UC Berkeley

Building Efficiency and Sustainability in the Tropics (SinBerBEST)

Title

Model Assessment with Renormalization Group in Statistical Learning

Permalink

<https://escholarship.org/uc/item/8fb30339>

Author

GingQuo, Wang

Publication Date

2013-06-12

Peer reviewed

Model Assessment with Renormalization Group in Statistical Learning

Qing-Guo Wang, *Member, IEEE*, Chao Yu and Yong Zhang

Abstract—This paper proposes a new method for model assessment based on Renormalization Group. Renormalization Group is applied to the original data set to obtain the transformed data set with the majority rule to set its labels. The assessment is first performed on the data level without invoking any learning method, and the consistency and non-randomness indices are defined by comparing two data sets to reveal informative content of the data. When the indices indicate informative data, the next assessment is carried out at the model level, and the predictions are compared between two models learnt from the original and transformed data sets, respectively. The model consistency and reliability indices are introduced accordingly. Unlike cross-validation and other standard methods in the literature, the proposed method creates a new data set and data assessment. Besides, it requires only two models and thus less computational burden for model assessment. The proposed method is illustrated with academic and practical examples.

I. INTRODUCTION

In statistical learning, given a data set, one can always fit it to some model. It is a very challenging problem to assess and select a model. C_p statistic, Akaike information criterion (AIC) and Bayesian information criterion (BIC) [1] are well-known model assessment methods. They estimate in-sample error for model selection. However, the assessment based solely on in-sample error may lead to an overfitting problem. Cross-Validation (CV) and the Bootstrap method try to estimate the out-of-sample error. But both the K -fold CV and the Bootstrap method need to split data into K sets and the resulting data sets are partially overlapped. They require the training algorithm to be run for K times, which will increase the computational burden as much as K times [1].

There seems less recent progress on model assessment recently. In this paper, we present a totally new method for data and model assessment using Renormalization Group (RG). RG was first proposed to study the critical phenomena in the quantum field by Kenneth G. Wilson, who won the Nobel Prize for physics in 1982 [2], due to this great contribution.

***This research is funded by the Republic of Singapore's National Research Foundation through a grant to the Berkeley Education Alliance for Research in Singapore (BEARS) for the Singapore-Berkeley Building Efficiency and Sustainability in the Tropics (SinBerBEST) Program. BEARS has been established by the University of California, Berkeley as a center for intellectual excellence in research and education in Singapore.**

Q.-G. Wang is with the Department of Electrical and Computer Engineering, National University of Singapore, 119260, Singapore (email: elewqg@nus.edu.sg)

C. Yu is with the Department of Electrical and Computer Engineering, National University of Singapore, 117576, Singapore (email: yuchao@nus.edu.sg)

Y. Zhang is with the Qiming Venture Partners, Shanghai, 200121, China (email: yong.zhang@qimingvc.com)

RG designs some Renormalization Group transformation (RGT) to relate macroscopic physics quantity to microscopic one and invokes “scale invariance” to solve the problem.

Technically, the proposed method groups the given data set into a RG data set, train one model with the given data and another model with the RG data, and compare their predictions. The consistent predictions between two models indicates informative data and reliable models. The contributions of this article can be summarized as follows.

- The proposed method produces a new data set from the given data set with the members of the former different from those of the latter, whereas the existing methods mentioned above only divide/separate the given data and the data points in the new sets are the same as those in the given set, and thus creates no new data.
- An assessment is made at the data level before any learning method is applied to train a model, whereas the domain works carry out model assessment only. For this, we introduce new indices to quantify the consistency of two data sets and non-randomness of the given data. The required computation is extremely fast as it involves no learning. With this assessment, if the given data is random, it is meaningless to model it. Only when the data is not random, should one proceed to model it.
- At the model level, our assessment is to compare predictions of two models learnt from the give data and the transformed data, respectively. The prediction consistency and model reliability are defined accordingly. This assessment relies on two models one of which has much smaller data size and thus much less computational burden, whereas K -fold CV or similar methods train K models with K usually much greater than 2, typically set at 10.

