Class Imbalance and The Curse of Minority Hubs

Nenad Tomašev, Dunja Mladenić

[a]*Institute Jožef Stefan*
Artificial Intelligence Laboratory
Jamova 39, 1000 Ljubljana, Slovenia
nenad.tomasev@ijs.si, dunja.mladenic@ijs.si

Abstract

Most machine learning tasks involve learning from high-dimensional data, which is often quite difficult to handle. *Hubness* is an aspect of the *curse of dimensionality* that was shown to be highly detrimental to k-nearest neighbor methods in high-dimensional feature spaces. *Hubs*, very frequent nearest neighbors, emerge as centers of influence within the data and often act as semantic singularities. This paper deals with evaluating the impact of hubness on learning under class imbalance with k-nearest neighbor methods. Our results suggest that, contrary to the common belief, minority class hubs might be responsible for most misclassification in many high-dimensional datasets. The standard approaches to learning under class imbalance usually clearly favor the instances of the minority class and are not well suited for handling such highly detrimental minority points. In our experiments, we have evaluated several state-of-the-art hubness-aware kNN classifiers that are based on learning from the neighbor occurrence models calculated from the training data. The experiments included learning under severe class imbalance, class overlap and mislabeling and the results suggest that the hubness-aware methods usually achieve promising results on the examined high-dimensional datasets. The improvements seem to be most pronounced when handling the difficult point types: borderline points, rare points and outliers. On most examined datasets, the hubness-aware approaches improve the classification precision of the minority classes and the recall of the majority class, which helps with reducing the negative impact of minority hubs. We argue that it might prove beneficial to combine the extensible hubness-aware voting frameworks with the existing class imbalanced kNN classifiers, in order to properly handle class imbalanced data in high-dimensional feature spaces.

Keywords: class imbalance, class overlap, classification, k-nearest neighbor, hubness, curse of dimensionality

1. Introduction

Nearest-neighbor methods form an important group of techniques involved in solving various types of machine learning tasks. They are based on a simple assumption that neighboring points share certain common properties. Often enough, they also share the same label, which is why so many different k-nearest neighbor classification algorithms have been developed over the years [28][54][36][64][53][90].

The basic k-nearest neighbor algorithm (kNN) [19] is quite simple. The label in the point of interest is derived from its k-nearest neighbors by a majority vote. The kNN rule has some favorable asymptotic properties [11].

Under the basic kNN approach, no model is generated in the training phase and the target function is inferred locally when the query is made to the system. Methods with this property are said to perform *lazy learning*.

Algorithms which induce classification models usually adopt the maximum generality bias [33]. In contrast, the k-nearest neighbor classifier exhibits high specificity bias, since it retains all the examples. The specificity bias is considered a desired property of algorithms designed for handling highly imbalanced data. Not surprisingly, kNN has been advocated as one way of handling such imbalanced data sets [84][33].

[0]This paper was published by Elsevier in the Knowledge-Based Systems journal in 2013. DOI: "http://dx.doi.org/10.1016/j.knosys.2013.08.031".

Data sets with significant class imbalance often pose difficulties for learning algorithms [87], especially those with a high generality bias. Such algorithms tend to over-generalize on the majority class, which in turn leads to a lower performance on the minority class. Designing good methods capable of coping with highly imbalanced data still remains a daunting task.

Certain concerns have recently been raised about the applicability of the basic kNN approach in imbalanced scenarios [23]. The method requires high densities to deliver good probability estimates. These densities are often closely related to class size, which makes kNN somewhat sensitive to the imbalance level. The difference among the densities between the classes becomes critical in the overlap regions. Data points from the denser class (usually the *majority class*) are often encountered as neighbors of points from the less dense category (usually the *minority class*). In high-dimensional data the task is additionally complicated by the well known *curse of dimensionality*.

High dimensionality often exhibits a detrimental influence on classification, since all data is sparse and density estimates tend to become less meaningful. It also gives rise to the phenomenon of *hubness* [59], which greatly affects nearest neighbor methods in high-dimensional data. The distribution of neighbor occurrences becomes skewed to the right and most points either never occur in k-neighbor sets or occur very rarely. A small number of points, *hubs*, account for most of the observed neighbor occurrences. Hubs are very frequent nearest neighbors[1] and, as such, exhibit a substantial influence on subsequent reasoning.

The hubness issue first emerged in music retrieval and recommendation systems, where some songs were being too frequently retrieved, even in such cases where it was impossible to discern some reasonable semantic correlation to the queries [3][2]. Such song hubs were detrimental to the system performance. It was initially thought that this was merely a consequence of the discrepancies between the perceptual similarity and the specific similarity measures employed by the systems. It was later demonstrated that *intrinsically* high-dimensional data with finite and well-defined means has a certain tendency for exhibiting hubness [59][51][60][61] and that changing the similarity measure can only reduce, but not entirely eliminate the problem. Boundary-less high-dimensional data does not necessarily exhibit hubness [47], but this case does not arise often in practical applications. The phenomenon of hubness will be discussed in more detail in Section 3.

The fact that neighbor occurrence distributions assume a certain shape in high-dimensional data gives us additional information which can be taken into account in algorithm design. Several simple *hubness-aware* kNN classification methods have recently been proposed in an attempt to tackle this problem explicitly. An instance-weighting scheme was first proposed in [59], which reduces the bad influence of hubs during voting. An extension of the fuzzy k-nearest neighbor framework was shown to be somewhat better on average [81], introducing the concept of *class-conditional hubness* of neighbor points and building an occurrence model which is used in classification. This approach was further improved by considering the information content of each neighbor occurrence [75]. An alternative approach in treating each occurrence as a random event was explored in [79], where it was shown that some form of Bayesian reasoning might be yet another feasible way of dealing with changes in the occurrence distribution. More details on the algorithms will be given in Section 3.4.

1.1. Project goal

The phenomenon of hubness has not been studied under the assumption of class imbalance in high-dimensional data and its impact on learning with kNN methods in skewed label distributions was unknown. This raises some concerns, as most real-world data is intrinsically high-dimensional and many important problems are also class-imbalanced.

The goal of this project was to examine the influence of hubness on learning under class imbalance, as well as test the performance and robustness of the existing hubness-aware kNN classification methods in order to evaluate whether they might be appropriate for handling such highly complex classification tasks.

Most misclassification is known to occur in borderline regions, where different classes meet and overlap. Class imbalance poses a problem only if a significant class overlap is present [56], so both of these factors must be considered carefully. In our experiments, we have generated several synthetic imbalanced high-dimensional data sets with severe overlap between different distributions in order to see if the hubness-aware algorithms are able to overcome this obstacle by relying on their occurrence models.

Real-world data labels are not always very reliable. Data is usually labeled by people and people make mistakes. This is why we decided to examine the influence

[1] Formally, in accordance with the existing definitions in the literature [59], we will say that *hubs* are points that have an occurrence count exceeding the mean (k) by more than two standard deviations of the neighbor occurrence distribution.

of very high levels of artificially induced mislabeling on the classification process.

1.2. Contributions

This research is the first attempt to correlate hubness as an aspect of the dimensionality curse with the problem of learning under class imbalance. Our analysis shows some surprising results, as our tests suggest that the minority class induces high misclassification of the majority class in many high-dimensional datasets, contrary to the low-dimensional case. We do not imply that this would always be the case, but it is an entirely new possibility that has so far been overlooked in algorithm design and needs to be carefully considered and taken into account.

We have performed an extensive experimental evaluation and shown that the recently proposed hubness-aware neighbor occurrence models achieve promising performance in several difficult types of classification problems: learning under class imbalance, mislabeling and class overlap in intrinsically high-dimensional data.

Our experiments suggest that the observed improvements stem from being able to better handle the difficult point types: borderline points, rare points and outliers. Additionally, the analysis reveals that, in most cases, the hubness-aware methods improve the recall of the majority class and the precision of the minority classes. This helps in improving the classification performance in presence of minority hubs.

Based on these encouraging results and the extensibility of the hubness-aware voting frameworks, we argue that it might be beneficial to combine them with the existing techniques for class imbalanced data classification, in order to improve system performance in high-dimensional data under the assumption of hubness.

2. Related work

2.1. Class imbalanced data classification

The problem of learning from imbalanced data has recently attracted attention of both industry and academia alike. Many classification algorithms used in real-world systems and applications fail to meet the performance requirements when faced with severe class distribution skews [31][18][39][5] and overlapping data distributions [56]. Various approaches have been developed in order to deal with this issue, including some forms of class under-sampling or over-sampling [9][24][30][45][91][4][25][46][93], , synthetic data generation [67], misclassification cost-sensitive techniques [49][68], decision trees [44], rough sets [42], kernel methods [89][34], ensembles [21][22] or active learning [15][14]. Novel classifier designs are still being proposed [48].

Many classification approaches for handling class imbalanced data are extensions of the basic kNN rule. Introducing an explicit bias towards the minority class is a standard strategy, either by introducing instance weights [65][86] or in some other way [92]. Even though such a bias might help in handling some minority classes in some datasets, global weighting approaches are known to face certain problems. Namely, performance depends mostly on the levels of imbalance in certain regions of the data space where different classes overlap, which often varies and is not constant throughout the data volume. Taking the local class distributions into account seems to be a somewhat more flexible approach [12].

The examplar-based kNN [41] introduces the concept of pivot minority points that are expanded to Gaussian balls, which makes them closer to other minority examples.

It has been suggested that the main problem when working with kNN under class imbalance lies in trying to estimate the prior class probabilities in the points of interest [43] and that somewhat more complex probabilistic models are required. When not much training data is available, semi-supervised approaches might be employed [26].

2.2. Hubness-aware methods

Hubness of the data is known to be detrimental to various machine learning and data mining tasks [59]. Several robust hubness-aware methods have recently been proposed for classification [59][81][79][75][76], instance selection for time series analysis [8], clustering [80][82], information retrieval [70], bug duplicate detection [69] and metric learning [73][74][63].

3. The hubness phenomenon

3.1. Emergence of hubs

Let $D = (x_1, y_1), (x_2, y_2), ..(x_n, y_n)$ be the data set, where each $x_i \in R^d$ resides in a high-dimensional Euclidean space [2] and $y_i \in c_1, c_2, ..c_C$ are instance labels. Denote by $D_k(x_i)$ the k-neighborhood defined by the

[2] For the sake of simplicity, we will restrict our discussion on the Euclidean case, as this is where the hubness phenomenon has been shown to arise as a consequence of distance concentration. It is, of course, possible for hubs to emerge in categorical or mixed datasets as well.

nearest neighbors of x_i. Also, let $N_k(x_i)$ be the number of k-occurrences (occurrences in k-neighbor sets) of x_i and by $N_{k,c}(x_i)$ the number of such occurrences in neighborhoods of elements from class c. We will also refer to $N_{k,c}$ as *class-conditional occurrence frequency*.

The phenomenon of *hubness* is expressed as an increased *skewness* of the k-neighbor occurrence distribution in high dimensions. This is illustrated in Figure 1 for the Gaussian mixture data. A certain number of hub-points occur very frequently and permeate most k-neighbor sets, while most other points occur very rarely. This constitutes a sort of an information loss, as most available information is very poorly utilized. We will refer to the rarely occurring points as *anti-hubs* or *orphans*.

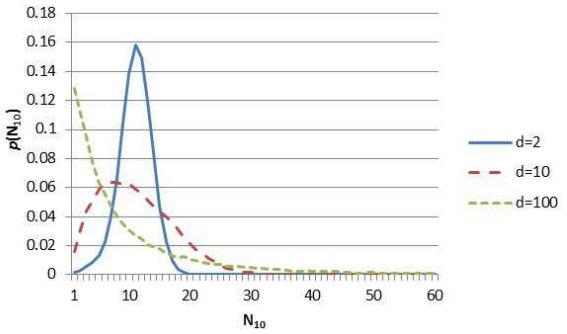


Figure 1: The change in the distribution shape of 10-occurrences (N_{10}) in i.i.d. Gaussian data with increasing dimensionality when using the Euclidean distance. The graph was obtained by averaging over 50 randomly generated data sets. Hub-points exist also with $N_{10} > 60$, so the graph displays only a restriction of the actual data occurrence distribution.

Dimensionality reduction can not entirely eliminate the problem [60]. Only by reducing the dimensionality well below the intrinsic dimensionality of the data it is possible to achieve a significant decrease in data hubness. This leads to an information loss that might also hurt system performance. It seems that taking the hubness into account while working with high-dimensional data might be a better practical decision.

Hubness is related to the distance concentration phenomenon, which is another well-known aspect of the dimensionality curse. The relative contrast between the maximal and the minimal distance observed on the data decreases with increasing dimensionality, thereby making it harder to distinguish between relevant and irrelevant points [20] [1]. Some researchers have even been inclined to question whether the concept of nearest neighbors is meaningful in high dimensional spaces [13].

