Quantification via Probability Estimators

A. Bella, C. Ferri, J. Hernández-Orallo and M.J. Ramírez-Quintana

DSIC, UPV, València, Spain.

{abella,cferri,jorallo,mramirez}@dsic.upv.es

The 10th IEEE International Conference on Data Mining
ICDM 2010

December 14-17, 2010
Sydney, Australia

Outline

The problem

Forman was the first in identifying and naming the quantification problem.

Quantification

A (novel) machine learning task which deals with correctly estimating the number of elements of one class in a set of examples.

Many problems in real applications can be seen as quantification problems.

Examples

- How many products will be bought?
- How many clients will be given bank credit?
- How many pieces will fail?

It is especially important when the training dataset does not represent a random sample of the target population.

The problem

Quantification versus Classification

Examples have the same presentation (several input features and a nominal output feature), but

- The test set is considered as a whole versus to apply to a single example alone.
- To determine the test class distributions versus individual predictions for each example.

Quantification versus Regression

The output of the quantification problem is a real value, but

- The test set is considered as a whole versus to apply to a single example alone.

Forman's methods

Forman developed several methods and defined new experimental settings to evaluate the task, focussing on the cases where the training class distribution is different to the test class distribution.

- Classify & Count (CC): to learn a classifier from the training dataset and counting the examples of the test set that the classifier predicts positive.
- Adjust & Count (AC): to scale the proportion of positive examples estimated ($\widehat{pos} = \frac{\widehat{pos}' - fpr}{tpr - fpr}$) and then clips the result to the range [0..1].
- T50: method which selects the threshold where the ratio of true positives is equal to 50%.

Always considering crisp classifiers.

The solution proposed

Scaled Probability Average, a quantifier method based on probability estimations and scaling.

Main features

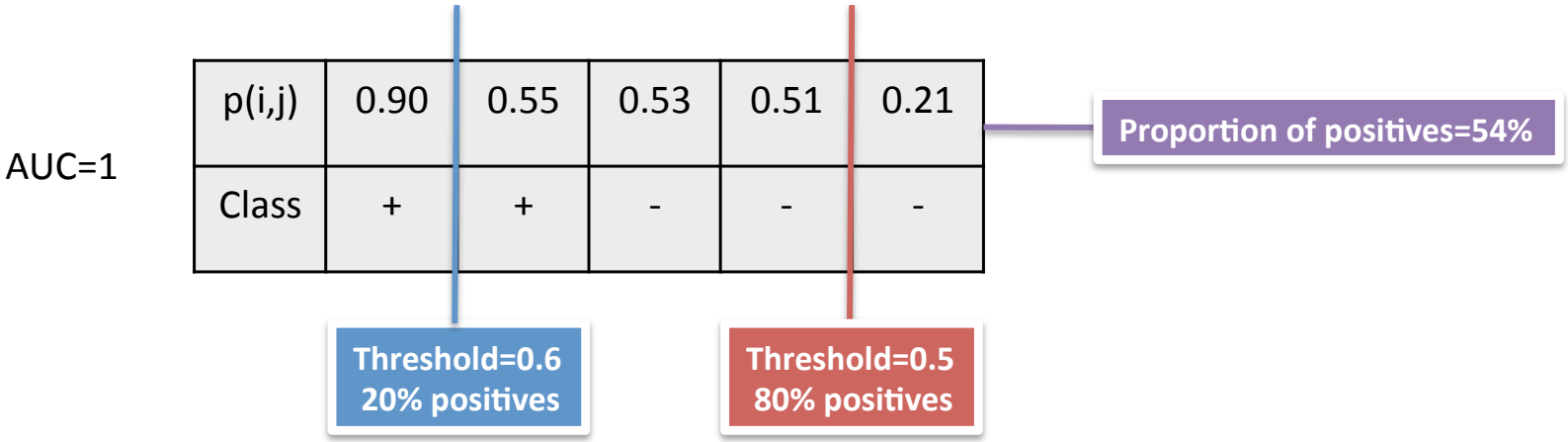
- Use a probability estimator instead of a classifier.
- Forman did not consider probabilities because *"probability estimates depend explicitly on the class distribution; the calibrated probabilities would become uncalibrated whenever the test class distribution varies"*.

The solution proposed

Scaled Probability Average, a quantifier method based on probability estimations and scaling.

Main features

- A different way to estimate the positive proportion by using probability estimations sharing the spirit of the AC method.
- It can be more robust to variations in the probability estimation than other methods based on thresholds.



