Information-theoretic partition-based models for interpretable machine learning

Yang, L.

Chapter 1

Introduction

1.1 Data-driven Models

Data may contain a large amount of information about the underlying process that generated it. The larger the dataset is, the more information it may contain, but meanwhile the more difficult it may be for humans to distinguish useful patterns and regularities from noise and randomness.

Algorithms can handle large datasets that are intractable for humans to process, either by summarizing datasets and extracting patterns that are comprehensible for humans, or by building predictive models that construct relationships between variables. Models and algorithms in *data mining* and *unsupervised machine learning* concern the former, and those in *supervised machine learning* and *statistical predictive modeling* concern the latter.

1.1.1 Interpretable patterns for knowledge discovery

Due to our curious nature, we explore, reflect and learn from past experiences for all kinds of tasks. This includes building new connections between phenomena, discovering new knowledge from observations, and getting deep understanding about fundamental and complicated matters.

We can now enhance such activities with the help of large amounts of collected data, with the following application scenarios as examples. First, for instance, with the help of customer purchase records in supermarkets, associative patterns such as customers who buy coffee tend to buy milk and cookies as well can be "mined" from the large amount of records. In addition, with data collected by sensors installed in different places in manufacturing factories, anomalies can be detected and insight of what perhaps causes the anomalies may be identified. Furthermore, with the records of patients' conditions in a hospital, knowledge of factors that may lead to dangerous situations may be discovered. Finally, with trajectories recorded by GPS devices, regions with different popularity can be detected and insight about urban planning and/or police force distribution may be obtained.

Various models and algorithms exist for different kinds of tasks in inducing patterns from data, including and not limited to clustering, anomaly detection, association rules, frequent pattern mining, and density estimation. However, with the increased complexity of such models and algorithms, it has become an im-

portant research problem to seek for *interpretability and transparency*, i.e., to ask how the model outputs are produced. For instance, given some tabular data with a large number of variables, a key task in understanding the data is to investigate the dependency structure among variables, as widely used in *graphical models and causal inference*. Both transparent models with interpretable patterns and black-box models like deep neural networks are commonly used for this task. However, only the former can provide insight into why the model outputs (conditionally) "independent" or "dependent" for a certain subset of variables.

1.1.2 Interpretable predictive models

Besides knowledge discovery, building predictive models for a given target variable from data with *machine learning algorithms* has become successful in many areas, in the sense that the accuracy of such models are (beyond) "reasonably good" nowadays.

Examples for areas where such models are applied include the following. First, streaming media like Netflix can predict whether a customer may like a new movie or not, based on their historic watch data. Second, banks can predict whether someone is likely to be capable of paying back their debts given their bank account transaction data. Third, physicians in hospitals may use data collected by monitoring patients' conditions to help them predict the risk of certain operations.

However, accuracy of a predictive model is not the only concern when we consider introducing such a model and putting it into real use in our society, especially in critical areas such as health care (in which decisions are to be made about diagnosis or medication use, for instance), the judicial system (in which decisions are to be made about whether someone is guilty), and financial services (in which decisions are to be made about whether someone can get a mortgage, or someone is conducting fraud).

Because of ethical reasons and/or because of the severity of the outcome after false predictions, decisions with major influence on people or the society cannot be made automatically merely based on predictions given by machine learning models. That is, only humans should be responsible for taking actions in such critical areas.

Consider a scenario where a physician needs to decide whether a patient in an intensive care unit (ICU) of a hospital is good enough to be discharged. While all

the data records on the conditions of the patient can be useful, physicians can use a machine learning model trained on historic patients dataset to make predictions *only* for decision support, instead of letting the model directly make a decision for the patient.

Another example may be the situation where analysts in an insurance company must decide whether a client is conducting fraud. Even if all the transactions related to this client contain a lot of information about their behavior, it still remains the job of the analyst to extract evidence from the data and the model outputs as we cannot let the model take the responsibility in judging whether someone is guilty of conducting fraud.