Due to the concentration of distances, high-dimensional data lies approximately on hyper-spheres centered around cluster means. Data points closer to the means have a much higher probability of being included in k-neighbor sets. Most hubs emerge precisely in the central cluster regions and the neighbor occurrence frequency can be used as a good indicator of local point centrality in intrinsically high-dimensional data [82].

3.2. Good and bad hubness

In labeled data, some k-occurrences are *good* and some are *bad*. Occurrences are bad when there is label mismatch - when an observed point and its neighbor do not share the same label. Bad occurrences are, naturally, detrimental to kNN classification. Hub-points that frequently occur as bad neighbors are referred to as *bad hubs* and their overall bad occurrence frequency as *bad hubness*. So, by $N_k(x_i) = GN_k(x_i) + BN_k(x_i)$, hubness of a point is decomposed into good and bad hubness.

3.3. "How bad can it be?": motivating examples

All misclassification in nearest-neighbor methods is ultimately a result of label mismatches in k-neighbor sets. In very high dimensional data, bad hubness of individual points becomes more important, as hubs become more influential and have a higher impact on the classification process. We will illustrate the increased influence of hubs by considering a peculiar data set described in [71].

The data comprised a set of 2731 quantized image representations based on Haar wavelet features, belonging to 3 different categories, with some imbalance. An unexpected problem was encountered while varying the dimensionality in order to determine the optimal size of the visual word vocabulary. The kNN classification performance deteriorated significantly in higher dimensions and even ended up being worse than zero-rule. The results are shown in Table 1.

Table 1: Classification accuracy of kNN and four hubness-aware kNN algorithms (hw-kNN, NHBNN, h-FNN, HIKNN) on one compromised high dimensional 3-category image dataset.

Data set	5-NN	hw-kNN	NHBNN	h-FNN	HIKNN
ImNet3Err	21.2 ± 2.1	27.1 ± 11.3	59.5 ± 3.2 ∘	59.5 ± 3.2 ∘	59.6 ± 3.2 ∘

Subsequent analysis of the data had revealed the underlying causes behind the apparent drop in classifier performance. It turned out that exactly 5 images had

been assigned empty representations (zero vectors) due to an I/O error. Removing these 5 points was enough to raise the kNN classification accuracy from 21.2% to around 90%. It was astonishing that only 5 erroneous points (out of 2731) were enough to render kNN useless. It was determined that this was a consequence of hubness.

An increase in data dimensionality had resulted in these 5 points becoming prominent hubs in a clearly pathological way, due to an interplay of certain properties of the metric and the feature representation. This is illustrated in Figure 2. Most observed occurrences induced label mismatches, since the hub points belonged to the minority class.

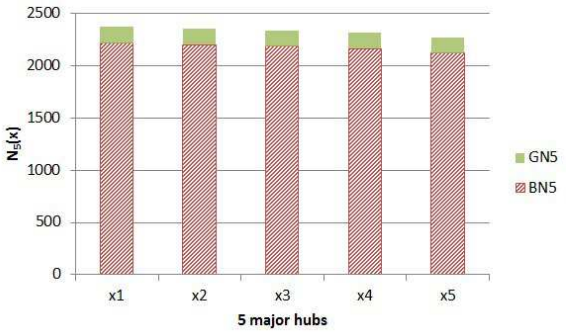


Figure 2: The 5 major hub-points in the data from the example analyzed in Table 1. We see that most of their hubness is in fact *bad hubness*. Hubs are not necessarily bad, but that is indeed often the case in practice.

This extreme example was a consequence of erroneous data processing and it might be argued that it does not reflect well the phenomena that occur in error-free data. However, it is usually not the erroneous points that become hubs in practice [58]. It is very difficult to predict where the hubs would emerge for a given data set.

In order to better illustrate that the minority class points might pose certain problems when they become hubs in high-dimensional data, we will briefly mention another real-world example, on WIKImage data [55, 78], a set of publicly available Wikipedia images. The distribution of bad hubs for a binary "person detection" problem (WM-l1) is shown in Figure 3. The majority class accounts for 79.5% of the data, yet it contains only a small portion of the bad hubs within the data, under several different feature representations: SIFT, SURF and ORB. This phenomenon will be discussed in more detail in Section 4.2, as it has significant consequences for data analysis.

An image data visualization tool has recently become

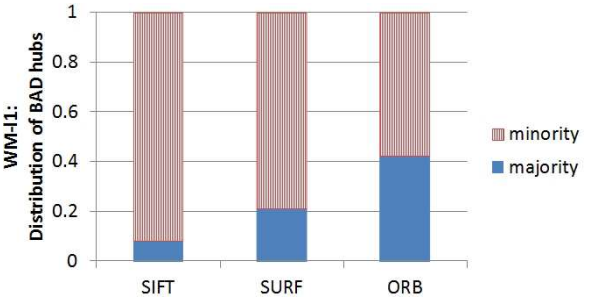


Figure 3: The proportion of bad image hubs in the majority and the minority class, for several different feature representations: SIFT, SURF and ORB.

available [77] that allows for quick and easy detection of critical hub points in the data and can be used to examine the nature of their influence. This allows the developers to detect and correct similar issues in their image search and object detection systems.

3.4. Hubness-aware classification

Several hubness-aware k-nearest neighbor methods have recently been proposed for robust high-dimensional data classification.

- **hw-kNN:** This weighting algorithm [59] is the simplest way to reduce the influence of bad hubs - they are simply assigned lower voting weights. Each neighbor vote is weighted by $e^{-h_b(x_i)}$, where $h_b(x_i)$ is the neighbor's standardized bad hubness score. All neighbors still vote by their own label (unlike in the algorithms considered below), which might prove disadvantageous sometimes, as implied by the example in Table 1.

- **h-FNN:** $u_c(x_i) = \frac{N_{k,c}(x_i)}{N_k(x_i)}$ (relative class hubness) can be interpreted as the fuzziness of the event that x_i had occurred as a neighbor. Hence, h-FNN [81] integrates class hubness into a fuzzy k-nearest-neighbor voting framework [38]. This means that the label probabilities in the point of interest are estimated as:

$$u_c(x) = \frac{\sum_{x_i \in D_k(x)} u_c(x_i)}{\sum_{x_i \in D_k(x)} \sum_{c \in C} u_c(x_i)} \qquad (1)$$

  Special care has to be given to anti-hubs and their occurrence fuzziness is estimated as the average fuzziness of points from the same class. Optional distance-based vote weighting is possible.

- **NHBNN:** Each k-occurrence can be treated as a random event. What NHBNN [79] does is that

it essentially performs a Naive-Bayesian inference from these k events.

$$p(y_i = c|D_k(x_i)) \propto p(y_i = c) \prod_{t=1}^{k} p(x_{it} \in D_k(x_i)|y_i = c). \quad (2)$$

Even though k-occurrences are highly correlated, NHBNN still offers some improvement over the basic kNN. Anti-hubs are, again, treated as a special case.

- **HIKNN:** Recently, class-hubness was also exploited in an information-theoretic approach to k-nearest neighbor classification [75]. Rare occurrences have higher self-information (Equation 3) and are favored by the algorithm. Hubs, on the other hand, lie closer to cluster centers and carry less local information relevant for the particular query.

$$p(x_{it} \in D_k(x)) \approx \frac{N_k(x_{it})}{N}$$
$$I_{x_{it}} = \log \frac{1}{p(x_{it} \in D_k(x))} \quad (3)$$

Occurrence self-information is used to define the absolute and relative relevance factors in the following way:

$$\alpha(x_{it}) = \frac{I_{x_{it}} - \min_{x_j \in D} I_{x_j}}{\log n - \min_{x_j \in D} I_{x_j}}, \quad \beta(x_{it}) = \frac{I_{x_{it}}}{\log N} \quad (4)$$

The final fuzzy vote combines the information contained in the neighbor's label with the information contained in its occurrence profile. The relative relevance factor is used for weighting the two information sources. This is shown in Equation 5

$$\bar{p}_k(y_i = c|x_{it} \in D_k(x_i)) = \frac{N_{k,c}(x_{it})}{N_k(x_{it})} = \bar{p}_{k,c}(x_{it})$$

$$p_k(y_i = c|x_{it}) \approx \begin{cases} \alpha(x_{it}) + (1 - \alpha(x_{it})) \cdot \bar{p}_{k,c}(x_{it}), & y_{it} = c \\ (1 - \alpha(x_{it})) \cdot \bar{p}_{k,c}(x_{it}), & y_{it} \neq c \end{cases} \quad (5)$$

The final class assignments are given by the weighted sum of these fuzzy votes, as shown in Equation 6. The distance weighting factor $d_w(x_{it})$ yields mostly minor improvements and can be left out in practice.

$$u_c(x_i) \propto \sum_{t=1}^{k} \beta(x_{it}) \cdot d_w(x_{it}) \cdot p_k(y_i = c|x_{it}) \quad (6)$$

NHBNN, HIKNN and h-FNN utilize class-conditional occurrence frequency estimates to perform classification based on the neighbor occurrence models. In high-dimensional data, this might be somewhat better than voting by label [75].

Computing all the k-neighbor sets accurately in the training phase could sometimes become overly time-consuming when working with big data. In such cases, approximate kNN graph construction methods can be considered instead. One such approach [10] was analyzed in [75] and it was shown that hubness-aware algorithms outperform the kNN baseline on high-dimensional data even if the entire graph is approximated in linear time (instead of $\Theta(dn^2)$) and that very good approximations are usually available with a modest time investment ($\Theta(dn^{1.2})$ or $\Theta(dn^{1.4})$).

4. Hypotheses and Methodology

4.1. Bad hubness in mislabeled data

Obviously, mislabeled and noisy instances both contribute to the overall bad hubness of the data. The case discussed in Table 1 and Figure 2 is a rather extreme example of how much damage can be caused by noisy measurements in many dimensions. The impact of erroneous labels and inaccurate numeric values is the highest precisely when they are present in hub-points. Hubs can easily spread both correct and incorrect/corrupted information.

Unfortunately, as we have already seen, there is no guarantee that errors will be contained among the rarely occurring examples. The exact distribution of hubness among data points depends heavily on the particular choice of feature representation and similarity measure and is, in general, very hard to predict.

Hypothesis: By using the neighbor occurrence models learned on the training data, the hubness-aware kNN algorithms should in most cases be able to cope with bad hubness caused by mislabeling and/or noisy data.

A neighbor occurrence model is any model that can be used for predicting the probability of a certain point occurring as a neighbor in a kNN set of a query

point that belongs to a specific class. In our experiments, these probabilities are directly estimated from the kNN graph on the training data, based on the class-conditional occurrence frequencies of all the training points.

In our experiments we have focused on the former, as it is easier to evaluate. Noise, on the other hand, can take various forms (Gaussian, non-Gaussian), be present in various intensities and distributed in various ways across the data.

An illustrative example explaining how the class-conditional occurrence information can be used in order to help with dealing with mislabeled data points is given in Figure 4.

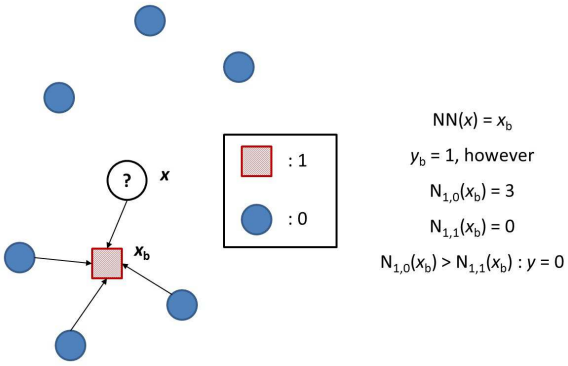


Figure 4: An illustrative example. Point under consideration is marked by "x" and NN(x) = x_b. However, x_b is a mislabeled point. Reasoning by the 1-NN rule, we would conclude that $y = 1$, which is probably wrong, looking at the data. On the other hand, if we were to reason according to *class hubness*, we would infer $y = 0$, because x_b was previously a neighbor of instances labeled "0". This shows how learning from previous occurrences can help in making the nearest neighbor classifiers less prone to errors in mislabeled data sets.

Mislabeled examples are not uncommon in large, complex systems. Detecting and correcting such data points is not an easy task and many correction algorithms have been proposed in an attempt to solve the problem [29][27][83]. Regardless, some errors always remain in the data. This is why robustness to mislabeling is very important in classification algorithms.

4.2. Bad hubness under class imbalance

The usual interpretation of the bad influence of class imbalanced data on kNN classification is that the majority class points would often become neighbors of the minority class examples, due to the relative difference in densities between different categories. As neighbors, they would often cause misclassification of the minority class. Consequently, the methods which are being proposed for imbalanced data classification and (briefly outlined in Section 2.1), are focused primarily on rectifying this by improving the overall classifier performance on the minority class. Naturally, something has to be sacrificed in return and usually it is the recall of the majority class.

