Noname manuscript No.
(will be inserted by the editor)

Outlier Detection for Object-Relational Data Based on Graphical Models

Fatemeh Riahi · Oliver Schulte

Received: date / Accepted: date

Abstract This paper extends unsupervised statistical outlier detection to the case of object-relational data. Object-relational data represent a complex heterogeneous network, which comprises objects of different types, links among these objects, also of different types, and attributes of these links. This special structure prohibits a direct vectorial data representation. We apply state-of-the-art probabilistic modeling techniques for object-relational data that construct a graphical model (Bayesian network), which compactly represents probabilistic associations in the data. We propose a new metric, based on the learned object-relational model, that quantifies the extent to which the individual association pattern of a potential outlier deviates from that of the whole population. The metric is based on *the likelihood ratio* of two parameter vectors: One that represents the population associations, and another that represents the individual associations. Our method is validated on synthetic datasets and on real-world data sets about soccer and hockey matches, IMDb movies and mutagenic compounds. Compared to baseline methods, our novel transformed likelihood ratio achieved the best detection accuracy.

The likelihood-based metric is then used to predict the value of the individuals and rank them. An empirical evaluation on soccer and movie data shows a strong correlation between the our outlier metric and success metrics: Individuals that our metric marks out as unusual tend to have unusual success.

Keywords Outlier Detection · Statistical-Relational Learning · Bayesian network · Likelihood Ratio

This work was supported by a Discovery Grant from the Natural Sciences and Engineering Research Council of Canada.

Simon Fraser University
E-mail: sriahi@sfu.ca ; oschulte@cs.sfu.ca

1 Introduction

Outlier detection is an important data analysis task in many domains. Statistical approaches to unsupervised outlier detection are based on a generative model of the data [2]. The generative model represents normal behavior. An individual object is deemed an outlier if the model assigns sufficiently low likelihood to generating it. We propose a new method for extending statistical outlier detection to the case of object-relational data using a novel likelihood-ratio comparison for probabilistic models.

The object-relational data model is one of the main data models for structured data [27]. The main characteristics of objects that we utilize in this paper are the following.

Object Identity. Each object has a unique identifier that is the same across contexts. For example, a player has a name that identifies him in different matches.

Class Membership. An object is an instance of a class, which is a collection of similar objects. Objects in the same class share a set of attributes. For example, van Persie is a player object that belongs to the class striker, which is a subclass of the class player.

Object Relationships. Objects are linked to other objects. Both objects and their links have attributes.

A common type of object relationship is a component relationship between a complex object and its parts. For example, a match links two teams, and each team comprises a set of players for that match. A difference between relational and vectorial data is therefore that an individual object is characterized not only by a list of attributes, but also by its links and by attributes of the objects linked to it. We refer to the substructure comprising this information as the *object data*.

Approach A class-model Bayesian network (BN) structure is learned with data for the entire population. The nodes in the BN represent attributes for links, of multiple types, and attributes of objects, also of multiple types. To learn the BN model, we apply techniques from statistical-relational learning, a recent field that combines AI and machine learning [17,45,11]. The BN provides dimensionality reduction, in the sense that it leverages independencies to represent the data distribution with exponentially fewer parameters than a non-factorized parametrization. Given a set of parameter values and an input database, it is possible to compute a *class model likelihood* that quantifies how well the BN fits the object data. The class model likelihood uses BN parameter values *estimated from the entire class data.* This is a relational extension of the standard log-likelihood method for i.i.d. vectorial data, which uses the likelihood of a data point as its outlier score. While the class model likelihood is a good baseline score, it can be improved by comparing it to *the object model likelihood*, which uses BN parameter values *estimated from the object data.* The *model log-likelihood ratio* (LR) is the log-ratio of the object model likelihood

to the class model likelihood. This ratio quantifies how the probabilistic associations that hold in the general population deviate from the associations in the object data substructure. While the likelihood ratio discriminates relational outliers better than the class model likelihood alone, it can be improved further by applying two transformations: (1) a mutual information decomposition, and (2) replacing log-likelihood differences by log-likelihood distances. We refer to the resulting novel score as the *log-likelihood distance*.

Evaluation Our code and datasets are available on-line at [39]. We validate the log-likelihood distance as an outlier metric in three ways.

Detection Accuracy. Our performance evaluation follows the design of previous outlier detection studies [15, 2], where the methods are scored against a test set of known outliers. We use three synthetic and four real-world datasets, from the UK Premier Soccer League, the Internet Movie Database (IMDb), the National Hockey League, and Mutagenesis. On the synthetic data we have known ground truth. For the real-world datasets, we use a one-class design, where one object class is designated as normal and objects from outside the class are the outliers. For example, we compare goalies as outliers against the class of strikers as normal objects. On all datasets, the log-likelihood distance metric achieves the best detection accuracy compared to the baseline methods.

Case Studies. We also offer case studies where we assess whether individuals that our score ranks as highly unusual in their class are indeed unusual. The case studies illustrate that our outlier score is *easy to interpret*, because the Bayesian network provides a platform that makes the detected outliers very easy to interpret. Interpretability is very important for users of an outlier detection method as there is often no ground truth to evaluate outliers suggested by the method.

Correlation with Success. We compare the log-likelihood distance to metrics of success for a given domain. Success rankings are one of the most interesting features to users. Our reasoning is that high success is an independent metric that indicates an unusual individual. So a correlation between log-likelihood distance and success is an independent validation of the log-likelihood distance, and also shows that it points to meaningful and interesting outliers.

Contributions Our main contributions may be summarized as follows.

1. The first approach to outlier detection for structured data that is based on a probabilistic model.
2. A new model-based outlier score based on a novel model likelihood comparison, the log-likelihood distance.

Previous Conference Publication. A preliminary version of this paper appeared in the Proceedings of the IEEE SSCI series [40] (Best Student Paper). The new material in this submission comprises the following:

1. We added new experiments on the real-world datasets.

2. We performed a comparison between our proposed metric and a well-known relational-based distance learning method (*RIBL*).
3. We introduced an application of the proposed metrics in detecting outstanding individuals and ranking them.

Paper Organization We review background about Bayesian networks for relational data. Then we introduce our novel log-likelihood distance outlier score. After presenting the details of our approach, we review related work. Empirical evaluation compares model-based and aggregation-based approaches to relational outlier detection, with respect to three synthetic and four real-world datasets.

2 Related Work

Outlier detection is a densely researched field, for a survey please see [2,4]. Figure 1 provides a tree picture of where our method is situated with respect to other outlier detection methods and other data models. Our method falls in the category of *unsupervised* statistical model-based approaches. To our knowledge, ours is the first model-based method tailored for object-relational data. Like other model-based approaches, it detects *global outliers.* Aggarwal [2] defines a global outlier to be a data point that notably deviates from the rest of the population. We review relevant approaches from different data models, the most common atomic object model—where data is represented by vectors—and structured data models.

a) Attribute Vector Data Model: By far most work on outlier detection considers atomic objects with flat feature vectors. This leads to an impedance mismatch: The required input format for these outlier detection methods is a single data matrix, not a structured dataset. For example, one cannot provide a relational database as input. This mismatch is not simply a question of choosing a file format, but instead reflects a different underlying data model: complex objects with both attributes and component objects vs. atomic objects with attributes only. It is possible to "flatten" structured data by converting it to unstructured feature vectors, for instance by using aggregate functions. We evaluated the aggregation approach in this paper by applying three standard methods for outlier detection.

Work on atomic contextual outliers [50] is like ours in that it considers the distinctness of a target individual from a reference class. A reference class is not specified for each object, but is constructed as part of outlier detection. Our work could be combined with a class discovery approach by providing a score of how informative the inferred classes are.

b) Structured Data Models: We discuss related techniques in three types of structured data models: SQL (relational), XML (hierarchical), and OLAP (multi-dimensional).

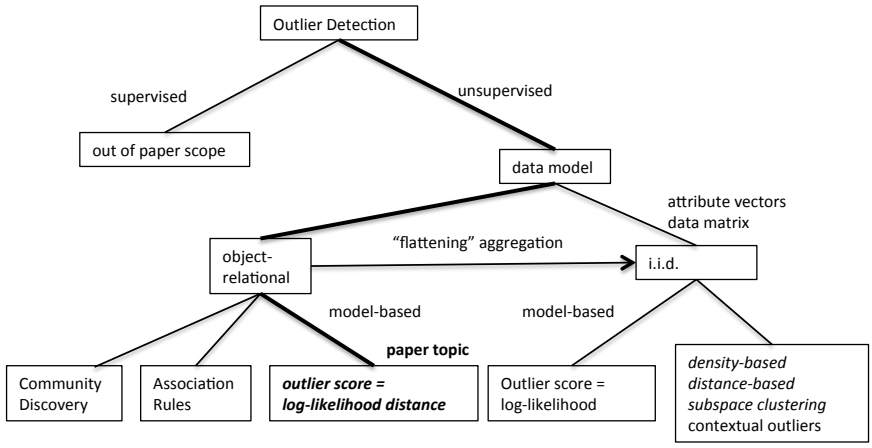


Fig. 1 A tree structure for related work on outlier detection for structured data. A path specifies an outlier detection problem, the leaves list major approaches to the problem. Approaches in italics appear in experiments.

For relational data, many outlier detection approaches aim to discover rules that represent the presence of anomalous associations for an individual or the absence of normal associations [30, 15]. In these approaches what we call the object data is usually referred to as an interpretation, so the problem is to score outlier interpretations [30]. The survey by [33] unifies within a general rule search framework related tasks such as exception mining, which looks for associations that characterize unusual cases, subgroup mining, which looks for associations characterizing important subgroups, and contrast space mining, which looks for differences between classes. Another rule-based approach uses Inductive Logic Programming techniques [6]. While local rules are informative, they are not based on a global statistical model and do not provide a single outlier score for each individual. As we show in our case studies (Section 8.3), the conditional probability parameters in a Bayesian network can be used to extract rules from the BN model (of the form $parent\_values \rightarrow child\_value$). The Distance-based algorithms designed for outlier detection can be applied to the relational case. Horvath *et al.* introduced a similarity measured for the first-order instance-based learner RIBL that uses the edit distance to compute the distance between lists and terms [21]. This distance has been used for the clustering task in [25]. To the best of our knowledge, distance-based relational learning has not been used for outlier detection. One-class classification can be viewed as a special case of outlier detection. Khot *et al.* introduced a non-parametric relational one-class classification based on first-order trees. They proposed a tree-based distance metric to discover new relational features and to differentiate relational examples [23].

Propositionalization summarizes the multi-relational data into a single data table and can be used for outlier detection and classification tasks [28, 36, 29, 12, 5].

A latent variable approach in information networks ranks potential outliers in reference to the latent communities inferred by network analysis [15]. Our model also aggregates information from entities and links of different types, but does not assume that different communities have been identified.

