

Conformal Prediction and Venn Predictors

A Tutorial on Predicting with Confidence

Ulf Johansson¹, Henrik Linusson², Tuve Löfström¹, Henrik Boström³, Alex Gammerman⁴

July 29, 2019

¹Jönköping University, Sweden. Email: {ulf.johansson,tuwe.lofstrom}@ju.se

²University of Borås, Sweden. Email: henrik.linusson, @hb.se

³KTH, Royal Institute of Technology, Sweden. Email: henrik.bostrom@dsv.su.se

⁴Royal Holloway, University of London, United Kingdom. Email: a.gammerman@cs.rhul.ac.uk

Agenda

Purpose and goal

A motivating example

Conformal prediction at a glance

Conformal regression

Conformal classification

Validity and efficiency

Conformal classification - a critical look

Probabilistic prediction

Algorithmic confidence for FAT and XAI

Nonconformist - conformal prediction in Python

Research opportunities

Conformal classification - some details

Predicting with confidence: the best machine learning idea you never heard of

- “Wouldn’t it be nice to have some tool to tell you how uncertain your prediction is when you’re not certain of your priors?”

¹V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005

Predicting with confidence: the best machine learning idea you never heard of

- “Wouldn’t it be nice to have some tool to tell you how uncertain your prediction is when you’re not certain of your priors?”
- “Well, it turns out, humanity possesses such a tool, but you probably don’t know about it.”

¹V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005

Predicting with confidence: the best machine learning idea you never heard of

- “Wouldn’t it be nice to have some tool to tell you how uncertain your prediction is when you’re not certain of your priors?”
- “Well, it turns out, humanity possesses such a tool, but you probably don’t know about it.”
- “The name of this basket of ideas is conformal prediction¹”.

¹V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005

Predicting with confidence: the best machine learning idea you never heard of

- “Wouldn’t it be nice to have some tool to tell you how uncertain your prediction is when you’re not certain of your priors?”
- “Well, it turns out, humanity possesses such a tool, but you probably don’t know about it.”
- “The name of this basket of ideas is conformal prediction¹”.
- “Vladimir Vovk is a former Kolmogorov student, who has had all kind of cool ideas over the years. Glenn Shafer is also well known for his co-development of Dempster-Shafer theory. Alexander Gammerman is a former physicist from Leningrad, who has done quite a bit of work in the past with Bayesian belief networks.”

¹V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005

Predicting with confidence: the best machine learning idea you never heard of

- “Wouldn’t it be nice to have some tool to tell you how uncertain your prediction is when you’re not certain of your priors?”
- “Well, it turns out, humanity possesses such a tool, but you probably don’t know about it.”
- “The name of this basket of ideas is conformal prediction¹”.
- “Vladimir Vovk is a former Kolmogorov student, who has had all kind of cool ideas over the years. Glenn Shafer is also well known for his co-development of Dempster-Shafer theory. Alexander Gammerman is a former physicist from Leningrad, who has done quite a bit of work in the past with Bayesian belief networks.”
- “Conformal prediction comes from deep results in probability theory and is inspired by Kolmogorov and Martin-Lof’s ideas on algorithmic complexity theory.”

¹V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005

Predicting with confidence: the best machine learning idea you never heard of

- “Wouldn’t it be nice to have some tool to tell you how uncertain your prediction is when you’re not certain of your priors?”
- “Well, it turns out, humanity possesses such a tool, but you probably don’t know about it.”
- “The name of this basket of ideas is conformal prediction¹”.
- “Vladimir Vovk is a former Kolmogorov student, who has had all kind of cool ideas over the years. Glenn Shafer is also well known for his co-development of Dempster-Shafer theory. Alexander Gammerman is a former physicist from Leningrad, who has done quite a bit of work in the past with Bayesian belief networks.”
- “Conformal prediction comes from deep results in probability theory and is inspired by Kolmogorov and Martin-Lof’s ideas on algorithmic complexity theory.”
- “Honestly, I think this is the best bag of tricks since boosting; everyone should know about and use these ideas.”

¹V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005

Purpose and goal

Predicting with confidence

Predicting with confidence

- Conformal and Venn predictors provide guarantees for your predictions!

Predicting with confidence

- Conformal and Venn predictors provide guarantees for your predictions!
- There is absolutely no magic involved - only mathematics!

Predicting with confidence

- Conformal and Venn predictors provide guarantees for your predictions!
- There is absolutely no magic involved - only mathematics!
- Hot topic - recently picked up by both academia and industry

Predicting with confidence

- Conformal and Venn predictors provide guarantees for your predictions!
- There is absolutely no magic involved - only mathematics!
- Hot topic - recently picked up by both academia and industry
- Plenty of open questions, i.e., research opportunities

Predicting with confidence

Predicting with confidence

- I find conformal and Venn predictors to be extremely powerful, yet very straightforward to use

Predicting with confidence

- I find conformal and Venn predictors to be extremely powerful, yet very straightforward to use
- My overall ambition with this tutorial is to introduce these techniques while trying to convey their potential

Predicting with confidence

- I find conformal and Venn predictors to be extremely powerful, yet very straightforward to use
- My overall ambition with this tutorial is to introduce these techniques while trying to convey their potential
- In my opinion - conformal and Venn predictors will soon be part of the standard toolbox for a data scientist

Predicting with confidence

- I find conformal and Venn predictors to be extremely powerful, yet very straightforward to use
- My overall ambition with this tutorial is to introduce these techniques while trying to convey their potential
- In my opinion - conformal and Venn predictors will soon be part of the standard toolbox for a data scientist
- So - maybe you can use them off-the-shelf...

Predicting with confidence

- I find conformal and Venn predictors to be extremely powerful, yet very straightforward to use
- My overall ambition with this tutorial is to introduce these techniques while trying to convey their potential
- In my opinion - conformal and Venn predictors will soon be part of the standard toolbox for a data scientist
- So - maybe you can use them off-the-shelf...
- ...or even be part of the small but growing conformal society

Predicting with confidence

- I find conformal and Venn predictors to be extremely powerful, yet very straightforward to use
- My overall ambition with this tutorial is to introduce these techniques while trying to convey their potential
- In my opinion - conformal and Venn predictors will soon be part of the standard toolbox for a data scientist
- So - maybe you can use them off-the-shelf...
- ...or even be part of the small but growing conformal society
- Disclaimer: I come from machine learning not algorithmic theory...

A motivating example

Motivating Example

How good is your prediction?

You want to estimate the risk of cancer recurrence in patient x_{k+1}

To your disposal, you have:

1. A set of historical observations $(x_1, y_1), \dots, (x_k, y_k)$
 - x_i describes a patient by age, tumor size, etc
 - y_i is a measurement of cancer recurrence in patient x_i
2. Some machine learning (classification or regression) algorithm

Motivating Example

```
import pandas as pd

breast_cancer = pd.read_csv('./data/breast-cancer.csv')

# (x_1, y_1), ..., (x_k, y_k)
x_train = breast_cancer.values[:-1, :-1]
y_train = breast_cancer.values[:-1, -1]

# (x_{k+1}, y_{k+1})
x_test = breast_cancer.values[-1, :-1]
y_test = breast_cancer.values[-1, -1]
```


Motivating Example

```
from sklearn.neighbors import KNeighborsClassifier

knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(x_train, y_train)

print(knn.predict(x_test))
print(knn.predict_proba(x_test))
```

```
['no-recurrence-events']
[[ 0.8  0.2 ]]
```

Motivating Example

How good is your prediction, really?

- Your classifier says that the patient will have no recurrence events.
Is it right?
- Your probability estimator says it's 80% likely that the patient won't have a recurrence event.
How good is the estimate?
- Your regression model says the patient should have 0.4 recurrence events in the future.
How close is that to the true value?

Will you trust your model?

Motivating Example

The simple answer:

We expect past performance to indicate future performance.

Motivating Example

The simple answer:

We expect past performance to indicate future performance.

- The model is 71% accurate on the test data,
so we assume it's accurate for 71% of production data.
- The model has an AUC of 0.65 on the test data,
so we assume it has an AUC of 0.65 on production data.
- The model has an RMSE of 0.8 on the test data,
so we assume it has an RMSE of 0.8 on production data.

Motivating Example

The simple answer:

We expect past performance to indicate future performance.

- The model is 71% accurate on the test data,
so we assume it's accurate for 71% of production data.
- The model has an AUC of 0.65 on the test data,
so we assume it has an AUC of 0.65 on production data.
- The model has an RMSE of 0.8 on the test data,
so we assume it has an RMSE of 0.8 on production data.

But...

How good are these estimates? Do we have any guarantees? Specifically, what about patient x_{k+1} ? What performance should we expect from the model for this particular instance?

We can use PAC (probably approximately correct) theory.

Gives us valid error bounds for the model.

But...

- Bounds are on model-level — don't consider whether instance is “easy” or “hard”.
- Bounds tend to be large².

²I. Nourtdinov, V. Vovk, M. Vyugin, and A. Gammerman, “Pattern recognition and density estimation under the general i.i.d. assumption,” in *Computational Learning Theory*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2001, vol. 2111, pp. 337–353

We can use Bayesian learning.

Gives us calibrated error bounds on a per-instance basis.

But...

- Only if we know the prior probabilities³.

³H. Papadopoulos, V. Vovk, and A. Gammerman, “Regression conformal prediction with nearest neighbours,” *Journal of Artificial Intelligence Research*, vol. 40, no. 1, pp. 815–840, 2011

We can use Conformal Prediction.

- Individual probabilities/error bounds per instance.
- Probabilities are well-calibrated: 80% means 80%.
- We don't need to know the priors.
- We make a single assumption — exchangeability (\sim i.i.d.)
- We can apply it to any machine learning algorithm.
- It's rigorously proven and simple to implement!
- Developed by Vladimir Vovk, Alex Gammerman & Glenn Shafer.⁴

⁴V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005

Conformal prediction at a glance

Conformal prediction: intuition

Some intuition

Assume we have

- Some distribution $\mathbf{Z} : \mathbf{X} \times \mathbf{Y}$ generating examples
- Some function $f(z) \rightarrow \mathbb{R}$

Some intuition

- Apply $f(z)$ to some, say 4, examples from Z
- Call the resulting scores $\alpha_1, \alpha_2, \alpha_3, \alpha_4$.
 - For simplicity, $\alpha_1 \leq \alpha_2 \leq \alpha_3 \leq \alpha_4$

α_1

α_2

α_3

α_4

Conformal prediction: intuition

Some intuition

If we draw new examples from \mathbf{Z} , and apply $f(\mathbf{z})$ to them

- Given that all examples are exchangeable,
- we can estimate distribution of scores, relative to $\alpha_1, \dots, \alpha_4$

Conformal prediction: intuition

Some intuition

If we draw new examples from \mathbf{Z} , and apply $f(z)$ to them

- Given that all examples are exchangeable,
- we can estimate distribution of scores, relative to $\alpha_1, \dots, \alpha_4$

20% 20% 20% 20% 20%

α_1 α_2 α_3 α_4

$$P[f(z) \leq \alpha_3] = 0.6$$

$$P[f(z) \leq \alpha_4] = 0.8$$

Conformal prediction: intuition

Some intuition

Let $f(z_i) = |y_i - h(x_i)|$, i.e., the absolute error.

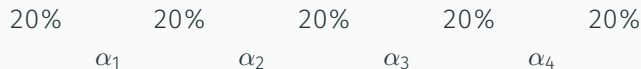
where h is a regression model trained on the domain of \mathbf{Z} .

Conformal prediction: intuition

Some intuition

Let $f(z_i) = |y_i - h(x_i)|$, i.e., the absolute error.

where h is a regression model trained on the domain of \mathbf{Z} .



Now these probabilities are about the size of the absolute errors for future instances!

$$P[|y_i - h(x_i)| \leq \alpha_3] = 0.6$$

$$P[|y_i - h(x_i)| \leq \alpha_4] = 0.8$$

Conformal prediction: intuition

Some intuition

We know (x_i, y_i) for all examples that generated $\alpha_1, \dots, \alpha_4$,
i.e., we can obtain values for $\alpha_1, \dots, \alpha_4$.

