HUBNESS-AWARE OUTLIER DETECTION FOR MUSIC GENRE RECOGNITION

Arthur Flexer *

Austrian Research Institute for Artificial Intelligence
Freyung 6/6/7, Vienna, Austria
arthur.flexer@ofai.at

ABSTRACT

Outlier detection is the task of automatic identification of unknown data not covered by training data (e.g. a new genre in genre recognition). We explore outlier detection in the presence of hubs and anti-hubs, i.e. data objects which appear to be either very close or very far from most other data due to a problem of measuring distances in high dimensions. We compare a classic distance based method to two new approaches, which have been designed to counter the negative effects of hubness, on two standard music genre data sets. We demonstrate that anti-hubs are responsible for many detection errors and that this can be improved by using a hubness-aware approach.

1. INTRODUCTION

Outlier detection[1] is the identification of new or unknown data that a machine learning system is not aware of during training (see [18] for a recent review and [29] for a survey on high-dimensional outlier detection). It is a fundamental requirement for every machine learning system to automatically identify data from regions not covered by the training data since in this case no reasonable decision can be made. In the case of music information retrieval (MIR), an application scenario is the rejection of songs from a previously unseen genre in genre recognition. The same holds for other classifications tasks (e.g. tag or mood), but also for retrieval of similar songs in case a query song is too different from all other songs in a data base. Another example is the automatic rejection of songs from play-lists because they do not fit the overall flavor of the majority of the list. Only little research on outlier detection in MIR so far exists [6, 12, 26, 8].

Hubness is a general problem of learning in high-dimensional spaces and has been recognized as a new aspect of the curse of dimensionality in machine learning literature [20, 23]. Hub objects appear very close to many other data objects and anti-hubs very far from most other data objects. It has been argued and demonstrated that anti-hubs might act as 'artificial' outliers since they are far away from many other data points [20]. A recent review on outlier detection in high dimensional data concluded that the "relation of hubness and outlier degree appears to be remaining an open issue" [29]. It has been demonstrated [10], that many MIR models are inherently high-dimensional and highly prone to hubness. In a first MIR study on outlier detection in high dimensions [8], we were able to show that it is possible to improve outlier detection by using a hubness reduction method as a preprocessing step.

In this paper we explore whether such improvements in high-dimensional outlier detection are on account of the changed role of hubs and anti-hubs due to the hubness reduction method. This is done by analyzing the performance of hubs and anti-hubs in a classic distance based method and in two hubness-aware approaches, all applied in a music genre recognition setting.

2. RELATED WORK

Outlier detection, also known as novelty detection, is the task of automatically recognizing data that differ in some respect from the data seen during training by a machine learning system. In case new data differs substantially from training data, no sensible decision can be made by a machine learning system. This should of course be an integral part of any data analysis system and therefore a vast literature concerning the topic exists. For this paper, we follow the systematic of a very recent and comprehensive review [18], which also contains a representative list of references concerning the topic. According to this review, outlier detection can be distinguished into probabilistic, distance-based, reconstruction-based, domain-based and information theoretic approaches. The methods we will present in Section 4 are all distance-based, more specifically based on nearest neighbor information. They are also all unsupervised, i.e. class labels are not needed for detection of outliers. Of greater importance is a recent review that deals specifically with outlier detection in high-dimensional data [29]. Although the authors show how some classic outlier detection methods are affected by the concentration of distances (see also next paragraph), hubness is only reviewed as a remaining open issue.

The concept and term of hubness has been discovered and first described in MIR [1], but then gained attention in a machine learning context where it has been discussed as a new aspect of the curse of dimensionality and a general problem of learning in high-dimensional spaces [20, 23]. Hubness is related to the phenomenon of concentration of distances, which is the fact that all points are at almost the same distance to each other for dimensionality approaching infinity [11]. Radovanović et al. [20] presented the argument that for any finite dimensionality, some points are expected to be closer to the center of all data than other points and are at the same time closer, on average, to all other points. Such points closer to the center have a high probability of being hubs, i.e. of appearing in nearest neighbor lists of many other points. Points which are further away from the center have a high probability of being anti-hubs, i.e. points that never appear in any nearest neighbor list. Hubness has been shown to have a negative impact on many tasks including classification [20], nearest neighbor based recommendation [10] and retrieval [24], clustering [27, 22] and visualization [4]. Many of these reports are from the MIR community (e.g. [14, 2, 10, 4, 5]). It also affects data from diverse domains including multimedia (text, music, images, speech), biol-

* This work was supported by the Austrian Science Fund (FWF, grant P27082).

[1] Please note that the terms outlier and novelty detection are closely related although not fully synonymous. We will use the term outlier throughout the paper without further distinction for reasons of convenience.

ogy and general machine learning (see [20, 23, 3] for large scale empirical studies).

