Seminario Permanente de Formación
en Inteligencia Artificial
Aplicada a la Defensa

Clasificación en conjuntos de datos con clases no balanceadas

Alberto Fernández

Instituto Andaluz de Investigación en Data Science and Computational Intelligence (DaSCI)

Dpto. Ciencias de la Computación e I.A.
Universidad de Granada

alberto@decsai.ugr.es
https://www.dasci.es

UNIVERSIDAD DE GRANADA

DaSCI

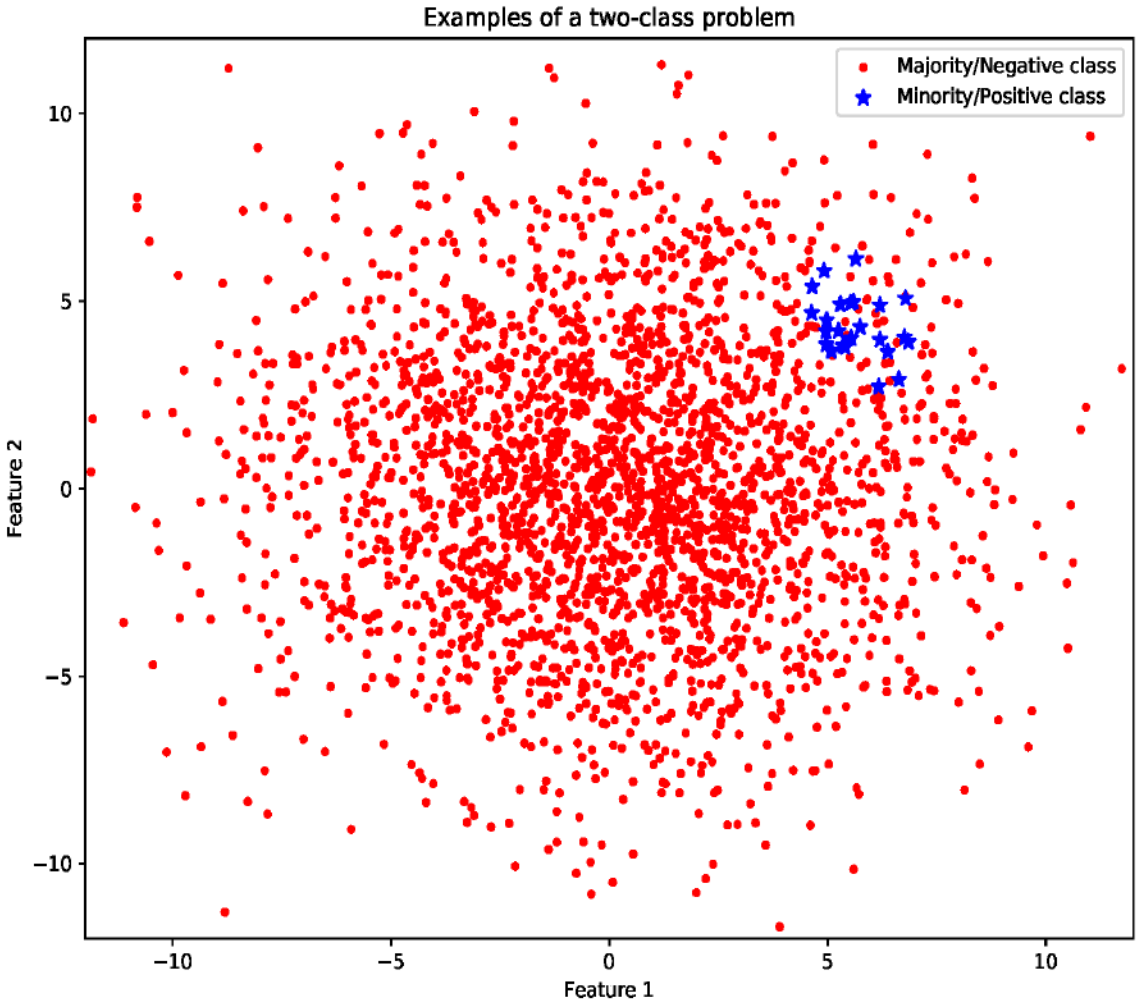
Outline

- Introduction: Definition, properties and difficulty
- Evaluation metrics
- Data Intrinsic Characteristics
- Addressing imbalanced datasets
- Software tools for classification with imbalanced data
- Final Comments
- Surveys for a deeper study

