2022 IEEE International Conference on Assured Autonomy (ICAA) | 978-1-6654-8539-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICAA52185.2022.00015

Selective Classification of Sequential Data Using Inductive Conformal Prediction

Dimitrios Boursinos and Xenofon Koutsoukos
Institute for Software Integrated Systems
Vanderbilt University
Nashville TN, USA
{dimitrios.boursinos, xenofon.koutsoukos}@vanderbilt.edu

***Abstract*—Cyber-Physical Systems (CPS) operate in dynamic and uncertain environments where the use of deep neural networks (DNN) for perception can be advantageous. However, DNN integration in CPS is not straightforward. Perception outputs must be complemented with assurance metrics that represent if they can be trusted or not. Further, the inputs to DNNs are typically sequential capturing time-correlated data that can affect the accuracy of the predictions since machine learning models require inputs to be independent and identically distributed. In this paper, we propose a selective classification approach that rejects predictions that are not trustworthy. We quantify the credibility and confidence of each prediction by computing aggregate p-values from multiple subsequent inputs. We examine different multiple hypothesis testing approaches for combining p-values computed using Inductive Conformal Prediction (ICP) focusing on their ability to produce valid p-values for sequential data. Empirical evaluation results using the German Traffic Sign Recognition Benchmark demonstrate that ICP validity can be recovered when p-values from sequential inputs are combined and selective classification based on aggregate p-values produces predictions with less risk.**

***Index Terms*—selective classification, assurance evaluator, inductive conformal predictors, distance metric learning**

I. Introduction

Cyber-Physical Systems (CPS) integrate computing, monitoring and control for operation in the physical world. Perception of the environment is a complex process because of the existence of objects that are difficult to model and have complex interaction with the controlled system. Deep Neural Networks (DNN) have the capacity to be trained and generalize their knowledge to make predictions in dynamic environments. CPS can benefit from the integration of DNNs but assurance guarantees are needed that are very challenging to compute. In CPS applications such as autonomous vehicles that needs to recognize traffic signs and take the correct control decisions, the cost of an incorrect classification is much higher than not making any classification when there is no clear distinction between the best prediction and the alternatives. In such a setting, the operation cost over time can be minimized using selective classifiers that evaluate the risk in each classification and either accept the classification or reject it.

Most discriminative machine learning (ML) frameworks make predictions with some notion of confidence under the assumption that the input data are independent and identically distributed (IID) [15]. When there is dependence between the input data this assumption does not hold and the confidence metrics are not accurate. This is an important challenge for CPS to overcome in order to use such frameworks. Sensors in a CPS perceive processes in the physical world that have some duration and individual time instances have some, usually unknown, dependence to previous instances. This leads to miscalibration of confidence estimates and error-rate guarantees are not satisfied [5].

Our approach for improving the confidence of the predictions is based on Inductive Conformal Prediction (ICP) [24]. ICP aims in producing prediction sets that satisfy any error-rate bound guarantees under the IID assumption. The main idea is to test if a new input example conforms to the training data set by utilizing a *nonconformity measure* which assigns a numerical score indicating how different the input example is from the training data set. For any test input, a p-value is assigned to each possible class to decide if a class should be part of the prediction set or not in order to satisfy the chosen error-rate guarantees. This property is valid when the test inputs are IID.

In this work we improve the calibration of the prediction sets computed by ICP and the classification accuracy when the input data are time correlated. We use statistical methods for computing aggregated p-values resulted from subsequent inputs in a sliding window. We approach the problem as a multiple hypothesis testing problem and show how different combination methods recover ICP validity. Our main contribution is the design of a selective classifier based on ICP that we call assurance evaluator. This classifier decides if a classification is possible based on the computed p-values for each class. When the highest p-value among all the classes is much higher than the second highest computed p-value we can trust the classification more than cases where at least two of the highest p-values are close to each other. Another contribution of this work is the computation of low-dimensional, appropriate, embedding representations of the original inputs in a space where the Euclidean distance is a measure of similarity between the original inputs. This is needed in order to find semantic similarities between data points and handle high-dimensional inputs in real-time. We evaluate the proposed

978-1-6654-8539-5/22/$31.00 ©2022 IEEE
DOI 10.1109/ICAA52185.2022.00015

approach on the German Traffic Sign Recognition Benchmark (GTSRB) which has sequential images of signs as a vehicle moves towards traffic signs.

II. Related Work

Confidence and uncertainty estimation in neural networks has received considerable attention especially in the context of classification tasks. Different frameworks have been developed that evaluate the confidence of predictions in different ways.

Selective classification Methods based on selective classification are used for decision-making when errors are costly and it is beneficial to not make any decision when none of the possible classes is trustworthy. Different threshold choices affect the number of errors, or *risk* and the decision frequency, or *coverage*. The trade-offs between the risk and coverage are studied in [8]. The use of selective classification in the context of DNNs is considered in [10], [11]. The process of learning a selective classifier with a chosen desired risk is shown in [10]. An uncertainty estimation algorithm to be used with selective classification in DNNs is proposed in [11]. This avoids the over-confident probability estimations that are common in DNN classifiers.

Output calibration A variety of methods have been developed for quantifying predictive uncertainty in ML models by calibrating the output values to represent real probabilities. The Platt's scaling method [27] is proposed for the calibration of Support Vector Machine (SVM) outputs. After the training of an SVM, the method computes the parameters of a sigmoid function to map the outputs into probabilities. Piecewise logistic regression is an extension of Platt scaling and assumes that the log-odds of calibrated probabilities follow a piecewise linear function [40]. Another variant of Platt's scaling is temperature scaling [12] which can be applied in DNNs with a softmax output layer. After training of a DNN, a temperature scaling factor T is computed on a validation set to scale the softmax outputs to represent true probabilities. However, while the temperature scaling shows good calibration results when the input data are IID, there is no calibration guarantee under distribution shifts [23]. Recent evaluation of Platt's scaling and tempereture scaling presented in [17] shows that they are not as well-calibrated as it is reported and it is difficult to know how miscalibrated they are.

