# Conformal prediction

Pierre Geurts

Institut Montefiore, University of Liège, Belgium

INFO8004
Advanced Machine Learning
February 18, 2021

# Outline

❶ Introduction

❷ Conformal prediction

❸ Inductive conformal prediction

❹ Two recent papers

❺ Discussion and links

# Motivation

Predictions obtained from supervised learning models are uncertain for several reasons:

- Noise uncertainty: output for a given input is not perfectly known due to noise.
- Sampling uncertainty: we use empirical distribution as a proxy for real distribution.
- Modeling uncertainty: our assumptions may not be correct.

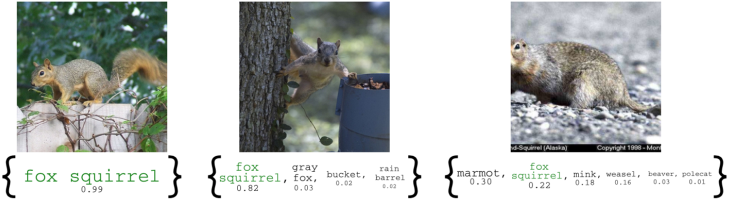
Measuring uncertainty is important to characterize the quality of a given prediction and take informed decisions.

# Quantifying prediction uncertainty

One way to quantify uncertainty is to provide **set predictions** instead of (or in addition to) the usual pointwise predictions.

- E.g.: a confidence interval in regression or a set of classes in classification.

$$x \in \mathcal{X} \to y \in \mathcal{Y} \quad \Rightarrow \quad x \in \mathcal{X} \to \hat{C}(x) \subseteq \mathcal{Y}$$


{ fox squirrel 0.99 }
{ fox squirrel 0.82, gray fox 0.03, bucket 0.02, rain barrel 0.02 }
{ marmot 0.30, fox squirrel 0.22, mink 0.18, weasel 0.16, beaver 0.03, polecat 0.01 }
d-Squirrel (Alaska)
Copyright 1998 - Mon

(Angelopoulos et al., ICLR, 2021)

# Quantifying prediction uncertainty

One way to quantify uncertainty is to provide **set predictions** instead of (or in addition to) the usual pointwise predictions.

- E.g.: a confidence interval in regression or a set of classes in classification.

$$x \in \mathcal{X} \to y \in \mathcal{Y} \quad \Rightarrow \quad x \in \mathcal{X} \to \hat{C}(x) \subseteq \mathcal{Y}$$

We want such sets to be both:

- **Valid**: the *probability* that the true output of the test example $x$ falls in $\hat{C}(x)$ should be as high as possible
- **Informative**: these sets should be as small as possible, $|\hat{C}(x)| \ll$.

# Some approaches to measure uncertainty

Focusing on the regression setting, several approaches have been explored to obtain predictions in the form of confidence intervals:

- Estimating prediction variance
- Bayesian approaches
- Ensemble methods
- Quantile regression
- **Conformal prediction**
- ...

# Conformal prediction

**Conformal prediction** is a general approach to get set-wise predictions with validity guarantees even in finite sample size settings, with very few assumptions.

This is in contrast with most other techniques that are correct only asymptotically and/or when modeling assumptions are correct.

Theory can be tricky but the final (inductive) method is very intuitive, trivial to implement and it can furthermore be combined with other uncertainty estimation techniques.

Invented by Vovk and his colleagues at the end of the nineties, re-invented and analysed later by statisticians (under the name "distribution free inference"). See references at the end.

# Outline

1. Introduction

2. Conformal prediction
   - Prediction without inputs
   - Supervised learning

3. Inductive conformal prediction

4. Two recent papers

5. Discussion and links

# A simple prediction problem

Let us assume a sequence of $n$ values $z_1, \ldots, z_n$ each drawn independently from the same (unknown) distribution:

$17, 20, 10, 17, 12, 15, 19, 22, 17, 19, 14, 22, 18, 17, 13, 12, 18, 15, 17, z_{n+1}?$

We want to predict $z_{n+1}$.

A natural prediction is the sample average $\bar{z}_n = \frac{1}{n} \sum_{i=1}^{n} z_i$, in our case 16.53.

How good is this prediction? How to get a confidence interval instead?

# The parametric approach

Denoting by $s_n$ the sample variance:

$$s_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}(z_i - \bar{z}_n)^2,$$

and assuming that the underlying distribution is **gaussian**, one can show that:

$$\frac{z_{n+1} - \bar{z}_n}{s_n}\sqrt{\frac{n}{n+1}}$$

follows a $t$-distribution with $n-1$ degrees of freedom.

One can thus derive the following 95% confidence interval for $z_{n+1}$:

$$\bar{z}_{n+1} \pm t_{n-1}^{0.025} s_n \sqrt{\frac{n+1}{n}}.$$

For our example, $s_n = 3.31$ and therefore the 95% confidence interval is $[9.4, 24.13]$. The latter is however correct only when the distribution is truly gaussian.

# Conformal prediction: a non parametric approach

From the training set $z^n = \{z_1, \dots, z_n\}$, we want to find a set of values

$$\hat{C}(z^n) \subseteq IR$$

such that $P(Z_{n+1} \in \hat{C}(Z^n)) \geq 1 - \epsilon$ for some user-defined risk $\epsilon$.