20%	20%	20%	20%	20%
0.03	0.07	0.11	0.13	

$$P[|y_i - h(x_i)| \leq 0.11] = 0.6$$

$$P[|y_i - h(x_i)| \leq 0.13] = 0.8$$

Conformal prediction: intuition

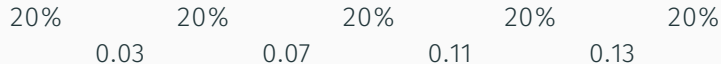
Some intuition

For a novel example, where we know x_i but not y_i , we still know that

$$P[|y_i - h(x_i)| \leq 0.11] = 0.6$$

$$P[|y_i - h(x_i)| \leq 0.13] = 0.8$$

and can obtain $h(x_i)$ from our regression model, e.g. $h(x_i) = 0.3$.



Conformal prediction: intuition

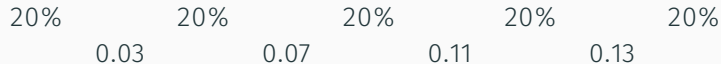
Some intuition

For a novel example, where we know x_i but not y_i , we still know that

$$P[|y_i - h(x_i)| \leq 0.11] = 0.6$$

$$P[|y_i - h(x_i)| \leq 0.13] = 0.8$$

and can obtain $h(x_i)$ from our regression model, e.g. $h(x_i) = 0.3$.



$$P[|y_i - 0.3| \leq 0.11] = 0.6$$

$$P[|y_i - 0.3| \leq 0.13] = 0.8$$

Conformal prediction: intuition

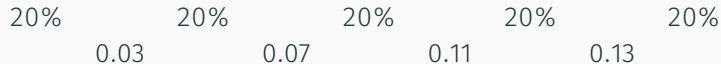
Some intuition

For a novel example, where we know x_i but not y_i , we still know that

$$P[|y_i - h(x_i)| \leq 0.11] = 0.6$$

$$P[|y_i - h(x_i)| \leq 0.13] = 0.8$$

and can obtain $h(x_i)$ from our regression model, e.g. $h(x_i) = 0.3$.



$$P[|y_i - 0.3| \leq 0.11] = 0.6$$

$$P[|y_i - 0.3| \leq 0.13] = 0.8$$

$$P[y_i \in 0.3 \pm 0.11] = 0.6$$

$$P[y_i \in 0.3 \pm 0.13] = 0.8$$

This is actually exactly how conformal regression works!

When does conformal prediction work?

We already noted a few things:

- Training data and test data belong to the same distribution (they are identically distributed)
- Choice of $f(z)$ is irrelevant (w.r.t. validity), as long as it is symmetric (training patterns and test patterns are treated equally)

Conformal prediction at a glance

Conformal predictors output multi-valued **prediction regions**

Given

- a test pattern x_i , and
- a significance level ϵ

A conformal predictor outputs

- A prediction region Γ_i^ϵ that contains y_i with probability $1 - \epsilon$
- In **regression**: real-valued intervals
- In **classification**: (possibly empty) subsets of the possible labels

Let's look at two problems; one multi-class and one regression.

$$Y_c = \{iris_setosa, iris_versicolor, iris_virginica\}$$

$$Y_r = \mathbb{R}$$

Conformal prediction at a glance

Point predictions

$$h_c(x_{k+1}) = \textit{iris_setosa}$$

$$h_c(x_{k+2}) = \textit{iris_versicolor}$$

$$h_c(x_{k+3}) = \textit{iris_virginica}$$

$$h_r(x_{k+1}) = 0.3$$

$$h_r(x_{k+2}) = 0.2$$

$$h_r(x_{k+3}) = 0.6$$

Conformal prediction at a glance

Point predictions

$$h_c(x_{k+1}) = \textit{iris_setosa}$$

$$h_c(x_{k+2}) = \textit{iris_versicolor}$$

$$h_c(x_{k+3}) = \textit{iris_virginica}$$

$$h_r(x_{k+1}) = 0.3$$

$$h_r(x_{k+2}) = 0.2$$

$$h_r(x_{k+3}) = 0.6$$

$$P[y_i = h_c(x_i)] = ?$$

$$\Delta[y_i, h_r(x_i)] = ?$$

Conformal prediction at a glance

Prediction regions

$$h_c(x_{k+1}) = \{iris_setosa\}$$

$$h_c(x_{k+2}) = \{iris_setosa, iris_versicolor\}$$

$$h_c(x_{k+3}) = \{iris_setosa, iris_versicolor, iris_virginica\}$$

$$h_c(x_{k+4}) = \{\}$$

$$h_r(x_{k+1}) = [0.2, 0.4]$$

$$h_r(x_{k+2}) = [0, 0.5]$$

$$h_r(x_{k+3}) = [0.5, 0.7]$$

Conformal prediction at a glance

Prediction regions

$$h_c(x_{k+1}) = \{iris_setosa\}$$

$$h_c(x_{k+2}) = \{iris_setosa, iris_versicolor\}$$

$$h_c(x_{k+3}) = \{iris_setosa, iris_versicolor, iris_virginica\}$$

$$h_c(x_{k+4}) = \{\}$$

$$h_r(x_{k+1}) = [0.2, 0.4]$$

$$h_r(x_{k+2}) = [0, 0.5]$$

$$h_r(x_{k+3}) = [0.5, 0.7]$$

$$P[y_i \in h_c(x_i)] = 1 - \epsilon$$

$$P[y_i \in h_r(x_i)] = 1 - \epsilon$$

Conformal prediction at a glance

To perform conformal prediction, we need

- A function $f(z) \rightarrow \mathbb{R}$
- A set of training examples, $Z^k \subset Z : X^n \times Y$
- A statistical test

Overall rationale

1. Apply $f(z)$ to training examples in Z^k , estimate distribution of $f(z)$
2. For every possible output $\tilde{y} \in Y$, apply $f(z)$ to (x_{k+1}, \tilde{y})
3. Reject \tilde{y} if it appears unlikely that $f[(x_{k+1}, \tilde{y})]$

Conformal prediction at a glance

The function $f(z)$

We call this the **nonconformity function**

- A function that measures the “strangeness” of a pattern (x_i, y_i)
- Any function $f(z) \rightarrow \mathbb{R}$ works (produces valid predictions)

Properties of a good nonconformity function (that produces small prediction sets)

- Give low scores to patterns (x_i, y_i)
- Give large scores to patterns $(x_i, \neg y_i)$

Common choice: $f(z) = \Delta[h(x_i), y_i]$

- h is called the **underlying model**
- “Our random forest misclassified this example, it must be weird!”

Nonconformity functions

Probability estimate for correct class

If the probability estimate for an example's correct class is low, the example is non-conforming.

Margin of a probability estimating model

If an example's true class is not clearly separable from other classes, it is non-conforming.

Distance to neighbors with same class (or distance to neighbors with different classes)

If an example is not surrounded by examples that share its label, it is non-conforming.

Absolute error of a regression model

If the prediction is far from the true value, the example is non-conforming.

`rand(0, 1)`

Even if it's not useful, it's still valid.

Conformal prediction process

1. Define a *nonconformity function*.
2. Measure the nonconformity of labeled examples $(x_1, y_1), \dots, (x_k, y_k)$.
3. For a new pattern x_i , test all possible outputs $\tilde{y} \in Y$:
 - 3.1 Measure the nonconformity of (x_i, \tilde{y}) .
 - 3.2 Is (x_i, \tilde{y}) particularly nonconforming compared to the training examples? Then \tilde{y} is probably an incorrect prediction. Otherwise, include it in the prediction region.

Conformal prediction: formal definition

To determine whether an example is “too nonconforming”, we use a statistical test.

Conformal prediction: formal definition

To determine whether an example is “too nonconforming”, we use a statistical test.

$$p_i^{\tilde{y}} = \frac{\left| \left\{ z_j \in Z : \alpha_j > \alpha_i^{\tilde{y}} \right\} \right|}{k+1} + \theta \frac{\left| \left\{ z_j \in Z : \alpha_j = \alpha_i^{\tilde{y}} \right\} \right| + 1}{k+1}, \theta \sim U[0, 1]$$

(Portion of examples at least as nonconforming as the tentatively labeled test example)

Conformal prediction: formal definition

To determine whether an example is “too nonconforming”, we use a statistical test.

$$p_i^{\tilde{y}} = \frac{\left| \left\{ z_j \in Z : \alpha_j > \alpha_i^{\tilde{y}} \right\} \right|}{k+1} + \theta \frac{\left| \left\{ z_j \in Z : \alpha_j = \alpha_i^{\tilde{y}} \right\} \right| + 1}{k+1}, \theta \sim U[0, 1]$$

(Portion of examples at least as nonconforming as the tentatively labeled test example)

Prediction region

$$\Gamma_i^\epsilon = \left\{ \tilde{y} \in Y : p_i^{\tilde{y}} > \epsilon \right\}$$

Conformal prediction: formal definition

To determine whether an example is “too nonconforming”, we use a statistical test.

$$p_i^{\tilde{y}} = \frac{\left| \left\{ z_j \in Z : \alpha_j > \alpha_i^{\tilde{y}} \right\} \right|}{k+1} + \theta \frac{\left| \left\{ z_j \in Z : \alpha_j = \alpha_i^{\tilde{y}} \right\} \right| + 1}{k+1}, \theta \sim U[0, 1]$$

(Portion of examples at least as nonconforming as the tentatively labeled test example)

Prediction region

$$\Gamma_i^\epsilon = \left\{ \tilde{y} \in Y : p_i^{\tilde{y}} > \epsilon \right\}$$

- Classification — known $\alpha_i^{\tilde{y}}$, find $p_i^{\tilde{y}}$
- Regression — known $p_i^{\tilde{y}}$, find $\alpha_i^{\tilde{y}}$

Types of conformal predictors

Transductive conformal prediction (TCP) — $f(z, Z)$

Original conformal prediction approach

- Requires retraining model for each new test example
- For regression problems, only certain models (e.g. kNN) can be used as of yet

Inductive conformal prediction (ICP) — $f(z)$

Revised approach

- Requires model to be trained only once
- Requires that some data is set aside for calibration
 - To avoid violating exchangeability assumption

Conformal regression

Inductive Conformal Regression

Divide the training set Z into two disjoint subsets

A proper training set Z_t

A calibration set Z_c where $|Z_c| = q$

Inductive Conformal Regression

Divide the training set Z into two disjoint subsets

A proper training set Z_t

A calibration set Z_c where $|Z_c| = q$

Fit a model h using Z_t

This is the underlying model

Inductive Conformal Regression

Divide the training set Z into two disjoint subsets

A proper training set Z_t

A calibration set Z_c where $|Z_c| = q$

Fit a model h using Z_t

This is the underlying model

Let $f(z_i) = |y_i - h(x_i)|$

This is the nonconformity function

Inductive Conformal Regression

Divide the training set Z into two disjoint subsets

A proper training set Z_t

A calibration set Z_c where $|Z_c| = q$

Fit a model h using Z_t

This is the underlying model

Let $f(z_i) = |y_i - h(x_i)|$

This is the nonconformity function

Apply $f(z)$ to $\forall z_i \in Z_c$

Save these calibration scores, sorted in descending order

We denote these $\alpha_1, \dots, \alpha_q$

Inductive Conformal Regression

Fix a significance level $\epsilon \in (0, 1)$

Let $s = \lfloor \epsilon(q + 1) \rfloor$.

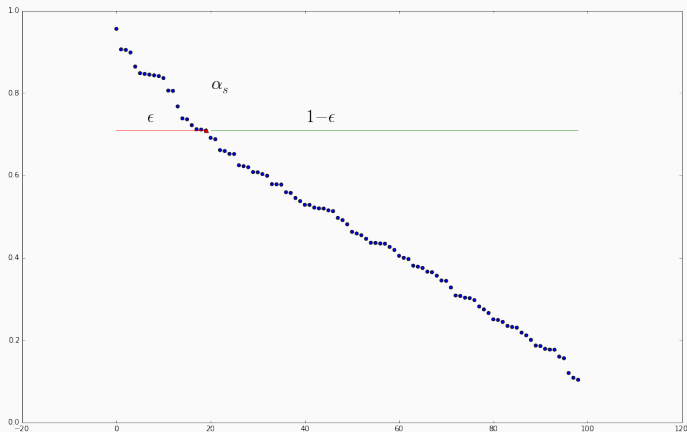
This is the index of the $(1 - \epsilon)$ -percentile nonconformity score, α_s .