Conformal prediction Unlike most classification models that make point predictions for a given input, the conformal prediction framework [37], [32], [2] computes prediction sets that bound the error-rate to a chosen value. In [25] the authors suggest a modified version of the CP framework, they call Inductive Conformal Prediction (ICP), that has less computatonal overhead and they evaluate the results using DNNs as undelying model. Deep k-Nearest Neighbors (DkNN) is an approach based on ICP for classification problems that uses the activations from all the hidden layers of a neural network as features to the ICP [26]. Other implementations have been proposed for decision trees [13], random forests [3], [7] as well as SVMs [20]. CP and ICP require that the data are IID and exchangeable.

III. Problem Formulation

A perception component in a CPS aims to observe and interpret the environment in order to provide information for decision making. In safety-critical systems, predictions on unseen inputs need to have a well-calibrated and bounded error-rate according to predefined safety rules. An efficient and robust approach must ensure a small and well-calibrated error rate while limiting the number of alarms to enable real-time monitoring. Finally, it must be computationally efficient for applications operating on high-dimensional data that require low latency like, for example, in autonomous vehicles.

During the system operation of a CPS, inputs arrive one by one. The inputs may be dependent on each other as shown in Figure 1, for example, in a traffic sign recognition system. After receiving each input, the objective is to compute a valid prediction set that satisfies a bound on the error-rate as well as produce classifications based on their trustworthiness. The objective is twofold: (1) provide guarantees for the error-rate of the classification and (2) design an assurance evaluator which minimizes the number of input examples for which a confident prediction cannot be made. The assurance evaluator operates as a selective classifier that generates warnings when no classification can be made and human intervention is needed.

IV. Selective Classification

ICP computes p-values for each class to construct prediction sets with a chosen significance level. However, its applications are not limited to cases where valid prediction sets are needed. We use multiple hypothesis testing methods to combine the p-values computed on multiple time instances. The aggregate p-values indicate the trustworthiness of each class for particular inputs, over a time horizon, and can be used for point predictions. We define the *credibility* and *confidence* metrics based on the two highest aggregate p-values, $p_{(c)}, p_{(c-1)}$, of all possible classes $p_i, i = 1, \ldots, c$:

$$\text{credibility} = p_{(c)} \tag{1}$$

$$\text{confidence} = 1 - p_{(c-1)} \tag{2}$$

For a test input x_{l+1} and classification $\hat{y} = \arg\max_{i=1,\ldots,c} p_i$, the credibility shows how credited $\hat{y}$ is and the confidence shows how special it is compared to the other possible labels. These two metrics define the four scenarios shown in Table I. The preferred situation is when the largest p-value is close to one and the rest close to zero. We use an assurance evaluator to decide if a trusted classification can be made and, if not, it will raise an alarm which may require further investigation. For this operation we use the concept of *selective classification* [8], [39]. A selective classifier (f, g) decides whether to keep the classification from an underlying model or reject it and is defined as:

$$(f, g)(x) \triangleq \begin{cases} f(x), & \text{if } g(x) = 1 \\ \text{reject}, & \text{if } g(x) = 0 \end{cases} \tag{3}$$

Figure 1: Traffic sign over time (in frames)

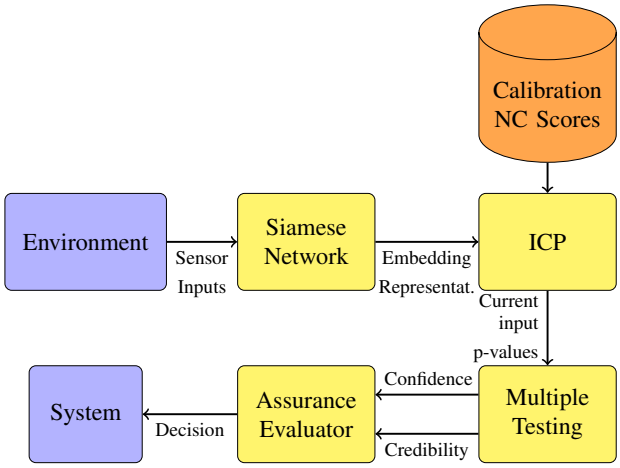


Figure 2: Execution time architecture

Table I: Scenarios that can be observed for different values of confidence and credibility.

Credibility	Confidence	Description
High	High	The prefered situation that usually leads into accepting a classification. $p_{\hat{y}}$ is high and much higher than the p-values of the other classes.
High	Low	$p_{\hat{y}}$ is high but there are other high p-values so choosing a single credible class may not be possible.
Low	High	None of the p-values are high for a credible decision.
Low	Low	Not applicable.

where f is the ICP based classifier, and $g : \mathcal{X} \to \{0, 1\}$ is a selective function that we call *assurance evaluator*. Consider a function k that evaluates the classifications of f and a threshold θ. The selective function g is defined as, $g_\theta(x|k, f) = \mathbb{1}[k(x, \hat{y}_f(x)|f) > \theta]$. A selective classifier is evaluated using the *coverage* and *risk* metrics. The coverage, $\phi(f, g)$, measures the frequency that the classifications of f are accepted. The risk, $R(f, g)$, is the error-rate in the accepted classifications. These measures can be empirically evaluated using any finite labeled set S_m. The empirical coverage $\hat{\phi}$ and risk $\hat{r}$ are computed as:

$$\hat{\phi}(f, g|S_m) \triangleq \frac{1}{m} \sum_{i=1}^{m} g(x_i) \tag{4}$$

$$\hat{r}(f, g|S_m) \triangleq \frac{\frac{1}{m} \sum_{i=1}^{m} l(f(x_i), y_i) g(x_i)}{\hat{\phi}(f, g|S_m)} \tag{5}$$

where $l(f(x_i), y_i) = 1$ if $f(x_i) = y_i$ otherwise $l(f(x_i), y_i) = 0$.

For a given classifier f we optimize the assurance evaluator g based on the area under the risk-coverage (RC) curve (AURC) defined in [11]. Consider a set of n labeled points V_n and let the set $\Theta \triangleq \{k(x, \hat{y}_f(x)|f) : (x, y) \in V_n\}$ of threshold values. Using these threshold values to define the selective function g we can compute n empirical risk and coverage values and plot a RC curve. When two assurance evaluators are compared, preferable is the one with lower risk at the same coverage. So a metric for evaluation of pairs (f, g) is the AURC:

$$\text{AURC}(k, f|V_n) = \frac{1}{n} \sum_{\theta \in \Theta} \hat{r}(f, g_\theta|V_n). \tag{6}$$

The assurance evaluator is constructed with a choice of a classification evaluator function k and a threshold θ. A function k needs to be chosen to minimize AURC, which intuitively minimizes the average empirical risk. We express k as a linear combination of the credibility and confidence, computed by ICP,

$$\begin{aligned} k(x, \hat{y}_f(x)) = a \cdot \text{credibility}(x, \hat{y}_f(x)) \\ + b \cdot \text{confidence}(x, \hat{y}_f(x)) \end{aligned} \tag{7}$$

We compute the optimal parameters a and b that minimize the AURC with a grid search in $[-1, 1]$. Based on the RC curve and the application requirements regarding the accepted risk and coverage of the assurance evaluator (r^*, c^*), a threshold θ is chosen such that $(\hat{r}, \hat{c}) = (r^*, c^*)$.

