Controversy Rules — Discovering Regions Where Classifiers (Dis-)Agree Exceptionally

Oren Zeev-Ben-Mordehai, Wouter Duivesteijn, Mykola Pechenizkiy

arXiv:1808.07243v1 [cs.LG] 22 Aug 2018

***Abstract*—Finding regions for which there is higher controversy among different classifiers is insightful with regards to the domain and our models. Such evaluation can falsify assumptions, assert some, or also, bring to the attention unknown phenomena. The present work describes an algorithm, which is based on the Exceptional Model Mining framework, and enables that kind of investigations. We explore several public datasets and show the usefulness of this approach in classification tasks. We show in this paper a few interesting observations about those well explored datasets, some of which are general knowledge, and other that as far as we know, were not reported before.**

***Index Terms*—Classification Confusion, Error Analysis, Subgroup Discovery, Exceptional Model Mining**

I. Introduction

In the recent flood of papers analyzing the details of the inner workings of classifiers [22], [11], [8], [20], the attention typically is focused on a single classifier. We might want to know how a black-box classifier arrives at its predictions [11], [20], where the classifier predicts well or badly [8], which input attributes influence the output predictions [22] and to which degree. Important as that might be, we propose that more can be learned by investigating the collective behavior of a *set* of classifiers. Let us illustrate this with a practical example.

Suppose that we work at a bank, and we have to decide on whether or not to lend a mortgage to a series of customers. We have a rule-based system in place to make this decision. Since the economic tide ebbs and flows over time, we may need to adapt the rule-based system periodically, to achieve appropriate results. On every point in time, the system can predict for every customer whether the person gets the mortgage or not. Interesting would be to find out when and why the rule-based system changes its mind: if subsequent iterations of the system suddenly grants a loan to a previously rejected customer, or vice versa, the era of responsible data science compels us to properly motivate why. Ideally, we would not just identify single customers for which this holds, but coherent groups of customers that come with a concise description: it would be interesting to know if the system has changed its mind about granting mortgages to people under the age of thirty with at least two kids, for example. Such descriptions give us more information on whether the behavior displayed by the system is, in fact, desirable.

In this paper we introduce few variants of the following problem. **Given a dataset, and a collection of relevant classifiers, identify and name the regions of the domain for which there is a high disagreement**. We describe an algorithm which is based on the Exceptional Model Mining framework [16], [7], and provide quality measures to address a few possible motivations and preferences. We evaluate the usefulness of the algorithm on publicly available datasets and bring qualitative and quantitative findings.

II. Related work

Given a classifier and a relevant dataset, investigating the interactions among the model and the data is often referred to as model debugging, providing model transparency, or also model interpretability. Some interpretability mechanisms treat a model as a black box [11], [12], [20], [1] while other employ methods that are tailored to specific classification techniques. The algorithm described in [8], enables to investigate a single soft classifier against a dataset. It requires that the ground truth is provided, and also that the model outputs probabilities (it is a soft classifier). The method investigates the degree to which the ranking of the model in a specific subgroup is in agreement with the ground truth. This is done by counting the obvious errors (when a negative is ranked before a positive). Regions for which the rate of obvious errors is significantly higher, or significantly lower, from the same measure for the whole dataset, are then reported.

Black box auditing, or discrimination aware approach, is described in [1]. GoldenEye/GoldenEye++ [11], [12] is highlighting the feature importance/feature interaction by shuffling the values in columns of specific features within a predicted label, and measuring the label changes. EXPLAIN [22], [23] checks the effect of blinding the model with respect to values of a specific attribute. SHAP [18] divides the contribution to a classification among the features of a case. LIME [20] attempts to describe the model in a locality of a case under scrutiny using an interpretable proxy. Also interesting approaches for interpretability explore regions of uncertainty [9], attention given a case under test [25], or cognitive psychology traits of the model [21]. Some works extract rules, or provide a simplified model [17], [15], [10].

If one manages to compress the model and the data, for example following the *Minimum Description Length* (MDL) framework [4], then they somehow capture the essence of the model/data. One should describe a model and then the exceptions in the data that do not follow the model and to find the point for which the overall description is minimal (the model is described in sufficient detail and the leftover exceptions are few).

III. Prerequisites

Assume given a dataset DS from a domain $\mathcal{D}$, consisting of m *cases* (or *records* of the form $r = (\text{attr}_1, \text{attr}_2, \ldots, \text{attr}_k, \ell)$. We refer to the final element of each case, as *target*, or also the *true label*. The target is nominal, with the set of possible values $\mathcal{C}$, thus $\ell \in \mathcal{C}$. All other elements of each case are referred to as the *attributes*, which can be either of a numeric type, or a nominal type. While the domain of each individual attribute is left free, we denote the collective domain of the k attributes by $\mathcal{A}$. This notation allows us to formally define what a classifier is:

Definition 1 (Classifier)**.** *Given a domain $\mathcal{D}$ with collective attribute domain $\mathcal{A}$ and the nominal type of the target $\mathcal{C}$, a classifier C is a function $C : \mathcal{A} \to \mathcal{C}$, assigning a label to every possible input value from $\mathcal{A}$.*

The main goal of a classifier, as it is generally understood in machine learning, is to predict: assigning labels to cases whose real target value we do not know. To arrive at a formal definition of such predictions, we need to introduce some more notation. Let DS be a dataset, where the true labels, in the general case, are not known. We denote by superscript i the ith case of the dataset DS or elements thereof. Hence, the first case is denoted by r^1, the target value of the seventh case, whether it is known or not, by ℓ^7, and the value for the fourth attribute in the eighth case by attr_4^8.

Definition 2 (Predictions)**.** *Given a dataset DS consisting of m cases, and a classifier C, we define the* predictions *of C on DS to be the vector $(y_C^1, \ldots, y_C^m) \in \mathcal{C}^m$, where $y_C^i = C(attr_1^i, \ldots, attr_k^i)$.*

Hence, the vector of predictions collects the outputs of the classifier function C on all cases in the dataset DS.

The main goal of this paper is to find regions, or subgroups of cases, of high controversy across a set of classifiers. Hence, we assume as given a set of n classifiers $\{C_1, \ldots, C_n\}$. For the purposes of this paper, it is irrelevant exactly how any of these classifiers arrive at their predictions: we are agnostic of the internal workings of a classifier function. Instead, we merely analyze them in terms of their predictions:

Definition 3 (Prediction matrix)**.** *Given a dataset DS consisting of m cases, and a set of n classifiers $\{C_1, \ldots, C_n\}$, the* prediction matrix *M is the $(m \times n)$-matrix with entries from $\mathcal{C}$ defined by:*

$$M_{ij} = y_{C_j}^i$$

Hence, the first row of the prediction matrix M collects the predictions of all n classifiers for the first case in the original matrix DS, etcetera.

A. Local Pattern Mining

We would like to identify one or more subgroups of the cases, for example $SG \subset DS$, such that on average for cases $i \in SG$, and the n classifiers $j \in (1 \ldots n)$, there is high controversy among the relevant entries M_{ij}. The ground truth, GT, for the classification problem, referred above as the target values, or the true labels, is not a necessity for the problem we describe next in its basic form, yet once present, new options and questions can be investigated.