Inductive Conformal Regression

Fix a significance level $\epsilon \in (0, 1)$

Let $s = \lfloor \epsilon(q + 1) \rfloor$.

This is the index of the $(1 - \epsilon)$ -percentile nonconformity score, α_s .



Inductive Conformal Regression

The prediction for a new example is $\Gamma_i^\epsilon = h(x_i) \pm \alpha_s$

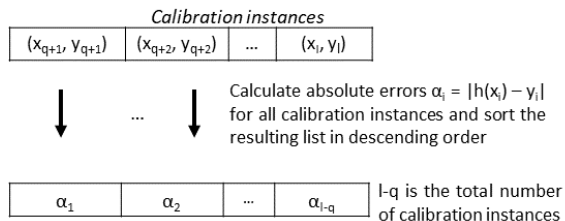
The interval contains y_i with probability $1 - \epsilon$

The motivation is very straightforward; if the calibration and test sets are exchangeable, the probability of a test instance obtaining a larger absolute error than the absolute error of the $(1 - \epsilon)$ -percentile calibration instance must be exactly (ϵ) .

As an example: we expect to see 20% of the test instances to have larger absolute errors than the calibration instance corresponding to the 80-percentile.

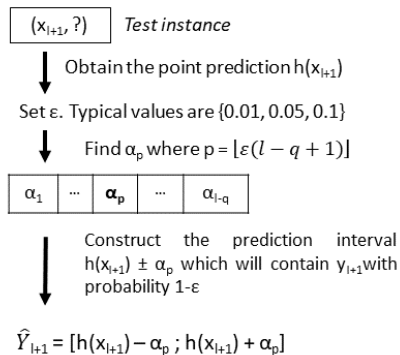
Inductive Conformal Regression - Summary

Calibration step



1. Use an underlying regression model and a calibration set with known targets not used for training.
2. Calculate the absolute errors α_i for all calibration instances and sort this list in descending order
3. When predicting – obtain the point prediction $h(x_{l+1})$ from the underlying model.
4. Set a significance level ε
5. Pick the absolute error from the calibration set corresponding to the chosen significance level, i.e., α_p where $p = \lfloor \varepsilon(l - q + 1) \rfloor$
6. The prediction interval is $h(x_{l+1}) \pm \alpha_p$

Prediction step



Inductive Conformal Regression

A sample regression problem - Boston Housing

Attributes:

- CRIM: per capita crime rate by town
- ZN: proportion of residential land zoned for lots over 25,000 sq.ft.
- INDUS: proportion of non-retail business acres per town
- CHAS: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
- NOX: nitric oxides concentration (parts per 10 million)
- RM: average number of rooms per dwelling
- AGE: proportion of owner-occupied units built prior to 1940
- DIS: weighted distances to five Boston employment centres
- RAD: index of accessibility to radial highways
- TAX: full-value property-tax rate per \$10000
- PTRATIO: pupil-teacher ratio by town
- B: $1000(B_k - 0.63)^2$ where B_k is the proportion of blacks by town
- LSTAT: % lower status of the population
- Price

Inductive Conformal Regression

Predicting price - 16 sample instances

	$\epsilon = 0.2$		$\epsilon = 0.1$		$\epsilon = 0.05$		$\epsilon = 0.01$	
Correct	Min	Max	Min	Max	Min	Max	Min	Max
10.8	6.7	23.2	2.7	27.3	0.0	31.0	0.0	40.7
14.9	9.9	26.4	5.8	30.4	2.1	34.1	0.0	43.8
12.6	10.4	26.3	6.6	30.1	3.0	33.7	0.0	43.0
14.9	16.8	30.2	13.5	33.5	10.5	36.5	2.6	44.4
19.1	9.2	25.6	5.2	29.6	1.5	33.3	0.0	43.0
20.1	11.7	28.1	7.7	32.1	4.1	35.8	0.0	45.4
19.9	10.2	26.5	6.2	30.5	2.5	34.2	0.0	43.9
23	12.9	29.2	8.9	33.2	5.2	36.9	0.0	46.6
23.7	20.5	36.4	16.7	40.2	13.1	43.8	3.8	53.1
21.8	13.1	28.5	9.4	32.2	6.0	35.7	0.0	44.7
20.6	13.0	29.4	9.0	33.4	5.3	37.1	0.0	46.7
19.1	11.1	27.4	7.1	31.4	3.4	35.1	0.0	44.8
15.2	10.3	26.8	6.3	30.8	2.6	34.5	0.0	44.3
7.0	7.7	24.2	3.6	28.2	0.0	31.9	0.0	41.6
24.5	18.0	23.4	16.6	24.8	15.4	26.0	12.2	29.2
11.9	17.8	24.1	16.3	25.6	14.9	27.1	11.1	30.8

Inductive Conformal Regression

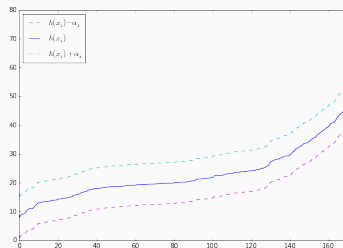
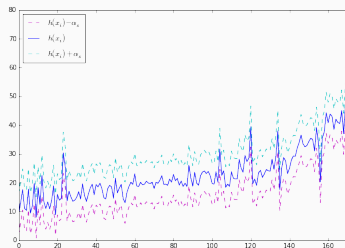
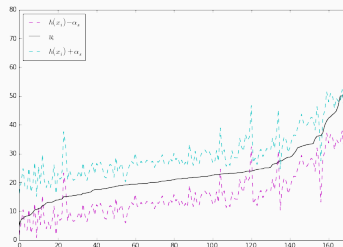
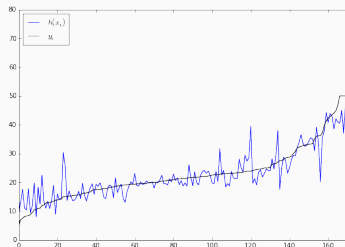
Overall results

	$\epsilon = 0.2$	$\epsilon = 0.1$	$\epsilon = 0.05$	$\epsilon = 0.01$
Errors	0.201	0.090	0.053	0.011
Average interval	10.1	16.0	19.4	32.8

- For regression problems, an error is when the target variable is outside of the interval.
- The probability for an error is always the chosen ϵ .
- An obvious and user-controlled trade-off between errors and prediction size
- This data set is rather small, so the empirical error rates differ slightly from ϵ

Inductive Conformal Regression

Boston Housing, Random Forest, $\epsilon = 0.1$



Normalized Inductive Conformal Regression

Static prediction interval size

Using $f(z_i) = |y_i - h(x_i)|$ and $\Gamma_i^\epsilon = h(x_i) \pm \alpha_s$

means each prediction interval has the same size ($2\alpha_s$).

But we want individual bounds for each x_i ...

Normalized Inductive Conformal Regression

Static prediction interval size

Using $f(z_i) = |y_i - h(x_i)|$ and $\Gamma_i^\epsilon = h(x_i) \pm \alpha_s$

means each prediction interval has the same size ($2\alpha_s$).

But we want individual bounds for each x_i ...

Normalized nonconformity functions

Normalized nonconformity functions utilize an additional term σ_i .

$$f(z_i) = \frac{|y_i - h(x_i)|}{\sigma_i}$$

σ_i is an estimate of the difficulty of predicting y_i

A common practice is to let σ be predicted by a model, e.g., $\sigma_i = \hat{\Delta}[y_i, h(x_i)]$, but there are several other possibilities.

Normalized Inductive Conformal Regression

Static prediction interval size

Using $f(z_i) = |y_i - h(x_i)|$ and $\Gamma_i^\epsilon = h(x_i) \pm \alpha_s$
means each prediction interval has the same size ($2\alpha_s$).

But we want individual bounds for each x_i ...

Normalized nonconformity functions

Normalized nonconformity functions utilize an additional term σ_i .

$$f(z_i) = \frac{|y_i - h(x_i)|}{\sigma_i}$$

σ_i is an estimate of the difficulty of predicting y_i

A common practice is to let σ be predicted by a model, e.g., $\sigma_i = \hat{\Delta}[y_i, h(x_i)]$, but there are several other possibilities.

The normalized prediction for a new example is $\Gamma_i^\epsilon = h(x_i) \pm \alpha_s \sigma_i$

Normalized Inductive Conformal Regression

Divide the training set Z into two disjoint subsets

A proper training set Z_t

A calibration set Z_c

Fit a model h using Z_t

In addition

- Let E_t be the residual errors of h (i.e. the errors that h makes on Z_t)
- Fit a model g using $X_t \times E_t$

$$f(z_i) = \frac{|y_i - h(x_i)|}{g(x_i) + \beta}$$

β is a sensitivity parameter that determines the impact of normalization

Apply $f(z)$ to $\forall z_i \in Z_c$

Save these calibration scores, sorted in descending order

Normalized Inductive Conformal Regression

Fix a significance level $\epsilon \in (0, 1)$

Let $s = \lfloor \epsilon(q + 1) \rfloor$

This is the index of the $(1 - \epsilon)$ -percentile nonconformity score, α_s .

Prediction region

The prediction for a new example is $\Gamma_i^\epsilon = h(x_i) \pm \alpha_s(g(x_i) + \beta)$

Interval contains y_i with probability $1 - \epsilon$

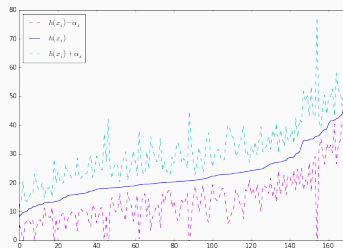
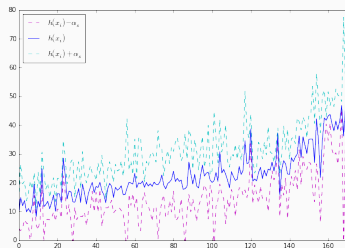
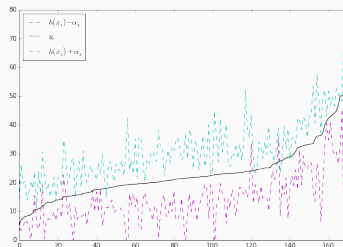
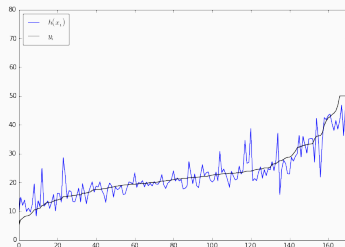
Effects of normalization

Normalization produces more specific (individualized) predictions.

The intervals tend to be tighter, on average, when using normalization.

Inductive Conformal Regression

Boston Housing, Random Forest, normalized nonconformity function, $\epsilon = 0.1$



Some alternative difficulty estimators for normalization

- The variance in the target values for k nearest neighbors⁵
- The variance in the predictions from the different trees in a random forest⁶
- The variance in target values for the training instances falling in a specific leaf in a regression tree⁷

⁵U. Johansson, H. Boström, T. Löfström, and H. Linusson, “Regression conformal prediction with random forests,” *Machine Learning*, vol. 97, no. 1-2, pp. 155–176, 2014

⁶H. Boström, H. Linusson, T. Löfström, and U. Johansson, “Accelerating difficulty estimation for conformal regression forests,” *Annals of Mathematics and Artificial Intelligence*, pp. 1–20, 2017

⁷U. Johansson, H. Linusson, T. Löfström, and H. Boström, “Interpretable regression trees using conformal prediction,” *Expert Syst. Appl.*, vol. 97, pp. 394–404, 2018

Conformal classification

Conformal prediction can also be used for classification, in fact a majority of published studies are about classification.