Koh *et al.* [26] propose a method for hierarchical structures represented in XML document trees. Their aim is to identify feature outliers, not class outliers as in our work. Also, they use aggregate functions to convert the object hierarchy into feature vectors. Their outlier score is based on local correlations, and they do not construct a model.

The multi-dimensional data model defines numeric measures for a set of dimensions. The differences in the two data models mean that multi-dimensional outlier detection models [42] do not carry over to object-relational outlier detection. (1) The object data model allows but does not require any numeric measures. In our datasets, all features are discrete. Nor do we assume that it is possible to aggregate numeric measures to summarize lower-level data at higher levels. (2) In scoring a potential outlier object, our method considers other objects *both* below and above the target object in the component hierarchy. OLAP exploration methods consider only cells below or at the same level as the target cell. For example, in scoring a player, our method would consider features of the player's team. Also, the log-likelihood distance outlier score of an object is not determined by the outlier scores of its components, in contrast to approaches derived from Sarawagi *et al.* [42]. They use values such as the most unusual cell that is below a target cell. (3) Our approach models a joint distribution over features, exploiting correlations among features. Most of the OLAP-based methods consider only a single numeric measure at a time, not a joint model.

3 Background: Bayesian Networks for Relational Data

We adopt the Parametrized Bayesian network (PBN) formalism [37] that combines Bayesian networks with logical syntax for expressing relational concepts.

3.1 Relational Data

We adopt a term-based notation for combining logical and statistical concepts [37,24]. Table 1 summarizes our notation. A functor is a function or predicate symbol. Each functor has a set of values (constants) called the **domain** of the functor. The domain of a **predicate** is $\{T, F\}$. Predicates are usually written with uppercase Roman letters, other terms with lowercase letters. A predicate of arity at least two is a **relationship** functor. Relationship functors specify which objects are linked. Other functors represent **features** or **attributes** of an object or a tuple of objects (i.e., of a relationship). A **population** is a set of objects. A **term** is of the form $f(\sigma_1, \ldots, \sigma_k)$ where f is a functor and each σ_i is a first-order variable or a constant denoting an object.

Symbol	Definition
$a, b, a_1, b_1, \ldots$	Constant
$A, B, T, M, \ldots$	First-order variable
$f, g, \ldots,$	Functor, function symbol
$f(A, \ldots, A_k)$	First-order term
$f(A = a_1, \ldots, A_k = a_k)$	Ground term
$\mathcal{D}$	Relational database
$\mathcal{D}_C$	Database for the entire class of objects
$\mathcal{D}_o$	Restriction of the input database to the target object
$V(A, ..., B)$	A parametrized random variable
$\boldsymbol{V}$	A set of first-order random variable
$\boldsymbol{V} = \mathbf{v}$	Joint assignment of values to a set of PRVs
$P(\boldsymbol{V} = \mathbf{v}) \equiv P(\mathbf{v})$	Joint probability that each variable V_i takes on value $\mathbf{v}_i$
$\#_{\mathcal{D}}(\boldsymbol{V} = \mathbf{v})$	Count of groundings that satisfy the assignment
$A \backslash v$	Ground a first-order variable
B	A Bayesian network structure
B_C	A Bayesian network structure learned with $\mathcal{D}_C$ as the input database
$\boldsymbol{\theta}_C$	Parameters learned for B_C using $\mathcal{D}_c$ as the input database
$\boldsymbol{\theta}_o$	Parameters learned for B_C using $\mathcal{D}_o$ as the input database
$\mathbf{pa}_i$	Parent of node i

Table 1 Notation and Definition

A term is **ground** if it contains no first-order variables; otherwise it is a first-order term. In the context of a statistical model, we refer to first-order terms as **Parametrized Random Variables** (PRVs) [24]. A **grounding** replaces each first-order variable in a term by a constant; the result is a ground term. A grounding may be applied simultaneously to a set of terms. A relational database $\mathcal{D}$ specifies the values of all ground terms.

Consider a joint assignment $P(\boldsymbol{V} = \mathbf{v})$ of values to a set of PRVs $\boldsymbol{V}$. The *grounding space* of the PRVs is the set of all possible grounding substitutions, each applied to all PRVs in $\boldsymbol{V}$. The *count* of groundings that satisfy the assignment with respect to a database $\mathcal{D}$ is denoted by $\#_{\mathcal{D}}(\boldsymbol{V} = \mathbf{v})$. The **database frequency** $P_{\mathcal{D}}(\boldsymbol{V} = \mathbf{v})$ is the grounding count divided by the number of all possible groundings.

Example. The Opta dataset represents information about Premier League data (Sec. 7.2). The basic populations are teams, players, matches, with corresponding first-order variables T, P, M. As shown in Table 2, the groundings count can be visualized in terms of a groundings table [45], also called a universal schema [41]. The first three column headers show first-order variables ranging over different populations. The remaining columns represent terms. Each row represents a single grounding and the values of the ground terms defined by the grounding. In terms of the grounding table, the grounding count of a joint assignment is the number of rows that satisfy the conditions in the joint assignment. The database frequency is the grounding count, divided by the total number of rows in the groundings table. Counts are based on the 2011-2012 Premier League Season. We count only groundings $(team, match)$ such that *team* plays in *match*. Each team, including Wigan Athletics, appears in 38 matches. The total number of team-match pairs is $38 \times 20 = 760$.

A novel aspect of our paper is that we learn model parameters for specific objects as well as for the entire population. The appropriate **object data**

MatchId M	TeamId T	PlayerId P	TimePlayed(P,M)	ShotEff(T,M)	result(T,M)
117	WA	McCarthy	90	0.53	*win*
148	WA	McCarthy	85	0.57	*loss*
15	MC	Silva	90	0.59	*win*
...	...	...	...	...	

Table 2 Sample Population Data Table (Soccer).

MatchId M	TeamId $T = WA$	PlayerId P	TimePlayed(P,M)	ShotEff(WA,M)	result(WA,M)
117	WA	McCarthy	90	0.53	*win*
148	WA	McCarthy	85	0.57	*loss*
...	WA	...	...	...	

Table 3 Sample Object Data Table, for team $T = WA$.

Database	Count or $\#_D(\boldsymbol{V} = \mathbf{v})$	Frequency or $P_D(\boldsymbol{V} = \mathbf{v})$
Population	76	$76/760 = 0.10$
Wigan Athletics	7	$7/38 = 0.18$

Table 4 Example of Grounding Count and Frequency in Premier League Data, for the conjunction $passEff(T, M) = hi, shotEff(T, M) = hi, Result(T, M) = win$.

table is formed from the population data table by restricting the relevant first-order variable to the target object. For example, the object database for target Team *WiganAthletic*, forms a subtable of the data table of Table 2 that contains only rows where TeamID = *WA*; see Table 3. In database terminology, an object database is like a view centered on the object.

3.2 Bayesian Networks

A Bayesian network (BN) structure B is a Directed Acyclic Graph (DAG) whose nodes comprise a set of random variables [34]. Depending on context, we interchangeably refer to the nodes and variables of a BN. Fix a set of variables $\boldsymbol{V} = \{V_1, \ldots, V_n\}$. The possible values of V_i are enumerated as $\{v_{i1}, \ldots, v_{ir_i}\}$. The notation $P(V_i = v) \equiv P(v)$ denotes the probability of variable V_i taking on value v. We also use the vector notation $P(\boldsymbol{V} = \mathbf{v}) \equiv P(\mathbf{v})$ to denote the joint probability that each variable V_i takes on value $\mathbf{v}_i$.

The conditional probability parameters of a Bayesian network specify the distribution of a child node given an assignment of values to its parent nodes. For an assignment of values to its nodes, a BN defines the joint probability as the product of the conditional probability of the child node value given its parent values, for each node in the network. This means that the log-joint probability can be *decomposed* as the node-wise sum

$$\ln P(\boldsymbol{V} = \mathbf{v}; B, \boldsymbol{\theta}) = \sum_{i=1}^{n} \ln \theta(\mathbf{v}_i | \mathbf{v}_{\mathbf{pa}_i}) \tag{1}$$

where $\mathbf{v}_i$ resp. $\mathbf{v}_{\mathbf{pa}_i}$ is the assignment of values to node V_i resp. the parents of V_i determined by the assignment $\mathbf{v}$. To avoid difficulties with ln(0), here and below we assume that joint distributions are positive everywhere. Since

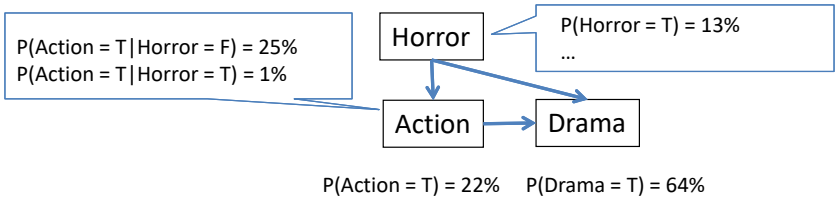


Fig. 2 Example of joint and marginal probabilities computed from a simple Bayesian network structure that was learned from the *Movies* table in the IMDb dataset described in Section 7.2. The parameters were estimated from the IMDb dataset. The conditional probability parameters for the *Drama* node are not shown. The Bayesian network shows that movie genres are largely but not completely exclusive. For instance, among horror movies, only 1% are also classified as action movies. The marginal probabilities are the base rate frequencies of horror, action, and drama movies, which are respectively 13%, 22%, 64%.

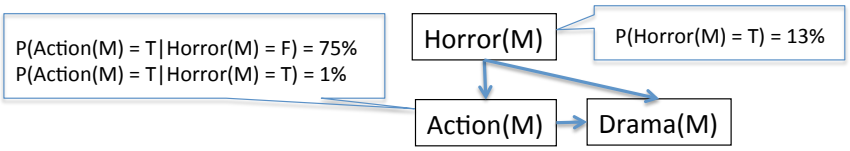


Fig. 3 The Bayesian network from Figure 2, where the node names are expanded using the syntax of parametrized random variables.

the parameter values $\boldsymbol{\theta}$ for a Bayesian network define a joint distribution over its nodes, they therefore entail a marginal, or unconditional, probability for a single node. We denote the **marginal probability** that node V has value v as $P(V = v; B, \boldsymbol{\theta}) \equiv \theta(v)$. In the following we use the term Bayesian network model to refer to a network structure with parameters (i.e., a pair $(B, \boldsymbol{\theta})$); for brevity, we also use the terms "Bayesian network" or "model".

Example. Figure 2 shows an example of a Bayesian network model and associated conditional and marginal probabilities.

3.3 Bayesian Networks for Relational Data

A **Parametrized Bayesian Network Structure** (PBN) is a Bayesian network structure whose nodes are PRVs [37]. The relationships and features in an object database define a set of nodes for a Parametrized Bayesian network. I.i.d. data represented in a single table can be viewed as a special limiting case of multi-relational data with no relationships [32]. Synatically, this means that columns in an i.i.d. data table represent unary functors, where the relevant population is assumed to be clear from the context rather than explicitly specified as a first-order variable. Figure 3 illustrates how the usual syntax for i.i.d. Bayesian networks is a special case of the PBN syntax.