This is certainly reasonable. In many real-world problems the misclassification cost is much higher for the minority class. Some well known examples include cancer diagnosis, oil spill recognition, earthquake prediction, terrorist detection, etc. However, things are not so simple as they might seem. Often enough, the cost of misclassifying the majority class is almost equally high. In fraud detection [16][17], accusing innocent people of fraud might lose customers for the companies involved and incur a significant financial loss. Even in breast cancer detection it has recently been shown that the current diagnostic techniques lead to significant over-diagnosis of cancer cases [37]. This leads to many otherwise healthy women being admitted for treatment and subjected to various drug courses and/or operating procedures.

In Section 3.3, we have seen how things may go awry if the minority instances turn into bad hubs. This can be caused by noise or mislabeling, but it is not necessarily the case in practice. Problems might arise in completely 'clean' datasets as well.

Hypothesis: The examples outlined in Section 3.3 had led us to hypothesize that, in intrinsically high-dimensional data, the primary concern should be the *minority class hubs causing misclassification of the majority class points* instead of the other way around.

This is exactly the opposite of what most imbalanced data classification algorithms are trying to solve. It is a very important observation, especially because most of the data that is being automatically processed and mined is in fact high-dimensional and exhibits hubness, whether it is text, images, video, time series, etc. [59][60][71][62]

If our hypothesis were to hold, this would pose a new challenge for the imbalanced data classification algorithm design, as future algorithms would need to incorporate mechanisms of improving both the minority and the majority class recall at the same time. This is non-trivial problem.

Such a phenomenon is easy to overlook, as it is highly counterintuitive. In lower dimensional data, most misclassification in imbalanced data sets occurs in border regions where classes overlap and have different densities. As the minority classes usually have a lower density in those regions, they get misclassified more often. However, most misclassification in high-dimensional

data is caused by bad hubs - and they can emerge in unpredictable places. As point-wise occurrence frequencies depend heavily on the choice of metric and feature representation, the arising structure of influence does not necessarily reflect the semantics of the data well. In fact, hubs often become semantic singularities and places where the semantic consistency of the kNN structure becomes most compromised [59][60][61].

With that in mind, consider a simplified example given in Figure 5. The 1-NN misclassification rate for a particular hub-point would trivially be maximized if its label were to match the minority class in its occurrence profile. In the more general case of kNN, these label mismatches do not necessarily induce misclassification, but a cumulative effect of several co-occurring hub points would have the same negative outcome. If we were to think of hubness as a purely geometric property that is not well aligned with data semantics, we would expect the distribution of classes in the occurrence profiles of major hubs to tend towards the (local) class priors. In those cases, the minority class in the occurrence profile would often match the overall minority class. This means that most label mismatches would be caused by the minority hubs.

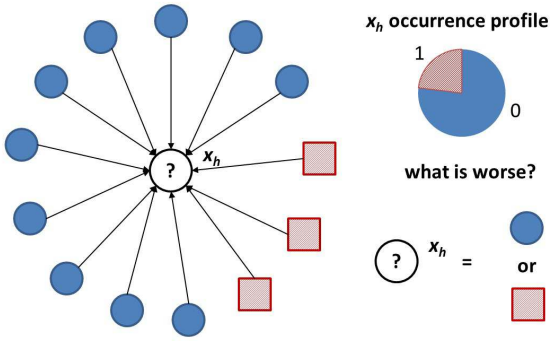


Figure 5: An illustrative example. x_h is a hub, neighbor to many other points. There is a certain label distribution among its reverse nearest neighbors, defining the occurrence profile of x_h. It is obvious that most damage would be done to the classification process by x_h if it were to share the label of the minority part of its reverse neighbor set. On average, we would expect this to equal the overall minority class in the data. This suggests that minority hubs might have a higher average tendency to become bad hubs and that this might prove to be, in general, quite detrimental to classifier performance.

4.3. Methodology

We propose to analyze the interplay between hubness and class imbalance in several steps. First, we perform a detailed analysis of class-to-class occurrence distributions and the kNN confusion matrices in order to detect the principal gradients of misclassification. We proceed by examining the distributions of different types of points among different classes. This includes a characterization of points into hubs, regulars and anti-hubs [59], as well as the characterization of points into safe, borderline, rare and outliers [52]. Points are considered safe if 4 or 5 of their 5-NNs belong to their class, borderline if it is 2 or 3, rare if only 1 neighbor share the same label and outliers otherwise. Finally, we evaluate the performance of the hubness-aware classification approaches by comparisons to the baseline kNN and characterize the nature of their improvements by examining the improvements in the precision and recall of both the majority and minority class or classes. Both the accuracy and the F_1-score [88] will be used to evaluate the overall aggregate classifier performance.

We propose to analyze the influence of mislabeling on the hubness-aware classification process by randomly introducing mislabeling into the training data during the cross-validation folds while testing the algorithms on existing real-world datasets. By observing how the classification performance changes for different mislabeling levels, we are able to estimate the robustness of different approaches. This testing functionality is fully supported in the Hub Miner library (`http://ailab.ijs.si/nenad_tomasev/hub-miner-library/`), which we have used in our experiments.

A general approach to hubness-aware classification is outlined in Figure 6.

5. Experiments and Discussion

In order to test the above stated hypotheses, we performed extensive experimental evaluation.

The results have been structured in the following way: Section 5.2 examines the role of minority hubs in class imbalanced kNN classification and presents a series of experiments that support our initial hypothesis stated in Section 4.2. Section 5.3 deals with robustness to high mislabeling levels and confirms our hypothesis that the neighbor occurrence models learned on the training data can increase the kNN classification performance under high mislabeling levels. Section 5.4 examines algorithm performance under severe class overlap in high-dimensional class imbalanced Gaussian mixtures.

5.1. Data Overview

In our experiments we have used both low hubness data sets (mostly balanced) and high-hubness image data sets (mostly imbalanced).

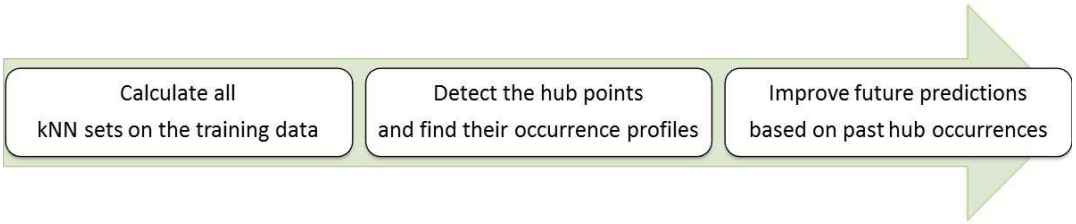


Figure 6: The hubness-aware analytic framework: learning from past neighbor occurrences.

Table 2: Summary of the real-world data sets. Each data set is described by the following set of properties: size, number of features (d), number of classes (c), skewness of the 5-occurrence distribution (S_{N_5}), the percentage of *bad* 5-occurrences (BN_5), the degree of the largest hub-point ($\max N_5$), relative imbalance of the label distribution (RImb) and the size of the majority class ($p(c_M)$)

Data set	size	d	C	S_{N_5}	BN_5	$\max N_5$	RImb	$p(c_M)$
diabetes	768	8	2	0.19	32.3%	14	0.30	65.1%
ecoli	336	7	8	0.15	20.7%	13	0.41	42.6%
glass	214	9	6	0.26	25.0%	13	0.34	35.5%
iris	150	4	3	0.32	5.5%	13	0	33.3%
mfeat-factors	2010	216	10	0.83	7.8%	25	0	10%
mfeat-fourrier	2000	76	10	0.93	19.6%	27	0	10%
ovarian	2534	72	2	0.50	15.3%	16	0.28	64%
segment	2310	19	7	0.33	5.3%	15	0	14.3%
sonar	208	60	2	1.28	21.2%	22	0.07	53.4%
vehicle	846	18	4	0.64	35.9%	14	0.02	25.8%
ImNet3	2731	416	3	8.38	21.0%	213	0.40	50.2%
ImNet4	6054	416	4	7.69	40.3%	204	0.14	35.1%
ImNet5	6555	416	5	14.72	44.6%	469	0.20	32.4%
ImNet6	6010	416	6	8.42	43.4%	275	0.26	30.9%
ImNet7	10544	416	7	7.65	46.2%	268	0.09	19.2%
ImNet3Imb	1681	416	3	3.48	17.2%	75	0.72	81.5%
ImNet4Imb	3927	416	4	7.39	38.2%	191	0.39	54.1%
ImNet5Imb	3619	416	5	9.35	41.4%	258	0.48	58.7%
ImNet6Imb	3442	416	6	4.96	41.3%	122	0.46	54%
ImNet7Imb	2671	416	7	6.44	42.8%	158	0.46	52.1%

The former were taken from the UCI repository (http://archive.ics.uci.edu/ml/datasets.html), the latter from the ImageNet public collection (http://www.image-net.org/). More info on the image data feature representation is available in [75][71].

From the first five image data sets we removed a random subset of instances from all the minority classes in order to make the data even more imbalanced for the experiments. The relevant properties of the data sets are given in Table 2. The listed UCI data sets were mostly not imbalanced and we included the results in Table 3 only for comparison with the mislabeled case which follows in Section 5.3. The classification accuracies given in Table 3 have already been reported in our earlier work [75][73] and will serve as a starting point for further analysis.

All classification tests were performed as 10-times 10-fold cross-validation. Corrected re-sampled t-test was used to detect statistical significance [6]. Manhattan metric was used in all real-world experiments, while the Euclidean distance was used for dealing with Gaussian mixtures in Section 5.4. All feature values in UCI and ImageNet data were normalized to the $[0, 1]$ range. All the hubness-aware algorithms were tested under their default parameter configurations, according to what was specified in the respective papers.

Table 3: Experiments on UCI and ImageNet data. Classification accuracy is given for kNN, hubness-weighted kNN (hw-kNN), hubness-based fuzzy nearest neighbor (h-FNN), naive hubness-Bayesian k-nearest neighbor (NHBNN) and hubness information k-nearest neighbor (HIKNN). All experiments were performed for $k = 5$. The symbols •/○ denote statistically significant worse/better performance ($p < 0.05$) compared to kNN. The best result in each line is in bold.

Data set	kNN	hw-kNN	h-FNN	NHBNN	**HIKNN**
diabetes	67.8 ± 3.7	75.6 ± 3.7 ○	75.4 ± 3.2 ○	73.9 ± 3.4 ○	**75.8 ± 3.6** ○
ecoli	82.7 ± 4.2	86.9 ± 4.1 ○	**87.6 ± 4.1** ○	86.5 ± 4.1 ○	87.0 ± 4.0 ○
glass	61.5 ± 7.3	65.8 ± 6.7	67.2 ± 7.0 ○	59.1 ± 7.5	**67.9 ± 6.7** ○
iris	95.3 ± 4.1	**95.8 ± 3.7**	95.3 ± 3.8	95.6 ± 3.7	95.4 ± 3.8
mfeat-factors	94.7 ± 1.1	96.1 ± 0.8 ○	95.9 ± 0.8 ○	95.7 ± 0.8 ○	**96.2 ± 0.8** ○
mfeat-fourier	77.1 ± 2.2	81.3 ± 1.8 ○	82.0 ± 1.6 ○	**82.1 ± 1.7** ○	**82.1 ± 1.7** ○
ovarian	91.4 ± 3.6	92.5 ± 3.5	93.2 ± 3.5	93.5 ± 3.3	**93.8 ± 2.9**
segment	87.6 ± 1.5	88.2 ± 1.3	88.8 ± 1.3 ○	87.8 ± 1.3	**91.2 ± 1.1** ○
sonar	82.7 ± 5.5	83.4 ± 5.3	82.0 ± 5.8	81.1 ± 5.6	**85.3 ± 5.5**
vehicle	62.5 ± 3.8	65.9 ± 3.2 ○	64.9 ± 3.6	63.7 ± 3.5	**67.2 ± 3.6** ○
ImNet3	72.0 ± 2.7	80.8 ± 2.3 ○	**82.4 ± 2.2** ○	81.8 ± 2.3 ○	82.2 ± 2.0 ○
ImNet4	56.2 ± 2.0	63.3 ± 1.9 ○	**65.2 ± 1.7** ○	64.6 ± 1.9 ○	64.7 ± 1.9 ○
ImNet5	46.6 ± 2.0	56.3 ± 1.7 ○	**61.9 ± 1.7** ○	61.8 ± 1.9 ○	60.8 ± 1.9 ○
ImNet6	60.1 ± 2.2	68.1 ± 1.6 ○	69.3 ± 1.7 ○	69.4 ± 1.7 ○	**69.9 ± 1.9** ○
ImNet7	43.4 ± 1.7	55.1 ± 1.5 ○	**59.2 ± 1.5** ○	58.2 ± 1.5 ○	56.9 ± 1.6 ○
ImNet3Imb	72.8 ± 2.4	87.7 ± 1.7 ○	87.6 ± 1.6 ○	84.9 ± 1.9 ○	**88.3 ± 1.6** ○
ImNet4Imb	63.0 ± 1.8	68.8 ± 1.5 ○	69.9 ± 1.4 ○	69.4 ± 1.5 ○	**70.3 ± 1.4** ○
ImNet5Imb	59.7 ± 1.5	63.9 ± 1.8 ○	64.7 ± 1.8 ○	63.9 ± 1.8 ○	**65.5 ± 1.8** ○
ImNet6Imb	62.4 ± 1.7	69.0 ± 1.7 ○	**70.9 ± 1.8** ○	68.4 ± 1.8 ○	70.2 ± 1.8 ○
ImNet7Imb	55.8 ± 2.2	63.4 ± 2.0 ○	64.1 ± 2.3 ○	63.1 ± 2.1 ○	**64.3 ± 2.1** ○
AVG	69.77	75.40	76.38	75.23	**76.75**

5.2. Class imbalanced data

While analyzing the connection between hubness and class imbalance we will focus on the image datasets shown in the lower half of Table 2. To measure the imbalance of a particular dataset, we will observe two quantities: $p(c_M)$, which is the relative size of the majority class - and relative imbalance (RImb) of the label distribution which we define as the normalized standard deviation of the class probabilities from the absolutely homogenous mean value of $1/c$ for each class. In other words, $\mathrm{RImb} =$

$\sqrt{(\sum_{c\in C}(p(c)-1/C)^2)/((C-1)/C)}$.

