Data Min Knowl Disc
DOI 10.1007/s10618-009-0158-x

Medical data mining: insights from winning two competitions

Saharon Rosset · Claudia Perlich ·
Grzergorz Świrszcz · Prem Melville · Yan Liu

Received: 23 March 2009 / Accepted: 19 November 2009

Abstract Two major data mining competitions in 2008 presented challenges in medical domains: KDD Cup 2008, which concerned cancer detection from mammography data; and Informs Data Mining Challenge 2008, dealing with diagnosis of pneumonia based on patient information from hospital files. Our team won both of these competitions, and in this paper we share our lessons learned and insights. We emphasize the aspects that pertain to the general practice and methodology of medical data mining, rather than to the specifics of each modeling competition. We concentrate on three topics: information leakage, its effect on competitions and proof-of-concept projects; consideration of real-life model performance measures in model construction and evaluation; and relational learning approaches to medical data mining tasks.

Keywords Medical data mining · Leakage · Model evaluation · Relational learning

Responsible editor: R. Bharat Rao and Romer Rosales.

S. Rosset
School of Mathematical Sciences, Tel Aviv University, 69978 Tel Aviv, Israel
e-mail: saharon@post.tau.ac.il

C. Perlich (✉) · G. Świrszcz · P. Melville · Y. Liu
IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA
e-mail: perlich@us.ibm.com

G. Świrszcz
e-mail: swirszcz@us.ibm.com

P. Melville
e-mail: pmelvil@us.ibm.com

Y. Liu
e-mail: liuya@us.ibm.com

Published online: 23 December 2009

1 Introduction

During 2008, there were two major data mining competitions that presented challenges in medical domains: KDD Cup 2008, which concerned cancer detection from mammography data; and Informs Data Mining Challenge 2008, dealing with diagnosis of pneumonia based on patient information from hospital files. Our team won both of these competitions, and in this paper we share our lessons learned and insights. We emphasize the aspects that pertain to the general practice and methodology of medical data mining, rather than to the specifics of each modeling competition. Although fundamentally different, these two challenges turned out to have several major characteristics in common. Correct understanding and handling of these characteristics are critical in our success in the competitions, and they should play a major role in many medical data mining modeling tasks. After an introduction to the competitions, their tasks, their data and their results in Sect. 2, we discuss three main issues that are in our view both critical and generalizable to the larger medical mining domain, and beyond:

Data leakage Fundamentally, data leakage is a lack of coordination between the data at hand and the desired prediction task, which exposes data that should not ‘legitimately’ be available for modeling. As a trivial example, a model built on data with leakage may conclude that people who miss a lot of work days are likely to be sick, when in fact the number of sick days should not be used in a predictive model for sickness (we will discuss later the time-separation aspect that causes the leakage in this example). In Sect. 3, we propose a definition, discuss causes, prevalence and detection, and offer some thoughts on a general methodology for handling and preventing leakage.

Adapting to measures of predictive performance Building models that are truly useful for medical purposes clearly requires taking into account the environment they will be used in and the decision process they are supposed to support. This is reflected in the choice of measures for model evaluation and selection. However, it also should affect the manner in which models are built. In some cases the estimation might be optimizing a specific measure directly. In other cases it may lead to model post-processing with the specific goal of improving the relevant performance measure. We discuss and demonstrate these different aspects in Sect. 4.

Relational and multi-level data The complexity of medical data typically exceeds by far the limitations of a simple ‘flat’ feature vector representation. The data often contain multiple levels (e.g., patients, tests, medications), temporal dependencies in the patient history, related multimedia objects such as ECG or images. Some of them require highly specialized modeling approaches, and at the very least a very thoughtful form of feature construction or relational learning. We discuss the complexity of the competition data in Sect 2 and present in Sect. 5 a number of approaches to represent and capture the complex interdependencies beyond the flat world of propositional learning.

It is important to note that while we use the two competitions as motivating examples for our discussion throughout this paper, our main goal in this paper is not to demonstrate good performance on them. Rather, it is to show how the three points above come into play in each of them. Consequently, we do not limit the discussion to solutions that work well, but rather present unsuccessful ones as well, if they lead to algorithmic or theoretical insights.

1.1 Competitions and real life projects

An important question pertains to the relevance of competitions in general, and their lessons learned in particular, to real life projects in medical modeling and other domains. We believe that the relevance is very high, and that most lessons learned from competitions, in particular the ones we discuss here, are bound to have implications for actual modeling project, for several reasons.

First and foremost, practically all real-life modeling projects start with a proof-of-concept and/or development phase, in which the feasibility and utility of the project are being examined. This phase often involves multiple external vendors competing for the project, or else a competition between internal groups in an organization, with differing approaches. Even if there is only a single modeling approach being considered, it is still critical to gauge its utility and return on investment in a proof-of-concept. To get useful information out of this phase, it is usually inevitable to arrange a 'competition-like' setup in which relevant data are extracted, models are built, and their performance examined (against each other in the case of a competitive process or against financial/performance targets). The important aspect here is not the competition, but the process of extracting and preparing data, then modeling and evaluating as in a competition. Only after a successful proof-of-concept can a judicious decision be made whether to make the much bigger investments and commitments involved in implementing the project or selecting a vendor. As far as this aspect of the modeling process is concerned, every single issue that comes up in competitions is directly relevant (and in our experience, also occurs in practice). Issues such as leakage, which could invalidate the proof-of-concept process, could have devastating long term effects on the success of modeling projects involving large investments.

Second, well organized competitions like the two we discuss here make an honest effort to mimic real-life projects, including the complications in the data and issues pertaining to real-life usefulness and evaluation approaches. Competitions, where ultimate predictive performance is the only criterion, require modelers to carefully consider these aspects, which are often treated off-handedly in real-life scenarios, due to lack of resources, or lack of the required technical skills in the project teams.

For our three main issues, the first (leakage) applies mainly to proof-of-concept scenarios, where it is a major and common problem in our experience. The other two (real-life evaluation and relational data) are more general, and are fundamental and critical for ensuring success. We address these points in more detail in the relevant sections below.

2 The two competitions: description of challenges, data and results

While the specific details of the two medical competitions are not the direct focus of this paper, we present here a brief overview to provide the context for our more general observations about data mining applications in medical domains.

2.1 KDD Cup 2008: breast cancer detection

KDD Cup is the oldest data mining competition and has been held for over 10 years in conjunction with the annual leading conference SIGKDD. It served as a forerunner and thanks to its success and popularity many similar venues have been started. The 2008 Cup was organized by Siemens Medical Solutions and consisted of two prediction tasks in breast cancer detection from mammography images.

The organizers provided data from 1,712 patients for training; of these 118 had cancer. Siemens uses proprietary software to identify in each image (two views for each breast) suspect locations that are called *candidates*. Each candidate was described by its coordinates and 117 normalized numeric features. No explanation of the features was given. Overall the training set included 102,294 candidates, 623 of which were positive. A second dataset with similar properties was used as the test set for the competition evaluation. For more details see Rao et al. (2008).

The two modeling tasks were:

Task 1 Rank the candidates by the likelihood of being cancerous in decreasing order. The evaluation criterion for this task was a limited area under the free-response operating characteristic (FROC) curve, which measures how many of the actual *patients* with cancer are identified while limiting the number of *candidate* false alarms to a range between 0.2 and 0.3 per image. This was meant to reflect realistic requirements when the prediction model is used as an actual decision support tool for radiologists.

Task 2 Suggest a maximal list of patients who are surely healthy. In this task including any patient with cancer in the list would disqualify the entry. This was meant to be appropriate for a scenario where the model is used to save the radiologist work by ruling out patients who are *definitely healthy*, and thus the model was required to have *no false negatives*.

Our winning solutions to Task 1 included three main components (more details on our methodology can be found in Perlich et al. (2008)):

1. *Leakage* Our initial data analysis identified that the patient IDs carried predictive information about a patient's likelihood to have cancer. We discuss the details of this phenomenon in Sect. 3. We included this information as an additional feature for the classification model.
2. *Classification model* Linear models seemed to be most suitable for the task. We considered a number of model classes including logistic regression, SVM, neural networks, and decision trees. The superiority of logistic regression and linear SVM are probably due to the nature of the 117 normalized numeric features and the dangers of overfitting with such few positive examples for less constrained model classes.

Table 1 FROC comparison, between 0.2 and 0.3 candidate false alarms per image, for different models on the KDD Cup 2008 Task 1 using tenfold cross validation on the training data (above the line), and actual competition results on the test set (below the line)

Model	FROC
Linear SVM	0.0834
Leakage only	0.0736
Linear SVM +leakage	0.0882
Linear SVM + leakage + bagging	0.0902
Winning solution: linear SVM + leakage + bagging + post processing	0.0933
Second place competitor on test set	0.089

3. *Post processing* The FROC evaluation metrics for Task 1 is considerably different from traditional machine learning evaluation measures such as accuracy, log-likelihood or AUC. We optimized the model scores for the FROC as shown in detail in Sect. 4. The solution for task two required predictions on the patient level. We again used some form of post-processing to aggregate the candidate level predictions.

Our final submission used bagged linear SVMs (Valentini and Dietterich 2003) fitted using the SVMlight package (Joachims 1999) with an additional identifier-based feature, maximizing zero-one loss, $c = 20$ and heuristic post processing. This approach scored the winning result of 0.0933 on the test set compared to 0.089 of the runner up. We show some comparative results in Table 1. Since the organizers never published the true labels in the test set, we show for some of our intermediate models the 10-fold cross validation results. The winning submission was actually never submitted to this process, since bagging was a lengthy process, and we did not have time to try it out. We had evidence that the post-processing was useful (see Sect. 4), and we knew bagging worked well, so we took the chance of submitting this solution without extensive validation. In this table we indeed see very strong improvements when combining all three components: leakage, modeling approach (bagging) and post-processing.