In order to reduce hubness and its negative effects, we have proposed two unsupervised methods to re-scale high-dimensional distance spaces [23]: Local Scaling (LS) and Mutual Proximity (MP). Both methods aim at repairing asymmetric nearest neighbor relations. The asymmetric relations are a direct consequence of the presence of hubs. A hub y is the nearest neighbor of x, but the nearest neighbor of the hub y is another point a ($a \neq x$). This is because hubs are by definition nearest neighbors to very many data points but only one data point can be the nearest neighbor to a hub. The principle of the scaling algorithms is to re-scale distances to enhance symmetry of nearest neighbors. A small distance between two objects should be returned only if their nearest neighbors concur. Application of LS and MP resulted in a decrease of hubness and an accuracy increase in k-nearest neighbor classification on thirty real world datasets including text, image and music data.

Just recently we re-visited our own results [6] on outlier detection in MIR and tried to use mutual proximity (MP) for the task [8]. After all, MP rescales distances to probabilities of mutual proximity which allows for convenient thresholding to detect outlier data due to its probabilistic interpretation. We were able to show that outlier detection based on MP improves the ability to reject outlier data when compared to a classic distance based method, again in a genre classification context. A necessary next step is to investigate whether the improvement achieved is on account of the changed role of hubs and anti-hubs due to the application of mutual proximity. It seems clear that anti-hubs, being far away from most points, will probably always be rejected as outliers and that hub objects, being close to many points, should be harder to reject. It is therefore our hypothesis that in high-dimensional data, hub and anti-hub points are responsible for many errors being made when rejecting data.

Such a first analysis of the role of anti-hubs in outlier detection has recently been presented [19]. More specifically, the authors have analysed two variants of the ODIN method [13], k-NN outlier scoring [21] and three other methods concerning their relation to anti-hubs. The two variants of the ODIN method use reverse nearest neighbor counts, i.e. counts of how often every data object appears among the k nearest neighbors of every other data object. Per definition anti-hubs have very small or even zero reverse nearest neighbor counts. The authors show that outlier scores based on these counts are correlated to scores from other detection methods but do provide some extra information. Outlier detection results of the ODIN-based methods are however rather mixed when compared to other methods applied to twelve real world data sets. It also has to be said that these data sets are not really very high dimensional (from only 5 to at most 100) and therefore most of them are probably not affected by hubness at all.

Concerning the dimensionality of data sets, it is important to note that the degree of concentration and hubness is linked to the intrinsic rather than extrinsic dimension of the data space. Whereas the extrinsic dimension is the actual number of dimensions of a data space, the intrinsic dimension is the, often much smaller, number of degrees of freedom of the submanifold in which the data space can be represented [11]. Our previous research [23] has shown that real world data with extrinsic dimensionality as small as 34 can already exhibit the negative effects of hubness, while other data with extrinisc dimensionality of more than 10000 is still not affected. It is also true that simple dimensionality reduction does not reduce hubness. On the contrary it has been shown that only projections to very few dimensions, well below the intrinsic dimensionality of a data set, are able to reduce hubness, but at the cost of a loss of distance information [20].

3. DATA

For our analysis, we chose to use a genre classification framework, following the hypothesis that songs within a certain genre are more similar to each other than songs from different genres. During evaluation in Section 5, we will always reserve all songs belonging to one of the genres as outlier songs to be detected against the rest of the songs from all other genres. Songs from an unknown genre therefore act as outliers.

For our experiments we used two standard music databases: the "GTZAN" collection consisting of $N = 1000$ audio tracks (each 30 s length) evenly spread over $G = 10$ music genres [28]; the "ISMIR2004" [2] collection containing $N = 1458$ tracks of $G = 6$ genres, with full-length audio being available and exhibiting a highly imbalanced genre distribution with classical music comprising almost half of the tracks.

We decided to compute timbre information from the audio, since this is an integral part of many MIR systems and at the same time has already been shown to be susceptible to hubness [10]. Every track is divided into overlapping frames for which 20 MFCCs are being computed which are modeled via a single Gaussian with full covariance matrix. To compute a distance value between two Gaussians the symmetrized Kullback-Leibler (SKL) divergence is used (see [17] for details on both MFCCs and SKL). This results in $N \times N$ distance matrices D^G and D^I for the GTZAN and ISMIR data sets. Therefore the data sets are represented as distance spaces only, not vector spaces, and it is not possible to report their extrinsic dimensionality. Their intrinsic dimensionality measured via a maximum likelihood estimator [16] is 10.94 for GTZAN and 9.00 for ISMIR.

4. METHODS

We now describe all three methods we will use for outlier detection. All of them compute an outlier score (S^{kNN}, S^{AH} and S^{MP}), which is bounded between 0 and 1 and is compared to a threshold p to decide whether a data object is an outlier or not. A data object is rejected if:

$$S > p. \tag{1}$$

4.1. kNN-reject

The first method is a standard distance-based approach known as k-NN outlier scoring [21]. The outlier score is the average distance to the k nearest neighbors:

$$S^{kNN}(x) = \frac{1}{k}\sum_{i=1}^{k} D_{x,\mathrm{NN}_i(x)}, \tag{2}$$

with $\mathrm{NN}_i(x)$ being the ith nearest neighbor of x. The distance matrices D^G and D^I (see Section 3) are normalized to the interval 0 to 1 by subtracting the minimum distance and dividing through the maximum distance. This also bounds the outlier score S^{kNN} between 0 and 1.

