Proceedings of Machine Learning Research 230:1–18, 2024 Conformal and Probabilistic Prediction with Applications

Inductive Venn-Abers Predictive Distributions: New Applications & Evaluation

Ilia Nouretdinov I.R.Nouretdinov@rhul.ac.uk
Centre for Reliable Machine Learning, Royal Holloway University of London, Egham, Surrey, UK
James Gammerman jgammerman@gmail.com
Centre for Reliable Machine Learning, Royal Holloway University of London, Egham, Surrey, UK

Editor: Simone Vantini, Matteo Fontana, Aldo Solari, Henrik Boström and Lars Carlsson

Abstract

Venn-Abers predictors offer a distribution-free probabilistic framework that generates calibrated predictions from the outputs of scoring classifiers, relying on minimal assumptions about the data distribution. This paper explores the extension of this framework from classification to regression, producing predictive distributions. We show how to evaluate the efficacy of the framework by comparing various metrics that assess the accuracy and informativeness of the predictions. We also show that the framework can be used for real-time prediction, using datasets from predictive maintenance and energy consumption forecasting.

Keywords: VENN-Abers prediction, predictive distributions, regression.

1. Introduction

This paper explores machine learning methods that produce reliable and informative outputs under minimal assumptions. Our focus is on combining regression with probabilistic prediction. This is achieved using the Venn Prediction framework, which ensures that predicted probabilities are calibrated, providing statistical confidence in the results.

Specifically, we re-visit the Inductive Venn-Abers Predictive Distribution (IVAPD) framework for regression problems, building on prior work by (Nouretdinov et al., 2018). That study extended the Venn Prediction approach to regression by applying a dynamic threshold method, effectively converting regression into a classification problem. This approach poses challenges in maintaining consistency across thresholds. However, the inductive approach to learning solves these issues, as shown in (Lambrou et al., 2015), confirming the reliability of IVAPD for regression tasks.

Evaluating IVAPD is not a straightforward task due to the nature of predictive distributions. We distinguish between scoring rules—assessing the alignment between the predicted distributions and actual outcomes—and sharpness metrics, which measure the informativeness of the distribution as it is. We propose that, in the absence of ground truth labels, the

interval width or another measure of informativeness of the distribution can be used as a proxy for the commonly used Continuous Ranked Probability Score (CRPS).

The primary contributions of this paper are as follows:

- The application of the IVAPD approach to real-world predictive maintenance and energy consumption forecasting tasks, demonstrating the framework's practical utility, and analysis of the outputs.
- An extension of the algorithm for online (real-time) learning settings, in contrast to the batch learning setting in (Nouretdinov et al., 2018).
- An examination of evaluation metrics for the predictive distributions produced by IVAPD.

2. Background

In this section we provide a full theoretical background for IVAPD, in a more comprehensive manner than that given in (Nouretdinov et al., 2018). We start with an introduction to Venn and Venn-Abers prediction.

2.1. Venn Prediction

Venn Prediction is a statistical framework developed within the broader field of conformal prediction which allows us to generate reliable probabilistic predictions. The only assumption is that data points are independently and identically distributed (i.i.d.). If this assumption is met, then the Venn Predictors come with *calibration* guarantees automatically. These guarantees ensure that the probabilities predicted by the system align closely with the actual frequencies of outcomes in the long term.

Originally, Venn Prediction was primarily applied to binary and multi-class classification problems. In these settings, the framework uses a method known as a Venn Taxonomy to group similar instances together based on certain characteristics, allowing for the calculation of empirical probabilities for the classes.

Consider a dataset of labeled instances $z_1, \ldots, z_n$, where each instance z_i is a pair (x_i, y_i) with a feature vector x_i from a vector scape $\mathbb{R}^d$ and a label $y_i \in \{0, 1\}$ for binary classification tasks. The idea is to assign a test instance x_{n+1} to an equivalence class that groups it with training instances of similar characteristics. This equivalence is formally defined by a Venn Taxonomy A that applies an equivalence relation $\sim$ consistent across data permutations:

$$(i \sim j | z_1, \ldots, z_n) \implies (\pi(i) \sim \pi(j) | z_{\pi(1)}, \ldots, z_{\pi(n)}),$$

where π denotes a permutation of indices. As with the non-conformity measure in conformal prediction, this grouping can be used to integrate a Venn Predictor with an underlying machine learning algorithm.

The taxonomy enables the Venn Predictor to compute empirical probabilities for each possible class label of the test instance. The probability for the test instance x_{n+1} that it belongs to a given class y is calculated as follows:

$$P_y = \frac{|\{i \in A(n+1|z_1, \ldots, z_n, (x_{n+1}, y)) : y_i = y\}|}{|A(n+1|z_1, \ldots, z_n, (x_{n+1}, y))|}, \quad y \in \{0, 1\}.$$

In this formula, P_y represents the probability for a given class y. This parameter can also refer to the upper and lower bounds of the predicted probability, where the difference between them reflects the calibration cost of the Venn Predictor. Normally, the difference between the lower and upper bounds is not significant unless the model is over-fitted or the training data is insufficient.

Venn Predictors ensure calibration, meaning that the predicted probabilities should align with the actual frequency of each label in the long run, under the assumption of independent and identically distributed (i.i.d.) data. That is,

$$\mathbb{P}(Y = y|P_y) = P_y \quad \text{almost surely}$$

2.2. Venn-Abers Predictors

2.2.1. Scoring and Calibration through Isotonic Regression

As previously mentioned, the taxonomy in the Venn framework can be integrated with an underlying machine learning classifier. But the framework does not prescribe a particular way for constructing this taxonomy from a classifier.

