

Chapter 2

Proactive Data Mining: A General Approach and Algorithmic Framework

In the previous section we presented several important data mining concepts. In this chapter, we argue that with many state-of-the-art methods in data mining, the overly-complex responsibility of deciding on this action or that is left to the human operator. We suggest a new data mining task, proactive data mining. This approach is based on supervised learning, but focuses on actions and optimization, rather than on extracting accurate patterns. We present an algorithmic framework for tackling the new task. We begin this chapter by describing our notation.

2.1 Notations

Let $A = \{A_1, A_2, \dots, A_k\}$ be a set of explaining attributes that were drawn from some unknown probability distribution p_0 , and $D(A_i)$ be the domain of attribute A_i . That is, $D(A_i)$ is the set of all possible values that A_i can receive. In general, the explaining attributes may be continuous or discrete. When A_i is discrete, we denote by a_{ij} the j -th possible value of A_i , so that $D(A_i) = \{a_{i,1}, a_{i,2}, \dots, a_{i,|D(A_i)|}\}$, where $|D(A_i)|$ is the finite cardinality of $D(A_i)$. We denote by $D = D(A_1) \times D(A_2) \times \dots \times D(A_k)$ the Cartesian product of $D(A_1), D(A_2), \dots, D(A_k)$ and refer to it as the input domain of the task. Similarly, let T be the target attribute, and $D(T) = \{c_1, c_2, \dots, c_{|D(T)|}\}$ the discrete domain of T . We refer to the values in $D(T)$ as the possible classes (or results) of the task. We assume that T depends on D , usually with an addition of some random noise.

Classification is a supervised learning task, which receives training data, as input. Let $\langle X; Y \rangle = \langle x_{1,n}, x_{2,n}, \dots, x_{k,n}; y_n \rangle$, for $n = 1, 2, \dots, N$ be a training set of N classified records, where $x_{i,n} \in D(A_i)$ is the value of the i -th explaining attribute in the n -th record, and $y_n \in D(T)$ is the class relation of that record. Typically, in a classification task, we search for a model—a function $f: D \rightarrow D(T)$, so that given $x \in D$, a realization of the explaining attributes, randomly drawn from the joint, unknown probability distribution function of the explaining attributes, and $y \in D(T)$, the corresponding class relation, the probability of correct classification, $\Pr[f(x) = y]$,

is maximized. This criterion is closely related to the accuracy¹ of the model. Since the underlined probability distributions are unknown, the accuracy of the model is estimated by an independent dataset for testing, or through a cross-validation procedure.

2.2 From Passive to Proactive Data Mining

Data mining algorithms are used as part of the broader process of knowledge-discovery. The role of the data-mining algorithm, in this process, is to extract patterns hidden in a dataset. The extracted patterns are then evaluated and deployed. The objectives of the evaluation and deployment phases include decisions regarding the interest of the patterns and the way they should be used (Kleinberg et al. 1998; Cao 2006; Cao and Zhang 2007; Cao 2010, 2012).

While data mining algorithms, particularly those dedicated to supervised learning, extract patterns almost automatically (often with the user making only minor parameter settings), humans typically evaluate and deploy the patterns manually. In regard to the algorithms, the best practice in data mining is to focus on description and prediction and not on action. That is to say, the algorithms operate as *passive* “observers” on the underlying dataset while analyzing a phenomenon (Rokach 2009). These algorithms neither affect nor recommend ways of affecting the real world. The algorithms only report to the user on the findings. As a result, if the user chooses not to act in response to the findings, then nothing will change. The responsibility for action is in the hands of humans. This responsibility is often overly complex to be handled manually, and the data mining literature often stops short of assisting humans in meeting this responsibility.

Example 2.1 In marketing and customer relationship management (CRM), data mining is often used for predicting customer lifetime value (LTV). Customer LTV is defined as the net present value of the sum of the profits that a company will gain from a certain customer, starting from a certain point in time and continuing through the remaining lifecycle of that customer. Since the exact LTV of a customer is revealed only after the customer stops being a customer, managing existing LTVs requires some sort of prediction capability. While data mining algorithms can assist in deriving useful predictions, the CRM decisions that result from these predictions (for example, investing in customer retention or customer-service actions that will maximize her or his LTV) are left in the hands of humans.

In *proactive* data mining we seek automatic methods that will not only describe a phenomenon, but also recommend actions that affect the real world. In data mining, the world is reflected by a set of observations. In supervised learning tasks, which are the focal point of this book, each observation presents an instance of the explaining

¹ In other cases, rather than maximal accuracy, the objective is minimal misclassification costs or maximal lift.

attributes and the corresponding target results. In order to affect the world and to assess the impact of actions on the world, the data observations must encompass certain changes. We discuss these changes in the following section.