For the most part, we re-used the Task 1 methodology for the submission to Task 2. Obviously the aggregation to patients is different from the Task 1 post-processing. In addition, we switched from linear SVM to logistic regression. This model performed slightly better here due to the high sensitivity of likelihood to extreme errors. We submitted the 1020 first ranked patients from a logistic model that included the leakage features in addition to the original 117 provided features.

2.2 INFORMS data mining contest

The first INFORMS Data Mining Contest was announced by the Data Mining Section of INFORMS in April 2008. The focus of the task was on nonsocomial infections—infections obtained while staying in the hospital secondary to the patient's original condition.

The contestants were given 2 years of patient data from 2003 and 2004 in four separate files, including a hospital file with the information whether a patient contracted

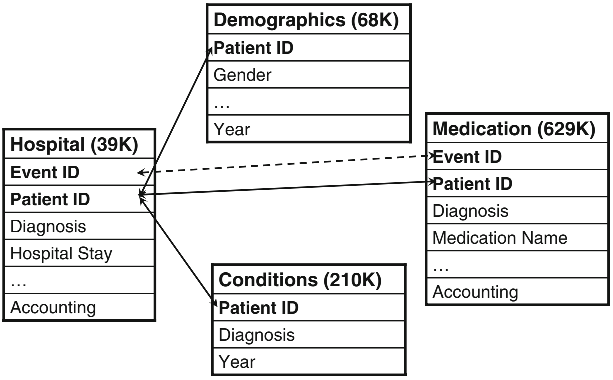

Fig. 1 Database schema of the medical domain for the INFORMS competition. The Hospital file is the main focus of the contest and only links from it are included

an infection during a medical procedure. Figure 1 shows the database schema of the 4 provided tables, how they link to the main hospital table through one of two keys (Patient ID and Event ID), and the number of rows in each of them. Near the end of the contest, data from patients during 2005 were provided. Similar to the KDD Cup, the INFORMS contest had two tasks, but only the first focused on data mining.

Task 1 The goal of the first task was to detect instances in the hospital file that contained the diagnosis code for nonsocomial pneumonia in one of the four provided diagnosis columns. Observe that contrary to most data mining competitions, the participants were asked to design a 'clean' training set with a target label themselves.

Task 2 The second task was more aligned with the actual decision process and required the design of a cost metric and the design of some optimal treatment policy. We did not participate in that task.

Our winning solution to the first task also had three main components:

1. *Cleaning and the design of a suitable representation* Similar to most real-world data, this dataset was rather messy. First we observed plenty of missing numerical values in some columns of the hospital and medication table. We also observed that a large set of hospital rows had large parts of features consistently missing—we suspect these were mostly accounting entries.[1] We ultimately removed some of the numeric columns from the hospital tables that had mostly accounting information (how much was charged against which insurance and how much was ultimately paid). The demographic table had plenty of duplicates for the same patient, but with different feature values. We decided to remove all duplicates and pick randomly one of the two feature sets. We also considered the 'distributed' appearance in multiple tables of the diagnosis codes—one of the most relevant pieces of information—to be unsuitable for modeling and pooled all diagnosis codes in just one condition table and removed them from the other tables. We finally converted the medication names into a bag-of-word representation.

[1] We strongly suspected that a row in the table did not correspond to a particular visit, but rather to some event during a visit—even just the submission of a bill to an insurance.

Table 2 AUC comparison for different models on the INFORMS 2008 challenge using tenfold cross validation on the training data (above the line), and actual competition results on the test set (below the line)

Model	AUC
Logistic propositional	0.81
Leakage type 1 only	0.84
Leakage general only	0.86
Logistic + relational	0.90
Logistic + relational + leakage	0.91
Winning submission: logistic + relational + leakage + testtraining	0.88
Second place competitor on test set	0.83

2. *Relational learning* We observed two types of relational characteristics in this domain. Aside from the demographic information which (after duplications were corrected) can be joined easily and can be included directly in the hospital table; the relationship between each row in the hospital table and the other two tables is a one-to-many relationship. There is no simple solution to a one-to-many case and we decided to use a propositionalization (Krogel and Wrobel 2003) approach and use the ACORA (Perlich and Provost 2006) system to automatically bring in and aggregate the relevant information. We also noted a potential manipulation in the linkage between the rows in the hospital table for the same patient and event.[2]
3. *Leakage* The main challenge for the organizers of this competition was the fact, that the data actually contained the target deeply embedded. Three of the four tables contained a diagnosis code field, and the target of interest could show up in any of them. To make the competition worthwhile, all instances needed to be removed. This turned out to be impossible without leaving certain traces of alteration behind. And indeed we observed that a certain combination of missing diagnosis code and some other features provided predictive information. This could identify a substantial subset of hospital rows as certain positives and others as certain negatives. Similar to the KDD Cup solution we included this information as additional feature on two levels. We discuss this in more detail in Sect. 3.

We show the relative performances of the components in Table 2. Our final submission for the competition used logistic regression on one leakage feature, some of the hospital features and the automatically constructed features using ACORA on the slightly modified representation of the three additional tables. We also took advantage of the identified leakage, and included some additional examples from the test set (labeled by leakage) into our training set for the final model.

It is interesting to note the decreased performance of our submitted model on the test set (AUC = 0.88) compared to our in-sample tenfold CV estimate of 0.91 without the additional leakage-based labels. It is possible that the usage of these labels neg-

[2] In an attempt to recreate a patient history from the hospital file we observed that no patient had more than 1 year history and only a very small but consistent percentage of patients appeared in two consecutive years, but none in 3 consecutive years. There is no predictive information in this, but some evidence of an id assignment process.

atively impacted our performance, but that seems unlikely. More likely, there was a substantial concept drift, as the test was done on a different year.

An important observation about two of our major topics in this paper relates to the ability of relational learning to ‘automatically’ identify and make use of non-trivial leakage. In this case it appears that employing the relational data for feature construction allowed the linear model to capture the majority of leakage implicitly in some indirect way, in addition to other legitimate information not available to the ‘flat’ approaches. This explains the relatively modest improvement of adding the leakage to the relational model in Table 2.

2.3 Similarities across both domains

At this point we would like to point out similar characteristics of both domains that are very typical for medical data and should have strong implications on the application of data mining to medical domains. We explore the last three in more detail in the remaining sections of the paper.

- *Independent data collection* In both cases the data were collected independently of the particular data mining effort. In particular, the collection process was a part of the standard medical procedures long prior to the modeling. This leads to some problematic artifacts, that are related to the leakage issues we are discussing in Sect. 3. One of the issues is that no precise time stamps were recorded in the case of the hospital data. This prevented the proper data cleaning and could have lead to the situations where the model identified effects instead of causes.
- *Privacy issues* There have been strong privacy concerns in almost all medical datasets. As a result, some of the information had to be removed and other was obscured for the sake of preserving the identity of the patient. While the first may just limit the quality of models, the latter can be a part of the process that ultimately leads to the examples of leakage we observed. In particular, replacing true patient identifiers like social security number with some IT-system generated number that would carry information about the time and place it was recorded.
- *Leakage* In both domains we were able to build a ‘more accurate’ model based on information that either should not have been predictive (the identifier in KDD Cup 2008) or would most likely not be available in a real application scenario of the model (the trace of diagnosis removal in INFORMS). In Sect. 3 we discuss implications, identification, and prevention of the leakage.
- *Non-trivial evaluation* In most real-world applications the standard performance measures used in the data mining literature are of limited value. Most models are used in the framework of decision support and the application-specific decision process is often highly complex and not entirely quantifiable. While cost-sensitive learning has been focusing on some of the issues arising in decision support, medical applications often have additional legal and practical constraints that go far beyond the existing work in machine learning. In Sect. 4 we discuss the issue of evaluation in more detail.
- *Relational data* We observed in both domains rich relationship information between examples. In the case of the KDD Cup, different candidates belong to

the same breast of one patient and have some spatial relationship. Similarly, multiple rows in the hospital table are also linked to the same patient and should clearly influence the prediction of the model. In addition, we observed in the INFORMS case the typical relational database structure where relevant information for the modeling task is located in additional tables and cannot simply be joined if there is a one-to-many relationship between the entities. This scenario calls either for feature construction (manual or automatic) or a first order model representation that is able to express such dependencies.

3 Information leakage

In the context of predictive modeling, leakage is the unintentional introduction of predictive information about the target by the data collection, aggregation and preparation processes. As a trivial example, assume we are building a model trying to predict which people are likely to get the flu. If the model is built on data with leakage, it may conclude that people who miss a lot of work days are likely to be sick, while in fact the number of sick days is a consequence of getting the flu, and should not be used in a predictive model for sickness (we will discuss later the time-separation aspect that causes the leakage in this example). Such information leakage—while potentially highly predictive out-of-sample *within* the study—leads to limited generalization and model applicability, and to overestimation of the predictive performance. As we elaborate below, such leakage was present in both competitions discussed here. However, it is by no means limited to such competitions—practically every modeling project has a proof of concept or model development phase, in which historical data are used to simulate the 'real' modeling scenario, build models, evaluate them, and draw conclusions about which modeling approaches are preferable and about expected performance and impact. In our experience, many such real life proof of concept projects are plagued by leakage problems, which render their results and conclusions useless, often leading to incorrect conclusions and unrealistic expectations. It is also common in data mining competitions. They resemble in some respects the proof of concept state of projects, where data are prepared for the explicit goal of evaluating the ability of various tools/teams/vendors to model and predict the outcome of interest. Examples of leakage in competitions include the two competitions we discuss here and also KDD Cup 2007 (Rosset et al. 2007), where the organizers' preparation of the data for one task exposed some information about the response for the other task; and KDD Cup 2000 (Inger et al. 2000), where internal testing patterns that were left in the data by the organizers supplied a significant boost to those who were able to identify them.

While it is clear that such leakage does not represent a useful pattern for real applications, we consider its discovery and analysis an integral and important part of successful data analysis.

