE with increasing coverage. We drew a curve of the empirical risk $\hat{r}$ for the one using the ideal uncertainty function. Here we started the curves from the coverage value of 0.01. We also drew a dashed line to show the original regression risk without selection. There are several observations we can make from for Fig. 6(c). Firstly, for both the Blend-Var and ideal uncertainty functions, we can see that the empirical risk $\hat{r}_z$ bounded by r^* increases when the coverage increases as we expected. Secondly, the gap t between r^* and $\hat{r}_z$ is large when the coverage is small, because small z values make $t = \sqrt{-\frac{(b-a)^2 \ln \delta}{2z}}$ large. When the coverage increases, the gap t becomes stable and relatively small w.r.t. both r^* and $\hat{r}_z$. Therefore, we can use our SGRR algorithm to search for g_θ only when g_θ selects more than m_0 (the minimum number of samples needed) samples on the validation set. Finally, compared with the perfect empirical risk-coverage curve achieved by the ideal uncertainty function, Blend-Var achieved a worse empirical risk-coverage curve. However, as suggested by Fig. 6(c), Blend-Var can still construct selective regression models that decrease the regression risk significantly while covering most of the samples.

Given a risk bound r^*, we used the SGRR algorithm to search for g_θ on the validation set. g_θ was then applied to the test set. We calculated the empirical risk (in terms of MSE) and coverage on both validation and test sets for TCIR, which are shown in Table II. The original model f achieved a RMSE value of 10.496 knots (equivalent to an MSE value of 110.16) and 10.486 knots (equivalent to an MSE value of 109.95) on the validation and test sets, respectively. As shown in Table II, the MSE and coverage values are very similar on the validation and test sets. Both val-MSE and test-MSE values are bounded by r^* with a gap introduced by the Hoeffding's inequality. Finally, our SGRR algorithm with Blend-Var as the uncertainty function constructed a selective regression model that guaranteed a risk bound (in terms of MSE) of 90.25 while covering more than 75% test samples with a guided confidence level of 0.05.

2) Apparent Age Estimation: The risk-coverage curves of the risk bound r^* using MC-dropout, the empirical risk $\hat{r}_z$ using MC-dropout, and the empirical risk $\hat{r}$ using the ideal uncertainty function in terms of MAE with increasing coverage for LAP are shown in Fig. 6(d). Trends are similar for this case to those for TCIR as well. Because the size of the validation

TABLE II
SELECTION RESULTS ON TCIR WITH $\delta = 0.05$.

MSE*(r*)	val-MSE	val-Coverage	test-MSE	test-Coverage
72.25	65.81	45.82	64.94	46.84
81	74.47	60.51	76.47	61.03
90.25	83.82	75.37	85.28	75.90
100	93.33	88.57	97.08	89.42
105	98.18	91.74	99.87	92.48
109	102.02	95.57	103.19	96.24
-	110.16	100	109.95	100

TABLE III
SELECTION RESULTS ON LAP WITH $\delta = 0.05$.

MAE* (r*)	val-MAE	val-Coverage	test-MAE	test-Coverage
4	3.19	42.95	3.46	43.27
4.5	3.88	70.47	4.03	66.54
5	4.40	82.65	4.36	79.48
5.2	4.64	86.67	4.53	86.54
5.5	4.96	93.10	4.86	93.22
-	5.37	100	5.22	100

set is smaller (around 1,000) for LAP, the gap between r^* and $\hat{r}_z$ using the MC-dropout uncertainty function is larger, which means the risk bound is looser. Nevertheless, MC-dropout can be used to construct selective regression models that decrease the regression risk significantly while covering most of the samples. Similarly, we applied the found selective regression model to the test set, and calculated the empirical risk (in terms of MAE) and coverage on both validation and test sets, which are shown in Table III. The original model f achieved a MAE value of 5.37 and 5.22 on the LAP validation set and the LAP test set, respectively. Our SGRR algorithm with MC-dropout as the uncertainty function constructed a selective regression model that guaranteed a risk bound (in terms of MAE) of 4.5 while covering more than 66% test samples with a guided confidence level of 0.05.

VI. CONCLUSION

We presented a general method to construct selective regression DNN models that can minimize reject rates under the control of regression risk bounds. We proposed an uncertainty estimation function, *Blend-Var*, which could be used in both classification and regression DNNs with blending. We evaluated our proposed method with two real-world applications and achieved promising results, which suggests that selective regression models are promising solutions to the real-world applications where predictions with large regression errors need to be avoided.