The rest of this article is organized as follows. Section II presents the proposed method. Section III details the implementation issue. Section IV addresses the theoretical issues with assessment criteria. Section V gives simulation studies. Section VI concludes the paper.

II. THE PROPOSED METHOD

Consider the classical binary classification problem in statistical learning with a data set, $\mathbf{S} = \{S_1, S_2, \dots, S_N\}$, with $S_i = (x_i, y_i)$, $i = 1, 2, \dots, N$, $x_i \in \mathbb{R}^p$ and $y_i \in \{-1, 1\}$. A learning method is to determine the boundary function $f(x)$. The present paper is not to develop a new learning method, but to propose a new approach based on RG to assess the degree of randomness of the data and reliability of a model built from a learning method for prediction, assuming that

the data are from a process with a fixed probability density. The core of this approach is the “scale invariance” property of RGT. It follows from this property that the information contained in the original data set is invariant under RGT, that is, the transformed data set obtained from RGT should contain the same information as in the original data set. In the context of statistical learning, the information of the data set refers here to the probability density function, or the relationship from x to y . Its two extremes are the purely random and completely deterministic cases. We will define and compute an index on measure of such information on the data set. Furthermore, a model learnt from the data set reflects its information. If the information is same for the original and transformed data sets, then two models learnt from them, respectively, should be consistent with each other in terms of their predictions. Hence, we will also define and compute an index on measure of model consistency. It is found that this index will not be zero for the pure random data. This makes it necessary to find the index with the case of completely randomized labels and re-scale it for a general case to define another index on measure of model reliability. The idea and steps of the proposed approach will be described and illustrated by a 2D example in this section, while design of RGT and relevant indices will be discussed in the next two sections, respectively.

The first step of the proposed method is to perform a RGT on the given data set to obtain a transformed data set. For easy reference, we call the given data set, $\mathbf{S}$, as the fine data set and the model obtained from it as the fine model. The data set $\hat{\mathbf{S}} = \{\hat{S}_1, \hat{S}_2, \dots, \hat{S}_N\}$, with $\hat{S}_j = (\hat{x}_j, \hat{y}_j)$, $j = 1, 2, \dots, \hat{N}$, $\hat{x}_j \in \mathbb{R}^p$ and $\hat{y}_j \in \{-1, 1\}$ obtained from a RGT on the fine data set is called as the coarse data set and the resulting model as the coarse model. A RGT on the fine data set is to group a number of data points of the fine data set into one data point in the coarse data set in a systematic way. There are different ways to do this grouping, which will be discussed in detail in the next section. One example of RGT is to group all the data of the fine set in a geometric unit of the fixed shape and size to one point of the coarse set. Generally, one RGT transforms the fine set to many groups and one group will become one data point in the coarse set. For each group, one has to specify its representative label ($\hat{y}$) and its representative feature values ($\hat{x}$) to define one data point in the coarse set. The appealing way to assign the group label seems the majority rule of the labels of the points in that group, while its representative $\hat{x}$ may be determined by the simple average of the x_i of the major class. For example, Fig. 1a and Fig. 1b show such a group of 10 points in the fine set and the resulting point in the coarse set using the above rules.

For clarity and illustration, we construct a 2D example to explain each step of the proposed method in detail with numerical results. Let the fine data have two features, $x_i = (x_i(1), x_i(2))$. Let the fine set have 4000 data points in the square of $[-1, 1]$, which are drawn randomly with uniform distribution. Specify a boundary function as $x_i(2) = x_i^2(1) - 0.5$. Consider first the deterministic case where each data

(a) A group of data in the fine set. (b) One point in the coarse set.

Fig. 1. RGT.

point is assigned with the green label if it is located on and above the parabola, otherwise with the red label. The resulting fine set is depicted in Fig. 2a. Suppose that we use the majority and average rules mentioned above. The coarse set is then obtained and shown in Fig. 2b.

