Conformal Prediction under Hypergraphical Models*

Valentina Fedorova, Alex Gammerman, Ilia Nouretdinov, and Vladimir Vovk

Computer Learning Research Centre
Royal Holloway, University of London, UK
{valentina,ilia,alex,vovk}@cs.rhul.ac.uk

Abstract

Conformal predictors are usually defined and studied under the exchangeability assumption. However, their definition can be extended to a wide class of statistical models, called online compression models, while retaining their property of automatic validity. This paper is devoted to conformal prediction under hypergraphical models that are more specific than the exchangeability model. We define conformity measures for such hypergraphical models and study the corresponding conformal predictors empirically on benchmark LED data sets. Our experiments show that they are more efficient than conformal predictors that use only the exchangeability assumption.

1998 ACM Subject Classification I.2.6 Learning

Keywords and phrases conformal prediction, hypergraphical models, conformity measure

Digital Object Identifier 10.4230/OASIcs.ICCSW.2013.27

1 Introduction

The method of conformal prediction was introduced and is usually used for producing valid prediction sets under the exchangeability assumption; the validity of the method means that the probability of making a mistake is equal to (or at least does not exceed) a prespecified significance level ([5], Chapter 2). However, the definition of conformal predictors can be easily extended to a wide class of statistical models, called online compression models (OCMs; [5], Chapter 8). OCMs compress data into a more or less compact summary, which is interpreted as the useful information in the data. With each “conformity measure”, which, intuitively, estimates how well a new piece of data fits the summary, one can associate a conformal predictor, which still enjoys the property of automatic validity.

This paper studies conformal prediction under the OCMs known as hypergraphical models ([5], Section 9.2). Such models describe relationships between data features. In the case where every feature is allowed to depend in any way on the rest of the features, the hypergraphical model becomes the exchangeability model. More specific hypergraphical models restrict the dependence in some way. Such restrictions are typical of many real-world problems: for example, different symptoms can be conditionally independent given the disease. A popular approach to such problems is to use Bayesian networks (see, e.g., [2]). The definition of Bayesian networks requires a specification of both the pattern of dependence between features and the distribution of the features. Usual methods guarantee a valid probabilistic outcome if the used distributions of features are correct. Several algorithms (see, e.g., [2],

* A longer version of this paper appeared in the proceedings of COPA **2013** [4].

2013 Imperial College Computing Student Workshop (ICCSW'13).
Editors: Andrew V. Jones, Nicholas Ng; pp. 27–34
OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Chapter 9) are known for estimating the distribution of features; however, the accuracy of such approximations is a major concern in applying Bayesian networks. The conformal predictors constructed from hypergraphical OCMs use only the pattern of dependence between the features but do not involve their distribution. This makes conformal prediction based on hypergraphical models more robust and realistic than Bayesian networks.

As far as we know, conformal prediction has been studied, apart from the exchangeability model and its variations, only for the Gauss linear model and Markov model (see [5], Chapter 8, and [3]). Hypergraphical OCMs have been used only in the context of Venn rather than conformal prediction (see [5], Chapter 9).

The rest of the paper is organised as follows. Section 2 formally defines hypergraphical OCMs and briefly reviews their basic properties. Section 3 describes the method of conformal prediction in the context of hypergraphical models and introduces a class of conformity measures for hypergraphical OCMs. Section 4 reports the performance of the corresponding conformal predictors on benchmark LED data sets. Section 5 concludes.

2 Background

Consider two measurable spaces $\mathbf{X}$ and $\mathbf{Y}$; elements of $\mathbf{X}$ are called *objects* and elements of $\mathbf{Y}$ are called *labels*. Elements of the Cartesian product $\mathbf{X} \times \mathbf{Y}$ are called *examples*. A *training set* is a sequence of examples $(z_1, \dots, z_l)$, where each example $z_i = (x_i, y_i)$ consists of an object x_i and its label y_i. The general prediction problem considered in this paper is to predict the label for a new object given a training set. We focus on the case where $\mathbf{X}$ and $\mathbf{Y}$ are finite.

