# Prediction-Based Outlier Detection with Explanations

Liang-Chieh Chen[1] Tsung-Ting Kuo[1] Wei-Chi Lai[1] Shou-De Lin[1] Chi-Hung Tsai[2]
[1]Department of Computer Science and Information Engineering, National Taiwan University, Taiwan
[2]Institute for Information Industry, Taiwan
aquariusjay@gmail.com d97944007@csie.ntu.edu.tw d96922031@ntu.edu.tw sdlin@csie.ntu.edu.tw brick@iii.org.tw

## Abstract

*General outlier detection strategies, be a distribution-based, clustering-based, or distance-based method, all resort to the comparison among instances to define abnormality. In this paper we introduce an additional dimension into the outlier definition. That is, we not only consider externally how one instance differs from others but internally the dependency and abnormality among its own attributes, denoted as the prediction-based outlier detection. Prediction-based outliers possess certain attributes which are difficult to be predicted based on the neighborhood information. Furthermore, we propose three neighborhood functions to generate predictions. Finally, acknowledging the lack of the gold standard to evaluate an outlier detection system, we propose four general evaluation strategies. Experiments conducted on several real-world datasets demonstrate the validity, novelty, power-law distribution, and robustness of our method.*

## 1. Introduction

Outlier detection aims at finding data points which are abnormal according to certain measures of normality. This is an important problem which includes a wide variety of applications such as fraud detection, homeland security, and disease discovery. Studies of outlier detection can generally be divided into three categories: distribution-based [1][25], cluster-based [5][6][8][14], and distance-based [2][9][10][11][19] approaches. Those methods, nevertheless, define an outlier based on some kind of comparison between it and other instances, but do not pay too much attention to the internal inconsistency among attributes. This paper tries to explicitly incorporate this dimension into account by considering the prediction-based outliers as instances that possess certain unpredictable attributes. Our experiments show that by adopting this additional dimension into account, we are able to find some outliers that can hardly be detected otherwise.

The major observation and assumption made here is that in the real world, certain attribute of a data point has some correlation with others. Therefore, some objects possess certain attributes whose values cannot be accurately predicted based on other attributes together with other similar objects. Other outlier detection methods can hardly achieve such a purpose since their emphasis is not on one single attribute (or a small subset of them), and such divergence can easily be overlooked when all attributes are considered at the same time.

Our second observation is that prediction-based outliers heavily depend on the definition of neighbors. Using neighbors to determine outliers is by no means novel, because the majority of the distance-based outlier methods find outliers as those significantly *different* from the neighbors. The novelty of our proposal, however, lies in the different exploitation of those neighbors, since we instead use them as the basis to *predict* a certain attribute.

Our principle of outliers naturally facilitates the generation of explanation. Such an explanation allows the users to judge the validity of the results and data. Being able to generate explanations is important for many outlier applications in domains like security [13].

To achieve the above goals, we exploit LOESS regression [3], which is a local linear regression method that builds a model for each data point based on its neighboring points. We suggest modifying the way neighbors are chosen in LOESS by integrating three types of neighborhood functions (numerical-based, categorical-based, and social-based neighborhood functions), which further provides a mechanism to deal with heterogeneous data. Note that our prediction-based outlier detection is indeed an unsupervised strategy because no human-labeled training outlier data is required. The regression model is utilized to learn the dependency between attributes of data rather than fitting training inputs of outlierness.

The major contributions of this paper can be summarized as follows:

1. We propose an idea of prediction-based outlier detection that utilizes learning approaches to discover instances with internal inconsistency, which can sometimes be overlooked by other outlier detection algorithms. This is a domain-independent, unsupervised approach for outlier detection.
2. We propose to integrate the ideas of numerical, categorical, and social neighbors into a LOESS regression model to determine the outliers. This enables us to deal with data with heterogeneous attributes. Furthermore, we propose to normalize the outlier factor (i.e. a real number that represents an object's degree of being an outlier) using the predictability of regression to improve its robustness against independent attributes.
3. We propose an explanation framework that is capable of producing natural language explanations of outliers for advanced verification.

4. To address the lack-of-gold-standard drawback for outlier detecting research, this paper proposes four strategies to evaluate an outlier detection system with the aim of providing a more general, systematic evaluation guideline.

## 2. Related Work

Previous studies of outlier detection can generally be divided into three categories: distribution-based, cluster-based, and distance-based outliers. In the area of statistics, researchers define distribution-based outlier [1] as instances that deviates significantly from a given distribution. The problem with distribution-based outlier detection is that there is no guarantee that the underlying data distribution is accessible or learnable, particularly in the case of heterogeneous datasets.

Cluster-based outlier detection finds clusters first, and then declares data points that do not belong to any cluster as outliers [5][6][8][14]. The drawbacks for cluster-based outlier are twofold. First, the computation for clustering is generally more expensive. Second, the idea may fail if normal data points do not form clusters.