V. Distance Metric Learning

We consider a training set $\{z_1, \ldots, z_l\}$ of examples, where each $z_i \in Z$ is a pair (x_i, y_i) with x_i the feature vector and y_i the label of that example. This set is split into the proper training set $(z_1, \ldots, z_m)$ of size $m < l$ and the calibration set $(z_{m+1}, \ldots, z_l)$ of size $l - m$. ICP tests the trustworthiness of a candidate class, $\hat{y}_{l+1}$, with respect to a given input, x_{l+1},

by computing how similar the test pair $(x_{l+1}, \hat{y}_{l+1})$ is to the pairs $(x_i, y_i), i = 1, \ldots, m$ in the proper training set. When the input data are high-dimensional, for example images, computing the Euclidean distance between two inputs is not a proper way to estimate their similarity. The Euclidean distance does not take into consideration the spatial relationships of pixels so small translations or rotations between similar images may lead to a large distance.

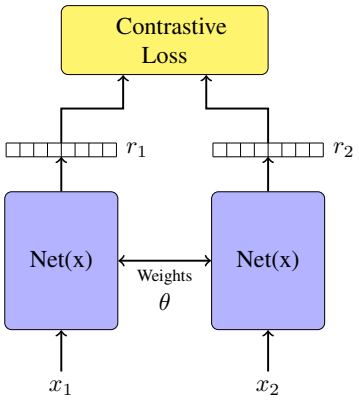


Figure 3: Siamese network training

We use a *siamese network* to transform the original inputs into lower dimensional embedding representations in a space where the Euclidean distance is a measure of how semantically similar the original inputs are. The advantage of siamese networks over classical analytical distance metric learning methods is that it can scale to larger dimensions like the data of our test cases. A siamese network is composed using two copies of the same neural network with shared parameters [16] as shown in Figure 3. During training, each identical copy of the siamese network is fed with different training samples x_1 and x_2 belonging to classes y_1 and y_2. The embedding representations produced by each network copy are $r_1 = \text{Net}_\theta(x_1)$ and $r_2 = \text{Net}_\theta(x_2)$. The learning goal is to compute the Net parameters θ^* that minimize the Euclidean distance between the embedding representations of inputs belonging to the same class and maximize it for inputs belonging to different classes:

$$\begin{cases} \min d(r_1, r_2), & \text{if } y_1 = y_2 \\ \max d(r_1, r_2), & \text{otherwise} \end{cases} \tag{8}$$

This optimization problem can be solved using the *contrastive loss* function [21]:

$$L(r_1, r_2, y) = y \cdot d(r_1, r_2) + (1 - y) \max[0, m - d(r_1, r_2)]$$

where y is a binary flag equal to 0 if $y_1 = y_2$ and to 1 if $y_1 \neq y_2$ and m is a margin parameter. In particular, when $y_1 \neq y_2$, $L = 0$ when $d(r_1, r_2) \geq m$, otherwise the parameters of the network are updated to produce more distant representations for those two elements. The reason behind the use of the margin is that when the distance between pairs of different classes are large enough and at most m, there is no reason to update the network to put the representations even further away from each other and instead train on harder examples.

We denote $f : X \to V$ the mapping from the input space X to the embedding space V by a single copy of the DNN pair in the siamese network. Using the trained network, the embeddings $v_i = f(x_i)$ are computed and stored for all the training data x_i. The same transformation takes place online as new test input data arrive to the system.

VI. Multiple Testing Of Single Hypothesis

ICP forms prediction sets with theoretical guarantees on the error-rate based on computations of p-values for each class. When the inputs to be classified are composed of sequential data arriving one after the other, it is natural to utilize statistical methods for combining individual p-values to improve the accuracy and efficiency over the individual classifications. We, first, briefly present how ICP computes p-values and prediction sets and in then second part of this section we present different ways of computing aggregate p-values.

A. Inductive Conformal Prediction

Given a test input x_{l+1}, ICP computes a prediction set Γ^ϵ of labels with enough evidence to be the true label. We consider the more fundamental question: given a test input x_{l+1} belonging in class $y_{l+1} \in Y$, is label $\hat{y}_{l+1} : \hat{y}_{l+1} \in Y$ the true label? Hypothesis testing is a statistical method used to make decisions on whether a hypothesis is true based on a finite number of data. The question to be answered is translated into two competing and non-overlapping hypothesis. (1) The *null hypothesis*, H_0, is the argument believed to be true and (2) the *alternative hypothesis*, H_1, is the argument to be proven true based on the collected data. We determine whether to accept or reject the alternative hypothesis based on the likelihood of the null hypothesis being true. Considering again a test input x_{l+1}, the question whether $\hat{y}_{l+1}$ is the true class, is written using the above notation. We are certain that exactly one of the labels in Y is true so $\hat{y}_{l+1} = y_{l+1}$ is the null hypothesis. This hypothesis needs to be rejected for the $c - 1$ incorrect labels so $\hat{y}_{l+1} \neq y_{l+1}$ is the alternative hypothesis.

The p-value is a measure of how likely it is that the pair $(x_{l+1}, \hat{y}_{l+1})$ has occurred under the null hypothesis, $\hat{y}_{l+1} = y_{l+1}$. It is the probability for this data point to occur or something that is as, or more, extreme. On the assumptions that the null hypothesis is true and that the sampling distribution is given by a probability density function (PDF), the distribution of p-values is uniform in the interval $[0, 1]$.