When selecting a subset of the cases in DS, we restrict ourselves to regions that can be identified with a description $desc$ that belongs to a description language $\mathcal{L}$. Thus for example, if $desc_1 \equiv \text{Class} = 3 \wedge \text{Age} < 5$ is a valid description in $\mathcal{L}$, then the matching subgroup of cases SG_{desc_1}, those for which the description $desc_1$ evaluates to true, is a valid candidate as a subgroup. This is often the approach with Subgroup Discovery [14], and with Exceptional Model Mining [16], [7]. In *Subgroup Discovery* (SD), one can identify the most interesting subgroups w.r.t. a single target. With *Exceptional Model Mining* (EMM), one can address multiple target attributes when evaluating how exceptional a subgroup is. Both frameworks require that one declares a set of attributes that can be part of the *description* for a subgroup, therefore, the identification of the region. Also required is a single or, for EMM, a set of attributes, that are used when evaluating the exceptionality of the region. Formally both SD, and EMM, require a declaration of a subset of the attributes of the dataset, $\{a_1, \ldots, a_k, t_1, \ldots, t_m\}$, where $\{a_1, \ldots, a_k\}$ are used to describe subgroups, and $\{t_1, \ldots, t_m\}$ are used to evaluate subgroups. Thus the description language $\mathcal{L}$ is based on $\{a_1, \ldots, a_k\}$ and the relevant domains. Given a dataset DS, a description, $desc \in \mathcal{L}$, is interchangeable with the subgroup $SG_{desc} \subset DS$ that corresponds to the cases $i \in DS$ for which $desc(a_1^i, \ldots, a_k^i)$ is true. For evaluating the subgroups, as mentioned above, $\{t_1, \ldots, t_m\}$ are used. For SD, $m = 1$, for EMM, $m \geq 1$. Of course there are many ways to evaluate the exceptionality of a subgroup, for EMM in particular, but also for SD. Therefore a specific *quality measure* $\varphi : \mathcal{L} \to \mathbb{R}$ (for EMM, based on a *model class*) must be chosen to evaluate the quality of the region in terms of exceptionality. Hence, $\varphi(desc)$ assigns a value to the description $desc$ based on the $\{t_1, \ldots, t_m\}$ attributes of the relevant entries in SG_{desc}. A reasonable choice to realize the search involved with SD or EMM, is with the *Beam Search* algorithm [3].

IV. The Controversy Rules Model Class for EMM

Our prerequisites and the standard EMM terminology can be naturally mapped onto one another, as follows. The descriptors $\{a_1, \ldots, a_k\}$ from EMM will be $\{\text{attr}_1, \text{attr}_2, \ldots, \text{attr}_k\}$ of the dataset, and the targets $\{t_1, \ldots, t_m\}$ of EMM will be the n predictions from M. In some situations we augment M, where available and relevant, with the ground truth label ℓ.

We illustrate the core concept of Controversy Rules by a single case, or row, r^1 compared to another row r^2. If the set of n predictions over r^1 has higher entropy than the set over r^2, we would claim that r^1 is more interesting than r^2. We use here the base 2 Shannon entropy, $\mathcal{H} : \mathcal{C}^n \to [0, \infty)$:

$$\mathcal{H}(c_1, \ldots, c_n) = -\sum_{c \in \mathcal{C}} P(c) \cdot \log_2 P(c)$$

$$\text{where for every } c \in \mathcal{C}, P(c) = \frac{1}{n}\left(\sum_{i=1}^{n} \mathbb{1}(c_i = c)\right)$$

Table I: Toy dataset A

	C_1	C_2	C_3	C_4	
1.	1	1	0	1	
2.	0	1	0	1	*
3.	1	0	1	1	
4.	1	1	1	0	
5.	0	1	1	0	*
6.	0	0	0	0	
7.	0	0	0	0	
8.	0	0	0	1	

Table II: Toy dataset B

	C_1	C_2	C_3	C_4	
1.	1	1	0	1	*
2.	0	1	0	1	
3.	1	1	0	1	*
4.	1	1	0	1	*
5.	0	1	1	0	
6.	0	0	0	0	
7.	0	0	0	0	
8.	0	0	0	1	

Following this definition, if we have 10 classifiers ($n = 10$), and a binary target, then a row where five classifiers predict the one label and five the other, is more interesting than if the votes were six versus four. If we have 3 classes, then a tally of $(3, 4, 3)$ is as interesting as of $(4, 3, 3)$ and both are more interesting than a tally of $(1, 1, 8)$. This is of course if we look for regions with disagreement. If we seek for regions with high agreement, we prefer the lower entropy.

Now consider subgroups of the cases, or collections of rows. As to help the reader to follow the intuition, we give two M matrices for two toy datasets and their respective classifiers, in Table I and Table II. For simplicity, assume that a description exists for each subset of both toy datasets. Therefore one can name any of those subsets and evaluate their quality measures. Below we refer, for example, to the subgroup containing rows 1 and 3 as subgroup $\{1, 3\}$.

A. Row Controversy

In the first scenario, we seek regions with *high per row controversy* across the classifiers. We measure this by mean per row entropy over the cases in a subgroup. Therefore the quality measure that we use here is:

$$\varphi_{row}(SG) \equiv \frac{1}{|SG|} \sum_{i \in SG} \mathcal{H}(M_{i1}, \ldots, M_{in})$$

Note that we ignore the identity of the classifiers, or the actual predictions, and we just evaluate the mean per row entropy for the subgroup. We set a minimum threshold for number of rows, so that the reported subgroups are actionable, yet other from that, a smaller subgroup with higher mean entropy is still ranked before bigger subgroups with smaller mean entropy. The use case for this scenario is when we are interested in subgroups of the domain for which different classifiers predict differently or even completely at random. The rationale for this desire, *described here, for simplicity, in binary classification terms*, is that we are less concerned by a big subgroup, for which at any given row, one classifier gets it wrong (or only one gets it right), while the other get it right (wrong), than by a smaller subgroup for which always half of the classifiers get those cases wrong. The subgroup on which half of the classifiers get the cases wrong should be ranked higher. In Table I, toy dataset A and its relevant classifiers, we would like to discover first the subgroup $\{2, 5\}$.

B. Consistent Classification

In the next scenario, we consider the following objective. We are interested in controversy but of less random nature: a scenario in which few classifiers consistently differ from the other classifiers. *We assume here that the classifiers are consistent in the regions (low entropy per classifier)*. In the example from Table II, toy dataset B, we would like to discover first subgroup $\{1, 3, 4\}$ or subgroup $\{1, 2, 3, 4\}$. This is because all four classifiers, $\{C_1, C_2, C_3, C_4\}$ are each internally consistent in those regions, while there is a disagreement across the four. Notice that using that intuition, *we direct the search to a region in which the per-classifier entropy is low*, but mean per-row entropy is high. To this end, we define the following quality measure:

$$\varphi_{ccl}(SG) \equiv \frac{1}{|SG|} \sum_{i \in SG} \mathcal{H}(M_{i1}, \ldots, M_{in}) - \frac{1}{n} \sum_{j=1}^{n} \mathcal{H}(M_{ij} | i \in SG)$$

For example, $\varphi_{ccl}(\{1, 2, 3, 4\}) = 0.858 - 0.203 = 0.656$ (rounded), $\varphi_{ccl}(\{1, 3, 4\}) = 0.811 - 0 = 0.811$, $\varphi_{ccl}(\{5, 6, 7, 8\}) = 0.453 - 0.608 = -0.156$, $\varphi_{ccl}(\{5, 6, 7\}) = 0.333 - 0.459 = -0.126$, and $\varphi_{ccl}(\{5, 6\}) = 0.5 - 0.5 = 0$. The use case for this scenario is to identify regions in which few classifiers behave different, yet limiting the search for regions in which each classifier is consistent.