To deal with these drawbacks, distance-based outlier detection is proposed [2][9][11][19][25]. One apparent deficiency of the distance-based outliers is that generally all attributes are utilized in the analysis while some interesting local variations between attributes are neglected. To handle such a deficiency, we predict each individual attribute independently using the other attributes rather than the condensing information of all dimensions into one single distance value.

Some outlier detection methods have been designed for graphs in which data instances are connected to each other [4][16][21]. These methods, however, focus on finding outliers given certain graph structure, and fail to consider the attributes of instances.

Other studies focus on outlier explanations [10][12][13][23][24]. Different from most of the above mechanisms that consider explanation generation as finding a subset of attributes that still allow the distinguishing of a given outlier from others, our explanation mechanism directly reflect the outlier discovery process. Furthermore, most of the above methods are offline and computationally expensive, while our approach produces the explanation on the spot with only constant complexity.

## 3. Prediction-Based Outlier Detection

An outlier is used to be defined as an object which appears to be inconsistent by a specific metric with the rest of the dataset. Such principle suffers the drawback of neglecting the dependency between attributes. We argue another kind of outliers exists, which we refer to as prediction-based outliers. That is, objects that possess certain attributes that can hardly be predicted by neighbors. In other words, we look for outliers that display the internal inconsistency rather than considering inconsistency with the rest of the world. We assume a dataset can be heterogeneous, meaning it might contain both numerical and categorical attributes. Moreover, there might even exist some explicit connections between objects.

The flow of the proposed algorithm is as follows. Assuming there are n data points $\{p_1, p_2, \ldots, p_n\}$ in the dataset, our goal is to learn whether a point $p_i=\{p_{i1}, p_{i2},\ldots p_{im}\}$ is a prediction-based outlier, where $p_{ik}$ represents the k-th attribute of $p_i$. Suppose it is possible to construct a neighborhood function $f(p_i)$ which returns a set of $p_i$'s neighboring instances N, we propose then to use the instances in N to construct m regression models $R_1 \ldots R_m$, each of which tries to predict the value of one attribute using the rest of the m-1 attributes. The learned model $R_k$ is then used to predict the $p_{ik}$ value. Eventually $p_i$ is regarded as an outlier in the k-th attribute if it turns out the prediction of the k-th attribute of $p_i$ is significantly different from its true value. Such principle is naturally suitable for explanation generation since it identifies not only outliers but also one specific deviated attribute along with its residual (i.e. how far it is away from the predicted value).

### 3.1 Neighborhood generation functions

To identify the basis of prediction, we propose three types of neighbors: numerical neighbors ($N_n$), categorical neighbors ($N_c$) and social neighbors ($N_s$). For numerical data, $N_n$ can be generated simply by using the k-nearest neighbor algorithm. For data with categorical features, $N_c$ can be obtained by identifying objects with identical categorical feature values. If the data are connected through a social network or a graph, $N_s$ can be defined by using the conventional community detection algorithm [15] or role-based clustering algorithm [22].

### 3.2 LOESS regression for outlier detection

Once the neighbors of a given instance $p_j$ are identified, they can be exploited as the training samples to predict attribute $p_{jk}$ of $p_j$. Point $p_j$ is regarded as an outlier with respect to attribute k if the predicted value, $y_{ik}$, is far from $p_{jk}$.

The LOESS regression model [3] is suitable for such a purpose. LOESS regression is a locally weighted linear regression model. We suggest that the neighborhood function of LOESS can be modified to incorporate heterogeneous types of attributes. In addition to using the numerical x to determine the neighbors, it is possible to utilize other two kinds of neighborhood functions to identify different kinds of outliers. Note that different from the distance-based or density-based outlier detection methods, our concept of outliers does not necessary look for *external inconsistency* between a point and its neighbors. Instead, our method looks for the *internal inconsistency* of

the attributes, while the neighbors are chosen so that we can learn the dependency of attributes, not for comparison.

We describe a major concern of our idea and propose a solution. Based on the above proposal, LOESS determines that in Figure 1, $p_1$ and $p_2$ are equally abnormal since their projected distances to the regression line are the same. However, $p_2$ should look more like an outlier since in Figure 1(a), the attribute x can hardly be used to predict y.

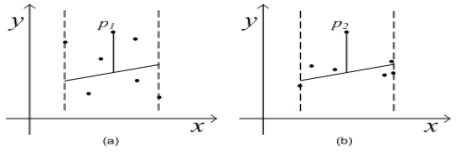


Figure 1. Outliers with different Mean-Square-Errors.