Unbalancing the original five datasets (ImNet3-ImNet7) seems not to have increased the overall difficulty in terms of the achieved classification accuracy and the total induced bad hubness (Table 2). As bad hubness is not directly caused by class imbalance and results as an interplay of various contributing factors, this is not altogether surprising.

ImNetImb data sets were selected via random undersampling and it is always difficult to predict the effects of data reduction on hubness. Removing anti-hubs makes nearly no difference, but removing hub-points certainly does. After a hub is removed and all neighbor lists are recalculated, the occurrence profiles of many other hub-points change, as they fill in the thereby released 'empty spaces' in neighbor lists where the removed hub participated.

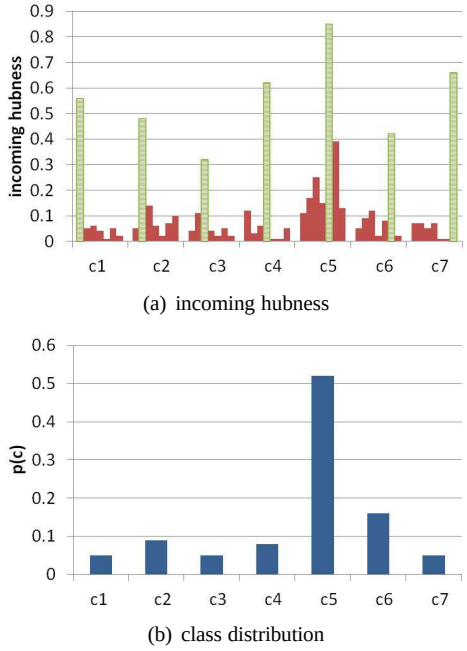
5.2.1. Correlating bad hubness and class imbalance

Consider a class-to-class k-occurrence matrix for the ImNet7Imb dataset that is given in Table 4. Each row contains average outgoing hubness from one category to another. On the diagonal we are able to see the percentage of occurrences of points from each category in neighborhoods of points from the same category (i.e. good hubness). We see that in ImNet7Imb the majority class has highest relative good hubness. It also seems that most of the bad hubness expressed by the minority classes is directed towards the majority class. We can see this more clearly by observing the graph of *incoming hubness*, shown in Figure 7. In this case, most bad hubness is generated by the minority classes and most of this bad influence is directed towards the majority class (c5).

Table 4: Class-to-class hubness between different classes in ImNet7Imb for $k = 5$. Each row contains the outgoing occurrence rate towards other categories. For instance, in the first row we see that only 56% of all neighbor occurrences of points from the first class are in the neighborhoods of elements from the same class. The diagonal elements (self-hubness) are given in bold, as well as the majority class.

	p(c)	c1	c2	c3	c4	c5	c6	c7
c1	0.05	**0.56**	0.05	0.04	0.12	0.11	0.05	0.07
c2	0.08	0.05	**0.48**	0.11	0.03	0.17	0.09	0.07
c3	0.05	0.06	0.14	**0.32**	0.06	0.25	0.12	0.05
c4	0.08	0.04	0.06	0.04	**0.62**	0.15	0.02	0.07
c5	**0.52**	0.01	0.02	0.02	0.01	**0.85**	0.08	0.01
c6	0.17	0.05	0.07	0.05	0.01	0.39	**0.42**	0.01
c7	0.05	0.02	0.10	0.02	0.05	0.13	0.02	**0.66**

Since individual label mismatches do not necessarily cause misclassification, analyzing the class-to-class k-occurrence matrix is in itself not sufficient. The kNN confusion matrix helps in analyzing the actual misclassification gradients and the confusion matrix for ImNet7Imb data is given in Table 5, generated by averaging after 10 runs of 10-fold cross validation.



(a) incoming hubness

(b) class distribution

Figure 7: The *incoming hubness* towards each category expressed by other categories in the data shown for ImNet7Imb data set. The 7 bars in each group represent columns of the class-to-class k-occurrence Table 4. Neighbor sets were computed for $k = 5$. We see that most hubness expressed by the minority classes is directed towards the majority class. This gives some justification to our hypothesis that in high-dimensional data with hubness it is mostly the minority class instances that cause misclassification of the majority class and not the other way around.

Several things in Table 5 are worth noting. First of all, the majority class FP rate is lower than its FN rate, which means that more errors are made on average by misclassifying the majority class points than by misclassifying the minority class points into the majority class. Also, the highest FP rate is not achieved by the majority class, but rather by one of the minority classes - $c6$. Both of these observations are very important, as we have already mentioned that there are various scenarios where the cost of misclassifying the majority class points is quite high. [16][17][37]

The previously discussed correlation between relative class size and bad hubness can be established also by inspecting a collection of imbalanced data sets (ImNet3Imb-ImNet7Imb) at the same time. Pearson correlation between class size and class-conditional bad

Table 5: The average 5-NN confusion matrix for ImNet7Imb data after 10-times 10-fold cross-validation. Each row displays how elements of a particular class were assigned to other classes by the 5-NN classifier. The overall number of false negatives (FN) and false positives (FP) for each category is calculated. The results for the majority class are in bold.

	p(c)	c1	c2	c3	c4	c5	c6	c7	FN
c1	0.05	42.9	13.5	3.8	11.8	**6.2**	60.7	1.1	97.1
c2	0.08	22.8	48.0	15.3	8.9	**54.9**	77.1	0.0	179.0
c3	0.05	8.9	21.0	13.0	3.3	**25.6**	55.2	0.0	114.0
c4	0.08	44.0	6.0	2.0	100.5	**15.5**	43.0	0.0	110.5
c5	**0.52**	**78.5**	**36.7**	**25.9**	**21.9**	**1028.1**	**200.9**	**0.0**	**363.9**
c6	0.17	16.9	19.1	10.2	4.3	**142.9**	254.6	0.0	193.4
c7	0.05	17.9	8.3	6.1	12.1	**41.0**	36.9	3.7	122.3
	FP	189.0	104.6	63.3	62.3	**286.1**	473.8	1.1	

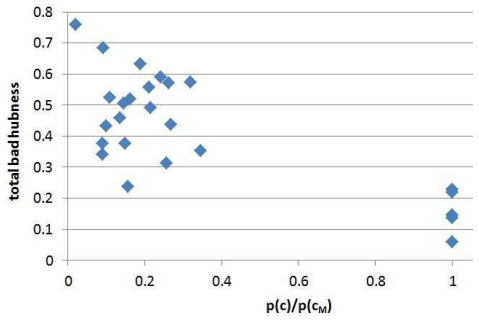


Figure 8: Average bad hubness exhibited by each class from data sets ImNet3Imb-ImNet7Imb plotted against relative class size ($p(c)/p(c_M)$). We see that the minority classes exhibit on average much higher bad hubness than the majority classes.

hubness is -0.76 when taken for $k = 5$. This implies that there might be a very strong negative correlation between the two quantities and that the minority classes indeed exhibit high bad hubness relative to their size. A plot of all $(\frac{p(c)}{p(c_M)}, BN_5(c))$ is shown in Figure 8.

In Section 4.2, we have conjectured that bad hubs among the minority points are expected to have higher bad hubness on average. In order to check this hypothesis, we have examined class distributions among different types of points, namely: hubs, anti-hubs and bad hubs. Similarly to hubs [60], bad hubs were formally defined as those points that have an unusually high bad occurrence frequency: $\{x : BN_k(x) > \mu_{BN_k(x)} + 2 \cdot \sigma_{BN_k(x)}\}$. We took as many anti-hubs as hub-points, by taking those with least occurrences from the ordered list.

Class distributions among these types of points can be compared to the prior distribution over all data points. The comparison for ImNet7Imb data is shown in Figure 9. Similar trends are present in the rest of the image data sets, as well. We see that the class distribution is entirely different for different types of points. This needs to be taken into account when modeling the data. Most importantly, we see that in this data set, all top bad hubs come from the minority classes, in accordance with our hypothesis. In the rest of the examined image data sets the situation is very similar, though the majority class is naturally not always at 0% among the top hubs, but it is always less frequent than among all points combined.

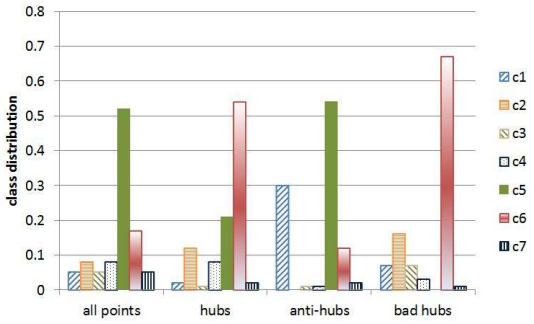


Figure 9: Distribution of classes among different types of points in ImNet7Imb data: hubs, anti-hubs and bad hubs. We see that there are nearly no majority class points among the top bad hubs in the data. Data points of class c6 exhibit highest bad hubness, which explains the high FP rate observed in Table 5

By considering the anti-hub distribution in Figure 9, we might also gain some insight into the outlier structure of the data. Previous research [59][60][61][80] suggests that outliers tend to be anti-hubs in the data, though anti-hubs are not always outliers. The fact that class $c1$ contributes so much to anti-hubs suggests that this particular minority class consists mostly of outliers.

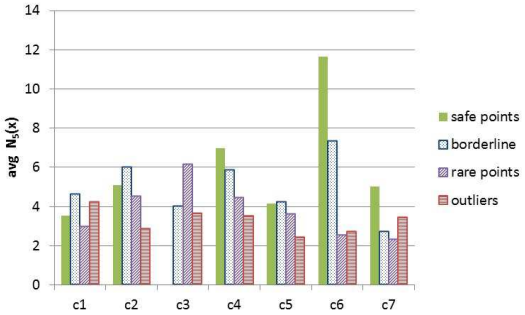


Figure 10: Average hubness of different point types in different categories. Safe points are not consistently the points of highest hubness. Quite frequently borderline examples and even rare points of the minority classes end up being neighbors to other points. This also means that less typical points exhibit a substantial influence on the classification process.

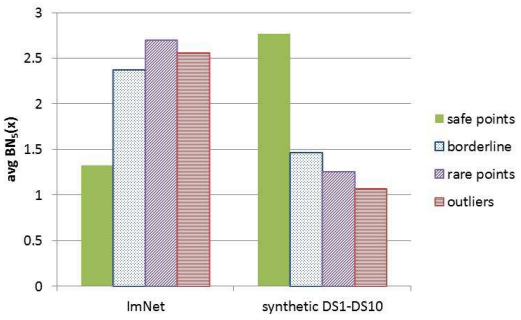


Figure 11: Average 5-NN bad hubness of different point types shown both for ImNet and high-dimensional synthetic Gaussian mixtures given in Table 9, Section 5.4. We give both bad hubness distributions here for easier comparison. It is clear that they are quite different. In the analyzed image data, most bad influence is exhibited by atypical class points (borderline examples, rare points, outliers), while most bad influence in the Gaussian mixture data is generated by safe points. The latter is quite counterintuitive, as we usually expect for such typical points to be located in the inner regions of class distributions.

In Figure 10 we can see the distribution of occurrence frequencies among safe points, borderline points, rare points and outliers given separately for each category of the ImNet7Imb data set. The results indicate a strong violation of the cluster assumption, as point hubness is closely linked to within-cluster centrality [80][82]. High hubness of borderline points indicates that data clusters are not homogenous with respect to the label space. Indeed, our initial tests have shown that this data does not cluster well. Another thing worth noting is that points that we usually think of as reliable might have a detrimental influence on the classification process, which is clear from examining the hubness/bad hubness distribution across different point types for $c6$, which has a high overall bad hubness and FP rate. It is precisely the safe points that exhibit both the highest hubness (AVG. 11.66) and the highest bad hubness (AVG. 6.63). This is yet another good illustration of the differences between low-dimensional and high-dimensional data. Intuitively, we would expect the safe points to be located in the innermost part of the class distribution space and not to become neighbors to *many* other points from different categories. This is precisely what happens here and is yet another slightly counterintuitive result.