[2] `http://ismir2004.ismir.net/genre_contest/index.html`

4.2. AH-reject

The next method is based on previous work of using reverse nearest neighbor counts for outlier detection [19]. In hubness research, the reverse nearest neighbor count of a point x is usually called n-occurrence $O^n(x)$ [20]. It is the number of times x occurs in the first n nearest neighbors of all other objects in the collection. The proposed method simply uses $O^n(x)$ as the outlier score. It has been termed "Antihub" because anti-hubs have very small or even zero n-occurrence and are therefore very likely to be rejected as outliers. The same authors proposed a variant called "Antihub2" which also includes information from the k nearest neighbors of x:

$$S^{AH}(x) = (1-\alpha)\frac{1}{O^n(x)+1} + \alpha\sum_{i=1}^{k}\frac{1}{O^n(\mathrm{NN}_i(x))+1}. \quad (3)$$

We set $\alpha = k/(k+1)$, which basically gives the average across a function of the n-occurrences of x itself and its k nearest neighbors $\mathrm{NN}_i(x)$. The outlier score S^{AH} is also bounded between 0 and 1, with $S^{AH} = 1$ in case all involved n-occurrences O^n are equal zero, and $S^{AH} = 1/N$ in case all involved $O^n = N-1$. An n-occurrence is equal $N-1$ in case a data point appears in all neighborhood lists of all other data points.

4.3. MP-reject

Mutual Proximity (MP) [23] rescales the original distance space so that two objects sharing similar nearest neighbors are more closely tied to each other, while two objects with dissimilar neighborhoods are repelled from each other. MP has been devised to counter the negative effects of hubness in high dimensional data spaces. For MP-reject we exploit the fact that MP rescales distances to probabilities which enables comparability and simple thresholding. MP reinterprets the distance of two objects as a mutual proximity in terms of their distribution of distances. To compute MP, we assume that the distances $D_{x,i=1..N}$ from an object x to all other objects in our data set follow a certain probability distribution $P(X)$, thus any distance $D_{x,y}$ can be reinterpreted as the probability of y being the nearest neighbor of x, given their distance $D_{x,y}$ and the probability distribution $P(X)$:

$$P(X > D_{x,y}) = 1 - P(X \le D_{x,y}) = 1 - \mathcal{F}_x(D_{x,y}), \quad (4)$$

with $\mathcal{F}$ denoting the cumulative distribution function (cdf). MP is then defined as the probability that y is the nearest neighbor of x given $P(X)$ and x is the nearest neighbor of y given $P(Y)$:

$$MP(D_{x,y}) = P(X > D_{x,y} \cap Y > D_{y,x}). \quad (5)$$

To compute MP in our experiments we assume that the distances $D_{x,i=1..N}$ follow a Gaussian distribution. We define the outlier score as the average of the MP-distances to the k nearest neighbors of x:

$$S^{MP}(x) = \frac{1}{k}\sum_{i=1}^{k}(1 - MP(x, \mathrm{NN}_i(x))), \quad (6)$$

with $\mathrm{NN}_i(x)$ being the ith nearest neighbor of x. Please note that we use the term $(1 - MP)$ because mutual proximity computes similarities and we need distances for the rejection rule. Outlier score S^{MP} is bounded between 0 and 1 since it is based on MP which computes a probability.

5. RESULTS

Before evaluating the outlier detection methods, we present an analysis of the hubness of the data sets GTZAN and ISMIR in Table 1. The table gives the number of data objects N, number of genres G, the number of hubs $\#hub$, anti-hubs $\#anti$ and normal $\#normal$ data objects. Anti-hubs are defined as data objects with an n-occurrence O^n (see Sec. 4.2) equal 0, hubs with $O^n > 5n$, all based on numbers of nearest neighbors of $n = 5$. Normal data objects are all non-hub and non-anti-hub objects, i.e. $0 < O^n \le 5n$. Please note that the mean n-occurrence across all objects in a data base is equal to n. Any n-occurrence significantly bigger than n therefore indicates existence of a hub. As in previous work [23], we chose objects appearing more than five times the expected value ($5n = 25$) as hub objects. As can be seen, consistent with theory, in both data sets there are small numbers of hubs (GTZAN 21, ISMIR 24) and large numbers of anti-hubs (GTZAN 186, ISMIR 276). The last column of Table 1 gives the hubness H^n. It is the skewness of the distribution of n-occurrences, i.e. the third moment of the distribution. A data set having high hubness produces few hub objects with very high n-occurrence and many anti-hubs with n-occurrence of zero. This makes the distribution of n-occurrences skewed with positive skewness indicating high hubness. The hubness values H^n of 3.29 for GTZAN and 3.94 for ISMIR show that there is a clear hubness effect in these data sets. Previous work [23] has shown that values above 1.4 are already problematic.