- The overall procedure is very similar; use a calibration set and a non-conformity function to find prediction regions.
- Here, the key idea is to test the nonconformity of **each possible label**, together with a test instance, in order to reject unlikely labels.
- The resulting predictions are (sometimes empty) **subsets** of the possible labels.
- An error is when the correct label is not in the prediction set.
- The error rate will, in the long run, be equal to ϵ .

Conformal classification

Conformal prediction can also be used for classification, in fact a majority of published studies are about classification.

- The overall procedure is very similar; use a calibration set and a non-conformity function to find prediction regions.
- Here, the key idea is to test the nonconformity of **each possible label**, together with a test instance, in order to reject unlikely labels.
- The resulting predictions are (sometimes empty) **subsets** of the possible labels.
- An error is when the correct label is not in the prediction set.
- The error rate will, in the long run, be equal to ϵ .

I will not cover conformal classification in detail, since we recommend the usage of Venn predictors instead, but I have left some slides in the presentation, after the references.

Predicting whether a customer will churn or not

- A data set from one of the leading e-retailers in Sweden consisting of altogether 255298 customers.
- The target variable for the analysis is whether the specific customer will churn or not, i.e., no purchase one year after the previous order.
- Each customer is described using altogether 276 attributes.
- We are not allowed to give a detailed description of all the attributes, but they include statistics like number of orders, number of visits to the website and whether the customer has clicked on promotion emails sent by the retailer.

A real-world example

Predicting whether a customer will churn or not - 16 sample instances

Correct	$\epsilon = 0.2$	$\epsilon = 0.1$	$\epsilon = 0.05$	$\epsilon = 0.01$
Churn	{Churn}	{Churn}	{Churn}	{Churn}
Loyal	{Churn}	{Churn}	{Loyal, Churn}	{Loyal, Churn}
Loyal	{}	{Loyal}	{Loyal}	{Loyal}
Churn	{Loyal, Churn}	{Loyal, Churn}	{Loyal, Churn}	{Loyal, Churn}
Churn	{Churn}	{Churn}	{Loyal, Churn}	{Loyal, Churn}
Churn	{Churn}	{Churn}	{Churn}	{Loyal, Churn}
Loyal	{Loyal}	{Loyal}	{Loyal, Churn}	{Loyal, Churn}
Churn	{Churn}	{Churn}	{Churn}	{Churn}
Loyal	{Loyal}	{Loyal, Churn}	{Loyal, Churn}	{Loyal, Churn}
Loyal	{Loyal}	{Loyal}	{Loyal}	{Loyal, Churn}
Churn	{Churn}	{Loyal, Churn}	{Loyal, Churn}	{Loyal, Churn}
Churn	{Churn}	{Loyal, Churn}	{Loyal, Churn}	{Loyal, Churn}
Loyal	{Loyal}	{Loyal}	{Loyal}	{Loyal}
Churn	{Loyal}	{Loyal}	{Loyal}	{Loyal, Churn}
Loyal	{Loyal, Churn}	{Loyal, Churn}	{Loyal, Churn}	{Loyal, Churn}
Loyal	{Loyal}	{Loyal}	{Loyal}	{Loyal, Churn}

A real-world example

Predicting whether a customer will churn or not - overall results

	$\epsilon = 0.2$	$\epsilon = 0.1$	$\epsilon = 0.05$	$\epsilon = 0.01$
RF 300				
AvgC	1.061	1.334	1.519	1.791
OneC	0.939	0.666	0.481	0.209
Errors	0.202	0.100	0.052	0.010
LogReg				
AvgC	1.075	1.347	1.525	1.790
OneC	0.925	0.653	0.475	0.210
Errors	0.199	0.096	0.050	0.011

- For classification, an error is when the correct label is not in the prediction set, i.e., for two-class problems incorrect singleton predictions and empty predictions.
- The probability for an error is always the chosen ϵ .
- An obvious and user-controlled trade-off between errors and prediction size

Validity and efficiency

Conformal predictors are subject to two desiderata

Validity — coherence between ϵ and error rate

Efficiency — size of prediction regions (i.e. informativeness)

Conformal predictors are automatically valid

Efficiency depends on the nonconformity function (and thus the underlying model)

Conformal predictors are subject to two desiderata

Validity — coherence between ϵ and error rate

Efficiency — size of prediction regions (i.e. informativeness)

Conformal predictors are automatically valid

Efficiency depends on the nonconformity function (and thus the underlying model)

Confidence-efficiency trade-off

The more confidence we require in a prediction, the larger the prediction region will be

ϵ	errors	size
0.01	0.006	38.31
0.05	0.040	16.90
0.10	0.089	11.46
0.20	0.191	7.562

Table 1: Boston 10x10 RF CV

ϵ	errors	size
0.01	0.011	2.347
0.05	0.055	1.052
0.10	0.100	0.930
0.20	0.202	0.804

Table 2: Iris 10x10 RF CV

Empirical validity is measured by observing the error rate of a conformal predictor.

Efficiency can be measured in many different ways⁸.

Examples — regression

- Average size of prediction interval

Examples — classification

- Average number of classes per prediction (AvgC)
- Rate of predictions containing a single class (OneC)
- Average p -value

⁸V. Vovk, V. Fedorova, I. Nouretdinov, and A. Gammerman, “Criteria of efficiency for conformal prediction,” 2014

Choosing a calibration set size

The calibration set

Inductive conformal predictors need some data set aside for calibration? — How much?

25% ~ 33% are common choices, and provide a good balance between underlying model performance and calibration accuracy⁹.

Alternatives

Bagged ensembles can use out-of-bag examples for calibration^{10 11}.

⁹H. Linusson, U. Johansson, H. Boström, and T. Löfström, “Efficiency comparison of unstable transductive and inductive conformal classifiers,” in *Artificial Intelligence Applications and Innovations*. Springer, 2014, pp. 261–270

¹⁰U. Johansson, H. Boström, T. Löfström, and H. Linusson, “Regression conformal prediction with random forests,” *Machine Learning*, vol. 97, no. 1-2, pp. 155–176, 2014

¹¹H. Boström, H. Linusson, T. Löfström, and U. Johansson, “Accelerating difficulty estimation for conformal regression forests,” *Annals of Mathematics and Artificial Intelligence*, pp. 1–20, 2017

Choosing a calibration set size

The calibration set cont.

For an inductive conformal predictor to be exactly valid, it requires exactly $k\epsilon^{-1} - 1$ calibration instances.

- Otherwise, discretization errors come into play
 - (Rendering the conformal predictor conservatively valid)
- Of particular importance when calibration set is small
 - e.g. when using conditional conformal prediction

Alternatives

Interpolation of p -values can alleviate this problem.^{12 13}

¹²L. Carlsson, E. Ahlberg, H. Boström, U. Johansson, and H. Linusson, “Modifications to p-values of conformal predictors,” in *Statistical Learning and Data Sciences*. Springer, 2015, pp. 251–259

¹³U. Johansson, E. Ahlberg, H. Boström, L. Carlsson, H. Linusson, and C. Sönströd, “Handling small calibration sets in mondrian inductive conformal regressors,” in *Statistical Learning and Data Sciences*. Springer, 2015, pp. 271–280

Conformal classification - a critical look

The problem with conformal classification

Counter-intuitive?

The problem with conformal classification

Counter-intuitive?

- We must be very careful when interpreting conformal classifiers.

The problem with conformal classification

Counter-intuitive?

- We must be very careful when interpreting conformal classifiers.
- We will make exactly ϵ errors in the long run.

The problem with conformal classification

Counter-intuitive?

- We must be very careful when interpreting conformal classifiers.
- We will make exactly ϵ errors in the long run.
- An error is when the correct label is not in the predicted label set.

The problem with conformal classification

Counter-intuitive?

- We must be very careful when interpreting conformal classifiers.
- We will make exactly ϵ errors in the long run.
- An error is when the correct label is not in the predicted label set.
- With this in mind, the guarantee really only applies *apriori*, i.e., once we have seen a specific prediction, *we can not say that the probability for that prediction to be wrong is ϵ .*

The problem with conformal classification

Counter-intuitive?

- We must be very careful when interpreting conformal classifiers.
- We will make exactly ϵ errors in the long run.
- An error is when the correct label is not in the predicted label set.
- With this in mind, the guarantee really only applies *apriori*, i.e., once we have seen a specific prediction, *we can not say that the probability for that prediction to be wrong is ϵ .*
- As an example, consider a two-class problem. Here a number of instances are likely to get prediction sets containing both classes, meaning that these instances cannot be erroneous.

The problem with conformal classification

Counter-intuitive?

- We must be very careful when interpreting conformal classifiers.
- We will make exactly ϵ errors in the long run.
- An error is when the correct label is not in the predicted label set.
- With this in mind, the guarantee really only applies *apriori*, i.e., once we have seen a specific prediction, *we can not say that the probability for that prediction to be wrong is ϵ .*
- As an example, consider a two-class problem. Here a number of instances are likely to get prediction sets containing both classes, meaning that these instances cannot be erroneous.
- Thus, all errors must be made on the remaining singleton predictions.

The problem with conformal classification

Counter-intuitive?

- We must be very careful when interpreting conformal classifiers.
- We will make exactly ϵ errors in the long run.
- An error is when the correct label is not in the predicted label set.
- With this in mind, the guarantee really only applies *apriori*, i.e., once we have seen a specific prediction, *we can not say that the probability for that prediction to be wrong is ϵ .*
- As an example, consider a two-class problem. Here a number of instances are likely to get prediction sets containing both classes, meaning that these instances cannot be erroneous.
- Thus, all errors must be made on the remaining singleton predictions.
- So, once we have observed a singleton prediction, the probability for that being incorrect is most likely much higher than ϵ .

The problem with conformal classification

Counter-intuitive?

- We must be very careful when interpreting conformal classifiers.
- We will make exactly ϵ errors in the long run.
- An error is when the correct label is not in the predicted label set.
- With this in mind, the guarantee really only applies *apriori*, i.e., once we have seen a specific prediction, *we can not say that the probability for that prediction to be wrong is ϵ .*
- As an example, consider a two-class problem. Here a number of instances are likely to get prediction sets containing both classes, meaning that these instances cannot be erroneous.
- Thus, all errors must be made on the remaining singleton predictions.
- So, once we have observed a singleton prediction, the probability for that being incorrect is most likely much higher than ϵ .
- It must be noted that this “problem” does not exist in conformal regression.

Probabilistic prediction

Introduction

- Many classifiers are able to output not only the predicted class label, but also a probability distribution over the possible classes.
- Naturally, all probabilistic prediction requires that the probability estimates are **well-calibrated**, i.e., the predicted class probabilities must reflect the true, underlying probabilities.
- If the model is overconfident or underconfident, the probabilistic predictions actually become **misleading**.

Calibration

- In probabilistic prediction, the goal is to obtain a **valid** predictor.
- In general, validity means that the probability distributions from the predictor must perform well against statistical tests based on subsequent observation of the labels.
- We are interested in **calibration**: $p(c_j | p^{c_j}) = p^{c_j}$, where p^{c_j} is the probability estimate for class j .
- Informally, if we make a number of predictions where the highest class membership probability is, say, 0.9, we expect 10% of these predictions to be errors.

Calibration

- While most models are capable of producing probability estimates, these are often very poorly calibrated.
- Models like Naive Bayes¹⁴ and decision trees¹⁵ are two well-known examples.
- But recent studies show that even models assumed to be well-calibrated off-the-shelf, like modern (i.e., deep) neural networks¹⁶ and traditional neural networks¹⁷ often are not.