The general idea of CP to construct $\hat{C}(z^n)$ is to consider every possible values $z \in \mathcal{Z}$ as candidate for $z_{n+1}$ and decide whether it should be included in $\hat{C}(z^n)$ using a (non parametric) statistical test.

Main idea of the test:

- Define a **nonconformity score** that quantifies how *unlikely* is $z$ with respect to $z^n$
- Compute a ***p*-value** that quantifies how likely $z$'s score is with respect to other $z_i$'s scores.
- Include $z$ if the $p$-value is not too small.

# Nonconformity score

The **nonconformity** score $A(z^n, z) \in IR$ measures how different $z \in \mathcal{Z}$ is from the examples in sample $z^n$.

Besides measuring nonconformity, the only condition on $A$ is that it should consider the examples in $z^n$ as **exchangeable**. I.e., the value of $A$ should not change if examples in $z^n$ are permuted.

In our example, we can use the absolute deviation of $z$ from the average of all sampled values, including $z$:

$$A(z^n, z) = \left| \frac{\sum_{i=1}^{n+1} z_i + z}{n+1} - z \right| = \left| \frac{314 + z}{20} - z \right|$$

$17, 20, 10, 17, 12, 15, 19, 22, 17, 19, 14, 22, 18, 17, 13, 12, 18, 15, 17, z?$

# Statistical test

For a given $z \in \mathcal{Z}$ and a user-defined risk $\epsilon$:

1. Compute $\alpha_{n+1} = A(z^n, z)$
2. Compute nonconformity scores $\alpha_i$ for all samples $z_i, i = 1, \ldots, n$:

$$\alpha_i = A(z^n \cup \{z\} \setminus \{z_i\}, z_i)$$

3. Derive a $p$-value for $z$:

$$p_{z^n}(z) = \frac{\#\{i = 1, \ldots, n+1 | \alpha_i \geq \alpha_{n+1}\}}{n+1}$$

4. Include $z$ in $\hat{C}(z^n)$ if $p_{z^n}(z) > \epsilon$.

# Statistical test: illustration

$$17, 20, 10, 17, 12, 15, 19, 22, 17, 19, 14, 22, 18, 17, 13, 12, 18, 15, 17, z?$$

$\Rightarrow \alpha_{n+1} = \left|\frac{314+z}{20} - z\right|$, $\alpha_i = \left|\frac{314+z}{20} - z_i\right| = \frac{1}{20}\left|314 + z - 20z_i\right|$

Exemple: If $\epsilon = 0.05$ et $z = 15$, $\alpha_{n+1} = 5.25$, $\{\alpha_1, \ldots, \alpha_{20}\} =$

$0.75, 3.75, 6.25, 0.75, 4.25, 1.25, 2.75, 5.75, 0.75, 2.75, 2.25, 5.75, 1.75, 0.75, 3.25, 4.25, 1.75, 1.25, 0.75, 5.25$

$\Rightarrow p_{z^n}(15) = \frac{4}{20} = 0.2 \Rightarrow 15 \in \hat{C}(z^n)$

# Statistical test: illustration

$$17, 20, 10, 17, 12, 15, 19, 22, 17, 19, 14, 22, 18, 17, 13, 12, 18, 15, 17, z?$$

$\Rightarrow \alpha_{n+1} = \left|\frac{314+z}{20} - z\right|, \alpha_i = \left|\frac{314+z}{20} - z_i\right| = \frac{1}{20}\left|314 + z - 20z_i\right|$

Exemple: If $\epsilon = 0.05$ et $z = 15$, $\alpha_{n+1} = 5.25$, $\{\alpha_1, \ldots, \alpha_{20}\} =$

$0.75, 3.75, 6.25, 0.75, 4.25, 1.25, 2.75, 5.75, 0.75, 2.75, 2.25, 5.75, 1.75, 0.75, 3.25, 4.25, 1.75, 1.25, 0.75, 5.25$

$\Rightarrow p_{z^n}(15) = \frac{4}{20} = 0.2 \Rightarrow 15 \in \hat{C}(z^n)$

**Computing the confidence interval:**

For $z$ not to be excluded, the corresponding $\alpha_{n+1}$ should not be larger than all other $\alpha_i$ (otherwise the $p$-value would be lower than $\frac{1}{20}$).

For a given $z$, $|314 + z - 20z_i|$ can only take its maximum values for the largest (22) or the smallest (10) of the $z_i$s.

We thus have that $z$ is included iff:

$$|314 - 19z| \leq \max\{|314 + z - (20 \times 22)|, |314 + z - (20 \times 10)|\},$$

which leads to $\hat{C}(z^n) = [10, 23.8]$.

# A non parametric approach: validity

Is the produced prediction set valid in any sense? Yes!

**Theorem.** *If* $Z_i, i = 1, \ldots, n+1$ *are i.i.d., then*

$$P(Z_{n+1} \in \hat{C}(Z^n)) \geq 1 - \epsilon.$$

In words: if we draw $n+1$ samples $z_i$ independently from the same distribution $D$, then the probability that the $n+1$th samples will belong to the set $\hat{C}(z^n)$ as constructed by the CP algorithm using the first $n$ samples is greater than $1-\epsilon$.

**This is true, whatever $n$, $A$ and $D$.**

# Proof

$P(Z_{n+1} \notin \hat{C}(Z^n))$ is the probability that $Z_{n+1}$ does not belong to $\hat{C}(Z^n)$.