Table 1: *Hubness analysis of data sets GTZAN and ISMIR, see Sec. 5.*

data set	N	G	$\#hub$	$\#anti$	$\#normal$	H^n
GTZAN	1000	10	21	186	793	3.29
ISMIR	1458	6	24	276	1158	3.94

To evaluate the three outlier detection methods described in Sec. 4 we use the following approach shown as MATLAB style pseudo-code in Table 2. First we set aside all songs belonging to a genre `g` as new songs (`[new,data]=separate(alldata, g)`) which yields data sets `new` and `data` (all songs not belonging to genre g). Then we do a $C = 10$-fold crossvalidation using `data` and `new`: we randomly split `data` into `train` and `test` fold (`[train,test] = split(data,c)`) with `train` always consisting of 90% and `test` of 10% of `data`. We compute the percentage of `new` songs which are rejected as being outliers (`outlier_reject(g,c) = outlier(new)`) and do the same for the `test` songs (`test_reject(g,c) = outlier(test)`). Last we compute the classification accuracy on `test` data that has not been rejected as being outliers (`accuracy(g,c) = classify(test(not test_reject)))`). As a classifier we use simple one-nearest neighbor classification. The evaluation procedure gives $G \times C$ (GTZAN 10×10, ISMIR 6×10) matrices of `outlier_reject`, `test_reject` and `accuracy` for each parameterization of the outlier detection approaches, i.e. for different values of k (see Sec. 4). In what follows we always report average numbers across these $G\times C$ sized matrices of results, i.e. averages across crossvalidation folds and genres.

The results for outlier detection are given in Figs. 5 and 6 as Receiver Operating Characteristic (ROC) curves. To obtain an ROC curve, the fraction of false positives (object is not an outlier

Table 2: *Outline of evaluation procedure, see Sec. 5.*

```
for g = 1 : G
  [new,data] = separate(alldata,g)
  for c = 1 : C
    [train,test] = split(data,c)
    outlier_reject(g,c) = outlier(new)
    test_reject(g,c) = outlier(test)
    accuracy(g,c) =
      classify(test(not test_reject))
  end
end
```

but it is rejected, in our case `test_reject`) is plotted versus the fraction of true positives (object is an outlier and correctly rejected, in our case `outlier_reject`) for varying threshold values p. We vary the threshold values from $p = 0$ to $p = 1$ in steps of .02. An ROC curve shows the trade off between how sensitive and how specific a method is. Any increase in sensitivity will be accompanied by a decrease in specificity. If a method becomes more sensitive towards outlier objects it will reject more of them but at the same it will also become less specific and also falsely reject more non-outlier objects. Consequently, the closer a curve follows the left-hand border and then the top border of the ROC space, the better the performance of the method is.

To summarize the information contained in ROC curves, we also compute the Area Under the Curve (AUC), which gives the percentage of the whole ROC space that lies underneath an ROC curve. An AUC of 1 indicates perfect performance, while an AUC of .5 indicates performance at chance level.

As a first analysis step, we tried to find optimal parameters k (neighborhood size for algorithms kNN, AH, MP) by comparing AUC results based on values of $k = 1, 2, 3, 5, 10, 20, 30, 40, 50$. Results for GTZAN can be found in Figure 1, for ISMIR in Figure 2. Looking at the GTZAN results, the AUC values (y-axis) for all three methods monotonically decrease with neighborhood size increasing beyond $k = 1$. The only exception is a very small gain in AUC for MP going from $k = 1$ to $k = 2$. Looking at the results for ISMIR in Figure 2, the AUC values again monotonically decrease beyond $k = 1$ for methods kNN and MP. Only for method AH, there is a small and slow rebound starting at about $k = 20$. We therefore conclude that there is no gain in increasing the neighborhood size k beyond 1 for any of the methods. All following results are therefore based on the choice of $k = 1$. We can also see from Figures 1 and 2, that MP improves AUC results over kNN across the whole range of k and for both data sets. It is also already evident that AH is never able to improve performance of kNN.

We now make a detailed AUC analysis separate for all data as well as for hubs, anti-hubs and normal data as defined at the beginning of this section (neighborhood size $k = 1$ for all outlier detection methods). Looking at the results for GTZAN in Figure 3, we can see that for all data together (left most group of bars in figure), MP increases performance to .81 compared to kNN which achieves .70. Method AH actually decreases performance to .67. Looking at hubs and anti-hubs separately for method kNN, it is clear that anti-hub objects perform worst with an AUC of .59 versus .71 for hubs and .73 for the remaining normal data. We see the same pattern of lowest AUC for anti-hubs for method AH also, albeit at an even lower level. Method MP is able to increase AUC compared to kNN for both anti-hubs (from .59 to .71) and normal data (from .73 to .86). There is a small decrease in AUC for hubs (from .71 to .67). Looking at results for ISMIR in Figure 4, we basically see the same pattern. Method kNN performs worst for anti-hubs, normal and hub objects perform at about the same level. Method AH repeats this pattern at a lower level. Method MP is able to improve the performance for anti-hubs and normal objects but not for hubs. Gains in performance compared to kNN are smaller than for GTZAN (e.g. from .78 to .83 for all data objects).