C. Consistent Accordance

We next identify controversy of consistent nature, while overcoming the rigidity of φ_{ccl}, where different predictions over different cases result in high classifier-wise entropy, thus lower rank for the relevant subgroup. Achieving this goal allows us to identify regions where a few classifiers are the negation of the majority. We cannot normally achieve this with φ_{row}, unless the same regions indeed contain the greatest per-row entropy on average. To allow for different predictions per-classifier we move from the prediction space to the accordance space. Thus we first identify the top predicted class per row (most frequently predicted), and then compare it to the prediction. In case of a tie, we choose one of the classes. Thus for every row, i, $Top_i \leftarrow most_frequent_in_row(M_i)$, and then for every classifier j, $M'_{ij} \leftarrow \mathbb{1}(M_{ij} = Top_i)$. We next search for interesting regions based on M', using the following quality measure:

$$\varphi_{cac}(SG) \equiv \frac{1}{|SG|} \sum_{i \in SG} \mathcal{H}(M'_{i1}, \ldots, M'_{in}) - \frac{1}{n} \sum_{j=1}^{n} \mathcal{H}(M'_{ij} | i \in SG)$$

D. Consistent Correctness

In this scenario and all subsequent ones, we assume the availability of the ground truth, GT. The availability of the ground truth enables us to attempt to identify hard-to-classify regions, on which few models actually succeed, or the other way around: easy regions, on which a few models consistently fail. We start by collecting the correctness of the predictions, hence for every case i and for every classifier j,

$M''_{ij} \leftarrow \mathbb{1}(M_{ij} = \ell_i)$. We then evaluate using the mean of row-wise entropies minus the mean of classifier-wise entropies. Note that also here, once we switch from the output space to the correctness space, the per classifier consistency is of a different nature. Hence, classifiers that are the negation of other classifiers may result in higher ranking for the relevant regions. We use the following quality measure:

$$\varphi_{cco}(SG) \equiv \frac{1}{|SG|} \sum_{i \in SG} \mathcal{H}(M''_{i1}, \ldots, M''_{in}) - \frac{1}{n} \sum_{j=1}^{n} \mathcal{H}(M''_{ij} | i \in SG)$$

Differences between φ_{cco} and φ_{cac} are possible, where there are cases for which the majority of classification is different from the true label.

E. Ground Truth as Yet Another Classifier

If we treat the ground truth as yet another classifier, we can evaluate the mean per-row entropy as is done for φ_{row}. What is the effect of adding GT as an additional classifier? Rows for which most of the classifiers predict correctly, now have a lower entropy. Rows for which only a minority of the classifiers predict rightly, have a higher entropy. The search for regions for which the mean row-wise entropy is the highest, results in finding regions that are hard to predict correctly. We add GT as an additional classifier, as described above, and also in another experiment, add GT as additional n classifiers.

$$\varphi_{GT_as_yac}(SG) \equiv \frac{1}{|SG|} \sum_{i \in SG} \mathcal{H}(M_{i1}, \ldots, M_{in}, \ell^i)$$

$$\varphi_{GT_as_yac'}(SG) \equiv \frac{1}{|SG|} \sum_{i \in SG} \mathcal{H}(M_{i1}, \ldots, M_{in}, (\ell^i)_{\times n})$$

Note that $\varphi_{GT_as_yac}$ is expected to be similar to φ_{row} yet puts some additional emphasis on regions with errors. $\varphi_{GT_as_yac'}$ should take this aspect even further: by matching each classifier's prediction with a copy of the ground truth, the weight of mistakes, as is reflected in the ranking of the subgroups, should be even higher.

F. Relative Average Subranking Loss

The last scenario in this paper is applicable to binary classification only. We adapt the existing SCaPE model class for EMM [8], to identify regions that are exceptionally hard or easy to predict. By examining M, we calculate the empirical probability of predicting the positive class per row. Thus for every row i,

$$Prob^i \leftarrow count_of_positives_in_row(M_i)/n \ .$$

We obtain therefore a soft classifier, $Prob$, to be contrasted with the ground truth GT, gauged with the quality measure φ_{rasl} used in SCaPE.

Table III: Datasets used for experiments.

	Dataset	#cases (m)	#attributes discrete	#attributes numeric	$\lvert\mathcal{C}\rvert$
1.	Mushroom	8,124	22	0	2
2.	Titanic	891	3	4	2
3.	Adult	48,842	8	6	2
4.	Balance-scale	625	0	4	3
5.	Car	1,728	6	0	4
6.	Pima-indians	768	0	8	2
7.	Covertype	581,012	44	10	7
8.	YearPredictionMSD	515,345	0	90	10

V. Experiments

We illustrate the workings of the Controversy Rules model class for EMM, by experimenting on the following classifiers: Decision Tree [19], Naïve Bayes [24], 3-Nearest Neighbors [2], Random Forest [13], and Support Vector Machine with linear kernel [5]. The choice of classifiers is purely for illustrative purposes and should not be confused with the core contribution of this paper: we provide a method to find regions of controversy between classifiers, which we illustrate with this selection of well-known classifiers (which should not be taken as endorsement of the classifiers themselves). We obtain the predictions by running 10-fold cross validation for each of the model classes. Hence, technically, each prediction column is created by 10 different classifiers; the perceived classifiers are virtual, and have never existed. We mention this for the benefit or reproducibility; how the predictions were obtained is not fundamental to the core contribution of this paper.

We run the experiments on the eight datasets listed in Table III. Most are taken from the UCI ML repository [6]. The *Titanic* dataset is taken from Kaggle (https://www.kaggle.com/c/titanic/data), and *Pima-indians* (which is no longer available in the UCI ML repository) can also be accessed there (https://www.kaggle.com/uciml/pima-indians-diabetes-database/data). The *YearPredictionMSD* dataset comes with a naturally in-built regression task (predicting the year in which a song was released). We define our own classification task on this dataset, converting the year into decades (the floor of the year divided by 10 is taken as the true label). Some of the datasets are suitable for binary classification tasks (*Mushroom*, *Titanic*, *Adult*, *Pima-indians*), while other contain more than 2 classes, although sometimes ordinal in nature (*Balance-scale*, *Car*, *YearPredictionMSD*).

To discover subgroups, we employ the Beam Search algorithm for Exceptional Model Mining, as described in [7, Algorithm 1]. The parameters are set as follows: beam width $w = 25$, search depth $d = 3$. To avoid tiny subgroups, we require a minimum support of 4% of the cases.

A. Mushroom

On the *Mushroom* dataset, four out of five classifiers predicted almost all test cases correct (Naïve Bayes has 216 false positives, and 3 false negatives, k-Nearest Neighbors has 2 false negatives, and the other three classifiers do not make errors). The prediction matrix is displayed in Figure 1. The

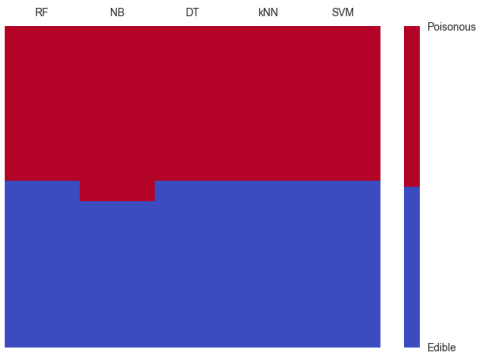


Figure 1: Predictions matrix, ordered from left to right based on the classifiers' predictions. The *Mushroom* dataset.

Table IV: Subgroups found with the φ_{row} quality measure, for the *Mushroom* dataset ($\varphi_{row}(DS) = 0.046$).