This means that to improve the soundness of our model, we need to take the *predictability* of an attribute into account. A non-predictable feature should not be considered as an indicator for outlierness. In this sense, we propose a prediction-based outlier factor (PBOF) as

$$\mathrm{PBOF}(p_j) = \frac{|\hat{y}_j - y_j|}{\sqrt{\mathrm{WMSE}}}$$

where WMSE is the *weighted mean square error* of training in the regression and the weight is identical to the elements of W used in the LOESS regression (i.e. the penalty on the error of the closer points is larger). The absolute residual, $|\hat{y}_j - y_j|$, measures the unpredictability of the point $p_j$, while WMSE captures the unpredictability of the neighborhood. Then in Figure 1, it becomes apparent that the PBOF of $p_2$ is higher than that of $p_1$. One can infer that attributes that are independent of others can hardly be considered as an abnormal attribute for any point in our framework since their WMSE is generally very high. Finally, our system ranks all point-attribute combination based on this outlier factor and returns the top ones as outliers.

## 3.3 Outlier explanation

Outlier explanation attempts to generate a reasonable and understandable (usually in natural language) explanation to describe why a system considers a particular data point to be an outlier. Automatic explanation plays an important role in many intelligent systems [7][18][20]. We argue that explanation is even more important in anomaly detection, which usually targets security-related tasks with less tolerance for margin of error. One advantage of a prediction-based outlier lies in the ease of explanation generation as the discovery process of outliers can be elaborated easily. Based on LOESS regression, a data point is an outlier if its true value of certain attribute is significantly different from the predicted one, which is generated by a model learned from its neighbors. Therefore, we design a natural language explanation template as follows (note that the explanations for neighbors can undergo domain-specific modifications such as changing "connected to it" to "teammates" to improve readability):

| |
|---|
| "<$p_i$> is an outlier in <Attribute$_i$> (<value of $y_i$>) because given <explanations for neighbors>, he is predicted to have <$\lvert \hat{y}_i - y_i \rvert$> <higher/lower> <Attribue$_i$>." |
| <Explanations for neighbors><br>For numerical-based neighbors,<br>*" instances who have similar <numerical attributes> (e.g. <sample neighbors>)"*<br>For categorical-based neighbors,<br>*"instances of the same <categorical attributes> (e.g. <sample neighbors>)"*<br>For social-based neighbors<br>*"instances connected to it (e.g. <sample neighbors>)"* |

Therefore, each outlier has its own "type", which means it is an outlier with respect to a specific attribute given certain neighbors. Moreover, the explanation points out that this outlier is expected to have higher or lower values in this particular attribute. Finally, the learned least-square weight vector offers even deeper explanations for advanced users. Note that different from many other outlier explanation algorithms such as [10][12][13] where offline computation (and notable amount of additional time) is required for explanation generation, our explanation can seamlessly reflect the discover process and can be acquired at the same time when the outlier factor is generated, without having to impose non-constant time and space complexity.

# 4. Experiments

Generally, there is no gold standard for outlier detections and researchers have proposed different scenarios to define abnormality. The lack of a gold standard, unfortunately, imposes significant difficulty for outlier evaluation. In most of the previous works introduced in Section 2, researchers generally adopted an evaluation policy that performs outlier detection on well-known dataset and explains the legitimacy behind the outputs. However, we argue that there should be at least three additional dimensions that are worthy of examining. Therefore, below we propose four strategies to evaluate an outlier system:

(1) Explaining the validity of results: this is a strategy adopted in conventional studies of outlier detection [2][9][10][11][17][19]. To do so, experiments have to be conducted on real-world datasets and the rationale behind the outputs has to be explained (usually by the authors). In fact, the explanations generated by our system target exactly at automatizing such purpose. This allows users to make quicker and more accurate decision about the validity of results while avoiding the biases from the manually produced explanations.

(2) Evaluating novelty: We propose to compare the approach with other well-known methods, and argue that a designed outlier detection algorithm is novel if it finds some meaningful outliers that can hardly be detected by other methods.

(3) Evaluating the distribution of the output factor: an outlier factor is a real number that represents the level of outlierness of an instance. Since abnormality is more likely to be a relative comparison than simply a binary decision (i.e. abnormal or not), outlier factor is generally considered as a more reasonable output of an outlier detecting algorithm. Using the outlier factor of every single instance, it is possible to generate a histogram of the outlier factor. Here, we suggest that a power-law histogram is preferable for outlier detection since people generally expect most of the instances are normal with only very few exceptions (i.e. long tail).

(4) Evaluating the robustness of the system: an outlier detection method should be robust in the presence of noise. We demonstrate that by taking the predictability into account, it is possible to avoid the interference from the manually added independent attributes.