Two of the most common causes for leakage are:

1. Combination of data from multiple sources and/or multiple time points, followed by a failure to completely anonymize the data and hide the different sources.
2. Accidental creation of artificial dependencies and additional information while preparing the data for the competition or proof-of-concept.

Our definition of leakage is related to a problem-dependent notion of what constitutes ‘legitimate’ data for modeling. It is related to several notions in the literature, including spuriousness (Simon 1954) and causality (Glymour et al. 1987)—causal and non-spurious associations are guaranteed to be legitimate. However, non-causal associations can also be legitimate, as long as they are legitimately useful for prediction.

An interesting related concept commonly discussed in the medical context is that of ‘double blinding’, which intends to prevent ‘subjective’ knowledge about case-control allocations from affecting the ‘objective’ results of clinical trials. It has been noted in the literature that such blinding often fails to accomplish this goal, for example because irrelevant side effects of the medication help the ‘blinded’ doctors identify which patients are getting the treatment and which are on placebo (White and Dufresne 1997). Although clinical trials are usually a hypothesis testing scenario and not a predictive modeling one, and thus this problem does not exactly correspond to our notion of leakage, the phenomenon of information leakage is similar in nature.

3.1 Leakage in the competitions

KDD Cup 2008 data suffered from leakage that was probably due to the first cause above. The patient IDs in the competition data carried significant information towards identifying patients with malignant candidates. This is best illustrated through a discretization of the patient ID range, as demonstrated in Fig. 2. The patient IDs are naturally divided into three disjoint bins: between 0 and 20,000 (254 patients; 36% malignant); between 100,000 and 500,000 (414 patients; 1% malignant); and above 4,000,000 (1,044 patients, of them 1.7% malignant). We can further observe that all 18 afflicted patients in the last bin have patient IDs in the range 4,000,000–4,870,000, and there are only three healthy patients in this range. This gives us a four-bin division of the data with great power to identify sick patients. This binning and its correlation with the patient’s health generalized to the test data. Our hypothesis was that this leakage reflects the compilation of the competition data from different medical institutions and possibly from different equipment, where the identity of the source is reflected in the ID range and is highly informative of the patient’s outcome. For example, one source might be a preventive care institution with only very low base rate of malignant patients and another could be a treatment-oriented institution with much higher cancer prevalence.[3]

In the INFORMS 2008 competition, the leakage was a result of an attempt to remove only the occurrence of pneumonia while leaving the rest of the patient record untouched, creating abnormally looking patient records. In particular, any patient record that had no conditions mentioned was more likely to be a positive example, i.e., a patient that had only pneumonia-related conditions in the record, which were then removed. Some additional glitches in the removal process exacerbated this problem. In this instance, it was easy to build models that benefited from the leakage without even being aware of it. To clarify how, we give a few additional details on the

[3] The organizers later explained that in order to increase the number of positive examples, the dataset was comprised of examples from different time periods.

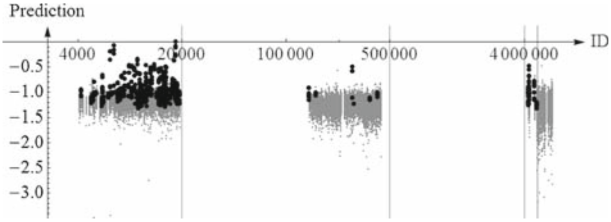

Fig. 2 Distribution of malignant (*black*) and benign (*gray*) candidates depending on patient ID on the X-axis in log scale. The Y-axes is the score of a linear SVM model on the 117 features. *Vertical lines* show the boundaries of the identified ID bins

data and removal process: the hospital records contained fields that held codes for each medical condition of this the patient record (up to four different codes, named icdx1 to icdx4), and an indicator SPECCOND of whether or not the record actually pertains to *any* medical condition (as opposed to accounting records, for example). Any record in the test data with NULL in all icd# fields and 1 in SPECCOND was guaranteed to be a leakage-based positive. Thus, a model (say, logistic regression) which uses the observed number of condition codes of the patient as a *categorical* variable and also the variable SPECCOND would have been able to nail down the leakage-based effects by assigning a high weight to both the case of no condition codes and SPECCOND=1, and lower weights for records that have condition codes. As we show below, a more complex relational modeling approach leads to taking advantage of the leakage in even less obvious ways. This point is critically important when we consider the competitions as simulating proof of concept projects, since they corresponded to a case where, even without careful analysis and identification of the leakage, predictive models would still have been likely to take advantage of the leakage. This would obviously render their evaluation on held-out data (with the leakage present) useless in terms of real prediction performance. The performance impact due to leakage is shown in Table 2. Type 1 refers to the positives that can be identified by the above rule and has by itself and AUC of 0.84. The information spreads even further—we can observe an additional increase in performance to 0.86 when we also flag all records of patients that have type 1 leakage.

3.2 Detection

We feel that it cannot be overstated how important and difficult complete leakage detection and avoidance really is. We are by no means certain that we have observed all leakage issues in the above competitions or in our proof of concept modeling projects. Contrary to our discovery and intentional exploitation of leakage in the artificial competitive settings, the much more common scenario is a real world application where the model takes advantage of some leakage WITHOUT the modeler even being aware of it. This is where the real danger lies and what may be the cause of many failures of data mining applications.

While for KDD Cup 2008 it seems clear that the patient ID should NOT be part of a model, the INFORMS example demonstrates a case where we could have accidentally built a model that used leakage without knowing about it, through the relational modeling approach.

So how can one find out that there might be an issue? We discuss three different approaches for detection of leakage: exploratory data analysis (EDA), model evaluation, and real use-case scenario.

Exploratory data analysis EDA seems to have become something of a lost art in the KDD community. In proper EDA, the modeler carefully examines the data with little preconception about what it contains, and allows patterns and phenomena to present themselves, only then analyzing them and questioning their origin and validity (NIST 2006). It seems that many instances of leakage can be identified through careful and thoughtful EDA, and their consequences mitigated. In the two competitions we discuss here, EDA was critical for identifying and characterizing the leakage. In KDD Cup 2008, the patient ID is not naturally a variable one would use in building models for malignancy detection, but EDA led us to the image seen in Fig. 2 and its consequences. In the INFORMS competition, EDA supported the discovery of the glitches in the removal mechanism. We hope that our discussion here can serve as a reminder of the value of open-minded exploratory analysis.

Critical model evaluation The second key tool in leakage detection is critical examination of modeling results. Ideally, one should form a concept of what predictive performance a ‘reasonable’ model is expected to achieve, and examine the results on held-out data against this standard. Models that perform either much worse or much better than their reasonable expectation should be investigated further. If no such prior concept of reasonable performance exists, the performance of various modeling approaches on the same task can be compared, and significant differences should be further investigated. For example, in one experiment on the INFORMS challenge data, a logistic regression model which used the number of condition codes as a numerical variable gave hold-out area under the ROC curve (AUC) of 0.8. By switching this variable to categorical, the AUC increased to 0.88. Such a significant improvement from a small change in model form should have raised some concerns, as could the fact that this implies a non-monotonic relationship between the number of diagnosis codes and the probability of contracting pneumonia. Judicious comparison of the two models would have been likely to expose the leakage, had it not been discovered by EDA.

Exploration of usage scenarios Finally, in the spirit of ‘The proof of the pudding is in the eating’, a very relevant strategy for leakage detection is to push early during the proof-of-concept to get as close as possible to the true application setting. This might involve extended communications with the potential future users or domain experts, considerations of the real data feeds that will be utilized at the time, etc. It might also include an early real-world test run that puts the models into place and monitors their performance over a period of time, and compares it to the prior expectations and out-of-sample results.

3.3 Approaches for leakage avoidance

Our definition of leakage points at data that should not 'legitimately' be available to the model. The prevention or avoidance of leakage is therefore intimately tied to a careful definition of what the modeling problem is and how the model will be used, in order to judge, what data can and cannot be used as part of the predictive model.

One important scenario where, in principle, leakage can be completely avoided, is based on the famous saying, attributed to Niels Bohr: "Prediction is very difficult, especially about the future". Medical applications of data mining are typically tied into some decision process: Should the patient be examined further and biopsies be taken or sent home? Should an incoming patient be given special preventive treatment against pneumonia or not? All such decision processes have a temporal component: there are things that are known at the time of the decision, and there are outcomes (pneumonia infection) and consequences of actions taken (antibiotics given) that are only known later.

In these decision scenarios, leakage can be avoided by a clean temporal separation of (1) the data that can be used as explanatory variables for modeling up to the decision time and (2) everything thereafter, in particular the predicted outcome and any possible implication thereof. The formal definition of the predictive modeling task is to build a model $\hat{y} = \hat{f}(\mathbf{x})$ which describes the dependence of y on $\mathbf{x}$ and will be used to predict y given new values of $\mathbf{x}$. In *prediction about the future*, we further assume that at prediction time all the explanatory data in an observation $\mathbf{x}$ are observed and predictions are made at some time $t(\mathbf{x})$, and the response y is determined only at a later time, say $t(\mathbf{x}) + \Delta t$. The task is, naturally, to make a good prediction at time $t(\mathbf{x})$ about what y will be.

If this is indeed the case, then an obvious approach to avoid leakage is to make sure that the data used for modeling complies with this time separation as well. Assume the data for training are made of n observations $\{\mathbf{x}_i, y_i\}_{i=1}^{n}$. Then to avoid leakage, one simply has to make sure that the values in $\mathbf{x}_i$ were observed at the appropriate 'observation time' $t(\mathbf{x}_i)$ and not affected by any subsequent updating and information flow, including (but not limited to) the observation of the response y_i.

This seemingly simple requirement is often hard to implement when dealing with databases that are constantly updated and enhanced, however, when it is successfully implemented, it guarantees non-leakage. The importance and difficulty of this requirements was previously noted in medical applications (Shahar 2000; Russ 1989). If it cannot be implemented, it is advisable to investigate the reasons for the difficulties in implementation, and that process itself may expose potential leakages. We now discuss the implementation of *prediction about the future* in the two competitions and one other predictive modeling scenario.