The second step of the proposed method is to assess the information of the given data. Suppose that the label for a coarse unit is assigned according to the majority rule and the coarse label for each fine data point in this unit is defined as the same as its unit label. This enables us to compare the fine and coarse labels at each fine data point and define the data consistency as follows:

$$C_D = 1 - \frac{1}{2} \frac{\sum_{i=1}^N |L(x_i) - \hat{L}(x_i)|}{N}, \quad (1)$$

where $L(\cdot) = 1$ or -1 , stands for the fine labels and $\hat{L}(\cdot) = 1$ or -1 for the coarse labels. A high index value means a high extent of uniformity of two data sets, which should imply that the given data is informative. For the above example, the data consistency index is 0.9818, indicating a high consistency of two data sets.

The third step of the proposed method is to apply some learning method on both fine and coarse data sets to get the fine and coarse models, $f(x)$ and $\hat{f}(x)$, respectively. Any learning method can do but the same one should be used on both data sets to avoid inconsistency due to application of different methods. For this study, the support vector machines (SVM) is chosen as our classification tool, since it is popular and representative and has many attractive features and emphatic performance in many applications [3]. In our simulation, the LibSVM kit [4] is employed for the above 2D example and produces the fine and coarse models in Fig. 2 marked with black curves. Then we assess the reliability of the fine model, comparing predictions of the fine model with those of the coarse model. A model consistency index C_M is defined as

$$C_M = 1 - \frac{1}{2} \frac{\sum_{i=1}^N |\text{sign}(f(x_i)) - \text{sign}(\hat{f}(x_i))|}{N}, \quad (2)$$

where $\text{sign}(f(x_i))$ and $\text{sign}(\hat{f}(x_i))$ are the prediction labels of $f(x_i)$ and $\hat{f}(x_i)$ on x_i in the fine data set, respectively. A high index value means a high degree of agreement of predictions of two models and indicates a high similarity of two models, which should imply that the learnt models

Fig. 2. Deterministic 2D case.

are reliable. The model consistency index also captures the information contained in the data. For our example, the model consistency index is 0.9970, indicating a very high reliability of the models, which is of course true, as the data are constructed with a perfect boundary function.

Next, we consider a purely random data case for comparison (more realistic cases will be shown in Section V). Take all x_i from the above case or randomly choose a new set of 4000 of x_i , which does not matter for this study. Assign now their labels randomly, which yields the data shown in Fig. 3a. The majority and average rules above are applied to get the coarse set shown in Fig. 3b. The data consistency index is 0.6282, a much smaller value, which actually indicates that the given data is not informative as will be shown in Section IV. Fig. 3 also shows the fine and coarse models marked with black curves. For this case, the model consistency index is

Fig. 3. Pure random 2D case.

0.6743, a much smaller value compared with the previous case. The model is not reliable though the consistency index is not equal to zero but about the lowest, which will be shown in Section IV. This is also obvious since the boundary functions in Fig. 3 are messy and useless.

III. DESIGN OF RGT

A RGT is a way to group the fine data and choose a representative in each group [5]. For our problem of assessment of data-driven models, we present two broad methods of grouping data for RGT based on geometric and distributional considerations in the feature space, respectively.

A. GEOMETRICAL GROUPING

Choose a geometric unit of fixed size. For the 2D case, possible geometric units include, but not limited to, a square,

rectangle, and triangle. It is trivial to have their n -dimensional counterparts. We have used squares in the previous section for a 2D example. We can have a cube for 3D and a hypercube for n -D. Suppose a hypercube for illustration. Fill the given feature space with a grid of hypercubes of equal size, one by one without gap or overlap. Include in the coarse set a non-empty hypercube which has at least one fine data point in it and exclude all empty hypercubes. The resulting non-empty hypercubes are numbered as $j = 1, 2, \dots, \hat{N}$.

The next task of RGT is to form a coarse data set. One has to specify its features, $\hat{x}_j$ and label, $\hat{y}_j$, for unit j , based on all $S_i = (x_i, y_i)$ of the fine set in the same unit. The label, $\hat{y}_j$, for unit j , is usually determined according to the majority rule on all (x_i, y_i) of the fine set in this same unit. Let unit j have N_j^+ of such x_i with $y_i = 1$ and N_j^- of x_i with $y_i = -1$. Then the majority rule is defined as follows.