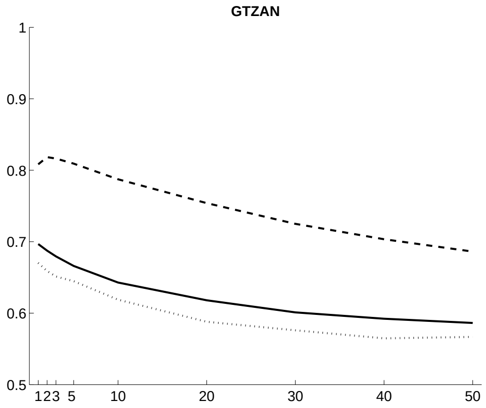

Figure 1: *AUC (y-axis) analysis for data set GTZAN and parameter k ranging from 1,2,3,5,10,20,30,40 to 50 (x-axis), solid line for kNN, dotted for AH, dashed for MP.*

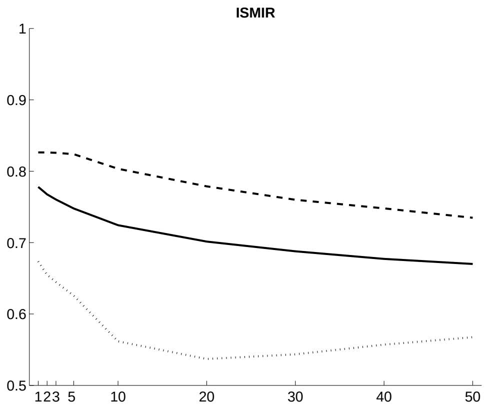

Figure 2: *AUC (y-axis) analysis for data set ISMIR and parameter k ranging from 1,2,3,5,10,20,30,40 to 50 (x-axis), solid line for kNN, dotted for AH, dashed for MP.*

We now present the ROC plots that the above AUC results are based on, again for all data as well as for hubs, anti-hubs and normal data separately (neighborhood size $k = 1$ for all outlier detection methods). Looking at the results for GTZAN in Figure 5, we can see that the ROC curve for method MP (dashed line) is above those for kNN (solid line) and AH (dotted line) for almost the whole ROC space for all data together as well as for anti-hubs and normal objects (sub-plots titled 'ALL', 'ANTI-HUB' and 'NORMAL'). The ROC curves for hub objects are quite comparable for all three methods. It is also interesting to see, that the ROC curve for method AH and anti-hub objects (sub-plot titled 'ANTI-

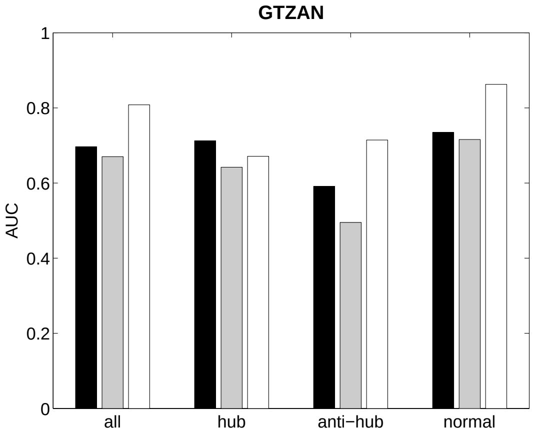

Figure 3: *AUC plot for data set GTZAN, black bars for kNN, gre for AH, white for MP (neighborhood size $k = 1$).*

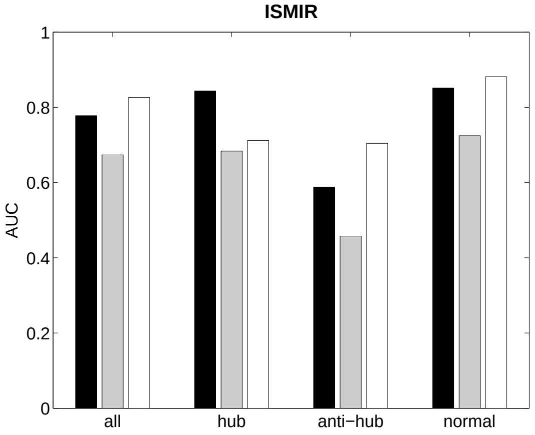

Figure 4: *AUC plot for data set ISMIR, black bars for kNN, grey for AH, white for MP (neighborhood size $k = 1$).*