If $Z_{n+1}$ does not belong to $\hat{C}(Z^n)$, it means that $p_{Z^n}(Z_{n+1}) \leq \epsilon$ (by definition of $\hat{C}(Z^n)$).

The probability that $p_{Z^n}(Z_{n+1}) \leq \epsilon$ is the probability that $Z_{n+1}$'s nonconformity score is at least the $\lfloor (n+1)\epsilon \rfloor$th largest among all examples.

Given that the $Z_i$s, and therefore the $\alpha_i$s, are i.i.d., this event has a probability no larger than $\epsilon$ to occur.

20% 20% 20% 20% 20%

$\alpha_1$ $\alpha_2$ $\alpha_3$ $\alpha_4$

# Some thoughts

- If the set of possible outputs is large (infinite), computing $C$ might be difficult depending on the nonconformity score chosen.
- Coverage guarantee (validity) does not depend on the nonconformity score but different nonconformity scores will lead to prediction sets of different sizes.
- The approach is insensitive to a monotone transformation of the score.
  - E.g., taking the square of the deviation or not including the example in the mean does not affect the predictions.
- CP remains valid in an online setting: if you apply the procedure iteratively for increasing $n$, the error rate that you will obtain over time will never exceed $\epsilon$.

# Outline

1 Introduction

2 Conformal prediction
- Prediction without inputs
- Supervised learning

3 Inductive conformal prediction

4 Two recent papers

5 Discussion and links

# Conformal prediction: supervised learning

The exact same idea can be applied to address supervised learning problems.

Inputs:

- A training set of input-output pairs:

$$z^n = \{(x_1, y_1), \ldots, (x_n, y_n)\} \subseteq \mathcal{X} \times \mathcal{Y}$$

  and a new unlabeled test example $x_{n+1}$, all iid from the same distribution
- A significance level $\epsilon$

Output:

- A prediction set $\hat{C}(x_{n+1}; z^n) \subseteq \mathcal{Y}$ which is valid at the $1 - \epsilon$ level.

# The conformal algorithm (transductive setting)

Let us define a nonconformity score function $A(z^n, (x, y)) \in IR$ that quantifies how likely the pair $(x, y)$ is with respect to the training sample $z^n$.

Then, to decide whether to include $y \in \mathcal{Y}$ in $\hat{C}(x_{n+1}; z^n)$

1. Set $(x_{n+1}, y_{n+1}) = (x_{n+1}, y)$ and $z^{n+1} = z^n \cup \{(x_{n+1}, y_{n+1})\}$
2. For $i = 1, \ldots, n+1$, compute $\alpha_i = A(z^{n+1} \setminus \{(x_i, y_i)\}, (x_i, y_i))$
3. Compute
$$p_{z^n, x_{n+1}}(y) = \frac{\#\{i = 1, \ldots, n+1 | \alpha_i \geq \alpha_n\}}{n+1}$$
4. Include $y$ in $\hat{C}(x_{n+1}; z^n)$ if and only if $p_{z^n, x_{n+1}}(y) > \epsilon$.

# Which nonconformity function?

One needs to define a function $A(z^n, (x, y))$ which measures how unlikely the pair $(x, y)$ is with respect to the set $z^n$.

Common choice:

$$A(z^n, (x, y)) = \ell(h_{z^n}(x), y),$$

where

- $h_{z^n}$ is a model trained on $z^n$ with a given learning algorithm, called the **underlying algorithm**,
- $\ell : \mathcal{Y} \times \mathcal{Y} \to I\!R$ is a loss function.

Notes:

- Usually, the better the underlying algorithm, the smaller the prediction set.
- Model $h$ can be trained with the example $(x, y)$ (to reduce computing times).

# Nonconformity functions: a few examples

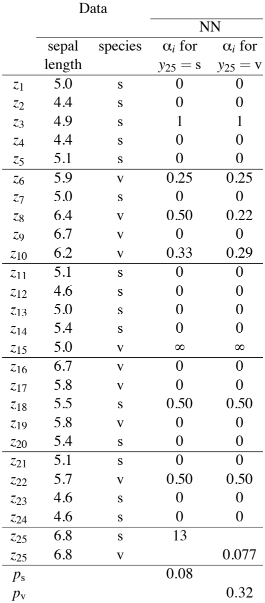
- One minus the probability estimate for the correct class
- Classification margin: probability of the best other class minus probability of correct class
- Distance to neighbors of the same class
- Absolute error of a regression model
- A random number in $[0, 1]$ (useless but valid)
- ...(more example later)

# An example in classification (Shafer and Vovk, 2008)

- Iris dataset, one feature, two classes, 24 training examples
- $A(z^n, (x, y))$ is the ratio:

$$\frac{\text{distance to } x_i\text{'s NN in } z^n \text{ with same label}}{\text{distance to } x_i\text{'s NN in } z^n \text{ with different label}}$$