The Venn-Abers variant of Venn prediction provides such a method, while also being more computationally efficient. First introduced in (Vovk and Petej, 2014), it relies on a scoring classifier that assigns a score to each instance, indicating the likelihood of the label y being '1'. A higher score suggests a greater probability of the positive class. To convert these raw scores into calibrated probabilities, isotonic regression is used. This non-decreasing function minimizes the mean squared error between predicted and observed outcomes:

$$\sum_{i=1}^{n} (g(s(x_i)) - y_i)^2 \rightarrow \min,$$

where $g(s)$ maps scores to calibrated probabilities. The isotonic constraint ensures the calibration functions are monotonic, allowing for probability estimates that are consistent with the ordering of the scores output by the classifier.

The probability for each instance x_i is defined as:

$$p_i = \begin{cases} g(s(x_i)), & \text{if } y_i = 1, \\ 1 - g(s(x_i)), & \text{if } y_i = 0, \end{cases}$$

This calibration forms the foundation for Venn-Abers Predictors, ensuring reliable and consistent predictions. The product of these probabilities across all instances, $\prod_{i=1}^{n} p_i$, is maximized to achieve an optimal fit to the observed data.

2.2.2. Taxonomy Construction

After isotonic regression, the calibrated scores are used to construct the taxonomy, which is a partitioning of the score range into equivalence classes. Each group in the taxonomy corresponds to a segment in the isotonic regression graph. Instances within the same group are treated similarly, as their scores fall within the same range of calibration.

This taxonomy structure allows for more efficient predictions, as it uses the calibrated probabilities to create a system of equivalence classes.

2.2.3. Output of Predictive Probabilities

Following calibration and taxonomy construction, Venn-Abers Predictors yield a range of predictive probabilities for a new instance x. This range is represented as:

$$(p_0, p_1) = (g_0(s_0(x)), g_1(s_1(x))),$$

Where g_0 and g_1 are the calibrated probability functions derived from isotonic regression for two possible labels 0 and 1 of the new example, as is required in Venn machines.

The lower bound, p_0, represents the minimum probability that the instance's label is 1, while the upper bound, p_1, represents the maximum probability. This range reflects the uncertainty in the prediction, with a smaller gap indicating greater confidence and a larger gap suggesting more variability or uncertainty in the underlying data.

2.3. Inductive Learning

So far we have limited our discussion to transductive Venn prediction. The key difference between the inductive and transductive modes is similar to that for conformal predictors.

Transductive Venn Predictors generate predictions by re-calibrating the model for each test instance. This process requires retraining the model for every new example, considering all potential label assignments, which is very computationally intensive and makes it impractical for large datasets or real-time applications.

Inductive Venn Predictors, on the other hand, streamline the process by training the model once on a *proper training set*, then applying it to a separate *calibration set* to generate the calibrated probabilities. This one-time training approach significantly reduces computational load, making it better suited for practical scenarios where predictions must be made quickly. Despite the simplification, inductive methods retain accuracy and consistency through statistical calibration techniques similar to those used in transductive approaches.

2.4. Inductive Venn-Abers Predictors

Inductive Venn-Abers Predictors, first introduced in (Petej and Vovk, 2019), refine the Venn-Abers framework for operational efficiency in large-scale or real-time predictive settings.

Central to this adaptation is the use of isotonic regression, previously introduced for calibrating probabilities. Specifically, the equation used in this setting:

$$\sum_{i=1}^{r}(g(s(x_{-i}, T_P)) - y_{-i})^2 \tag{1}$$

differs from the general Venn-Abers setup, where isotonic regression involves the entire dataset:

$$\sum_{i=1}^{n}(g(s(x_i)) - y_i)^2$$

In the inductive adaptation, $g(s)$ is computed only once using the scores $s(x_{-i}, T_P)$ from the proper training set T_P and the corresponding outcomes y_{-i}.

In our approach, the model incorporates a nearest neighbours classifier to approximate the calibration function $g(s)$ for new data points whose scores were not directly observed during the training phase. This method involves identifying the training instances closest to the new data points in the feature space, and using their calibrated probabilities to estimate the probabilities for the new data points.

2.5. Application to Real-Time Predictions

By using isotonic regression on a predefined subset of the training data, Inductive Venn-Abers Predictors can deliver faster real-time (online) predictions.

Although the calibration function $g(s)$ requires recalculation as the training set grows, it only needs to be done once at each time step, ensuring that the model remains efficient.

2.6. Inductive Venn-Abers Predictive Distributions (IVAPD)

The Inductive Venn-Abers Predictive Distribution (IVAPD) uses the calibration function $g(s)$, derived from isotonic regression, to construct Cumulative Distribution Functions (CDFs) for modeling continuous outcomes. This function maps predictive scores into calibrated probabilities, which are then used to estimate the likelihood that the outcomes will not surpass specific thresholds t.

Within regression contexts, IVAPD treats each potential outcome level as a discrete threshold t. It computes the probability $P\{y \leq t\}$ for these thresholds using the calibrated scores, effectively converting the regression problem into a series of binary classification tasks.

By grouping observations based on their calibrated scores and aggregating these groups to form the CDFs, IVAPD provides a detailed and probabilistic view of the outcomes' uncertainty. This approach ensures that the predictions are not only well-calibrated but also comprehensively represent the possible scenarios, allowing for improved decision-making.