## 4.1 Explaining the Validity of the Results

The proposed algorithm is evaluated in two real-world datasets, namely a National Basketball Association (NBA) dataset and Major League Baseball (MLB) dataset. Note that the NBA dataset has been utilized for outlier evaluation by several papers [17][19]. We pick some representative outliers for demonstration, and describe the rationale behind them. We also show that the learned prediction function and the generated explanation can assist users in examining the outliers. We conduct scaling of data to make sure each attribute is treated equally in the analysis.

TABLE I. NBA DATASET: OUTLIERS FOR NN, NC AND NS

| Att. | Rank | Numerical Neighbors | | | | Categorical Neighbors | | | | Social Neighbors | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Name | Actual Value | Expected Value | PBOF | Name | Actual Value | Expected Value | PBOF | Name | Actual Value | Expected Value | PBOF |
| Pts | 1 | Marcus Camby | 9.13 | 23.45 | 8.48 | Jason Kidd | 10.80 | 28.15 | 6.21 | Jason Kidd | 10.80 | 34.86 | 6.25 |
| | 2 | Jason Kidd | 10.80 | 26.52 | 7.48 | Chris Paul | 21.05 | 31.70 | 5.87 | Marcus Camby | 9.13 | 36.68 | 5.06 |
| | 3 | Amare Stoudemire | 25.18 | 12.87 | 6.40 | Marcus Camby | 9.13 | 30.79 | 5.74 | Chris Paul | 21.05 | 33.19 | 4.91 |
| Reb | 1 | Jason Kidd | 7.50 | 3.10 | 8.92 | Jason Kidd | 7.50 | 3.48 | 6.91 | Jason Kidd | 7.50 | 3.23 | 6.26 |
| | 2 | Joel Przybilla | 8.42 | 3.82 | 8.50 | Chris Paul | 4.01 | 2.31 | 4.86 | Chris Paul | 4.01 | -2.78 | 4.96 |
| | 3 | Reggie Evans | 7.54 | 2.94 | 7.45 | Allen Iverson | 2.96 | 4.91 | 4.80 | Tyson Chandler | 11.75 | 6.73 | 4.90 |
| Ast | 1 | Jason Kidd | 10.08 | 3.35 | 8.66 | Steve Nash | 11.07 | 3.48 | 5.62 | Jason Kidd | 10.08 | 4.47 | 6.00 |
| | 2 | Steve Nash | 11.07 | 3.31 | 8.32 | Jason Kidd | 10.08 | 4.18 | 5.37 | Marcus Camby | 3.28 | -5.52 | 5.13 |
| | 3 | Marcus Camby | 3.28 | 1.38 | 6.58 | Brad Miller | 3.67 | 1.83 | 4.82 | Chris Paul | 11.56 | 2.88 | 5.07 |
| Blk | 1 | Josh Smith | 2.80 | 0.95 | 7.50 | Josh Smith | 2.80 | 0.80 | 7.53 | Jason Kidd | 0.33 | 0.85 | 5.44 |
| | 2 | Marcus Camby | 3.61 | 1.65 | 7.40 | Chris Paul | 0.05 | 0.50 | 6.45 | Marcus Camby | 3.61 | 0.39 | 5.06 |
| | 3 | Andrei Kirilenko | 1.51 | 0.52 | 7.14 | Marcus Camby | 3.61 | 1.12 | 5.55 | Jermaine O'Neal | 2.07 | 0.33 | 4.91 |
| Stl | 1 | Ronnie Brewer | 1.70 | 0.69 | 7.50 | Chris Paul | 2.71 | 1.08 | 7.30 | Shawn Marion | 1.98 | 0.86 | 5.83 |
| | 2 | Chris Paul | 2.71 | 1.51 | 7.20 | Ron Artest | 2.33 | 1.13 | 5.95 | Ron Artest | 2.33 | 1.18 | 5.57 |
| | 3 | Anfernee Hardaway | 1.19 | 0.55 | 6.98 | Marcus Camby | 1.06 | 0.38 | 5.40 | Marcus Camby | 1.06 | 2.74 | 5.09 |

### *4.1.1 NBA dataset*

We collected the NBA data from the 2007-2008 regular season. There are 449 players with 5 numerical features and 2 categorical features in the dataset. The numerical features include Pts, Reb, Ast, Blk, and Stl. The categorical features include Position and Division, and thus the categorical neighbors are defined based on these two variables. The social network of NBA players is a bipartite graph that contains two types of nodes: team and player. The graph is constructed using the data from the 2005-2008 regular seasons. A player is connected to a team if he has played for that team during that period. The two-step distance is used to determine the social neighbors. The top 3 outliers identified based on numerical neighbors ($N_n$), categorical neighbors ($N_c$), and social neighbors ($N_s$) for each attribute are shown in Table 1.

