On Efficiency of Learning Under Privileged Information

Ilia Nouretdinov

Royal Holloway, University of London

i.r.nouretdinov@rhul.ac.uk

2022

Abstract

The paradigm of Learning Under Privileged Information (LUPI) was used in various practical applications, including its combination with Conformal Prediction (CP) framework.
In this note, we discuss possible sources and limitations of its efficiency.
We try to argue that accuracy improvement coming from using privileged information is not occasional.
For this goal, we consider some minimalist models of LUPI where the contribution of the privileged information appears in its noise-free essence.
Then, we discuss connection of LUPI paradigm and CP framework in relation with the models.

Introduction

Learning Under Privileged Information (LUPI) paradigm of machine learning was initially presented by Vapnik and Vashist.
In this view, a data example consists of:

	feature vector x	privileged info x^*	label y
Training set	known	known	known
Testing set To be predicted?	known -	unknown -	unknown yes

As a baseline, it is always possible just to ignore all PI.
Although applications demonstrate a positive contribution of PI to accuracy, using LUPI paradigm may be useless if PI is noisy or redundant.
In this work, we take a step back from practical applications to some minimalist artificial models. This is needed to study the effect of PI clarified from noise and side circumstances.

Minimalist model of LUPI

We assume that the data follows i.i.d. (power) assumption that all the triples (x_i, x_i^*, y_i) are generated independently by the same distribution P.
For our first example, we consider the following data generating model:

x	x^*	y	$P\{(x, x^*, y)\}$
0	1	odd	0.4
0	2	even	0.2
0	3	odd	0.2
0	4	even	0.2

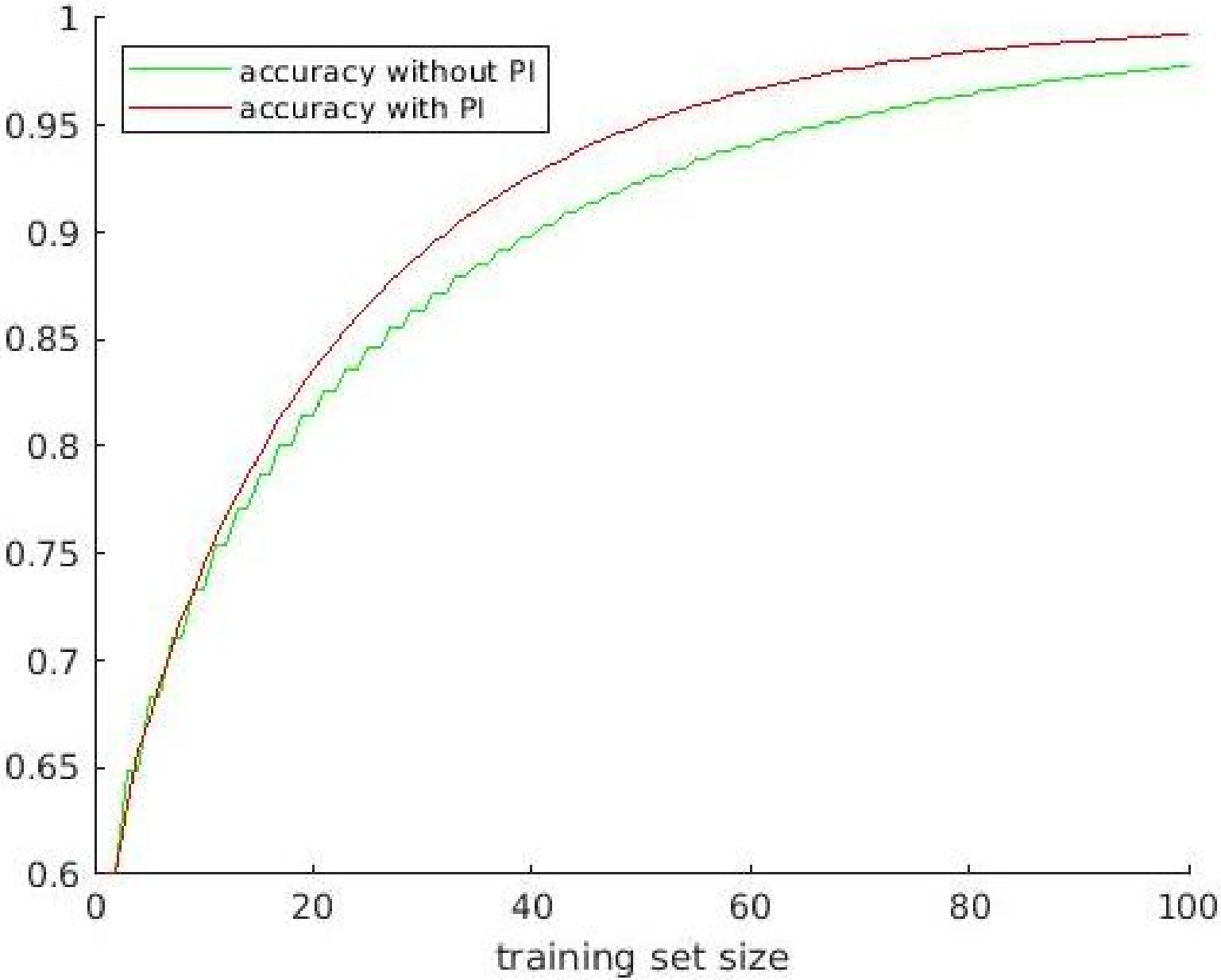
Regardless whether PI is used or not, the final machine learning task is to answer the question: "is y_{n+1} even or odd"?
The best answer is "odd" because the true probability of this event is $P\{y = odd|x\} = 0.6$ that can not be improved.
So, we will measure the accuracy of a prediction algorithm by its chance of giving the output "odd" after training.