Outline

- Introduction: Definition, properties and difficulty
- Evaluation metrics
- Data Intrinsic Characteristics
- Addressing imbalanced datasets
- Software tools for classification with imbalanced data
- Final Comments
- Surveys for a deeper study

Problem Definition



Problem Definition

- Real application areas in engineering characterised by having **a very different distribution** of examples among their classes.
- Intrinsic to the problem or due to **limitations** during the data collection process.
- **Positive class** often represents the concept of the highest interest for the problem, whereas the negative class represents counter-examples.
- Problem of imbalanced data-sets: **imposes a bias** for the correct identification of the different concepts to be learnt.

Example of applications

Year	Domain	Subcategory	Application	Data-level	Internal	Cost-sensitive	Ensemble
2017	Medicine	Prognosis	Donor-recipient matching prediction in liver transplantation	×			
2017	Security	Video surveillance	Still-to-video face recognition			×	
2017	Medicine	Diagnosis	Detection of microaneurysm				×
2018	Engineering	Rotating machinery	Fault diagnosis in wind turbines	×		×	×
2018	Information technology	Computer vision	Object recognition in images				×
2018	Engineering	Rotating machinery	Fault diagnosis in wind turbines			×	
2018	Security	Video surveillance	Face re-identification				×
2018	Engineering	Rotating machinery	Fault diagnosis in wind turbines				×

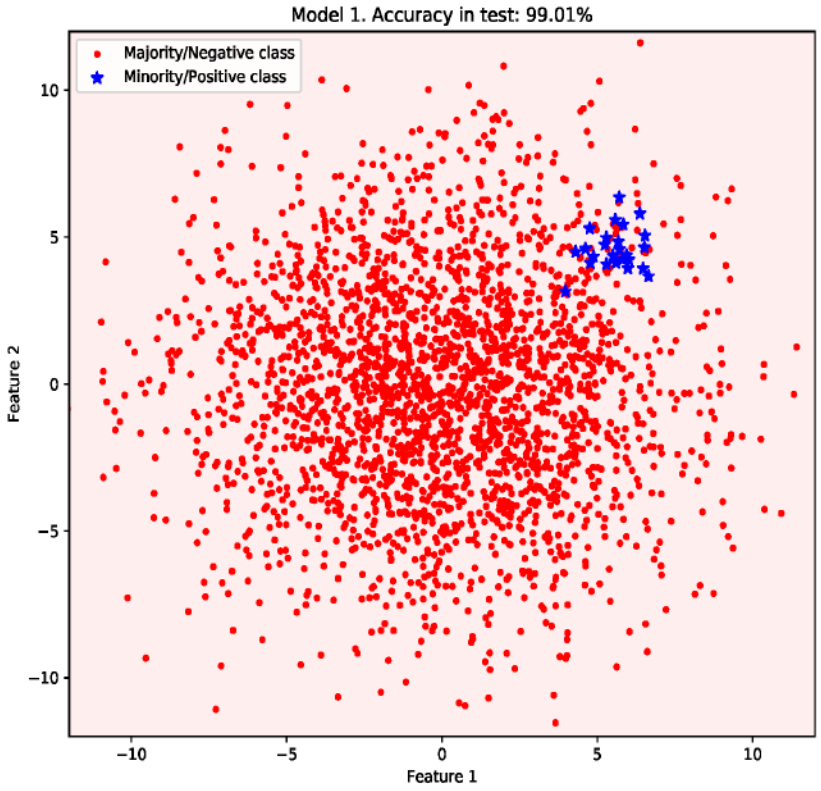
Taken from: G. Haixiang et al. ESwA 73 (2017), 220-239