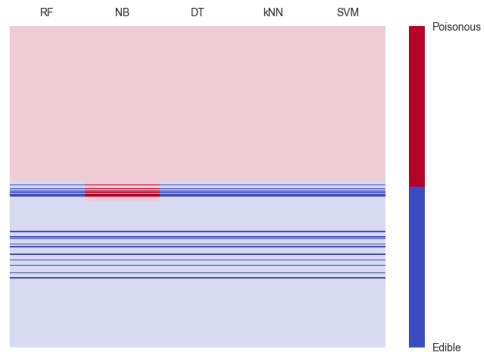
description	#cases	φ_{row}
$odor = n \wedge stalk_color_above_ring \neq g \wedge stalk_color_below_ring = p$	384	0.385
$ring_type = p \wedge stalk_color_above_ring \neq g \wedge stalk_color_below_ring = p$	384	0.385
$bruises = t \wedge stalk_color_above_ring \neq g \wedge stalk_color_below_ring = p$	384	0.385
$odor = n \wedge stalk_color_above_ring = p \wedge stalk_color_below_ring \neq g$	384	0.376
$ring_type = p \wedge stalk_color_above_ring = p \wedge stalk_color_below_ring \neq g$	384	0.376
$bruises = t \wedge stalk_color_above_ring = p \wedge stalk_color_below_ring \neq g$	384	0.376

Table V: Subgroups found with the φ_{ccl} quality measure, for the *Mushroom* dataset ($\varphi_{ccl}(DS) = -0.952$).

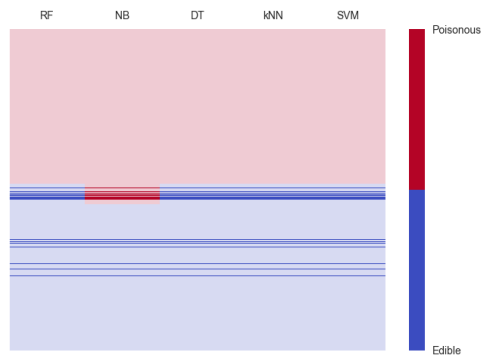
description	#cases	φ_{ccl}
$odor = n \wedge stalk_color_above_ring \neq g \wedge stalk_color_below_ring = p$	384	0.186
$odor = n \wedge stalk_color_above_ring = p \wedge stalk_color_below_ring \neq g$	384	0.176
$gill_color \neq u \wedge odor = n \wedge stalk_color_below_ring = p$	432	0.143
$gill_color \neq u \wedge odor = n \wedge stalk_color_above_ring = p$	432	0.135
$gill_color \neq n \wedge odor = n \wedge stalk_color_below_ring = p$	432	0.114
$gill_color = p \wedge gill_spacing = c \wedge odor = n$	468	0.110

order of the classifiers, from left to right, is as those are listed for the experiment. The cases, or the rows, are ordered by the predictions of the classifiers, lexicographically from left to right. As can be seen, one classifier (Naïve Bayes in this case) is predicting differently from the rest for numerous cases, while the other agree almost always. The exact descriptions ordered the same, are reported also by φ_{GT_yac} and by $\varphi_{GT_yac'}$. This is expected as errors and disagreements here are in the same cases.

φ_{rasl} did not find anything interesting as the errors made (by the Naïve Bayes classifier) are to mistake consistently a negative to be a positive for a few of the cases, and hence the probability for those cases is indeed between the 0 for most



(a) Top description both for φ_{row} and for φ_{ccl}. $odor = n \wedge stalk_color_above_ring \neq g \wedge stalk_color_below_ring = p$



(b) A new description reported by φ_{ccl}. $gill_color \neq u \wedge odor = n \wedge stalk_color_below_ring = p$

Figure 2: Prediction matrices on the *Mushroom* dataset, with highlighted subgroups.

Table VI: Subgroups found with the φ_{cac} quality measure, for the *Mushroom* dataset ($\varphi_{cac}(DS) = -0.023$).

description	#cases	φ_{cac}
$gill_color = w \wedge habitat = d \wedge stalk_color_above_ring \neq g$	390	0.109
$gill_color = w \wedge habitat = d \wedge stalk_color_below_ring \neq g$	390	0.109
$gill_color = w \wedge odor = n \wedge population = v$	342	0.096
$gill_color = w \wedge stalk_color_below_ring \neq w \wedge stalk_root = b$	352	0.092
$gill_color = w \wedge stalk_color_above_ring \neq g \wedge stalk_root = b$	544	0.090
$gill_color = w \wedge stalk_color_below_ring \neq g \wedge stalk_root = b$	544	0.090

of the negative cases, and the 1 for most of the positive cases.

Table V lists the descriptions reported by the φ_{ccl} quality measure. The top subgroup is the same as the one found with φ_{row}, but subsequent subgroups differ. To illustrate the difference, Figure 2 displays two prediction matrices: one (Figure 2a) for the top subgroup for both measures, and one (Figure 2b) for the new description $gill_color \neq u \wedge odor = n \wedge stalk_color_below_ring = p$. Comparing those two

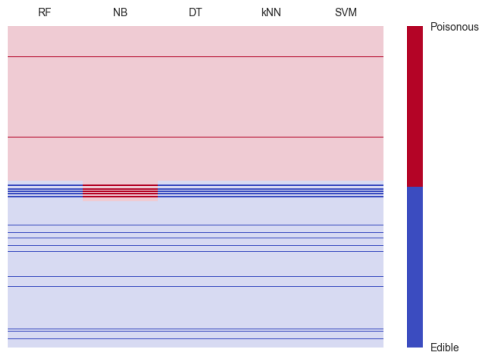


Figure 3: Top subgroup found with φ_{cac} on the *Mushroom* dataset: $gill_color = w \wedge habitat = d \wedge stalk_color_above_ring \neq g$.

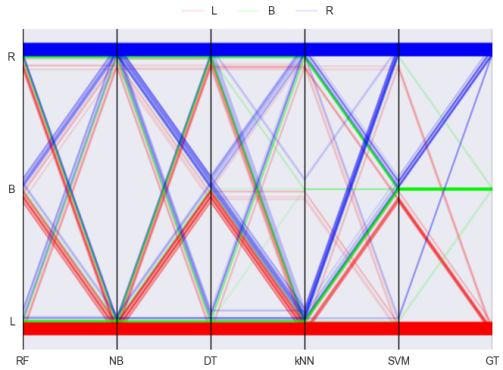


Figure 4: Parallel coordinates for the predictions and the true labels, *Balance-scale* dataset

subgroups, we note that as one classifier, the Naïve Bayes, is more consistent when there are fewer negative cases, relevant descriptions are being ranked higher with φ_{ccl}.

Table VI lists the subgroups found with φ_{cac}. The top description $gill_color = w \wedge habitat = d \wedge stalk_color_above_ring \neq g$ is illustrated in Figure 3. We see that some more positive cases are included. The internal accordance of the classifiers is intact by adding those cases, and this subgroup is more interesting than the top one reported by φ_{row}, if taking into account also the consistent accordance. The descriptions reported by φ_{cco} are the same as those reported by φ_{cac}. This is not surprising, since the majority of the classifiers get all the cases correct.

B. Balance-scale

The task for the Balance-scale dataset is classification, where 3 possible classes exist *L* for left, *B* for balanced, and *R* for right. The datasets represents a scale, where both on the left and on the right side a single weight is placed at a single spot. For both the weight and the distance from the spot to the center of the balance, integer unit values between one and five can be chosen. Hence, there are $5^4 = 625$ configurations. The underlying physical law states that the scale is in balance, if and only if $Left_Distance \times Left_Weight$ equals $Right_Distance \times Right_Weight$.

