# Sistemas de Múltiplos Classificadores

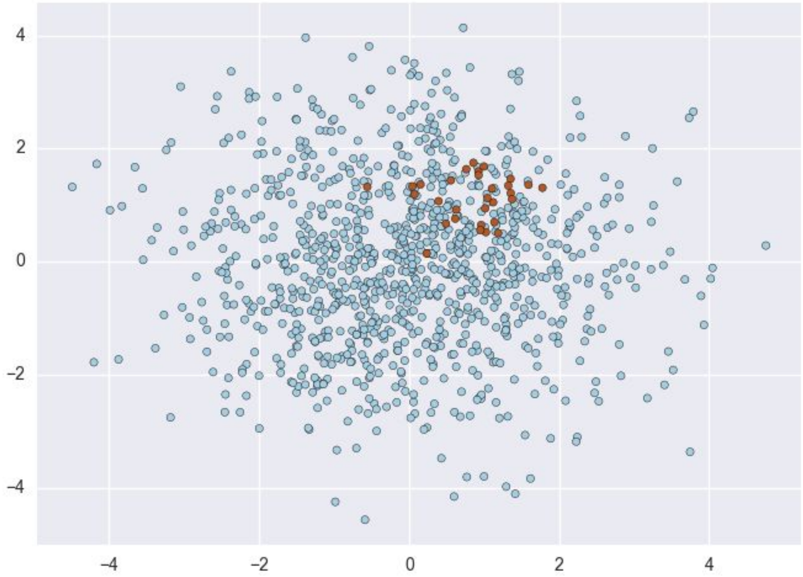
## A novel ensemble method for classifying imbalanced data