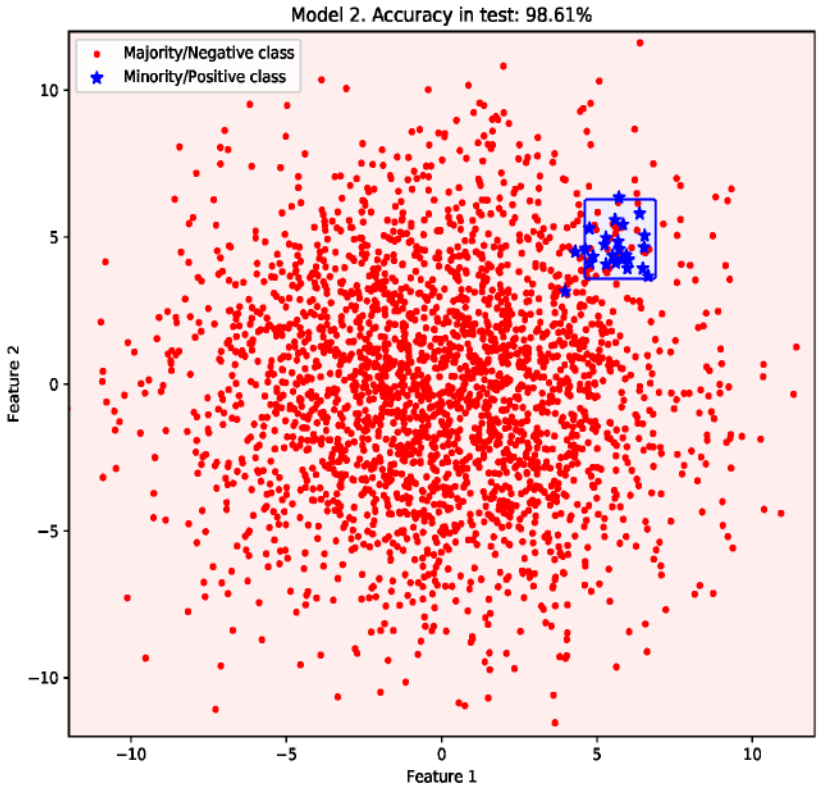
Example of applications

Application domains	No. of papers
(1) Chemical, biomedical engineering	47
(2) Financial management	37
(3) Information technology	24
(4) Energy management	8
(5) Security management	7
(6) Electronics and communications	6
(7) Infrastructure and industrial manufacturing	9
(8) Business management	7
(9) Emergency management	4
(10) Environmental management	5
(11) Policy, social and education	4
(12) Agriculture and horticulture	1
(13) Other areas and non-specific areas	3

Taken from: G. Haixiang et al. ESwA 73 (2017), 220-239

Example of Imbalanced Classification

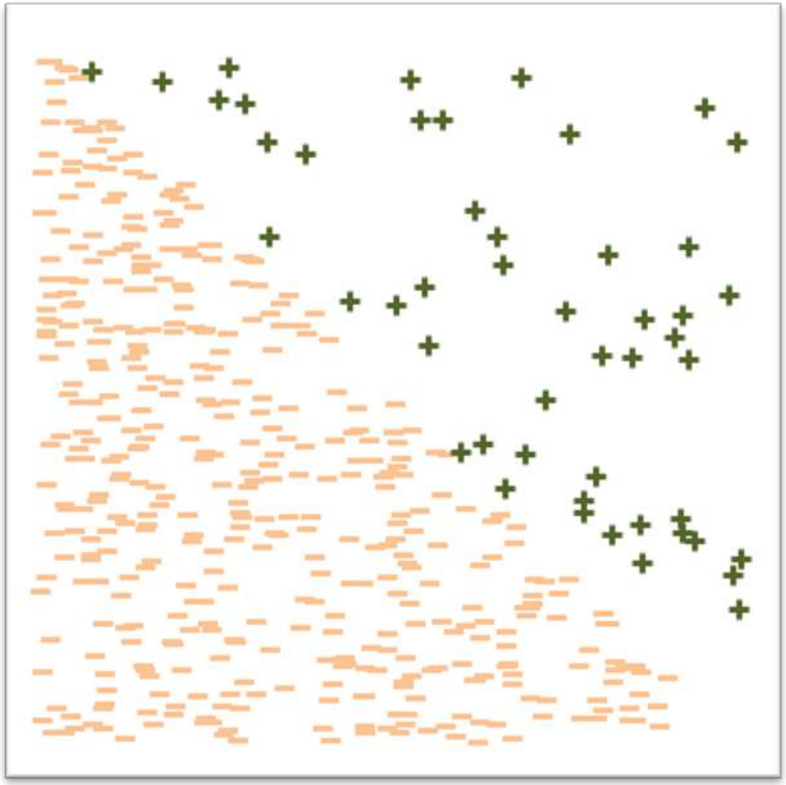




Why learning from imbalanced data-sets might be difficult?