One can intuitively understand that, assuming the classifiers do not have access to the exact mechanism, higher confusion can be found near the decision boundaries and around the balanced state. Naïve Bayes is expected to have some difficulties, as the assumption of independence among the conditional probabilities conflicts with the underlying multiplicative physical law (as just outlined). Naive Bayes resolves this problem by simply ignoring all the *B* cases. This is surprisingly effective, compared with the other classifiers. A view of the predictions is displayed as a parallel coordinates plot in Figure 4, where the true labels are also included (far right).

Table VII: Subgroups found with the φ_{row} quality measure, for the *Balance-scale* dataset ($\varphi_{row}(DS) = 0.304$).

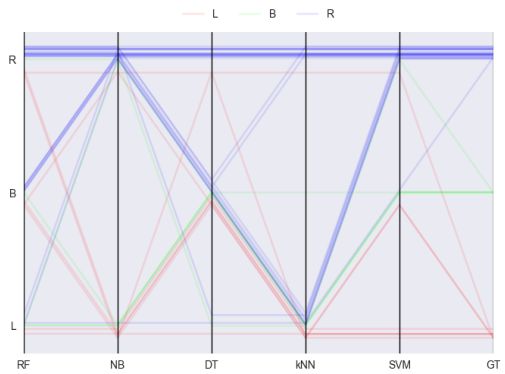
description	#cases	φ_{row}
$Left_Weight \leq 2 \wedge Right_Distance \leq 2 \wedge Right_Weight > 3$	40	0.650
$Left_Weight \leq 2 \wedge Right_Distance \leq 1 \wedge Right_Weight > 2$	30	0.634
$Left_Weight \leq 2 \wedge Right_Distance \leq 2 \wedge Right_Weight > 2$	60	0.617
$Left_Weight \leq 3 \wedge Right_Distance \leq 1 \wedge Right_Weight > 3$	30	0.607
$Left_Weight \leq 3 \wedge Right_Distance \leq 2 \wedge Right_Weight > 3$	60	0.604
$Left_Distance > 3 \wedge Left_Weight \leq 2 \wedge Right_Distance \leq 2$	40	0.589

Table VIII: Subgroups found with the φ_{GT_yac} quality measure, for the *Balance-scale* dataset ($\varphi_{GT_yac}(DS) = 0.310$).

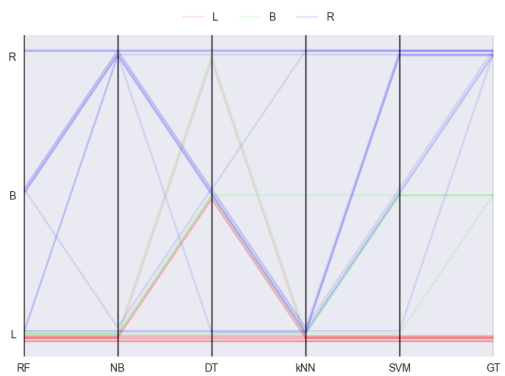
description	#cases	φ_{GT_yac}
$Left_Weight \leq 2 \wedge Right_Distance \leq 2 \wedge Right_Weight > 3$	40	0.663
$Left_Weight \leq 2 \wedge Right_Distance \leq 1 \wedge Right_Weight > 2$	30	0.654
$Left_Weight \leq 2 \wedge Right_Distance \leq 2 \wedge Right_Weight > 2$	60	0.633
$Left_Distance > 2 \wedge Left_Weight \leq 1 \wedge Right_Distance \leq 2$	30	0.630
$Left_Distance > 3 \wedge Left_Weight \leq 1 \wedge Right_Distance \leq 3$	30	0.627
$Left_Weight \leq 3 \wedge Right_Distance \leq 1 \wedge Right_Weight > 3$	30	0.622

Table IX: Subgroups found with the $\varphi_{GT_yac'}$ quality measure, for the *Balance-scale* dataset ($\varphi_{GT_yac'} = 0.262$).

description	#cases	$\varphi_{GT_yac'}$
$Left_Distance > 2 \wedge Left_Weight \leq 1 \wedge Right_Distance \leq 2$	30	0.559
$Left_Distance > 3 \wedge Left_Weight \leq 1 \wedge Right_Distance \leq 3$	30	0.556
$Left_Weight \leq 2 \wedge Right_Distance \leq 1 \wedge Right_Weight > 2$	30	0.555
$Left_Weight \leq 2 \wedge Right_Distance \leq 2 \wedge Right_Weight > 3$	40	0.552
$Left_Weight \leq 1 \wedge Right_Distance \leq 2 \wedge Right_Weight > 1$	40	0.535
$Left_Weight \leq 2 \wedge Right_Distance \leq 2 \wedge Right_Weight > 2$	60	0.533



(a) $Left_Weight \leq 2 \wedge Right_Distance \leq 2 \wedge Right_Weight > 3$, top ranked by φ_{row}



(b) $Left_Distance > 2 \wedge Left_Weight \leq 1 \wedge Right_Distance \leq 2$, top ranked by $\varphi_{GT_yac'}$

Figure 5: Parallel coordinates. The *Balance-scale* dataset.

Table VII lists the subgroups found with φ_{row}. The top description restricts three of the four variables: a small weight on the left side, and a large weight at a small distance on the right side. This is indeed a volatile situation, where a small change in any of the remaining choices will cause the scales to tip over. Hence, it makes sense that classifiers disagree.

Top subgroups for φ_{GT_yac} and $\varphi_{GT_yac'}$ are listed in Tables VIII and IX, respectively. This stepwise increase in the importance of the true label can be expected to affect the ranking of the top subgroups. Indeed we see the description $Left_Distance > 2 \wedge Left_Weight \leq 1 \wedge Right_Distance \leq 2$ appearing in the fourth place in φ_{GT_yac}, where it is not reported by φ_{row}, and then it climbs to the top in $\varphi_{GT_yac'}$. In Figure 5 we contrast the top subgroup for φ_{row} (a) and the top subgroup for $\varphi_{GT_yac'}$ (b).

φ_{ccl} reports descriptions for which the quality measure is 0. This value calculated from $0 - 0$, that is 0 for the mean per row entropies, and 0 for the per classifier entropy, for example $Left_Weight \leq 3 \wedge Right_Distance > 3 \wedge Right_Weight > 4$, which always results in *R* true label, as can be seen in Figure 6. The φ_{ccl} quality measure score for the whole dataset is -0.942. These descriptions correspond to a total agreement among the classifiers, which is not exactly what we are looking for yet is also interesting. To have descriptions for which the measure is bigger than 0, the mean per-row entropies should be higher than the mean per-classifier entropies. Similar reports are given by φ_{cac} and by φ_{cco}.

φ_{rasl} is not applicable here as we have more than two labels.

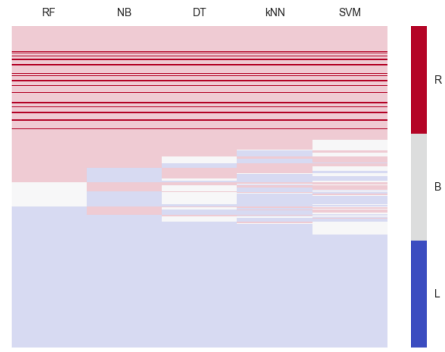


Figure 6: Prediction matrix, *Balance-scale* dataset, highlighted the subgroup for the description $Left_Weight \leq 3 \wedge Right_Distance > 3 \wedge Right_Weight > 4$, which is ranked highest by the φ_{ccl} quality measure.