### *4.1.2 MLB dataset*

The MLB dataset contains 542 players in the 2008 regular season with 4 numerical features and 2 categorical features. The numerical features are HR (Homeruns), BB (Base on Balls), SO (Strike Out), and SB (Stolen Bases). The categorical features are Position (Infielder, Outfielder, and Catcher), and Division (American League and National League). The social network is based on the relationships of teammates from 2007 to 2008. As shown in Table 2, our approach can not only identifies apparent outliers, such as leaders in certain attributes, but also finds some interesting and less apparent outliers.

TABLE II. MLB DATASET: OUTLIERS FOR NN, NC, AND NS

| Att. | Rank | Numerical Neighbors | | | | Categorical Neighbors | | | | Social Neighbors | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Name | Actual Value | Expected Value | PBOF | Name | Actual Value | Expected Value | PBOF | Name | Actual Value | Expected Value | PBOF |
| HR | 1 | A. Ramirez | 21 | 5.88 | 7.04 | Albert Pujols | 37 | 17.90 | 5.79 | Jose Reyes | 16 | 2.57 | 4.75 |
| | 2 | Carlos Lee | 28 | 7.51 | 6.81 | Gregor Blanco | 1 | 22.32 | 5.01 | Willy Taveras | 1 | 22.42 | 4.74 |
| | 3 | Albert Pujols | 37 | 17.22 | 5.82 | Joe Mauer | 9 | 0.36 | 4.76 | Albert Pujols | 37 | 12.43 | 4.63 |
| BB | 1 | Ryan Theriot | 73 | 35.73 | 7.55 | B.J. Upton | 97 | 44.38 | 6.33 | Adam Dunn | 122 | 77.62 | 5.00 |
| | 2 | Joe Mauer | 84 | 26.39 | 6.99 | Brian Giles | 87 | 27.53 | 4.74 | Albert Pujols | 104 | -66.08 | 4.84 |
| | 3 | Brian Giles | 87 | 28.76 | 6.97 | Carlos Gomez | 25 | 65.91 | 4.67 | C. Delgado | 72 | 101.98 | 4.70 |
| SO | 1 | Carlos Gomez | 142 | 62.59 | 9.67 | Albert Pujols | 54 | 146.51 | 6.50 | Carlos Gomez | 142 | 55.13 | 5.17 |
| | 2 | Albert Pujols | 54 | 144.28 | 6.10 | Mark Reynolds | 204 | 113.53 | 6.01 | Jose Reyes | 82 | 168.30 | 4.89 |
| | 3 | C. Gonzalez | 81 | 36.71 | 5.90 | Jack Cust | 197 | 135.25 | 5.52 | Albert Pujols | 54 | 154.93 | 4.82 |
| SB | 1 | Willy Taveras | 68 | 10.28 | 10.58 | Jose Reyes | 56 | 9.33 | 7.84 | Rajai Davis | 29 | 1.24 | 5.04 |
| | 2 | Juan Pierre | 40 | 4.24 | 10.52 | R. Martin | 18 | 4.26 | 6.45 | Albert Pujols | 7 | -13.00 | 4.71 |
| | 3 | Rajai Davis | 29 | 2.01 | 9.62 | Jimmy Rollins | 47 | 8.50 | 6.24 | H. Ramirez | 35 | 0.08 | 4.64 |

TABLE III. OUTLIERS IN NBA DATASET BASED ON NN , NC, AND NS AND THEIR RANKS IN KNN AND LOF

| Rank | Numerical Neighbors | | | | | Categorical Neighbors | | | | | Social Neighbors | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Name | PBOF | Att. | KNN Rank | LOF Rank | Name | PBOF | Att. | KNN Rank | LOF Rank | Name | PBOF | Att. | KNN Rank | LOF Rank |
| 1 | Jason Kidd | 8.92 | Reb | 3 | 4 | Josh Smith | 7.53 | Blk | 4 | 13 | Jason Kidd | 6.26 | Reb | 3 | 4 |
| 2 | Joel Przybilla | 8.50 | Reb | 38 | 33 | Chris Paul | 7.30 | Stl | 2 | 2 | Shawn Marion | 5.83 | Stl | 7 | 36 |
| 3 | Marcus Camby | 8.48 | Pts | 1 | 1 | Jason Kidd | 6.91 | Reb | 3 | 4 | Ron Artest | 5.57 | Stl | 13 | 27 |
| 4 | Steve Nash | 8.32 | Ast | 6 | 3 | Ron Artest | 5.95 | Stl | 13 | 27 | Marcus Camby | 5.13 | Ast | 1 | 1 |
| 5 | Ronnie Brewer | 7.50 | Stl | 25 | 18 | Marcus Camby | 5.74 | Pts | 1 | 1 | Chris Paul | 5.07 | Ast | 2 | 2 |
| 6 | Josh Smith | 7.50 | Blk | 4 | 13 | Steve Nash | 5.62 | Ast | 6 | 3 | Jermaine O'Neal | 4.91 | Blk | 36 | 42 |
| 7 | Reggie Evans | 7.45 | Reb | 42 | 15 | Shawn Marion | 5.35 | Stl | 7 | 36 | Tyson Chandler | 4.90 | Reb | 44 | 61 |
| 8 | Chris Paul | 7.20 | Stl | 2 | 2 | Allen Iverson | 5.34 | Pts | 8 | 21 | Yao Ming | 4.86 | Blk | 32 | 29 |
| 9 | Andrei Kirilenko | 7.14 | Blk | 20 | 17 | Amare Stoudemire | 5.07 | Pts | 14 | 31 | Dwyane Wade | 4.86 | Pts | 18 | 56 |
| 10 | Ben Wallace | 7.07 | Reb | 15 | 43 | Brad Miller | 4.82 | Ast | 31 | 133 | Baron Davis | 4.85 | Pts | 10 | 12 |