- if $N_j^+ > N_j^-$, then $\hat{y}_j = 1$;
- if $N_j^+ < N_j^-$, then $\hat{y}_j = -1$;
- if $N_j^+ = N_j^-$, then $\hat{y}_j$ is the same as the first counted y_i in unit j .

Now turn to the issue of setting features. We calculate the average of all x_i in units with the major class to get $\hat{x}_j$.

It follows from the above description of RGT that the major parameter of a RGT is the number of units, or the size of a unit in each geometric grouping method. It depends on the transformation ratio, r , which we define as the number of geometric units (the data points in the coarse set) to the number of data points in the fine set. If this ratio is small, there will be more fine data points in each unit on average. The majority makes good sense with less chance error, but it will give fewer data points in the coarse set, which may have negative effect on modeling accuracy. On the other hand, a large ratio will make less sense of majority but have more coarse data points. In the previous example, the side of each small square is chosen 0.1 and the ratio is approximately to 1/10. To see effects of r , we vary r and run simulations for the 2D example in the previous section. It follows that for the deterministic case, the consistency indexes, $\mathcal{C}_D = 0.9700, 0.9818, 0.9515$ and $\mathcal{C}_M = 0.9985, 0.9970, 0.9922$, for $r = \frac{1}{5}, \frac{1}{10}, \frac{1}{20}$, respectively, while for the pure random case, $\mathcal{C}_D = 0.5740, 0.6282, 0.6560$ and $\mathcal{C}_M = 0.7107, 0.6743, 0.6275$, for $r = \frac{1}{5}, \frac{1}{10}, \frac{1}{20}$, respectively. It is seen that the ratio in the range of $\frac{1}{5} - \frac{1}{20}$ does not affect the assessment much. In the rest of this paper, the ratio of $\frac{1}{10}$ is adopted.

B. DISTRIBUTIONAL GROUPING

We may also design a RGT based on data distribution by using some clustering method. For example, the k -means clustering algorithm may be adopted. The k -means clustering groups N observations into k clusters [6].

When a clustering method is applied to group the fine data to clusters, the resulting clusters play the same role as the units in the geometrical grouping. k/N is equal to the transformation ratio r and k is chosen based on the same guidelines as for the latter. And the rules to determine the

features and labels of a cluster to form a coarse data point can be also same as those for the units before.

To see how the clustering method works as RGT, let us re-visit the 2D example in the previous section. Take $k/N = 0.1$, which is equivalent to the transformation ratio of 0.1, used before. The fine data set and the fine model are shown in Fig. 4a (same as Fig. 2a). Perform the k-means clustering algorithm, which produces 400 clusters. The corresponding coarse data and coarse model are shown in Fig. 4b. The consistency indexes are computed as $\mathcal{C}_D=0.9832$

Fig. 4. The k-means clustering algorithm—deterministic.

and $\mathcal{C}_M=0.9960$, almost same as in Section II, very high values. The fine and coarse data and models for the pure random case are shown in Fig. 5. Its consistency indexes are

Fig. 5. The k-means clustering algorithm—pure random.

found to be $\mathcal{C}_D=0.6300$, nearly the same as in Section II, and $\mathcal{C}_M=0.4973$, a low value, which is lower than before.

IV. ASSESSMENT CRITERIA

With a RGT, one gets a coarse data set. It is possible to compare it with the fine set to assess the information contained in the fine data. When a learning method is applied to both data sets, one obtains two models and can compare their predictions to assess reliability of the fine model. This section will develop relevant indices to measure data information and model reliability with illustrations.

A. DATA INFORMATION

There are many different learning methods available to train a model from a data set. Each method has its assumptions and tuning parameters, and will inevitably bring in its error or bias. Thus, the performance of any learnt model will depend, not only on the data, but also on the method and

parameters chosen. It will be highly desirable to analyze only data to know how informative it is, regardless of a learning method adopted. It would be meaningless to model a data set if it is purely random. To find a suitable information index on the fine data, we look at how random the labels assigned by the majority rule for the coarse data are. Imagine that if the fine data is purely random, then the labels on the fine data are random, and each unit or cluster in the coarse feature space has $1/r$ fine data points with equal number of each label on average, which gives rise to equal probability for each label of the coarse unit/cluster. In other word, labels of the coarse data are random. Suppose that the coarse labels for all the fine data points in a coarse unit/cluster are same and defined as that of the majority. One can then compare the coarse label with the fine label on each fine data point. Their consistency will be a low value determined by chance.