Table X: Subgroups found with the φ_{row} quality measure, for the *Titanic* dataset ($\varphi_{row}(DS) = 0.294$).

description	#cases	φ_{row}
$Embarked = C \wedge Pclass \leq 2 \wedge Sex = male$	52	0.842
$Embarked \neq S \wedge Pclass \leq 1 \wedge Sex = male$	43	0.838
$Embarked = C \wedge Pclass \leq 1 \wedge Sex = male$	42	0.835
$Embarked = S \wedge Pclass > 2 \wedge Sex = female$	88	0.830
$Age > 23 \wedge Pclass > 2 \wedge Sex = female$	43	0.804
$Fare > 73.18988571428572 \wedge Sex = male$	40	0.780

Table XI: Subgroups found with the φ_{GT_yac} quality measure, for the *Titanic* dataset ($\varphi_{GT_yac}(DS) = 0.337$).

description	#cases	φ_{GT_yac}
$Embarked \neq S \wedge Pclass \leq 1 \wedge Sex = male$	43	0.849
$Embarked = C \wedge Pclass \leq 1 \wedge Sex = male$	42	0.846
$Age > 23 \wedge Pclass > 2 \wedge Sex = female$	43	0.845
$Embarked = C \wedge Pclass \leq 2 \wedge Sex = male$	52	0.844
$Embarked = S \wedge Pclass > 2 \wedge Sex = female$	88	0.837
$Age > 11 \wedge Pclass > 2 \wedge Sex = female$	79	0.818

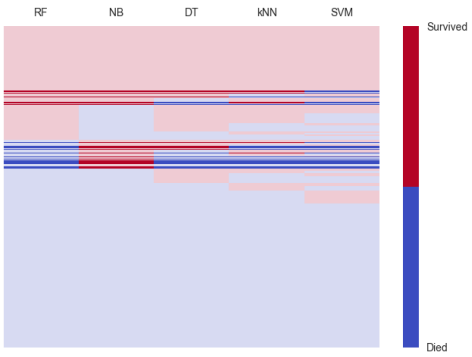
C. Titanic

The task for the *Titanic* dataset is binary classification (life or death), based on attributes known about the passengers of the Titanic. The famous ship collected passengers from three ports. Some of the passengers traveled alone, while others traveled with family members. There were three classes of cabinets with different price levels. In general, once the ship hit the iceberg, children and women were offered a place in a lifeboat before the other passengers. Unfortunately there were not enough boats for everyone.

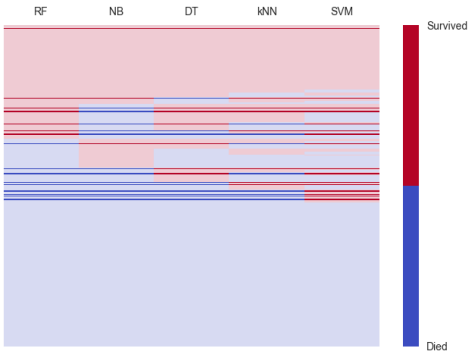
Table X, XI, and XII list the top subgroups found with φ_{row}, φ_{GT_yac}, and $\varphi_{GT_yac'}$, respectively. As can be seen in

Table XII: Subgroups found with the $\varphi_{GT_yac'}$ quality measure, for the *Titanic* dataset ($\varphi_{GT_yac'}(DS) = 0.311$).

description	#cases	$\varphi_{GT_yac'}$
$Age > 23 \wedge Pclass > 2 \wedge Sex = female$	43	0.750
$Age > 11 \wedge Pclass > 2 \wedge Sex = female$	79	0.736
$Embarked \neq S \wedge Pclass \leq 1 \wedge Sex = male$	43	0.728
$Embarked = C \wedge Pclass \leq 1 \wedge Sex = male$	42	0.725
$Embarked = C \wedge Pclass \leq 2 \wedge Sex = male$	52	0.714
$Embarked = S \wedge Pclass > 2 \wedge Sex = female$	88	0.711



(a) $Embarked = C \wedge Pclass \leq 2 \wedge Sex = male$, top ranked by φ_{row}



(b) $Age > 23 \wedge Pclass > 2 \wedge Sex = female$, top ranked by $\varphi_{GT_yac'}$

Figure 7: Prediction matrix ordered from left to right. The *Titanic* dataset. For many cases highlighted for φ_{row}, Naïve Bayes is in a minority, predicting *Survived*.

Figure 7, the top description for φ_{row} picks on a region where the Naïve Bayes classifier predicts mostly *Survived*. This is also the case for the top description for φ_{GT_yac}, which is ranked in the second place for φ_{row}. By contrast, $\varphi_{GT_yac'}$ ranks other descriptions first.

Tables XIII and XIV list subgroups found when maximizing and minimizing, respectively, φ_{rasl}. The top description from maximizing φ_{rasl} is $Parch \leq 0 \wedge Pclass > 1 \wedge Sex = male$. SVM has the highest accuracy 89% in this region by predicting all those 385 cases as Died, while 44 passengers did survive (cf. Figure 8). All attempts by the other classifiers

Table XIII: Subgroups found with the φ_{GT_rasl} quality measure, for the *Titanic* dataset ($\varphi_{rasl}(DS) = 274.0$).

description	#cases	φ_{rasl}
$Parch \leq 0 \wedge Pclass > 1 \wedge Sex = male$	385	127.32
$Parch \leq 1 \wedge Pclass > 1 \wedge Sex = male$	429	105.93
$Pclass > 1 \wedge Sex = male$	455	105.74
$Fare \leq 146.38 \wedge Pclass > 1 \wedge Sex = male$	455	105.74
$Fare \leq 292.76 \wedge Pclass > 1 \wedge Sex = male$	455	105.74
$Fare \leq 365.95 \wedge Pclass > 1 \wedge Sex = male$	455	105.74

Table XIV: Subgroups found when minimizing the φ_{rasl} quality measure, for the *Titanic* dataset. $Age \leq -1$ is referring to cases for which the age is not recorded. The classifiers used the median age for those cases.

description	#cases	φ_{rasl}
$Age \leq -1$	177	0.0
$Age \leq -1 \wedge Pclass > 1$	147	0.0
$Age \leq -1 \wedge Pclass \leq 2$	41	0.0
$Age \leq -1 \wedge Pclass > 2$	136	0.0
$Age \leq -1 \wedge Sex = male$	124	0.0
$Age \leq -1 \wedge Sex = female$	53	0.0

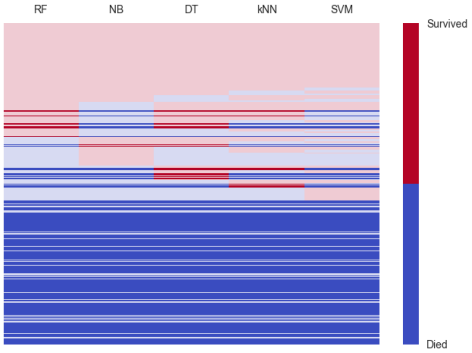


Figure 8: $Parch \leq 0 \wedge Pclass > 1 \wedge Sex = male$, *Titanic* dataset. This description is ranked at the top when maximizing φ_{rasl}. SVM does the best by predicting always *Died*.

to identify the survivors result in many false positives. The top description when minimizing φ_{rasl}, $Age \leq -1$, refer to cases for which the age is unknown. The Decision Tree and the Random Forest classifiers, achieve in this region the top accuracy of 85%, by predicting correctly 39 *Survived* cases out of 52. Other models also predict correctly most of the positive cases. We have maximized φ_{rasl} and found a region for which the SVM classifier achieves the highest accuracy of 89%, and we have minimized φ_{rasl} and found a region for which those are the DT and RF classifiers that achieve the highest accuracy, this time only 85%, which is lower than 89%. Therefore we note that φ_{rasl} with this setting of classifiers' predictions, is not discriminating subgroups based on individual classifiers' accuracy but rather based on whether the collection of classifiers, if used as a voting ensemble, correctly ranks the cases. Therefore, as a voting ensemble the classifiers do a better job for $Age \leq -1$ than for $Parch \leq 0 \wedge Pclass > 1 \wedge Sex = male$.