The null distribution is usually unknown in practice and ICP approaches it using a labeled calibration set. First we consider a training set $\{z_1, \ldots, z_l\}$ of examples, where each $z_i \in Z$ is a pair (x_i, y_i) with x_i the feature vector and y_i the label of that example. This set is split into the proper training set $(z_1, \ldots, z_m)$ of size $m < l$ and the calibration set $(z_{m+1}, \ldots, z_l)$ of size $l - m$. Central to the framework is the use of nonconformity measures (NCM), a metric that indicates how different an example z_{l+1} is from the examples of the

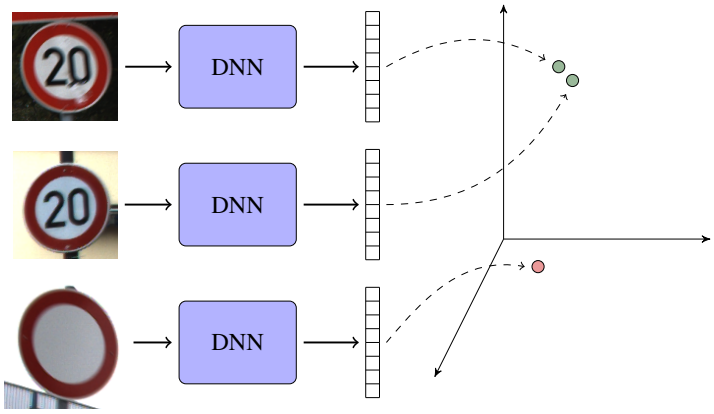


Figure 4: Image mapping to a lower dimensional space based on their similarity.

training set. Assuming that the test data are generated from the same distribution as the calibration data, the null distribution is the distribution of the nonconformity (NC) scores of the calibration set pairs (x_i, y_i), $i = m+1, \ldots, l$.

NCM is a function that computes the dissimilarity between an example z_{l+1} and the examples of the training set $z_1, \ldots, z_l$. There are many different possible NCMs that can be used [4], [6], [2], [37], [32], [14]. Having used all the above NCMs in our previous research work, we find the *nearest centroid* to have good trade-off characteristics between the NC scores quality, the memory requirements and the applicability in real-time systems when combined with distance metric learning. The nearest centroid NCM simplifies the task of computing individual training examples that are similar to a test input when there is a large amount of training data. We expect the embedding representations of examples that belong to a particular class to be close to each other, so for each class y_i we compute its centroid $\mu_{y_i} = \frac{\sum_{j=1}^{n_i} v_j^i}{n_i}$, where v_j^i is the embedding representation of the j^{th} training example from class y_i and n_i is the number of training examples in class y_i. The NC function is then defined as

$$\alpha(x, y) = \frac{d(\mu_y, v)}{\min_{i=1,\ldots,n:y_i \neq y} d(\mu_{y_i}, v)} \tag{9}$$

where v the embedding representation of the feature vector x. It should be noted that for computing the nearest centroid NCM, we need to store only the centroid of each class.

The trustworthiness of a particular class given a test input is expressed as a p-value. It is computed as the fraction of the calibration NC scores that are greater or equal to the NC score computed for the current hypothesis testing. Assume a test input x_{l+1}, $v_{l+1} = f(x_{l+1})$, and we test the null hypothesis, the class y^j is the correct class, $y_{l+1} = y^j$:

$$p_j(x_{l+1}) = \frac{|\{\alpha \in A : \alpha \geq \alpha(x_{l+1}, y^j)\}|}{|A|} \tag{10}$$

where A the set of calibration NC scores and $\alpha(x_{l+1}, y^j)$ the NC score of the pair (x_{l+1}, y^j). This hypothesis is accepted if $p_j(x_{l+1}) > \epsilon$, where ϵ the significance level. This process is repeated for all possible classes $y^j, j = 1, \ldots, |Y|$. All the classes that were not rejected for a chosen ϵ are added in the prediction set Γ^ϵ.

ICP computes well-calibrated prediction sets, defined as $P(y_{l+1} \notin \Gamma^\epsilon) < \epsilon$, for any choice of ϵ with the underlying assumption that all examples (x_i, y_i), $i = 1, 2, \ldots$ are IID generated from the same but typically unknown probability distribution and exchangable [33]. However, the choice of ϵ is important for the computation of efficient prediction sets. The best case scenario for Γ^ϵ is $|\Gamma^\epsilon| = 1$ and the only class that satisfies $p_j(x_{l+1}) > \epsilon$ is the ground truth class.

B. Combining Multiple p-values

The problem of multiple hypothesis testing appears when a decision about a null hypothesis needs to be made after a number of tests $K > 1$. According to the problem we consider in this paper, the same null hypothesis is tested over K consecutive frames of a sequence. The p-values $p_1, \ldots, p_K$, obtained from the K individual hypothesis tests need to be combined into a single p-value. Since the individual tests take place on consequent frames of a sequence, it is expected the p-values are dependent with each other. For this combination to be used in the ICP framework, the combined p-values during testing should lead to valid prediction sets with error-rate equal to the chosen significance-level.

Merging Functions In [38], authors propose general methods for combining a number of p-values without making any assumptions about their dependence structure. Different functions for combining p-values can be derived from the generalized mean p-value (GMP):

$$M_{r,K}(p_1, \ldots, p_K) = \left(\frac{p_1^r + \cdots + p_K^r}{K} \right)^{1/r} \tag{11}$$

where $r \in [-\infty, \infty]$, $K = 2, 3, \ldots,$. Merging functions for combing p-values without the independence assumption are constructed as

$$p_{\text{comb}} = a_{r,K} M_{r,K}(p_1, \ldots, p_K)$$

Special cases of merging functions that we use in our evaluation and are derived from (11) are the minimum, maximum, arithmetic mean and geometric mean. When $r \to -\infty$ the resulting merging function is known as the Bonferroni method and one of the most well-known methods for multiple testing:

$$p_{\text{min}} = K \min(p_1, \ldots, p_K) \tag{12}$$

Similarly when $r \to \infty$:

$$p_{\text{max}} = \max(p_1, \ldots, p_K) \tag{13}$$

(12) and (13) are generalized in [30] to use order statistics of the p-values:

$$p_{\text{ord}} = \frac{K}{k} p_{(k)} \tag{14}$$

where $p_{(k)}$ is the kth smallest p-value of the $p_1, \ldots, p_K$. When $r = 1$ individual p-values are combined using the arithmetic average. However the arithmetic average of a number of p-values is not always a p-value. In [31], Theorem 1, it is shown that multiplying the arithmetic average with a factor of 2 is a p-value.