¹⁴A. Niculescu-Mizil and R. Caruana, “Predicting good probabilities with supervised learning,” in *ICML*. ACM, 2005, pp. 625–632

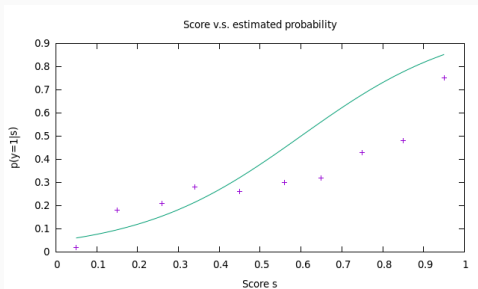
¹⁵F. Provost and P. Domingos, “Tree induction for probability-based ranking,” *Mach. Learn.*, vol. 52, no. 3, pp. 199–215, 2003

¹⁶C. 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*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 2017, pp. 1321–1330

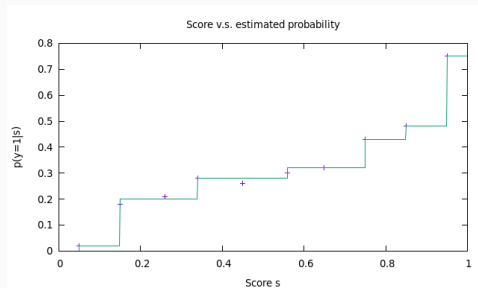
¹⁷U. Johansson and P. Gabrielsson, “Are traditional neural networks well-calibrated?” in *IJCNN*, 2019, In Press

Existing approaches for calibration

Platt scaling fits a sigmoid function to a calibration set.



Isotonic regression fits an isotonic, i.e., non-decreasing, calibration function.



Venn predictors¹⁸, are multi-probabilistic predictors with proven validity properties.

Venn predictors was originally suggested in a transductive setting, but here we describe the inductive variant:

To construct an inductive Venn predictor, the available labeled training examples $\{(x_1, y_1), \dots, (x_l, y_l)\}$ are split into two parts, the proper training set $\{(x_1, y_1), \dots, (x_q, y_q)\}$, used to train an underlying model, and a calibration set $\{(x_{q+1}, y_{q+1}), \dots, (x_l, y_l)\}$ used to estimate label probabilities for each new test example.

When presented with a new test object x_{l+1} , the aim of Venn prediction is to estimate the probability that $y_{l+1} = Y_j$, for each Y_j in the set of possible labels $Y_j \in \{Y_1, \dots, Y_c\}$.

¹⁸V. Vovk, G. Shafer, and I. Nouretdinov, “Self-calibrating probability forecasting,” in *Advances in Neural Information Processing Systems*, 2004, pp. 1133–1140

The key idea of Venn prediction is to divide all calibration examples into a number of k **categories** and use the relative frequencies of labels $Y_j \in \{Y_1, \dots, Y_c\}$ in each category to estimate label probabilities for test instances falling into that category.

The categories are defined using a **Venn taxonomy** and every taxonomy leads to a different Venn predictor.

Typically, the taxonomy is based on the underlying model, trained on the proper training set, and for each calibration and test object x_i , the output of this model is used to assign (x_i, y_i) into one of the categories.

One basic Venn taxonomy, which can be used with every kind of classification model, simply puts all examples predicted with the same label into the same category.

For test instances, the category is first determined using the underlying model, in an identical way as for the calibration instances. Then, **the label frequencies** of the calibration instances in that category are used to calculate the estimated label probabilities.

To ensure validity, the test instance z_{l+1} must be included in this calculation. However, since the true label y_{l+1} is not known for the test object x_{l+1} , **all possible labels** $Y_j \in \{Y_1, \dots, Y_c\}$ are used to create **a set of label probability distributions**.

Instead of dealing directly with these distributions, the lower $L(Y_j)$ and upper $U(Y_j)$ probability estimates for each label Y_j are often used.

Let k be the category assigned to the test object x_{l+1} by the Venn taxonomy, and Z_k be the set of calibration instances belonging to category k . Then the lower and upper probability estimates are defined by:

$$L(Y_j) = \frac{|\{(x_m, y_m) \in Z_k \mid y_m = Y_j\}|}{|Z_k| + 1} \quad (1)$$

and:

$$U(Y_j) = \frac{|\{(x_m, y_m) \in Z_k \mid y_m = Y_j\}| + 1}{|Z_k| + 1} \quad (2)$$

In order to make a prediction \hat{y}_{l+1} for x_{l+1} using the lower and upper probability estimates, the following procedure is often used:

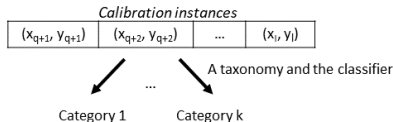
$$\hat{y}_{l+1} = \max_{Y_j \in \{Y_1, \dots, Y_c\}} L(Y_j) \quad (3)$$

The output of a Venn predictor is the above prediction \hat{y}_{l+1} together with the probability interval:

$$[L(\hat{y}_{l+1}), U(\hat{y}_{l+1})] \quad (4)$$

Inductive Venn prediction - Summary

Calibration step



1. Use an underlying classifier and a taxonomy to divide all calibration instances into a number of categories
2. When predicting a test instance, find the category in an identical way as for the calibration instances
3. Use the relative frequencies of the labels in that category as the probability estimates, but include the test instance with all possible labels in that calculation, leading to a set of probability intervals
4. Aggregate this into a prediction and a probability interval for the prediction

Prediction step

$(x_{l+1}, ?)$ Test instance

Find category using identical procedure as when calibrating

Category i	
Label	Frequency
C_1	f_1
...	...
C_m	f_m

Assign each possible label, one at the time, to the test instance and calculate probability distributions using the relative frequencies of the labels in that category – including the tentatively assigned label for the test instance

For each label: calculate the relative frequency when the test instance belongs to that class and not, resulting in m probability intervals for x_{l+1}

Category i		
Label	Low	High
C_1	$f_1 / (l-q+1)$	$(f_1+1) / (l-q+1)$
...
C_m	$f_m / (l-q+1)$	$(f_m+1) / (l-q+1)$

$l-q$ is the total number of calibration instances

Predict the label j with the highest Low, and return that label together with the corresponding probability interval

$$\hat{Y}_{l+1} = C_j : [f_j / (l-q+1); (f_j+1) / (l-q+1)]$$

Venn predictors

While the multiprobability predictions produced by Venn predictors are **automatically valid**, regardless of the taxonomy used, the taxonomy affects both the **accuracy** of the Venn predictor and the **size** of the prediction interval.

Obviously, the probability estimates should preferably be as close to one or zero as possible, and tighter probability intervals are more informative.

The more categories that are used in the taxonomy, the more **specific** the predictions will be.

For two-class problems, the basic taxonomy that puts all the examples predicted with the same label into the same category will have exactly two categories, i.e., the Venn predictor will for every test instance output **one of only two** possible prediction intervals.

On the other hand, with too many categories, the calibration will depend on just a few instances, resulting in very large intervals.

Venn-Abers predictors¹⁹ are Venn predictors applicable to two-class problems, where the taxonomy is automatically optimized using isotonic regression.

Thus, the Venn-Abers predictor inherits the **validity guarantee** of Venn predictors, while providing specific predictions.

Venn-Abers predictors regard the underlying models as *scoring classifiers*, i.e., when an underlying model makes a prediction for a test object, the output is a *prediction score* $s(x)$, where a higher value indicates a larger belief in that the test instance has the label 1.

Venn-Abers predictors use isotonic regression for the calibration.

An isotonic calibrator is **fitted twice** to the calibration set and the test instance, once with the tentative label 0 and once with the tentative label 1.

¹⁹V. Vovk and I. Petej, “Venn-abers predictors,” *arXiv preprint arXiv:1211.0025*, 2012

let s_0 be the scoring function for $\{z_{q+1}, \dots, z_l, (x_{l+1}, 0)\}$,

let s_1 be the scoring function for $\{z_{q+1}, \dots, z_l, (x_{l+1}, 1)\}$,

let g_0 be the isotonic calibrator for $\{(s_0(x_{q+1}), y_{q+1}), \dots, (s_0(x_l), y_l), (s_0(x_{l+1}), 0)\}$

let g_1 be the isotonic calibrator for $\{(s_1(x_{q+1}), y_{q+1}), \dots, (s_1(x_l), y_l), (s_1(x_{l+1}), 1)\}$

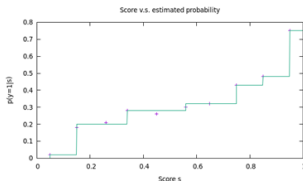
Then the probability interval for $y_{l+1} = 1$ is $[g_0(s_0(x_{l+1})), g_1(s_1(x_{l+1}))]$

Inductive Venn-Abers prediction - Summary

In Venn-Abers, the taxonomy is automatically optimized using isotonic regression

1. Venn-Abers classifiers assume *scoring classifiers*, i.e., they are restricted to two-class problems.
2. An *isotonic calibrator* is fitted twice to the calibration set for every test instance x_{i+1} , once augmented with the test instance $(x_{i+1}, 0)$ and once with $(x_{i+1}, 1)$
3. These isotonic calibrators are used to find the probability interval for $\hat{Y}_{i+1} = 1$

Isotonic regression:



Calibration step

Calibration set

(x_{q+1}, y_{q+1})	(x_{q+2}, y_{q+2})	...	(x_l, y_l)
----------------------	----------------------	-----	--------------



Calculate $s(x_i)$ for all calibration instances, where s is a scoring function, i.e., a higher value indicates a higher belief in class 1

$(s(x_{q+1}), y_{q+1})$	$(s(x_{q+2}), y_{q+2})$...	$(s(x_l), y_l)$
-------------------------	-------------------------	-----	-----------------

Prediction step

$(x_{i+1}, ?)$



Let g_0 be the isotonic calibrator for $\{(s(x_{q+1}), y_{q+1}), \dots, (s(x_l), y_l), (s(x_{i+1}), 0)\}$

Let g_1 be the isotonic calibrator for $\{(s(x_{q+1}), y_{q+1}), \dots, (s(x_l), y_l), (s(x_{i+1}), 1)\}$



$$\hat{Y}_{i+1} = 1 : [g_0(s(x_{i+1})); g_1(s(x_{i+1}))]$$

Results - Predictive performance

Here are some results from a recent paper²⁰ using Venn-Abers for calibrating decision trees. All results are over the 25 data sets. Detailed results can be found in the paper.

Accuracy								
	RF				LaP			
	Tree	Platt	Iso	VAP	Tree	Platt	Iso	VAP
Mean	.757	.768	.770	.770	.757	.771	.775	.776
Mean rank	2.92	2.84	2.26	1.98	3.16	2.84	2.24	1.76

AUC								
	RF				LaP			
	Tree	Platt	Iso	VAP	Tree	Platt	Iso	VAP
Mean	.726	.710	.727	.731	.770	.735	.756	.760
Mean rank	2.24	3.72	2.56	1.48	1.12	3.96	3.00	1.92

²⁰U. Johansson, T. Löfström, and H. Boström, "Calibrating probability estimation trees using venn-abers predictors," in *SIAM International Conference on Data Mining*. Society for Industrial and Applied Mathematics, 2019, pp. 28–36

Results - Quality of estimates

Difference (prediction - true target)

	RF				LaP			
	Tree	Platt	Iso	VAP	Tree	Platt	Iso	VAP
Mean	.168	.005	.021	.003	.121	.009	.025	.001

Logloss

	RF				LaP			
	Tree	Platt	Iso	VAP	Tree	Platt	Iso	VAP
Mean	∞	.725	∞	.710	.795	.707	∞	.681
Mean rank	3.54	1.96	3.42	1.08	2.80	2.12	4.00	1.08

Brier loss

	RF				LaP			
	Tree	Platt	Iso	VAP	Tree	Platt	Iso	VAP
Mean	.201	.164	.163	.161	.175	.160	.157	.155
Mean rank	3.72	2.68	2.44	1.16	3.52	3.04	2.28	1.16

Results - VAP intervals and empirical error rates

	RF			LaP		
	Low	High	Acc	Low	High	Acc
colic	.798	.833	.818	.811	.853	.834
creditA	.824	.849	.834	.827	.861	.838
diabetes	.714	.737	.712	.717	.749	.719
german	.694	.709	.701	.692	.711	.700
haberman	.699	.742	.701	.696	.748	.703
heartC	.741	.788	.761	.745	.803	.764
heartH	.759	.806	.768	.763	.819	.775
heartS	.741	.792	.757	.743	.806	.758
hepati	.777	.839	.784	.776	.846	.788
iono	.863	.894	.880	.859	.906	.883
je4042	.688	.740	.695	.695	.757	.703
je4243	.624	.662	.613	.626	.675	.616
kc1	.729	.742	.730	.733	.753	.737
kc2	.748	.784	.747	.757	.805	.767
kc3	.848	.883	.862	.842	.886	.861
liver	.635	.676	.626	.641	.691	.639
pc1req	.617	.723	.626	.621	.730	.635
pc4	.873	.887	.874	.876	.895	.882
sonar	.691	.737	.705	.697	.763	.707
spect	.854	.901	.884	.849	.901	.884
spectf	.778	.813	.783	.774	.823	.786
transfusion	.725	.757	.727	.725	.765	.733
ttt	.907	.927	.918	.894	.929	.919
wbc	.895	.926	.914	.892	.931	.915
vote	.832	.870	.844	.831	.873	.846

Algorithmic confidence for FAT and XAI

- As AI is increasingly used not only for decision support, but also automated decision making, **trust** in the resulting decisions or recommendations becomes vital.
- Consequently, how to make AI solutions trustworthy is today a key question addressed by researchers from many disciplines..
- AI trustworthiness is also strongly manifested in the two vibrant areas **Explainable AI (XAI)** and **Fairness, Accountability and Transparency (FAT)**.