TABLE IV. OUTLIERS IN MLB BASED ON NN, NC, AND NS AND THEIR RANKS IN KNN AND LOF

| Rank | Numerical Neighbors | | | | | Categorical Neighbors | | | | | Social Neighbors | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Name | PBOF | Att. | KNN Rank | LOF Rank | Name | PBOF | Att. | KNN Rank | LOF Rank | Name | PBOF | Att. | KNN Rank | LOF Rank |
| 1 | Willy Taveras | 10.58 | SB | 2 | 2 | Jose Reyes | 7.84 | SB | 7 | 7 | Carlos Gomez | 5.17 | SO | 11 | 5 |
| 2 | Juan Pierre | 10.52 | SB | 9 | 1 | Albert Pujols | 6.50 | SO | 5 | 25 | Rajai Davis | 5.04 | SB | 28 | 3 |
| 3 | Carlos Gomez | 9.67 | SO | 11 | 5 | R. Martin | 6.45 | SB | 20 | 72 | Adam Dunn | 5.00 | BB | 6 | 15 |
| 4 | Rajai Davis | 9.62 | SB | 28 | 3 | B.J. Upton | 6.33 | BB | 1 | 11 | Jose Reyes | 4.89 | SO | 7 | 7 |
| 5 | Ryan Theriot | 7.55 | BB | 25 | 18 | Jimmy Rollins | 6.24 | SB | 14 | 13 | Albert Pujols | 4.84 | BB | 5 | 25 |
| 6 | A. Ramirez | 7.04 | HR | 56 | 35 | Willy Taveras | 6.24 | SB | 2 | 2 | Willy Taveras | 4.74 | HR | 2 | 2 |
| 7 | Joe Mauer | 6.99 | BB | 23 | 21 | Mark Reynolds | 6.01 | SO | 4 | 33 | C. Delgado | 4.70 | BB | 101 | 119 |
| 8 | Brian Giles | 6.97 | BB | 26 | 26 | Brian Roberts | 5.80 | SB | 10 | 24 | Lance Berkman | 4.70 | BB | 19 | 110 |
| 9 | Carlos Lee | 6.81 | HR | 45 | 48 | Jack Cust | 5.52 | SO | 13 | 14 | Brian Giles | 4.69 | BB | 26 | 26 |
| 10 | Jose Reyes | 6.71 | SB | 7 | 7 | G. Sizemore | 5.29 | SB | 8 | 28 | H. Ramirez | 4.65 | SO | 15 | 43 |

## 4.2 Novelty Analysis

To demonstrate the novelty of our system, in this section we focus on evaluating whether the proposed algorithm is

capable of detecting outliers that are hardly found by other methods. We compare our method with two popular methods: the k-nearest distance (KNN) outlier [19] and local outliers [2]. In the KNN outlier detection algorithm, the outlier factor is determined based on the Euclidean distance to the k-nearest neighbor (k=5 in our experiments). For local outliers, the local outlier factor LOF is captured as the ratio of density between the point itself and its neighbors, in which the number of neighbors is set to 50 in our experiments. If the local density of a point is significantly lower than that of its neighbors, it is likely to be an outlier. Note that these two algorithms produce an outlier factor for each player, while our outlier factor is for a certain attribute of each player. To enable comparison, we define the outlier factor of one player as the highest PBOF among all attributes, and then rank the players based on their highest PBOF.

### *4.2.1 NBA dataset*

The outlier rankings of players based on $N_n$, $N_c$, and $N_s$ are shown in Table 3, respectively. The KNN and LOF rankings of those outliers are also displayed for comparison. It is interesting to learn that although many of the outliers are universal (i.e. identified by all three methods), there are still some prediction-based outliers that can hardly be found by other algorithms.

### *4.2.2 MLB dataset*

As shown in Table 4, some prediction-based outliers in the MLB dataset that were not ranked high in KNN and LOF, such as A. Ramirez and Carlos Lee. can hardly be detected by KNN and LOF.