Examples of "all-neighbours" rule applied

- **Without privileged information:**
  training data: $(y_1, \ldots, y_n)$;
  decision rule: choose between "even", "odd" by majority of votes;
  breaking ties: fairly randomised;
  (even,odd,even)$\to$even;
  (even,odd,odd)$\to$odd;
  (even,odd,odd,even)$\to \frac{1}{2}$ odd, $\frac{1}{2}$ even.
- **With privileged information:**
  training data: $(x_1^*, \ldots, x_n^*)$;
  decision rule: choose between 1,2,3,4 by the highest number of votes, then convert to "odd" or "even";
  breaking ties: equally randomised between the winners;
  (1,2,1,3,4)$\to 1 \to$ odd;
  (1,2,1,2,4)$\to \frac{1}{2}$ 1, $\frac{1}{2}$ 2 $\to \frac{1}{2}$ odd, $\frac{1}{2}$ even;
  (1,2,3,1,2,3,4)$\to \frac{1}{3}$ 1, $\frac{1}{3}$ 2, $\frac{1}{3}$ 3 $\to \frac{2}{3}$ odd, $\frac{1}{3}$ even.

The final answers will match in majority of cases. But there may be exceptions: (1,1,1,2,2,4,4)=(odd,odd,odd,even,even,even,even).

Dependence on the training set size (minimalist example)