Thus, substantial research has been put into *obtaining interpretability* of predictive machine learning models, to accelerate the introduction of such models in critical areas. This includes both 1) explaining black-box models and 2) developing new intrinsically interpretable models, with the general goal of providing transparency for data-driven decisions and building trust between the model and the end user (e.g., data analysts and domain experts) (Doshi-Velez and Kim 2017; Rudin et al. 2022).

1.2 What is Interpretability?

Interpretability is an umbrella term that can have very different meanings in different contexts. Conceptually, interpretability may refer to *transparency, global interpretability*, and *local interpretability* (Molnar 2020).

First of all, transparency concerns the extent to which a human can understand the process of an algorithm "learning" the model from data—how an algorithm takes the data as input and then outputs the model (Molnar 2020). Further, local interpretability often refers to an explanation of how the model output of a single instance is obtained; in contrast, global interpretability refers to the explanation of the model as a whole.

Thus, different models are considered (intrinsically) interpretable for different reasons. For instance, decision tree models (Breiman et al. 1984; Quinlan 2014) and rule-based models (Clark and Boswell 1991; Cohen 1995) are considered interpretable often due to the fact that the decision logic of every single prediction can be directly read by humans. In addition, linear models are consid-

ered interpretable as the marginal effect of a unit change in some feature value on a predicted value is described by a linear function, which is assumed to be "easily understandable". Further, generalized additive models (GAMs) are sometimes also considered interpretable (Caruana et al. 2015) as the marginal effect of feature changes on the target variable can be described by some non-linear function that can be visualized (and hence examined by the end user).

Meanwhile, what "interpretable models" mean in *unsupervised learning* is a bit more vague. While we may consider the K-means method for clustering interpretable, as it can more or less be explained why two instances belong to the same cluster (or two different clusters), it may be difficult to justify a clustering method based on deep neural network to be interpretable. Similarly, we may consider (linear) principle component analysis (PCA) interpretable as the associated "importance" for each dimension after the "rotation" of the basis of a vector space can be directly calculated; however, an embedding method based on an auto-encoder can hardly be understood by a human.

Yet, the concept becomes much more intuitive when we talk about interpretability in a *comparative* manner. For instance, a model that can make predictions together with *feature importance* (i.e., how much each feature "contributes" to the given model output) seems *more* interpretable than a model without feature importance. For instance, this is widely used in the field of *computer vision*, i.e., to attribute the model output to each pixel and visualize it (Adebayo et al. 2018).

Thus, one may argue that by introducing an approach for obtaining feature importance, the interpretability of a machine learning model class is increased[1]. Besides obtaining feature importance, it is also common to increase interpretability by 1) providing (local) surrogate models that are much "simpler" than the model to be explained (Ribeiro et al. 2016), and 2) reducing model complexity while maintaining predictive performance (Wu et al. 2018).

[1]Nevertheless, this may bring another issue that the method used for obtaining the feature importance may be complicated and hence cannot be regarded as transparent.

1.3 Partition-based Models

In this dissertation, we focus on partition-based models. Specifically, we consider *rule-based models* for supervised learning and *(adaptive) histogram models* for unsupervised learning, for which we now provide a very gentle introduction.

1.3.1 Probabilistic rule sets

A probabilistic rule is in the form of **IF some condition is met, THEN** $P(Y)$ **is equal to a certain value**, where $P(Y)$ denotes the (estimated) probability distribution for the target variable Y. As an example, consider a dataset that contains information on all flights in an airport within a certain period; then, one rule that may be induced from this dataset looks like **"IF Weather = Fog AND Flight_time $\leq$ 9 a.m. THEN P(Delay) = 0.8"**.

Further, a probabilistic rule set is simply a set of probabilistic rules put together. Rule sets are often considered as intrinsically interpretability models, as such probabilistic rules can be directly read and comprehended by humans. In Table 1.1 we show an example rule set learned from a real dataset that we will elaborate on in Chapter 4.