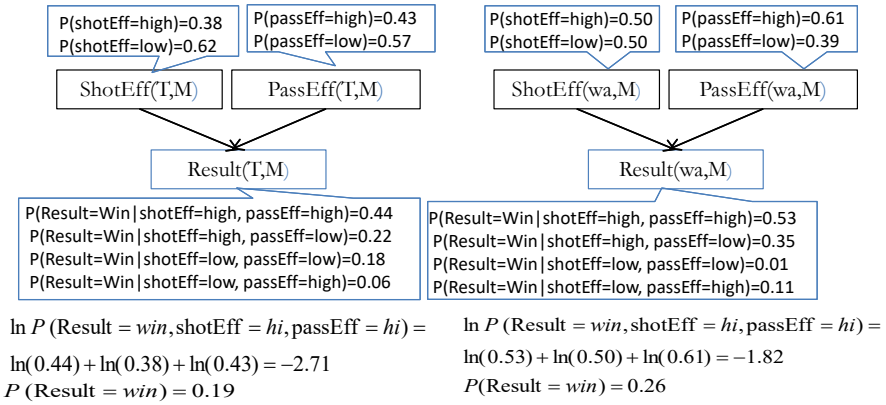


Fig. 4 Example of joint and marginal probabilities computed from a toy Bayesian network structure. The parameters were estimated from the Premier League dataset. (left): A class model Bayesian network B_c for all teams with class parameters $\boldsymbol{\theta}_c$. The first-order variable T ranges over teams, and the first-order variable M over matches. (right): The same Bayesian network structure with object parameters $\boldsymbol{\theta}_o$ learned for Wigan Athletics ($T = WA$).

Figure 4 shows a Parametrized Bayesian network for the Premier League domain that is properly relational in that the functors depend on more than one population variable. For example, shot efficiency is not a function of a match only, but also depends on specifying a team. The BN product formula (1) can be applied to any PBN to compute (estimated) frequencies. In the case of a truly relational PBN, as in Figure 4, the PBN can be viewed as representing database frequencies (rather than data table frequencies as in Figure 3). Using Getoor's terminology, the PBN can be viewed as a Statistical-Relational Model (SRM) [16, 46, 44].

3.4 Likelihood Score for Parametrized Bayesian Networks.

A standard method for applying a generative model assumes that the generative model represents normal behavior since it was learned from the entire population. An object is deemed an outlier if the model assigns sufficiently low likelihood to generating its features [8]. This likelihood method is an important baseline for our investigation. The other outlier scores we consider can be viewed as improved variants of the likelihood approach. Defining a likelihood for relational data is more complicated than for i.i.d. data, because an object is characterized not only by a feature vector, but by an object database. We employ the previously defined relational pseudo log-likelihood [43] score, which can be computed as follows for a given Bayesian network and database.

$$LOG(\mathcal{D}, B, \boldsymbol{\theta}) = \sum_{i=1}^{n} \sum_{j=1}^{r_i} \sum_{\mathbf{pa}_i} P_{\mathcal{D}}(v_{ij}, \mathbf{pa}_i) \ln \theta(v_{ij}|\mathbf{pa}_i) \tag{2}$$

Equation (2) has the same form of the standard BN log-likelihood function for vectorial i.i.d. data [9], except that parent-child instantiation counts are standardized to be proportions [43]. For the notations please refer to Table 1. The condition under which the score of Equation (2) is normalized (sums to 1 over all relational datasets) are discussed in [43].[1] The equation can be read as follows.

1. For each parent-child configuration, use the conditional probability of the child given the parent.
2. Multiply the logarithm of the conditional probability by the database frequency of the parent-child configuration.
3. Sum this product over all parent-child configurations and all nodes.

The maximum of the pseudo-likelihood (2) is given by the empirical database frequencies [43, Prop.3.1.]. In all our experiments we use these maximum likelihood parameter estimates.

Example. The family configuration

$$\mathit{passEff}(T, M) = \mathit{high}, \mathit{shotEff}(T, M) = \mathit{high}, \mathit{Result}(T, M) = \mathit{win}$$

contributes one term to the pseudo log-likelihood for the BN of Figure 4. For the population database, this term is $0.1 \times \ln(0.44) = -0.08$. For the Wigan Athletics database, the term is $0.18 \times \ln(0.44) = -0.14$.

4 Likelihood-Distance Object Outlier score

In this section, we introduce a novel model-based outlier score, that extends the log-likelihood (2), and outline the steps involved in the score computation.

4.1 Computation Flow

We use the following notation to define the computation steps for the relational outlier scores examined in this paper.

- $\mathcal{D}_C$ is the database for the entire class of objects; cf. Table 2. This database defines the **class distribution** $P_C \equiv P_{\mathcal{D}_c}$.
- $\mathcal{D}_o$ is the restriction of the input database to the target object; cf. Table 3. This database defines the **object distribution** $P_o \equiv P_{\mathcal{D}_o}$.
- B_C is a Bayesian network structure learned with $\mathcal{D}_C$ as the input database. Note that we use the entire population data to learn the Bayesian network structure. Therefore, Bayesian network structure is the same across different individuals.

[1] Briefly, if the Parametrized Bayesian Network is viewed as a template for an unrolled ground network, then (1) the ground network cannot contain cycles, and (2) each ground network cannot have multiple parent instantiations; see also [19].

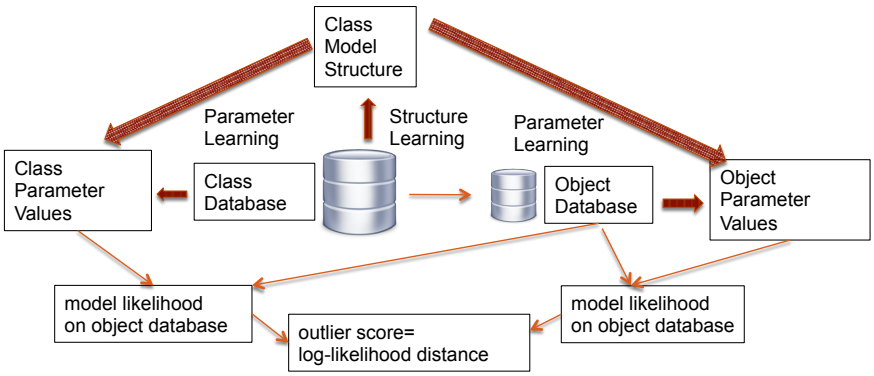


Fig. 5 Computation of outlier score.

– $\boldsymbol{\theta}_C$ resp. $\boldsymbol{\theta}_o$ are parameters learned for B_C using $\mathcal{D}_c$ resp. $\mathcal{D}_o$ as the input database.

Figure 5 illustrates these concepts and the system flow for computing an outlier score. First, we learn a Bayesian network structure B_C for the entire population using a previous learning algorithm (see Section 7.4). We then evaluate *how well the class model fits the target object data.*

4.2 Definition of Log-Likelihood Distance Outlier Metric

The score that we propose for quantifying the model fit is the **expected log-likelihood distance** (ELD). It is defined as follows for each feature i; the total score is the sum of feature-wise scores. Section 5 provides example computations.

$$ELD_i = \sum_{j=1}^{r_i} P_o(v_{ij}) \left| \ln \frac{\theta_o(v_{ij})}{\theta_C(v_{ij})} \right| + \tag{3}$$

$$\sum_{j=1}^{r_i} \sum_{\mathbf{pa}_i} P_o(v_{ij}, \mathbf{pa}_i) \left| \ln \frac{\theta_o(v_{ij}|\mathbf{pa}_i)}{\theta_o(v_{ij})} - \ln \frac{\theta_C(v_{ij}|\mathbf{pa}_i)}{\theta_C(v_{ij})} \right|. \tag{4}$$

4.3 Interpretation

The first sum (3) is the **single-feature** component, where each feature is considered independently of all others. It computes the expected log-distance with respect to the single feature value probabilities between the object and the class models. The second ELD sum (4) is the **mutual information component**, based on the mutual information among all features. It computes the expected log-distance between the object and the class models with respect

to the mutual information of feature value assignments. Intuitively, the first sum measures how the models differ if we treat each feature in isolation. The second sum measures how the models differ in terms of how strongly parent and child features are associated with each other.

4.4 Motivation

The motivation for the mutual information decomposition is two-fold. (1) *Interpretability*, which is very important for outlier detection. The single-feature components are easy to interpret since they involve no feature interactions. Each parent-child local factor is based on the average relevance of parent values for predicting the value of the child node, where relevance is measured by

$$\ln \frac{\theta(v_{ij}|\mathbf{pa}_i)}{\theta(v_{ij})}.$$

This relevance term is basically the same as the widely used lift measure [52], therefore an intuitively meaningful quantity. The *ELD* score compares how relevant a given parent condition is in the object data with how relevant it is in the general class.

(2) *Avoiding cancellations.* For different child-parent configurations, the different components of the *ELD* sum may have different signs. This leads to a partial cancelling of differences between the class and object distribution. Since our goal is to assess the distinctness of an object, *we do not want differences to cancel out.* Taking distances as in Equations (3) and (4) avoids the undesirable loss of information. The general point is that averaging differences is appropriate when considering costs, or utilities, but not appropriate for assessing the distinctness of an object. In contrast, the absolute values add differences regardless of their sign. The next section provides example computations that illustrate the cancelling phenomenon that occurs without adding absolute values.

5 Score Computation Examples

We provide three simple examples with only two variables/features that illustrate the computation of the outlier scores. They are designed so that outliers and normal objects are easy to distinguish, and so that it is easy to trace the behavior of an outlier score. The examples therefore serve as thought experiments that bring out the strengths and weaknesses of model-based outlier scores. Figure 6 describes the BN representation of the examples. For intuition, we can think of a soccer setting, where each match assigns a value to each attribute $F_i, i = 1, 2$ for each player.

Example Computations. Table 5 illustrates the computation of the scores. Scores for the F_2 feature are computed conditional on $F_1 = 1$. Expectation terms are computed first for $F_2 = 1$, then $F_2 = 0$.