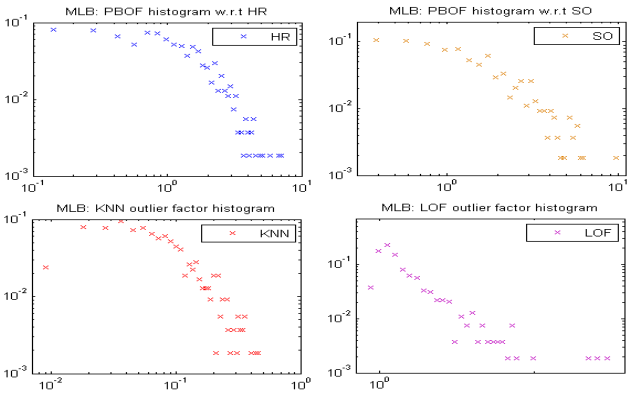


Figure 2. The log-log histograms for different outlier algorithms in MLB dataset. The uppers are histograms for PBOF and the lowers are that of KNN and LOF. The circled part represents the portion against power-law.

## 4.3 Power law distribution

The outlier factor is a real number that captures how abnormal an object is. By assigning each instance an outlier factor, it is possible to generate a histogram for the outlier factor. In the histogram, the x-axis represents the outlier factor, and the y-axis represents the percentage of instances of that value (as shown in Figure 2). Here we suggest that it is preferable for the histogram for any unsupervised outlier detection algorithm to follow power law distribution, which exhibits the property that as the outlier factor increases, the frequency of occurrence decreases at a greater scale. We believe maintaining the sparseness of outliers is reasonable because if the majorities become outliers, those outliers are not "against-normal" anymore, and furthermore, the demand for an accurate outlier detector would go down.

One interesting observation is that PBOF outlier factors roughly follow power law distribution in Figure 2. There is a long tail in which very few instances are extremely abnormal and the majority of the instances have very low outlier factors. KNN and LOF also follow power-law with one exception: the lowest-value instances (as circled in Figure 2) do not occupy a significant amount of the population. This implies the so-called "normal" instance discovered by KNN or LOF is indeed not that common.

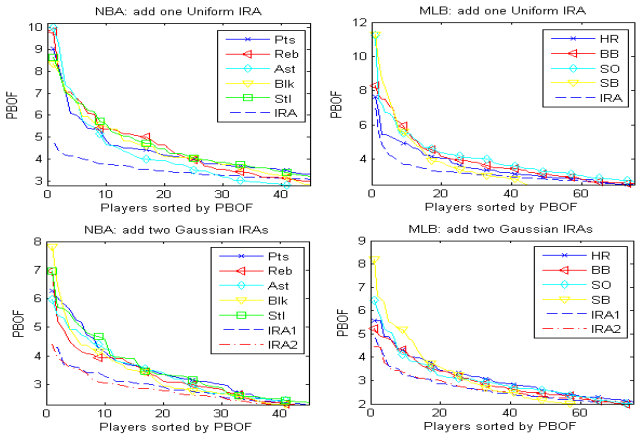


Figure 3. Add one Uniform IRA / add two Gaussian IRAs to NBA dataset and MLB dataset.

## 4.4 The Robustness of the Results

In Section 3, we incorporate the prediction error into the PBOF equation, to deal with independent attributes. We conduct an experiment (shown in Figure 3) to show that our approach can indeed deal with the issue of independent attributes. First, we add a manually created, numerical independent random attribute (IRA1) into the original dataset. This attribute is distributed uniformly from 0 to 1, and is independent of any other attributes in the dataset. Since each data point (player) has its own PBOF with respect to a certain attribute, we can sort and plot the PBOF for every attribute. The curve for IRA1 is represented by a dashed line; the PBOF values of IRA1 are not as high as that of other dimensions. Therefore, while comparing the outlier factors among all instances in all attributes, the outliers with respect to IRA1 would less likely be ranked as the top ones. We also tried different setups such as assigning two independent attributes as well as changing the distribution of IRA1 to Gaussian (IRA2). Similar results were observed in those cases. The results show that

our algorithm is not only robust to the presence of noise but also resists picking the independent attributes to describe outliers.

## 5. Conclusions

The proposed prediction-based outliers not only take into account the external similarity between instances, but also consider the internal dependency amount attributes. Note that our algorithm cannot produce meaningful results when there does not exist any dependent pair of attributes. However, such situation is unlikely to happen when the number of attributes grows. Our framework brings together numerical, categorical, and social neighborhood functions to handle heterogeneous attributes. To overcome the lack of a gold standard for evaluation, we propose explanation, novelty, power-law distribution, and robustness strategies for verification, and demonstrates that our approach has a decent level of fulfillment on all of them. We have also realized that these four strategies might be general enough to be applied to verify other types of knowledge discovery tasks in which no gold standard exists.