Zhongbin Sun, Qinbao Song, Xiaoyan Zhu, Heli Sun, Baowen Xu, Yuming Zhou

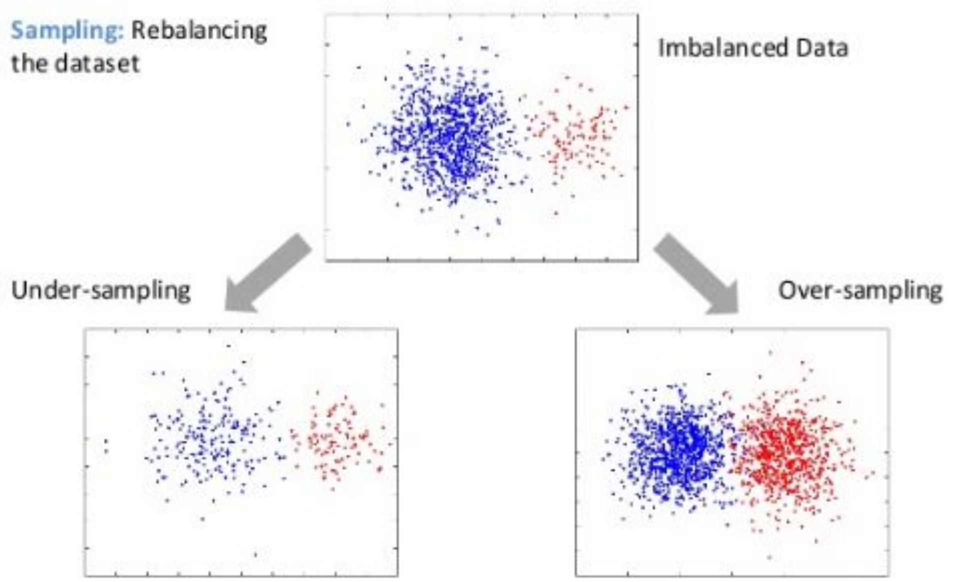
Pattern Recognition, 2014

Daniel Bion Barreiros - dbb2@cin.ufpe.br

# Dados desbalanceados

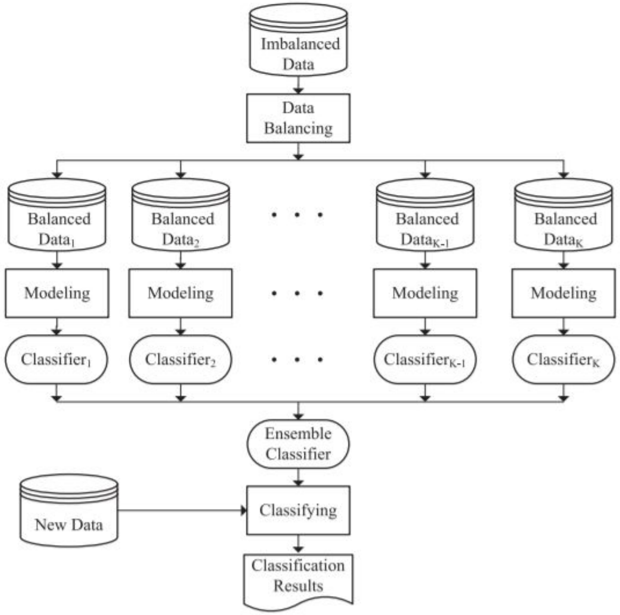


# Sampling



Sampling: Rebalancing the dataset
Imbalanced Data
Under-sampling
Over-sampling

# ClusterBal e SplitBal



# Regras de combinação (literatura)

| Rule | Strategy | Description |
|---|---|---|
| Max | $R_1 = \arg\max_{1 \le i \le K} P_{i1},\ R_2 = \arg\max_{1 \le i \le K} P_{i2}$ | Use the maximum classification probability of these $K$ classifiers for each class label |
| Min | $R_1 = \arg\min_{1 \le i \le K} P_{i1},\ R_2 = \arg\min_{1 \le i \le K} P_{i2}$ | Use the minimum classification probability of these $K$ classifiers for each class label |
| Product | $R_1 = \prod_{i=1}^{K} P_{i1},\ R_2 = \prod_{i=1}^{K} P_{i2}$ | Use the product of classification probability of these $K$ classifiers for each class label |
| Majority vote | $R_1 = \sum_{i=1}^{K} f(P_{i1}, P_{i2}),\ R_2 = \sum_{i=1}^{K} f(P_{i2}, P_{i1})$ | For the $i$th classifier, if $P_{i1} \ge P_{i2}$, class $C_1$ gets a vote, if $P_{i2} \ge P_{i1}$, class $C_2$ gets a vote |
| Sum | $R_1 = \sum_{i=1}^{K} P_{i1},\ R_2 = \sum_{i=1}^{K} P_{i2}$ | Use the summation of classification probability of these $K$ classifiers for each class label |

The function f(x,y) is defined as follows:

$$f(x,y) = \begin{cases} 1 & x \ge y \\ 0 & x < y \end{cases} \quad (1)$$

# Regras de combinação (propostas)

| Rule | Strategy | Description |
|---|---|---|
| MaxDistance | $R_1 = \arg\max_{1 \le i \le K} \frac{P_{i1}}{D_{i1}+1}$, $R_2 = \arg\max_{1 \le i \le K} \frac{P_{i2}}{D_{i2}+1}$ | Use the inverse of average distance to adjust the Max Rule |
| MinDistance | $R_1 = \arg\min_{1 \le i \le K} \frac{P_{i1}}{D_{i1}+1}$, $R_2 = \arg\min_{1 \le i \le K} \frac{P_{i2}}{D_{i2}+1}$ | Use the inverse of average distance to adjust the Min Rule |
| ProDistance | $R_1 = \prod_{i=1}^{K} \frac{P_{i1}}{D_{i1}+1}$, $R_2 = \prod_{i=1}^{K} \frac{P_{i2}}{D_{i2}+1}$ | Use the inverse of average distance to adjust the Product Rule |
| MajDistance | $R_1 = \sum_{i=1}^{K} \frac{f(P_{i1},P_{i2})}{D_{i1}+1}$, $R_2 = \sum_{i=1}^{K} \frac{f(P_{i2},P_{i1})}{D_{i2}+1}$ | Use the inverse of average distance to adjust the Majority Vote Rule |
| SumDistance | $R_1 = \sum_{i=1}^{K} \frac{P_{i1}}{D_{i1}+1}$, $R_2 = \sum_{i=1}^{K} \frac{P_{i2}}{D_{i2}+1}$ | Use the inverse of average distance to adjust the Sum Rule |

# Configurações

- 46 conjuntos de dados desbalanceados
- 10-fold Cross Validation
- 6 algoritmos base do Weka (Naive Bayes, C4.5, RIPPER, Random Forest, SMO e IBK)
- 10 regras de combinação de classificadores
- AUC (Area Under ROC Curve)