- Prediction set (at $\epsilon = 0.1$) is singleton $\{v\}$



| | Data | | NN | |
|---|---|---|---|---|
| | sepal length | species | $\alpha_i$ for $y_{25} = \text{s}$ | $\alpha_i$ for $y_{25} = \text{v}$ |
| $z_1$ | 5.0 | s | 0 | 0 |
| $z_2$ | 4.4 | s | 0 | 0 |
| $z_3$ | 4.9 | s | 1 | 1 |
| $z_4$ | 4.4 | s | 0 | 0 |
| $z_5$ | 5.1 | s | 0 | 0 |
| $z_6$ | 5.9 | v | 0.25 | 0.25 |
| $z_7$ | 5.0 | s | 0 | 0 |
| $z_8$ | 6.4 | v | 0.50 | 0.22 |
| $z_9$ | 6.7 | v | 0 | 0 |
| $z_{10}$ | 6.2 | v | 0.33 | 0.29 |
| $z_{11}$ | 5.1 | s | 0 | 0 |
| $z_{12}$ | 4.6 | s | 0 | 0 |
| $z_{13}$ | 5.0 | s | 0 | 0 |
| $z_{14}$ | 5.4 | s | 0 | 0 |
| $z_{15}$ | 5.0 | v | $\infty$ | $\infty$ |
| $z_{16}$ | 6.7 | v | 0 | 0 |
| $z_{17}$ | 5.8 | v | 0 | 0 |
| $z_{18}$ | 5.5 | s | 0.50 | 0.50 |
| $z_{19}$ | 5.8 | v | 0 | 0 |
| $z_{20}$ | 5.4 | s | 0 | 0 |
| $z_{21}$ | 5.1 | s | 0 | 0 |
| $z_{22}$ | 5.7 | v | 0.50 | 0.50 |
| $z_{23}$ | 4.6 | s | 0 | 0 |
| $z_{24}$ | 4.6 | s | 0 | 0 |
| $z_{25}$ | 6.8 | s | 13 | |
| $z_{25}$ | 6.8 | v | | 0.077 |
| $p_\text{s}$ | | | 0.08 | |
| $p_\text{v}$ | | | | 0.32 |

# Validity

**Theorem:** *If training data $z^n$ and test example $(x_{n+1}, y_{n+1})$ are iid, then we have:*

$$P(Y_{n+1} \in C(X_{n+1}; Z^n)) \geq 1 - \epsilon.$$

Proof is similar to the "without inputs" setting.

The probability is over the training set as well as the test example $(X_{n+1}, Y_{n+1})$ (and remains valid in an online setting).

This is called **marginal** coverage (see later for a discussion).

# Outline

1. Introduction

2. Conformal prediction

3. Inductive conformal prediction
   - General algorithm
   - Classification
   - Regression

4. Two recent papers

5. Discussion and links

# Inductive conformal prediction

The conformal algorithm presented so far is **transductive**:

- This is the original conformal prediction approach
- No model is retained and all computations need to be redone for each new test example.
- For regression problems, it is sometimes not trivial to compute the prediction set

A more practical setting is provided by **inductive** (or split) conformal prediction (ICP):

- Requires to train a single model only
- Much more computationally efficient and flexible
- Requires that some data is set aside for calibration, which adds randomness and reduces sample size for model fitting.

# Inductive conformal prediction

- Divide the training set $z^n$ into two disjoint subsets
  - A proper training set $z^l$
  - A calibration set $z^q$ (with $l + q = n$)
- Fit a model $h_{z^l}$ on $z^l$ using the underlying algorithm.
- Choose a nonconformity function $A(h, (x, y)) \in IR$
- Apply $A$, using $h_{z^l}$, to all pairs $(x_i, y_i) \in z^q$ to get the scores:

$$\alpha_1, \ldots, \alpha_q$$

- For all $y \in \mathcal{Y}$:
  - Compute $\alpha_{n+1} = A(h_{z^l}, (x_{n+1}, y))$ and

$$p_{z^n, x_{n+1}}(y) = \frac{\#\{i = 1, \ldots, q, n+1 | \alpha_i \geq \alpha_{n+1}\}}{q+1}$$

  - Include $y$ in the prediction set $\hat{C}(x_{n+1}; z^n)$ if and only if $p_{z^n, x_{n+1}}(y) > \epsilon$

# Inductive conformal prediction: validity

Validity is preserved

> **Theorem:** *If calibration examples $Z^q$ and test example $(X_{n+1}, Y_{n+1})$ are i.i.d., then we have:*
>
> $$P(Y_{n+1} \in \hat{C}(X_{n+1}; Z^n)) \geq 1 - \epsilon.$$

This is still marginal coverage, ie., probability is over (full) training data and test example. We however also have that:

$$P(Y_{n+1} \in \hat{C}(X_{n+1}; z^l, Z^q) | z^l) \geq 1 - \epsilon,$$

i.e., validity conditionally on the training set $z^l$.

# Inductive conformal prediction

Trivial to implement, very fast.

Only drawback is to reduce the size of the training set, which should lead to worse models that TCP and more unstable results.

Other schemes have been proposed in between TCP and ICP, that extend standard cross-validation techniques (leave-one-out, boostrap, etc.). Keeping validy however requires special care.

The rest of this talk will focus on inductive conformal prediction.

# How to assess conformal predictors?

Two criteria:

- **Validity:** coherence between $\epsilon$ and actual error rate
- **Efficiency:** size of prediction regions (i.e. informativeness)

There is usually a confidence-efficiency tradeoff: the lower $\epsilon$, the larger the prediction set.

Validity:

- Guaranteed and fixed in advance by the user
- It's safe neverthless to estimate the actual error rate on an independent test set

Efficiency:

- Depends on the nonconformity function and on the underlying algorithm
- Different ways to measure it. E.g., average size of prediction sets over a test set.