2.6.1. IVAPD algorithm

Algorithm 1 outlines the process of generating predictive distributions using the Inductive Venn-Abers methodology, designed for regression tasks.

- **Step 1: Initialisation** — Initialises the process by loading the full dataset D, consisting of feature vectors x_i and target values y_i, along with the specifications for the underlying scoring predictor s that utilises nearest neighbours. This step also includes the setting of algorithmic parameters like the number of neighbours k, details of normalisation and feature selection if applicable.
- **Step 2: Dynamic Data Splitting and Underlying Predictor Training** — For each data instance indexed by j, the dataset D is dynamically divided into a training set T_P and a calibration set T_C, based on a predefined ratio $\frac{h}{r}$ that optimises the balance between these sets.[1] The training set comprises the first r instances, while the calibration set contains the subsequent h instances, leading up to the instance j. This step also includes an optional feature selection based on T_P to refine the

1. In this work, we set the ratio to 1.

Algorithm 1 IVAPD - Online Version

Step 1: Initialization

INPUT: Full dataset $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ where $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$.
INPUT: Underlying predictor s for scoring using nearest neighbours.
INPUT: Parameters - number of neighbours k
INPUT: ratio R of the proper training : calibration set desired split

for $j := 1$ to n **do**

Step 2: Dynamic Data Splitting and Underlying Predictor Training

Define r and h such that $j = h + r + 1$, where $h : r$ is chosen according to the desired proper training set: calibration set split ratio R.
Split D into proper training set $T_P = \{(x_1, y_1), \ldots, (x_r, y_r)\}$, calibration set $T_C = \{(x_{r+1}, y_{r+1}), \ldots, (x_h, y_h)\}$, and testing example x_j.
Optionally perform feature selection based on T_P.
Calculate scores $s_i := s(x_i, T_P \setminus \{(x_i, y_i)\})$ for instances in T_P, excluding each instance itself.

Step 3: Calibration using Isotonic Regression

Apply isotonic regression on T_P to find calibrated scores:
$\sum_{i=1}^{r}(g_i - y_i)^2 \to \min$ subject to $(s_i \leq s_j) \Rightarrow (g_i \leq g_j)$ for all i, j in T_P.

Step 4: Scoring and Calibration for Test Example

for $i := r + 1$ to $h + 1$ **do**
Calculate score $s_i := s(x_i, T_P)$ for each instance in T_C and x_j.
Find s_k closest to s_i where $k \leq r$, and assign $g_i := g_k$.
end for

Step 5: Construction of Predictive Distribution

Identify set $A := \{i = r + 1, \ldots, h : g_i = g_{h+1}\}$.
Construct predictive set $\hat{Y} := \{y_i : i \in A\}$.
Calculate probabilities for the testing instance:
$\hat{P}_0\{y_j \leq t\} := \frac{|\{\hat{y} \in \hat{Y} : \hat{y} \leq t\}|}{|A|+1}$
$\hat{P}_1\{y_j \leq t\} := \frac{|\{\hat{y} \in \hat{Y} : \hat{y} \leq t\}|+1}{|A|+1}$

end for

input feature set. Scores for the training instances are computed by the predictor s, excluding the instance itself to avoid bias.

- **Step 3: Calibration using Isotonic Regression** — Applies isotonic regression to the scores from T_P to calibrate these into more accurate probabilities. This calibration minimizes the squared error between the calibrated scores and the actual outcomes, constrained by the order of scores to maintain isotonicity.
- **Step 4: Scoring and Calibration for Test Example** Extends the scoring process to the calibration set and the current test instance using the trained predictor s. Each score for these instances is then calibrated by matching it with the closest score from T_P, assigning the corresponding calibrated value.
- **Step 5: Construction of Predictive Distribution** — Forms a taxonomy by identifying all instances from T_C that share the same calibrated score as the test instance, forming the set A. This subset is used to construct a predictive set $\hat{Y}$, which represents the potential outcomes for the test instance. The probabilities $\hat{P}_0$ and $\hat{P}_1$ are then computed, reflecting the likelihoods of the test outcome not exceeding various thresholds, thereby converting the regression task into a probabilistic prediction problem.
- In the on-line mode, **Steps 2-5** are repeated for each example.

3. Datasets and implementation details

In this study we apply IVAPD to real data. The algorithm is applicable to any kind of regression task, but here we focus on problems related to predictive maintenance and energy consumption.

3.1. Datasets

3.1.1. Maintenance of Naval Propulsion Plants (NPP)

This application involves the Condition-Based Maintenance of Naval Propulsion Plants Dataset[2].

We use 16 features relating to steady-state measurements of a physical asset's Gas Turbine (GT). Labels to be predicted (for maintenance purposes) include the (1) Compressor degradation coefficient and (2) Turbine degradation coefficient. In this paper we refer to these two tasks as NPP (1) and NPP (2).

For analysis, we randomly shuffle and select the first 2,000 samples from the dataset. This ensures that the i.i.d. assumption remains true. The degradation coefficients, initially

2. Available at: http://archive.ics.uci.edu/ml/datasets/condition+based+maintenance+of+naval+propulsion+plants

dimensionless and ranging from $95 - 100\%$, are transformed to complement to 1. E.g. a 98% condition would be represented as 0.02 (2% degradation).

3.1.2. Energy Consumption Prediction (ECP)

Another regression task comes from the UCI Individual Household Electric Power Consumption Dataset[3].