Probability Average (PA) method

1. Learn a probabilistic classifier.
2. To apply it to obtain the probability estimation for each instance $i \in Test$, $p(i, \oplus)$.
3. Average estimated probabilities for each class.

$$\hat{\pi}^{PA}_{Test}(\oplus) = \frac{\sum_{i \in Test} p(i, \oplus)}{|Test|}$$

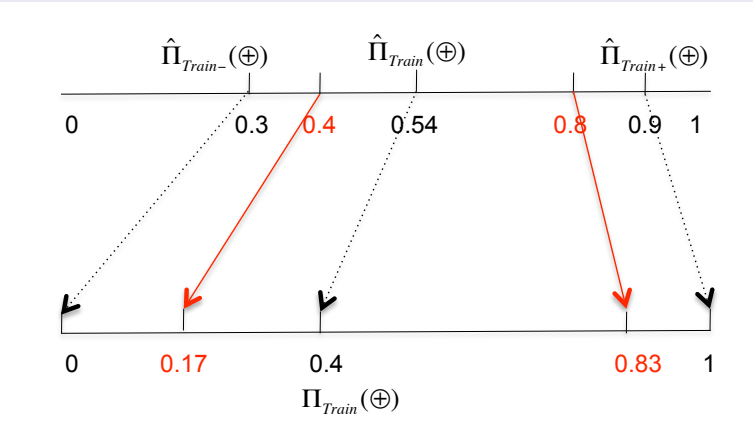
- Problem: If the proportions of positives in training and test sets are different, the result will not be satisfactory.
- Solution: Use a proper scaling similar to AC method.

Scaled Probability Average (SPA) method

From the training set, it is possible to calculate

- Actual proportion of positive examples, $\pi_{Train}(\oplus)$.
- Estimated positive probability average, $\hat{\pi}_{Train}(\oplus)$.
- Estimated positive probability average for the positives, $\hat{\pi}_{Train_{\oplus}}(\oplus)$.
- Estimated positive probability average for the negatives, $\hat{\pi}_{Train_{\ominus}}(\oplus)$.

$$\hat{\pi}^{SPA}_{Test}(\oplus) = \frac{\hat{\pi}^{PA}_{Test}(\oplus) - \hat{\pi}_{Train_{\ominus}}(\oplus)}{\hat{\pi}_{Train_{\oplus}}(\oplus) - \hat{\pi}_{Train_{\ominus}}(\oplus)}$$



Scaled Classify & Count (SCC) method

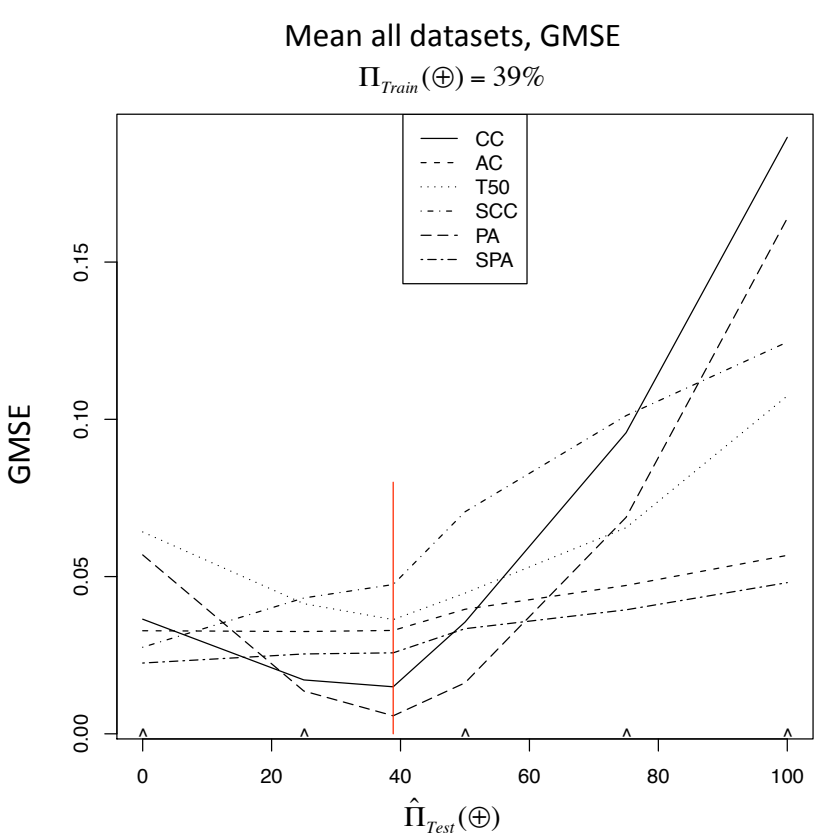
Scaling CC method in the same way as SPA method.