- Interpretability is currently recognized as a key property of trustworthy predictive models
- Only interpretable models make it possible to **understand** individual predictions, without the usage of specialized, and often very complex, explanation modules.
- In addition, with interpretable models, **inspection** and **analysis** of the model itself becomes straightforward.
- The importance of interpretable models, e.g., for user acceptance, has been present in the AI discourse since the era of expert systems, and it is also prominent in recent high-impact publications within machine learning, such as the LIME framework²¹

²¹M. T. Ribeiro, S. Singh, and C. Guestrin, “Why should i trust you?: Explaining the predictions of any classifier,” in *Proceedings of the 22Nd ACM SIGKDD*, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 1135–1144

- The FAT/ML Principles for Accountable Algorithms and a Social Impact Statement for Algorithms²² include both **explainability** and **accuracy** as vital components of accountable algorithms.
- One guiding question for Accountable Algorithms is: “How confident are the decisions output by your system?” Thus, accountability puts demands on not only explainability and accuracy, but also an ability to, at the very least, **report uncertainty**.
- In fact, the ability for an algorithm to somehow **reason about its own competence**, specifically about confidence in **individual recommendations**, is deemed to be extremely valuable.
- In our opinion, the prediction with confidence framework is **uniquely well** positioned to meet these demands.

²²N. Diakopoulos, S. Friedler, M. Arenas, S. Barocas, M. Hay, B. Howe, H. V. Jagadish, K. Unsworth, A. Sahuguet, S. Venkatasubramanian, C. Wilson, and B. Z. C. Yu, *Principles for Accountable Algorithms and a Social Impact Statement for Algorithms*, FAT/ML, 2017

Requirements

- Interpretable models, e.g., decision trees or rule sets.
- Well-calibrated models
- Exhibiting confidence for individual predictions.
- Fixed models (after the calibration step) making them available for inspection and analysis.

Example - conformal regression trees²³

```
x6 < 10.58
|
|   x6 < 7.25
|   |
|   |   x5 < 4.54
|   |   |
|   |   |   y = 0.078 {15}
|   |   |   x5 >= 4.54
|   |   |   |
|   |   |   |   y = 0.185 {34}
|   |   |   |
|   |   |   x6 >= 7.25
|   |   |   |
|   |   |   |   x5 < 7.195
|   |   |   |   |
|   |   |   |   |   y = 0.293 {11}
|   |   |   |   |   x5 >= 7.195
|   |   |   |   |   |
|   |   |   |   |   |   y = 0.394 {19}
|   |   |   |   |
|   |   |   |   x6 >= 10.58
|   |   |   |   |
|   |   |   |   |   y = 0.689 {27}
```

Figure 1: Regression tree for mortgage data set

²³U. Johansson, H. Linusson, T. Löfström, and H. Boström, "Interpretable regression trees using conformal prediction," *Expert Syst. Appl.*, vol. 97, pp. 394–404, 2018

Example - conformal regression trees

```
x6 < 10.58
|
|   x6 < 7.25
|   |
|   |   x5 < 4.54
|   |   |
|   |   |   y = [0, 0.227] {15/0}
|   |   |   x5 >= 4.54
|   |   |   y = [0.035, 0.335] {34/0}
|   |   x6 >= 7.25
|   |   |
|   |   |   x5 < 7.195
|   |   |   |
|   |   |   |   y = [0.143, 0.443] {11/0}
|   |   |   |   x5 >= 7.195
|   |   |   |   y = [0.244, 0.544] {19/0}
|   |   x6 >= 10.58
|   |   |
|   |   |   y = [0.539, 0.839] {27/11}
```

Figure 2: Standard global ICP for mortgage. $\epsilon = 0.1$

Example - conformal regression trees

```
x6 < 10.58
|   x6 < 7.25
|   |   x5 < 4.54
|   |   |   y = [0, 0.216] {15/0}
|   |   x5 >= 4.54
|   |   |   y = [0.050, 0.320] {34/0}
|   x6 >= 7.25
|   |   x5 < 7.195
|   |   |   y = [0.157, 0.429] {11/0}
|   |   x5 >= 7.195
|   |   |   y = [0.257, 0.531] {19/0}
x6 >= 10.58
|   y = [0.539, 0.839] {27/11}
```

Figure 3: Normalized local ICP for mortgage. $\epsilon = 0.1$

Example - conformal regression trees

```
x6 < 10.58
|
|   x6 < 7.25
|   |
|   |   x5 < 4.54
|   |   |
|   |   |   y = [0.011, 0.144] {15/1}
|   |   |   x5 >= 4.54
|   |   |   y = [0.144, 0.226] {34/3}
|   |   x6 >= 7.25
|   |   |
|   |   |   x5 < 7.195
|   |   |   |
|   |   |   |   y = [0.231, 0.355] {11/1}
|   |   |   |   x5 >= 7.195
|   |   |   |   y = [0.332, 0.456] {19/1}
|   |   x6 >= 10.58
|   |   |
|   |   |   y = [0.468, 0.910] {27/2}
```

Figure 4: Mondrian ICP for mortgage. $\epsilon = 0.1$

- When a Venn-Abers predictor is applied on top of a decision tree, the number of categories is of course **dynamic**.
- But, at the same time, when the inductive variant is used, all instances falling in the same leaf will obtain the same estimate, and these estimates can be determined from the calibration set.
- The resulting model is a **fixed** decision tree, available for inspection and analysis, where each leaf contains a **specific** prediction, consisting of a label and an associated confidence (a probability interval)
- Clearly this is a very informative model.

²⁴U. Johansson, T. Löfström, and H. Boström, “Calibrating probability estimation trees using venn-abers predictors,” in *SIAM International Conference on Data Mining*. Society for Industrial and Applied Mathematics, 2019, pp. 28–36

```
x2 < 3.5
|
|   x6 < 4.5
|   |   y= Class 0 [0.9833, 1.0000] {22/0}
|   |   x6 >= 4.5
|   |   |   x3 < 1.5
|   |   |   |   y= Class 0 [0.5000, 0.6000] {1/0}
|   |   |   |   x3 >= 1.5
|   |   |   |   y= Class 1 [0.8000, 0.9091] {2/1}
|   |   x2 >= 3.5
|   |   y= Class 1 [0.8864, 0.9091] {22/2}
```

Nonconformist - conformal prediction in Python

How good is your prediction?

You want to estimate the risk of cancer recurrence in patient x_{k+1}

To your disposal, you have:

1. A set of historical observations $(x_1, y_1), \dots, (x_k, y_k)$
 - x_i describes a patient by age, tumor size, etc
 - y_i is a measurement of cancer recurrence in patient x_i
2. Some machine learning (classification or regression) algorithm
3. Conformal prediction

Motivating Example Revisited

```
import pandas as pd

breast_cancer = pd.read_csv('./data/breast-cancer.csv')

# proper training set
x_train = breast_cancer.values[:-100, :-1]
y_train = breast_cancer.values[:-100, -1]

# calibration set
x_cal = breast_cancer.values[-100:-1, :-1]
y_cal = breast_cancer.values[-100:-1, -1]

# (x_k+1, y_k+1)
x_test = breast_cancer.values[-1, :-1]
y_test = breast_cancer.values[-1, -1]

# Omitted: convert y_train, y_cal, y_test to numeric
```

Motivating Example Revisited

```
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
from nonconformist.icp import IcpClassifier
from nonconformist.nc import NcFactory

knn = KNeighborsClassifier(n_neighbors=5)
nc = NcFactory.create_nc(knn)
icp = IcpClassifier(nc)

icp.fit(x_train, y_train)
icp.calibrate(x_cal, y_cal)

print(icp.predict(np.array([x_test])), significance=0.05))
```

```
[[ True  False ]]
```

Nonconformist

Installation options:

- `git clone http://github.com/donlnz/nonconformist`
- `pip install nonconformist`

Nonconformist supports:

- Conformal classification (inductive)
- Conformal regression (inductive)
- Mondrian (e.g., class-conditional) models
- Normalization
- Aggregated conformal predictors (\approx icp ensembles)
- Out-of-bag calibration
- Plug-and-play using sklearn
- User extensions

Questions, suggestions, feedback, contributions, etc.?

`henrik.linusson@hb.se`

Research opportunities

Other scenarios for conformal prediction

- Anomaly detection with guaranteed maximum false positive rates.²⁵
- Concept drift detection / i.i.d. checking with maximum false positive rates.²⁶
- Rule extraction with guaranteed fidelity.²⁷
- Semi-supervised learning.²⁸

²⁵R. Laxhammar and G. Falkman, “Conformal prediction for distribution-independent anomaly detection in streaming vessel data,” in *Proceedings of the First International Workshop on Novel Data Stream Pattern Mining Techniques*. ACM, 2010, pp. 47–55

²⁶V. Fedorova, A. Gammerman, I. Nourtdinov, and V. Vovk, “Plug-in martingales for testing exchangeability on-line,” in *29th International Conference on Machine Learning*, 2012

²⁷U. Johansson, R. König, H. Linusson, T. Löfström, and H. Boström, “Rule extraction with guaranteed fidelity,” in *Artificial Intelligence Applications and Innovations*. Springer, 2014, pp. 281–290

²⁸X. Zhu, F.-M. Schleif, and B. Hammer, “Semi-supervised vector quantization for proximity data,” in *Proc. of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2013)*, Louvain-La-Neuve, Belgium, 2013, pp. 89–94

Suggested research opportunities

Topics

- Underlying models e.g., XGBoost
- Nonconformity functions or Venn taxonomies
- Difficulty estimators for normalized conformal regressors
- Ensembles of confidence predictors
- Multiclass and multi-label
- Applications, especially non safety-critical
- Explanation and reasoning modules utilizing confidence predictors

If you have found this tutorial interesting: Don't miss checking out the most recent framework from Vovk et al. called [Conformal predictive distributions](#).

Nonconformity functions and underlying models

- H. Papadopoulos, V. Vovk, and A. Gammerman, “Regression conformal prediction with nearest neighbours,” *Journal of Artificial Intelligence Research*, vol. 40, no. 1, pp. 815–840, 2011
- U. Johansson, H. Boström, and T. Löfström, “Conformal prediction using decision trees,” in *International Conference Data Mining (ICDM)*. IEEE, 2013
- H. Papadopoulos, “Inductive conformal prediction: Theory and application to neural networks,” *Tools in Artificial Intelligence*, vol. 18, pp. 315–330, 2008
- U. Johansson, H. Boström, T. Löfström, and H. Linusson, “Regression conformal prediction with random forests,” *Machine Learning*, vol. 97, no. 1-2, pp. 155–176, 2014
- U. Johansson, H. Linusson, T. Löfström, and H. Boström, “Interpretable regression trees using conformal prediction,” *Expert Syst. Appl.*, vol. 97, pp. 394–404, 2018

Combined conformal predictors

- V. Vovk, “Cross-conformal predictors,” *Annals of Mathematics and Artificial Intelligence*, pp. 1–20, 2013
- L. Carlsson, M. Eklund, and U. Norinder, “Aggregated conformal prediction,” in *Artificial Intelligence Applications and Innovations*. Springer, 2014, pp. 231–240
- H. Papadopoulos, “Cross-conformal prediction with ridge regression,” in *Statistical Learning and Data Sciences*. Springer, 2015, pp. 260–270

Not (yet) proven valid

But seems to be working well in practice.