For this task, we focus on the first three features:

1. Date in format dd/mm/yyyy
2. Time in format hh:mm:ss
3. Global active power: minute-averaged active power (in kilowatts)

The rest of the features are excluded from the analysis.

The regression task aims to predict the power consumption at a specific evening time (18:00) based on the consumption vector from the first half of the day (00:00-12:00). We assume that historical data from previous days is available when making predictions for a given date.

Unlike the previous dataset, for this dataset we check applicability in real time mode rather than shuffling. Each preceding day is transformed into a training instance with the morning feature vector as the input and the consumption at 18:00 as the label, creating a progressively growing training set.

3.2. Implementation details

3.2.1. Dynamic splitting of the data

To implement Algorithm 1, the dataset must be divided into proper training and calibration sets. For simplicity, we make them approximately equal.

For the instance j being tested, the proper training set includes instances from $1, 2, \ldots, \text{round}\left(\frac{j-1}{2}\right)$, while the calibration set consists of instances from $\text{round}\left(\frac{j-1}{2}\right) + 1, \ldots, j-1$. This configuration enables IVAPD to continuously integrate new data to refine future forecasts.

3. Available at: http://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption

3.2.2. Underlying Method

The underlying predictor $P : (x, T) \to s$, a critical element of Algorithm 1, mapping a feature vector x to a score s using a set T of labeled feature vectors (x, y).

For this study, the k Nearest Neighbours (k-NN) algorithm was selected for its effectiveness with nonlinear problems, as opposed to Linear Regression, which does not perform as well in this setting. However, the k-NN method's sensitivity to noisy features means that a feature selection step is needed to improve performance. This process involves optimising two parameters: the number of neighbours k and the number of selected features f, determined through experimental comparison.

Feature selection relies on the proper training set T. It starts by dividing T into two subsets based on the median label value y_m: $T_1 = \{(x, y) \in T : y < y_m\}$ and $T_2 = \{(x, y) \in T : y > y_m\}$. For each feature j, the Mann-Whitney-Wilcoxon test is used to compare distributions in T_1 and T_2, resulting in a p-value p_j for each feature. The f features with the lowest p_j values are then selected for use in the k-NN model.

The k-NN regression calculates the score $s(x)$ for a new example x by identifying its k nearest neighbours in the selected f-dimensional feature space, using the Euclidean distance as the metric, and averaging the labels y_i of these neighbours.

Following score generation, isotonic regression is used to calibrate these k-NN derived scores into calibrated probabilities, as described in section 2. Figure 1 illustrates the result of this, showing the non-linear transformation from scores to calibrated probabilities.

4. Evaluating IVAPD

Algorithm 1 produces calibrated predictions regardless of the underlying method used. But how do we evaluate the predictive distributions that it yields?

Table 1 lists various evaluation criteria. According to the literature (Gneiting and Katzfuss, 2014), these criteria can be classified into two groups: *scoring rules*, which measure the degree of disagreement between the predictive distribution and observed values, and *sharpness measures*, which evaluate the precision of the probabilistic distribution itself. The key distinction between these groups is that scoring rules require the true labels, while sharpness measures do not.

A similar observation has been made in the conformal prediction framework (Vovk et al., 2016): examples of sharpness measures that do not rely on ground truth include the average size of the prediction set or the average individual confidence.

Note that the effectiveness of sharpness measures relies on the proven calibration properties, as is the case with Venn machines. Because of this, these evaluation criteria should not be

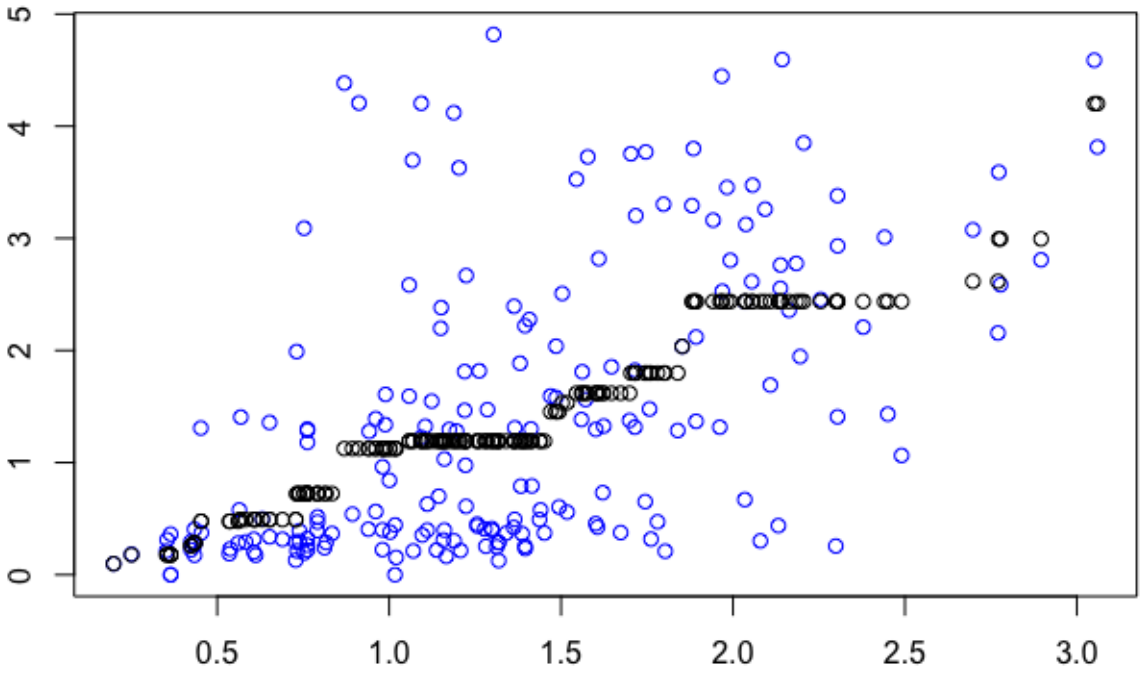

Figure 1: Example of isotonic regression solution (for the Energy Consumption data set). Horizontal axis: the predicted value. Vertical axis: the true label. The plot shows the transformation (blue) $s \to g$ (black), forcing the relationship to be monotonic.