# ClusterBal + Regras

| Classifier | Max | Min | Product | Majority | Sum | MaxDistance | MinDistance | ProDistance | MajDistance | SumDistance |
|---|---|---|---|---|---|---|---|---|---|---|
| Naive Bayes | 0.8351 | 0.8457 | 0.8158 | 0.8350 | 0.8305 | 0.8041 | 0.8360 | 0.8297 | 0.8340 | **0.8467** |
| C4.5 | 0.7704 | 0.7715 | 0.7826 | 0.7772 | 0.7667 | 0.7747 | 0.7757 | 0.7783 | 0.7620 | **0.7835** |
| RIPPER | 0.7838 | 0.7633 | 0.7689 | 0.7757 | 0.7791 | 0.7645 | 0.7776 | 0.7672 | **0.7875** | 0.7619 |
| Random Forest | 0.8844 | 0.8538 | 0.8646 | 0.8895 | 0.8886 | 0.8888 | **0.8896** | 0.8562 | 0.8878 | 0.8479 |
| SMO | 0.7899 | 0.7836 | 0.7895 | 0.8090 | 0.7835 | 0.8001 | 0.7791 | 0.7875 | **0.8189** | 0.7961 |
| IBK | 0.8269 | 0.8370 | 0.8310 | 0.8327 | 0.8360 | 0.8340 | 0.8531 | **0.8588** | 0.8422 | 0.8385 |

| Classifier | Max | Min | Product | Majority | Sum | MaxDistance | MinDistance | ProDistance | MajDistance | SumDistance |
|---|---|---|---|---|---|---|---|---|---|---|
| Naive Bayes | 3 | 10 | 8 | 9 | 4 | 2 | 6 | 5 | 7 | 1 |
| C4.5 | 2 | 9 | 8 | 9 | 4 | 1 | 6 | 7 | 5 | 3 |
| RIPPER | 2 | 8 | 8 | 8 | 4 | 1 | 5 | 7 | 6 | 3 |
| Random Forest | 9 | 4 | 4 | 7 | 10 | 8 | 2 | 1 | 3 | 6 |
| SMO | 3 | 8 | 8 | 8 | 4 | 2 | 7 | 5 | 6 | 1 |
| IBK | 8 | 5 | 5 | 9 | 10 | 2 | 3 | 1 | 7 | 4 |
| Sum | 27 | 44 | 41 | 50 | 36 | 16 | 29 | 26 | 34 | 18 |
| Rank | 4 | 9 | 8 | 10 | 7 | 1 | 5 | 3 | 6 | 2 |

# SplitBal + Regras

| Classifier | Max | Min | Product | Majority | Sum | MaxDistance | MinDistance | ProDistance | MajDistance | SumDistance |
|---|---|---|---|---|---|---|---|---|---|---|
| Naive Bayes | 0.6155 | 0.5330 | 0.5354 | 0.5453 | 0.6112 | 0.6393 | 0.5869 | 0.5892 | 0.5690 | **0.6473** |
| C4.5 | 0.6648 | 0.5262 | 0.5263 | 0.5262 | 0.6411 | **0.6847** | 0.5932 | 0.5291 | 0.5963 | 0.6595 |
| RIPPER | 0.6770 | 0.5223 | 0.5223 | 0.5223 | 0.6563 | **0.7216** | 0.6249 | 0.6024 | 0.6043 | 0.6759 |
| Random Forest | 0.6798 | 0.7643 | 0.7643 | 0.7148 | 0.6746 | 0.7016 | 0.8072 | **0.8274** | 0.7827 | 0.7393 |
| SMO | 0.7459 | 0.5159 | 0.5159 | 0.5159 | 0.7177 | 0.7632 | 0.5208 | 0.5783 | 0.5562 | **0.7673** |
| IBK | 0.7299 | 0.7816 | 0.7816 | 0.7187 | 0.7176 | 0.8171 | 0.8025 | **0.8378** | 0.7678 | 0.7987 |

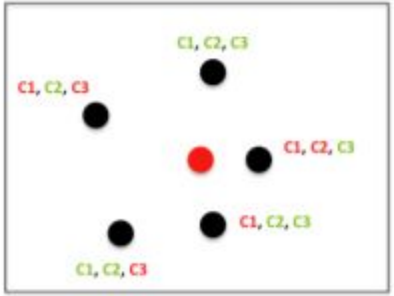
| Classifier | Max | Min | Product | Majority | Sum | MaxDistance | MinDistance | ProDistance | MajDistance | SumDistance |
|---|---|---|---|---|---|---|---|---|---|---|
| Naive Bayes | 4 | 2 | 9 | 5 | 7 | 10 | 3 | 8 | 6 | 1 |
| C4.5 | 8 | 7 | 2 | 4 | 9 | 6 | 5 | 3 | 10 | 1 |
| RIPPER | 2 | 9 | 6 | 5 | 3 | 8 | 4 | 7 | 1 | 10 |
| Random Forest | 6 | 9 | 7 | 2 | 4 | 3 | 1 | 8 | 5 | 10 |
| SMO | 5 | 8 | 6 | 2 | 9 | 3 | 10 | 7 | 1 | 4 |
| IBK | 10 | 5 | 9 | 8 | 6 | 7 | 2 | 1 | 3 | 4 |
| Sum | 35 | 40 | 39 | 26 | 38 | 37 | 25 | 34 | 26 | 30 |
| Rank | 6 | 10 | 9 | 2 | 8 | 7 | 1 | 5 | 2 | 4 |