Condition of Rules	**Probability of Readmission to ICU**
Ureum-max-all $\geq$ 12.1 Ademfrequentie-median-value-last24h $\geq$ 23.5	0.223
APTT-max-all $\geq$ 43.1 Ureum-mean-all $\geq$ 16.338	0.199
Leukocyten-mean-last $\geq$ 20.81	0.162
Kalium-count-first $\geq$ 6.0 specialty-Organization-value-sub-ICCTC = FALSE	0.131
Trombocyten-count-first $\geq$ 2.0 Ureum-last-last $<$ 9.2 specialty-Organization-value-sub-ICCTC = TRUE	0.019
None of the above	0.059

Table 1.1: A rule set describing readmission risk that is learned from patients admitted to the intensive care unit of a hospital (described in detail in Chapter 4).

We are particularly interested in rule-based models due to the following reasons. First, one appealing property of rule-based models is that it connects interpretable predictive modeling and knowledge discovery, in the sense that it on one hand can be used for making (probabilistic) predictions for the target variable, and on the other hand, each rule is a local pattern that summarizes a subset of

the data and hence can be used for understanding the data itself and obtaining insights.

Second, the interpretability of rule-based models concerns both global interpretability and local interpretability. That is, individual rules can be used for explaining why a single prediction is made; meanwhile, a human can comprehend the rule set as a whole to grasp the internal logic of the model. Thus, rule-based models do not rely on post-hoc, external, and potentially non-transparent methods for obtaining interpretability.

Third, as rules are readable by humans, rule-based models are very accessible to domain experts who are not experts on machine learning methods. Thus, rule-based models are suitable to be used as a foundation for developing *interactive machine learning* methods: to allow the domain expert to give feedback to rules and to let the model incorporate the feedback by means of self-updating.

1.3.2 Multi-dimensional adaptive histograms

Histograms are widely used as a tool for visualizing the distribution of one- or two-dimensional data. For one- and multi-dimensional datasets in general, histograms can also be used as a tool for density estimation, data summarization, and discretization.

As an unsupervised partition-based model, histograms partition data points into bins, and within each bin the probability density is estimated as one constant. Specifically, an *adaptive* histogram is a histogram with variable bin sizes. For multi-dimensional histograms, bins may refer to as (hyper-)boxes or even more flexible subsets from a certain data partitioning process.

A multi-dimensional adaptive histogram is a simple yet powerful model that can effectively capture dependency structures among different dimensions. Specifically, multi-dimensional bins can be regarded as interpretable patterns that highlight subsets of data points for which the empirical marginal and conditional distributions differ from each other. This makes multi-dimensional adaptive histograms suitable for 1) discretization that incorporates the dependencies among dimensions, and 2) learning dependency structures for probabilistic graphic models.

We illustrate an example of a two-dimensional adaptive histogram in Figure 1.1, which is obtained by our proposed method that will be discussed in Chapter 5 on a simulated Gaussian dataset.