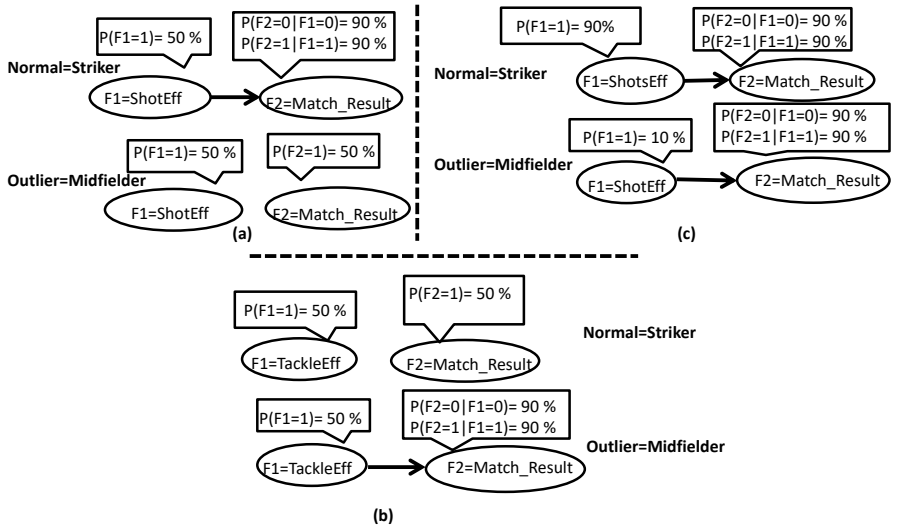


Fig. 6 Illustrative Bayesian networks with two nodes. The networks are not learned from data, but hand-constructed to be plausible for the soccer domain.

Table 5 Example computation of different outlier scores for outliers given the distributions of Figure 6(a),(b).

Score	$F1 = 1$ Computation	$F2\|F1 = 1$ Computation	Result
LR	$1/2\ln(0.5/0.5) = 0$	$1/4\ln(0.5/0.9) + 1/4\ln(0.5/0.1)$	0.36
FD	$\|\ln(0.5/0.5)\| = 0$	$1/2\|\ln(0.5/0.5)\| + 1/2\|\ln(0.5/0.5)\|$	0
ELD	0 (no parents)	$1/2\|\ln(0.5/0.5)\| + 1/2\|\ln(0.5/0.5)\| + 1/4\|\ln(0.5/0.5) - \ln(0.9/0.5)\| + 1/4\|\ln(0.5/0.5) - \ln(0.1/0.5)\|$	0.79

Table 5(a): High Correlation Case. Figure 6(a).

Score	$F1 = 1$ Computation	$F2\|F1 = 1$ Computation	Result
LR	$1/2\ln(0.5/0.5) = 0$	$0.5 \cdot 0.9\ln(0.9/0.5) + 0.5 \cdot 0.1\ln(0.1/0.5)$	0.26
FD	$\|\ln(0.5/0.5)\| = 0$	$1/2\|\ln(0.5/0.5)\| + 1/2\|\ln(0.5/0.5)\|$	0
ELD	0 (no parents)	$1/2\|\ln(0.5/0.5)\| + 1/2\|\ln(0.5/0.5)\| + 0.5 \cdot 0.9\|\ln(0.9/0.5) - \ln(0.5/0.5)\| + 0.5 \cdot 0.1\|\ln(0.1/0.5) - \ln(0.5/0.5)\|$	0.50

Table 5(b): Low Correlation Case. Figure 6(b).

6 Comparison Scores

We introduce several outlier scores for comparison, following a lesion design where different scores omit different components of our main *ELD* proposal. We introduce the scores. To illustrate their essential difference with *ELD*, we give toy examples before we evaluate them on full datasets.

6.1 Definition of Comparison Outlier Scores

Log-likelihood Ratio Score. Our first comparison score omits the absolute values from the *ELD* score:

$$LR_i = \sum_{j=1}^{r_i} P_o(v_{ij}) \ln \frac{\theta_o(v_{ij})}{\theta_C(v_{ij})} + \sum_{j=1}^{r_i} \sum_{\mathbf{pa}_i} P_o(v_{ij}, \mathbf{pa}_i)(\ln \frac{\theta_o(v_{ij}|\mathbf{pa}_i)}{\theta_o(v_{ij})} - \ln \frac{\theta_C(v_{ij}|\mathbf{pa}_i)}{\theta_C(v_{ij})}).$$

By using the **mutual information decomposition:**

$$\ln \frac{\theta_o(v_{ij}|\mathbf{pa}_i)}{\theta_C(v_{ij}|\mathbf{pa}_i)} = \ln \frac{\theta_o(v_{ij}|\mathbf{pa}_i)}{\theta_o(v_{ij})} - \ln \frac{\theta_C(v_{ij}|\mathbf{pa}_i)}{\theta_C(v_{ij})} + \ln \frac{\theta_o(v_{ij})}{\theta_C(v_{ij})}, \tag{5}$$

it can be shown that the *ELD* score without the absolute values is equivalent to the the likelihood ratio, or **log-likelihood difference**:

$$LR(\mathcal{D}_o, B_C, \boldsymbol{\theta}_o) \equiv LOG(\mathcal{D}_o, B_C, \boldsymbol{\theta}_o) - LOG(\mathcal{D}_o, B_C, \boldsymbol{\theta}_C). \tag{6}$$

Assuming maximum likelihood parameter estimation, *LR* is equivalent to the Kullback-Leibler divergence between the class-level and object-level parameters [9]. The log-likelihood difference compares how well the class-level parameters fit the object data with respect to a particular object, vs. how well the object parameters fit the object data. In terms of the conditional probability parameters, it measures how much the log-conditional probabilities in the class distribution differ from those in the object distribution.

Log-Likelihood Score. In previous work on applying Bayesian networks to outlier detection for i.i.d. non-relational data, the outlier metric used was the log-likelihood of a datapoint [8]. This means that an object was deemed an outlier if the model assigns sufficiently low likelihood to generating its features. For relational data, we use the relational log-likelihood (2) of the target *database*:

$$LOG(\mathcal{D}_o, B_C, \boldsymbol{\theta}_C) = \sum_{i=1}^{n} \sum_{j=1}^{r_i} \sum_{\mathbf{pa}_i} P_{\mathcal{D}}(v_{ij}, \mathbf{pa}_i) \ln \theta(v_{ij}|\mathbf{pa}_i). \tag{7}$$

The Feature Divergence Score. The feature divergence outlier score *FD* uses only part (3) of the *ELD* score. That is, it considers each feature independent of all others. *FD* computes the expected log-distance with respect to the single feature value probabilities between the object and the class models. This can be computed using the following formula:

$$FD_i = \sum_{i=1}^{n} \sum_{j=1}^{r_i} P_o(v_{ij}) \left| \ln \frac{\theta_o(v_{ij})}{\theta_C(v_{ij})} \right| \tag{8}$$

6.2 Comparison on Examples

Table 5 illustrates the undesirable cancelling effects in LR. In the high correlation scenario 6(a), the outlier object has a lower probability than the normal class distribution of $Match\_Result = 0$ given that $Shot\_Efficiency = 1$. Specifically, 0.5 vs. 0.9. The outlier object exhibits a higher probability $Match\_Result = 1$ than the normal class distribution, conditional on $Shot\_Efficiency = 1$; specifically, 0.5 vs. 0.1. In line 1, column 2 of Table 5 the log-ratios ln(0.5/0.9) and ln(0.5/0.1) therefore have different signs. In the low correlation scenario 6(b), the cancelling occurs in the same way, but with the normal and outlier probabilities reversed. The cancelling effect is even stronger for attributes with more than two possible values.

While log-likelihood LOG is a good baseline score for detecting outliers, it fails to detect some clear outliers, as shown in Figure 7. In that example the strength of the correlation among the attributes is the same in both normal and outlier examples and the only difference is in their feature distributions.

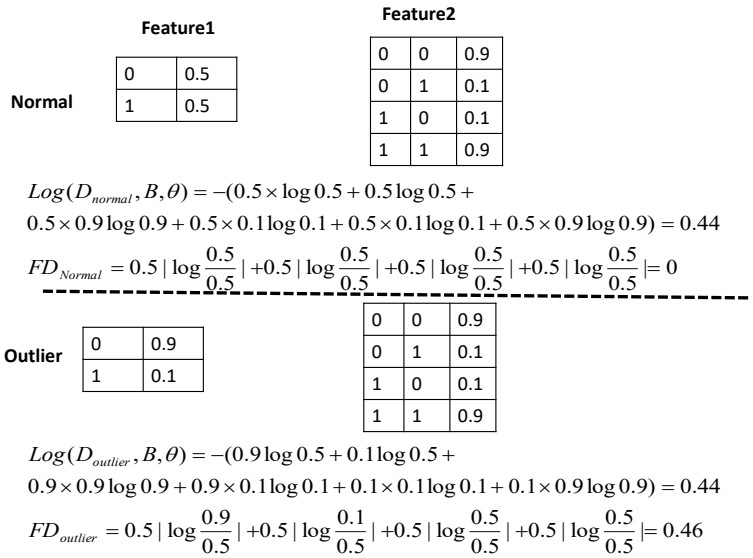


Fig. 7 An example of normal and outlier individuals and their conditional probability tables created using Bayesian network shown in Figure 6(c). Log-likelihood assigns the same score to the normal and individuals in this example, while FD is able to differentiate between these two individuals.

Conversely, if there is a correlation among the attributes of individuals, the feature divergence score FD fails to take it into account and therefore fails to differentiate between normal and outlier individuals. Figure 8 shows an example where the normal individual has a correlation among its attributes

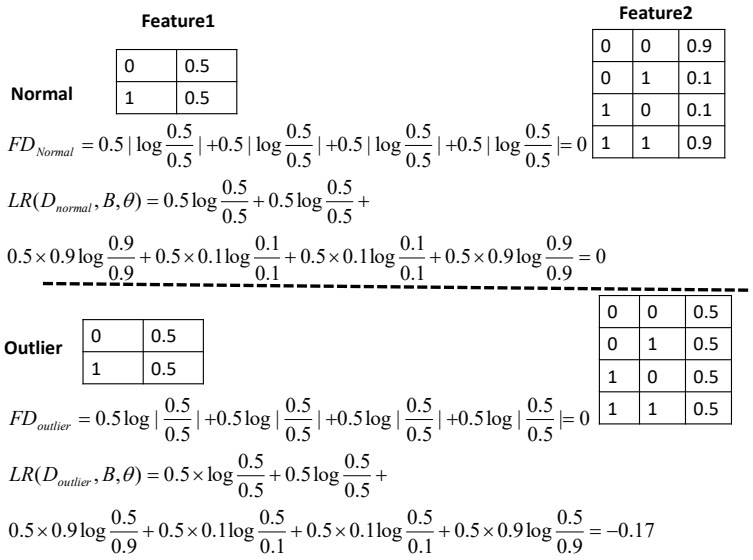


Fig. 8 An example of normal and outlier individuals and their conditional probability tables created using Bayesian network shown in Figure 6(a). FD assigns the same score to the normal and individuals in this example, while LR is able to differentiate between these two individuals.

while the outlier object does not have a correlation. The FD metric cannot separate those two individuals.