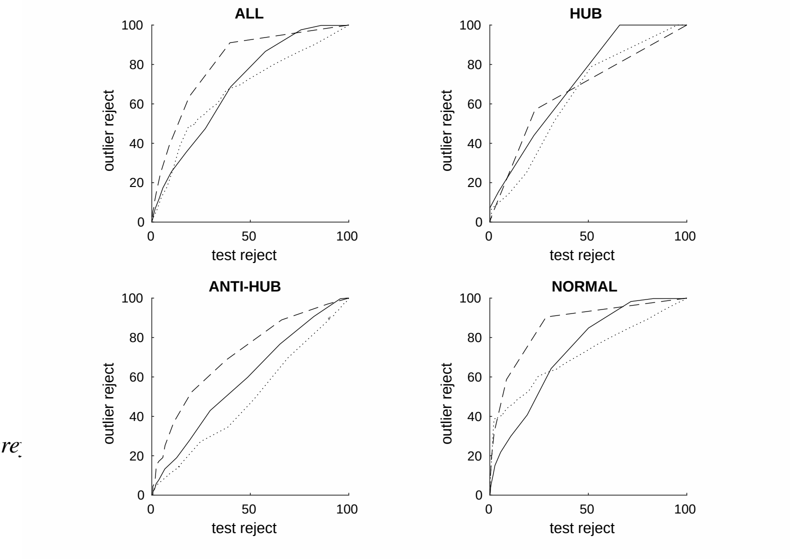

Figure 5: *ROC plots for data set GTZAN, solid line for kNN, dotted for AH, dashed for MP (neighborhood size $k = 1$).*

Table 3: *Accuracy analysis of data set GTZAN, see Sec. 5.*

method	acc	acc_rej	o_rej	t_rej
kNN	77.54	98.90	99.82	85.94
AH	77.54	82.03	76.94	56.77
MP	82.06	94.69	91.02	39.69

Table 4: *Accuracy analysis of data set ISMIR, see Sec. 5.*

method	acc	acc_rej	o_rej	t_rej
kNN	87.11	100.00	100.00	62.76
AH	87.11	93.08	73.72	46.60
MP	91.11	98.04	87.77	28.10

HUB', dotted line) is very close to the main diagonal indicating performance at chance level. This explains the very low AUC of .59 for this curve, with .5 indicating chance level. This is due to the fact, that method AH is based on n-occurrence counts (see Section 4.2), basically detecting everything with a low n-occurrence as outliers. It therefore rejects all anti-hubs as outliers, no matter whether they are true outliers or test data that should not be rejected. Figure 6 gives the ROC plots for ISMIR, repeating the same patterns of behavior we just described, albeit less clearly. ROC curves for method MP more or less dominate those for kNN and AH for all data together as well as anti-hubs and normal data. Method AH performs even below chance level for anti-hubs with an AUC of only .46.

Finally, we present results concerning one-nearest neighbor classification accuracy gains due to the outlier rejection. While steadily lowering threshold value p and rejecting more and more test and outlier data, less and less data is being classified but usually with increased accuracy. The following results are averages across test data not used for training of the classifier and not rejected by the respective outlier detection methods. In Tables 3 and 4 we present for data sets GTZAN and ISMIR the classification accuracy without any outlier rejection (acc), the maximum achieved accuracy after rejection (acc_rej), the percentage of rejected outlier data (o_rej) and the percentage of rejected test data (t_rej) at the respective threshold level. The baseline accuracy acc of using method kNN and not rejecting at all is 77.54% for GTZAN. Please note that this baseline accuracy is already at 82.06% when using distances rescaled via MP, again while not rejecting at all. With outlier rejection this can be improved up to 98.90% with kNN and 94.69% with MP, but only 82.03% with AH. But to reach this maximum accuracy, kNN does not only correctly reject 99.82% of outliers, but also falsely rejects 85.94% of test data. Method MP on the other hand reaches its maximum accuracy while correctly rejecting 91.02% of outliers but only 39.69% of test data. Data set ISMIR shows the same pattern with accuracies improving up to 100.00% and 98.04% for kNN and MP, with AH lagging behind at 93.08%. Again kNN has to reject much more test data than MP to reach these results (62.76% vs. only 28.10%).

6. DISCUSSION

In discussing our results obtained in Section 5, we like to recapitulate our main findings.

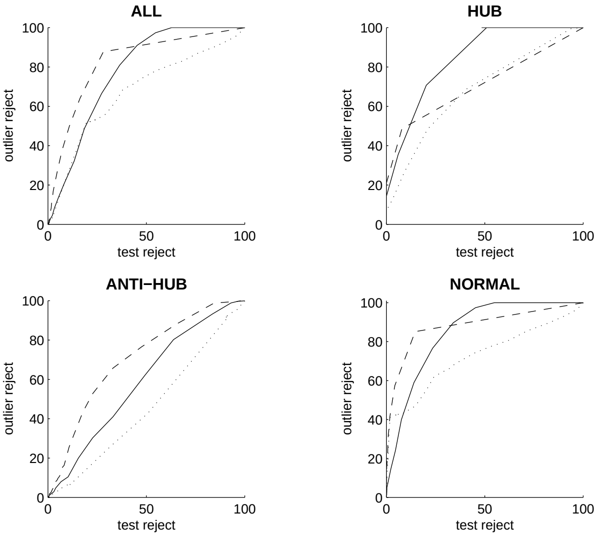

Figure 6: *ROC plots for data set ISMIR, solid line for kNN, dotted for AH, dashed for MP (neighborhood size $k = 1$).*