compared to those of non-calibrated methods, where achieving sharpness might be simpler but the results may not be reliable.

- **Scoring Rules**: These rules evaluate the accuracy of the predictive distributions and require the true labels for comparison. The primary measure, the Continuous Ranked Probability Score (CRPS), calculates the integral of the squared differences between the predicted and observed cumulative distribution functions (CDFs).[4]
- **Sharpness Measures** do not require labels and assess the precision of the probabilistic predictions. These include:
  - **Interval Width (W)**: This metric identifies the narrowest interval within which the actual label is expected to fall, with a confidence level of at least $(1 - \varepsilon)$.[5] It evaluates the informativeness and precision of the predictive distribution: A narrower interval indicates higher precision and a more informative prediction, as it suggests that the model has a high degree of certainty about the range of possible outcomes.
  - **Variance (V)**: This metric measures the average variance within the predictive distribution, reflecting the spread of potential outcomes. A lower variance

4. Here, the observed distribution means one fully concentrated on the observed label.
5. So, W-measure is actually a family of measures, with ε as a parameter.

indicates that the predictions are tightly clustered, suggesting higher precision and confidence in the predicted values. Conversely, a higher variance indicates a wider spread, which might suggest less certainty in the predictions but potentially greater coverage of actual outcomes, balancing precision with reliability.

– **Probability Distribution Spread (P)**: This metric evaluates the difference between the upper and lower cumulative distribution function estimates, $P_1(U)$ and $P_0(U)$, over a certain region U. A smaller difference indicates that the model is consistent in its predictions, suggesting confidence in the estimation. Conversely, a larger difference indicates uncertainty in the model's predictions.

These metrics allow us to assess reliability of the predictive distributions produced by IVAPD making comparisons with other frameworks possible (with the exception of P-measure which is specific to Venn-based predictors.)

Note that both the V-measure and the P-measure are related to the informativeness of a prediction. However, they are considered separately because Venn predictions come in dual form, with both lower and upper distributions. The V-measure is applicable to either of these distributions individually, or ideally, to a combined distribution, which is derived by merging them in some manner. This interpretation treats the output as a conventional distribution, making it potentially comparable to other methods that generate similar distributions. However, this comparison is outside the scope of this study.

Table 1: Evaluation Criteria for Predictive Distributions

Metric	**Label Required?**	**Interpretation**
(C) CRPS	Yes	Accuracy of predicted vs. actual distributions.
(W) Interval Width	No	Precision of prediction via minimal covering interval.
(V) Variance	No	Variance of the predictive distribution.
(P) Probability Distribution Spread	No	Difference between predictive distribution bounds.

5. Results

5.1. Visualising The Predictive Distributions

Figure 2 shows the prediction intervals produced by IVAPD on the ECP dataset. It takes the form of a box-plot and illustrates how the predictive distribution evolves over time, with the yellow area representing the full range and the green area indicating the 50%

interval (between the 25th and 75th percentiles), compared to the actual observed values (represented by black points).

The objective of online learning is to improve predictive performance over time as more training data becomes available. We can observe from the plot that, as time goes on, the distributions are shifted downwards as the algorithm adjusts to the lower label values. For the last 80 days, the 50% interval bars (in green) are also smaller than before, indicating increased confidence in the predictions.