Bad occurrence distributions summarized in Figure 11 illustrate that different underlying bad hub structures exist in different types of data. In the analyzed image data (ImNet3-7, ImNetImb3-7), the previously described pathological case of safe/inner points arising as top bad hubs in the data is still more an exception than a rule, while in high-dimensional Gaussian mixtures it becomes a dominating feature. Further analysis of the synthetic datasets is given in Section 5.4, where class overlap is discussed.

5.2.2. Hubness-aware classification under class imbalance

In order to learn more about the way in which the hubness-aware classifiers handle the minority and the majority class points, we have performed an in-depth analysis of the classification results summarized in Table 3, by focusing on certain imbalanced image datasets.

Unbalancing the original five datasets (ImNet3-ImNet7) seems not to have increased the overall difficulty in terms of the achieved classification accuracy and the total induced bad hubness (Table 2). As bad hubness is not directly caused by class imbalance and results as an interplay of various contributing factors, this is not altogether surprising.

ImNetImb data sets were selected via random undersampling and it is always difficult to predict the effects of data reduction on hubness. Removing anti-hubs makes nearly no difference, but removing hub-points certainly does. After a hub is removed and all neighbor lists are recalculated, the occurrence profiles of many other hub-points change, as they fill in the thereby released 'empty spaces' in neighbor lists where the removed hub participated.

An analysis of precision and recall for each class separately is shown in Table 6, for the ImNet7Imb dataset. It can be see that all hubness-aware algorithms improve on average both precision and recall for most individual categories.

Table 6: Precision and recall for each class and each method separately on ImNet7Imb data set. Values greater or equal to the score achieved by kNN are given as bold. The last column represents the Spearman correlation between the improvement over kNN in precision or recall and the size of the class. In other words, $\mathrm{corrImp} = corr(\frac{p(c)}{\max p(c)}, \mathrm{improvement})$.

method	measure	c_1	c_2	c_3	c_4	c_5	c_6	c_7		
priors:		0.05	0.08	0.05	0.08	0.52	0.17	0.05	AVG	corrImp
kNN	precision	0.20	0.32	0.18	0.62	0.78	0.35	0.31	0.39	
	recall	0.31	0.21	0.10	0.47	0.74	0.57	0.03	0.35	
hw-kNN	precision	**0.46**	**0.39**	**0.28**	**0.72**	**0.79**	**0.41**	**0.58**	**0.52**	-0.96
	recall	0.30	**0.30**	**0.19**	**0.73**	**0.81**	**0.59**	**0.17**	**0.44**	-0.43
h-FNN	precision	**0.65**	**0.46**	**0.37**	**0.72**	0.69	**0.44**	**0.76**	**0.58**	-0.86
	recall	0.18	0.19	0.09	**0.73**	**0.92**	0.43	**0.12**	**0.38**	-0.07
NHBNN	precision	**0.36**	**0.37**	**0.22**	**0.62**	**0.79**	**0.47**	**0.45**	**0.47**	-0.39
	recall	**0.43**	**0.22**	**0.22**	**0.80**	**0.81**	0.50	**0.20**	**0.45**	-0.68
HIKNN	precision	**0.55**	**0.45**	**0.30**	**0.74**	**0.78**	0.40	**0.67**	**0.55**	-0.75
	recall	0.24	**0.23**	**0.14**	**0.74**	**0.84**	**0.61**	**0.17**	**0.42**	0.0

To further analyze the structure of this improvement, an analysis of the correlation between class size and

the improvement in precision or recall was performed for each tested algorithm. As it turns out, hubness-aware algorithms improve precision much more consistently than recall - and this improvement has high negative correlation with relative class size. In other words, *hubness-aware classification improves the precision of minority class categorization*, and the improvement grows for smaller and smaller classes. Actually, NHBNN is an exception, as it soon becomes clear that it behaves differently. A closer examination reveals that the recall of the majority class is improved in all the imbalanced data sets, except when NHBNN is used. This is shown in Figure 12. On the contrary, NHBNN is best at improving the minority class recall, which is not always improved by other hubness-aware algorithms, as shown in Figure 13.

HIKNN is essentially an extension of the basic h-FNN algorithm, so it is interesting to observe such a clear difference between the two. h-FNN is always better at improving the majority class recall, while HIKNN achieves better overall minority class recall. Both algorithms rely on neighbor occurrence models, but HIKNN derives more information directly from a neighbor's label and this is why it has a higher specificity bias, which is reflected in the results. The results of NHBNN, on the other hand, are not so easy to interpret. It seems that the Bayesian modeling of the neighbor-relation differs from the fuzzy model in some subtle way.

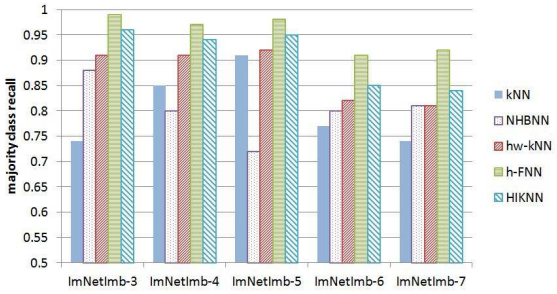


Figure 12: A comparison of majority class recall achieved by both kNN and the hubness-aware classification algorithms on five imbalanced image data sets. Improvements are clear in hw-kNN, h-FNN and HIKNN.

Observing precision and recall separately does not allow us to rank the algorithms according to their relative performance, so we will rank them according to the F_1-measure scores [88]. We report the micro- and macro-averaged F_1-measure (F_1^μ and F_1^M, respectivelly) for each algorithm over the imbalanced data sets in Table 7. Micro-averaging is affected by class imbalance, so the macro-averaged F_1 scores ought to be preferred. In this case it makes no difference. The results show that all

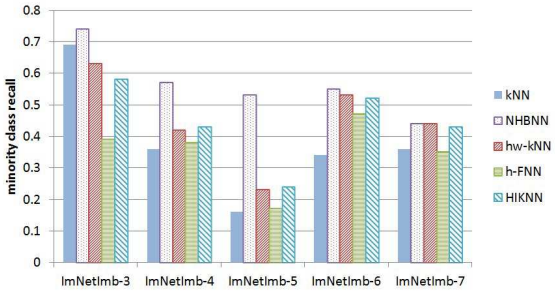


Figure 13: A comparison of the cumulative minority class recall (micro-averaged) achieved by both kNN and the hubness-aware classification algorithms on five imbalanced image data sets. NHBNN seems undoubtedly the best in raising the minority class recall. Other hubness-aware algorithms offer some improvements on ImNetImb4-7, but under-perform at ImNet3Imb data. In this case, HIKNN is better than h-FNN on all data sets, just as h-FNN was constantly slightly better than HIKNN when raising the majority class recall.

of the hubness-aware approaches improve on the basic kNN in terms of both F_1^μ and F_1^M. NHBNN achieves the best F_1-score, followed by HIKNN and hw-kNN, while h-FNN is, in this case, the least balanced of all the considered hubness-aware approaches.

Table 7: Micro- and macro-averaged F_1 scores of the classifiers on the imbalanced data sets. The best score in each line is in bold.

	kNN	hw-kNN	h-FNN	NHBNN	HIKNN
F_1^μ	0.61	0.68	0.66	**0.70**	0.69
F_1^M	0.43	0.52	0.47	**0.57**	0.53

In order to see if the hubness-aware approaches actually achieve their improvements by utilizing the learned occurrence information about the minority hubs, we have performed additional tests. We have tracked which point-wise class predictions improve over the baseline kNN and which predictions end up being worse, averaged over the 10-times 10-fold cross-validation. In both cases, we checked for presence of hubs of different classes in the kNN sets of individual points for each test run separately. For each hub point, all the improvements and deteriorations in prediction quality over the set of its reverse neighbors have been summed in order to estimate the overall change in prediction quality in the kNN sets where the hub point occurs. The results for the ImNet7Imb dataset are shown in Figure 14. Similarly, we can focus on bad hubs specifically and the distribution of average improvements in prediction quality in presence of bad hubs is shown in Figure 15.

In both cases, the improvements are most pronounced for class $c6$, which is not the majority class and is the class with highest bad hubness on the dataset. This suggests that the improvements are indeed obtained by ex-

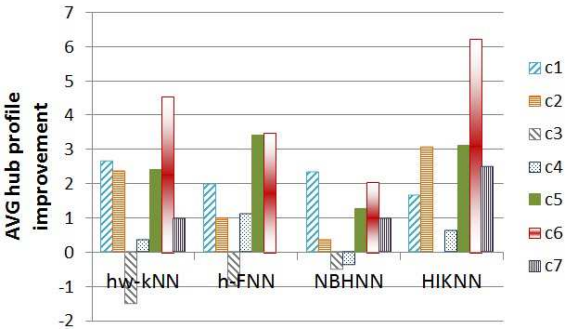


Figure 14: The average number of improvements in prediction quality among the reverse neighbors of hubs points, on ImNet7Imb data.

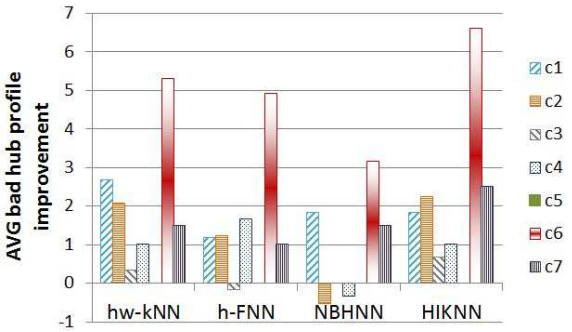


Figure 15: The average number of improvements in prediction quality among the reverse neighbors of bad hubs points, on ImNet7Imb data.

ploiting the relevant hubness information.

The property of hw-kNN, h-FNN and HIKNN of significantly raising the recall of the majority class is a very useful one, especially since they are able to do so without harming the minority class recall. This helps with handling class imbalanced data under the assumption of hubness.

As most standard approaches to learning under class imbalance aim in the opposite direction, it might be useful to consider hybrid approaches in the future, by combining both types of prediction strategies. As the hubness-aware classification methods mostly modify the final voting, they can easily be combined with over-sampling/under-sampling [9][30][45][91][4][40], instance weighting [66] or examplar-based learning [41]. They can also, in principle, support cost-sensitive learning, unlike many other kNN methods. This is made possible by the occurrence model, as not every occurrence has to be given the same weight when calculating $N_{k,c}(x)$. Distance-weighted occurrence models were already considered [72], but cost-sensitive occurrence models are certainly an option that we wish to explore in our future work.

5.3. Robustness to mislabeling

Instance mislabeling is not unrelated to class imbalance. [35] Algorithm performance depends on the distribution of mislabeling across the categories in the data. Even more importantly, the impact of mislabeling on algorithm performance in high-dimensional data depends heavily on the average hubness of mislabeled examples. Mislabeling anti-hubs makes no difference whatsoever. Mislabeling even a couple of hub-points should be enough to cause significant misclassification.

In our experiments, mislabeling was distributed uniformly across different categories and only the training data on each cross-validation fold was mislabeled. Evaluation was performed on the original labels. An overview of algorithm performance under 30% mislabeling rate is shown in Table 8. The results confirm our hypothesis that the hubness-aware algorithms exhibit *much higher robustness* to mislabeling than kNN.

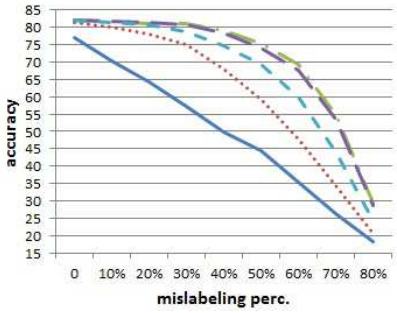
Table 8: Experiments on mislabeled data. 30% mislabeling was artificially introduced to each data set at random. All experiments were performed for $k = 5$. The symbols •/◦ denote statistically significant worse/better performance ($p < 0.05$) compared to kNN. The best result in each line is in bold.