2.3 Changing the Input Data

In this book, we focus on supervised learning tasks, where the user seeks to generalize a function that maps explaining attribute values to target values. We consider the training record, $\langle x_{1,n}, x_{2,n}, \dots, x_{k,n}; y_n \rangle$, for some specific n . This record is based on a specific object in the real world. For example, $x_{1,n}, x_{2,n}, \dots, x_{k,n}$ may be the explaining attributes of a client, and y_n , the target attribute, might describe a result that interests the company, whether the client has left or not.

It is obvious that some results are more beneficial to the company than others, such as a profitable client remaining with the company rather than leaving it or those clients with high LTV are more beneficial than those with low LTV. In proactive data mining, our motivation is to search for means of actions that lead to desired results (i.e., desired target values).

The underlying assumption in supervised learning is that the target attribute is a dependent variable whose values depend on those of the explaining attributes. Therefore, in order to affect the target attribute towards the desired, more beneficial, values, we need to change the explaining attributes in such a way that target attributes will receive the desired values.

Example 2.2 Consider the supervised learning scenario of churn prediction, where a company observes its database of clients and tries to predict which clients will leave and which will remain loyal. Assuming that most of the clients are profitable to the company, the motivation in this scenario is churn prevention. However, the decision of a client about whether to leave or not may depend on other considerations, such as her or his price plan. The client's price plan, hardcoded in the company's database, is often part of the churn-prediction models. Moreover, if the company seeks for ways to prevent a client from leaving, it can consider changing the price plan of the client as a churn-prevention action. Such action, if taken, might affect the value of an explaining attribute towards a desired direction.

When we refer to “changing the input data”, we mean that in proactive data mining we seek to implement actions that will change the values of the explaining attributes and consequently lead to a desired target value. We do not consider any other sort of action because it is external to the domain of the supervised learning task. To look at the matter in a slightly different light, the objective in proactive data mining is *optimization*, and not prediction. In the following section we focus on the required domain knowledge that results from the shift to optimization, and we define an *attribute changing cost* function and a *benefit* function as crucial aspects of the required domain knowledge.

2.4 The Need for Domain Knowledge: Attribute Changing Cost and Benefit Functions

The shift from supervised learning to optimization requires us to consider additional knowledge about the business domain, which is exogenous to the actual training records. In general, the additional knowledge may cover various underlying business issues behind the supervised learning task, such as: What is the objective function that needs to be optimized? What changes in the explaining attributes can and cannot be achieved? At what cost? What are the success probabilities of attempts to change the explaining attributes? What are the external conditions, under which these changes are possible? The exact form of the additional knowledge may differ, depending on the exact business context of the task. Specifically, in this book we consider a certain form of additional knowledge that consists of attribute changing costs and benefit functions. Although we describe these functions below as reasonable and crucial considerations for many scenarios, nevertheless, one might have to consider additional aspects of domain knowledge, or maybe even different aspects, depending on the particular business scenario being examined.

The attribute changing cost function, $C: D \times D \rightarrow R$, assigns a real value cost for each possible change in the values of the explaining attributes. If a particular change cannot be achieved (e.g., changing the gender of a client, or making changes that conflict with laws or regulations), the associated costs are infinite. If for some reason the cost of an action depends on attributes that are not included in the set of explaining attributes, we include these attributes in D , and call them *silent attributes*—attributes that are not used by the supervised learning algorithms, but are included in the domain of the proactive data mining task.

The benefit function $B: D \times D(T) \rightarrow R$ assigns a real value benefit (or outcome) that represents the company's benefit from any possible record. The benefit from a specific record depends not only on the value of the target attribute, but also on the values of the explaining attributes. For example, benefit from a loyal client depends not only on the target value of churning = 0, but also on the explaining attributes of the client, such as his or her revenue. As in the case of the attribute changing cost function, the domain D may include silent attributes. In the following section we combine the benefit and the attribute changing functions and formally define the objective of the proactive data mining task.

2.5 Maximal Utility: The Objective of Proactive Data Mining Tasks

The objective in proactive data mining is to find the *optimal decision making policy*. A policy is a mapping $O: D \rightarrow D$ that defines the impact of some actions on the values of the explaining attributes. In order for a policy to be optimal, it should maximize the expected value of a utility function. The utility function that we consider in this

book results from the benefit and attribute changing cost functions in the following manner: the addition to the benefit due to the move minus the attribute changing cost that is associated with that move.