# ICP in classification

DIGITS datasets from `sklearn`:

- 64 features ($8 \times 8$ images), 10 classes,
- 1797 examples splitted into 599 for training, 599 for calibration, and 599 for testing.

Nonconformity score: the classification margin:

$$A(h, (x, y)) = \max_{y_c \in \mathcal{Y} | y_c \neq y} P_h(y_c|x) - P_h(y|x),$$

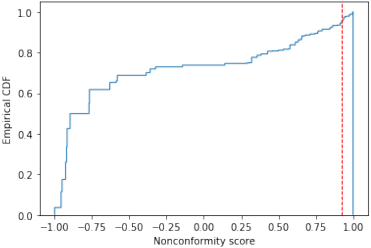
where $P_h(y|x)$ is the conditional probability estimate at $x$ of classifier $h$ for class $y$.

Two classifiers:

- Decision tree (`max_leaf_nodes` $= 20$)
- Random forests (1000 trees)

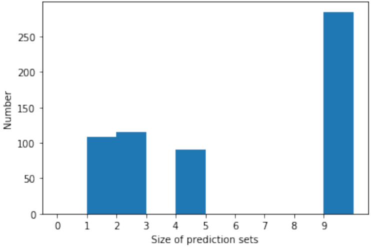
# Decision tree on DIGITS

Cumulative distribution of nonconformity scores on the calibration set with the decision tree.
⇒ at $\epsilon = 0.05$ accept a class $y$ if $A(h, (x_{n+1}, y)) < 0.93$.



Evaluation:

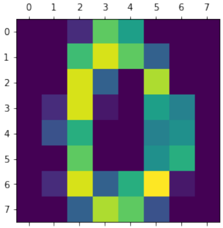
- Accuracy of the initial model: 73%
- Error rate of the predictions on the test set: 4%
- Average size of prediction sets: 5.9



# Decision tree on DIGITS: one example

Predicted probability vector: $[0.95, 0, 0, 0, 0, 0.017, 0., 0., 0.033, 0]$

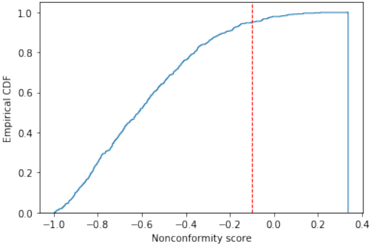
| $y$ | $A(h, (x, y))$ | $p(y)$ |
|---|---|---|
| 0 | -0.917 | 0.665 |
| 1 | 0.95 | 0.0283 |
| 2 | 0.95 | 0.0283 |
| 3 | 0.95 | 0.0283 |
| 4 | 0.95 | 0.0283 |
| 5 | 0.933 | 0.045 |
| 6 | 0.95 | 0.0283 |
| 7 | 0.95 | 0.0283 |
| 8 | 0.917 | 0.06 |
| 9 | 0.95 | 0.0283 |



Prediction set: $\{0, 8\}$
True class: 0

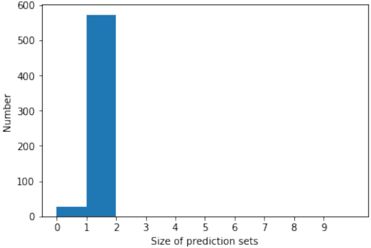
# Random forests on DIGITS

Cumulative distribution of nonconformity scores on the calibration set with Random forests.
$\Rightarrow$ at $\epsilon = 0.05$ accept a class $y$ if $A(h, (x_{n+1}, y)) < -0.1$.



Evaluation:

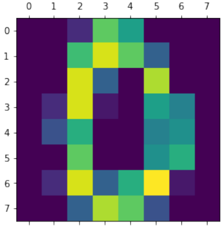
- Accuracy of the initial model: 98%
- Error rate of the predictions on the test set: 5%
- Average size of prediction sets: 0.95



# Random forests on DIGITS: one example

Predicted probability vector:
$[0.871, 0.003, 0.005, 0.006, 0.014, 0.033, 0.017, 0.007, 0.003, 0.041]$

| $y$ | $A(h,(x,y))$ | $p(y)$ |
|---|---|---|
| 0 | -0.83 | 0.805 |
| 1 | 0.868 | 0.0016 |
| 2 | 0.866 | 0.0016 |
| 3 | 0.865 | 0.0016 |
| 4 | 0.857 | 0.0016 |
| 5 | 0.838 | 0.0016 |
| 6 | 0.854 | 0.0016 |
| 7 | 0.864 | 0.0016 |
| 8 | 0.868 | 0.0016 |
| 9 | 0.83 | 0.0016 |



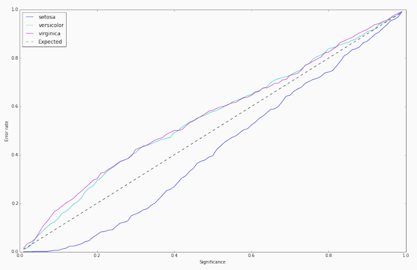
Prediction set: $\{0\}$
True class: 0

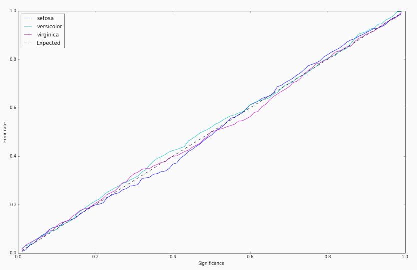
# ICP in classification: Mondrian approach

Error rate is guaranteed globally for all test examples, irrespectively of their classes.