Data set	kNN	hw-kNN	**h-FNN**	NHBNN	HIKNN
diabetes	54.1 ± 3.7	64.7 ± 3.9 ◦	**66.2 ± 3.4** ◦	66.1 ± 3.4 ◦	65.4 ± 3.9 ◦
ecoli	68.1 ± 5.6	80.2 ± 4.7 ◦	**85.8 ± 4.1** ◦	79.3 ± 4.8 ◦	81.7 ± 4.6 ◦
glass	50.6 ± 7.3	61.6 ± 7.3 ◦	**62.8 ± 6.8** ◦	56.8 ± 6.6	61.5 ± 6.7 ◦
iris	71.1 ± 8.5	88.2 ± 6.0 ◦	90.7 ± 5.4 ◦	**93.2 ± 4.6** ◦	87.8 ± 6.3 ◦
mfeat-factors	70.7 ± 2.3	91.4 ± 1.5 ◦	**94.9 ± 1.1** ◦	94.7 ± 1.2 ◦	93.9 ± 1.2 ◦
mfeat-fourier	57.1 ± 2.5	75.0 ± 2.1 ◦	**81.0 ± 1.7** ◦	80.7 ± 1.9 ◦	78.7 ± 1.7 ◦
ovarian	58.1 ± 6.6	76.3 ± 6.1 ◦	**81.1 ± 5.6** ◦	79.4 ± 5.6 ◦	78.3 ± 5.5 ◦
segment	62.7 ± 2.2	81.1 ± 1.9 ◦	**84.3 ± 1.7** ◦	83.8 ± 1.6 ◦	80.8 ± 1.7 ◦
sonar	61.5 ± 7.7	70.8 ± 6.8 ◦	72.4 ± 6.4 ◦	**72.9 ± 6.3** ◦	71.4 ± 6.8 ◦
vehicle	48.2 ± 3.9	57.5 ± 3.9 ◦	58.1 ± 4.0 ◦	56.8 ± 4.0 ◦	**59.2 ± 3.8** ◦
ImNet3	51.0 ± 2.3	69.9 ± 2.2 ◦	**81.2 ± 1.8** ◦	80.6 ± 1.6 ◦	75.3 ± 2.0 ◦
ImNet4	44.6 ± 1.4	52.5 ± 1.3 ◦	**63.3 ± 1.3** ◦	63.1 ± 1.2 ◦	57.6 ± 1.3 ◦
ImNet5	40.0 ± 1.4	47.2 ± 1.4 ◦	**60.6 ± 1.2** ◦	60.0 ± 1.2 ◦	53.1 ± 1.3 ◦
ImNet6	49.5 ± 1.7	55.1 ± 1.4 ◦	**68.0** ± 1.3 ◦	67.4 ± 1.3 ◦	62.8 ± 1.4 ◦
ImNet7	33.1 ± 1.1	44.8 ± 1.1 ◦	**57.6 ± 1.1** ◦	56.8 ± 1.1 ◦	51.0 ± 1.1 ◦
ImNet3Imb	56.7 ± 3.0	78.7 ± 2.2 ◦	**87.0 ± 1.6** ◦	81.1 ± 2.2 ◦	83.2 ± 2.1 ◦
ImNet4Imb	51.8 ± 1.7	55.0 ± 1.7 ◦	**68.7 ± 1.7** ◦	67.3 ± 1.8 ◦	63.9 ± 1.7 ◦
ImNet5Imb	50.7 ± 2.1	53.5 ± 2.0 ◦	**64.2 ± 2.0** ◦	60.5 ± 1.8 ◦	60.6 ± 1.2 ◦
ImNet6Imb	54.7 ± 2.1	55.8 ± 2.0 ◦	**69.7 ± 1.7** ◦	66.6 ± 1.9 ◦	62.8 ± 2.0 ◦
ImNet7Imb	33.1 ± 2.3	52.0 ± 1.9 ◦	**62.9 ± 1.9** ◦	61.1 ± 1.9 ◦	58.6 ± 1.7 ◦
AVG	53.37	65.57	**73.03**	71.41	69.38

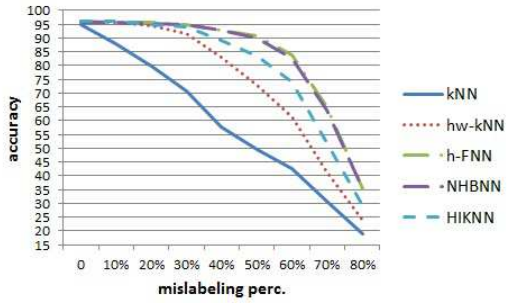
Out of the compared hubness-aware algorithms, h-FNN dominates in this experimental setup. On many datasets h-FNN is no more than 1-2% less accurate than before, which is astounding considering the level of mislabeling in the data. On the other hand, the hubness-weighting approach (hw-kNN) fails in this case and is not able to cope with such high mislabeling rates.

Similarly, Figure 16 shows the drop in accuracy as mislabeling is slowly introduced in the data. The kNN performance seems to be decreasing at a linear rate with increasing noise. At the same time, hubness-aware approaches retain most of their accuracy as the mislabeling rate goes all the way up to $40\% - 50\%$. This can

be explained by the fact that the voting in the hubness-aware approaches is based on the hub occurrence profiles and very high noise levels are required in order to sufficiently compromise the occurrence profiles of the most prominent hubs.



(a) mfeat-fourier



(b) mfeat-fact

Figure 16: The drop in accuracy as the mislabeling rate increases. The kNN accuracy drops linearly, but that is not the case with hubness-aware approaches, which retain good performance even under high mislabeling rates.

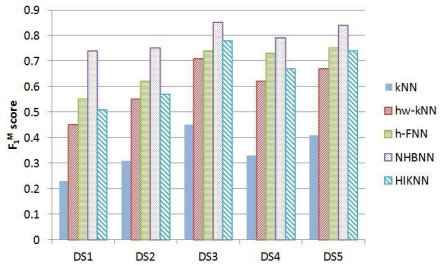
5.4. Overlapping categories

Class imbalance is by itself usually not enough to cause serious misclassification. It has to be coupled with some overlap between different class distributions.

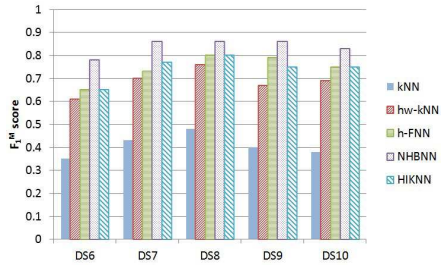
In order to independently study the impact of severe class overlap on the classification performance, we have performed extensive analysis on high-dimensional synthetic data. Assuring substantial overlap between classes in high-dimensional data is non-trivial, as points tend to be spread far apart. A degree of overlap high enough to induce severe misclassification was required, in order to make the data challenging for nearest-neighbor methods. A series of 10 synthetic data sets was generated as random 100-dimensional 10-category Gaussian mixtures. High overlap degree was achieved by placing each feature distribution center randomly within a certain multiple of the standard deviation from some other randomly chosen, previously determined, distribution center.

As shown in Table 9, all the data sets exhibited very high hubness and very high bad hubness. Imbalance level in the data was moderate. There were no clear majority or minority classes, but some overall imbalance was present, with RImb≈ 0.2 in most data sets. As in previous experiments, we performed 10-times 10-fold cross validation and the corrected re-sampled t-test was used to verify the statistically significant differences. For this round of experiments, we have opted for setting the neighborhood size to $k = 10$, in order to reach better estimates in the borderline regions. As the data was Gaussian, the Euclidean distance was used.

The results are given in Table 9. The baseline kNN is on average only able to achieve 58.09% accuracy, while NHBNN stands best among the hubness-aware methods with an impressive average accuracy of 86.18%. Not only NHBNN, but all hubness-aware approaches clearly and convincingly outperform kNN in this experimental setup. The weighted approach (hw-kNN) was again slightly inferior to the class-hubness-based methods (h-FNN, NHBNN, HIKNN). The differences in the macro-averaged F_1-score are even more pronounced, as shown in Figure 17, which suggests that hubness-aware voting helps in successfully dealing with class distribution overlap.



(a)



(b)

Figure 17: Macro-averaged F_1 score on overlapping Gaussian mixture data.

Figure 18 shows the precision that each of the al-

Table 9: Classification accuracies on synthetic Gaussian mixture data for $k = 10$. For each data set, the skewness of the N_{10} distribution is given along with the bad occurrence rate (BN_{10}). The symbols •/○ denote statistically significant worse/better performance ($p < 0.01$) compared to kNN. The best result in each line is in bold.

Data set	size	SN_{10}	BN_{10}	kNN	hw-kNN	h-FNN	**NHBNN**	HIKNN
DS_1	1244	6.68	53.5%	43.8 ± 3.1	64.4 ± 5.3 ○	72.6 ± 2.8 ○	**80.7 ± 2.4** ○	65.8 ± 3.0 ○
DS_2	1660	4.47	49.2%	48.4 ± 2.8	73.6 ± 6.9 ○	79.3 ± 2.2 ○	**83.9 ± 2.2** ○	73.1 ± 2.5 ○
DS_3	1753	5.50	42.0%	67.3 ± 2.3	85.3 ± 2.6 ○	86.8 ± 1.7 ○	**90.0 ± 1.4** ○	86.7 ± 1.9 ○
DS_4	1820	3.45	51%	52.2 ± 2.6	72.8 ± 2.3 ○	78.4 ± 2.2 ○	**81.9 ± 2.0** ○	72.2 ± 2.3 ○
DS_5	1774	4.39	46.3%	59.2 ± 2.7	80.2 ± 3.4 ○	84.6 ± 1.8 ○	**87.2 ± 1.5** ○	81.1 ± 2.1 ○
DS_6	1282	3.98	45.6%	58.6 ± 3.3	80.0 ± 3.3 ○	81.7 ± 2.5 ○	**86.6 ± 2.2** ○	79.4 ± 2.5 ○
DS_7	1662	4.64	41.5%	65.0 ± 2.4	84.6 ± 2.4 ○	85.4 ± 1.9 ○	**90.1 ± 1.5** ○	84.5 ± 2.0 ○
DS_8	1887	4.19	40.0%	71.0 ± 2.3	82.7 ± 2.5 ○	85.9 ± 1.9 ○	**88.4 ± 1.8** ○	83.9 ± 2.3 ○
DS_9	1661	5.02	47.5%	57.9 ± 2.7	76.3 ± 3.3 ○	82.3 ± 2.0 ○	**87.5 ± 1.7** ○	77.7 ± 2.4 ○
DS_{10}	1594	4.82	46.9%	57.5 ± 2.9	78.1 ± 3.3 ○	81.1 ± 2.3 ○	**85.5 ± 1.9** ○	77.7 ± 2.2 ○
AVG				58.09	77.80	81.81	**86.18**	78.21

gorithms achieves on safe points, borderline examples, rare points and outliers, separately [52]. Not surprisingly, kNN is completely incapable of dealing with rare points and outliers - and performs badly even on borderline points. We should point out that the reason why the precision isn't 100% on safe points is that $k = 5$ is used (as described in [52]) to determine point types, but here we are observing 10-NN classification. Hubness-aware methods achieve higher precision on all point types, safe points included. The difference in performance is most pronounced for more difficult point types and this is where most of the improvement stems from. Also, we are able to see why NHBNN scores better than the other hubness-aware algorithms on this data. It performs better when classifying all the difficult point types in the overlap regions. On average, NHBNN manages to correctly assign the labels to more than 90% of borderline points, about 75% of rare points and 35% of outliers. We have verified that this is indeed true for all 10 examined Gaussian mixtures. It is interesting to note that the same trend is not detected in ImgNet data that was discussed in Section 5.2. Bad hubness in ImgNet data is not exclusively due to class overlap, so it is a different story altogether.

As a final remark, we report the performance of some other well-known algorithms on class overlap data. Table 10 contains a summary of results given for the fuzzy k-nearest-neighbor (FNN) [38], probabilistic nearest neighbor (PNN) [32], neighbor-weighted kNN (NWKNN) [65], adaptive kNN (AKNN) [85], J48 (a WEKA [88] implementation of the Quinlan's C4.5 algorithm [57]), random forest classifier [7] and Naive Bayes [50]. Default parameter configurations were used for the Weka implementations of the tree-based algorithms.

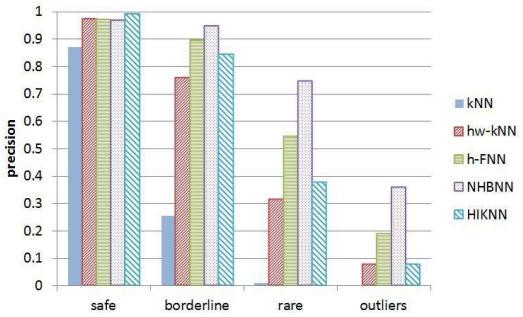


Figure 18: Classification precision on certain types of points on DS_0: safe points, borderline points, rare examples and outliers. We see that the baseline kNN is completely unable to deal with rare points and outliers and this is precisely where the improvements in hubness-aware approaches stem from.

The first thing to notice is that FNN scores much worse than its hubness-aware counterpart h-FNN. This shows that there is a large difference in semantics between the fuzziness derived from direct and reverse k-nearest neighbor sets. The best performance among all the tested hubness-unaware kNN methods is attained by the adaptive kNN (AKNN), which is not surprising since it was designed specifically for handling class-overlap data [85]. Its performance is still, however, somewhat inferior to that of NHBNN, at least in this experimental setup.

Decision trees, on the other hand, seem to have been heavily affected by the induced class overlap, as using either C4.5 or random forest classifiers results in low overall accuracy rates. Naive Bayes was the best among the tested approaches on these Gaussian Mixtures.