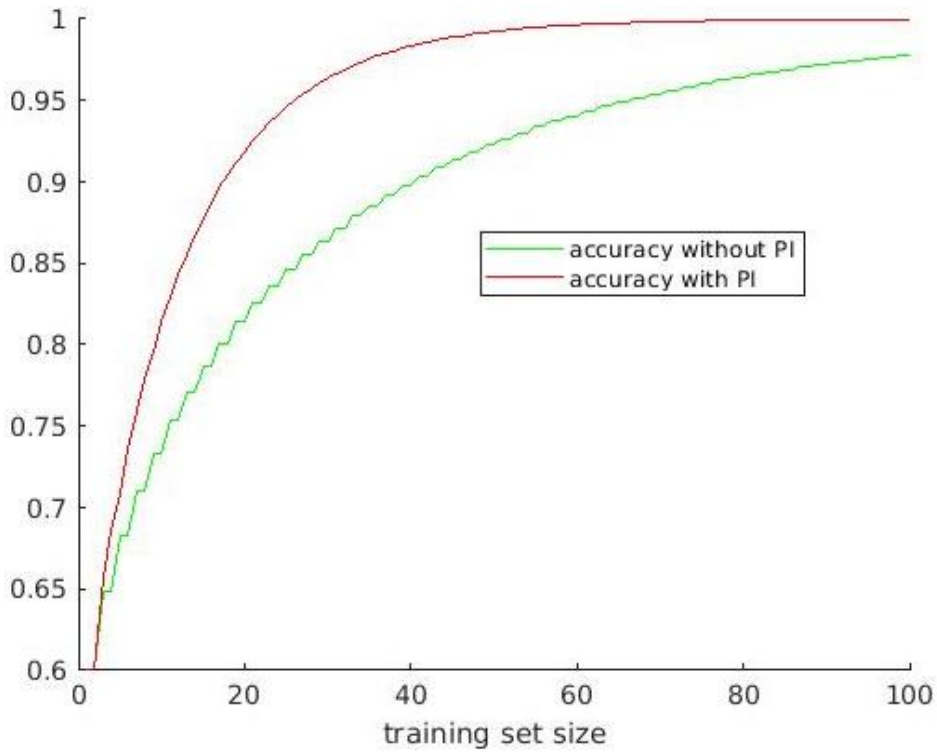
Observations

1. Starting from $n = 10$, PI always makes a positive contribution.
2. Starting from $n = 40$, the contribution of Pi decreases although remains positive.
3. There is a permanent difference between odd and even values of n

To sum, we can make a hypothesis that involving PI is useless in the very early stage, not essential asymptotically, but it can help to accelerate the learning in the middle stage.
In the initial period ($n < 10$ in this example), PI does not help in classification because it split the data into very small classes, and this prevents any essential analysis. Later (starting from $n > 10$), PI becomes useful by giving the algorithm a 'hint' about the data structure. However, asymptotically the difference between learning with/without PI gradually becomes negligible.