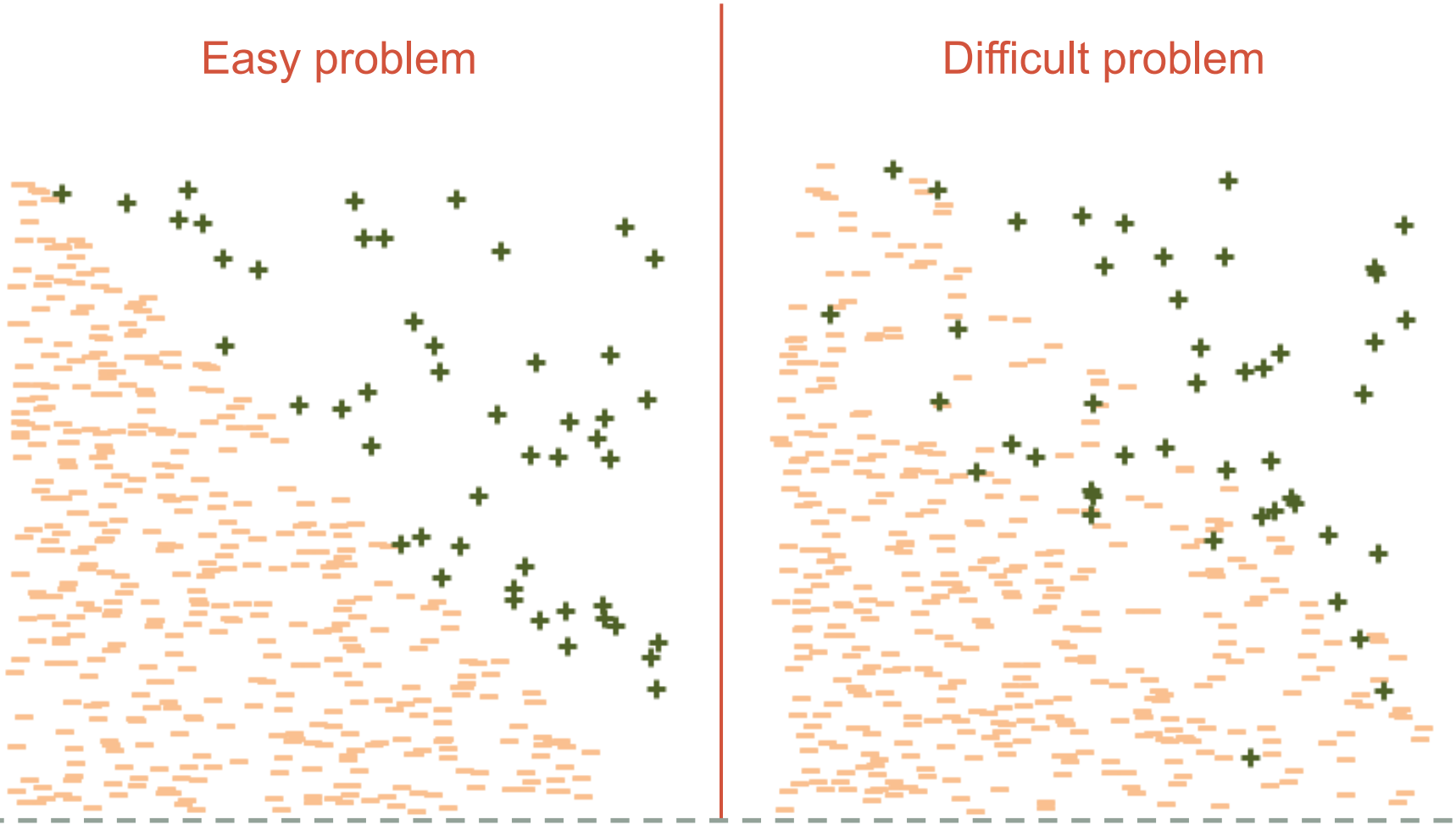
Properties and Difficulty

- Skewed class distribution:
  - Measured by the fraction between majority and minority samples
  - Imbalance ratio (IR)
- **Intrinsic Data Characteristics**
  - Not only imbalance hinders classification performance
  - IR ≈ 9



