Handling Class Imbalance in k-Nearest Neighbor Classification by Balancing Prior Probabilities

Jonatan M. N. Gøttcke[1], Arthur Zimek[1]

[1]University of Southern Denmark

Abstract

It is well known that recall rather than precision is the performance measure to optimize in imbalanced classification problems, yet most existing methods that adjust for class imbalance do not particularly address the optimization of recall. Here we propose an elegant and straightforward variation of the k-nearest neighbor classifier to balance imbalanced classification problems internally in a probabilistic interpretation and show how this relates to the optimization of the recall. We evaluate this novel method against popular k-nearest neighbor-based class imbalance handling algorithms and compare them to general oversampling and undersampling techniques. We demonstrate that the performance of the proposed method is on par with SMOTE yet our method is much simpler and outperforms several competitors over a large selection of real-world and synthetic datasets and parameter choices while having the same complexity as the regular k-nearest neighbor classifier.

Quality Measures

The *precision* measure reports, for one class against all other classes $\frac{TP}{FP+TP}$. In class imbalanced problems, precision is not a viable measure since the majority classes will have relatively few false positives no matter how many of the minority points they predict as majority points. Examples for the minority classes on the other hand are rarely mistaken for majority points. *Accuracy* and *error rate* are strongly biased to favor the majority class. The problem with accuracy and error rate is obvious when the class imbalances are extreme. Thus the G-mean score is a popular measure [1, 2, 3] in imbalanced classification problems, that is the geometric mean over recall: $(\prod_{i=1}^{n} r_i)^{\frac{1}{n}} = \sqrt[n]{r_1 \cdot r_2 \cdot \ldots \cdot r_n}$, where r_i is the recall for class c_i.

Imbalanced Datasets

#	Dataset	n	Dim	IR	Cl
1	appendicitis	106	7	4.05	2
2	balance	625	4	5.88	3
3	cleveland	297	13	12.31	5
4	coil 2000	9822	85	15.76	2
5	dermatology	358	34	5.55	6
6	ecoli	336	7	71.5	7
7	glass	214	9	8.44	6
8	haberman	306	3	2.78	2
9	hayes roth	160	4	2.10	3
10	hepatitis	80	19	5.15	2
11	marketing	6876	13	2.49	9
12	page-blocks	5472	10	175.46	5
13	phoneme	5404	5	2.41	2
14	satimage	6435	36	2.45	6
15	spectfheart	267	44	3.85	2
16	shuttle	58000	9	4558.60	7
17	thyroid	7200	21	40.16	3
18	titanic	2201	3	2.10	2
19	wine-red	1599	11	68.10	6
20	wine-white	4898	11	439.60	7
21	yeast	1484	8	92.60	10
22	usps	1500	50	4.00	2
23	new thyroid	215	5	5.00	3

*k*NN-BPP

Theorem: Given some query object x in a classification problem with a set C of m classes, let k_i be the number of instances among the k nearest neighbors of x that belong to class c_i, let n_i be the number of instances that belong to class c_i overall (i.e., $n_i = |c_i|$). For the k nearest neighbor classifier, adjusting the prior class probabilities such that all classes are equally likely, i.e., $\forall_i \Pr(c_i) = \frac{1}{m}$, is equivalent to choosing $\arg\max_{c_i \in C} \left(\frac{k_i}{n_i}\right)$, which is the local recall for x.

Proof: The proxy for the probability $\Pr(x|c_i)$ is the density estimation given by the k nearest neighbors, conditional on class c_i, that we can describe as

$$\Pr(x|c_i) \propto \frac{k_i}{n_i V(x)} \tag{1}$$

where $V(x)$ is the volume, centered at x, required to capture k nearest neighbors of x. We can therefore rewrite Equation 6 as follows:

$$\Pr(c_i|x) \propto \frac{\frac{k_i}{n_i V(x)} \cdot \Pr(c_i)}{\sum_{j=1}^{m} \frac{k_j}{n_j V(x)} \cdot \Pr(c_j)} \tag{2}$$

Choosing equal prior class probabilities results in:

$$\Pr(c_i|x) \propto \frac{\frac{k_i}{n_i V(x)} \cdot \frac{1}{m}}{\sum_{j=1}^{m} \frac{k_j}{n_j V(x)} \cdot \frac{1}{m}} \tag{3}$$

which simplifies to

$$\Pr(c_i|x) \propto \frac{\frac{k_i}{n_i}}{\sum_{j=1}^{m} \frac{k_j}{n_j}} \tag{4}$$

where the denominator is obviously identical for all classes. We therefore have

$$\arg\max_{c_i \in C} \Pr(c_i|x) = \arg\max_{c_i \in C} \left(\frac{k_i}{n_i}\right) \tag{5}$$

Average Rank

Performance for $k \in [3, 35]$ in terms of the mean rank over all datasets.