An imbalanced model and gain of PI

x	x^*	y	$P\{(x, x^*, y)\}$
0	1	odd	0.5
0	2	even	0.2
0	3	odd	0.1
0	4	even	0.2

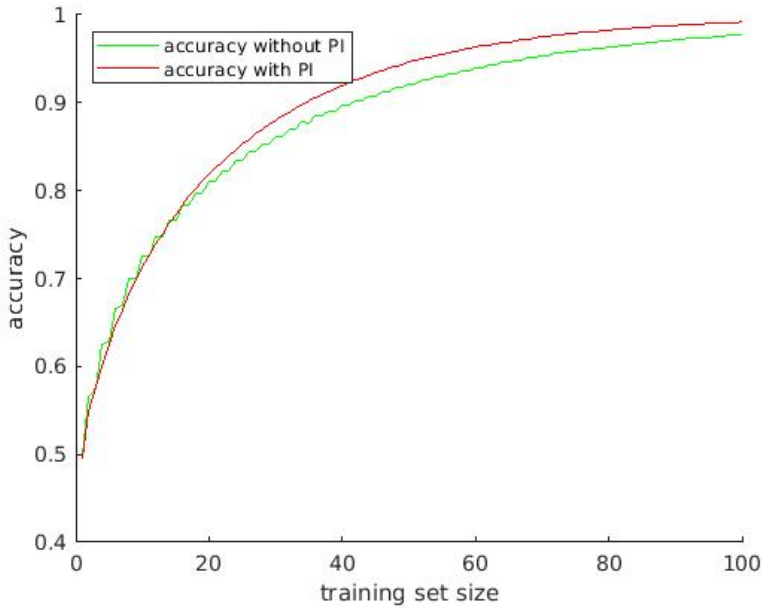


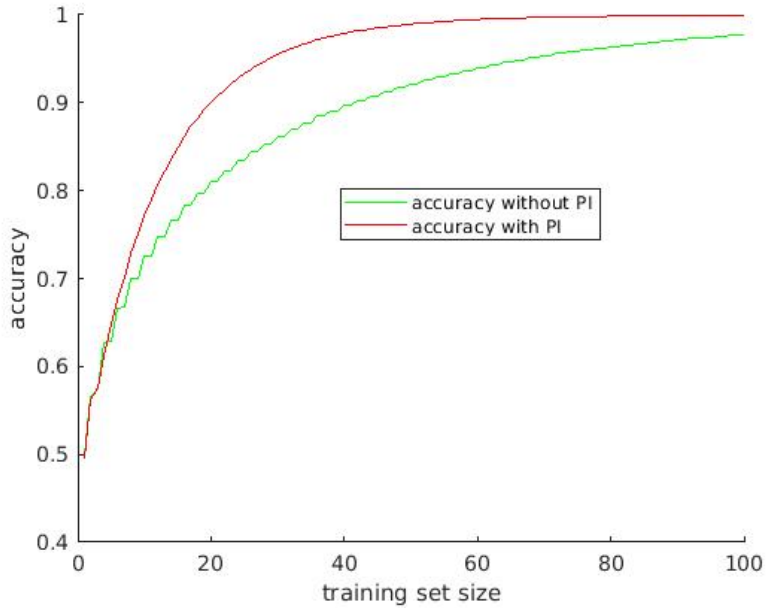
Using LUPI within Conformal Prediction

INPUT: training data (triples) $(x_1, x_1^*, y_1), \ldots, (x_n, x_n^*, y_n) \in X \times X^* \times Y$
INPUT: testing example x_{n+1}
INPUT: conformity score $A : (z, Z) \to [-\infty, +\infty]$ where $z = (x, x^*, y)$
INPUT: using PI? (yes/no)
if PI=no **then**
  set $X^* := \{0\}$ and all $x_i^* := 0$
end if
for $(x^*, y) \in X^* \times Y$ **do**
  $x_{n+1}^* := x^*$, $y_{n+1} := y$
  for i:=1,...,n+1 **do**
    $\alpha_i := A\left((x_i, x_i^*, y_i), \{(x_j, x_j^*, y_j) | j = 1, \ldots, j-1, j+1, \ldots, n+1\}\right)$
  end for
  generate θ with uniform distribution on [0,1]
  $p(x^*, y) = \frac{|\{i:\alpha_i<\alpha_{n+1}\}|+\theta|\{i:\alpha_i=\alpha_{n+1}\}|}{n+1}$
end for
OUTPUT: $p(y) = \max_{x^*} p(x^*, y)$

Conformal versions of minimalist and imbalanced examples

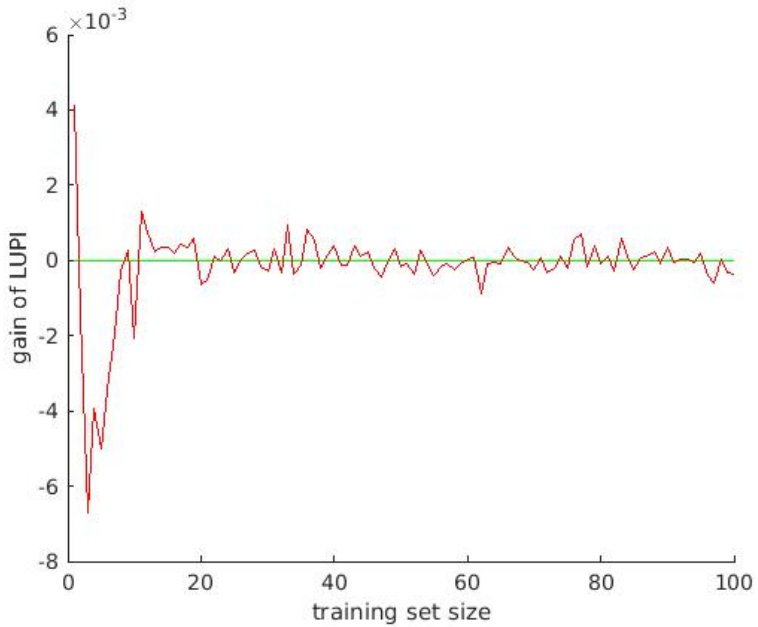
In that minimalist examples, the conformity score is also defined simply as: 'the proportion of examples having the same extended label'.