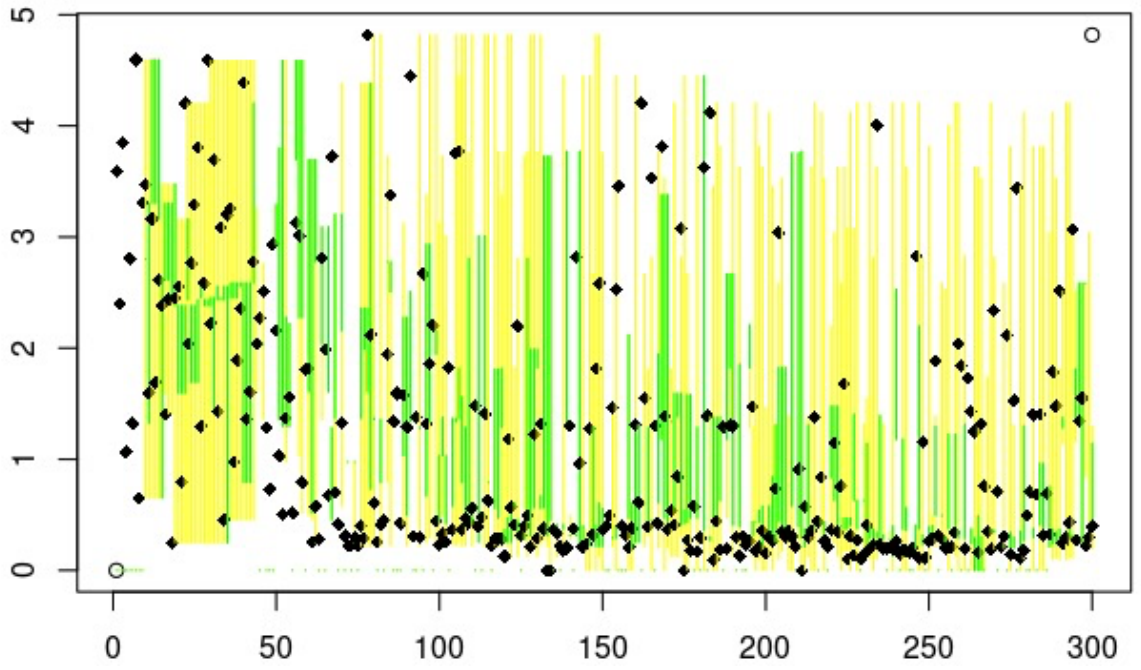

Figure 2: Real-time prediction: ECP. Horizontal axis: instance index (in time order). Vertical axis: predicted value (energy consumption at 18:00:00). Black points: observed labels. Yellow area: full range of the predictive distribution. Green area: range between the 25th and 75th percentiles.

5.2. Evaluating The Predictive Distributions

Table 2 presents the performance of the k-NN based IVAPD model, using the evaluation metrics described in Section 4. The model's predictions are structured as an ordered set $A = \{A_1, \ldots, A_T\}$, which captures both lower and upper predictive distributions. Except for the probability distribution spread (P), which compares the lower and upper bounds, the measures are applied to the distribution uniformly distributed on A, as it represents a midpoint between the lower and upper predictive distributions in the CDF.

Table 2: Evaluation of Results

parameters feat.	parameters nei.	C	V	W $\varepsilon = 0.25$	W $\varepsilon = 0.5$	W $\varepsilon = 0.75$	P
NPP(1) data set							
5	5	0.00117	0.00225	0.00424	0.00213	0.000582	0.0493
5	20	0.00130	0.00237	0.00484	0.00263	0.000809	0.0379
5	100	0.00143	0.00250	0.00577	0.00338	0.00110	0.0189
all	5	**0.000964**	**0.00207**	**0.00339**	**0.00160**	**0.000369**	0.0559
all	20	0.00124	0.00232	0.00454	0.00239	0.000744	0.0442
all	100	0.00143	0.00249	0.00579	0.00335	0.00107	**0.0172**
best param.		(all,5)	(all,5)	(all,5)	(all,5)	(all,5)	(all,100)
NPP(2) data set							
5	5	0.00183	0.00636	0.00773	0.00321	0.00116	0.108
5	20	0.00237	**0.00635**	0.00964	0.00450	0.00166	0.0880
5	100	0.00379	0.00729	0.0153	0.00841	0.00329	0.0507
all	5	**0.00179**	0.00642	**0.00757**	**0.00283**	**0.000954**	0.112
all	20	0.00239	0.00648	0.00993	0.00449	0.00158	0.0925
all	100	0.00415	0.00747	0.170	0.00989	0.00399	**0.0294**
best param.		(all,5)	(5,20)	(all,5)	(all,5)	(all,5)	(all,100)
ECP data set							
5	5	0.607	1.780	2.418	0.973	0.425	0.166
5	20	0.624	1.751	2.365	0.923	0.398	0.150
5	100	0.591	**1.657**	**2.211**	**0.872**	**0.380**	**0.123**
20	5	0.622	1.797	2.496	1.009	0.437	0.168
20	20	0.608	1.764	2.380	0.969	0.422	0.157
20	100	0.591	1.675	2.287	0.934	0.436	0.137
all	5	0.613	1.825	2.485	0.995	0.450	0.178
all	20	0.604	1.752	2.343	0.942	0.402	0.153
all	100	**0.586**	1.692	2.276	0.917	0.419	0.136
best param.		(all,100)	(5,100)	(5,100)	(5,100)	(5,100)	(5,100)

Variance (V) is computed as the mean squared deviation, including both the minimum and maximum predicted values, y_{min} and y_{max}, to avoid infinite variance. The width (W) is included for three values of ε, in order to analyse its influence on the results.

A comparison is also made between different settings of the two hyperparameters: the number of selected features (at the feature selection stage), and the number k of neighbours (in k-NN algorithm).

5.2.1. Observations & Discussion

As discussed in Section 4, the C-criterion requires access to actual labels and is not applicable without them. In contrast, V, W, and P criteria can independently assess the informativeness of the predictive distribution. Optimal parameter settings, highlighted in bold in the table, indicate that the W-criterion is most consistent with the C-criterion's conclusions. For example, in the NPP(1) dataset, the optimal parameter settings identified by the C, V, and W criteria coincide at (all,5), while P selects (all,100). This pattern of agreement between W and C, where W mirrors the choices of C, underscores its effectiveness as a metric when true labels are unavailable. In the ECP dataset, all three criteria (V, W, P) prefer the setting (5,100), which is the second-best according to C.

Another point to note is that the specific value of the parameter ε for the interval width (W-criterion) does not impact the results significantly. This observation suggests that the advantage of the variance (V-criterion) being parameterless may not offer much benefit in this context.

6. Conclusion & Future Work

In this work, we further explored the Inductive Venn-Abers Predictive Distributions (IVAPD) approach to regression, which generates rich and reliable predictions in the form of predictive distributions. We demonstrated that it can be applied in an online setting for applications in energy consumption and predictive maintenance.