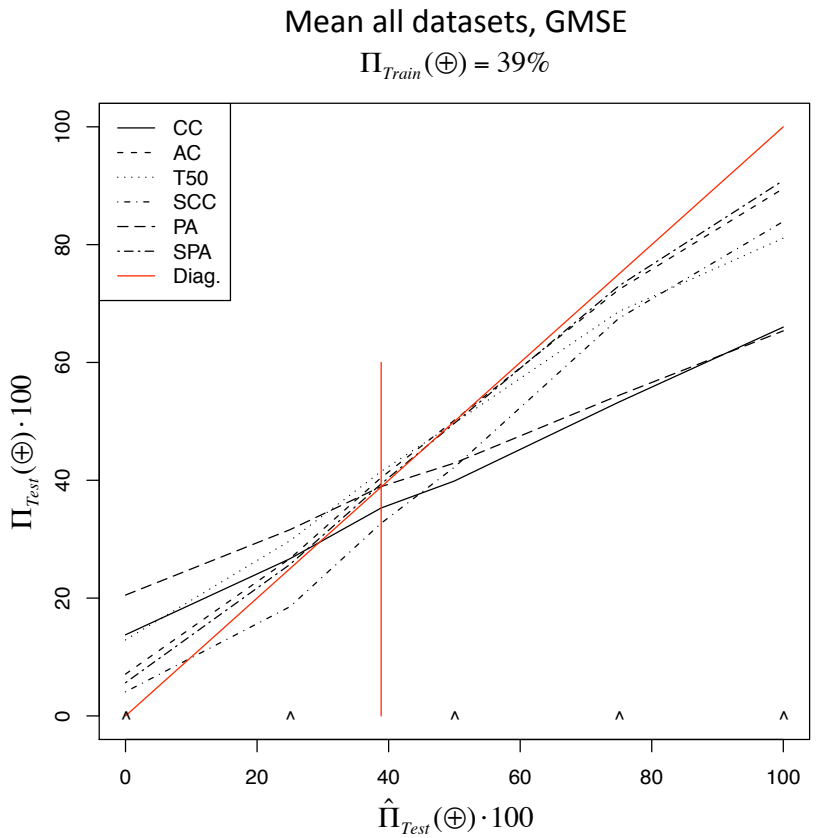
$$\hat{\pi}^{SCC}_{Test}(\oplus) = \frac{\sum_{i \in Test} C(i,\oplus) - \hat{\pi}_{Train_\ominus}(\oplus)}{\hat{\pi}_{Train_\oplus}(\oplus) - \hat{\pi}_{Train_\ominus}(\oplus)}$$

Experimental settings

- Methods: CC, AC, T50, SCC, PA and SPA.
- 6 test sets: original proportion of classes, 100% of positives and 0% of negatives, 75% and 25%, 50% and 50%, 25% and 75%, and 0% and 100%.
- Probabilistic classifiers: Nave Bayes, J48, IBk (k=10) and Logistic.
- 100 repetitions (25 times each classifier) 20 binary datasets from UCI repository.