Pneumonia prediction Taking the INFORMS challenge as an example, the response y was the existence or non-existence of pneumonia, and the explanatory data $\mathbf{x}$ contained all hospital records with the references to pneumonia removed (as well as other relational information such as medications, which we ignore for simplicity). This formulation already violates the *prediction about the future* paradigm, because the hospital records contain information that was updated after the onset of pneumonia.

In fact, it seems that a modeling task based on hospital records (of which each patient may have multiple) has no chance of complying with this paradigm. However, if the task was to be switched to the patient level, and information was available of the date on which each record was created and each condition was diagnosed, then one could hope to create a *prediction about the future* modeling scenario. In this scenario, the explanatory data $\mathbf{x}_i$ for each patient would represent all the information available about this patient up to some time t_i and the response y_i would correspond to appearance of pneumonia in some fixed Δt after this time.

Breast cancer identification The example of KDD Cup is an interesting one what highlights the need to define the prediction task very clearly. It is not entirely obvious whether the patient ID is a case of leakage or not. Let us assume that the ID is indeed an indicator of a certain subset of the population. Is it legitimate to use this information? The answer depends on whether the assignment of a patient to a subset was an outcome of her having cancer (which seems to be the case in this competition). If yes, then using this ID would clearly be a violation of the *prediction of the future* rule. If on the other hand, the sub-populations are coincidental, e.g., there may be geographical or demographical locations that have a higher cancer prevalence rates, then it would be legitimate to incorporate this information into a model that is used across those different populations. It seems, however, awkward to define the population based on the range of the patient IDs and the optimal model should ideally be given a more direct indicator of the legitimate underlying driver of this change in prevalence rate.

A business intelligence example Modeling propensity to purchase IBM products by companies (Lawrence et al. 2007), we defined $\mathbf{x}$ as the historical relationship a company has with IBM up to some fixed time t (say, end of year 2006), and its firmographics (i.e., characteristics of the company). The response y was the purchase of the product in some period Δt (say, 1 year) following t. However, in later work we sought to also utilize information from companies' websites to improve the model (Melville et al. 2008). This appeared to be very useful, but we encountered a problem with *predicting about the future*—the websites were only available in their present form, and we had no access to what they looked like at the end of 2006. Indeed, by examining their content we found an obvious leakage, where the name of IBM products purchased (such as Websphere) often appeared on the companies' websites. A predictive model built on such data would naturally conclude that the word Websphere indicates propensity to buy this product, but the true time relationship between the purchase and the appearance is likely reversed. We removed the obvious leakage words manually, but the potential for more subtle leakage remained.

4 Adapting to real-world performance measures

For predictive modeling solutions to be useful, in particular in the medical domain, it is critical to take into account the manner in which these models will ultimately be used. This should affect the way models are built, judged, selected and ultimately implemented, to make sure the predictive modeling solutions actually end up addressing the problem in a useful and productive manner.

In many real life applications, and in particular in the medical domain, the model performance measures are very different from the standard statistical and data mining evaluation measures. For classification and probability estimation the typical standard evaluation measures are accuracy, likelihood, and AUC. They have their benefits in terms of general properties such as robustness, invariance, etc. However, the same properties that make them useful for data mining evaluation across domains typically render them irrelevant for a particular application domain with the goal of supporting a specific decision.

For example, in KDD Cup 2008, the organizers chose to rely on the specialized *Free Response ROC* (FROC) curve (Bandos et al. 2008; DeLuca et al. 2008), which we explain in detail below. This reflected their conception of how the resulting scores will be used, and what is required to make them most useful. In this case, they surmised that physicians are likely to be comfortable with manually surveying an average of 0.2–0.3 suspicious regions per patient image (or about one suspicious region per patient), when the patient is in fact healthy. Accordingly, the performance criterion essentially measures what percentage of actual malignant patients would be identified in this scenario (by having at least one of their malignancies flagged).

The measure of the second task for the same competition had a similar justification: Since the cost of a false negative (sending a sick patient home) is close to infinite, the performance criterion used was the maximal number of patients that a model can rule out, provided they contained no false negatives.

The first task in the INFORMS contest relied on the AUC, which is a standard performance measure in data mining and not really specific to medical domains. However, the second task asked explicitly for the design of an appropriate metric as well as a preventive strategy. We did not participate in this task due to time constraints, but we see again a special focus on evaluation.

Intuitively, the specific choice of performance metric should drive the construction and selection of models, and to some extent the issue of model performance for decision support has been considered in sub-areas of data mining such as utility-based data mining (Weiss et al. 2008) and cost-sensitive learning (Turney 2000). However, there are still two fundamental inhibitors to a greater focus on real-world measures, namely,

- There is not just one relevant ‘real’ measure, but as many as there are applications, for every application requires a potentially different performance measure. Furthermore, these measure are often not fully defined because the ‘cost’ of decisions can typically only be approximated.
- Many of the empirically relevant measures present statistical difficulties (high variance, non-robustness, non-convexity, etc.) This often makes statistically valid inference difficult, and hinders progress.

4.1 Impact of performance measures on the modeling process

Considerations of the ultimate performance measure should enter the modeling process in different stages. It is not just a question of final evaluation, but might come into play much earlier in the model building stage. As a simple example, if the performance measure is the sum of absolute errors, one might consider estimating a model using absolute loss rather than squared error loss. However, it is often the case that integration of complex performance criteria into the model building stage is very difficult, due to computational difficulties (e.g., if it results in non-convex loss functions) or implementation difficulties, including a reluctance to forsake tried and tested modeling tools which use ‘standard’ objectives in favor of development of new ones for ‘specialized’ objectives.

An interesting example is the modeling approaches that have been developed in the classification community, which use the area under the ROC curve (AUC) as an optimization criterion instead of the error rate or its convex approximations (Ferri 2002; Joachims et al. 2005). This was an attempt to directly build models that are expected to perform well in terms of AUC performance, directly using the non-convex AUC as an optimization objective for modeling. However, these approaches did encounter computational difficulties, and it was not always clear that they do empirically better than ‘standard’ classification approaches in practice, in terms of predictive AUC. Thus, even for a commonly used measure like AUC, it has proven difficult to make much progress by designing specialized modeling algorithms, compared to using standard ‘out of the box’ tools.

An alternative, less ambitious approach is to use the performance measure as a guide for *post processing* of the results of standard modeling approaches. In this approach, once model scores have been calculated, one might ask, how should these be manipulated, changed, or re-ordered, given the real-life performance measure to be used for the model. For the rest of this section, we concentrate on this approach, and its application to Task 1 of KDD Cup 2008.

4.2 KDD Cup 2008 example: optimizing FROC

As already discussed, the performance measure for the KDD Cup Task 1 is specifically designed by the radiology community and goes beyond the typical variations of evaluation in machine learning. Our research into model post-processing approaches to optimize this measure led us to results and algorithms that were interesting and important, both theoretically and empirically. We present them for their independent interest, but also as a case study into the value of post-processing for adapting to real-world performance measures. We present some comparative results in Table 3.

4.2.1 FROC definitions

Assume the objects in the data have two levels, which we will name *patients* and *candidates* as in KDD Cup 2008 (in other applications the names may be different).

Table 3 Typical impact of post-processing on FROC comparison on

Post processing method	FROC
None	0.09
Theoretical	0.085
Heuristic	0.093

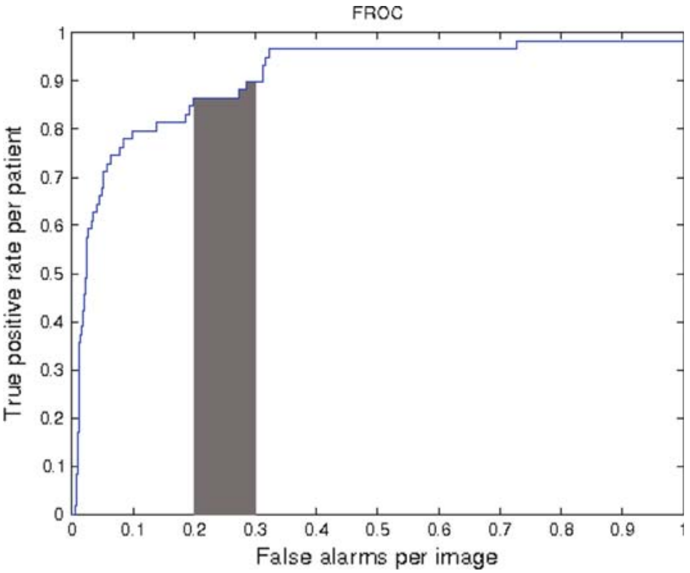

Fig. 3 An example Free Response Receiver Operating Curve (FROC)

Each patient has multiple candidates (suspected locations in mammography images), and each candidate has a label of positive (malignant) or negative (non-malignant). A patient is considered malignant if *any* of her candidates are in fact malignant. Assume we have a model which ranks the candidates according to some criterion. Then the FROC curve plots the cumulative percentage of *true positive patients* on the y-axis versus the cumulative percentage (often expressed as false alarms per image) of *false positive candidates* as we go down the ranked list. For KDD Cup Task 1 the evaluation measure was the area under this curve in the range of 0.2–0.3 false positive candidates per image, as discussed before. An example FROC curve is shown in Fig. 3, where the area of the shaded region corresponds to the evaluation measure used.

In what follows, we denote the area under the FROC curve by AUFROC, and competition evaluation criterion, i.e., the area in the 0.2–0.3 region by fFROC.

4.2.2 Optimizing the AUFROC

In this section we are going to describe a postprocessing procedure that improves AUFROC. We prove that this procedure is optimal under the assumption that the model provides us with accurate probabilities of every candidate being malignant.