# Problema do ClusterBal

| Bin | SplitBal | | ClusterBal | |
|---|---|---|---|---|
| | Majority | Minority | Majority | Minority |
| 1 | 321 | 293 | 760 | 293 |
| 2 | 321 | 293 | 525 | 293 |
| 3 | 321 | 293 | 154 | 293 |
| 4 | 321 | 293 | 82 | 293 |
| 5 | 321 | 293 | 83 | 293 |

# OLA e LCA

K = 5



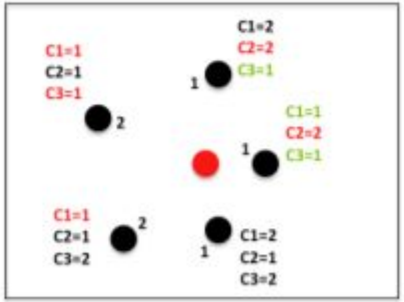

Resultado:

C1 = 2/5 = 0,4 = **3°**

**C2 = 4/5 = 0,8 = 1°**

C3 = 3/5 = 0,6 = **2°**

K = 5




C1(●) = 1
C2(●) = 2
C3(●) = 1

Resultado:

C1 = 1/3 = 0,33 = 2°

C2 = 0/2 = 0,0 = 3°

**C3 = 2/3 = 0,67 = 1°**

# Resultado Final

| | | Conventional Imbalance Methods | | | Proposed by the Author | | My methods | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Classifier | Original | RUS | ROS | SMOTE | SplitBal+MinDist | ClusterBal+MaxDist | SplitBal+OLA | SplitBal+LCA | ClusterBal+OLA | ClusterBal+LCA |
| Naive Bayes | 0.8130 | 0.8157 | 0.8139 | 0.8095 | 0.8360 | 0.6393 | 0.8395 | **0.8453** | 0.6708 | 0.7191 |
| C4.5 | 0.8162 | 0.8159 | 0.8153 | 0.8160 | 0.7757 | 0.6847 | **0.8445** | 0.8106 | 0.6814 | 0.8206 |
| RIPPER | 0.8118 | 0.8168 | 0.8150 | 0.8144 | 0.7776 | 0.7216 | **0.8302** | 0.7870 | 0.7130 | 0.8113 |
| Random Forest | 0.8165 | 0.8150 | 0.8157 | 0.8156 | **0.8896** | 0.7016 | 0.8762 | 0.8707 | 0.7038 | 0.8386 |
| SMO | 0.8138 | 0.8161 | 0.8171 | 0.8173 | 0.7791 | 0.7632 | **0.8473** | 0.8451 | 0.7341 | 0.8232 |
| IBK | 0.8172 | 0.8189 | 0.8159 | 0.8159 | 0.8531 | 0.8151 | **0.8545** | 0.8459 | 0.7128 | 0.8255 |

# Observações

1. O autor propõe um método de combinação dinâmica de classificadores e compara com métodos de tratamento de dados desbalanceados, ou seleção estática;
2. No artigo original o ClusterBal tem bom desempenho, porém não fica claro se existe algum passo adicional para fazê-lo funcionar, pois os problemas listados no artigo aconteceram até mesmo na replicação.
3. As regras de combinação propostas são muito lentas;

# Referências


[1] Zhongbin Sun, Qinbao Song, Xiaoyan Zhu, Heli Sun, Baowen Xu, Yuming Zhou, A novel ensemble method for classifying imbalanced data, Pattern Recognition, Volume 48, Issue 5, 2014, Pages 1623-1637.

[2] J. Alcalá-Fdez, L. Sánchez, S. García, M. del Jesus, S. Ventura, J. Garrell, J. Otero, C. Romero, J. Bacardit, V. Rivas, et al., Keel: a software tool to assess evolutionary algorithms for data mining problems, Soft Comput. – Fusion Found. Methodol. Appl. 13 (2009) 307–318.

[3] Witten I. H. and Frank E. (2005). Data Mining: Practical Machine Learning Tools and Techniques, 2nd edition. Morgan Kaufmann, San Francisco.

[4] Hornik K., Buchta C. and Zeileis A. (2009). Open-Source Machine Learning: R Meets Weka. Computational Statistics, 225-232.

[5] Woods K., Kegelmeyer Jr W. P. and Bowyer K., Combination of multiple classifiers using local accuracy estimates, Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 19, no. 4, pp. 405-410, 1997.