Application domains

- A. Lambrou, H. Papadopoulos, E. Kyriacou, C. S. Pattichis, M. S. Pattichis, A. Gammerman, and A. Nicolaides, “Assessment of stroke risk based on morphological ultrasound image analysis with conformal prediction,” in *Artificial Intelligence Applications and Innovations*. Springer, 2010, pp. 146–153
- D. Devetyarov, I. Nouretdinov, B. Burford, S. Camuzeaux, A. Gentry-Maharaj, A. Tiss, C. Smith, Z. Luo, A. Chervonenkis, R. Hallett *et al.*, “Conformal predictors in early diagnostics of ovarian and breast cancers,” *Progress in Artificial Intelligence*, vol. 1, no. 3, pp. 245–257, 2012
- M. Eklund, U. Norinder, S. Boyer, and L. Carlsson, “The application of conformal prediction to the drug discovery process,” *Annals of Mathematics and Artificial Intelligence*, vol. 74, no. 1-2, pp. 117–132, 2015

Application domains

- I. Nourtdinov, S. G. Costafreda, A. Gammerman, A. Chervonenkis, V. Vovk, V. Vapnik, and C. H. Fu, “Machine learning classification with confidence: application of transductive conformal predictors to mri-based diagnostic and prognostic markers in depression,” *Neuroimage*, vol. 56, no. 2, pp. 809–813, 2011
- J. Vega, A. Murari, S. Dormido-Canto, R. Moreno, A. Pereira, A. Acero, and J.-E. Contributors, “Adaptive high learning rate probabilistic disruption predictors from scratch for the next generation of tokamaks,” *Nuclear Fusion*, vol. 54, no. 12, p. 123001, 2014

Venn predictors

- H. Papadopoulos, “Reliable probabilistic classification with neural networks,” *Neurocomputing*, vol. 107, no. Supplement C, pp. 59 – 68, 2013
- A. Lambrou, I. Nourtdinov, and H. Papadopoulos, “Inductive venn prediction,” *Annals of Mathematics and Artificial Intelligence*, vol. 74, no. 1, pp. 181–201, 2015
- V. Vovk and I. Petej, “Venn-abers predictors,” *arXiv preprint arXiv:1211.0025*, 2012
- U. Johansson, T. Löfström, H. Sundell, H. Linusson, A. Gidenstam, and H. Boström, “Venn predictors for well-calibrated probability estimation trees,” in *Seventh Symposium on Conformal and Probabilistic Prediction with Applications*, ser. Proceedings of Machine Learning Research, vol. 91. PMLR, 2018, pp. 1–12

Suggested reading







- V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005
- www.alrw.net
- G. Shafer and V. Vovk, “A tutorial on conformal prediction,” *The Journal of Machine Learning Research*, vol. 9, pp. 371–421, 2008
- A. Gammerman, V. Vovk, and V. Vapnik, “Learning by transduction,” in *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1998, pp. 148–155
- A. Gammerman and V. Vovk, “Hedging predictions in machine learning the second computer journal lecture,” *The Computer Journal*, vol. 50, no. 2, pp. 151–163, 2007







Suggested reading cont.







- H. Papadopoulos, K. Proedrou, V. Vovk, and A. Gammerman, “Inductive confidence machines for regression,” in *Machine Learning: ECML 2002*. Springer, 2002, pp. 345–356
- H. Papadopoulos and H. Haralambous, “Reliable prediction intervals with regression neural networks,” *Neural Networks*, vol. 24, no. 8, pp. 842–851, 2011
- U. Johansson, H. Boström, T. Löfström, and H. Linusson, “Regression conformal prediction with random forests,” *Machine Learning*, vol. 97, no. 1-2, pp. 155–176, 2014




Questions?








References





-  V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005.
-  I. Nourtdinov, V. Vovk, M. Vyugin, and A. Gammerman, “Pattern recognition and density estimation under the general i.i.d. assumption,” in *Computational Learning Theory*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2001, vol. 2111, pp. 337–353.
-  H. Papadopoulos, V. Vovk, and A. Gammerman, “Regression conformal prediction with nearest neighbours,” *Journal of Artificial Intelligence Research*, vol. 40, no. 1, pp. 815–840, 2011.
-  U. Johansson, H. Boström, T. Löfström, and H. Linusson, “Regression conformal prediction with random forests,” *Machine Learning*, vol. 97, no. 1-2, pp. 155–176, 2014.
-  H. Boström, H. Linusson, T. Löfström, and U. Johansson, “Accelerating difficulty estimation for conformal regression forests,” *Annals of Mathematics and Artificial Intelligence*, pp. 1–20, 2017.
-  U. Johansson, H. Linusson, T. Löfström, and H. Boström, “Interpretable regression trees using conformal prediction,” *Expert Syst. Appl.*, vol. 97, pp. 394–404, 2018.






-  V. Vovk, V. Fedorova, I. Nourtdinov, and A. Gammerman, “Criteria of efficiency for conformal prediction,” 2014.
-  H. Linusson, U. Johansson, H. Boström, and T. Löfström, “Efficiency comparison of unstable transductive and inductive conformal classifiers,” in *Artificial Intelligence Applications and Innovations*. Springer, 2014, pp. 261–270.
-  L. Carlsson, E. Ahlberg, H. Boström, U. Johansson, and H. Linusson, “Modifications to p-values of conformal predictors,” in *Statistical Learning and Data Sciences*. Springer, 2015, pp. 251–259.
-  U. Johansson, E. Ahlberg, H. Boström, L. Carlsson, H. Linusson, and C. Sönströd, “Handling small calibration sets in mondrian inductive conformal regressors,” in *Statistical Learning and Data Sciences*. Springer, 2015, pp. 271–280.
-  A. Niculescu-Mizil and R. Caruana, “Predicting good probabilities with supervised learning,” in *ICML*. ACM, 2005, pp. 625–632.
-  F. Provost and P. Domingos, “Tree induction for probability-based ranking,” *Mach. Learn.*, vol. 52, no. 3, pp. 199–215, 2003.






-  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*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 2017, pp. 1321–1330.
-  U. Johansson and P. Gabrielsson, “Are traditional neural networks well-calibrated?” in *IJCNN*, 2019, In Press.
-  V. Vovk, G. Shafer, and I. Nourtdinov, “Self-calibrating probability forecasting,” in *Advances in Neural Information Processing Systems*, 2004, pp. 1133–1140.
-  V. Vovk and I. Petej, “Venn-abers predictors,” *arXiv preprint arXiv:1211.0025*, 2012.
-  U. Johansson, T. Löfström, and H. Boström, “Calibrating probability estimation trees using venn-abers predictors,” in *SIAM International Conference on Data Mining*. Society for Industrial and Applied Mathematics, 2019, pp. 28–36.
-  M. T. Ribeiro, S. Singh, and C. Guestrin, “Why should i trust you?: Explaining the predictions of any classifier,” in *Proceedings of the 22Nd ACM SIGKDD*, ser. KDD ’16. New York, NY, USA: ACM, 2016, pp. 1135–1144.

-  N. Diakopoulos, S. Friedler, M. Arenas, S. Barocas, M. Hay, B. Howe, H. V. Jagadish, K. Unsworth, A. Sahuguet, S. Venkatasubramanian, C. Wilson, and B. Z. C. Yu, *Principles for Accountable Algorithms and a Social Impact Statement for Algorithms*, FAT/ML, 2017.
-  R. Laxhammar and G. Falkman, “Conformal prediction for distribution-independent anomaly detection in streaming vessel data,” in *Proceedings of the First International Workshop on Novel Data Stream Pattern Mining Techniques*. ACM, 2010, pp. 47–55.
-  V. Fedorova, A. Gammernan, I. Nouretdinov, and V. Vovk, “Plug-in martingales for testing exchangeability on-line,” in *29th International Conference on Machine Learning*, 2012.
-  U. Johansson, R. König, H. Linusson, T. Löfström, and H. Boström, “Rule extraction with guaranteed fidelity,” in *Artificial Intelligence Applications and Innovations*. Springer, 2014, pp. 281–290.

-  X. Zhu, F.-M. Schleif, and B. Hammer, “Semi-supervised vector quantization for proximity data,” in *Proc. of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2013)*, Louvain-La-Neuve, Belgium, 2013, pp. 89–94.
-  U. Johansson, H. Boström, and T. Löfström, “Conformal prediction using decision trees,” in *International Conference Data Mining (ICDM)*. IEEE, 2013.
-  H. Papadopoulos, “Inductive conformal prediction: Theory and application to neural networks,” *Tools in Artificial Intelligence*, vol. 18, pp. 315–330, 2008.
-  U. Johansson, H. Boström, T. Löfström, and H. Linusson, “Regression conformal prediction with random forests,” *Machine Learning*, vol. 97, no. 1-2, pp. 155–176, 2014.
-  V. Vovk, “Cross-conformal predictors,” *Annals of Mathematics and Artificial Intelligence*, pp. 1–20, 2013.
-  L. Carlsson, M. Eklund, and U. Norinder, “Aggregated conformal prediction,” in *Artificial Intelligence Applications and Innovations*. Springer, 2014, pp. 231–240.
-  H. Papadopoulos, “Cross-conformal prediction with ridge regression,” in *Statistical Learning and Data Sciences*. Springer, 2015, pp. 260–270.

-  A. Lambrou, H. Papadopoulos, E. Kyriacou, C. S. Pattichis, M. S. Pattichis, A. Gammerman, and A. Nicolaides, “Assessment of stroke risk based on morphological ultrasound image analysis with conformal prediction,” in *Artificial Intelligence Applications and Innovations*. Springer, 2010, pp. 146–153.
-  D. Devetyarov, I. Nourtdinov, B. Burford, S. Camuzeaux, A. Gentry-Maharaj, A. Tiss, C. Smith, Z. Luo, A. Chervonenkis, R. Hallett *et al.*, “Conformal predictors in early diagnostics of ovarian and breast cancers,” *Progress in Artificial Intelligence*, vol. 1, no. 3, pp. 245–257, 2012.
-  M. Eklund, U. Norinder, S. Boyer, and L. Carlsson, “The application of conformal prediction to the drug discovery process,” *Annals of Mathematics and Artificial Intelligence*, vol. 74, no. 1-2, pp. 117–132, 2015.
-  I. Nourtdinov, S. G. Costafreda, A. Gammerman, A. Chervonenkis, V. Vovk, V. Vapnik, and C. H. Fu, “Machine learning classification with confidence: application of transductive conformal predictors to mri-based diagnostic and prognostic markers in depression,” *Neuroimage*, vol. 56, no. 2, pp. 809–813, 2011.

-  J. Vega, A. Murari, S. Dormido-Canto, R. Moreno, A. Pereira, A. Acero, and J.-E. Contributors, “Adaptive high learning rate probabilistic disruption predictors from scratch for the next generation of tokamaks,” *Nuclear Fusion*, vol. 54, no. 12, p. 123001, 2014.
-  H. Papadopoulos, “Reliable probabilistic classification with neural networks,” *Neurocomputing*, vol. 107, no. Supplement C, pp. 59 – 68, 2013.
-  A. Lambrou, I. Nouredinov, and H. Papadopoulos, “Inductive venn prediction,” *Annals of Mathematics and Artificial Intelligence*, vol. 74, no. 1, pp. 181–201, 2015.
-  U. Johansson, T. Löfström, H. Sundell, H. Linusson, A. Gidenstam, and H. Boström, “Venn predictors for well-calibrated probability estimation trees,” in *Seventh Symposium on Conformal and Probabilistic Prediction with Applications*, ser. Proceedings of Machine Learning Research, vol. 91. PMLR, 2018, pp. 1–12.
-  G. Shafer and V. Vovk, “A tutorial on conformal prediction,” *The Journal of Machine Learning Research*, vol. 9, pp. 371–421, 2008.