$$p_{\text{arith\_avg}} = \frac{2}{K} \sum_{i=1}^{K} p_i \tag{15}$$

When $r = 0$ individual p-values are combined using the geometric average:

$$p_{\text{geom\_avg}} = e \times \left(\prod_{i=1}^{K} p_i \right)^{1/K} \tag{16}$$

Quantile Combination Approaches The general quantile combination approach produces p-values that are uniformly distributed in $[0, 1]$ when the combined p-values are independent. If $X_1, X_2, \ldots, X_n$ samples from a continuous distribution X with CDF F_X, then the samples $U_i = F_X(X_i)$ follow a uniform distribution U with CDF $F_U(u) = u$. The proof is straightforward and is added here for completeness.

$$\begin{aligned} F_U(u) &= \mathbb{P}[U \le u] = \mathbb{P}[F_X(X) \le u] = \\ &= \mathbb{P}\left[X \le F_X^{-1}(u)\right] = F_X\left(F_X^{-1}(u)\right) = u \end{aligned} \tag{17}$$

The advantage of leveraging this property is that p-values can be combined using any arbitrary function f and then transform the resulting p-values to a uniform distribution using the CDF of f. However, in our application the p-values computed by ICP on consequent frames are not independent and cannot be considered as independent samples from a continuous distribution. This means that the transformed p-values may not be uniformly distributed affecting the global validity of ICP. Since we do not know the dependence structure between the inputs, these methods could still result in an ICP valid in regions of interest and we experiment with their use in CPS. There is a large number of quantile combination methods proposed in the literature that transform and combine p-values using functions with CDF that can be expressed in closed form or can be computed efficiently. However, because of their independence requirement, this is not an exhaustive list of methods but an evaluation of the most commonly used ones. A number of quantile combination methods is evaluated using ICP computed p-values for multiple underlying model ensembles in [35], [36].

One way of combining multiple p-values is using their product. This is commonly known as the Fisher's method [9]. Assuming that $p_1, p_2, \ldots, p_k$ are samples from a uniform distribution, then

$$h_i = -2 \log p_i$$

follows a chi-squared distribution with 2 degrees of freedom. The sum of independent chi-squared distributions is also a chi-squared distribution with degrees of freedom equal to the sum of the degrees of freedom of the individual chi-squared distributions. The CDF of the chi-squared distribution is expressed in closed form so a sequence of k independent p-values can be combined efficiently as

$$p_{\text{prod}} = \mathbb{P}\left\{ y \le -2 \sum_{i=1}^{K} \log p_i \right\} = t \sum_{i=0}^{K-1} \frac{(-\log t)^i}{i!} \tag{18}$$

where $t = \prod_{i=1}^{K} p_i$.

A similar approach is the Stouffer's z-transform [34], which first maps the uniformly distributed and independent p-values to random variables that follow the normal distribution

$$h_i = \Phi^{-1}(1 - p_i)$$

where Φ is the cumulative normal distribution. The random variables $h_i, i = 1, \ldots, K$ are then combined such that

$$h = \frac{\sum_{i=1}^{K} h_i}{\sqrt{K}}.$$

The sequence of p-values, $p_i, i = 1, \ldots, K$, is combined by sampling the CDF of h

$$p_{\text{z}} = \mathbb{P}\left\{ y \le \frac{\sum_{i=1}^{K} h_i}{\sqrt{K}} \right\} = 1 - \Phi(h) \tag{19}$$

which is not in closed form but easily computed by most mathematical software. This method is extended in [18] to assign weights on independent experiments. This can be useful in our applications as more recent inputs may be more significant than older ones. We call it the weighted Stouffer's method:

$$p_{\text{z\_weighted}} = 1 - \Phi\left(\frac{\sum_{i=1}^{K} w_i Z_i}{\sqrt{\sum_{i=1}^{K} w_i^2}} \right) \tag{20}$$

The weights we assign are larger for recent inputs in a sliding window and decrease over time so that $w_i = i / \sum_{j=1}^{K} j$.

To keep our evaluation of quantile combination methods consistent with the merging function presented earlier, order statistics functions, like min and max, can be used to produce

p-values by sampling their CDF. Let $p_{(r)}$ be the rth smallest among K independent p-values. These p-values follow the $Beta(r, K-r+1)$ distribution [28]. In this case the CDF is an incomplete beta function. It cannot be expressed in closed form but it is easily computed by most mathematical software.

The Cauchi combination test [19] is more recent and although it is based on the quantile combination methods, it is developed to be applied under arbitrary correlation structures. Assuming that $p_1, p_2, \ldots, p_k$ are samples from a uniform distribution, then the components

$$h_i = \tan\{(0.5 - p_i)\pi\}$$

follow a standard Cauchi distribution. The sum $T = \sum_{i=1}^{K} h_i$ also has a standard Cauchi distribution under the null and the its CDF can be computed in closed form:

$$p_{\text{Cauchi}} = \mathbb{P}\left\{ y \leq \sum_{i=1}^{K} h_i \right\} = \frac{1}{2} - \frac{\arctan T}{\pi} \tag{21}$$

Expressed in closed form, similar to the previous methods, it has low computational requirements.

Emprirical CDF Computation In practice, p-value transformations using CDFs are not always possible. The reason is twofold: (1) not all arbitrary combination functions have a CDF that can be expressed in closed form and (2) the p-values to be combined may be dependent. Instead of using CDFs we compute an Empirical Cumulative Distribution Function (ECDF) from a set of calibration sequences. We first compute the combined p-values that are consistent with the null hypothesis using any arbitrary law $f(p_1, \ldots, p_K)$. Then the ECDF $F_X(x)$ is computed on the finite set of combined p-values. During test time when a sliding window of K frames is present, the p-values of each class are combined with an arbitrary law and the computed ECDF is used to recover validity of the combined p-values

$$p_{\text{comb}} = F_X\left[f(p_1, \ldots, p_K)\right] \tag{22}$$

where F_X is the computed ECDF. To understand the effects of ECDF, during evaluation we use simple combination laws consistent with the CDFs above.

VII. Evaluation

Our assurance evaluator design leverages distance metric learning techniques to compress the input data to lower dimensions in order to make the ICP application more efficient and with lower memory requirements. The objective of the evaluation is to compare how the suggested architecture of combining p-values from multiple inputs performs against the baseline ICP approach that processes one input at a time as well as investigate the validity/calibration, efficiency (size of set predictions) and decision making.