One could guarantee an error rate per class by computing *p*-values for each class only from examples of the same class in the calibration set.

The same approach could be used to make the classifier (or regressor) fair with respect to a qualitative feature (e.g. sex, age).





[Source: Henrik Linusson]

# ICP in regression

A common nonconformity score for regression is the absolute error:

$$A(h, (x, y)) = |y - h(x)|$$

Let us denote by $\{\alpha_1, \dots, \alpha_q\}$ the nonconformity scores on the calibration set. The prediction set at $x_{n+1}$ is defined by all $y \in IR$ such that

$$|y - h(x_{n+1})| < \alpha_{1-\epsilon},$$

with $\alpha_{1-\epsilon}$ the score at the $1 - \epsilon$ percentile in $\{\alpha_1, \dots, \alpha_q\}$.

The prediction set for a new test example $x_{n+1}$ is thus given by the following confidence interval:

$$[h(x_{n+1}) - \alpha_{1-\epsilon}, h(x_{n+1}) + \alpha_{1-\epsilon}]$$

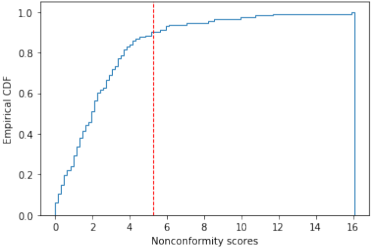
# ICP in regression: illustration

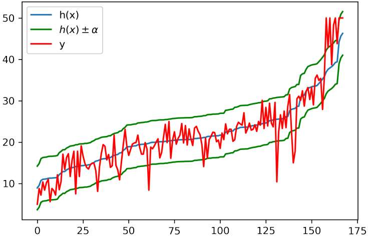
Boston dataset from `scikit-learn`:

- 13 features, one numerical output (house values)
- 506 examples splitted into 225 for training, 113 for calibration, and 168 for testing.

With Random forests (1000 models, no pruning, $\epsilon = 0.1$):

- $\alpha_{1-\epsilon} = 5.27$
- Error rate of the predictions on the test set: 9.5% (16/168)
- Average size of prediction sets: 10.54
- Root mean square error: 3.74





# ICP in regression

Using $A(h,(x,y)) = |y - h(x)|$ with ICP, one gets confidence intervals of **constant sizes**, whatever the test example $x_{n+1}$:

$$|\hat{C}(x_{n+1}; z^n)| = 2\alpha_{1-\epsilon}$$

This is also mostly the case in the TCP setting.

Ideally, we would like **individual bounds** that depend on $x_{n+1}$.

This can be achieved by using **normalized** nonconformity scores.

# Normalized nonconformity scores

Typical form of a normalized nonconformity score:

$$A((h, \sigma), (x, y)) = \frac{|y - h(x)|}{\sigma(x) + \beta},$$

which leads to confidence intervals in the following form:

$$\hat{C}(x_{n+1}; z^n) = [h_{z^l}(x_{n+1}) \pm \alpha_s(\sigma(x_{n+1}) + \beta)]$$

$\sigma(x)$ should quantify the **difficulty** of predicting $y$ at $x$.

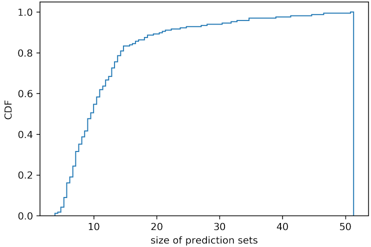
- It can be computed from statistics derived from the regressor itself (e.g., variance of ensemble predictions)
- It can be a second model trained to fit residuals of $h(x)$

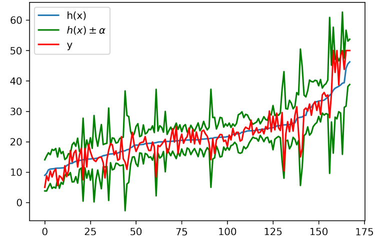
$\beta$ is a **sensitivity** parameter that can determine the impact of normalization (and cope with case when $\sigma(x) = 0$).

# Illustration on the Boston dataset

Using as $\sigma(x)$ the standard deviation of the predictions provided by the 1000 trees in the Random forest and setting $\beta = 0.05$ ($\epsilon = 0.1$):

- Error rate of the predictions on the test set: 11.3% (19/168)
- Average size of prediction sets: 12.21 (10.54 without normalization)
- Root mean square error: 3.74 (unchanged)





# Outline

1. Introduction

2. Conformal prediction

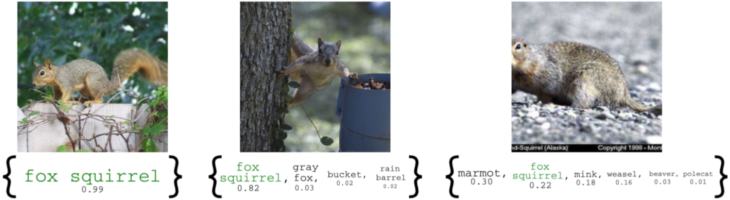
3. Inductive conformal prediction

4. Two recent papers
   - Regularized adaptive prediction sets
   - Conformalized quantile regression

5. Discussion and links

# Regularized adaptive prediction sets

“Uncertainty sets for image classifiers using conformal prediction”, Angelopoulos, Bates, Malik, Jordan, ICRL, 2021.