Our first result is that classic distance-based outlier rejection methods are negatively affected by hubness. As can be seen by looking at the results for distance-based method kNN separately for hubs, anti-hubs and normal data, especially anti-hubs present problems for outlier detection as evident from their lower AUC values. Since anti-hubs per definition are far away from most other data points in a data base, it seems logical that they are being detected as outliers even when they not really are. As for hub objects, we would have expected that they are also responsible for more detection errors than normal data, which is not really the case. In our analysis we only looked at hub objects as being candidates for outlier detection and there are maybe too few hub objects (21 in GTZAN and 24 in ISMIR) in the data sets to gain meaningful statistics. But hub objects are per definition in nearest neighbor lists of many other data objects and through these lists very likely enter computation of outlier scores of many data objects in a data set. It would definitely be very interesting to do another analysis of outlier detection considering this fact.

Our second result is that reverse nearest neighbor information cannot improve distance-based outlier detection that is affected by hubness. On the contrary, our results for method AH show that AUC values compared to kNN even deteriorate. Especially AUC results for anti-hubs are sometimes even below chance level. Given the fact that AH basically detects everything with a low reverse nearest neighbor count as an outlier, this is not surprising. After all this means that anti-hubs, which already are a problem for distance-based methods, are detected as outliers no matter whether they really are or not.

Our third result is that our hubness-aware algorithm MP is able to improve outlier detection and that this improvement is also due to the changed role of anti-hubs. Looking at our results for using MP, we can see that we gain overall improvements in AUC which are especially pronounced for anti-hub and normal data. Mutual proximity has been shown to decisively reduce the negative impact of hubs and anti-hubs and produce distance spaces with much more normal behavior [23]. Rescaling via MP is able to prevent anti-hubs from being far away from most other data points, therefore not acting as ‘artificial’ outliers anymore.

Given these three main results, it would now of course be very interesting to analyse the behavior of other outlier detection methods when confronted with data affected by hubness. Of special interest are methods that have already been designed for high-dimensional outlier detection like e.g. “angle-based outlier detection” (ABOD) [15].

Our data analysis is based on a model of timbre similarity only, since this is an important part of most MIR systems modeling music similarity. Previous results have shown that many different parametrizations of audio are susceptible to hubness [10], therefore our results are important for MIR models beyond timbre also. But it also clear that there exist models of music similarity that are not prone to hubness, e.g. certain combinations of timbre and rhythm aspects [9]. And it is also clear that there exist other methods to reduce hubness in MIR models, e.g. the usage of Universal Background Models [2], which is a method from speech analysis.

Concerning usage of data sets in our study, repetition with larger data sets and other MIR problems (e.g. tag classification) would of course be interesting. We are also aware of the criticism concerning the GTZAN data set and its faults [25] like e.g. mislabeling. But these problems mainly concern classification results and not so much outlier detection which after all is the main focus of our work here.

Of greater importance is maybe the fact that we do not use artist filters in our analysis. An artist filter [7] prevents songs from the same artist to be both in the training and test set. This is important since songs from the same artist are often very similar and not using an artist filter can lead to over optimistic results, e.g. in terms of genre classification. But as we just said above, this is not our main focus here and very likely this problem affects all three of our methods (kNN, AH and MP) in the same way. But it is also a wellknown effect that songs from the same artist dominate the nearest neighbor lists in audio-based music similarity. We can only speculate whether songs from the same artist are able to even prevent hub songs from entering nearest neighbor lists. If this were the case, usage of an artist filter would definitely change our outlier detection results. Maybe the impact of hubness on outlier detection is even greater if artist filters are being used. Such an analysis is beyond the scope of this paper but definitely interesting future work.

7. CONCLUSIONS

We have presented a first detailed study of the role of hubness in outlier detection for music genre recognition. We found that classic distance-based methods for outlier rejection are negatively impacted by hubness, where especially anti-hubs pose a problem. We also showed that a recently proposed outlier method based on reverse nearest neighbor counts is not able to help in this respect. But a hubness-aware method based on hubness reduction via computation of mutual proximity is able to improve outlier detection results. Improvements concerning the problematic role of anti-hubs are part of the success. Since hubness is due to a general problem of measuring distances in high-dimensional spaces and since many models in music information retrieval have been shown to be affected, our results are of interest beyond the focus on genre recognition in this paper.

8. REFERENCES

[1] Aucouturier, J.-J., Pachet F.: Improving Timbre Similarity: How high is the sky?, Journal of Negative Results in Speech and Audio Sciences, 1(1), 2004.

[2] Charbuillet C., Tardieu D., Peeters G.: GMM supervector for Content Based Music Similarity, Proceedings of the 14th International Conference on Digital Audio Effects (DAFX'11), Paris, France, 2011.

[3] Feldbauer R., Flexer A.: Centering versus Scaling for Hubness Reduction, Proceedings of the 25th International Conference on Artificial Neural Networks (ICANN'16), Barcelona, Spain, 2016.

[4] Flexer A.: Improving visualization of high-dimensional music similarity spaces, 16th International Society for Music Information Retrieval Conference, Malaga, Spain, 2015.