Regarding evaluation, we discussed several possible performance metrics for comparing different methods within the IVAPD framework. We found that the interval width (W-criterion) is a particularly useful metric. Despite its simplicity, it provides realistic and reliable insights.

For future work, the following tasks should be prioritised.

First, we need to extend the analysis to confirm the conclusions with more datasets and underlying metrics.

Second, there is a need for comparison with other methods of probabilistic prediction such as classical Bayesian methods, Conformal Prediction Systems Vovk et al. (2017), and Inductive Venn-Abers predictions Petej and Vovk (2019).

Third, an important area of focus is the exploration of additional metrics. In this study, we primarily examined accuracy and sharpness of individual predictions. However, another important aspect is the diversity of these predictions. Do they vary significantly, or do they tend to follow a common pattern? We have presented some preliminary observations on this topic in the Appendix, but determining the most effective way to quantify diversity remains a challenge, especially given the varying interpretations of this concept.

References

T. Gneiting and M. Katzfuss. Probabilistic forecasting. *Annual Review of Statistics and Its Application*, 1 2014.

A. Lambrou, I. Nouretdinov, and H. Papadopoulos. Inductive venn prediction. *Annals of Mathematics and Artificial Intelligence*, 74:181–201, 1–2 2015.

I. Nouretdinov, D. Volkhonskiy, P. Lim, P. Toccaceli, and A. Gammerman. Inductive venn-abers predictive distribution. *Proceedings of Conformal Prediction with Applications*, 91: 1–22, 2018.

I. Petej and V. Vovk. Inductive venn-abers prediction for regression. *Technical Report*, 2019. URL http://clrc.rhul.ac.uk/documents/ReportIVAPR.pdf.

V. Vovk and I. Petej. Venn-abers predictors. *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, pages 829–838, 2014.

V. Vovk, V. Fedorova, I. Nouretdinov, and A. Gammerman. Criteria of efficiency for conformal prediction. *CoRR*, abs/1603.04416, 2016. URL http://arxiv.org/abs/1603.04416.

V. Vovk, J. Shen, V. Manokhin, and M.-G. Xie. Nonparametric predictive distributions based on conformal prediction. *Conformal and Probabilistic Prediction and Applications*, pages 82–102, 2017.

Appendix

Observations On Diversity in the CDFs

As mentioned in the Conclusion, an essential aspect of probabilistic predictions is their relative informativeness, particularly the diversity of the predictions when compared to each other. Uniform predictions across different examples, even if accurate on average, lack informativeness. Similarly, predictions that are merely shifted versions of a single distribution also indicate low informativeness.

For example, Figures 3–5 illustrate the results for three datasets—NPP(1), NPP(2), and ECP—using the optimal parameter settings as determined by the C-criterion. These figures display the cumulative distribution functions (CDFs) of the predictive distributions, which are monotonically increasing functions ranging from 0 to 1 within the distribution's support.

To demonstrate the diversity of our predictions, each plot presents several distributions jointly. Each line represents the CDF of one distribution. These distributions correspond to different examples but are generated from similarly large portions of the dataset, specifically the last 5% of the predictions, ensuring each is trained on 95-100% of the total data.

The most varied results are observed in NPP(2), where distributions differ significantly in their initial location (where the CDF begins to rise), variation (slope of the CDF), and overall shape.

Conversely, the distributions for ECP appear more uniform, resembling variations of a single pattern. While they exhibit a strong group characteristic, they still maintain some distinctiveness.

NPP(1) represents an intermediate case, showing less diversity than NPP(2) but more than ECP.

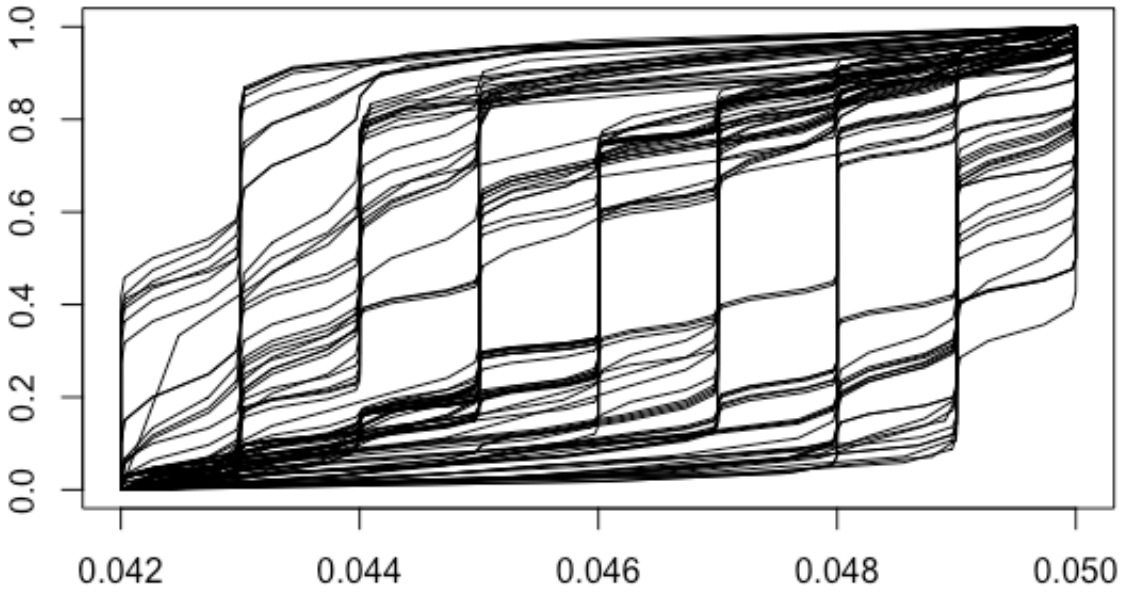

Figure 3: Diversity of CDF of the last few predictions in NPP(1). Horizontal axis: predicted value. Vertical axis: cumulative probability. Each line shows a smoothed CDF.