More complex examples (non-trivial x)

$P\{x\|(x^*, y)\}$	x^*	y	$P\{(x^*, y)\}$
$N(1, 1)$	1	odd	0.4
$N(2, 1)$	2	even	0.2
$N(3, 1)$	3	odd	0.2
$N(4, 1)$	4	even	0.2

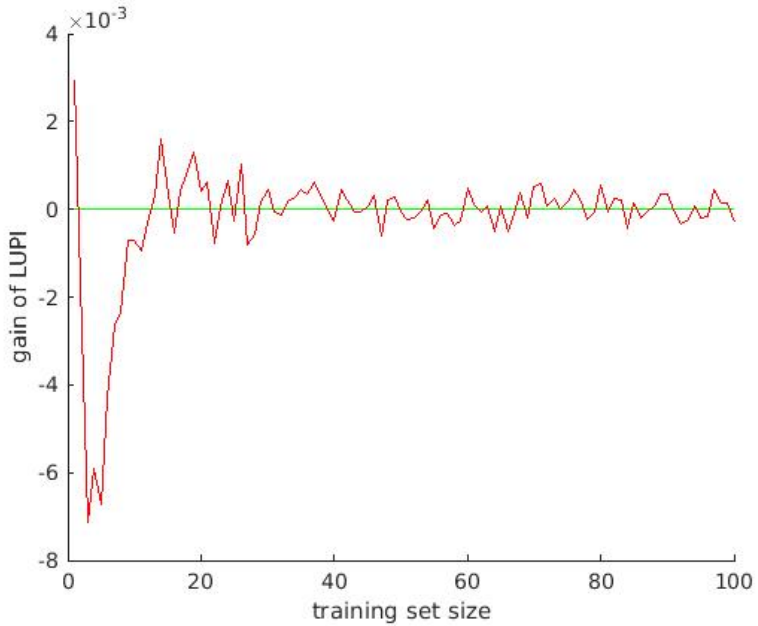


Conformity measure:

$$\frac{\textit{distance to the nearest neighbour with another label}}{\textit{distance to the nearest neighbour with the same label}}.$$

More complex examples (non-trivial x)

$P\{x\|(x^*, y)\}$	x^*	y	$P\{(x^*, y)\}$
$N(1, 1)$	1	odd	0.5
$N(2, 1)$	2	even	0.2
$N(3, 1)$	3	odd	0.1
$N(4, 1)$	4	even	0.2



Conclusion

Our observations confirm the following hypothesis about the dynamics of the impact of PI can be roughly divided into three stages, in dependence on the training set size.

1. Very small size – the gain from using PI is negative, as PI overloads the learning algorithm.
2. Medium size – the gain is positive, as PI accelerates the learning.
3. Large size – the gain is still positive but tends to decrease, as learning without PI also becomes efficient.

This report is just the beginning of the work in progress, and these conclusions are preliminary and due to further check and analysis. As far as the models become more complex, more limitations on the applicability of LUPI will be found. Even at this level, we have to note: if the noise in PI is high, then the pattern may be weaker than this one.
One important direction of future work is the validation with real-world data sets. However, it also may gain from a dynamic investigation for different sizes of the training set.