[5] Flexer A.: The impact of hubness on music recommendation, Machine Learning for Music Discovery Workshop at the 32nd International Conference on Machine Learning, Lille, France, 2015.

[6] Flexer A., Pampalk E., Widmer G.: Novelty detection based on spectral similarity of songs, in Proc. of the 6th Int. Conf. on Music Information Retrieval, 2005.

[7] Flexer A., Schnitzer D.: Effects of Album and Artist Filters in Audio Similarity Computed for Very Large Music Databases, Computer Music Journal, Volume 34, Number 3, pp. 20-28, 2010.

[8] Flexer A., Schnitzer D.: Using mutual proximity for novelty detection in audio music similarity, in Proceedings of the 6th International Workshop on Machine Learning and Music, Prague, Czech Republic, 2013.

[9] Flexer A., Schnitzer D., Gasser M., Pohle T.: Combining features reduces hubness in audio similarity, Proceedings of the Eleventh International Society for Music Information Retrieval Conference, 2010.

[10] Flexer A., Schnitzer D., Schlüter J.: A MIREX meta-analysis of hubness in audio music similarity, Proceedings of the 13th International Society for Music Information Retrieval Conference, 2012.

[11] Francois D., Wertz V., Verleysen M.: The concentration of fractional distances, IEEE Transactions on Knowledge and Data Engineering, 19:873-886, 2007.

[12] Hansen L.K., Lehn-Schiøler T., Petersen K.B., Arenas-Garcia J., Larsen J., Jensen S.H.: Learning and clean-up in a large scale music database, in Proc. of the European Signal Processing Conference (EUSIPCO), pp. 946-950, IEEE, 2007.

[13] Hautamäki V., Kärkkäinen I., Fränti P.: Outlier Detection Using k-Nearest Neighbour Graph, 17th International Conference on Pattern Recognition, pp. 430-433, 2004.

[14] Karydis I., Radovanović M., Nanopoulos A., Ivanović M.: Looking Through the "Glass Ceiling": A Conceptual Framework for the Problems of Spectral Similarity, in Proc. of the 11th Int. Conf. on Music Information Retrieval, pp. 267-272, 2010.

[15] Kriegel H. P., Schubert M., Zimek A.: Angle-based outlier detection in high-dimensional data, Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining, pp. 444-452, 2008.

[16] Levina E., Bickel P.J.: Maximum likelihood estimation of intrinsic dimension, in Advances in Neural Information Processing Systems 17, pp. 777-784, 2005.

[17] Mandel M., Ellis D.: Song-level features and support vector machines for music classification, in Proc. of the 6th Int. Conf. on Music Information Retrieval, 2005.

[18] Pimentel M.A.F., Clifton D.A., Clifton L., Tarassenko L.: A review of novelty detection, Signal Processing, Vol. 99, pp. 215-249, 2014.

[19] Radovanović M., Nanopoulos A., Ivanović M.: Reverse nearest neighbors in unsupervised distance-based outlier detection, IEEE Transactions on Knowledge and Data Engineering, 27(5), 1369-1382, 2015.

[20] Radovanović M., Nanopoulos A., Ivanović M.: Hubs in space: Popular nearest neighbors in high-dimensional data, Journal of Machine Learning Research, 11:2487-2531, 2010.

[21] Ramaswamy S., Rastogi R., Shim K.: Efficient algorithms for mining outliers from large data sets, Proceedings of the 2000 ACM SIGMOD international conference on Management of data (SIGMOD '00), pp. 427-438, 2000.

[22] Schnitzer D., Flexer A.: The Unbalancing Effect of Hubs on K-medoids Clustering in High-Dimensional Spaces, Proceedings of the International Joint Conference on Neural Networks (IJCNN), 2015.

[23] Schnitzer D., Flexer A., Schedl M., Widmer G.: Local and Global Scaling Reduce Hubs in Space, Journal of Machine Learning Research, 13(Oct):2871-2902, 2012.

[24] Schnitzer D., Flexer A., Tomašev N.: A Case for Hubness Removal in High-Dimensional Multimedia Retrieval, Proceedings of the 36th European Conference on Information Retrieval (ECIR), 2014.

[25] Sturm, B: An analysis of the GTZAN music genre dataset, Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies, ACM, 2012.

[26] Sturm B.L.: Music genre recognition with risk and rejection, in Proceedings of International Conference on Multimedia and Expo, 2013.

[27] Tomašev N., Radovanović M., Mladenić D., Ivanović M.: The Role of Hubness in Clustering High-dimensional Data, IEEE Transactions on Knowledge and Data Engineering, Volume 26, Issue 3, 2013.

[28] Tzanetakis G., Cook P.: Musical genre classification of audio signals, IEEE Transactions on Speech and Audio Processing, Vol. 10, Issue 5, 293-302, 2002.

[29] Zimek A., Schubert E., Kriegel H.-P.: A survey on unsupervised outlier detection in high-dimensional numerical data, Statistical Analysis and Data Mining, 5: 363-387, 2012.