2.1 Hypergraphical Structures

In this paper we assume that examples are structured, consisting of variables. Hypergraphical structures describe relationships between the variables. Formally a *hypergraphical structure*[1] consists of three elements $(V, \mathcal{E}, \Xi)$:

1. V is a finite set; its elements are called *variables*.
2. $\mathcal{E}$ is a finite collection of subsets of V whose union covers all variables: $\bigcup_{E \in \mathcal{E}} E = V$. Elements of $\mathcal{E}$ are called *clusters*.
3. Ξ is a function that maps each variable $v \in V$ into a finite set (of the values that v can take).

A *configuration* on a set $E \subseteq V$ (we are usually interested in the case where E is a cluster) is an assignment of values to the variables from E; let $\Xi(E)$ be the set of all configurations on E. A *table*[2] on a set E is an assignment of natural numbers to the configurations on E. The *size* of the table is the sum of values that it assigns to different configurations. A *table set* is a collection of tables on the clusters $\mathcal{E}$, one for each cluster $E \in \mathcal{E}$. The number assigned by a table set σ to a configuration on E is called its *σ-count*.

[1] The name reflects the fact that the components $(V, \mathcal{E})$ form a hypergraph, where a hyperedge $E \in \mathcal{E}$ can connect more than two vertices.

[2] Generally, a table assigns real numbers to configurations. In this paper we only consider *natural tables*, which assign natural numbers to configurations, and omit "natural" for brevity.

2.2 Hypergraphical Online Compression Models

The example space $\mathbf{Z}$ associated with the hypergraphical structure is the set of all configurations on V. One of the variables in V is singled out as the *label variable*, and the configurations on the label variable are denoted $\mathbf{Y}$. All other variables are *object variables*, and the configurations on the object variables are denoted $\mathbf{X}$. Since $\mathbf{Z} = \mathbf{X} \times \mathbf{Y}$, this is a special case of the prediction setting described at the beginning of this section.

An example $z \in \mathbf{Z}$ *agrees* with a configuration on a set $E \subseteq V$ (or the configuration agrees with the example) if the restriction $z|_E$ of z to the variables in E coincides with the configuration. A table set σ *generated* by a sequence of examples $(z_1, \dots, z_n)$ assigns to each configuration on each cluster the number of examples in the sequence that agree with the configuration; the size of each table in σ will be equal to the number of examples in the sequence, and this number is called the *size* of the table set. Different sequences of examples can generate the same table set σ, and we denote $\#\sigma$ the number of different sequences generating σ.

The *hypergraphical online compression model* (HOCM) associated with the hypergraphical structure $(V, \mathcal{E}, \Xi)$ consists of five elements $(\Sigma, \Box, \mathbf{Z}, F, B)$, where:

1. The *empty table set* $\Box$ is the table set assigning 0 to each configuration.
2. The set Σ is defined by the conditions that $\Box \in \Sigma$ and $\Sigma \setminus \{\Box\}$ is the set of all table sets σ with $\#\sigma > 0$. The elements $\sigma \in \Sigma$ are called *summaries*.
3. The *forward function* $F(\sigma, z)$, where σ ranges over Σ and z over $\mathbf{Z}$, updates σ by adding 1 to the σ-count of each configuration which agrees with z.
4. The *backward kernel* B maps each $\sigma \in \Sigma \setminus \{\Box\}$ to a probability distribution $B(\sigma)$ on $\Sigma \times \mathbf{Z}$ assigning the weight $\#(\sigma \downarrow z)/\#\sigma$ to each pair $(\sigma \downarrow z, z)$, where z is an example such that, for all configurations which agree with z, the corresponding σ-counts are positive, and $\sigma \downarrow z$ is the table set obtained by subtracting 1 from the σ-counts of the configurations that agree with z. Notice that $B(\sigma)$ is indeed a probability distribution, and it is concentrated on the pairs $(\sigma \downarrow z, z)$ such that $F(\sigma \downarrow z, z) = \sigma$.

We will use "hypergraphical models" as a general term for hypergraphical structures and HOCMs when no precision is required. When discussing hypergraphical models we will always assume that the examples $z_1, z_2, \dots$ are produced independently from a probability distribution Q on $\mathbf{Z}$ that has a decomposition

$$Q(\{z\}) = \prod_{E \in \mathcal{E}} f_E(z|_E) \tag{1}$$

for some functions $f_E : \Xi(E) \to [0, 1]$, $E \in \mathcal{E}$, where z is an example and $z|_E$ its restriction to the variables in E.