Table XV: Subgroups found with the φ_{ccl} quality measure, for the *Titanic* dataset ($\varphi_{ccl}(DS) = -0.645$).

description	#cases	φ_{ccl}
$Embarked = C \wedge Pclass \leq 2 \wedge Sex = male$	52	0.313
$Embarked \neq S \wedge Pclass \leq 1 \wedge Sex = male$	43	0.291
$Embarked = C \wedge Pclass \leq 1 \wedge Sex = male$	42	0.290
$Embarked \neq S \wedge Pclass \leq 2 \wedge Sex = male$	54	0.284
$Embarked = S \wedge Pclass > 2 \wedge Sex = female$	88	0.205
$Age > 23 \wedge Pclass > 2 \wedge Sex = female$	43	0.097

Table XVI: Subgroups found with the φ_{cac} quality measure, for the *Titanic* dataset ($\varphi_{cac}(DS) = -0.178$).

description	#cases	φ_{cac}
$Embarked = C \wedge Pclass \leq 1 \wedge Sex = male$	42	0.116
$Embarked \neq S \wedge Pclass \leq 1 \wedge Sex = male$	43	0.103
$Age \leq 22 \wedge Embarked = S \wedge SibSp > 2$	38	0.031
$Age \leq 22 \wedge Embarked \neq Q \wedge SibSp > 2$	38	0.031
$Embarked = S \wedge Pclass > 2 \wedge SibSp > 2$	38	0.021
$Embarked \neq Q \wedge Pclass > 2 \wedge SibSp > 2$	38	0.021

Table XVII: Subgroups found with the φ_{cco} quality measure, for the *Titanic* dataset ($\varphi_{cco}(DS) = -0.457$).

description	#cases	φ_{cco}
$Fare > 73.19 \wedge Parch \leq 1 \wedge Sex = female$	52	-0.014
$Fare > 73.19 \wedge Parch \leq 0 \wedge Sex = female$	41	-0.015
$Fare \leq 146.38 \wedge Fare > 73.19 \wedge Sex = female$	41	-0.031
$Parch \leq 1 \wedge Pclass \leq 1 \wedge Sex = female$	81	-0.110
$Fare \leq 146.38 \wedge Pclass \leq 1 \wedge Sex = female$	73	-0.124
$Parch \leq 0 \wedge Pclass \leq 1 \wedge Sex = female$	64	-0.131

Tables XV, XVI, and XVII list the top subgroups for φ_{ccl}, φ_{cac}, and φ_{cco}, respectively. Subgroups for φ_{ccl} are very similar to those for φ_{row}. The top two subgroups from φ_{cac} are also found with φ_{row} and with φ_{cco}. Those subgroups correspond to regions in which there is internal agreement for NB and for SVM, as in Figure 7 (a). The third ranked description in φ_{cac} is given in Figure 9. The fact that there are differences between φ_{cac} and φ_{cco} tells us that sometimes the classifiers agree, yet they agree on the wrong prediction. The top ranked subgroups for φ_{cco} are of regions with a lot of agreement for *Survived*.

D. YearPredictionMSD

On the YearPredictionMSD dataset we define a classification task with 10 classes. The features are real numbers, taken from the Echo Nest API, for analyzing sound tracks. The first 12 features are averages of timbre values (fea1-fea12), and the next 78 features are covariances of timbre values (fea13-fea90). It is assumed that from the sound analysis of the songs it is possible in theory, to predict the year the song was released (or in our case the decade). The accuracies that our classifiers reported are as follows: 61% for SVM, 58% for RF, 51% for kNN, 48% for DT, and 23% for NB.

Tables XVIII and XIX list subgroups found with φ_{row} and φ_{ccl}, respectively. The latter delivers larger subgroups (cf. Table XIX). For both φ_{cac} and φ_{cco}, the description $fea1 > 44.76 \wedge fea11 \leq -2.09 \wedge fea6 \leq -10.93$, appears at the top, while for φ_{ccl}, as can be seen in Table XIX, that description appears last.

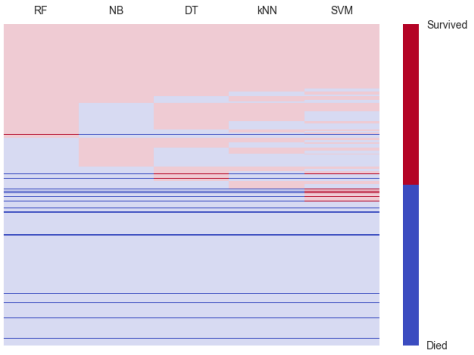


Figure 9: $Age \leq 22 \wedge Embarked = S \wedge SibSp > 2$, *Titanic* dataset. This is the third description reported by φ_{cac}. The accordance with the majority, leads to apparent internal consistency. Yet the majority of classifiers may be wrong for some of the cases in this region, and therefore this description is not be ranked high for φ_{cco}.

Table XVIII: Subgroups found with the φ_{row} quality measure, for the *YearPredictionMSD* dataset ($\varphi_{row}(DS) = 1.07597$).

description	#cases	φ_{row}
$fea13 \leq 78.65 \wedge fea3 > 55.48 \wedge fea6 > -10.93$	21375	1.42877
$fea3 > 55.48 \wedge fea6 > -10.93 \wedge fea60 > -159.88$	21414	1.42746
$fea3 > 55.48 \wedge fea6 > -10.93 \wedge fea72 > -270.76$	21090	1.42742
$fea1 \leq 44.7641 \wedge fea13 \leq 78.65 \wedge fea3 > 55.48$	22172	1.42707
$fea10 > -12.50 \wedge fea3 > 55.48 \wedge fea6 > -10.93$	21773	1.42465
$fea3 > 55.48 \wedge fea6 > -10.93 \wedge fea7 \leq 17.85$	21158	1.42462

Table XIX: Subgroups found with the φ_{ccl} quality measure, for the *YearPredictionMSD* dataset ($\varphi_{ccl}(DS) = -0.48005$).

description	#cases	φ_{ccl}
$fea1 > 44.76 \wedge fea38 \leq 278.49 \wedge fea6 \leq -10.93$	129,197	0.03053
$fea1 > 44.76 \wedge fea56 \leq 35.74 \wedge fea6 \leq -10.93$	112,228	0.02556
$fea1 > 44.76 \wedge fea6 \leq -10.93 \wedge fea70 \leq 8.27$	105,874	0.02076
$fea1 > 44.76 \wedge fea6 \leq -10.93 \wedge fea65 > -51.89$	110,647	0.01772
$fea1 > 44.76 \wedge fea10 > 2.07 \wedge fea6 \leq -10.93$	93,033	0.01620
$fea1 > 44.76 \wedge fea11 \leq -2.09 \wedge fea6 \leq -10.93$	30,640	0.01467

E. Covertype, Car, Pima-indians, Adult

We find similar observations with *Covertype*, *Car*, *Pima-indians*, and with *Adult*. Therefore, to save space, we will not list the subgroups found on these datasets here. φ_{row} reports regions with high controversy among classifiers. Moving to φ_{GT_yac} and then to $\varphi_{GT_yac'}$ there is a trend of new descriptions climbing in the ranking and pushing other descriptions down, as more emphasis is given to cases predicted wrongly.