It should be noted that the stated objective is to find an optimal policy. The optimal policy may depend on the probability distribution of the explaining attributes which is considered unknown. We use the training set as the empirical distribution, and search for the optimal actions with regard to that dataset. That is, we search for the policy that, if followed, will maximize the sum of the utilities that are gained from the N training observations.

It should be also noted that the cost, which is associated to O , can be calculated directly from the function C . The cost of a *move*—that is, changing the values of the explaining attributes from $x_i = \langle x_{1,i}, x_{2,i}, \dots, x_{k,i} \rangle$ to $x_j = \langle x_{1,j}, x_{2,j}, \dots, x_{k,j} \rangle$ is simply $C(x_i, x_j)$. However, in order to evaluate the benefit that is associated with the move, we must also know the impact of the change on the target attribute. This observation leads to our algorithmic framework for proactive data mining which we present in the following section.

2.6 An Algorithmic Framework for Proactive Data Mining

In order to evaluate the benefit of a move, we must know the impact of a change on the value of the target attribute. Fortunately, the problem of evaluating the impact of the values of the explaining attributes on the target attribute is well-known in data mining and is solved by supervised learning algorithms. Similarly our algorithmic framework for proactive data mining also uses a supervised learning algorithm for evaluating impact. Our framework consists of the following phases:

1. Define the explaining attributes and the target result as in the case of any supervised-learning task.
2. Define the benefit and the attribute changing cost functions.
3. Extract patterns that model the dependency of the target attribute on the explaining attributes by using a supervised learning algorithm.
4. Using the results of phase 3, optimize by finding the changes in values of the explaining attributes that maximize the utility function.

The main question regarding phase 3 is what supervised algorithm to use. One alternative is to use an existing algorithm, such as a decision-tree (which we use in the following chapter). Most of the existing supervised learning algorithms are built in order to maximize the accuracy of their output model. This desire to obtain maximum accuracy, which in the classification case often takes the form minimizing the 0–1 loss, does not necessarily serve the maximal-utility objective that we defined in the previous section.

Example 2.3 Consider a supervised learning scenario in which a decision tree is being used to solve a question of churn prediction. Let us consider two possible splits: (a) according to client gender, and (b) according to the client price plan. It

might be the case (although typically this is not the case) that splitting according to client gender results in more homogeneous sub-populations of clients than splitting according to the client price plan. Although contributing to the overall accuracy of the output decision tree, splitting according to client gender provides no opportunity for evaluating the consequences of actions, since the company cannot act to change that gender. On the other hand, splitting according to the client price plan, even if inferior in terms of accuracy, allows us to evaluate the consequences of an important action: changing a price plan.

Another alternative for a supervised learning algorithm is to design an algorithm that will enable us to find better changes in the second phase, that is, to design an algorithm that is sensitive to the utility function and not to accuracy. In Chap. 3 we propose a decision tree algorithm that displays these characteristics in regard to classification scenarios. Then, in chap. 4 we demonstrate that this alternative can contribute to the accumulated utility of the overall proactive data-mining task.

2.7 Chapter Summary

We observed in this chapter that data mining in general and supervised learning tasks in particular, tends to operate in a passive way. Accordingly, we defined a new data mining task, proactive data mining. We showed that shifting from supervised learning to proactive data mining requires additional domain knowledge. We focused on two aspects of such knowledge: the benefit function and the attribute changing cost function. Based on these two functions, we formally defined the task of proactive data mining as finding the actions, which maximize utility. We defined utility as benefit minus cost. We concluded the chapter by describing an algorithmic framework for proactive data mining.

References

- Cao L (2006) Domain driven actionable knowledge discovery in real world, PAKDD2006. pp 1021–1030
- Cao L (2010) Domain driven data mining, challenges and prospects. *IEEE Trans Knowl Data Eng* 22(6):755–769
- Cao L (2012) Actionable knowledge discovery and delivery. *WIREs Data Min Knowl Discov* 2(2):149–163
- Cao L, Zhang C (2007) The evolution of KDD: towards domain-driven data mining. *Int J Pattern Recognit Artif Intell* 21(4):677–692
- Kleinberg J, Papadimitriou C, Raghavan P (1998) A microeconomic view of data mining. *Knowl Discov Data Min* 2(4):311–324
- Rokach L (2009) Collective-agreement-based pruning of ensembles. *Comput Stat Data Anal* 53(4):1015–1026

Proactive Data Mining with Decision Trees

Dahan, H.; Cohen, S.; Rokach, L.; Maimon, O.

2014, X, 88 p. 20 illus., Softcover

ISBN: 978-1-4939-0538-6