A. Experimental Setup

We apply the proposed method to the German Traffic Sign Recognition Benchmark (GTSRB). A vehicle uses an RGB camera to recognize the traffic signs that are present in its surroundings. The dataset consists of 43 classes of signs and provides videos of 30 frames as well as individual images that are not part of sequences. The data are collected in various light conditions and include different artifacts like motion blur and obstructions by other objects. The image resolution depends on how far the sign is from the vehicle as shown in Figure 1. Since the input size is variable, we convert all inputs to size 30x30x3. 10% of the available sequences is randomly sampled to form the sequences used for testing. From the remaining sequences, 20% is used to compute the ECDFs an the remaining sequences form the training set. The training set is split into the proper training set and the calibration set with a ratio of 5:1. 90% of the individual images that are not part of sequences augment the proper training set and the calibration set with the same ratio as above and the remaining 10% forms another test set so that we can use it as unity test for the baseline ICP.

The siamese network is formed using two identical convolutional DNNs with shared parameters. The architecture we chose to use is the one described in [1] for a similar application. A dense layer of 256 units is used to generate the embedding representation of the inputs. All the experiments run in a desktop computer equipped with and Intel(R) Core(TM) i9-9900K CPU and 32 GB RAM and a Geforce RTX 2080 GPU with 8 GB memory.

B. Siamese Network Evaluation

We first investigate how well the siamese network is trained looking at two separate metrics. One is the classification accuracy. The siamese network can be used for classification of inputs using a k-Nearest Neighbors classifier in the embedding space. One basic hypothesis of machine learning models is that the training and testing data sets should consist of IID samples. This is confirmed in Table II where the accuracy for the test set of IID examples is similar to the training accuracy while the testing accuracy for the set that includes sequences is lower.

Then we evaluate how well the siamese network clusters data of each class. A commonly used metric of the separation between classes is the *silhouette* [29]. For each sample, we first compute the mean distance between i and all other data points in the same cluster in the embedding space

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j) \ .$$

Then we compute the smallest mean distance from i to all the data points in any other cluster

$$b(i) = \min_{k \notin i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j) \ .$$

The silhouette value is defined as

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \ .$$

Each sample i in the embedding space is assigned a silhouette value $-1 \leq s(i) \leq 1$ depending on how close it is to

samples belonging to the same class and how far it is to samples belonging to different classes. The closer $s(i)$ is to 1, the closer the sample is to samples of the same class and further from samples belonging to other classes. To compare the representations learned in the different sets of data, we compute the mean silhouette value.

Table II: Siamese accuracy evaluation

	Accuracy %	Silhouette
Training	0.962	0.48
Validation	0.95	0.45
Test IID	0.955	0.44
Test Sequences	0.923	0.51

C. Validity

ICP is proven to be valid when the input data are IID and exchangeable regardless the choice of the significance level and NCM. The first problem we work on is to recover the validity in cases where the input data are dependent. We examine the property, $P(y_{l+1} \notin \Gamma^{\epsilon}) < \epsilon$ in the baseline ICP using the test set containing IID and the test set containing sequences. For this, we plot the performance and calibration curves shown in Figure 5. For different values of the significance level, the calibration curve show the percentage of test data with their ground truth class not contained in their prediction set, while the performance curve shows the percentage of test data that lead to prediction sets of more than one class. ICP is valid in the case of IID data but it under-estimates the true error-rate when data are sequential.

We evaluate the validity of ICP using the Expected Calibration Error (ECE). A well-calibrated ICP computes prediction sets with significance levels that are representative of the true error-rate. Formally a model is well-calibrated when

$$\mathbb{P}\left(y_{l+1} \in \Gamma^{\epsilon} | 1 - \epsilon = p\right) = p, \qquad \forall p \in [0, 1] \tag{23}$$

where p is the actual prediction accuracy. However, ϵ is a continuous random variable so the probability in (23) cannot be approximated using finitely many samples. According to (23) a measure of miscalibration can be expressed as $\mathbb{E}_{\epsilon}\left[\left|\mathbb{P}\left(y_{l+1} \in \Gamma^{\epsilon} | 1 - \epsilon = p\right) - p\right|\right]$. The *Expected Calibration Error* (ECE) [22] computes an approximation of this expected value across samples of the significance level:

$$\text{ECE} = \frac{1}{M} \sum_{i=1}^{M} \left|\text{acc}(\epsilon_i) + \epsilon_i - 1\right| \tag{24}$$

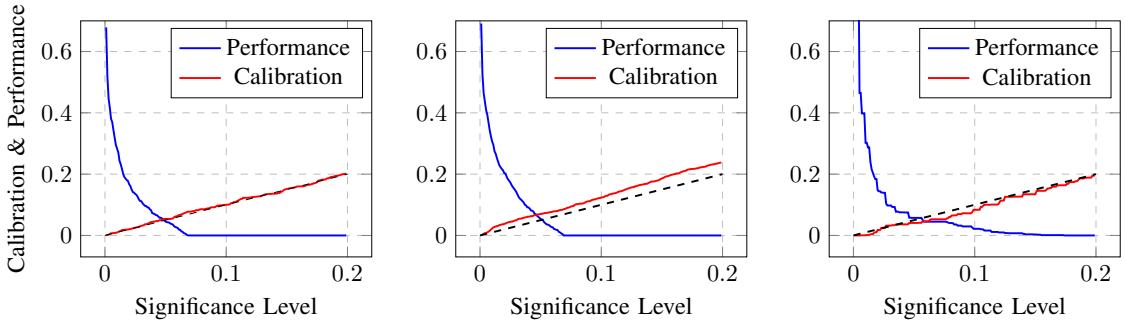
Assuming n test examples, $\text{acc}(\epsilon_i) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}(y_i \in \Gamma_i^{\epsilon})$ and ϵ_i the significance level samples. In this evaluation $\epsilon_i = \frac{i}{1000} : i = 1, \ldots, 200$, as $\epsilon > 0.2$ would not have any practical use in most CPS applications. Table III shows the calibration results of ICP when we combine multiple sequential inputs in a sliding windows of different size. For comparison when the baseline ICP is used, ECE is 0.042. Combining p-values using the ECDF based methods consistently improve the calibration over the baseline approach and have among the lowest ECE of all methods we tried. The same is observed for the merging functions that confirms the literature remarks about their validity under arbitrary dependence regardless their simplicity. Larger sliding window sizes affect the combination functions in different ways and do not guarantee better calibration. For example a low p-value that corresponds to a correct class will remain in the history for a longer time and depending on the combination function can significantly lower the aggregate p-value. Combining p-values using the quantile combination approaches, with the exception of the order statistics function min, produces prediction sets with large calibration error confirming their inability to deal with dependence between the performed statistical tests. Combining multiple p-values by using only their minimum value and transforming it into a p-value using the incomplete beta function seems to not be affected by the dependence structure of the inputs. However, when using the calibration sequences to capture these dependencies and learn the ECDF instead of using the incomplete beta function, further improves the calibration in sliding window sizes greater than six.