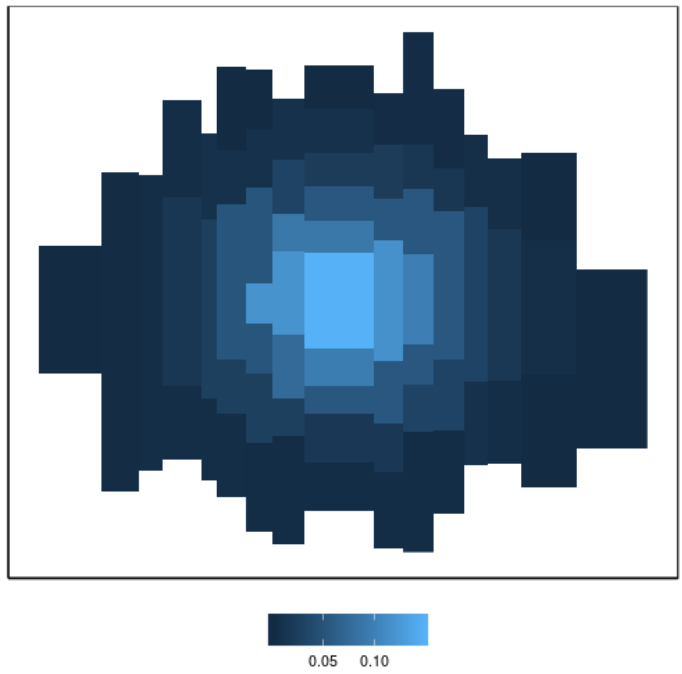

Figure 1.1: The histogram model for a simulated Gaussian dataset with density estimation (discussed in detail in Chapter 5).

1.4 A Gentle Introduction to the MDL Principle

We leverage information-theoretic tools and specifically the minimum description length (MDL) principle to formalize the problem of learning partition-based models from data as MDL-based model selection tasks.

The MDL principle has roots in information theory (Rissanen 1978). The core idea may be summarized as *learning by compression*. Specifically, the MDL principle states that the more we can compress the data in a *lossless* manner, the more structure and pattern we have found in the data. The degree of compression is measured by the code length, in bits, needed to encode data, together with the code length needed to encode the model that describes the regularities (structure and patterns) of the data.

Consider as an example learning regularities from the following two binary sequences: 1) a randomly generated binary sequence "100111101...", and 2) a binary sequence with the same length, which contains the regularity that a one is

always followed by a zero. Imagine we now need to communicate each sequence to a message receiver; for the first sequence, as there exists no regularity inside it, the only way is to enumerate each '1' and '0' in order. However, for the second sequence, we can first communicate the regularity itself to the message receiver, and then when we enumerate each '1' and '0' in order, we can skip the '0' after each '1' as the message receiver can add one '0' after receiving a '1' according to the regularity the receiver received. Thus, the number of bits needed to communicate the second sequence will be shorter than the length of the sequence itself. In this case, we say that the data is compressed with the help of the regularity. Reversely, regularity (instead of noise) is found if we find that it can be used to compress the data.

Thus, applying the MDL principle to certain tasks is about calculating the code length for the model and data together, which depends on the encoding scheme. Historically, choosing the encoding scheme was done in a crude and more or less arbitrary manner, and the earliest application of the MDL principle to partition-based models was to use the MDL principle in the well-known C4.5 (Quinlan 2014) and RIPPER (Cohen 1995) rule learning methods. In contrast, the modern version of the MDL principle (Grünwald and Roos 2019) exploits the connection between encoding and probabilistic modeling. Statistically, the length (in bits) of a given code[2] is connected to a corresponding probability distribution, as described by *Kraft's inequality* (Grünwald 2007).

The main motivation for adopting the MDL principle is that it removes the commonly used regularization parameter in the formalization of the learning problem, as the MDL principle automatically trades off between the goodness-of-fit and model complexity, which increases the transparency of how a learning method "creates" the model.

1.5 Research Questions

The overarching question we study in this dissertation is how to increase interpretability and transparency for partition-based models for supervised and unsupervised learning. This mainly concerns 1) how to make histograms more interpretable by having adaptive bins, as well as more transparent by reducing

[2]We assume all codes are prefix codes in this dissertation.

the number of user-defined parameters (e.g., the number of bins), and 2) how to increase the interpretability of rule-based models towards the level so that human-guided rule learning is possible. We next present our three main research questions in detail.

1.5.1 Towards rule sets for interactive rule learning

Although rule-based models carry significant interpretability because of the readability of the rules, our goal is to bring their interpretability to an even higher level so that domain experts can comprehend and potentially edit individual rules without considering the effect of/on other rules.