Assume we have a perfect estimate for the probability p_i, that each candidate is malignant, and that malignancy of each candidate is independent of any other candidates (given their probabilities). Assume further that we have a patient, for which we have already included k candidates at the top of our list. Denote the probabilities of these candidates by $p_1, \ldots, p_k$. The probability PM_k that none of them is malignant (and therefore we have not yet identified this patient as malignant) is $\prod_{i=1}^{k}(1 - p_i)$. Given another candidate for this patient with probability of malignancy p_{k+1}, we can

add the candidate to the list, and identify a new malignant patient with probability $PM_k \times p_{k+1}$ or add another false alarm with probability $1 - p_{k+1}$. This motivates the following definition:
For a given candidate $\mathcal{C}$ let $p_1 \geq p_2 \geq \cdots \geq p_k \geq \cdots \geq p_K$ be probabilities of malignancy of all candidates belonging to the same patient, with p_k corresponding to the candidate $\mathcal{C}$ itself. Define

$$y(\mathcal{C}) = (1 - p_1) \cdot (1 - p_2) \cdot \cdots \cdot (1 - p_{k-1}) \cdot p_k \cdot \frac{1}{1 - p_k}.$$

The main results of this section are

Theorem 1 *Let* $\{\mathcal{C}_i\}_{i=1}^N$ *be a sequence of candidates ordered in such a way that for every* $i < j$ *there holds* $y(\mathcal{C}_i) \geq y(\mathcal{C}_j)$*. Then the expected value of* $\mathrm{AUFROC}(\{\mathcal{C}_i\}_{i=1}^N)$ *is maximal among all orderings of candidates.*

Theorem 2 *Let* $\{\mathcal{C}_i\}_{i=1}^N$ *be a sequence of candidates ordered in such a way that for every* $i < j$ *there holds* $y(\mathcal{C}_i) \geq y(\mathcal{C}_j)$*. Then the expected value of* $\mathrm{fFROC}(\{\mathcal{C}_\mathrm{i}\}_\mathrm{i=1}^\mathrm{N})$ *is maximal among all orderings of candidates.*

In words, if we order the candidates by the values of y, we are guaranteed to maximize expected AUFROC and fFROC. The proofs of these theorems are given in the Appendix.

The following algorithm makes the optimal policy in terms of AUFROC and fFROC explicit.

Algorithm 1 (postprocessing) Input: the sequence X of pairs $\{ID_i, p_i\}_{i=1}^N$. Output: $Y = \{y_i\}_{i=1}^N$.

```
1. Set $\zeta = 1$.
2. Sort $X$ using the ordering $\{ID_i, p_i\} \prec \{ID_j, p_j\}$ if and only if $ID_i < ID_j$ or
   $ID_i = ID_j$ and $p_i < p_j$ in descending order within a patient
3. Append $\{-1, -1\}$ at the end of $X$ (for technical reasons, it is assumed here that
   all $ID_i > 0$)
4. For $i = 1$ to $N$
   (a) Set $PM = \zeta$.
   (b) If $ID_i = ID_{i+1}$ set $\zeta = \zeta * (1 - p_i)$ else set $\zeta = 1$.
   (c) Set $y_i = PM * \frac{p_i}{1-p_i}$ for $p_i < 1$ and $y_i = 1$ for $p_i = 1$.
   Note that if $p_i = 1$ then $PM = 0$ for all $j > i$, $ID_j = ID_i$.
```

The sorting by patient is a technical trick allowing the algorithm to run in a linear time.

For the Theorems 1 and 2 to hold, and thus for the sequence y obtained using Algorithm 1 to yield better expected AUFROC (and fFROC) values than any other transformation of the value of the p_i's, the p_i's must be true probabilities of malignancy for each candidate. Clearly, this is not what our models generate. Some modeling approaches, like SVMs, do not even generate scores that can be interpreted as probabilities. In the case of SVMs, Platt correction (Platt 1998) is a common approach to alleviate this problem. We thus applied this post-processing approach to three different models:

- Logistic regression raw predictions. These are expected to be somewhat overfitted and therefore not good as probability estimates. Since the emphasis of the algorithm is on the largest p_i's we modified them simply by capping them, leading to:
- Logistic regression predictions, capped at different thresholds (e.g., 0.5)
- SVMs with Platt correction

Disappointingly, Algorithm 1 did not lead to a significant improvement in holdout fFROC on any of these models, implying that our algorithm, while theoretically attractive, has little practical value when dealing with (bad) probability estimates instead of true probabilities. We did observe that the AUFROC improved initially (below 0.05 false positives per image) but not in the relevant area of 0.2-0.3. There it actually seems to hurt our performance as shown in Table 3.

4.2.3 Heuristic AUFROC post-processing

To develop a heuristic approach, we return to the differences between the ROC and FROC curves. In our case, a good ROC curve would result from the algorithm's correctly ranking candidates according to their probabilities of being malignant. However, if many malignant candidates are identified for the same patient, this does not improve the *patient-level* true positive rate, drawn on the FROC curve Y-axis. As such, a higher true positive rate at a candidate-level does not improve FROC unless the positive candidates are from different patients. For instance, it is better to have 2 correctly identified candidates from different patients, instead of 5 correctly identified candidates from the same. So it is best to re-order candidates based on model scores so as to ensure we have many different patients up front.

In order to do this, we create a pool of the top n candidates, as ordered by our model. We then select the candidates with the highest scores for each patient in this pool, and move these to the top of our list. We repeat this process iteratively with the remaining candidates in our pool until we have exhausted all candidates.

We only do this for the top n candidates, since the fFROC metric is based only on the area under the curve for a small range of false alarm rates at the beginning of the curve. We leave the ordering of the remaining candidates untouched. The only parameter this post-processing procedure requires is the choice of n for the number of top-ranked candidates we want to re-order. The specific fFROC metric used to evaluate the KDD Cup Task 1 was the area under the FROC curve in the false alarm range of 0.2–0.3. Re-ordering scores beyond this range, has no effect on the area in this range. Furthermore, since the true positive rate per patient (i.e. the y-axis of the FROC curve) is monotonically increasing, any increase in AUFROC below the false alarm rate of 0.3 leads to an increase in the range 0.2–0.3. We select the value of n so as to minimize the number of scores that need to be reordered. So the value of n is the smallest number of candidates that must be classified as positive before we hit the upper bound of the false alarm rate used in the fFROC metric. The top n candidates can be composed of both true positives and false positives; so the smallest value of n is given by the maximum number of true and false positives within the prescribed false alarm range. True positives do no contribute to the false alarm rate, so there can be as many true positives as there are positive candidates in the test set. The maximum number of false negatives is dictated by the upper bound on the false alarm rate, i.e.,

$$\text{False alarm rate} = \frac{\text{Number of false positives}}{4 \times \text{Number of patients}} = 0.3$$

Combining the maximum number of true and false positives, we get the minimum number of the top candidate-scores to be reordered,

$$n = \text{Number of positive candidates} + 1.2 \times \text{Number of patients}$$

Since the true number of positive candidates in the test set is not known, we estimate this from the positive rate in the training set. The impact of this post-processing can be seen in Fig. 4 for a fifty-fifty train-test split of the labeled data provided in the competition. Since we were not provided with the labels on the competition test set, the actual contribution of this post-processing to our winning solution is unknown. Table 3 shows typical results we observed in our internal evaluations. It should be noted that the significant increase in fFROC via post-processing comes with no additional modeling cost, and is solely derived from a better understanding of the domain-specific performance metric.

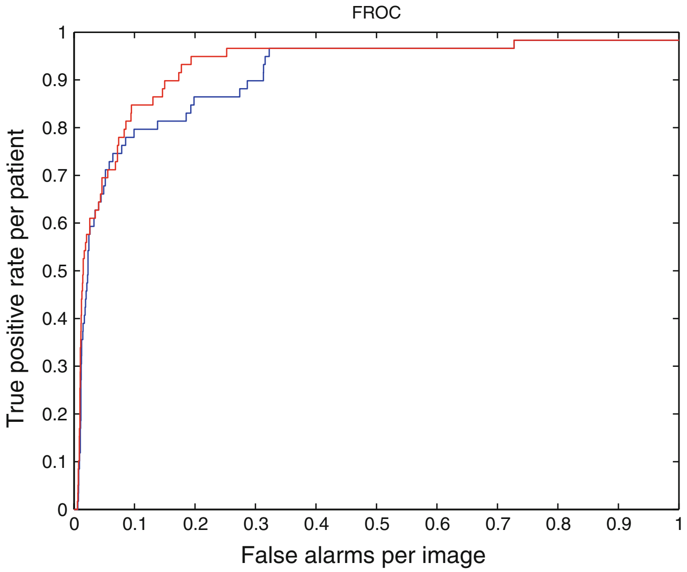

Fig. 4 Increase in FROC based on post-processing of model scores

4.2.4 Post-processing in real-life usage scenarios

The fFROC measure was designed by the KDD Cup 2008 organizers as a proxy for the situation where a radiologist with limited capacity is going to examine the most suspect *candidates*, and the goal is to support the discovery of a maximum number of malignant *patients* in the examined set. It is clear that this usage scenario is much more generally applicable than just to breast cancer detection.

Our post-processing solutions significantly improved the discovery rate, but they are contingent on seeing all scores and re-ranking them to achieve maximal effect. It is obvious that in similar real-life scenarios, the model is unlikely to have the entire ranked list to re-order before presenting it to the expert. Indeed, in real-time applications, the model would need to assign candidates for expert examination based on their scores only. However, many real-life usage scenarios correspond to an intermediate scenario, where the radiologist (or other expert) does not continually examine suspected images, but rather occasionally (say, daily) visits the testing facility and examines the list presented to her. In such a case, the list of candidates collected during the day can be reordered before presentation to the radiologist, and the benefits of post-processing can be enjoyed.