D. Assurance Evaluator

The assurance evaluator identifies when a prediction is trustworthy. ICP computes the credibility and confidence from the p-values of all classes [Eqs. (1), (2)] and the assurance evaluator combines them to minimize the risk for any given coverage. We compare the decision performance of the baseline ICP and ICP based on combining p-values from multiple inputs. We also investigate how the sliding window size affects the decision quality. This comparison is based on the AURC which evaluates the average risk for different coverage values. In this part of the evaluation we combine p-values using the ECDF-based approaches as they showed stability against dependence between subsequent inputs. Table IV shows the AURC value for all the ECDF methods and for different sliding windows. For comparison the computed AURC for the baseline ICP is 0.011. All four alternatives show that when predictions are based on sliding windows of more than one input, the average risk is always lower than in the case of predictions based on a single input at a time. Moreover, the size of the sliding window also affects the risk. Our evaluation results show that predictions based on larger sliding windows have lower average risk. Figure 6 shows the RC curves based on the four ECDF methods with sliding window size 9 compared with the RC curve produced with the baseline ICP.

VIII. Conclusion

CPS use machine learning components for dynamic tasks that are hard to model such as perception of the environment. These components make predictions with a non-zero error-rate which makes their use in safety critical systems challenging. We designed a selective classifier that evaluates the trustworthiness of each prediction based on credibility and confidence values computed by ICP. In order to make



(a) (b) (c)

Figure 5: (a) Baseline ICP on IID data, (b) Baseline ICP on sequential data, (c) Combination of p-values based on the min ECDF

Table III: Evaluation ECE Comparison

Method		Sliding Window Size							
		2	3	4	5	6	7	8	9
Merging	Arith Avg	0.018	0.009	0.006	0.006	0.007	0.009	0.010	0.011
	Geom Avg	0.038	0.035	0.032	0.029	0.027	0.025	0.023	0.022
	Min	0.090	0.122	0.147	0.166	0.182	0.195	0.206	0.216
	Max	0.014	0.033	0.043	0.050	0.055	0.059	0.062	0.065
CDF	Fisher	0.086	0.120	0.144	0.162	0.176	0.188	0.198	0.208
	Stouffer	0.126	0.180	0.214	0.235	0.252	0.264	0.275	0.285
	Stouffer W	0.111	0.157	0.188	0.208	0.223	0.235	0.244	0.252
	Min	0.018	0.012	0.007	0.008	0.009	0.011	0.014	0.016
	Cauchi	0.052	0.059	0.063	0.065	0.067	0.068	0.070	0.071
ECDF	Sum	0.026	0.029	0.027	0.026	0.025	0.024	0.023	0.023
	Product	0.023	0.025	0.025	0.025	0.025	0.025	0.025	0.024
	Min	0.018	0.014	0.012	0.010	0.009	0.008	0.009	0.010
	Max	0.026	0.030	0.029	0.027	0.024	0.021	0.018	0.015

Table IV: AURC Results

Sliding Window Size	ECDF			
	Sum	Product	Min	Max
2	0.007	0.007	0.007	0.007
3	0.005	0.005	0.005	0.006
4	0.004	0.004	0.004	0.005
5	0.004	0.004	0.004	0.004
6	0.003	0.003	0.004	0.004
7	0.003	0.003	0.004	0.004
8	0.003	0.003	0.003	0.004
9	0.003	0.003	0.003	0.003

efficient use of ICP we used a siamese network to map high-dimensional inputs to appropriate embedding representations. To recover the validity of ICP when subsequent inputs are time-correlated we combined the computed p-values using different multiple hypothesis testing methods. The experimental results using the GTSRB dataset, first demonstrate that taking into account one input at a time lead to over-confident classifiers. When p-values from more than one input data are combined using either merging functions or quantile functions based on ECDFs, we can recover the validity of the prediction sets. The approach is optimized to minimize the risk given a data coverage. The evaluation results showed that the use of more than one subsequent inputs is beneficial and larger sliding window sizes lead to lower risk. The use of more than one subsequent inputs is also beneficial for computing the credibility and confidence needed for the selective classifer. Classifications based on multiple inputs experience less risk for the same coverage compared to classifications based on single inputs. Comparison between different window sizes show that larger sliding windows lead to lower risk.

ACKNOWLEDGMENT

The material presented in this paper is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) through contract number FA8750-18-C-0089. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA.

REFERENCES

[1] K. Aslansefat, S. Kabir, A. Abdullatif, V. Vasudevan, and Y. Papadopoulos. Toward improving confidence in autonomous vehicle software: A study on traffic sign recognition systems. *Computer*, 54(8):66–76, 2021.

[2] V. Balasubramanian, S.-S. Ho, and V. Vovk. *Conformal Prediction for Reliable Machine Learning: Theory, Adaptations and Applications*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition, 2014.

[3] S. Bhattacharyya. Confidence in predictions from random tree ensembles. *Knowledge and Information Systems*, 35(2):391–410, May 2013.