2.3 Junction Tree Structures

An important type of hypergraphical structures is where clusters can be arranged into a "junction tree". For the corresponding HOCMs we will be able to describe efficient calculations of the backward kernels. If one wants to use the calculations for a structure that cannot be arranged into a junction tree it can be replaced by a more general junction tree structure before defining the HOCM.

Let (U, S) denote an undirected tree with U the set of vertices and S the set of edges. Then (U, S) is a *junction tree* for a hypergraphical structure $(V, \mathcal{E}, \Xi)$ if there exists a bijective mapping C from the set of vertices U of the tree to the set $\mathcal{E}$ of clusters of the hypergraphical structure that has the following property: $C_u \cap C_w \subseteq C_v$ whenever a vertex v lies on the path from a vertex u to a vertex w in the tree (we let C_x stand for $C(x)$).

If $s = \{u, v\} \in S$ is an edge of the junction tree connecting vertices u and v then C_s stands for $C_u \cap C_v$. It is convenient to identify vertices u and edges s of the junction tree with the corresponding clusters C_u and sets C_s, respectively.

If $E_1 \subseteq E_2 \subseteq V$ and f is a table on E_2, the *marginalisation* of f to E_1 is the table f^* on E_1 assigning to each $a \in \Xi(E_1)$ the number $f^*(a) = \sum_b f(b)$, where b ranges over the configurations on E_2 such that $b|_{E_1} = a$. If σ is a summary then for $u \in U$ denote σ_u the table that σ assigns to C_u, and for $s = \{u, v\} \in S$ denote σ_s the marginalisation of σ_u (or σ_v) to C_s. We will use the shorthand $\sigma_u(z)$ for the number assigned to the restriction $z|_{C_u}$ by the table for the vertex u and $\sigma_s(z)$ for the number assigned to $z|_{C_s}$ by the marginal table for the edge s. Consider the HOCM corresponding to the junction tree (U, S). We use the notation $P_\sigma(z)$ for the weight assigned by $B(\sigma)$ to $(\sigma \downarrow z, z)$. It has been proved ([5], Lemma 9.5) that

$$P_\sigma(z) = \frac{\prod_{u \in U} \sigma_u(z)}{n \prod_{s \in S} \sigma_s(z)}, \tag{2}$$

where n is the size of σ. If any of the factors in (2) is zero then the whole ratio is set to zero.

3 Conformal Prediction for HOCM

Consider a training set $(z_1, \dots, z_l)$ and an HOCM $(\Sigma, \Box, \mathbf{Z}, F, B)$. The goal is to predict the label for a new object x.

A *conformity measure* for the HOCM is a measurable function $A : \Sigma \times \mathbf{Z} \to \mathbb{R}$. The function assigns a *conformity score* $A(\sigma, z)$ to an example z w.r.t. to a summary σ. Intuitively, the score reflects how typical it is to observe z having the summary σ.

For each $y \in \mathbf{Y}$ denote $\sigma^* \in \Sigma$ the table set generated by the sequence $(z_1, \dots, z_l, (x, y))$ (the dependence of σ^* on y is important although not reflected in our notation). For $z \in \mathbf{Z}$ such that $\sigma^* \downarrow z$ is defined denote the conformity scores as $\alpha_z := A(\sigma^* \downarrow z, z)$ (notice that $\alpha_{(x,y)}$ is always defined). The *p-value* for y, denoted $p^{(y)}$, is defined by

$$p^{(y)} := \sum_{z : \alpha_z < \alpha_{(x,y)}} P_{\sigma^*}(z) + \theta \cdot \sum_{z : \alpha_z = \alpha_{(x,y)}} P_{\sigma^*}(z) \tag{3}$$

(cf. (8.4) in [5]), where $\theta \sim \mathbf{U}[0, 1]$ is a random number from the uniform distribution on $[0, 1]$, $P_{\sigma^*}(z)$ is the backward kernel, as defined above, and the sums involve only those $z \in \mathbf{Z}$ for which α_z is defined. Then for a significance level ϵ the *conformal predictor* Γ based on A outputs the prediction set

$$\Gamma^\epsilon(z_1, \dots, z_l, x) := \{y \in \mathbf{Y} : p^{(y)} > \epsilon\}.$$

The following section 3.1 defines one class of conformity measures for HOCMs and section 3.2 describes the criteria for the quality of conformal predictions which we use in the paper; for other conformity measures and more criteria see sections 3.1 and 3.2 in [4].