For φ_{ccl}, φ_{cac}, and φ_{cco} the top descriptions are sometimes regions with a lot of agreement, or in more interesting situations, regions with high controversy, similar to those reported by φ_{row}, yet of more structured nature, where a few classifiers differ consistently from the rest. In an informal manner, we

can say, that for *Adult*, $\varphi_{ccl} \neq \varphi_{cac} = \varphi_{cco}$, and for *Car*, $\varphi_{ccl} = \varphi_{cco} \neq \varphi_{cac}$. This difference has to do with whether the majority of classifiers predict correctly or wrongly, in a significant trend, in a region.

φ_{rasl} when maximized for *Adult* gives different and bigger subgroups than those reported by φ_{row}, and this is the case also for *Pima-indians*. Minimizing φ_{rasl} gives well defined regions for which most classifiers predict correctly.

VI. Conclusions

We introduce the Controversy Rules model class for Exceptional Model Mining, to find regions of the input space where a set of classifiers is in unusual (dis-)agreement. This level of (dis-)agreement can be gauged in many different ways; we introduce several quality measures to explore various options. φ_{row} reports regions with high controversy among the classifiers. φ_{GT_yac} is similar to φ_{row} yet puts some emphasis also on errors. $\varphi_{GT_yac'}$ puts even more emphasis on errors. We demonstrate, in Figure 7 this trend. Comparing descriptions reported by φ_{ccl} to those reported by φ_{row} tells us how consistent the controversy is, and whether internal consistency consideration yields different descriptions, as shown in Figure 2. Further comparing to φ_{cac} and to φ_{cco}, shows us also regions where classifiers have different predictions, and whether the majority of classifiers predict there correctly, as we show in Figure 3 and in Figure 9. Occasionally the reported subgroups with φ_{ccl}, φ_{cac}, and/or φ_{cco}, correspond to regions in which there is a lot of agreement among the classifiers, as for other regions, on average, the per classifier internal controversy is higher than the averaged per row controversy. φ_{rasl} is treating the collection of classifiers as a single voting-based ensemble, and shows us when this ensemble predicts (in-)correctly, as is discussed over the *Titanic* dataset in Subsection V-C. For the *Mushroom* dataset, in Subsection V-A we see a setting where φ_{row} reports interesting regions, while φ_{rasl} does not identify anything special. It is not surprising that there are similarities between the reports of the various quality measures, yet all quality measures evaluated in this paper are useful and shed unique light. The differences among the reports, give us additional clues to better understand the classifiers and their interaction with the modalities in the data.

We note the following challenges. All above quality measures are designed under the assumption, that all predictions are available (matrix M is full); missing values cannot be accommodated. While our experiments are with some unbalanced datasets, we do not study in detail the effects of this. We account for all discrepancies the same, yet with ordinal classes, some differential weight is more appropriate. Finally, an appropriate adjustment for regression tasks is nontrivial.

For future work we see two promising directions. The one is exploring ensembles. The differences between what we are doing in this paper and studying ensembles, is that we evaluate few classifiers of different type, while for ensembles there usually is a bigger number of base estimators, yet of more similar nature. The other promising direction we identify is focusing the discussion on a specific classifier's interpretability, contrasting the model with possible explanations.

References

[1] Philip Adler, Casey Falk, Sorelle A. Friedler, Tionney Nix, Gabriel Rybeck, Carlos Scheidegger, Brandon Smith, and Suresh Venkatasubramanian. Auditing black-box models for indirect influence. *Knowledge and Information Systems* 54(1):95–122, 2018.

[2] Naomi S. Altman. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician* 46(3):175–185, 1992.

[3] Roberto Bisiani. Beam Search. *Encyclopedia of Artificial Intelligence. In Shapiro, S.C., John Wiley & Sons.*, 1987.

[4] Diane J. Cook and Lawrence B. Holder. Substructure Discovery Using Minimum Description Length and Background Knowledge. *Journal of Artificial Intelligence Research* 1(1):231–255, 1994.

[5] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning* 20(3):273–297, Sep 1995.

[6] Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017.

[7] Wouter Duivesteijn, Ad J. Feelders, and Arno Knobbe. Exceptional Model Mining. *Data Mining and Knowledge Discovery* 30(1):47–98, 2016.

[8] Wouter Duivesteijn and Julia Thaele. Understanding Where Your Classifier Does (Not) Work – The SCaPE Model Class for EMM. In *Proc. ICDM*, pp. 809–814, 2014.

[9] Yarin Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.

[10] Satoshi Hara and Kohei Hayashi. Making Tree Ensembles Interpretable. In *Proc. ICML Workshop on Human Interpretability in Machine Learning*, pp. 81–85, 2016.

[11] Andreas Henelius, Kai Puolamäki, Henrik Boström, Lars Asker, and Panagiotis Papapetrou. A peek into the black box: Exploring classifiers by randomization. *Data Mining and Knowledge Discovery* 28(5-6):1503–1529, 2014.

[12] Andreas Henelius, Kai Puolamäki, Isak Karlsson, Jing Zhao, Lars Asker, Henrik Boström, and Panagiotis Papapetrou. Goldeneye++: A closer look into the black box. In Alexander Gammerman, Vladimir Vovk, and Harris Papadopoulos, editors, *Statistical Learning and Data Sciences*, pp. 96–105, Cham, 2015. Springer International Publishing.

[13] Tin Kam Ho. Random decision forests. *Proc. ICDAR*, 278–282, 1995.

[14] Willi Klösgen and Jan M. Zytkow, editors. *Handbook of Data Mining and Knowledge Discovery*. Oxford University Press, Inc., New York, NY, USA, 2002.

[15] Himabindu Lakkaraju, Stephen H. Bach, and Jure Leskovec. Interpretable Decision Sets: A Joint Framework for Description and Prediction. In *Proc. KDD*, pp. 1675–1684, 2016.

[16] Dennis Leman, Ad Feelders, and Arno Knobbe. Exceptional model mining. In *Proc. ECML PKDD*, pp. 1–16, 2008.

[17] Hongjun Lu, Rudy Setiono, and Huan Liu. NeuroRule: A Connectionist Approach to Data Mining. *ArXiv e-prints*, January 2017.

[18] Scott M. Lundberg, Gabriel G. Erion, and Su-In Lee. Consistent Individualized Feature Attribution for Tree Ensembles. *ArXiv e-prints*, February 2018.

[19] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[20] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?" Explaining the Predictions of Any Classifier. In *Proc. KDD*, pp. 1135–1144, 2016.

[21] Samuel Ritter, David G. T. Barrett, Adam Santoro, and Matt. M. Botvinick. Cognitive Psychology for Deep Neural Networks: A Shape Bias Case Study. *arXiv*, 2017.

[22] Marko Robnik-Šikonja and Igor Kononenko. Explaining classifications for individual instances. *IEEE Transactions on Knowledge and Data Engineering* 20(5):589–600, 2008.

[23] Marko Robnik-Šikonja, Igor Kononenko, and Erik Štrumbelj. Quality of classification explanations with PRBF. *Neurocomputing* 96(2012):37–46, 2012.

[24] Peter Russell, Stuart; Norvig. *Artificial Intelligence: A Modern Approach (2nd ed.).* Prentice Hall., 2003.

[25] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-CAM: Why did you say that? *arXiv preprint arXiv:1611.07450*, 2016.