## REFERENCES


[1] Barnett, V. and Lewis, T. *Outliers in statistical data*. John Wiley, 1994.

[2] Breunig, M., Kriegel, H.-P., Ng, R. and Sander, J. LOF: identifying density-based local outliers. *SIGMOD Rec.*, *29* (2). 93-104.

[3] Cleveland, W. Robust Locally Weighted Regression and Smoothing Scatterplots. *Journal of the American Statistical Association*, *74* (368). 829-836.

[4] Eberle, W. and Holder, L., Discovering Structural Anomalies in Graph-Based Data. in *Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on*, (2007), 393-398.

[5] Ester, M., Kriegel, H.-p. and Xu, J.S.a.X., A density-based algorithm for discovering clusters in large spatial databases with noise. in *Proceedings of KDD'96*, (Portland OR, USA, 1996).

[6] Guha, S., Rastogi, R. and Shim, K. ROCK: A Robust Clustering Algorithm for Categorical Attributes *15th International Conference on Data Engineering (ICDE'99)*, 1999.

[7] Haynes, S.R. Explanation in Information Systems: A Design Rationale Approach *The London School of Economics*, University of London, 2001.

[8] He, Z., Xu, X. and Deng, S. Discovering cluster-based local outliers. *Pattern Recogn. Lett.*, *24* (9-10). 1641-1650.

[9] Knorr, E. and Ng, R., Algorithms for Mining Distance-Based Outliers in Large Datasets. in *Proc. 24th Int. Conf. Very Large Data Bases, VLDB*, (1998), 392-403.

[10] Knorr, E.M. and Ng, R.T., Finding Intensional Knowledge of Distance-Based Outliers. in *Proceedings of the 25th VLDB Conference*, (Edinburgh, Scotland, 1999).

[11] Kriegel, H.-P., hubert, M.S. and Zimek, A. Angle-based outlier detection in high-dimensional data *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, Las Vegas, Nevada, USA, 2008.

[12] Lin, S.-D. Modeling, searching, and explaining abnormal instances in multi-relational networks, University of Southern California, 2006, 156.

[13] Lin, S.-d. and Chalupsky, H. Discovering and Explaining Abnormal Nodes in Semantic Graphs. *IEEE Trans. on Knowl. and Data Eng.*, *20* (8). 1039-1052.

[14] Nanopoulos, A., Theodoridis, Y. and Manolopoulos, Y. C2P: Clustering based on Closest Pairs *Proceedings of the 27th International Conference on Very Large Data Bases*, Morgan Kaufmann Publishers Inc., 2001.

[15] NEWMAN, M.E.J. and GIRVAN, M., Finding and Evaluating Community Structure in Networks. in *Physics Review, E 69.*, (2004).

[16] Noble, C.C. and Cook, D.J. Graph-based anomaly detection *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, Washington, D.C., 2003.

[17] Papadimitriou, S., Kitawaga, H., Gibbons, P.B. and Faloutsos, C., LOCI: Fast Outlier Detection Using the Local Correlation Integral. in *Data Engineering, 2003. Proceedings, 19th Internatinal Conference on (2003)*, (2003), 315-326.

[18] Pitt, J. *Theory of Explanation*. Oxford University Press, Oxford, 1988.

[19] Ramaswamy, S., Rastogi, R. and Shim, K. Efficient algorithms for mining outliers from large data sets *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, ACM, Dallas, Texas, United States, 2000.

[20] Schank, R. and Kass, A. Explanations, machine learning, and creativity. in *Machine learning: an artificial intelligence approach volume III*, Morgan Kaufmann Publishers Inc., 1990, 31-59.

[21] Sun, J., Qu, H., Chakrabarti, D. and Faloutsos, C., Neighborhood Formation and Anomaly Detection in Bipartite Graphs. in *ICDM '05: Proceedings of the Fifth IEEE International Conference on Data Mining*, (2005), IEEE Computer Society, 418-425.

[22] WHITE, H.C., BOORMAN, S. and BREIGER, R. Social Structure from Multiple Networks. I. Blockmodels of Roles and Positions. *American Journal of Sociology*, *81*. 730–780.

[23] Yamanishi, K. and Takeuchi, J.-I. Discovering outlier filtering rules from unlabeled data: combining a supervised learner with an unsupervised learner. 389-394.

[24] Yao, Y.Y., Zhao, Y. and Maguire, R.B. Explanation-oriented association mining using a combination of unsupervised and supervised learning algorithms, 2003, 527-531.

[25] Zhang, Y., Yang, S. and Wang, Y. LDBOD: A novel local distribution based outlier detector *Pattern Recognition Letters 29*, 2008, 967- 976.