On the other hand, if the fine data is deterministic, then the labels on the fine data are divided by the boundary function, and each unit or cluster in the coarse feature space has $1/r$ fine data points with the same label, which in turn assigns that label to the coarse unit/cluster, unless the unit/cluster intersects with the boundary function. In other word, labels of the coarse data are deterministic except for boundary units/clusters. Then, when one compares two labels of two data sets at each fine data point, their consistency will be almost one. In the view of the above observations, we thus define the consistency index on the given fine data in (1).

To make the value of $\mathcal{C}_D$ in (1) more meaningful, we also evaluate the index for the corresponding random case. Given the fine data set $\mathbf{S}$, keep x_i unchanged, but assign their labels randomly. Apply the same RGT as before, which results in the same units/clusters as before, but the majority rule will assign a new set of coarse labels due to changes of labels of fine data. We then evaluate the consistency index on the so-formed fine data, denoted by $\underline{\mathcal{C}}_D$. This number shall give the lower bound for this consistency index. The non-randomness index of a given data set is then defined as

$$\mathcal{D} = \frac{\mathcal{C}_D - \underline{\mathcal{C}}_D}{1 - \underline{\mathcal{C}}_D} \times 100\%. \quad (3)$$

The smaller $\mathcal{D}$ is, the more randomly the data distributes. The larger $\mathcal{D}$ is, the less randomly the data distributes. For the 2D example, the deterministic case yields $\mathcal{D} = 95.10\%$ from (3) with $\mathcal{C}_D = 0.9818$ and $\underline{\mathcal{C}}_D = 0.6282$ from Section II. The deterministic case gives a high value of $\mathcal{D}$, indicating a high non-randomness of the data, whereas the pure random case gives $\mathcal{D}$ equal to 0.

It is useful to know how the index $\mathcal{D}$ behaves with regard to randomness of data, hopefully it functions from 0 to 1 linearly. For the 2D example, suppose that $D\%$ denotes the percentage of the fine data points with the major class label. For instance, $D\%=80\%$ means that 80% of the data above $x_i(2) = x_i^2(1) - 0.5$ are randomly chosen and assigned with '1' while the remaining 20% data with '-1', and a similar assignment is done for the data below the parabola. Let $D\%=50\%, 60\%, 70\%, 80\%, 90\%$ and 100% , respectively. For each value, we apply the RGT and compute $\mathcal{D}$. The

results are shown in Fig. 6a with crosses, where the solid line is its linear fitting. Construct similarly the 3D case as follows. Take 80000 data points in a cube of $[-1,1]$ uniformly and apply a RGT to group the cube into 8000 identical units. Specify the boundary function as the paraboloid $x_i(3) = x_i^2(1) + x_i^2(2) - 0.6625$ and assign their labels to a given level of $D\%$. Calculate then $\mathcal{C}_D$, $\mathcal{C}_M$ and $\mathcal{D}$ and show $\mathcal{D}$ vs $D\%$ in Fig. 6b. It follows from Fig. 6 that $\mathcal{D}$ functions

Fig. 6. Non-randomness indices and linear fitting curves.

almost linearly and gives a good gauge of non-randomness.