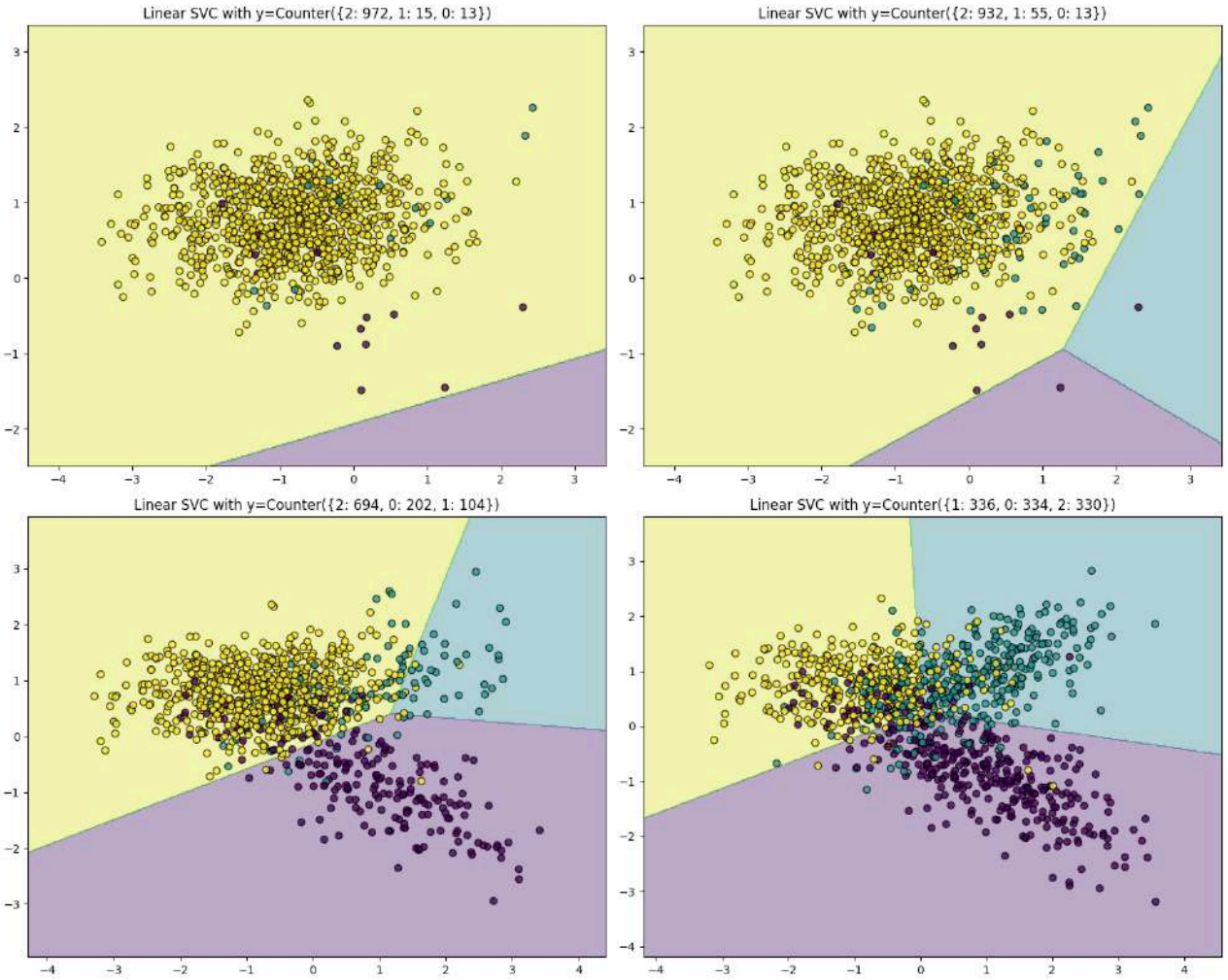
V. López, A. Fernandez, S. García, V. Palade, F. Herrera, **An Insight into Classification with Imbalanced Data: Empirical Results and Current Trends on Using Data Intrinsic Characteristics**. *Information Sciences* 250 (2013) 113-141