-  A. Gammerman, V. Vovk, and V. Vapnik, “Learning by transduction,” in *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1998, pp. 148–155.
-  A. Gammerman and V. Vovk, “Hedging predictions in machine learning the second computer journal lecture,” *The Computer Journal*, vol. 50, no. 2, pp. 151–163, 2007.
-  H. Papadopoulos, K. Proedrou, V. Vovk, and A. Gammerman, “Inductive confidence machines for regression,” in *Machine Learning: ECML 2002*. Springer, 2002, pp. 345–356.
-  H. Papadopoulos and H. Haralambous, “Reliable prediction intervals with regression neural networks,” *Neural Networks*, vol. 24, no. 8, pp. 842–851, 2011.
-  V. Vovk, “Conditional validity of inductive conformal predictors,” *Journal of Machine Learning Research - Proceedings Track*, vol. 25, pp. 475–490, 2012.

Conformal classification - some details

Inductive Conformal Classification

Divide the training set Z into two disjoint subsets

A proper training set Z_t

A calibration set Z_c where $|Z_c| = q$

Fit a model h using Z_t

This is the underlying model

Choose an $f(z)$, e.g. $f(z_i) = 1 - \hat{P}_h(y_i | x_i)$

This is the nonconformity function

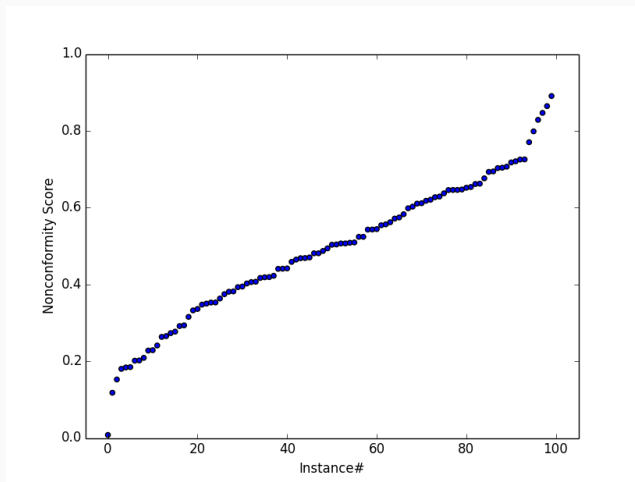
Apply $f(Z)$ to $\forall z_i \in Z_c$

Save these calibration scores

We denote these $\alpha_1, \dots, \alpha_q$

Inductive Conformal Classification

Apply $f(z)$ to Z_C , and obtain a set of calibration scores $\alpha_1, \dots, \alpha_q$



Inductive Conformal Classification

For each $\tilde{y} \in Y$

Let $\alpha_i^{\tilde{y}} = f[(x_i, \tilde{y})]$

Calculate

$$p_i^{\tilde{y}} = \frac{\left| \left\{ z_j \in Z_c : \alpha_j > \alpha_i^{\tilde{y}} \right\} \right|}{q+1} + \theta \frac{\left| \left\{ z_j \in Z_c : \alpha_j = \alpha_i^{\tilde{y}} \right\} \right| + 1}{q+1}, \theta \sim U[0, 1]$$

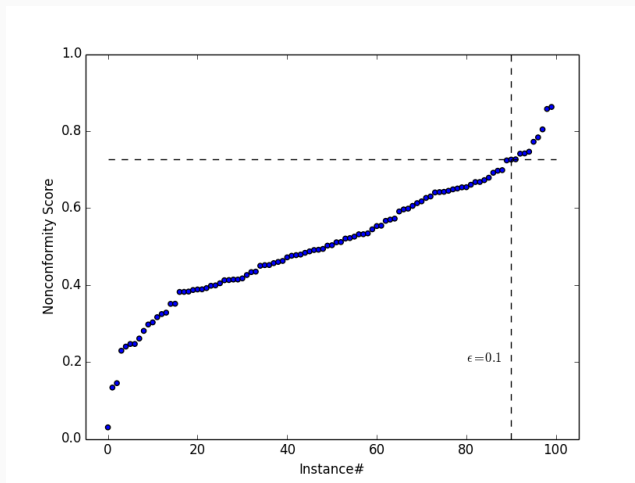
Fix a significance level $\epsilon \in (0, 1)$

Prediction region

$$\Gamma_i^\epsilon = \left\{ \tilde{y} \in Y : p_i^{\tilde{y}} > \epsilon \right\}$$

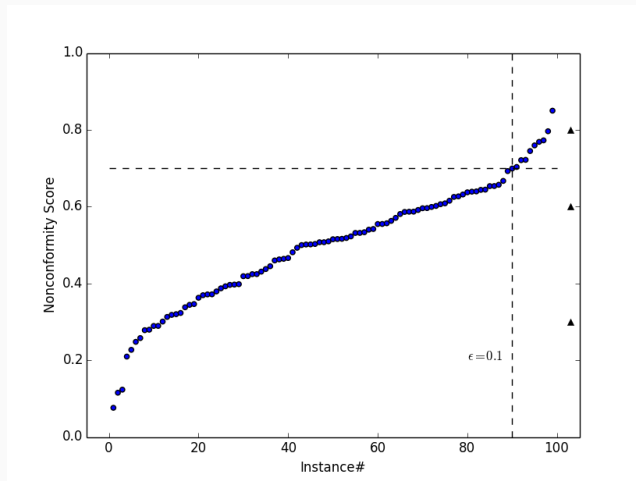
Inductive Conformal Classification

Choose a significance level ϵ



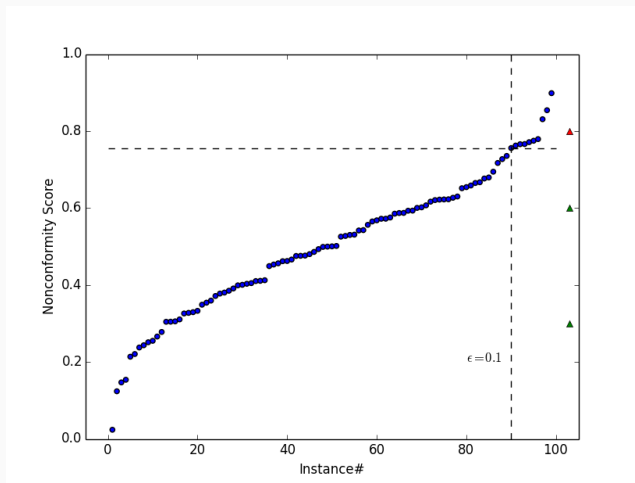
Inductive Conformal Classification

Obtain α_i using $f(z)$ for each possible class $(x_i, \tilde{y}_1), (x_i, \tilde{y}_2), (x_i, \tilde{y}_3), \dots$, resulting in $\alpha_i^{\tilde{y}_1}, \alpha_i^{\tilde{y}_2}, \alpha_i^{\tilde{y}_3}, \dots$



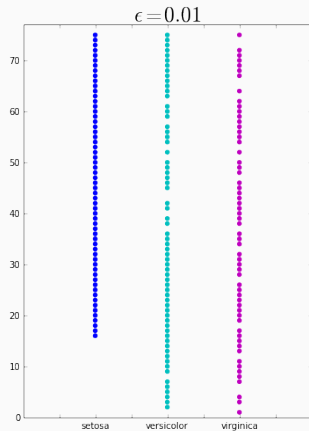
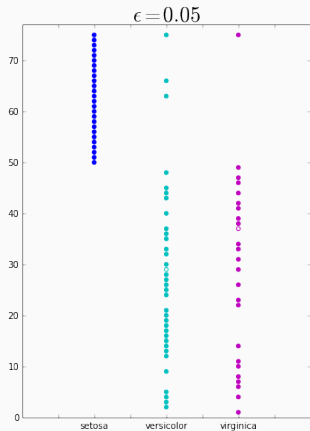
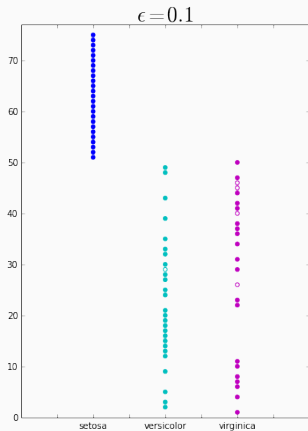
Inductive Conformal Classification

Reject/include based on the p -value statistic, and the chosen ϵ



Inductive Conformal Classification

Iris, Random Forest



Conformal predictors are, by default, unconditional

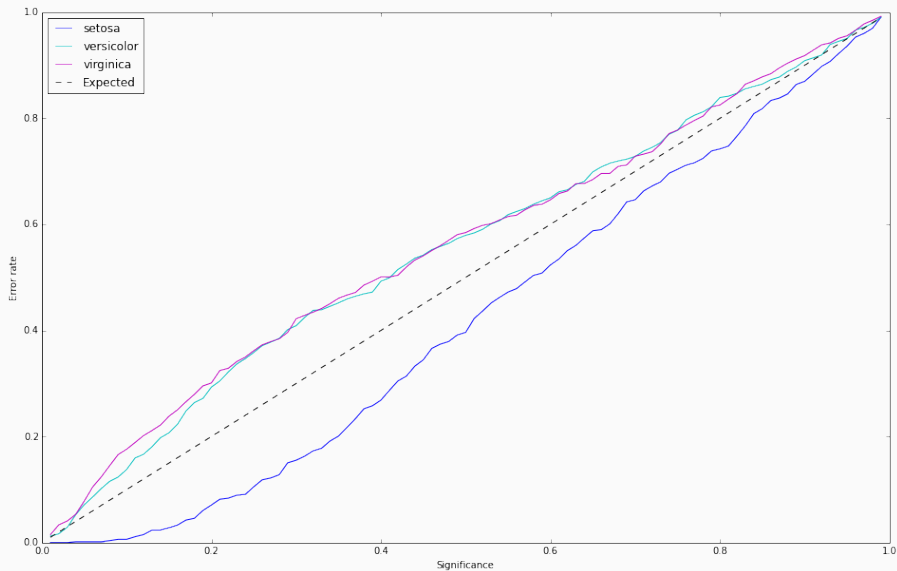
Their guaranteed error rate applies to the entire test set.

- Difficult patterns (e.g. minority class) may see a greater error rate than expected
- Easy patterns (e.g. majority class) may see a smaller error rate than expected

Example — Iris data set

- One linearly separable class (easy)
- Two linearly non-separable classes (difficult)

Conditional conformal prediction



Conditional conformal predictors²⁹ help solve this by

Dividing the problem space into several disjoint subspaces

- e.g. let each class represent a subspace, or
- define subspace based on some input variable(s) (age, gender, etc.)

Guaranteeing an error rate at most ϵ for each subspace

²⁹V. Vovk, “Conditional validity of inductive conformal predictors,” *Journal of Machine Learning Research - Proceedings Track*, vol. 25, pp. 475–490, 2012

Conditional conformal prediction

Define a mapping function $K(z_i) = \kappa_i$

Examples

$$K(z_i) = y_i \quad (5)$$

$$K(z_i) = \begin{cases} 1 & \text{if } x_{i,1} < 50 \\ 2 & \text{if } 50 \leq x_{i,1} < 100 \\ 3 & \text{otherwise} \end{cases} \quad (6)$$

Conditional p -value

$$p_i^{\tilde{y}} = \frac{|\{z_j \in Z_c : \alpha_j > \alpha_i^{\tilde{y}}\} \wedge K(z_i) = K(z_j)|}{|K(z_i) = K(z_j)| + 1} + \theta \frac{|\{z_j \in Z_c : \alpha_j = \alpha_i^{\tilde{y}}\} \wedge K(z_i) = K(z_j)|}{|K(z_i) = K(z_j)| + 1}, \theta \sim U[0, 1]$$

Conditional conformal prediction