B. RELIABILITY INDEX

In practice, one needs to use some learning method to train a model. Thus, it requires evaluation of models, in addition to data assessment. If the fine and coarse models are trained from the purely random data, then their prediction consistency will be a low value. However if the models are trained by the deterministic data, then their prediction consistency will be almost one. Thus, the consistency index $\mathcal{C}_M$ on the two models is defined in (2). Like the data assessment, we also evaluate this index for the corresponding random case. Keep x_i unchanged, but assign their labels randomly. Apply the same RGT and learning method as before, the models will change due to changes of labels of data. We then evaluate the consistency index between the so-formed models: $\mathcal{C}_M$. This number shall give the lower bound for this consistency index. Then we define the reliability index as

$$\mathcal{R} = \frac{\mathcal{C}_M - \mathcal{C}_M}{1 - \mathcal{C}_M} \times 100\%. \quad (4)$$

The smaller $\mathcal{R}$ is, the less reliable the models are. The larger $\mathcal{R}$ is, the more reliable the models are. For the 2D example, the deterministic case yields $\mathcal{R} = 99.08\%$ from (4) with $\mathcal{C}_M = 0.9970$ and $\mathcal{C}_M = 0.6743$ from Section II. The deterministic case gives a high value of $\mathcal{R}$, which shows a high reliability of the models. On the other hand, the pure random case gives $\mathcal{R}$ equal to 0.

We also want to know how the index $\mathcal{R}$ behaves with regard to randomness of data, hopefully it functions from 0 to 1 linearly. For the 2D example, let $D\% = 50\%, 60\%, 70\%, 80\%, 90\%$ and 100% . For each value, we apply the RGT and the learning method to compute its $\mathcal{R}$. The results are shown in Fig. 7a with crosses, where the solid line is its linear fitting. The same procedure is performed on the 3D

Fig. 7. Reliability indices and linear fitting curves.

case above and $\mathcal{R}$ vs $D\%$ is shown in Fig. 7b. It follows from Fig. 7 that $\mathcal{R}$ functions almost linearly and makes a good sense of reliability.

The proposed method is very different from the cross validation (CV) and will be now compared with the latter, since CV is the most popular model assessment methodology in the learning domain. For the 2D case, Table I shows the non-randomness index $\mathcal{D}$, the reliability index $\mathcal{R}$, the correct prediction rate of 5-fold CV and 10-fold CV, and their respective computational time for data sets. At the data level, the proposed method is super fast to get $\mathcal{D}$ and is independent of any learning method. At the model level, our method only needs to train two models and takes short computational time whereas a K -fold CV needs to train K models. Therefore, our method is more efficient. Also, it is observed from Table I that $\mathcal{D}$ and $\mathcal{R}$, and the correct prediction rates of CV have a similar rising trend vs $D\%$. To see more clearly, the indices $\mathcal{C}_D$, $\mathcal{C}_M$, $\mathcal{D}$ and $\mathcal{R}$, and the correct prediction rates of 5-fold CV and 10-fold CV are shown in Fig. 8. It follows that the

Fig. 8. Indices and CV.CorrectRates vs $D\%$.

correct prediction rate of CV has the same tendency as $\mathcal{C}_D$, $\mathcal{C}_M$, $\mathcal{D}$ and $\mathcal{R}$. But note that unlike CV which splits data to folds with overlapping, our method produces new data from the original data, which avoids the repeated use of data.

V. SIMULATION EXAMPLES

In this section, practical examples are given to illustrate the proposed method.

Example 1. Consider the banana data [7]. It has 5300 data points with two features. The fine data with normalization is shown in Fig. 9a. RGT is done by geometrical grouping with squares of side of 0.1. We have 233 identical squares and

TABLE I
COMPARISONS BETWEEN THE PROPOSED METHOD AND CV

$D\%$	Proposed method				5-fold CV		10-fold CV	
	$\mathcal{D}$	Time	$\mathcal{R}$	Time	rate	Time	rate	Time
50%	1.05%	0.08s	3.19%	85.42s	0.5200	265.33s	0.5155	623.02s
60%	5.81%	0.08s	11.61%	80.60s	0.5363	233.98s	0.5293	608.37s
70%	22.35%	0.08s	35.74%	70.70s	0.5938	194.91s	0.5962	518.59s
80%	44.73%	0.08s	59.93%	51.57s	0.6793	125.26s	0.6910	378.81s
90%	68.93%	0.08s	79.28%	14.18s	0.8135	31.25s	0.8190	88.70s
100%	95.10%	0.08s	97.48%	2.16s	0.9955	4.57s	0.9965	10.22s

the transformation ratio approximately equals to $r = 0.04$. The coarse data under the default RGT of Section III is obtained and shown in Fig. 9b. It follows from Section IV