7 Experimental Evaluation

All the experiments were performed on a 64-bit Centos machine with 4GB RAM and an Intel Core i5-480 M processor. The likelihood-based outlier scores were computed with SQL queries using JDBC, JRE 1.7.0. and MySQL Server version 5.5.34. We utilized the synthetic datasets and real-world datasets from the soccer, hockey, movie and biology domains.

7.1 Synthetic Datasets

We generated three synthetic datasets with normal and outlier players using the distributions represented in the three Bayesian networks of Figure 6. The main goal of designing synthetic experiments is to test the methods on easy to detect outliers. We used the *mlbench* package in R to generate synthetic features in matches, following these distributions for 240 normal players and 40 outliers. (There were 280 players in our Premier League dataset.) Each player participates in 38 matches, similar to the Premier League's real-world

Premier League Statistics		IMDB Statistics		NHL Statistics	
Number of Teams	20	Number of Movies	3060	Number of Teams	30
Number of Players	484	Number of Directors	220	Number of Players	921
Number of Matches	380	Number of Actors	98690	Number of Matches	720
avg player per match	26.01	avg actor per movie	36.42	avg player per match	29

Table 6 Summary Statistics for the IMDB and PL and NHL data sets

data. Each match assigns a value to each feature $F_i, i = 1, 2$ for each player. This design yields the following three synthetic datasets.

High Correlation Normal individuals exhibit a strong association between their features, outliers no association. Both normals and outliers have a close to uniform distribution over single features. See Figure 6(a).

Low Correlation Normal individuals exhibit no association between their features, outliers have a strong association. Both normals and outliers have a close to uniform distribution over single features. See Figure 6(b).

Single Features Both normal and outlier individuals exhibit a strong association between their features. In normals, 90% of the time, feature 1 has value 1. For outliers, feature 1 has value 1 only 10% of the time. See Figure 6(c).

7.2 Real-world Datasets

Our datasets and code are available on-line [39]. The following is the list of the real-world datasets we used in this work:

Premier League Data The Opta data were released by Manchester City. It lists box scores, that is, counts of all the ball actions within each game by each player, for the 2011-2012 season. For each player in a match, our data set contains eleven player features. For each team in a match, there are five features computed as player feature aggregates, as well as the team formation and the result (win, tie, loss). There are two relationships, $Appears\_Player(P, M)$, $Appears\_Team(T, M)$.

IMDb Data The Internet Movie Database (IMDb) is an on-line database of information related to films, television programs, and video games. The IMDb website offers a dataset containing information on cast, crew, titles, technical details and biographies into a set of compressed text files. We preprocessed the data like [35] to obtain a database with seven tables: one for each population and one for the three relationships $Rated(User, Movie)$, $Directs(Director, Movie)$, and $ActsIn(Actor, Movie)$.

National Hockey League Data The NHL provides information about the sequences of play-by-play events. We used the data crawled by [47] that consists of the player game statistics for the seasons 2009-2013. The box scores summarize player statistics for each match, a total of 13 features.

Individuals	Features
PL-Player per Match	$TimePlayed, Goals, SavesMade, ShotEff, PassEff, WinningGoal, FirstGoal, PositionID, TackleEff, DribbleEff,$ $ShotsOnTarget$
PL-Team per Match	$Result, TeamFormation, \sum Goals, \mu ShotEff, \mu PassEff, \mu TackleEff, \mu DribbleEff.$
IMDB-Actor	$Quality,$ $Gender$
IMDB-Director	$Quality, avgRevenue$
IMDB-Movie	$year, isEnglish, Genre, Country,$ $RunningTime,$ $Rating$ by User
IMDB-User	$Gender,$ $Occupation.$
NHL-Player per Match	$Goals, Assists, Points, PowerPlayTime, PlusMinus, Penalties, Shots, Hits, BlockedShots, Giveaways, ShotHandedTime, TimeOnIce$
NHL-Team per Match	$Goals,$ $GoalDifference$

Table 7 Attribute Features.

Mutagenesis Data This dataset contains mutagenic activity of 188 compounds. 125 of these compounds have positive levels of log mutagenicity that are labelled "active". The remaining compounds are labelled "inactive" and constitute the source of negative examples. In this dataset we considered examples of active compounds as the normal population and the inactive ones as the outlier.

7.3 Evaluation Techniques for Outlier Detection

Measuring the effectiveness of outlier detection methods is not often an easy task. Most of the time ground truth information, that shows which data points are outliers, is unavailable.
Several techniques have been employed in the literature to evaluate the performance of outlier detection methods:

1. Intuitive evaluation: case studies have been extensively used in the literature to evaluate outliers [2]. In Section 8.3 we use this method of evaluation for the top n ranked detected outliers and we try to explain and make sense of the detected outliers.
2. Synthetic data generation: another approach to evaluate anomaly detection methods is generating synthetic data and injecting synthetic outliers into the data [2]. We have designed and developed three synthetic datasets as discussed in section 7.1.

Normal class	#Normal	Outlier	#Outlier class
Striker	153	Goalie	22
Midfielder	155	Striker	74
Drama	197	Comedy	47
Defender	186	Forward	38
Positive Compound	125	Negative Compound	63

Table 8 Outlier/normal Objects in Real-World Datasets.

3. Anomaly injection: anomalies are injected into real-world datasets. Outlier detection methods are expected to detect the injected data points as outliers [4]. We employ this approach in our real-world datasets. In real-world data, there is no ground truth about which objects are outliers. To address this issue, we employ a one-class design: we learn a model for the class distribution, with data from that class only. Then we rank all individuals from the normal class together with all objects from a contrast class treated as outliers, to test whether an outlier score recognizes objects from the contrast class as outliers. Table 7.3 shows the normal and contrast classes for three different datasets. In-class outliers are possible, e.g. unusual strikers are still members of the striker class. Our case studies describe a few in-class outliers. In the soccer data, we considered only individuals who played more than 5 matches out of a maximum 38. For the three synthetic datasets, the scores are used to rank all 280 synthetic players, 240 of which are normal and 40 are outliers. The disadvantage of this evaluation metric is that the real-world data may also contain anomalies, known as in-class outliers. However, this metric treats only the injected data points as true positives and will score anything other than those as false positives.

7.4 Methods Compared

We compare three types of approaches, based on relational model likelihood, aggregation, and distance.

Likelihood-based Outlier Scores. The first approach evaluates the likelihood-based outlier scores described in Section 6. For relational Bayesian network structure learning we utilize the previous learn-and-join algorithm (LAJ), which is a state-of-the-art BN structure learning method for relational data [45]. The LAJ algorithm employs an iterative deepening strategy, which can be described as as search through a lattice of table joins. For each table join, different BNs are learned and the learned edges are propagated from smaller to larger table joins. For a full description, complexity analysis, and learning time measurements, please see [45]. We used the implementation of the LAJ algorithm due to its creators [22].

Aggregation-based Methods. The second approach first "flattens" the structured data into a matrix of feature vectors, then applies standard matrix-based

outlier detection methods. We refer to such methods as **aggregation-based** (cf. Figures 1). For example, this was the approach taken by Breunig *et al.* for identifying anomalous players in sports data [7]. Following their paper, for each continuous feature in the object data, we use the average over its values, and for each discrete feature, we use the occurrence count of each feature value in the object data. Aggregation tends to lose information about correlations. Our experiments address the empirical question of whether this loss of information affects outlier detection. We evaluated three standard matrix-based outlier detection methods: Density-based *LOF* [7], distance-based *KNNOutlier* [38] and subspace analysis *OutRank* [31]. These represent common, fundamental approaches for vectorial data. Like *ELD*, subspace analysis is sensitive to correlations among features. We used the available implementation of all three data matrix methods from the state of the art data mining software *ELKI* [1]. We used *PRO-CLUS* as the clustering function for *OutRank* as recommended by [31].

Relational Distance-based Method. The third approach is based on a first-order distance measure that was developed and first used for the instance-based learning system RIBL2 [21]. This measure has proven successful in several applications [25, 20]. We employ this metric and compute the distance between any two individuals in our population domain. Then, we rank the individuals based on their mean distance to the normal population.

8 Empirical Results

We present results regarding computational feasibility, predictive performance, and case studies.

8.1 Computational Cost of the *ELD* Score.

Table 9 shows that the computation of the *ELD* value for a given target object is feasible. On average, it takes a quarter of a minute for each soccer player, and one minute for each movie. This includes the time for parameter learning from the object database. Learning the class model BN takes longer, but needs to be done only once for the entire object class. *The BN model provides a crucial low-dimensional representation of the distribution information in the data.*

8.2 Detection Accuracy

Our performance score for outlier rankings is the area under curve (AUC) of the well-established receiver operating characteristic *ROC* curve [13]. This has been widely used to measure the performance of outlier ranking methods [8, 31]. The relationship between false positive rate (1- Specificity) and

Dataset	Class Model	Average per Object
PL: Strikers vs. Goalies	4.14	0.25
PL: Midfielder vs. Goalies	4.02	0.25
IMDb: Drama vs. Comedy	8.30	1.00
PL: Forward vs. Defender	5.30	0.35
Mutagenesis: Positive Compound vs. Negative Compound	1.40	0.1

Table 9 Time (min) for computing the *ELD* score.

true positive rate (Sensitivity) is captured by the *ROC* curve. Ideally, the best performance is achieved when we have the highest sensitivity and the highest specificity. The maximum values for AUC is, 1.0 indicating a perfect ranking with 100% sensitivity and 100% specificity. In order to compute the AUC value, we used the R package *ROCR* [48]. Given a set of outlier scores, one for each object, this package returns an AUC value.

Detection Accuracy in Synthetic Datasets Table 10 shows the AUC values for probabilistic methods and the outlier detection methods. On the synthetic data, it ought to be easy to distinguish the outliers and most methods have high performance to detect outliers. However, *ELD* is the only score that achieves perfect detection across all three synthetic datasets. *RIBL* is the only method that fails in detecting outliers. *RIBL* computes the distance between individuals by computing the distance of each instance of an individual to its nearest instance from other instances of the other individual. When comparing two individuals with only two binary features, *RIBL* results in the same distance from the center of normal community for each individual.

Detection Accuracy in Real-world Datasets Table 11 shows the AUC values for aggregation-based methods compared to *ELD* and *RIBL* in the real-world datasets. *RIBL*'s depth bound that is used to control the recursion is set to 2. *RIBL* and *ELD* perform substantially better than aggregation-based methods on all datasets confirming that it is important to develop outlier detection methods based on relational statistics for the relational data. The *ELD* works better than *RIBL* or equally good in all the datasets except Midfielder vs. Strikers. In that dataset *ELD* finds many in-class exceptional midfielders. However, *RIBL* seems to perform better in distinguishing these two classes from each other. The *LR* metrics outperforms both *RIBL* and *ELD* in the NHL dataset and demonstrate decomposing the log ratio does not alway result in better detection performance. *RIBL*'s behavior in synthetic and real-world datasets suggests that when we have a small number of discrete features in our dataset, *RIBL* may not be a good candidate to compute distance and hence outlier detection. But for numeric features it seems to be a good alternative.