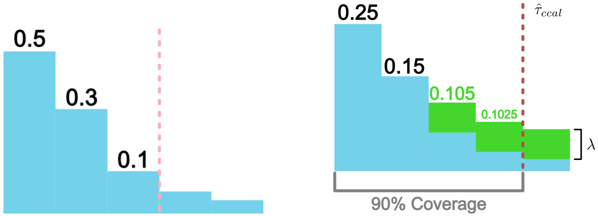

{ fox squirrel 0.99 }
{ fox squirrel, 0.82 gray fox, 0.03 bucket, 0.02 rain barrel 0.02 }
nd-Squirrel (Alaska)
Copyright 1998 - Mon
{ marmot, 0.30 fox squirrel, 0.22 mink, 0.18 weasel, 0.16 beaver, 0.03 polecat 0.01 }

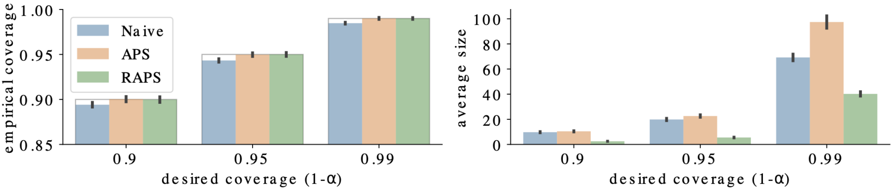
https://people.eecs.berkeley.edu/~angelopoulos/blog/posts/conformal-classification/

Video

- CP applied in classification
- A simple trick to reduce informativeness, while maintaining validity

# Regularized adaptive prediction sets





(ImageNet, ResNet-152, 100 random splits)

# Outline

1. Introduction

2. Conformal prediction

3. Inductive conformal prediction

4. Two recent papers
   - Regularized adaptive prediction sets
   - Conformalized quantile regression

5. Discussion and links

# Conformal quantile regression

“Conformalized Quantile Regression”, Romano, Patterson, Candès, NeurIPS, 2019.

https://sites.google.com/view/cqr

Main idea:

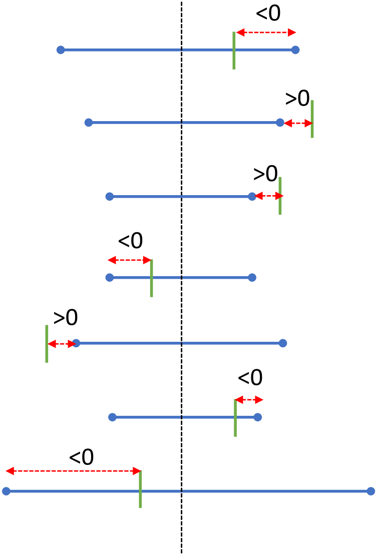
- Assume we already have a (quantile regression) method that produces confidence intervals but without guarantees.
- CP is used to update the predictions of this method to get back these guarantees
- An alternative to the normalization of nonconformality scores to get non constant sized confidence intervals.

# Conformalized quantile regression: algorithm

Given a learning set $z^n$ and a test input $x_{n+1}$

1. Like in ICP, split the training data $z^n$ into two sets $z^l$ and $z^q$.
2. Fit any quantile regression algorithm on $z^l$ to get two quantile functions: $\hat{q}_{\alpha_{lo}}$ and $\hat{q}_{\alpha_{hi}}$ such that $[\hat{q}_{\alpha_{lo}}(x), \hat{q}_{\alpha_{hi}}(x)]$ is expected to be a $1-\epsilon$ confidence interval for $y$ at $x$ (with no guarantees).
3. Compute the error made by these functions on the calibration set as:
$$E_i = \max\{\hat{q}_{\alpha_{lo}}(x_i) - y_i, y_i - \hat{q}_{\alpha_{hi}}(x_i)\}, i = 1, \ldots, q$$
4. Compute $E_{1-\epsilon}$ as the $(1-\epsilon)$th empirical quantile of $\{E_1, \ldots, E_q\}$.
5. The new prediction interval is finally defined as:
$$\hat{C}(x_{n+1}; z^n) = [\hat{q}_{\alpha_{lo}}(x) - E_{1-\epsilon}, \hat{q}_{\alpha_{hi}}(x) + E_{1-\epsilon}]$$

# Conformalized quantile regression: illustration



# Conformalized quantile regression: validity

We have the usual theorem:

**Theorem:** *If $Z^n$ is i.i.d., then we have:*

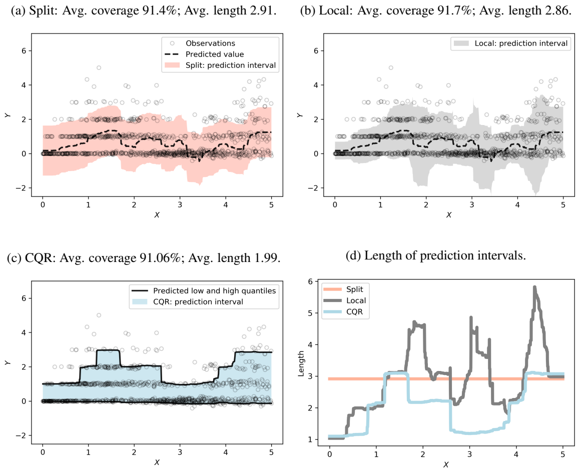
$$P(Y_{n+1} \in C(X_{n+1}; Z^n)) \geq 1 - \epsilon.$$

The authors furthermore show that if there are no ties in the scores $E_i$:

$$P(Y_{n+1} \in C(X_{n+1}; Z^n)) \leq 1 - \epsilon + \frac{1}{q+1},$$

which means the intervals are nearly perfectly calibrated (if $q$ is large enough).