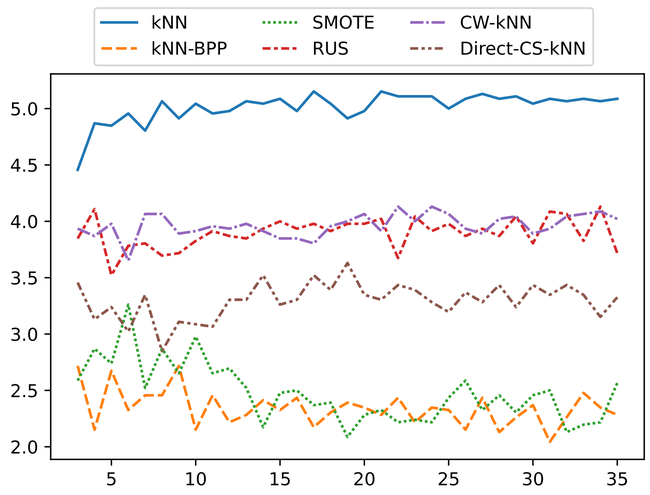


Figure 1: G-mean score

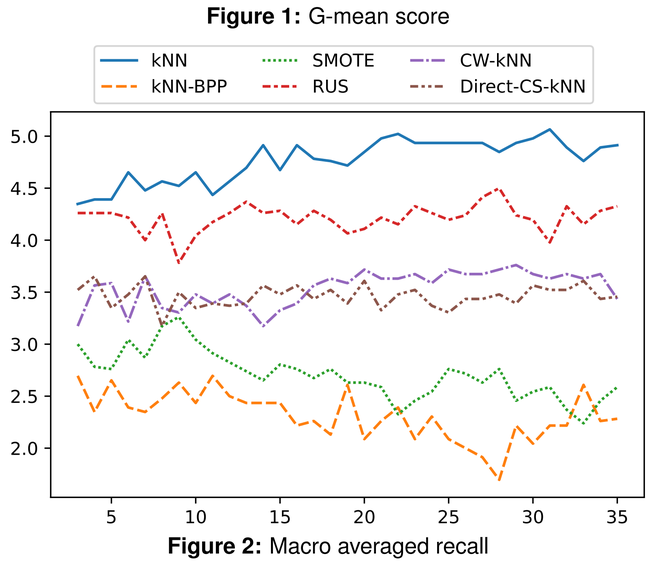


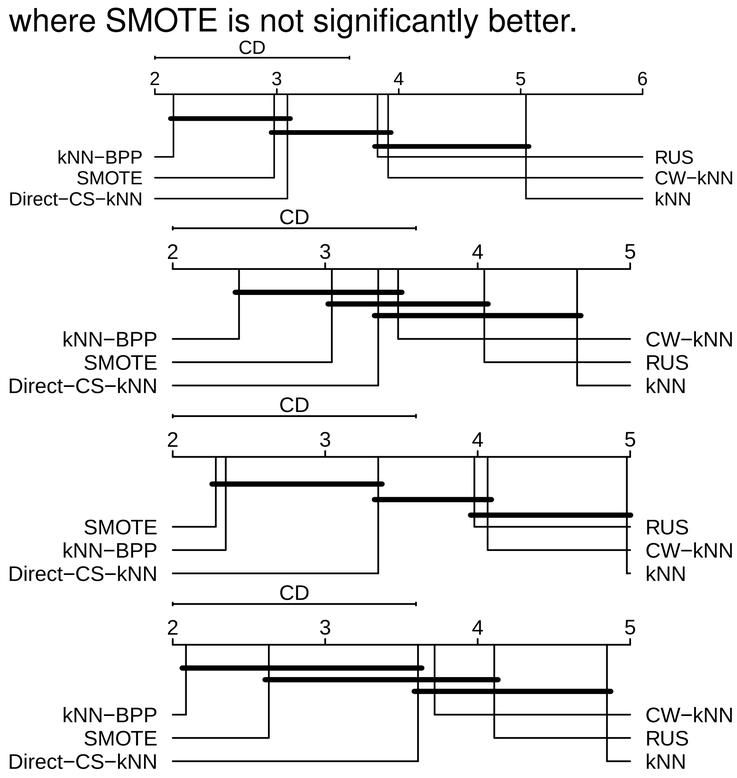
Figure 2: Macro averaged recall

Bayes Theorem

$$\Pr(c_i|x) = \frac{\Pr(x|c_i) \cdot \Pr(c_i)}{\sum_{j=1}^{m} \Pr(x|c_j) \cdot \Pr(c_j)} \tag{6}$$

Statistical evaluation

We performed a statistical analysis of the ranking differences for $k = 10, 20, 30$. The results for 10 and 20 are shown in the critical difference plots. The plots show the average ranking of methods over all datasets together with a bar connecting methods that are not performing differently with statistical significance. We see our method on top or, in one case, second to SMOTE, although their performance is only different with statistical significance from some other methods in most cases. The classic, unchanged *k*NN classifier is typically worst, and several of the improved versions are not better with statistical significance. *k*NN-BPP is significantly better than the unchanged *k*NN classifier in all tests and better than RUS and CW-*k*NN in several, also in cases where SMOTE is not significantly better.



In summary, the critical difference (CD) plots indicate that *k*NN-BPP performs as well or slightly better than a non-trivial oversampling technique, but without adding runtime to the original k-nearest neighbor classifier.

Conclusion

We developed an elegant and straightforward *k*NN classifier, *k*NN-BPP, that balances prior class probabilities and thus treats imbalanced classes in a fair manner. The proposed *k*NN-BPP algorithm shows performance on par with a popular oversampler applied to the datasets in combination with the conventional *k*NN-algorithm for all measured k-values, while having the same computational complexity as regular *k*NN. The algorithm's difference from a weighted *k*NN-algorithm is shown in the paper. *k*NN-BPP's advantage over other recent internal modifications of *k*NN over a wide set of k-values has been established.

References

[1] Miroslav Kubat and Stan Matwin. Addressing the curse of imbalanced training sets: One-sided selection. In *ICML*, pages 179–186. Morgan Kaufmann, 1997.

[2] Nathalie Japkowicz. Assessment metrics for imbalanced learning. In Haibo He and Yunqian Ma, editors, *Imbalanced Learning: Foundations, algorithms, and applications*, chapter 8, pages 187–206. John Wiley & Sons, 2013.

[3] Colin Bellinger, Shiven Sharma, Nathalie Japkowicz, and Osmar R. Zaïane. Framework for extreme imbalance classification: SWIM - sampling with the majority class. *Knowl. Inf. Syst.*, 62(3):841–866, 2020.

github.com/goettcke/kNN_BPP