3.1 Conformity Measures for HOCM

Consider a summary σ and an example (x, y). The conditional probability $P_{\sigma^*}(y \mid x)$ of y given x under P_{σ^*} can be computed using (2) as follows

$$P_{\sigma^*}(y \mid x) = \frac{P_{\sigma^*}((x, y))}{\sum_{y' \in \mathbf{Y}} P_{\sigma^*}((x, y'))},$$

where $\sigma^* := F(\sigma, (x, y))$ and $P_{\sigma^*}((x, y))$ is the backward kernel. Define the *predictability* of an object $x \in \mathbf{X}$ as

$$f(x) := \max_{y \in \mathbf{Y}} P_{\sigma^*}(y \mid x), \tag{4}$$

the maximum of conditional probabilities. If the predictability of an object is close to 1 then the object is "easily predictable". Fix a *choice function* $\hat{y} : \mathbf{X} \to \mathbf{Y}$ such that

$$\forall x \in \mathbf{X} : f(x) = P_{\sigma^*}(\hat{y}(x) \mid x).$$

The function maps each object x to one of the labels at which the maximum in (4) is attained. The *signed predictability conformity measure* is defined by

$$A(\sigma, (x, y)) := \begin{cases} f(x) & \text{if } y = \hat{y}(x) \\ -f(x) & \text{otherwise.} \end{cases} \tag{5}$$

3.2 Criteria for the Quality of Conformal Prediction

In this paper we study the performance of conformal predictors in the online prediction protocol (Protocol 1). Reality generates examples (x_n, y_n) from a probability distribution Q satisfying (1) for some hypergraphical structure. Predictor uses a conformal predictor Γ to output the prediction set $\Gamma_n^\epsilon := \Gamma^\epsilon(x_1, y_1, \dots, x_{n-1}, y_{n-1}, x_n)$ at each significance level ϵ.

Protocol 1 Online prediction protocol

```
for $n = 1, 2, \dots$ do
    Reality outputs $x_n \in \mathbf{X}$
    Predictor outputs $\Gamma_n^\epsilon \subseteq \mathbf{Y}$ for all $\epsilon \in (0, 1)$
    Reality outputs $y_n \in \mathbf{Y}$
end for
```

Two important properties of conformal predictors are their validity and efficiency; the first is achieved automatically and the second is enjoyed by different conformal predictors to a different degree. Predictor *makes an error* at step n if y_n is not in Γ_n^ϵ. The validity of conformal predictors means that, for any significance level ϵ, the probability of error $y_n \notin \Gamma_n^\epsilon$ is equal to ϵ. It has been proved that conformal predictors are automatically valid under their models ([5], Theorem 8.1). In this paper we study problems where the hypergraphical model used for computing the p-values is known to be correct; therefore, the predictions will always be valid, and there is no need to test validity experimentally. One possible way to measure efficiency is to count the *number of multiple predictions* Mult_n^ϵ over the first n steps defined by

$$\mathrm{mult}_n^\epsilon := \begin{cases} 1 & \text{if } |\Gamma_n^\epsilon| > 1 \\ 0 & \text{otherwise} \end{cases} \qquad \text{and} \qquad \mathrm{Mult}_n^\epsilon := \sum_{i=1}^{n} \mathrm{mult}_i^\epsilon$$

at each significance level $\epsilon \in (0, 1)$ (cf. [5], Chapter 3). In our experiments we will look at the *percentage of multiple predictions* $\mathrm{Mult}_n^\epsilon / n$; we would like it to be close to 0 for small significance levels.

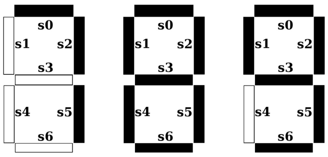

Figure 1 LED images for digits 7, 8, and 9 in the seven-segment display.

4 Experimental Results

4.1 LED Data Set

For our experiments we use benchmark LED data sets generated by a program from the UCI repository [1]. The problem is to predict a digit from an image in the seven-segment display. Figure 1 shows several objects in the data set (these are "ideal images" of digits; there are also digits corrupted by noise). The seven LEDs (light emitting diodes) can be lit in different combinations to represent a digit from 0 to 9. The program generates examples with noise. There is an ideal image for each digit. An example has seven binary attributes $s_0, \dots, s_6$ (s_i is 1 if the ith LED is lit) and a label c, which is a decimal digit. The program randomly chooses a label (0 to 9 with equal probabilities), inverts each of the attributes of its ideal image with probability $p_{\text{noise}} = 1\%$ independently, and adds the noisy image and the label to the data set.