Dataset	*ELD*	*LOG*	LR	FD	*RIBL*	*LOF*	*OutRank*	*KNN Outlier*
High Correlation	**1.00**	0.99	0.97	0.89	0.50	0.68	0.99	0.97
Low Correlation	**1.00**	0.97	0.99	0.42	0.50	0.58	0.83	0.97
Single Feature	**1.00**	0.79	**1.00**	**1.00**	0.50	0.63	0.88	0.86

Table 10 AUC of the *ELD* vs. other Outlier detection methods in Synthetic datasets.

Dataset	*ELD*	*RIBL*	*LOG*	*LR*	*FD*	*LOF*	*OutRank*	*KNN*
PL: Strikers vs. Goalies	**0.89**	0.61	0.61	0.65	0.71	0.61	0.60	0.61
PL: Midfielders vs. Strikers	0.66	**0.78**	0.45	0.55	0.59	0.76	0.71	0.58
IMDb: Drama vs. Comedy	**0.70**	**0.70**	0.66	0.66	0.64	0.51	0.68	0.68
NHL: Defender vs. Forward	0.78	0.71	0.58	**0.87**	0.79	0.73	0.73	0.66
Mutagenesis: Positive vs Negative	**0.86**	0.70	0.64	0.81	0.70	0.51	0.57	0.53

Table 11 AUC of *ELD* vs. other outlier detection methods in real-world datasets.

8.3 Case Studies

For a case study, we examine the three top outliers as ranked by *ELD*, shown in Table 12. The aim of the case study is to provide a qualitative sense of the outliers indicated by the scores. Also, we illustrate how the BN representation leads to an interpretable ranking. Specifically, we employ a *feature-wise decomposition* of the score combined with a *drill down* analysis:

1. Find the node V_i that has the highest ELD_i divergence score for the outlier object.
2. Find the parent-child combination that contributes the most to the ELD_i score for that node.
3. Decompose the *ELD* score for the parent-child combination into feature and mutual information component.

We present strong associations—indicated by the *ELD*'s mutual information component—in the intuitive format of association rules.

Strikers vs. Goalies In real-world data, a rare object may be a *within-class outlier*, i.e., highly anomalous even within its class. In an unsupervised setting without class labels, we do not expect an outlier score to distinguish such an in-class outlier from outliers outside the class. An example is the striker Edin Dzeko. He is a highly anomalous striker who obtains the top *ELD* divergence score among both strikers and goalies. His *ELD* score is highest for the Dribble Efficiency feature. The highest *ELD* score for that feature occurs when Dribble Efficiency is low, and its parents have the following values: Shot Efficiency high, Tackle Efficiency medium. Looking at the single feature divergence, we see that Edin Dzeko is indeed an outlier in the Dribble Efficiency subspace: His dribble efficiency is low in 16% of his matches, whereas a randomly selected striker has low dribble efficiency in 50% of their matches. Thus, Edin Dzeko is an unusually good dribbler. Looking at the mutual information component of *ELD*, i.e., the parent-child correlations, for Edin Dzeko the confidence of the rule

$$ShotEff = high, TackleEff = medium \rightarrow DribbleEff = low$$

is 50%, whereas in the general striker class it is 38%.

Strikers (Normal) vs. Goalies (Outlier)					
PlayerName	Position	*ELD* Rank	*ELD* Max Node	*ELD* Node Score	*FD* Max feature Value
Edin Dzeko	Striker	1	DribbleEfficiency	83.84	DE=low
Paul Robinson	Goalie	2	SavesMade	49.4	SM=Medium4
Michel Vorm	Goalie	3	SavesMade	85.9	SM=Medium
Midfielders (Normal) vs. Strikers (Outlier)					
PlayerName	Position	*ELD* Rank	*ELD* Max Node	*ELD* Node Score	*FD* Max feature Value
Robin Van Persie	Striker	1	ShotsOnTarget	153.18	ST=high
Wayne Rooney	Striker	2	ShotsOnTarget	113.14	ST=high
Scott Sinclair	Midfielder	6	DribbleEfficiency	71.9	DE=high
Drama (Normal) vs. Comedy (Outlier)					
MovieTitle	Genre	*ELD* Rank	*ELD* Max Node	*ELD* Node Score	*FD* Max feature Value
Brave Heart	Drama	1	ActorQuality	89995.4	a_quality=4
Austin Powers	Comedy	2	Cast_Position	61021.28	Cast_Num=3
Blue Brothers	Comedy	3	Cast_Position	24432.21	Cast_num=3
Defender (Normal) vs. Forward (Outlier)					
PlayerName	Position	*ELD* Rank	*ELD* Max Node	*ELD* Node Score	*FD* Max feature Value
Eric Staal	Forward	1	Points	49.57	Points=2
Phil Kessel	Forward	2	Points	43.34	Points=2
Dustin Byfuglien	Defender	3	Power_PlayTime	25.65	PP_time=2

Table 12 Case study for the top outliers returned by the log-likelihood distance score *ELD*

Midfielders vs. Strikers For the single feature score, Robin van Persie is recognized as a clear striker because of the *ShotsOnTarget* feature. It makes sense that strikers shoot on target more often than midfielders. Robin van Persie achieves a high number of shots on targets in 34% of his matches, compared to 3% for a random midfielder. The mutual information component shows that he also exhibits unusual correlations. For example, the confidence of the rule

$$\mathit{ShotEff} = \mathit{high}, \mathit{TimePlayed} = \mathit{high} \rightarrow \mathit{ShotsOnTarget} = \mathit{high}$$

is 70% for van Persie, whereas for strikers overall it is 52%.

The most anomalous midfielder is Scott Sinclair. His most unusual feature is *DribbleEfficiency*: For feature divergence, he achieves a high dribble efficiency 50% of the time, compared to a random midfielder with 30%.

Drama vs. Comedy The top outlier rank is assigned to the within-class outlier *BraveHeart*. Its most unusual feature is *ActorQuality*: In a random drama movie, 42% of actors have the highest quality level 4, whereas for *BraveHeart* 93% of actors achieve the highest quality level.

The *ELD* score identifies the comedies *BluesBrothers* and *AustinPowers* as the top out-of-class outliers. In a random drama movie, 49% of actors have casting position 3, whereas for *AustinPowers* 78% of actors have this casting position, and for *BluesBrothers* 88% of actors do.

Defender vs. Forward The first two players in the ranking belong to forward group and are identified as outliers for their unusual high value for points. Eric Staal has $\mathit{points} = 2$ in 30% of his matches, while an average player has that value for points only on 6% of his matches. Dustin Byfuglien is discovered as a within-class outlier. His most unusual feature is *PowerPlayTime*. While an average player in the population has $\mathit{PowerPlayTime} = 2$ for only 33% of the times, he has that value for 79% of the times.

9 Limitation of Model-based outlier detection

The main limitations of the work presented in this paper are the following:

1. **Limitation of Approach:**
   (a) Our proposed method ranks potential outliers, but does not set a threshold for a binary identification of outlier vs. non-outlier.
   (b) Our current Bayesian network learning method can only be applied to discrete data. Prior to learning the model, we take an extra step in data preprocessing and convert continuous data into discrete, which naturally causes some information loss.
   (c) Our generative model-based methods learn a generic Bayesian network structure for the entire population, ignoring the subgroups that inherently exist in the real-world datasets, as a result, the detected outliers are global outliers. However, there are more complex outliers that locally deviate from their subgroups and can be detected only by subgroup comparison. One direction for future work is to first detect subgroups in the population and then perform the outlier detection task.
2. **Limitation of Data Analysis:**
   In this work, to simplify the outlier detection task, we used only part of the full information available in our rich datasets. The model-based outlier detection can be extended in future work to take advantage of the full information, in the following manner.
   (a) In the Premier League dataset, players are naturally related to one another and modeling the interaction between players can be another way to detect anomalous players. The graph-based features, such as detecting near-clique nodes and star nodes, proved to be efficient in discovering patterns for anomaly detection task as shown in ODDBALL [3].
   (b) In this paper we did not use the temporal information available in the data. In the learning process we do not give a higher weight (importance) to the more recent action (performance) of an individual. This point is especially important when applying the methods to dynamic data or the data that are collected over long periods of time.
   (c) In the datasets that we used for the experiments, we did not have the missing value problem. Therefore, we did not incorporate ways to estimate missing values in our modeling. However, real-world datasets may involve arbitrary pattens of missing data. Maximum likelihood density estimation is a way to estimate such values. We leave this feature for future work.

10 Correlation with Success

The aim of this section is to compare the *ELD* metric with other meaningful metrics for comparing individuals. Our reference metrics are success rankings of individuals selected for a specific domain, shown in Table 13. We use the

same data as in our other experiments, described in Section 7 except Mutagenesis as there was no meaningful success metrics we could find in that dataset.

Success rankings are one of the most interesting features to users. Strong correlations between the *ELD* metric and meaningful success metrics provide evidence that the *ELD* metric is meaningful as well. We measure correlation strength by the standard Pearson correlation coefficient ρ. The coefficient ranges from -1 to 1, where 0 means no correlation and 1 or -1 indicates maximum strength [14].

The observed correlations are remarkable in at least two respects. 1) the strength of the correlation between the *ELD* metric and the success ranking are high: coefficients range from 0.45 to 0.82. 2) We observe this phenomenon across different domains, different types of individuals and different success metrics.

Dataset	Success Metric	Min	Max	Standard Dev.	Mean
IMDb	Sum of Rating	1.0	14795	1600.22	1057.58
PL-Player	TimePlayed	5.0	3420	1015.69	1484.00
PL-Player	Normalized Salary	0.007	0.28	0.62	0.10
PL-Player	Sum of Shot Efficiency	0	82	9.87	6.53
PL-Team	Standing	1.0	20	5.91	10.50
NHL-Player	Power Play Time	0	669	106.78	84.38
NHL-Player	Time on Ice	4	2099	278.03	1187.31
NHL-Player	Assists	0	4	0.49	0.20

Table 13 Success metrics and their distributions.

For a population with a diverse set of skills and resources, being different from the generic class can be interpreted as both exceptionally better or worse than normal population. In the domains we study in this data, we found that higher *ELD* scores indicate exceptionally good individuals but not exceptionally bad individuals. Our interpretation of the correlation between *ELD* and success, rather than failure, is that our domains featured skilled individuals, such that the average is quite successful already. For example, in Premier League we expect most players to be in the range of good players. Therefore, deviating from the rest of the population is a signal for detecting exceptionally good players. Our *ELD*-success scatterplots below provide empirical evidence for this interpretation: we typically see a large cluster of individuals around the origin, meaning that their success level is normal and their *ELD* score is low, see Figure 10.