5 Relational and multi-level data

Statistics and machine learning have historically made one fundamental assumption about the data: instances are independent and identically distributed (iid) and are represented in a ‘flat’ matrix that has for each instance a vector with feature values. While certain violations of this assumption have been acknowledged and partially addressed in specific cases, medical data are very prone to extreme violations of this assumption (Saar-Tsechansky et al. 2001).

In the two competitions we faced two instances of non-iid data that are typical for medical data. The INFORMS data have the common patient-centric view that links many different pieces of information about a particular patient from different sources and times. In the case of INFORMS there are multiple records for the same patient and in addition one-to-n and m-to-n relationships into additional tables. In the case of infections we may additionally suspect relevant interactions between patients if they were, for instance, members of the same household as provided in the demographics table, or hospitalized at the same time.

The KDD Cup has a more intrinsic form of non-iid data. Fundamentally, we want to answer the question if a patient has breast cancer or not. So the natural unit of analysis would be one single breast. However, the images are only dual 2D projections and the pre-processing of the images has to trade off the immense cost of false negatives and is therefore very conservative. As a result, many discrete candidates are identified even though they may be overlapping and pointing to the same suspicious region.

In the case of KDD Cup the organizers already pointed out that it might be possible to take advantage of the fact that two different candidates may very well be indicators of the same underlying lesion in the breast tissue and should therefore have similar labels. In addition to the biological linkage, there might be another re-enforcing human labeling bias. Once a candidate is tested positive in a biopsy, it seems likely that the examiner will label all corresponding candidates as positive.

While there has been substantial work in relational machine learning during the last decade, there are as of yet no verified standard cases or scenarios. In addition, most of the higher-level learning approaches (Domingos and Richardson 2007; Muggleton and DeRaedt 1994; Getoor et al. 2007) have not yet demonstrated to scale successfully to large domains. In particular, the winning approach in the relational learning challenge (ILP challenge 2005 (Perlich 2005)) on a large genetic domain was based on our feature construction algorithm ACORA (Perlich and Provost 2006).

Accordingly, we will provide a brief overview of our relational learning method ACORA that was applied to the INFORMS data where it provides substantial improvements. We also offer a conceptual discussion of possible modeling approaches for the relationships between candidates in the KDD Cup.

While we do not observe consistent performance improvements for all of them, we consider it still of interest and relevance to show the many different ways the dependencies can be modeled. The creative exploration of multiple different avenues to address a particular property of the application domain is conceptually similar to an open minded exploratory data analysis. While the performance is not necessarily improving, there are a number of potential reasons for this and understanding them can be of interest by itself. The initial question is if relational information is predictive. While this is the case in most domains we have encountered, lack of predictability can help to reject suspected dependencies and can provide valuable insight about the domain. However, the more commonly relevant question is, if it is predictive in addition to the already available propositional information. In the case of the KDD Cup, the answer to this last question seems to be no, while in the INFORMS contest the relational information is predictive in addition to the hospital information as shown in Table 2.

5.1 Neighborhood dependence in KDD Cup

We explored three fairly different but potentially equally valid approaches to utilize the suspected neighborhood relationship and will discuss them in more depth below.

1. Two-stage framework where we first ‘predict’ the labels of candidates, then use the labels of close neighbors as features;
2. Penalty on vastly different prediction values for close neighbors;
3. Feature construction from the neighboring candidates.

5.1.1 Two stage completion approach

We can frame the objective of incorporating the notion of a dependence of the candidate’s score on the score of its neighbors as a learning task with latent variables: the scores of the neighbors. This would suggest an iterative algorithm that would keep refining a model that feeds the scores back to populate features of neighboring candidates.

We explored this approach starting for the first stage with a basic ‘flat’ model based on the provided 117 numeric features $x_k = (x_{1,k}, \ldots, x_{117,k})$. With a *leave one patient out* approach, we calculated out of sample scores for each of the 1,712 patients (that is, we actually built 1,712 different regression models, each time leaving one patient’s candidates out of the model estimation, and then calculating their scores).

$$s_{k,Stage1} = f(x_{1,k}, \ldots, x_{117,k}) \tag{1}$$

The scores from this model are used to generate ‘neighborhood features’ in the second stage. There are a number of choices to define a set of such features. In essence, each of the neighborhood features is a function of the predicted scores and the distance between the candidate and the neighbors n. One such neighborhood feature $\bar{x}_{118,k}$ could be a distance weighted average of the scores using some kernel to translate Euclidean distances into weights. More generally, the second stage can incorporates a number of derived features:

$$s_{k,Stage2} = f(x_{1,k}, \ldots, x_{117,k}, \bar{x}_{118,k}, \ldots, \bar{x}_{p,k}) \tag{2}$$

Several encouraging results were observed from logistic regression models that used for stage 2 the score of the closest neighbor and its distance.

1. The logistic regression identified the neighbor’s score and the distance as the two most important features, with the expected signs (positive for neighbor score, negative for distance).
2. The two stage model was clearly better able to differentiate malignant candidates from benign ones.

However, disappointingly, it did not do better in the fFROC metric, and specifically failed to identify more *malignant patients* than the first-stage model. Thus, despite being a clearly more powerful model for malignancy detection, it did not manage to improve the ‘real world’ performance of our models. However, we still consider that it proved to be a useful conceptual approach to utilizing neighborhood information. This algorithm can be related to the work on ‘stacking’ for graphical models, which is a statistical learning model for collective inference over relational data (Wolpert 1992; Kou and Cohen 2007). However, our algorithm differs from stacking in that it does not assume an explicit graph structure between examples and the information propagation process is simpler.

5.1.2 Pairwise constrained kernel logistic regression

The data contain the coordinates of each candidate in a given image. We can define a match as a pair of candidates (x_k, x_m) with Euclidean distance less than a threshold t. The threshold could either be based on the number of pairs n or be derived from the underlying geometry such that each candidate would have the same number of matches. This leaves us with a set C of Pairs:

$$C = \{(x_k, x_m) \mid ||x_k - x_m|| < t\} \tag{3}$$

We would like candidates belonging to a pair (x_k, x_m) to have similar predicted labels, i.e. $f(x_k) \sim f(x_m)$. We shall incorporate this condition using pairwise kernel logistic regression, which is able to plug in additional pairwise constraints together with labeled data to model the decision boundary directly (Yan et al. 2004).

Suppose we have a set of training examples $\{(x_i, y_i)\}$, and our set C of pairs. To make the optimization problem feasible to solve, we define a convex loss function via the logit loss as follows:

$$\begin{aligned}\mathcal{O}(f) &= \frac{1}{N}\sum_{i=1}^{N}\log\left(1+e^{-y_i f(x_i)}\right) + \lambda\Omega(\|f\|_{\mathcal{H}}) \\ &\quad + \frac{\mu}{n}\sum_{(x_k,x_m)\in\mathcal{C}}\log\left(1+e^{f(x_k)-f(x_m)}\right) + \log\left(1+e^{f(x_m)-f(x_k)}\right),\end{aligned}$$

where the first term is the loss on labeled training examples, the second is the regularizer and third term is the loss associated with the difference between the predicted labels of the example pairs. The pairwise constraint coefficient μ is set to 1. For simplicity, we define f as a linear classifier, i.e. $f(x) = w^T x$. Since the optimization function is convex, a gradient search algorithm can guarantee the finding of the global optimum. It is easy to derive the parameter estimation method using the interior-reflective Newton method, and we omit the detailed discussion. The constrained logistic regression unfortunately did not improve the fFROC over the unconstrained baseline.

5.1.3 Feature construction

Whereas the first two approaches tried to add information about the response of neighbors feature set, we now instead include the features of the neighbors directly. This approach lacks the elegance of the two-stage framework or the penalty setup, but is closer in line with some standard relational learning methods that we used very successfully on the INFORMS domain (see below).

Similarly to the penalty setting we define for each candidate a set of neighbors based on their Euclidean distance within an image. We now add another 117 features to the original feature vector that contains the mean feature value of the neighbors.

Similarly to our previous results we observe that this methodology does not improve the fFROC performance substantially. We suspect that in the feature construction case we ultimately have too few datapoints to support 234 features and the models are subject to significant estimation error.

5.2 Patient-centric data in INFORMS

While the relational feature construction did not improve the results in the KDD Cup, we did observe a substantial improvement on the INFORMS competition in Table 2. The automated process of feature construction in relational datasets with one-to-many links is formally known as propositionalization (Krogel and Wrobel 2003).

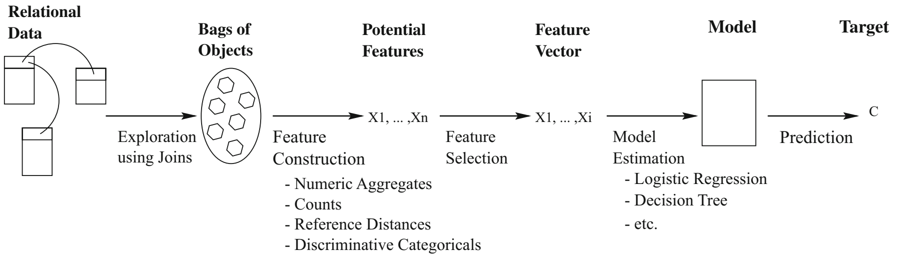

Fig. 5 ACORA's transformation process with four transformation steps: exploration, feature construction, feature selection, model estimation, and prediction. The first two (exploration and feature construction) transform the originally relational task (multiple tables with one-to-many relationships) into a corresponding propositional task (feature-vector representation)

The main challenge arises from one-to-many link to tables with high-dimensional categorical values. Contrary to the numeric features of the neighbors in the KDD Cup example we now have to aggregate sets of medications or conditions belonging to a patient.