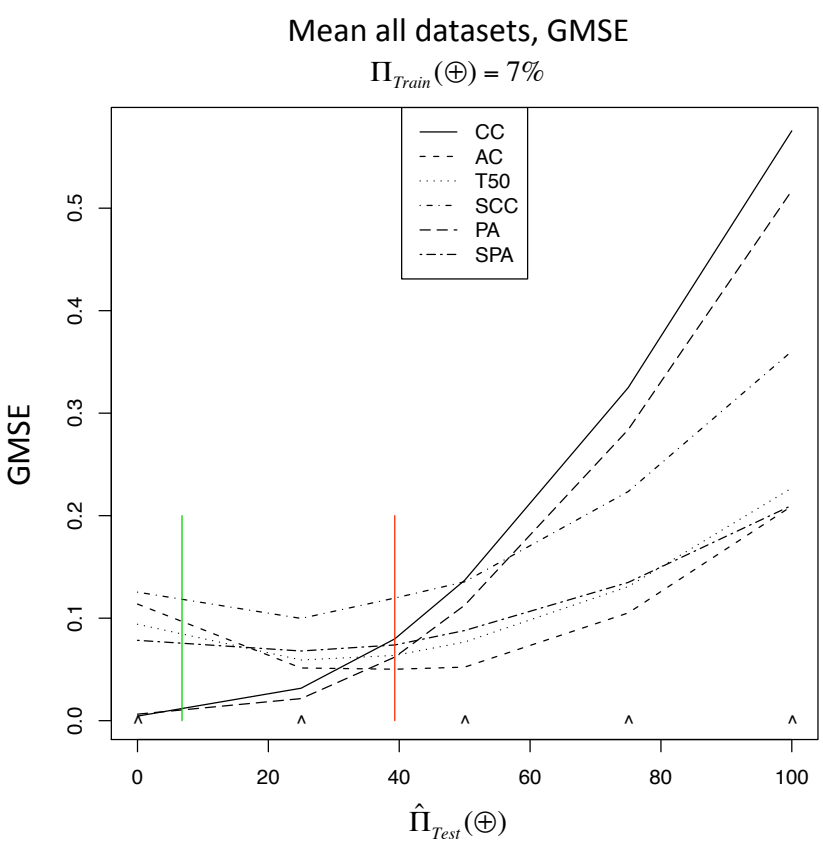
Analizing the effect of test imbalance

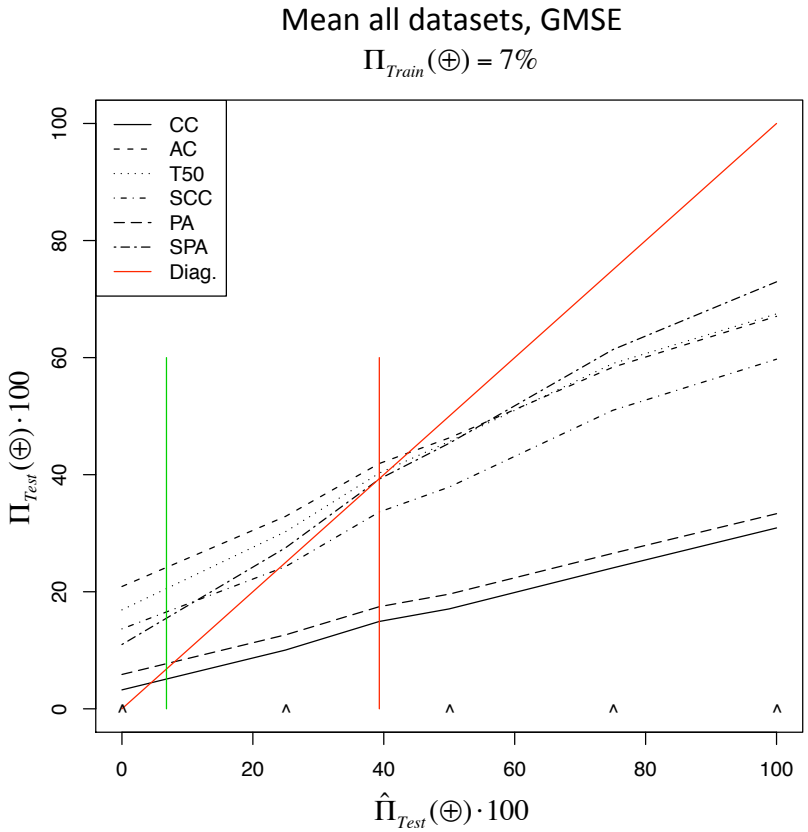




- The best results are obtained with class proportions which are close to the training class proportion.
- The more robust methods to changes in the class distributions are SPA and PA, which overcome the other methods.

Analizing the effect of imbalance in the training





- The SPA method is still obtaining good results, but in most cases the differences between the SPA method and the AC and T50 methods are not statistically significant.

Conclusions and Future Work

Conclusions

- One of the most natural ways to address the quantification problem is to average the probability estimations for each class (more information is used, the choice of a good threshold is not so important).
- We have derived a generalisation of Forman's scaling for probabilities, and we have derived a new method from it.
- The results are highly positive, and show that the use of probability estimators for quantification is a good way to solve it.

Future Work

- To explore other experimental settings.
- To extend quantification for more than two classes.

Thanks for your attention