Consider a set of classification rules, each rule in the form of $\bigwedge\{X_i \in R_i\} \rightarrow Y \sim P(Y)$, in which X_i represents a single feature variable and R_i represents a set/range of values. For instance, a single rule could be denoted as **"Weather = Fog $\wedge$ Flight_time $\leq$ 9 a.m. $\rightarrow$ P(Delay) = 0.8"** .

Enhancing the interpretability of a set of such rules requires properly handling the "overlap" of rules, a long unresolved issue in learning rule-based models. Specifically, overlap refers to the case where one instance (e.g., one flight) satisfies the conditions of multiple rules, *potentially with different probabilistic predictions for the target variable* (e.g., flight delay).

As overlaps among rules make rules "entangled", we aim to enhance the interpretability of rule-based models by obtaining rules that are "independent" with regard to each other. Thus, we consider the following research question:

- **Research Question 1**: How can we formalize rule sets as probabilistic models such that the individual rules are independent from each other? Further, how can we learn such models from data?

Notably, due to the lack of a widely accepted general definition of interpretability, we consider interpretability in a comparative manner. Different from the common approach of seeking more interpretable rule-based models by making rules "simpler" (i.e., fewer and shorter rules), we instead consider making a rule-based model more interpretable by reducing the conflicts caused by overlaps among rules. We explain in detail why and how conflicts caused by overlaps affect interpretability in Chapters 2 and 3.

1.5.2 Adaptive histograms for discretization

Discretization is the task of summarizing continuous values and transforming them into a certain discrete representation form. It is a necessary pre-processing step if the following step in the modeling pipeline requires discrete values as input.

Intuitively, discretization methods need to strike a balance between the amount of preserved information and the complexity of the discretized representation (as a simple representation of data has benefits in terms of interpretability).

We specifically consider unsupervised discretization, i.e., discretization for a dataset without a target variable. Hence, the quality of the discretization cannot be evaluated by evaluating the prediction loss of the following step. Instead, it is crucial under such circumstances to *discretize the data in a way that makes sense to domain experts*, which concerns providing transparency regarding how the discretization is obtained.

Histogram-based models have the advantage of being very interpretable in discretization, data summarization, and density estimation (Kontkanen and Myllymäki 2007b; Scott 2015). However, while fixed histograms (histograms with equal bin sizes) are still widely used, they are often constructed with user-defined, more or less arbitrarily set parameters that control the number of bins (and hence the bin sizes). Thus, different patterns and, as a result, summarizations of a given dataset may exist, without any principled way of justifying which one represents the data more accurately. We argue that this may cause confusion to domain experts in practice, and hence negatively affects the trust in the model output by humans.

Further, while histograms as probabilistic models "approximate" the density of a given dataset by piece-wise constant values, existing methods lack a justification of whether the density inside each bin of a histogram is indeed (approximately) homogeneous, and at the same time, whether the density of neighboring bins are "very" different. Hence, the empirical distribution of data points within each bin is not transparent to domain experts in this case. Similarly, it often remains unclear whether "neighboring" bins have similar density estimates. For domain experts, merging such neighboring bins makes the model simpler and hence is beneficial for interpretability.

We specifically focus on two-dimensional datasets because spatial data is widely collected and analyzed, while a large number of existing algorithms for

mining spatial (or spatio-temporal) patterns require discrete values as input. This brings additional challenges as previous methods rarely considered the dependency of different dimensions, but applied a one-dimensional discretization method for each dimension separately.

To address these challenges, we propose our second research question:

- **Research Question 2:** how can we construct parameter-free two-dimensional histograms with transparent and informative patterns (bins)?

1.5.3 Histograms for learning dependency structure

We further exploit histogram-based models for the task of conditional mutual information estimation, which is useful in learning dependency structures among variables. That is, given three random variables denoted as X, Y, Z, the conditional mutual information (CMI) $I(X;Y|Z)$ characterizes whether X and Y are conditionally independent given Z. CMI estimation has wide applications in feature selection, conditional independence testing, and dependency structure learning (for graphic models).