Figure 19 shows how both NHBNN and Naive Bayes

Table 10: Classification accuracy of a selection of algorithms on Gaussian mixture data. The results are given for fuzzy k-nearest-neighbor (FNN), probabilistic nearest neighbor (PNN), neighbor-weighted kNN (NWKNN), adaptive kNN (AKNN), J48 implementation of the Quinlan's C4.5 algorithm, random forest classifier and Naive Bayes, respectively. A neighborhood size of $k = 10$ was used in the nearest-neighbor-based approaches, where applicable. Results better than than the ones of NHBNN in Table 9 are given in bold.

Data set	FNN	PNN	NWKNN	AKNN	J48	R. Forest	Naive Bayes
DS_1	36.6 ± 3.0	39.8 ± 3.5	46.5 ± 3.3	79.5 ± 2.6	42.4 ± 4.3	59.5 ± 3.7	**95.6 ± 1.3**
DS_2	40.5 ± 2.9	35.9 ± 3.2	54.0 ± 2.6	82.7 ± 2.1	47.3 ± 3.9	65.4 ± 3.9	**97.1 ± 0.9**
DS_3	61.5 ± 2.7	71.3 ± 2.4	67.4 ± 2.5	88.7 ± 1.7	48.9 ± 3.9	69.2 ± 3.1	**98.6 ± 0.2**
DS_4	46.6 ± 2.4	43.4 ± 4.6	56.5 ± 2.9	84.7 ± 1.7	44.0 ± 3.7	59.7 ± 3.7	**98.4 ± 0.2**
DS_5	52.3 ± 2.9	54.1 ± 4.3	61.8 ± 2.6	83.2 ± 2.1	45.6 ± 2.9	64.1 ± 3.2	**98.3 ± 0.1**
DS_6	51.5 ± 3.0	51.5 ± 3.5	62.2 ± 3.0	78.6 ± 3.2	52.1 ± 4.2	67.2 ± 3.1	**97.3 ± 1.1**
DS_7	59.0 ± 2.7	60.0 ± 4.0	66.9 ± 2.6	90.1 ± 1.5	51.0 ± 3.7	70.7 ± 2.6	**98.3 ± 0.7**
DS_8	67.8 ± 2.6	72.6 ± 2.6	71.5 ± 2.5	85.2 ± 1.9	50.2 ± 3.7	67.1 ± 3.1	**98.7 ± 0.4**
DS_9	51.9 ± 2.7	48.9 ± 4.6	61.7 ± 2.6	84.5 ± 2.0	43.9 ± 3.6	64.5 ± 3.7	**98.3 ± 0.7**
DS_{10}	51.0 ± 2.7	47.8 ± 4.2	62.1 ± 2.5	79.6 ± 2.0	46.2 ± 3.8	64.0 ± 3.1	**97.9 ± 0.8**
AVG	51.87	52.53	61.06	83.68	47.16	65.14	**97.85**

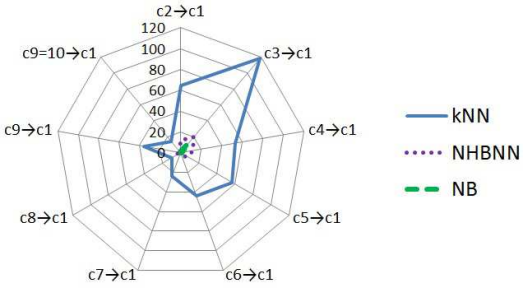


Figure 19: Misclassification towards the class c1 that exhibits highest overall bad hubness on DS_0. NHBNN and NB clearly outperform kNN here.

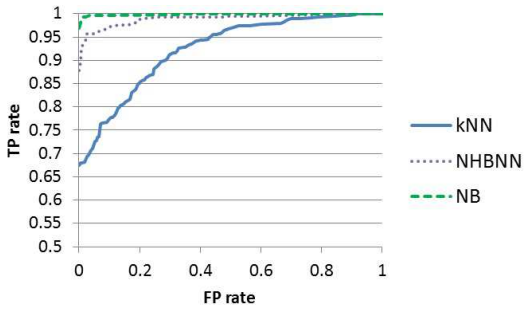


Figure 20: A one-vs-all ROC curve where one of the classes with a lower TP rate (c5) is taken as the negative class, on DS_0.

outperform the kNN baseline by reducing the misclassification caused by a class with high bad hubness.

An ROC curve that maps the TP rate against the FP rate is shown in Figure 20 for DS_0, where $c5$ is taken as the negative class and all other points are treated as positives. The area under the ROC curve (AUC) in this case is 0.923 for kNN, 0.974 for NWKNN, 0.965 for hw-kNN, 0.989 for NHBNN and 0.998 for Naive Bayes. Of course, the ROC analysis in the multi-class case is a bit more complex, but Figure 20 illustrates the common trends in this high-dimensional Gaussian Mixture data.

What these comparisons reveal is that the currently available hubness-aware k-nearest neighbor approaches rank rather well when compared to the other kNN-based methods, but there is also some room for improvement.

6. Conclusions and Future Work

Hubness is an important aspect of the curse of dimensionality related to k-nearest neighbor methods. It has a negative impact on the performance of many information systems, as it allows the errors to easily propagate through the data. In this paper, we have shown that it further complicates the issues concerning learning under class imbalance in high-dimensional data.

Class imbalance poses great difficulties for most machine learning methods and has been a focus of many serious studies. In low-to-medium-dimensional data, the majority class is known to often cause misclassification of the minority class.

Surprisingly, we have shown that this intuitive consequence of the difference in average relative density gradients does not necessarily hold in intrinsically high-dimensional data, under the assumption of hubness. In such cases, minority classes frequently exhibit high bad hubness and have the capacity to induce severe misclassification of the majority class. In high-dimensional data, most misclassification is caused by the classes which have the majority among the bad hubs. We have shown that the minority classes often achieve this bad

hub majority and become the principal sources of misclassification.

High-dimensional geometry allows for some more unexpected results, as we have shown that bad hubness is not expressed only by borderline points, but also by points expected to lie in the interiors of class distributions. This represents a strong violation of the cluster assumption.

In order to see if the arising problems can be solved by utilizing the neighbor occurrence models in order to predict and rectify the detrimental hub point occurrences, we have performed an extensive evaluation of several state-of-the-art hubness-aware k-nearest neighbor classifiers: hw-kNN, h-FNN, NHBNN and HIKNN. The methods were compared on high-dimensional problems involving class imbalance, mislabeling and class overlap. The results suggest that the tested approaches exhibit promising levels of robustness and tolerance to the arising problems. The Naive Bayesian way of handling the occurrence models was able to achieve very high precision when handling borderline examples, rare points and outliers.

A high misclassification rate caused by the minority class examples in many high-dimensional datasets suggests that the traditional kNN approaches to handling class imbalanced data that involve adopting an explicit bias towards the minority points are not in general well suited for the high-dimensional case. The design of these methods should be extended to support the modeling of minority-induced misclassification, in order to reduce the negative impact of bad hubs. One way to do this would be to employ the neighbor occurrence modeling within the class imbalanced kNN methods, by combining them with the existing hubness-aware approaches.

In future work we intend to investigate the possibilities for cost-sensitive learning and boosting in building the occurrence models for hubness-aware classification. We also plan on extending and improving the existing algorithms now that we have gained a deeper understanding of their advantages and disadvantages. Additionally, we will investigate various hubness-aware data preprocessing schemes for filtering out the mislabeled/noisy data.

Acknowledgments

This work was supported by the Slovenian Research Agency, the ICT Programme of the EC under XLike (ICT-STREP-288342), and RENDER (ICT-257790-STREP).

References

[1] Aggarwal, C. C., Hinneburg, A., and Keim, D. A. (2001). On the surprising behavior of distance metrics in high dimensional spaces. In *Proc. 8th Int. Conf. on Database Theory*, pages 420–434.

[2] Aucouturier, J. (2006). Ten experiments on the modelling of polyphonic timbre. Technical report, Doct. dissert., Univ. of Paris 6.

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

[4] Batista, G. E. A. P. A., Prati, R. C., and Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.*, 6(1):20–29.

[5] Batuwita, R. and Palade, V. (2009). Micropred: effective classification of pre-miRNAs for human miRNA gene prediction. *Bioinformatics*, 25(8):989–995.

[6] Bouckaert, R. and Frank, E. (2004). Evaluating the replicability of significance tests for comparing learning algorithms. In Dai, H., Srikant, R., and Zhang, C., editors, *Advances in Knowledge Discovery and Data Mining*, pages 3–12. Springer Berlin Heidelberg.

[7] Breiman, L. (2001). Random forests. *Mach. Learn.*, 45(1):5–32.

[8] Buza, K., Nanopoulos, A., and Schmidt-Thieme, L. (2011). Insight: efficient and effective instance selection for time-series classification. In *Advances in knowledge discovery and data mining*, PAKDD'11, pages 149–160, Berlin, Heidelberg. Springer-Verlag.

[9] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

[10] Chen, J., Fang, H., and Saad, Y. (2009). Fast approximate kNN graph construction for high dimensional data via recursive Lanczos bisection. *Journal of Machine Learning Research*, 10:1989–2012.

[11] C.J.Stone (1977). Consistent nonparametric regression. *Annals of Statistics*, 5:595–645.

[12] Dubey, H. and Pudi, V. (2013). Class based weighted k-nearest neighbor over imbalance dataset. In Pei, J., Tseng, V., Cao, L., Motoda, H., and Xu, G., editors, *Advances in Knowledge Discovery and Data Mining*, pages 305–316. Springer Berlin Heidelberg.

[13] Durrant, R. J. and Kabán, A. (2009). When is 'nearest neighbour' meaningful: A converse theorem and implications. *Journal of Complexity*, 25(4):385–397.

[14] Ertekin, S., Huang, J., and Giles, C. L. (2007a). Active learning for class imbalance problem. In *Proceedings of the 30th ACM SIGIR conference*, pages 823–824, New York, NY, USA. ACM.

[15] Ertekin, S. E., Huang, J., Bottou, L., and Giles, C. L. (2007b). Learning on the border: Active learning in imbalanced data classification. In *In Proc. of CIKM conference*.

[16] Ezawa, K., Singh, M., and Norton, S. W. (1996). Learning goal oriented Bayesian networks for telecommunications risk management. In *In Proceedings of the 13th International Conference on Machine Learning*, pages 139–147. Morgan Kaufmann.

[17] Ezawa, K. J. and Schuermann, T. (1995). Fraud/uncollectible debt detection using a bayesian network based learning system: a rare binary outcome with mixed data structures. In *Proc. of the 11th conf. on Uncertainty in AI*, pages 157–166, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[18] Fernandez, A., Garcia, S., and Herrera, F. (2011). Addressing the classification with imbalanced data: Open problems and new challenges on class distribution. In *In proc. of HAIS conference*, pages 1–10. Springer Berlin Heidelberg.

[19] Fix, E. and Hodges, J. (1951). Discriminatory analysis, nonparametric discrimination: consistency properties. Technical report, USAF School of Aviation Medicine, Randolph Field.

[20] François, D., Wertz, V., and Verleysen, M. (2007). The concentration of fractional distances. *IEEE Transactions on Knowledge and Data Engineering*, 19(7):873–886.

[21] Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., and Herrera, F. (2012). A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 42(4):463–484.

[22] Garca-Pedrajas, N. and Garca-Osorio, C. (2013). Boosting for class-imbalanced datasets using genetically evolved supervised non-linear projections. *Progress in Artificial Intelligence*, 2(1):29–44.

[23] Garcia, V., Mollineda, R. A., and Sanchez, J. S. (2008). On the k-nn performance in a challenging scenario of imbalance and overlapping. *Pattern Anal. Appl.*, 11:269–280.

[24] Garcia, V., Sanchez, J., Martn-Flez, R., and Mollineda, R. (2012). Surrounding neighborhood-based smote for learning from imbalanced data sets. *Progress in Artificial Intelligence*, 1(4):347–362.

[25] Garcia-Pedrajas, N., Perez-Rodriguez, J., and de Haro-Garcia, A. (2013). Oligois: Scalable instance selection for class-imbalanced data sets. *Cybernetics, IEEE Transactions on*, 43(1):332–346.

[26] Ghosh, A. K. (2012). A probabilistic approach for semi-supervised nearest neighbor classification. *Pattern Recogn. Lett.*, 33(9):1127–1133.

[27] Guan, D., Yuan, W., Lee, Y.-K., and Lee, S. (2011). Identifying mislabeled training data with the aid of unlabeled data. *Applied Intelligence*, 35:345–358.

[28] Hastie, T. and Tibshirani, R. (1996). Discriminant adaptive nearest neighbor classification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(6):607–616.

[29] Hayashi, K. (2012). A simple extension of boosting for asymmetric mislabeled data. *Statistics and Prob. Letters*, 82:348–356.

[30] He, H., Bai, Y., Garcia, E. A., and Li, S. (2008). Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *IJCNN'08*, pages 1322–1328.

[31] He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284.

[32] Holmes, C. C. and Adams, N. M. (2002). A probabilistic nearest neighbor method for statistical pattern recognition. *J R Stat Soc B*, 64:295–306.