4.2 Hypergraphical Assumptions for LED Data Sets

We consider two hypergraphical models that agree with the generating mechanism. These models make different assumptions about the pattern of dependence between the attributes and the label; they do not depend on a particular probability of noise p_{noise} or the fact that the same value of p_{noise} is used for all LEDs. For both hypergraphical structures the set of variables is $V := \{s_0, \dots, s_6, c\}$.

Nontrivial Hypergraphical Model. Consider the hypergraphical structure with the clusters $\mathcal{E} := \{\{s_i, c\} : i = 0, \dots, 6\}$. A junction tree for this hypergraphical structure can be defined as a chain with vertices $U := \{u_i : i = 0, \dots, 6\}$ and the bijection $C_{u_i} := \{s_i, c\}$.

Exchangeability Model. The hypergraphical model with no information about the pattern of dependence between the attributes and the label is the exchangeability model. The corresponding hypergraphical structure has one cluster, $\mathcal{E} := \{V\}$. The junction tree is the one vertex associated with V.

4.3 Experiments

For our experiments we create a LED data set with 10000 examples. The data are generated by the program described in section 4.1 with the probability of noise $p_{\text{noise}} = 1\%$.

We consider predictors based on the signed predictability conformity measure (5). The graph with no characters on it corresponds to the idealized predictor and represents an unachievable ideal goal. In the idealized case we know the true distribution for data and use it instead of the backward kernel P_{σ^*} in both (3) and (5). The *pure hypergraphical conformal predictor* (the graph with circles) is obtained using the nontrivial hypergraphical model both when computing p-values (3) and when computing the conformity measure (5). Analogously we use the exchangeability model to obtain the *pure exchangeability conformal predictor* (the

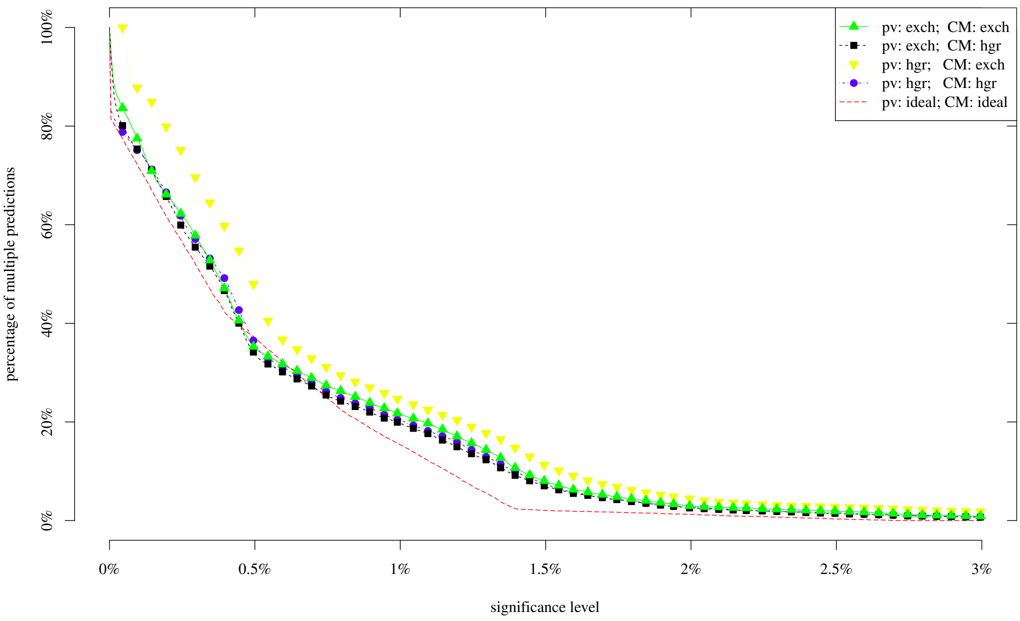

Figure 2 The final percentage of multiple predictions for significance levels between 0% and 3%. The results are for the LED data set with 1% of noise and 10000 examples.