We specifically consider CMI estimation for data with mixed types, of which each dimension can be continuous, discrete, and discrete-continuous mixtures. Although k-nearest neighbor (kNN) estimation is shown to work in such cases, we consider histogram-based models a more interpretable approach for such tasks, as each bin of the histogram can be regarded as an interpretable local pattern for explaining which subset of the data points contributes to the dependency among certain variables (and to what extent). This leads to our last research question:

- **Research Question 3:** How can we construct a multi-dimensional adaptive histogram-based model for interpretable CMI estimation?

Specifically, we extend our two-dimensional histogram-based models (discussed above) to multi-dimensional cases.

1.6 Contributions

This dissertation is composed of articles listed in the Table 1.2. The contributions for the paper corresponding to Chapter 6 were split half/half with Alexander Marx.

Article	Used in
Yang, L & van Leeuwen, M **Truly Unordered Probabilistic Rule Sets for Multi-class Classification**. In: Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECMLPKDD 2022), 2022.	Chapter 2
Yang, L & van Leeuwen, M **Probabilistic Truly Unordered Rule Sets**. Under review, submitted to JMLR.	Chapter 3
Yang, L, van der Meijden, S, Arbous, M.S & van Leeuwen, M **ICU Readmission Risk Analysis with Probabilistic Rule Set Model**. In Preparation.	Chapter 4
Yang, L, Baratchi, M & van Leeuwen, M **Unsupervised Discretization by Two-dimensional MDL-based Histogram.** Machine Learning, Springer, 2023.	Chapter 5
Marx, A, Yang, L & van Leeuwen, M **Estimating Conditional Mutual Information for Discrete-Continuous Mixtures using Multi-Dimensional Adaptive Histograms**. In: Proceedings of the SIAM Conference on Data Mining 2021 (SDM'21), 2021.	Chapter 6

Table 1.2: List of papers.

We briefly summarize the contributions of these chapters as follows. In Chapters 2 and 3, we introduce the truly unordered rule set (TURS) model and present our method for learning TURS models from data, which substantially improves the comprehensibility of rule set models. Specifically, in Chapter 2 we address the challenge of how we can treat overlaps as uncertainty in order to eliminate the need for post-hoc conflict-resolving schemes for overlap. We propose our first algorithm to learn TURS models from data, and showcase that rule sets learned from data, with overlaps representing uncertainty, can have on-par predictive performance in comparison to rule-based methods with explicit or implicit orders among rules (which are hence less interpretable).

Subsequently, in Chapter 3, we formalize the probabilistic modeling and the learning problem for TURS in a more rigorous way. Further, we propose a refined algorithm and conduct extensive experiments to present the appealing properties of learned models from various perspectives.

In Chapter 4, we apply the TURS model to the problem of ICU readmission risk analysis, and demonstrate that our method can be used for interactive rule

learning.

In Chapter 5, we study two-dimensional MDL-based histograms for unsupervised discretization. The main contributions are two fold. First, regarding the MDL theory, we show that the *parametric complexity* does *not* depend on the dimensionality of the data, which is defined as the *regret* term in the formula that calculates the code length (in bits). Second, we propose a novel method that can learn very flexible and expressive histograms for simulated and real-world datasets.

In Chapter 6, we extend the MDL histograms to multi-dimensional cases for the task of CMI estimation. Our main contributions include the following. First, we develop a series of theoretic results to construct our CMI estimator: 1) we define measure-theoretic entropy and prove the formula for calculating CMI based on entropy also holds for discrete-continuos mixtures, 2) we formally define histogram-based models for discrete-continuous mixture data, and 3) we prove the consistency of the proposed CMI estimator. Second, we propose an alternating algorithm to learn multi-dimensional adaptive histograms that are shown to be highly competitive when we benchmark against several widely used competitor methods.