# Some results: with RF on a synthetic problem



(a) Split: Avg. coverage 91.4%; Avg. length 2.91.

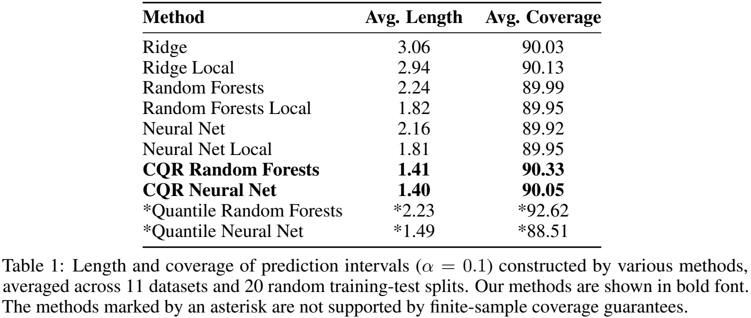
(b) Local: Avg. coverage 91.7%; Avg. length 2.86.

(c) CQR: Avg. coverage 91.06%; Avg. length 1.99.

(d) Length of prediction intervals.

Figure 2: Prediction intervals on simulated heteroscedastic data with outliers (see Figure 7 for a full range display): (a) the standard split conformal method, (b) its locally adaptive variant, and (c) CQR (our method). The length of the interval as a function of $X$ is shown in (d). The target coverage rate is 90%. The broken black curve in (a) and (b) is the pointwise prediction from the random forest estimator. In (c), we show two curves, representing the lower and upper quantile regression estimates based on random forests [22]. Observe how in this example the quantile regression estimates closely match the adjusted estimates—the boundary of the blue region—obtained by conformalization.

# Some results: comparison of different methods



| **Method** | **Avg. Length** | **Avg. Coverage** |
|---|---|---|
| Ridge | 3.06 | 90.03 |
| Ridge Local | 2.94 | 90.13 |
| Random Forests | 2.24 | 89.99 |
| Random Forests Local | 1.82 | 89.95 |
| Neural Net | 2.16 | 89.92 |
| Neural Net Local | 1.81 | 89.95 |
| **CQR Random Forests** | **1.41** | **90.33** |
| **CQR Neural Net** | **1.40** | **90.05** |
| *Quantile Random Forests | *2.23 | *92.62 |
| *Quantile Neural Net | *1.49 | *88.51 |

Table 1: Length and coverage of prediction intervals ($\alpha = 0.1$) constructed by various methods, averaged across 11 datasets and 20 random training-test splits. Our methods are shown in bold font. The methods marked by an asterisk are not supported by finite-sample coverage guarantees.

# Outline

1. Introduction

2. Conformal prediction

3. Inductive conformal prediction

4. Two recent papers

5. Discussion and links

# Summary

- Conformal prediction is a general framework that allows to make set (rather than point) predictions with guarantees in terms of error rate in finite setting.
- Can be used with any (predictive) learning algorithm.
- The ICP approach is intuitive, trivial to implement and extremely fast.
- CP can be leveraged to provide guarantees for other uncertainty estimation methods (quantile regression, bayesian methods, ensembles, etc.).
- Many possible applications outside supervised learning (outlier detection, variable importances, etc.).

# Limitations

- Not so easy to find good nonconformity functions and to understand how they impact efficiency.
- But there is an important literature on nonconformity functions for different underlying algorithms (e.g., Random forests, neural networks).
- Only marginal coverage is guaranteed, while what we would really like is conditional coverage.

# Conditional versus marginal coverage

CP offers guarantees in terms of marginal coverage:

$$P(Y_{n+1} \in \hat{C}(X)) \geq 1 - \epsilon,$$

which does not say anything about the error rate for a specific $x$.

In practice, we would really like conditional coverage:

$$P(Y_{n+1} \in \hat{C}(x) | X = x) \geq 1 - \epsilon$$

The Mondrian approach is a way to have coverage guarantees in specific subcategories of objects but it can not scale to numerical features.

Theoretical analyses tend to show that "conditional coverage guarantees are impossible to obtain without imposing assumptions on the underlying distribution".

See "The limits of distribution-free conditional predictive inference", Barber et al., 2019.

# References and links

Papers:

- "A tutorial on conformal prediction", Shafer and Vovk, 2008
- "Inductive conformal prediction: theory and application to neural networks", Papadopoulos, 2008
- "Distribution-Free Predictive Inference For Regression", Lei et al., 2016.

Talks, course:

- An introduction to conformal prediction, Henrik Linusson, 2017.
- Short course by Jing Lei, 2020.

Software:

- nonconformist (a scikit-learn compatible python package)
- Conformal Inference R Project

Conference:

- COPA: Annual symposium on Conformal and Probabilistic Prediction with Applications