REFERENCES

[1] B. Chen, B.-F. Chen, and H.-T. Lin, "Rotation-blended cnns on a new open dataset for tropical cyclone image-to-intensity regression," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 90–99.

[2] J. Miller, M. Maskey, and T. Berendes, "Using deep learning for tropical cyclone intensity estimation," in *AGU Fall Meeting Abstracts*, 2017.

[3] J. S. Combinido, J. R. Mendoza, and J. Aborot, "A convolutional neural network approach for estimating tropical cyclone intensity using satellite-based infrared images," in *Proceedings of the 24th International Conference on Pattern Recognition*. IEEE, 2018, pp. 1474–1480.

[4] S. Escalera, J. Fabian, P. Pardo, X. Baro, J. Gonzalez, H. J. Escalante, D. Misevic, U. Steiner, and I. Guyon, "Chalearn looking at people 2015: Apparent age and cultural event recognition datasets and results," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 1–9.

[5] G. Levi and T. Hassner, "Age and gender classification using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 34–42.

[6] A. S. Qureshi, A. Khan, A. Zameer, and A. Usman, "Wind power prediction using deep neural network based meta regression and transfer learning," *Applied Soft Computing*, vol. 58, pp. 742–755, 2017.

[7] J. Zhou, X. Hong, F. Su, and G. Zhao, "Recurrent convolutional neural network regression for continuous pain intensity estimation in video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 84–92.

[8] C.-K. Chow, "An optimum character recognition system using decision functions," *IRE Transactions on Electronic Computers*, no. 4, pp. 247–254, 1957.

[9] L. P. Cordella, C. De Stefano, F. Tortorella, and M. Vento, "A method for improving classification reliability of multilayer perceptrons," *IEEE Transactions on Neural Networks*, vol. 6, no. 5, pp. 1140–1147, 1995.

[10] C. De Stefano, C. Sansone, and M. Vento, "To reject or not to reject: that is the question-an answer in case of neural classifiers," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 30, no. 1, pp. 84–94, 2000.

[11] R. El-Yaniv and Y. Wiener, "On the foundations of noise-free selective classification," *Journal of Machine Learning Research*, vol. 11, no. May, pp. 1605–1641, 2010.

[12] Y. Geifman and R. El-Yaniv, "Selective classification for deep neural networks," in *Advances in neural information processing systems*, 2017, pp. 4878–4887.

[13] R. Pradhan, R. S. Aygun, M. Maskey, R. Ramachandran, and D. J. Cecil, "Tropical cyclone intensity estimation using a deep convolutional neural network," *IEEE Transactions on Image Processing*, vol. 27, no. 2, pp. 692–702, 2017.

[14] C. Chow, "On optimum recognition error and reject tradeoff," *IEEE Transactions on information theory*, vol. 16, no. 1, pp. 41–46, 1970.

[15] Y. Geifman and R. El-Yaniv, "Selectivenet: A deep neural network with an integrated reject option," *arXiv preprint arXiv:1901.09192*, 2019.

[16] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*, 2016, pp. 1050–1059.

[17] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[18] G. S. Babu, P. Zhao, and X.-L. Li, "Deep convolutional neural network based regression approach for estimation of remaining useful life," in *Proceedings of the 21st International conference on database systems for advanced applications*, 2016, pp. 214–228.

[19] R. Herbei and M. H. Wegkamp, "Classification with reject option," *Canadian Journal of Statistics*, vol. 34, no. 4, pp. 709–721, 2006.

[20] W. Hoeffding, "Probability inequalities for sums of bounded random variables," in *The Collected Works of Wassily Hoeffding*. Springer, 1994, pp. 409–426.

[21] H. Chernoff *et al.*, "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," *The Annals of Mathematical Statistics*, vol. 23, no. 4, pp. 493–507, 1952.

[22] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge university press, 2006.

[23] L. Zhao, L. Wang, and D.-w. Cui, "Hoeffding bound based evolutionary algorithm for symbolic regression," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 5, pp. 945–957, 2012.

[24] O. Maron, "Hoeffding races–model selection for mri classification," Ph.D. dissertation, Massachusetts Institute of Technology, 1994.

[25] R. Rothe, R. Timofte, and L. Van Gool, "Dex: Deep expectation of apparent age from a single image," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 10–15.

[26] ——, "Deep expectation of real and apparent age from a single image without facial landmarks," *International Journal of Computer Vision*, vol. 126, no. 2-4, pp. 144–157, 2018.