[33] Holte, R. C., Acker, L. E., and Porter, B. W. (1989). Concept learning and the problem of small disjuncts. In *Proc. 11th Int. Conf. AI - Volume 1*, pages 813–818. Morgan Kaufmann Publishers Inc.

[34] Hong, X., Chen, S., and Harris, C. J. (2007). A kernel-based two-class classifier for imbalanced data sets. *IEEE Transactions on Neural Networks*, pages 28–41.

[35] Hulse, J. V., Khoshgoftaar, T. M., and Napolitano, A. (2007). Skewed class distributions and mislabeled examples. In *Proceedings of the Seventh IEEE International Conference on Data Mining Workshops*, ICDMW '07, pages 477–482, Washington, DC, USA. IEEE Computer Society.

[36] Jensen, R. and Cornelis, C. (2008). A new approach to fuzzy-rough nearest neighbour classification. In *Proceedings of the 6th International Conference on Rough Sets and Current Trends in Computing*, RSCTC '08, pages 310–319, Berlin, Heidelberg. Springer-Verlag.

[37] Kalager, M., Adami, H., Bretthauer, M., and Tamimi, R. (2012). Overdiagnosis of invasive breast cancer due to mammography screening: results from the norwegian screening program. *Annals of Internal Medicine*, 156:491–499.

[38] Keller, J. E., Gray, M. R., and Givens, J. A. (1985). A fuzzy k-nearest-neighbor algorithm. In *IEEE Transactions on Systems, Man and Cybernetics*, pages 580–585.

[39] Kuang, D., Ling, C., and Du, J. (2012). Foundation of mining class-imbalanced data. In Tan, P.-N., Chawla, S., Ho, C., and Bailey, J., editors, *Advances in Knowledge Discovery and Data Mining*, volume 7301 of *Lecture Notes in Computer Science*, pages 219–230. Springer Berlin Heidelberg.

[40] Kubat, M. and Matwin, S. (1997). Addressing the curse of imbalanced training sets: One-sided selection. In *In Proceedings of the Fourteenth International Conference on Machine Learning*, pages 179–186. Morgan Kaufmann.

[41] Li, Y. and Zhang, X. (2011). Improving k-nearest neighbor with exemplar generalization for imbalanced classification. In *Advances in Knowledge Disc. and Data Mining*, pages 321–332. Springer.

[42] Liu, J., Hu, Q., and Yu, D. (2008). A comparative study on rough set based class imbalance learning. *Knowledge-Based Systems*, 21(8):753 – 763.

[43] Liu, W. and Chawla, S. (2011). Class confidence weighted knn algorithms for imbalanced data sets. In *Advances in Knowledge Discovery and Data Mining*, volume 6635, pages 345–356. Springer.

[44] Liu, W., Chawla, S., Cieslak, D. A., and Chawla, N. V. (2010). A Robust Decision Tree Algorithm for Imbalanced Data Sets. In *SDM*, pages 766–777. SIAM.

[45] Liu, X.-Y., Wu, J., and Zhou, Z.-H. (2006). Exploratory undersampling for class-imbalance learning. In *Proceedings of the Sixth International Conference on Data Mining*, ICDM '06, pages 965–969, Washington, DC, USA. IEEE Computer Society.

[46] Lopez, V., Fernndez, A., Moreno-Torres, J. G., and Herrera, F. (2012). Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. open problems on intrinsic data characteristics. *Expert Systems with Applications*, 39(7):6585 – 6608.

[47] Low, T., Borgelt, C., Stober, S., and Nrnberger, A. (2013). The hubness phenomenon: Fact or artifact? In Borgelt, C., Gil, M. n., Sousa, J. M., and Verleysen, M., editors, *Towards Advanced Data Analysis by Combining Soft Computing and Statistics*, volume 285 of *Studies in Fuzziness and Soft Computing*, pages 267–278. Springer Berlin Heidelberg.

[48] Martino, M. D., FernáNdez, A., Iturralde, P., and Lecumberry, F. (2013). Novel classifier scheme for imbalanced problems. *Pattern Recogn. Lett.*, 34(10):1146–1151.

[49] McCarthy, K., Zabar, B., and Weiss, G. (2005). Does cost-sensitive learning beat sampling for classifying rare classes? In *Proceedings of the 1st int. workshop on Utility-based data mining*, UBDM '05, pages 69–77, New York, NY, USA. ACM.

[50] Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

[51] Nanopoulos, A., Radovanović, M., and Ivanović, M. (2009). How does high dimensionality affect collaborative filtering? In *Proceedings of the third ACM conference on Recommender systems*, RecSys '09, pages 293–296, New York, NY, USA. ACM.

[52] Napierala, K. and Stefanowski, J. (2012). Identification of different types of minority class examples in imbalanced data. In *In proc. of. HAIS conference*, pages 139–150. Springer Berlin.

[53] Ougiaroglou, S., Nanopoulos, A., Papadopoulos, A. N., Manolopoulos, Y., and Welzer-druzovec, T. (2007). Adaptive k-nearest neighbor classification based on a dynamic number of nearest neighbors. In *Proceedings of ADBIS Conference*, ADBIS 2007.

[54] Peng, J., Heisterkamp, D. R., and Dai, H. K. (2004). Adaptive quasiconformal kernel nearest neighbor classification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(5):656–661.

[55] Pracner, D., Tomašev, N., Radovanović, M., Mladenić, D., and Ivanović, M. (2011). WIKImage: Correlated image and text datasets. In *Proc. of the 14th International Multiconference on Information Society (IS 2011)*, volume A, pages 141–144, Ljubljana, Slovenia.

[56] Prati, R., Batista, G., and Monard, M. (2004). Class imbalances versus class overlapping: An analysis of a learning system behavior. In *Proceedings of the Mexican International Conference on*

Artificial Intelligence, RSCTC '08, pages 312–321.

[57] Quinlan, R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers.

[58] Radovanović, M. (2011). *Representations and Metrics in High-Dimensional Data Mining*. Izdavačka knjižarnica Zorana Stojanovića, Novi Sad, Serbia.

[59] Radovanović, M., Nanopoulos, A., and Ivanović, M. (2009). Nearest neighbors in high-dimensional data: The emergence and influence of hubs. In *Proc. 26th Int. Conf. on Machine Learning (ICML)*, pages 865–872.

[60] Radovanović, M., Nanopoulos, A., and Ivanović, M. (2010a). Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research*, 11:2487–2531.

[61] Radovanović, M., Nanopoulos, A., and Ivanović, M. (2010b). On the existence of obstinate results in vector space models. In *Proc. 33rd Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 186–193.

[62] Radovanovic, M., Nanopoulos, A., and Ivanovic, M. (2010). Time-series classification in many intrinsic dimensions. In *SDM*, pages 677–688. SIAM.

[63] Schnitzer, D., Flexer, A., Schedl, M., and Widmer, G. (2012). Local and global scaling reduce hubs in space. *Journal of Machine Learning Research*, pages 2871–2902.

[64] Song, Y., Huang, J., Zhou, D., Zha, H., and Giles, C. L. (2007). Iknn: Informative k-nearest neighbor pattern classification. In *Proceedings of the 11th European conference on Principles and Practice of Knowledge Discovery in Databases*, PKDD 2007, pages 248–264, Berlin, Heidelberg. Springer-Verlag.

[65] Tan, S. (2005a). Neighbor-weighted k-nearest neighbor for unbalanced text corpus. *Expert Syst. Appl.*, 28:667–671.

[66] Tan, S. (2005b). Neighbor-weighted k-nearest neighbor for unbalanced text corpus. *Expert Syst. Appl.*, 28:667–671.

[67] Thanathamathee, P. and Lursinsap, C. (2013). Handling imbalanced data sets with synthetic boundary data generation using bootstrap re-sampling and adaboost techniques. *Pattern Recognition Letters*, 34(12):1339 – 1347.

[68] Ting, K. M. (2002). An instance-weighting method to induce cost-sensitive trees. *IEEE Trans. on Knowl. and Data Eng.*, 14(3):659–665.

[69] Tomašev, N., , Leban, G., and Mladenić, D. (2013a). Exploiting hubs for self-adaptive secondary re-ranking in bug report duplicate detection. In *Proceedings of the ITI conference*, ITI 2013.

[70] Tomašev, N., , Rupnik, J., and Mladenić, D. (2013b). The role of hubs in cross-lingual supervised document retrieval. In *Proceedings of the PAKDD Conference*, PAKDD 2013.

[71] Tomašev, N., Brehar, R., Mladenić, D., and Nedevschi, S. (2011a). The influence of hubness on nearest-neighbor methods in object recognition. In *Proceedings of the 7th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 367–374.

[72] Tomašev, N. and Mladenić, D. (2011). The influence of weighting the k-occurrences on hubness-aware classification methods. In *Proceedings of the SiKDD conference*.

[73] Tomašev, N. and Mladenić, D. (2012a). Hubness-aware shared neighbor distances for high-dimensional k-nearest neighbor classification. In *Proceedings of the 7th International Conference on Hybrid Artificial Intelligence Systems*, HAIS '12.

[74] Tomašev, N. and Mladenić, D. (2012b). Hubness-aware shared neighbor distances for high-dimensional k-nearest neighbor classification. *Knowledge and Information Systems*.

[75] Tomašev, N. and Mladenić, D. (2012c). Nearest neighbor voting in high dimensional data: Learning from past occurrences. *Computer Science and Information Systems*, 9:691–712.

[76] Tomašev, N. and Mladenić, D. (2013a). Hub co-occurrence modeling for robust high-dimensional knn classification. In *Proceedings of the ECML/PKDD Conference*. Springer.

[77] Tomašev, N. and Mladenić, D. (2013b). Image hub explorer: Evaluating representations and metrics for content-based image retrieval and object recognition. In *Proceedings of the ECML/PKDD Conference*. Springer.

[78] Tomašev, N., Pracner, D., Brehar, R., Radovanović, M., Mladenić, D., Ivanović, M., and Nedevshi, S. (2013c). Object recognition in wikimage data based on local invariant image features. In *Proceedings of the IEEE ICCP Conference*. IEEE.

[79] Tomašev, N., Radovanović, M., Mladenić, D., and Ivanović, M. (2011b). A probabilistic approach to nearest neighbor classification: Naive hubness bayesian k-nearest neighbor. In *Proceeding of the CIKM conference*.

[80] Tomašev, N., Radovanović, M., Mladenić, D., and Ivanović, M. (2011c). The role of hubness in clustering high-dimensional data. In *Advances in Knowledge Discovery and Data Mining*, volume 6634, pages 183–195. Springer.

[81] Tomašev, N., Radovanović, M., Mladenić, D., and Ivanović, M. (2013d). Hubness-based fuzzy measures for high-dimensional k-nearest neighbor classification. *International Journal of Machine Learning and Cybernetics*.

[82] Tomašev, N., Radovanović, M., Mladenić, D., and Ivanović, M. (2013e). The role of hubness in clustering high-dimensional data. *IEEE Transactions on Knowledge and Data Engineering*, 99(PrePrints).

[83] Valizadegan, H. and Tan, P.-N. (2007). Kernel based detection of mislabeled training examples. In *SDM*. SIAM.

[84] van den Bosch, A., Weijters, T., Herik, H. J. V. D., and Daelemans, W. (1997). When small disjuncts abound, try lazy learning: A case study.

[85] Wang, J., Neskovic, P., and Cooper, L. N. (2007). Improving nearest neighbor rule with a simple adaptive distance measure. *Pattern Recogn. Lett.*, 28:207–213.

[86] Wang, S., Li, X., Xia, J.-F., and Zhang, X.-P. (2010). Weighted neighborhood classifier for the classification of imbalanced tumor dataset. *Journal of Circuits, Systems, and Computers*, pages 259–273.

[87] Weiss, G. M. (2004). Mining with rarity: a unifying framework. *SIGKDD Explor. Newsl.*, 6:7–19.

[88] Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers, second edition.

[89] Wu, G. and Chang, E. Y. (2005). KBA: kernel boundary alignment considering imbalanced data distribution. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):786–795.

[90] Zhang, H., Berg, A. C., Maire, M., and Malik, J. (2006). Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, CVPR '06, pages 2126–2136, Washington, DC, USA. IEEE Computer Society.

[91] Zhang, J. and Mani, I. (2003). KNN Approach to Unbalanced Data Distributions: A Case Study Involving Information Extraction. In *Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Datasets*.

[92] Zhang, X. and Li, Y. (2013). A positive-biased nearest neighbour algorithm for imbalanced classification. In Pei, J., Tseng, V., Cao, L., Motoda, H., and Xu, G., editors, *Advances in Knowledge Discovery and Data Mining*, volume 7819 of *Lecture Notes in Computer Science*, pages 293–304. Springer Berlin Heidelberg.

[93] Zhou, L. (2013). Performance of corporate bankruptcy prediction models on imbalanced dataset: The effect of sampling methods. *Know.-Based Syst.*, 41:16–25.