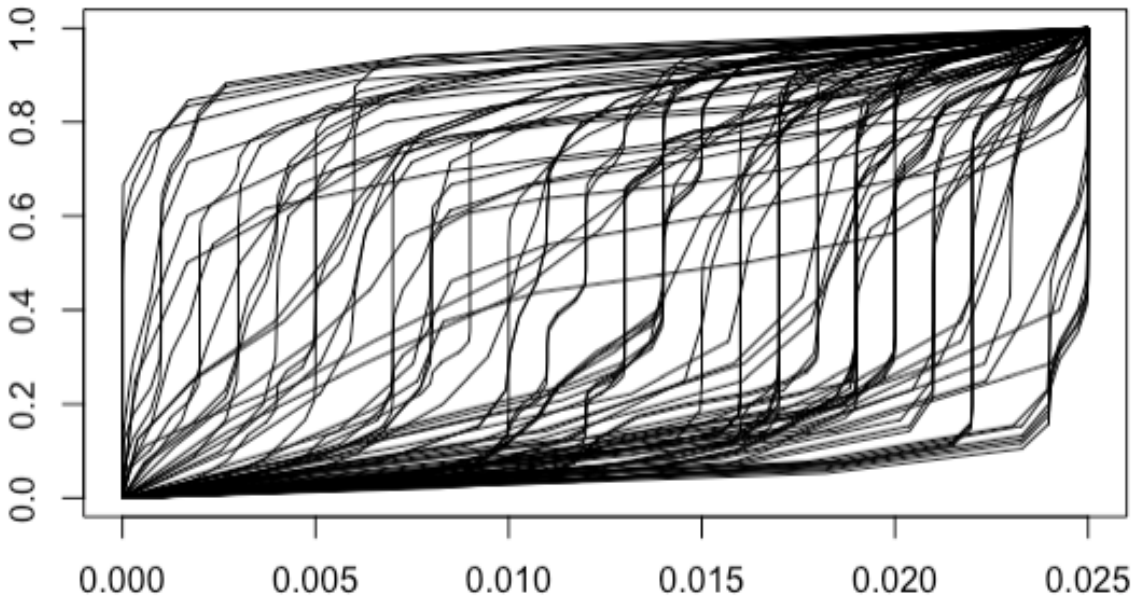

Figure 4: Diversity of CDF of the last few predictions in NPP(2). Horizontal axis: predicted value. Vertical axis: cumulative probability. Each line shows a smoothed CDF.

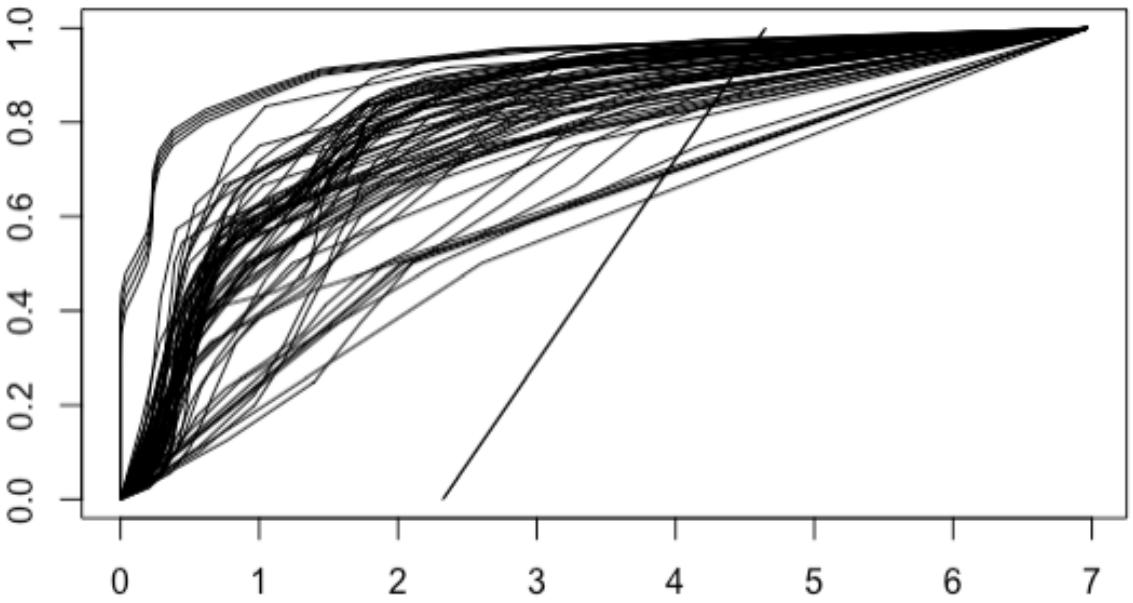

Figure 5: Diversity of CDF of the last few predictions in ECP. Horizontal axis: predicted value. Vertical axis: cumulative probability. Each line shows a smoothed CDF.