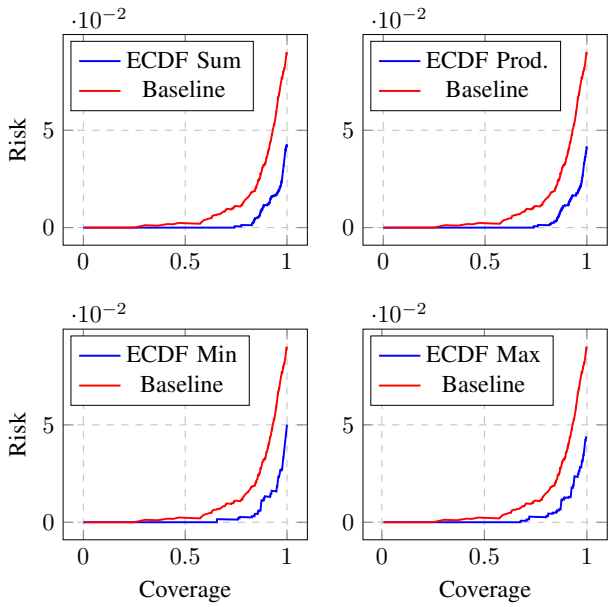


Figure 6: Risk-Coverage Curves

[4] D. Boursinos and X. Koutsoukos. Assurance monitoring of cyber-physical systems with machine learning components. In *Tools and Methods of Competitive Engineering*, pages 27–38, 2020.

[5] D. Boursinos and X. Koutsoukos. Improving prediction confidence in learning-enabled autonomous systems. In F. Darema, E. Blasch, S. Ravela, and A. Aved, editors, *Dynamic Data Driven Applications Systems*, pages 217–224, Cham, 2020. Springer International Publishing.

[6] D. Boursinos and X. Koutsoukos. Trusted confidence bounds for learning enabled cyber-physical systems. In *2020 IEEE Security and Privacy Workshops (SPW)*, pages 228–233, 2020.

[7] D. Devetyarov and I. Nouretdinov. Prediction with confidence based on a random forest classifier. In H. Papadopoulos, A. S. Andreou, and M. Bramer, editors, *Artificial Intelligence Applications and Innovations*, pages 37–44, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[8] R. El-Yaniv et al. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(5), 2010.

[9] R. A. Fisher et al. 224a: Answer to question 14 on combining independent tests of significance. 1948.

[10] Y. Geifman and R. El-Yaniv. Selective classification for deep neural networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4885–4894, Red Hook, NY, USA, 2017. Curran Associates Inc.

[11] Y. Geifman, G. Uziel, and R. El-Yaniv. Bias-reduced uncertainty estimation for deep neural classifiers. In *7th International Conference on Learning Representations, ICLR, New Orleans, LA, USA, May 6-9, 2019*, 2019.

[12] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, pages 1321–1330. JMLR.org, 2017.

[13] U. Johansson, H. Boström, and T. Löfström. Conformal prediction using decision trees. In *2013 IEEE 13th international conference on data mining*, pages 330–339. IEEE, 2013.

[14] U. Johansson, H. Linusson, T. Löfström, and H. Boström. Model-agnostic nonconformity functions for conformal classification. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2072–2079, 5 2017.

[15] M. Kääriäinen and J. Langford. A comparison of tight generalization error bounds. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 409–416. Association for Computing Machinery, 2005.

[16] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015.

[17] A. Kumar, P. S. Liang, and T. Ma. Verified uncertainty calibration. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32, pages 3792–3803. Curran Associates, Inc., 2019.

[18] T. Lipták. On the combination of independent tests. *Magyar Tud Akad Mat Kutato Int Kozl*, 3:171–197, 1958.

[19] Y. Liu and J. Xie. Cauchy combination test: A powerful test with analytic p-value calculation under arbitrary dependency structures. *Journal of the American Statistical Association*, 115(529):393–402, 2020. PMID: 33012899.

[20] L. Makili, J. Vega, S. Dormido-Canto, I. Pastor, and A. Murari. Computationally efficient svm multi-class image recognition with confidence measures. *Fusion Engineering and Design*, 86(6):1213 – 1216, 2011. Proceedings of the 26th Symposium of Fusion Technology (SOFT-26).

[21] I. Melekhov, J. Kannala, and E. Rahtu. Siamese network features for image matching. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 378–383. IEEE, 2016.

[22] M. P. Naeini, G. Cooper, and M. Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[23] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, and J. Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*, pages 13991–14002, 2019.

[24] H. Papadopoulos. *Inductive conformal prediction: Theory and application to neural networks*. INTECH Open Access Publisher Rijeka, 2008.

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

[26] N. Papernot and P. McDaniel. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *arXiv preprint arXiv:1803.04765*, 2018.

[27] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pages 61–74. MIT Press, 1999.

[28] S. M. Ross. *Introduction to Probability Models*. Academic Press, 2014.

[29] P. J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53 – 65, 1987.

[30] B. Rüger. Das maximale signifikanzniveau des tests: "lehneho ab, wennk untern gegebenen tests zur ablehnung führen". *Metrika*, 25:171–178, 1978.

[31] L. Rüschendorf. Random variables with maximum sums. *Advances in Applied Probability*, 14(3):623–632, 1982.

[32] G. Shafer and V. Vovk. A tutorial on conformal prediction. *J. Mach. Learn. Res.*, 9:371–421, June 2008.

[33] G. Shafer and V. Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

[34] S. A. Stouffer, E. A. Suchman, L. C. Devinney, S. A. Star, and R. M. Williams Jr. *The American soldier: Adjustment during army life. (Studies in social psychology in World War II), Vol. 1*. Princeton Univ. Press, 1949.

[35] P. Toccaceli. Conformal predictor combination using Neyman–Pearson Lemma. In *Conformal and Probabilistic Prediction and Applications*, pages 66–88. PMLR, 2019. ISSN: 2640-3498.

[36] P. Toccaceli and A. Gammerman. Combination of conformal predictors for classification. In A. Gammerman, V. Vovk, Z. Luo, and H. Papadopoulos, editors, *Proceedings of the Sixth Workshop on Conformal and Probabilistic Prediction and Applications*, volume 60 of *Proceedings of Machine Learning Research*, pages 39–61. PMLR, 13–16 Jun 2017.

[37] V. Vovk, A. Gammerman, and G. Shafer. *Algorithmic Learning in a Random World*. Springer-Verlag, Berlin, Heidelberg, 2005.

[38] V. Vovk and R. Wang. Combining p-values via averaging. *Biometrika*, 107(4):791–808, 2020.

[39] Y. Wiener and R. El-Yaniv. Agnostic selective classification. *Advances in neural information processing systems*, 24:1665–1673, 2011.

[40] J. Zhang and Y. Yang. Probabilistic score estimation with piecewise logistic regression. In *Proceedings of the twenty-first international conference on Machine learning*, page 115, 2004.