Fig. 9. The banana example.

that $\mathcal{C}_D = 0.8983$, $\mathcal{C}_D = 0.5764$ and $\mathcal{D} = 75.99\%$, indicating a high non-randomness of the data. With the SVM, the fine and coarse models are trained on two data sets, and exhibited in Fig. 9a and Fig. 9b with black curve, respectively. It follows that $\mathcal{C}_M = 0.9551$. With label randomization, one gets $\mathcal{C}_M = 0.6713$. Thus, the reliability index for this example is calculated from (4) as $\mathcal{R} = 86.34\%$, which implies a reliable fine model. The SVM model given in [7] has the prediction accuracy of 88.44%.

Example 2. Consider the “Astroparticle Physics” data [8]. It has 7089 data points with four features. We apply the k-means clustering method to do the grouping. With 709 clusters and the transformation ratio of $r = 0.1$, it follows that $\mathcal{C}_D = 0.9595$, $\mathcal{C}_D = 0.6194$ and $\mathcal{D} = 89.36\%$, indicating a high non-randomness of the data. With the SVM, the fine and coarse models are trained on two data sets. It follows that $\mathcal{C}_M = 0.8965$. With label randomization, one gets $\mathcal{C}_M = 0.6239$. Thus, the reliability index for this example is calculated from (4) as $\mathcal{R} = 72.48\%$, which implies that the reliability of the fine model is good. The raw SVM model in [8] has the prediction accuracy of 75.2%.

VI. CONCLUSION

In this paper, a new model assessment method has been proposed based on RG. Its implementation rules and performance criteria are presented accordingly. By RG, a coarse data set is created from the original fine data set and it is different from the latter in terms of their features and

labels. This new data set enables us to assess data information without using any learning method. The proposed method does not require a separate validation data set. If the assessment shows that the data is random, the modeling or learning is useless and should not be performed at all. These characterize the proposed method and distinguish it from the existing methods in the domain. In addition, a model is trained on the coarse set, which is impossible with other methods which only divide the data to different sets with the same data points as the original ones. The predictions from the coarse model are compared with those of the model based on the fine data to evaluate model consistency and thus reliability of the learnt model. The proposed method is shown to work well with examples. It should be pointed out that the proposed method requires sufficient data. Furthermore, its theoretical properties and additional potential values are to be investigated.

In this paper, we illustrate the proposed method for classification problem. The method can be also applied to regression problem with trivial modifications. In this case, simple averaging or a weighted average of responses, y_i , of fine points in a unit may be used to obtain the coarse response, $\hat{y}_j$, instead of the majority rule for classification, whereas the coarse feature can be determined in the same way as in classification. The comparison of fine and coarse data and models can be made with regard to, say, the standard squared errors for regression.

REFERENCES

- [1] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*. New York: Springer, 2009.
- [2] K. G. Wilson, “Renormalization Group and Critical Phenomena. I. Renormalization Group and the Kadanoff Scaling Picture,” *Physical Review B*, vol. 4, no. 9, p. 3174, 1971.
- [3] S. R. Gunn, “Support Vector Machines for Classification and Regression,” *ISIS Technical Report*, vol. 14, 1998.
- [4] C. C. Chang and C. J. Lin, “LIBSVM: A Library for Support Vector Machines,” *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.
- [5] D. Sornette, *Why Stock Markets Crash: Critical Events in Complex Financial Systems*. Princeton Univ Pr, 2004.
- [6] A. K. Jain, “Data Clustering: 50 years beyond K-means,” *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.
- [7] H. Chen, P. Tino, and X. Yao, “Probabilistic Classification Vector Machines,” *IEEE Transactions on Neural Networks*, vol. 20, no. 6, pp. 901–914, 2009.
- [8] C. W. Hsu, C. C. Chang, C. J. Lin, and Others, “A Practical Guide to Support Vector Classification,” 2003.