10.1 Methodology

We report the correlations between the *ELD* metric and metrics of success for a specific domain. We also focus on some unusually successful individuals as case studies. In considering the correlation between *ELD* and success, it

Team	Standing
Top Teams	-0.71
Bottom Teams	-0.33
All Teams	-0.13

Table 14 Correlation between *ELD* metric and standing of Teams. The best standing is place 1.

Class	Time Played	Salary	Saves Made	Shots On target	Pass Efficiency
Strikers	0.86	0.82	NA	0.79	NA
Midfielders	0.80	0.45	NA	NA	0.77
Goalies	0.77	NA	0.74	NA	NA
All players	0.18	0.56	NA	NA	NA

Table 15 Correlation between *ELD* metric and success metrics of Soccer Players.

Genre	Sum of Rating	Average of Rating	Number of Rating
Action	0.67	0.30	0.72
Drama	0.76	0.29	0.81
Comedy	0.85	0.41	0.84
All Movies	0.56	0.17	0.60

Table 16 Correlation between *ELD* metric and success metric of Movies.

is useful to investigate subgroups of individuals to ensure an apples-to-apples comparison [49]. For instance, the attributes that lead to success are different for strikers and goalies. Accordingly, we report correlations for subgroups as well as entire classes of individuals.

10.2 Correlations between the *ELD* outlier metric and success

The next three tables summarize the observed correlations between success and *ELD* metrics: Teams in Table 14, Players in Table 15, Movies in Table 16.

10.2.1 Soccer Teams

Team Standing The most successful team has Standing=1 and the least successful team has Standing=20 in the 2011-2012 Season. For the top teams, there is a very strong negative correlation emerges between *ELD* and standing: teams with higher *ELD* achieve a better (lower) standing.

Figure 9 shows the correlation of *ELD* with team success metrics in a scatter plot. The top two teams Manchester City and Manchester United stand out very strongly in terms of the *ELD* metric (bottom right corner).

10.2.2 Soccer Players

Players Time Played is the total time that a player played over all matches in the season. This metric was shown to correlate strongly with other success

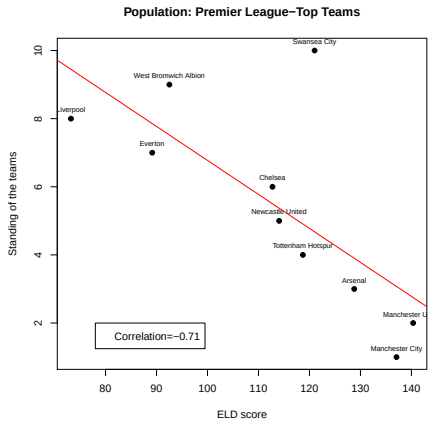


Fig. 9 Teams: Team Standing vs. *ELD* for the top teams in Premier League.

metrics, such as salary, in soccer data [51]. For each subgroup, there is a strong positive correlation with *ELD*, meaning that atypical players with higher *ELD* tend to play more minutes.

Salary is probably the most obvious, and at the same time often the most misleading way to measure success of the players. Previous studies suggest that salary of the players does not always follow their performance in many sports such as Baseball and Soccer [18,10]. They show that pay cannot be explained only by past performance and there are other factors that are hard to quantify and have great effects on the salaries.

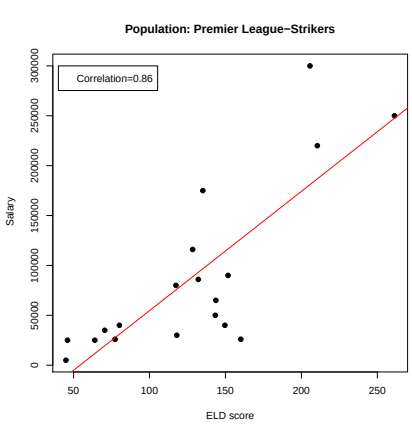
We manually collected salaries of 120 players that we could find on-line. Table 15 and Figures 10 and 11 show the correlation between *ELD* and this success metric. The correlation is high, especially for Strikers. We found salary data for only 5 goalies. We discuss the relatively weaker salary correlation for midfielders in more detail below.

Shots on Target applies to strikers only. This is defined as any shot attempt that would or does enter the goal if left unblocked. We record the total number of these shots over all matches of the strikers only. This metric was shown to correlate strongly with *ELD* (see Table 15, Figure 10(b)).

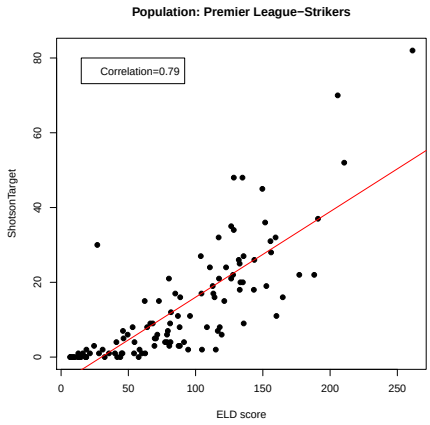
Figure 10 plots *ELD* against striker success metrics. We observe a large cluster around the origin, which points to a large base of normal strikers with both salaries and low *ELD* scores.

Saves Made applies to Goalies only. It is defined as the total number of saves that goalies had made over all the matches. This metric shows a strong correlation with *ELD* as well (see Table 15, Figure11(b)).

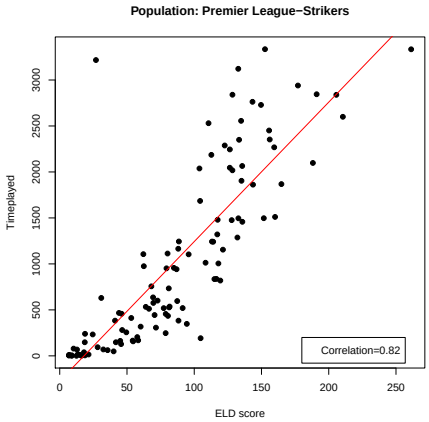
Figure 11 shows the correlation of *ELD* with Goalie success metrics in a scatter plot. Goalies do not vary much in terms of the time they play. Wayne



(a) Strikers: Salary vs *ELD*.



(b) Strikers: Shots On Target vs *ELD*

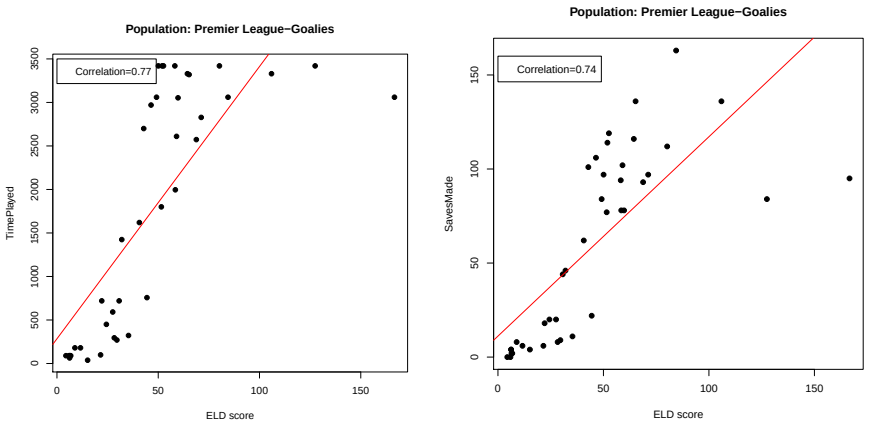


(c) Strikers: Time played vs *ELD*.

Fig. 10 Correlations in Strikers population

Hennessey has the highest number of Saves Made and also an unusually high *ELD* score, although not the highest.

Midfielder Salary We omit a scatterplot for midfielder salary vs. *ELD* because it is less informative due to the weaker correlation (0.45). To investigate the reason for the weaker correlation, we picked two midfielders: 1) Stephane Sessegnon who has been ranked second in the *ELD* ranking but does not draw a large salary. 2) Steven Gerrard who is a very well known player and ranked second in the Salary ranking but according to the *ELD* score, he has been ranked 21. Based on domain knowledge, we picked some of the features that are relevant to midfield performance from the raw data and compared the feature statistics for these two players. Table 17 shows the details of their ap-



(a) Goalies: sum of time played vs *ELD*. (b) Goalies: sum of saves made vs *ELD*

Fig. 11

pearances in different matches. Sessegnon scored higher than Gerrard in three out of the four categories (passes and Time Played). However, his salary was much lower than Gerrard's. Indeed his next contract with West Bromwich Albion netted him a club record fee. This is an example of how our *ELD* metric can identify future stars.

Name	Team	age	Salary Ranking	*ELD* Ranking	Time Played	Unsuccessful Passes	Successful Long Passes	Successful corners
Steven Gerrard	Liverpool	31	2	21	1212 min	244	52	25
Stephane Sessegnon	Sunderland	26	22	2	3133 min	231	82	15

Table 17 Comparison of two midfielders.

10.2.3 Hockey Players

For the hockey players the features that we have selected as success metrics are as follows: *TimeonIce*, that can be defined as the total amount of time a player has played over the course of a season. *PowerPlayTime* which is the total amount of power player time for a given player in a season. *Assists* is the total number of any actions that enabled a goal. And finally total number of *Goals* that a given player has scored. The correlation between these features

Class	Time On Ice	Power Play Time	Assists	Goals
Forwarder	0.79	0.78	0.81	0.78
Defender	0.66	0.57	0.60	0.40

Table 18 Correlation between *ELD* metric and success metrics of NHL Players.

Subgroup	Success metric	*ELD* correlation	Log-likelihood correlation
Comedy	Sum of Rating	0.85	0.87
Drama	Sum of Rating	0.78	0.82
PL-Midfielder	Pass Efficiency	0.77	0.89
PL-Midfielder	Time Played	0.80	0.86
PL-Goalies	Time Played	0.77	0.87
PL-Goalies	Saves Made	0.74	0.85
NHL-Forward	Time on Ice	0.79	0.95
NHL-Forward	Goals	0.78	0.83

Table 19 Subgroups that Log-likelihood metric's correlation with success is stronger than *ELD*'s correlation between *ELD* with success.

and *ELD* is shown in Table 18. ELD has high correlations in features of both categories, however, it is stronger in Forwarder subgroup.

Movie Sum of Ratings

10.2.4 Movies

Movie Sum of Ratings is the number of user ratings of a movie. Table 16 shows a high correlation with the *ELD* metric. The highest correlation obtains for the genre Comedy (0.84). The correlation between a movie and the sum of its ratings is equally strong, but the correlation with its average rating is much weaker. Thus the *ELD* score is related mainly with how many users have rated the movie rather than with how they have rated it. The number of ratings is a meaningful success metric as it indicates the number of people who have gone to see a movie.