ACORA is a learning system that automatically converts a relational domain into a flat feature-vector representation using aggregation to construct attributes given the database schema as shown in Fig. 1. ACORA consists of four nearly independent modules, as shown in Fig. 5:

- *Exploration* constructing bags of related entities using joins and breadth-first search based on the schema of the domain and identifiers that link the tables.
- *Aggregation* transforming bags of objects into single-valued features by aggregating one feature at a time (assuming independence) using a variety of aggregation operators including mean, min and max for numerical features and class-conditional vector distances for categorical features.
- *Feature selection* based on the AUC on the prediction task of a single feature.
- *Model estimation* using logistic regression or decision trees in combination with bagging.

The aggregation operator for categorical features mirrors a 'local' naive Bayes model and estimates vector distances to the class-conditional distributions of the relevant features. For more details see Perlich and Provost (2006).

The AUC increased from 0.81 for baseline model (using only the features in the hospital table) to 0.90 once ACORA included the information from all tables including class-conditional aggregates of the medication names and of the medical conditions as well as demographic information.

We observed an interesting interaction effect between leakage and relational learning. The type 1 leakage was related to the number of provided conditions. This number would also be one of the relational features that ACORA calcuates. By adding this piece of information, the logistic model could already pick up the leakage effect even if the model builder had not observed this effect.

6 Conclusion

In this paper we present three fundamental problems in medical data mining, as exemplified by their common appearances in both competitions we discussed. We use the instances of these problems in the competitions as motivating and running examples, to demonstrate the importance of these issues and how handling them helped us develop appropriate solutions for the competition. Our discussion combines high-level insights and guidelines, with specific detailed examples. We hope that the insights and results we offer will be useful for the larger medical data mining community, which encounters similar problems on a regular basis.

Notice that although we raise and discuss these issues in the context of medical data mining, and more specifically the competitions, it is clear that all three apply to practical data mining tasks in other domains as well. Different domains, however, are likely to emphasize different aspects and different flavors of these problems.

Appendix: proof of theorems 1, 2

Notation:

- $\mathcal{P}(\mathcal{C})$ denotes a patient to whom candidate $\mathcal{C}$ belongs,
- $\mathrm{ID}(\mathcal{P})$ is a patient ID of patient $\mathcal{P}$, and $\mathrm{ID}(\mathcal{C}) = \mathrm{ID}(\mathcal{P}(\mathcal{C}))$,
- $\wp(\mathcal{C})$ is a probability that candidate $\mathcal{C}$ is malignant, returned by the model,
- $\mathcal{C}(\mathcal{P})$ denotes a (finite) sequence $\{\mathcal{C}_1^i, \mathcal{C}_2^i, \ldots, \mathcal{C}_K^i\}$ of all candidates belonging to a patient $\mathcal{P}$, where $i = \mathrm{ID}(\mathcal{P})$, and for every $1 \le m < n \le K$ there holds $p_m^i \ge p_n^i$, where $p_m^i = \wp(\mathcal{C}_m^i)$ and $p_n^i = \wp(\mathcal{C}_n^i)$.

Before we prove Theorem 1 and Theorem 2 we are going to need a few technical results.

Lemma 1 *If for some patient $\mathcal{P}$ candidates $\mathcal{C}, \mathcal{C}' \in \mathcal{C}(\mathcal{P})$ and there holds $p_k \ge p_m$ for $p_k = \wp(\mathcal{C})$, $p_m = \wp(\mathcal{C}')$, then $y(\mathcal{C}) \ge y(\mathcal{C}')$.*

Proof It is sufficient to show that

$$(1-p_1)\cdot\dots\cdot(1-p_k)\cdot p_{k+1}\cdot\frac{1}{1-p_{k+1}} \le (1-p_1)\cdot\dots\cdot(1-p_{k-1})\cdot p_k\cdot\frac{1}{1-p_k}.$$

This follows immediately from $\frac{(1-p_k)^2}{p_k}\cdot\frac{p_{k+1}}{1-p_{k+1}} \le \frac{(1-p_k)^2}{p_k}\cdot\frac{p_k}{1-p_k} \le 1-p_k \le 1$ as $p_k \ge p_{k+1}$ and $\frac{x}{1-x}$ is an increasing function in $[0, 1)$.
Note that the inequality is not sharp only in case $p_k = 0$. □

Given an ordering of candidates $\{\mathcal{C}_i\}_{i=1}^N$ let

- $V_i = V(\mathcal{C}_i)$ equal 1 if candidate $\mathcal{C}_i$ is malignant and 0 if candidate $\mathcal{C}_i$ is benign,
- $\mathrm{FPC}(i) = i - \sum_{k=1}^{i} V_k$ be the number of false positives among first i candidates,
- $\mathrm{TPP}(i) = |\{\mathrm{ID}_k : k \le i, V_k = 1\}|$ be the number of true positive patients (patients with at least one malignant candidate at any of their 4 images) among first i candidates,

– nImages = $4 * |\{\mathrm{ID}_k : k \leq N\}|$ of all images (4 times the number of all patients).

Then

$$\mathrm{FAUC}(\{\mathcal{C}_i\}_{i=1}^N) = \frac{1}{\mathrm{TPP}}(N)\frac{1}{\mathrm{nImages}}\sum_{k=1}^{N}\mathrm{TPP}(k)\cdot(\mathrm{FPC}(k)-\mathrm{FPC}(k-1)) \quad (4)$$

(note that by definition $\mathrm{FPC}(0) = 0$ and that TPP(N) is simply the number of all malignant patients).

Given an ordering of candidates $\{\mathcal{C}_i\}_{i=1}^N$ each candidate $\mathcal{C}_k$ falls into one of the three classes:

I a false positive,
II a true positive for a patient which has already been identified as malignant,
III a true positive for a patient which has not yet been identified as malignant.

For simplification we define

$$T\left(\{\mathcal{C}_i\}_{i=1}^N, k\right) = \begin{cases} \text{I if } \mathcal{C}_\mathrm{k} \in \text{I} \\ \text{II if } \mathcal{C}_\mathrm{k} \in \text{II} \\ \text{III if } \mathcal{C}_\mathrm{k} \in \text{III} \end{cases}.$$

Whenever it does not lead to misunderstanding we will write $T(k)$ instead of $T\left(\{\mathcal{C}_i\}_{i=1}^N, k\right)$. Let $\{\mathcal{C}'_i\}_{i=1}^N(k)$ be another ordering of candidates with $\mathcal{C}_k$ and $\mathcal{C}_m$ swapped, $k < m$. We denote $\mathrm{FAUC} = \mathrm{AUFROC}(\{\mathcal{C}_i\}_{i=1}^N)$ and $\mathrm{AUFROC}' = \mathrm{AUFROC}(\{\mathcal{C}'_i\}_{i=1}^N)$.

Lemma 2 *The difference* $\Delta = \Delta_{k,m} = \mathrm{AUFROC}' - \mathrm{AUFROC}$ *depends only on* $T\left(\{\mathcal{C}_i\}_{i=1}^N, k\right)$ *and* $T\left(\{\mathcal{C}_i\}_{i=1}^N, m\right)$. *Moreover*

$$\begin{aligned} \Delta(I, I) = \Delta(II, II) = \Delta(III, III) &= 0, \\ \Delta(I, II) = -\Delta(II, I) &= 0, \\ \Delta(I, III) = -\Delta(III, I) &= \frac{1}{\mathrm{TPP(N)}}\frac{1}{\mathrm{nImages}}(FPC(m) - FPC(k)), \\ \Delta(II, III) = -\Delta(III, II) &= 0, \end{aligned}$$

where the first argument of $\Delta(\cdot,\cdot)$ *is* $T(k)$ *and the second one is* $T(m)$.

Proof Straightforward verification using (4). □

Corollary 1 *Swapping* $\mathcal{C}_k$ *and* $\mathcal{C}_{k+1}$ *leads to an increment of* AUFROC *by* $\frac{1}{\mathrm{TPP}(N)}\frac{1}{\mathrm{nImages}}$ *if* $T(k) = \mathrm{I}$ *and* $T(k) = \mathrm{III}$, *decrement of* AUFROC *by* $\frac{1}{\mathrm{TPP}(N)}\frac{1}{\mathrm{nImages}}$ *if* $T(k) = \mathrm{III}$ *and* $T(k) = \mathrm{I}$ *and has no influence on* AUFROC *in any other case.*

Corollary 2 *Expected values of* AUFROC *and* AUFROC′ *satisfy*

$$\mathbb{E}\mathrm{AUFROC}' = \mathbb{E}\mathrm{AUFROC} + \frac{1}{\mathrm{TPP(N)}}\frac{1}{\mathrm{nImages}} \cdot P(k, m),$$

where $P(k,m) = \wp\left((T(k)=I) \cap (T(m)=III)\right) - \wp\left((T(k)=III) \cap (T(m)=I)\right)$.

Proposition 1 *Let for some ordering* $\{\mathcal{C}_i\}_{i=1}^N$ *candidates* $\mathcal{C}_k$, $\mathcal{C}_m$, $k < m$ *belong to the same patient* $\mathcal{P}^i$. *Let moreover for every* $k < s < m$ $\mathcal{C}_s \notin \mathcal{C}(\mathcal{P}^i)$. *Let* $\{\mathcal{C}'_i\}_{i=1}^N$ *be the ordering of candidates with* $\mathcal{C}_k$, $\mathcal{C}_m$ *reversed. If* $y_k \leq y_m$ *then* $\mathbb{E}\mathrm{AUFROC}' \geq \mathbb{E}\mathrm{AUFROC}$.