Table 1 The final percentage of multiple predictions in Figure 2 for the significance level 1% and for the graphs with squares and circles.

Seed (10^4)	0	1	...	99	Average	St. dev.
pv: exch; CM: hgr	0.197	0.243	...	0.248	0.192	0.052
pv: hgr; CM: hgr	0.203	0.244	...	0.250	0.196	0.049

graph with triangles point up). The two *mixed conformal predictors* (the graphs with squares and triangles point down) are obtained when we use different models to compute the p-values and the conformity scores. The intuition behind the pure and mixed conformal predictors can be explained using the distinction between hard and soft models made in [6]. The model used when computing the p-values (3) is the hard model; the validity of the conformal predictor depends on it. The model used when computing conformity scores (5) is the soft model; when it is violated, validity is not affected, although efficiency can suffer. The true probability distribution for our generated data conforms to both the exchangeability model and the nontrivial hypergraphical model; so all four conformal predictors are automatically valid, and we study only their efficiency. Figure 2 shows the percentage of multiple predictions $\mathrm{Mult}^{\epsilon}_{10000}/10000$ as function of the significance level $\epsilon \in [0, 0.03]$. In the legend, the hard model used is indicated after "pv" (the way of computing the p-values), and the soft model used is indicated after "CM" (the conformity measure); "exch" refers to the exchangeability model, and "hgr" refers to the nontrivial hypergraphical model. The most interesting graph is the one with squares, corresponding to the exchangeability model as the hard model and the nontrivial hypergraphical model as the soft model. The performance of the corresponding conformal predictor is typically better than, or at least close to, the performance of any of the remaining realistic predictors. The fact that the validity of the conformal predictor only depends on the exchangeability assumption makes it particularly valuable. The graph with triangles point down corresponds to the nontrivial hypergraphical model as the hard model and the exchangeability model as the soft model; the performance of the corresponding conformal predictor is very poor in our experiments.

Table 1 shows the percentage of the multiple prediction at the significance level 1% for two graphs (with squares and with circles) for several seeds of the pseudorandom number generator. The values of the seed are given in the units of 10,000 (so that 0 stands for 0, 1 for 10,000, 2 for 20,000, etc.). The column "Average" gives the average of all the 100 values, and column "St. dev." gives the standard estimate of the standard deviation computed from those 100 values. The table confirms that the graphs are very close on average (see the penultimate column), but the accuracy of our experiments is insufficient to say which tends to be lower (see the last column).

5 Conclusion

The main finding of this paper is that nontrivial hypergraphical models can be useful for conformal prediction when they are true. More surprisingly, in our experiments they only need to be used as soft models; the performance does not suffer much if the exchangeability model continues to be used as the hard model. This interesting phenomenon deserves a further empirical study.

Acknowledgements We thank the COPA 2013 reviewers for comments that improved the results of the paper. We are indebted to Royal Holloway, University of London, for continued support and funding. This work has also been supported by: the EraSysBio+ grant SHIPREC from the European Union, BBSRC and BMBF; a VLA grant on machine learning algorithms; a grant from the National Natural Science Foundation of China (No. 61128003); a grant from the Cyprus Research Promotion Foundation (research contract TPE/ORIZO/0609(BIE)/24); grant EP/K033344/1 from EPSRC.

References

1 K. Bache and M. Lichman. UCI machine learning repository. School of Information and Computer Sciences, University of California, Irvine, CA, USA, 2013.

2 R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer, New York, 1999. Reprinted in 2007.

3 V. Fedorova, I. Nouretdinov, and A. Gammerman. Testing the Gauss linear assumption for on-line predictions. *Progress in Artificial Intelligence*, 1:205–213, 2012.

4 V. Fedorova, I. Nouretdinov, A. Gammerman, and V. Vovk. Conformal prediction under hypergraphical models. In *Proceedings of the Ninth International Conference on Artificial Intelligence Applications and Innovations (AIAI 2013)*, Paphos, Cyprus, 2013. To appear, available at www.alrw.net/articles/09.pdf.

5 V. Vovk, A. Gammerman, and G. Shafer. *Algorithmic Learning in a Random World*. Springer, New York, 2005.

6 V. Vovk, I. Nouretdinov, and A. Gammerman. On-line predictive linear regression. On-line Compression Modelling project (New Series), Working Paper 1, May 2005.