10.3 Correlations between other outlier metrics and success

In Section 3.4 and 6 we introduced other metrics that could be used in order to detect outliers. In this section we investigate the correlation between those metrics and success. The correlation between *ELD* and success is always stronger than *FD* and *LR* in all the datasets and subgroups, therefore we omit those results. The Log-likelihood is the only metric that results in a stronger correlation in some datasets and subgroups. Table 19 shows the subgroups that Log-likelihood has a stronger correlation than the *ELD* metric. In a way a high correlation with success can be interpreted as detecting in-class outliers and this results shows that Log-likelihood could be an alternative score to detect those type of outliers.

11 Conclusion and Future Work

We presented a new approach for applying Bayesian networks to object-relational outlier detection, a challenging and practically important topic for machine

learning. The key idea is to learn one set of parameter values that represent class-level associations, another set to represent object-level associations, and compare how well each parametrization fits the relational data that characterize the target object. The classic metric for comparing two parametrized models is their log-likelihood ratio; we refined this concept to define a new relational log-likelihood distance metric via two transformations: (1) a mutual information decomposition, and (2) replacing log-likelihood differences by log-likelihood distances. This metric combines a single feature component, where features are treated as independent, with a correlation component that measures the deviation in the features' mutual information.

In experiments on three synthetic and four real-world outlier sets, the log-likelihood distance achieved the best detection accuracy. The alternative of converting the structured data to a flat data matrix via aggregation had a negative impact. Case studies showed that the log-distance score leads to easily interpreted rankings. We found that the log-distance score correlated with success metrics to a surprising degree, across different domains and classes of individuals. The correlation with metrics of independent interest corroborates that the log-distance score produces meaningful and interesting results.

There are several avenues for future work. (i) A limitation of our current approach is that it ranks potential outliers, but does not set a threshold for a binary identification of outlier vs. non-outlier. (ii) Our divergence uses expected L1-distance for interpretability, but other distance scores like L2 could be investigated as well. (iii) Extending the expected L1-distance for continuous features is a useful addition. (iv) Compare our metric with the interestingness measures that have been developed for relational exception mining. (v) In the movie and soccer domains, our metric identified exceptionally successful individual objects, but not exceptionally unsuccessful ones. Our hypothesis was that in these domains, individuals have gone through a rigorous selection process, so the normal baseline performance is high. While we provided evidence for this hypothesis, it can be further investigated, by applying our outlier detection to datasets that feature a range of skills, rather than professional performance.

In sum, outlier metrics based on model likelihoods are a new type of structured outlier score for object-relational data. Our evaluation indicates that this model-based score provides informative, interpretable, and accurate rankings of objects as potential outliers.

References

1. E. Achtert, H. Kriegel, E. Schubert, and A. Zimek. Interactive data mining with 3d-parallel coordinate trees. In *Proceedings of the 2013 ACM SIGMOD*, New York, NY, USA, 2013.
2. C. Aggarwal. *Outlier Analysis*. Springer New York, 2013.
3. L. Akoglu, M. McGlohon, and C. Faloutsos. Oddball: Spotting anomalies in weighted graphs. In *Proceedings of the 14th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining - Volume Part II*, PAKDD'10, pages 410–421, Berlin, Heidelberg, 2010. Springer-Verlag.

4. L. Akoglu, H. Tong, and D. Koutra. Graph based anomaly detection and description: A survey. *Data Min. Knowl. Discov.*, 29(3), May 2015.
5. G. Anderson and B. Pfahringer. Exploiting propositionalization based on random relational rules for semi-supervised learning. In *Advances in Knowledge Discovery and Data Mining, 12th Pacific-Asia Conference, PAKDD 2008, Osaka, Japan, May 20-23, 2008 Proceedings*, pages 494–502, 2008.
6. F. Angiulli, G. Greco, and L. Palopoli. Outlier detection by logic programming. *ACM Transactions on Computer Logic*, 2004.
7. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. Lof: Identifying density-based local outliers. In *Proceedings of ACM SIGMOD*, 2000.
8. A. Cansado and A. Soto. Unsupervised anomaly detection in large databases using Bayes Nets. *Appllied Artificial Intelligence*, 2008.
9. L. de Campos. A scoring function for learning Bayes nets based on mutual information and conditional independence tests. *Journal of Machine learning Research*, 2006.
10. P. G. del Barrio and F. Pujol. Pay and Performance in the Spanish Soccer League: Who Gets the Expected Monopsony Rents? Faculty Working Papers 05/04, School of Economics and Business Administration, University of Navarra, Mar. 2004.
11. P. Domingos and D. Lowd. *Markov Logic: An Interface Layer for Artificial Intelligence*. Morgan and Claypool Publishers, 2009.
12. O. S. Fatemeh Riahi. Propositionalization for unsupervised outlier detection in multi-relational data. In *The Florida Association of Artificial Intelligence*, 2016.
13. T. Fawcett. An introduction to ROC analysis. *Pattern Recogn. Lett.*, 2006.
14. R. A. Fisher. On the probable error of a coefficient of correlation deduced from a small sample. *Metron*, 1:3–32, 1921.
15. J. Gao, F. Liang, W. Fan, Y. Wang, and J. Han. On community outliers and their detection in information network. In *Proceedings of ACM SIGKDD*, 2010.
16. L. Getoor. *Learning Statistical Models From Relational Data*. PhD thesis, Department of Computer Science, Stanford University, 2001.
17. L. Getoor and B. Taskar. *Introduction to statistical relational learning*. MIT Press, 2007.
18. S. Hall, S. Szymanski, and A. S. Zimbalist. Testing causality between team performance and payroll: The cases of major league baseball and english soccer. *Journal of Sports Economics*, 3(2):149–168, 2002.
19. D. Heckerman, C. Meek, and D. Koller. Probabilistic entity-relationship models, PRMs, and plate models. In Getoor and Taskar [17].
20. T. Horváth, Z. Alexin, T. Gyimóthy, and S. Wrobel. *Application of Different Learning Methods to Hungarian Part-of-Speech Tagging*, pages 128–139. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.
21. T. Horváth, S. Wrobel, and U. Bohnebeck. Relational instance-based learning with lists and terms. *Machine Learning*, 43(1):53–80, 2001.
22. H. Khosravi, T. Man, J. Hu, E. Gao, and O. Schulte. Learn and join algorithm code. [Online]. Available: URL = `http://www.cs.sfu.ca/~oschulte/jbn/`.
23. T. Khot, S. Natarajan, and J. Shavlik. Relational one-class classification: A non-parametric approach. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, pages 2453–2459. AAAI Press, 2014.
24. A. Kimmig, L. Mihalkova, and L. Getoor. Lifted graphical models: a survey. *Computing Research Repository*, 2014.
25. M. Kirsten, S. Wrobel, and T. Horváth. *Distance Based Approaches to Relational Learning and Clustering*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.
26. J. L. Koh, M. L. Lee, W. Hsu, and W. T. Ang. Correlation-based attribute outlier detection in XML. In *Proceedings of ICDE. IEEE 24th*, 2008.
27. D. Koller and A. Pfeffer. Object-oriented Bayes nets. In *Proceedings of UAI*, 1997.
28. S. Kramer, N. Lavrac, and P. Flach. Propositionalization approaches to relational data mining. In *Relational Data Mining*. Springer, 2000.
29. O. Kuvzelka and F. Zelezny. Hifi: Tractable propositionalization through hierarchical feature construction. In *Late Breaking Papers, ILP*, 2008.
30. J. Maervoet, C. Vens, G. Vanden Berghe, H. Blockeel, and P. De Causmaecker. Outlier detection in relational data: A case study. *Expert System Applications*, 2012.

31. E. Muller, I. Assent, P. Iglesias, Y. Mulle, and K. Bohm. Outlier ranking via subspace analysis in multiple views of the data. In *Proceedings of ICDM*, 2012.
32. M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2016.
33. P. K. Novak, G. I. Webb, and S. Wrobel. Supervised descriptive rule discovery: A unifying survey of contrast set, emerging pattern and subgroup mining. *Journal of Machine Learning Research*, 2009.
34. J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, 1988.
35. V. Peralta. Extraction and Integration of MovieLens and IMDb. Technical report, APDM project, 2007.
36. M. Perovsek, A. Vavpetic, B. Cestnik, and N. Lavrac. A wordification approach to relational data mining. In *Discovery Science*, Lecture Notes in Computer Science. Springer, 2013.
37. D. Poole. First-order probabilistic inference. In *Proceedings of IJCAI*, 2003.
38. S. Ramaswamy, R. Rastogi, and K. Shim. Efficient algorithms for mining outliers from large data sets. *SIGMOD*, 2000.
39. F. Riahi and O. Schulte. Codes and Datasets. [Online]. Available:. `ftp://ftp.fas.sfu.ca/pub/cs/oschulte/CodesAndDatasets/`, 2015.
40. F. Riahi and O. Schulte. Model-based outlier detection for object-relational data. In *IEEE Symposium Series on Computational Intelligence (SSCI*, 2015.
41. S. Riedel, L. Yao, A. McCallum, and B. M. Marlin. Relation extraction with matrix factorization and universal schemas. In *Human Language Technologies: NAA*, pages 74–84, 2013.
42. S. Sarawagi, R. Agrawal, and N. Megiddo. Discovery-driven exploration of OLAP data cubes. In *Proceedings of International Conference on Extending Database Technology*. Springer-Verlag, 1998.
43. O. Schulte. A tractable pseudo-likelihood function for Bayes nets applied to relational data. In *Proceedings of SIAM SDM*, 2011.
44. O. Schulte and S. Gholami. Locally consistent Bayesian network scores for multi-relational data. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2693–2700, 2017.
45. O. Schulte and H. Khosravi. Learning graphical models for relational data via lattice search. *Journal of Machine Learning*, 2012.
46. O. Schulte, H. Khosravi, A. Kirkpatrick, T. Gao, and Y. Zhu. Modelling relational statistics with bayes nets. *Machine Learning*, 94:105–125, 2014.
47. O. Schulte and K. Routley. Aggregating predictions vs. aggregating features for relational classification. In *Computational Intelligence and Data Mining (CIDM), 2014 IEEE Symposium on*, pages 121–128. IEEE, 2014.
48. T. Sing, O. Sander, N. Beerenwinkel, and T. Lengauer. *ROCR: Visualizing the performance of scoring classifiers.*, 2012. R package version 1.0-4.
49. Y. Sun, H. Jiawei, and P. Zhao. Rankclus: Integrating clustering with ranking for heterogeneous information network analysis. In *Proceedings of the International Conference on Extending Database Technology: Advances in Database Technology*, pages 565–576, New York, NY, USA, 2009. ACM.
50. G. Tang, J. Bailey, J. Pei, and G. Dong. Mining multidimensional contextual outliers from categorical relational data. In *Proceedings of SSDBM*, 2013.
51. M. P. T.Swartz, Adriano Arce. Assessing value of the draft positions in major league soccer's superdraft. *The Sport Journal*, 2013.
52. S. Tuffery. *Data Mining and Statistics for Decision Making*. Wiley Series in Computational Statistics, 2011.