Proof By Lemma 1 inequality $y_k \leq y_m$ implies $p_k \leq p_m$.
Then $\wp\left((T(k) = I) \cap (T(m) = III)\right) = \theta \cdot (1 - p_k) \cdot p_m$, $\wp\left((T(k) = III) \cap (T(m) = I)\right) = \theta \cdot p_k \cdot (1 - p_m)$, where $\theta = \prod_{j<k, \mathcal{C}_j \in \mathcal{C}(\mathcal{P}^i)} \left(1 - \wp(\mathcal{C}_j)\right)$. Thus $P(k,m) = \theta \cdot (1 - p_k) \cdot p_m - \theta \cdot p_k \cdot (1 - p_m) = \theta \cdot (p_m - p_k) \geq 0$ and proposition follows from Corollary 2. □

Proposition 2 *Let an ordering* $\{\mathcal{C}_i\}_{i=1}^N$ *satisfy the following: for every two candidates* $\mathcal{C}_s$, $\mathcal{C}_t$, $s < t$ *who belong to the same patient* $\mathcal{P}^i$ *there holds* $y_s \geq y_t$.
Under this assumption let candidates $\mathcal{C}_k$, $\mathcal{C}_{k+1}$ *belong to two different patients and* $\{\mathcal{C}'_i\}_{i=1}^N$ *be the ordering of candidates with* $\mathcal{C}_k$, $\mathcal{C}_{k+1}$ *reversed. If* $y_k \leq y_{k+1}$ *then* $\mathbb{E}\mathrm{AUFROC}' \geq \mathbb{E}\mathrm{AUFROC}$.

Proof Because $\mathcal{C}_k, \mathcal{C}_{k+1}$ belong to two different patients, whether either of them is of type I, II or III is independent of the type of the other one. Thus thanks to the assumption that probabilities of malignancies of all candidates are independent of each other we get $\wp\left((T(k) = I) \cap (T(k+1) = III)\right) = \wp(T(k) = I) \cdot \wp(T(k+1) = III)$ and $\wp\left((T(k) = III) \cap (T(k+1) = I)\right) = \wp(T(k) = III) \cdot \wp(T(k+1) = I)$.
Let $\mathcal{C}_k \in \mathcal{C}(\mathcal{P}^K)$, $\mathcal{C}_{k+1} \in \mathcal{C}(\mathcal{P}^M)$. Then $\wp(\mathcal{C}_k) = p_i^K$, $\wp(\mathcal{C}_{k+1}) = p_j^M$ for some i, j. We have an explicit formula for weights $y_k = y(i, K)$ and $y_{k+1} = y(j, M)$

$$y(i, K) = (1 - p_1^K) \cdot (1 - p_2^K) \cdot \dots \cdot (1 - p_{i-1}^K) \cdot p_i^K \cdot \frac{1}{1 - p_i^K} \tag{5}$$

and

$$y(j, M) = (1 - p_1^M) \cdot (1 - p_2^M) \cdot \dots \cdot (1 - p_{j-1}^M) \cdot p_j^M \cdot \frac{1}{1 - p_j^M}. \tag{6}$$

The probability of candidate $\mathcal{C}_k$ to be of type I is equal to $1 - p_i^K$. The probability of candidate $\mathcal{C}_k$ to be of type III is $(1 - p_1^K) \cdot (1 - p_2^K) \cdot \dots \cdot (1 - p_{i-1}^K) \cdot p_i^K$ and analogous formula holds for $\mathcal{C}_{k+1}$. Therefore

$$\begin{aligned} P(k, k+1) = (1 - p_1^K) \cdot \dots \cdot (1 - p_{i-1}^K) \cdot p_i^K \cdot (1 - p_j^M) \\ -(1 - p_1^M) \cdot \dots \cdot (1 - p_{j-1}^M) \cdot p_j^M \cdot (1 - p_i^K). \end{aligned}$$

Thus $P(k, k+1) \geq 0$ if and only if $y(i, K) > y(j, M)$ and proposition follows from Corollary 2. □

Proof (of Theorem 1) There exists a finite number of orderings $\{\mathcal{C}_i\}_{i=1}^N$, thus there exists an ordering $\{\bar{\mathcal{C}}_i\}_{i=1}^N$ for which the value of $\mathbb{E}$AUFROC is maximal. By Proposition 1 for each patient all candidates belonging to $\{\bar{\mathcal{C}}_i\}_{i=1}^N$ must be ordered according to their weights y. In the opposite case we could find two candidates such that swapping their order would increase expected value of $\mathbb{E}\text{AUFROC}\left(\{\bar{\mathcal{C}}_i\}_{i=1}^N\right)$ leading to a contradiction. Therefore $\{\bar{\mathcal{C}}_i\}_{i=1}^N$ satisfies the assumptions of Proposition 2 and it follows that all candidates in $\{\bar{\mathcal{C}}_i\}_{i=1}^N$ must be ordered according to their y's.
So far we have proven that having $y_i \geq y_j$ for every $i < j$ is a necessary condition for an ordering to yield maximum value of $\mathbb{E}$AUFROC. But—up to reordering candidates having equal values of y—there exists unique ordering $\{\mathcal{C}_i\}_{i=1}^N$ satisfying $y_i \geq y_j$ for every $i < j$. Thus, because a (global) maximum does exist, it is also a sufficient condition and the theorem follows.

Theorem 2 is proven by exactly the same methodology. The difference is that the analogue of Lemma 2 now contains more cases to consider depending on whether and how the region under FROC affected by the swap of candidates overlaps with the area selected as relevant in fFROC definition. We decided to leave the details to the reader instead of presenting here the tedious rigorous argumentation.

Acknowledgments We thank the organizers of both competitions, whose efforts made possible the enjoyable and instructive experiences we discuss here. Saharon Rosset's research is partially supported by EU grant MIRG-CT-2007-208019.

References

Bandos AI, Rockette HE, Song T, Gur D (2008) Area under the free-response ROC curve (FROC) and a related summary index. Biometrics 65(1):247–256

DeLuca PM, Wambersie A, Whitmore GF (2008) Extensions to conventional ROC methodology: LROC, FROC, and AFROC. J ICRU 8:31–35

Domingos P, Richardson M (2007) Markov logic: a unifying framework for statistical relational learning. In: Getoor L, Taskar B (eds) Introduction to statistical relational learning. MIT Press, Cambridge

Ferri C, Flach P, Hernandez-Orallo J (2002) Learning decision trees using the area under the ROC curve. In: Proceedings of the international conference on machine learning

Getoor L, Friedman N, Koller D, Pfeffer A, Taskar B (2007) Probabilistic relational models. In: Getoor L, Taskar B (eds) Introduction to statistical relational learning. MIT Press, Cambridge

Glymour C, Scheines R, Spirtes P, Kelly K (1987) Discovering causal structure: artificial intelligence, philosophy of science, and statistical modeling. Academic Press, San Diego

Inger A, Vatnik N, Rosset S, Neumann E (2000) KDD-Cup 2000: question 1 winner's report, SIGKDD explorations

Joachims T (2005) A support vector method for multivariate performance measures. In: Proceedings of the international conference on machine learning

Joachims T (1999) Making large-scale SVM learning practical. In: Scholkopf B, Burges C, Smola A (eds) Advances in Kernel methods—support vector learning. MIT Press, Cambridge

Kou Z, Cohen WW (2007) Stacked graphical learning for efficient inference in markov random fields. In: Proceedings of the international conference on data mining

Krogel M-A, Wrobel S (2003) Facets of aggregation approaches to propositionalization. In: Proceedings of the international conference on inductive logic programming

Lawrence R, Perlich C, Rosset S et al (2007) Analytics-driven solutions for customer targeting and sales-force allocation. IBM Syst J 46(4):797–816

Melville P, Rosset S, Lawrence R (2008) Customer targeting models using actively-selected web content. In: Proceedings of the conference on knowledge discovery and data mining

Muggleton SH, DeRaedt L (1994) Inductive logic programming: theory and methods. J Logic Program 19 & 20:629–680

NIST/SEMATECH (2006) e-Handbook of Statistical Methods, chap. 1. http://www.itl.nist.gov/div898/handbook/eda/eda.htm

Perlich C (2005) Approaching the ILP challenge 2005: class-conditional bayesian propositionalization for genetic classification. In: Proceedings of the conference on inductive logic programming

Perlich C, Provost F (2006) ACORA: distribution-based aggregation for relational learning from identifier attributes, special issue on statistical relational learning and multi-relational data mining. J Mach Learn 62:65–105

Perlich C, Melville P, Liu Y, Swirszcz G, Lawrence R, Rosset S (2008) Breast cancer identification: KDD cup winner's report, SIGKDD explorations

Platt J (1998) Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: Bartlett PJ, Schölkopf B, Schuurmans D, Smola AJ (eds) Advances in large margin classifiers. MIT Press, Cambridge

Rao RB, Yakhnenko O, Krishnapuram B (2008) KDD Cup 2008 and the workshop on mining medical data, SIGKDD explorations

Rosset S, Perlich C, Liu Y (2007) Making the most of your data: KDD Cup 2007 "How many ratings" winner's report, SIGKDD Explorations

Russ TA (1989) Using hindsight in medical decision making. In: Proceedings of the thirteenth annual symposium on computer applications in medical care

Saar-Tsechansky M, Pliskin N, Rabinowitz G, Porath A (2001) Monitoring quality of care with relational patterns. Top Health Inf Manag 22(1):24–35

Shahar Y (2000) Dimension of time in illness: an objective view. Ann Intern Med 132:45–53

Simon HA (1954) Spurious correlation: a causal interpretation. J Am Stat Assoc 49:467–479

Turney PD (2000) Types of cost in inductive concept learning In: Proceedings of the workshop on cost-sensitive learning at the international conference on machine learning

Valentini G, Dietterich TG (2003) Low bias bagged support vector machines. In: International conference on machine learning

Weiss GM, Saar-Tsechansky M, Zadrozny B (2008) Special issue on utility-based data mining (editors). Data Min Knowl Discov 17(2)

White K, Dufresne RL (1997) The placebo effect in drug trials and the double blind. In: Hertzman M, Feltner DE (eds) The handbook of psychopharmacology trials. NYU Press, New York pp 123–136

Wolpert DH (1992) Stacked generalization. Neural Networks 5:241–259

Yan R, Zhang J, Yang J, Hauptmann A (2004) A discriminative learning framework with pairwise constraints for video object classification. In: Proceedings of IEEE conference on computer vision and pattern recognition