Intrinsic Data Issuess: Same class ratio



V. López, A. Fernandez, S. García, V. Palade, F. Herrera, **An Insight into Classification with Imbalanced Data: Empirical Results and Current Trends on Using Data Intrinsic Characteristics**. *Information Sciences* 250 (2013) 113-141

The importance of data representation



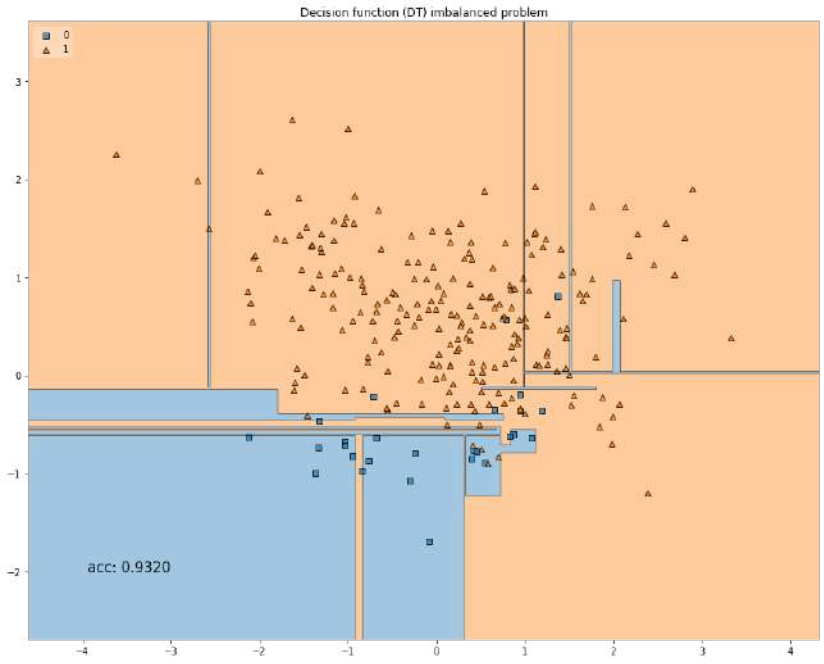
Taken from: https://imbalanced-learn.readthedocs.io/en/stable/introduction.html

Outline

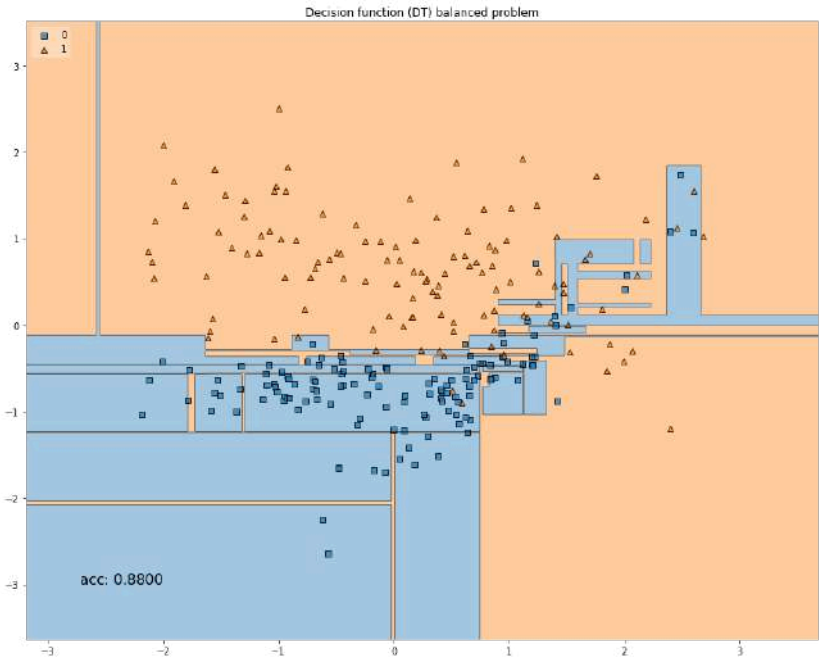
- Introduction: Definition, properties and difficulty
- Evaluation metrics
- Data Intrinsic Characteristics
- Addressing imbalanced datasets
- Software tools for classification with imbalanced data
- Final Comments
- Surveys for a deeper study

Evaluation: Common metrics (accuracy) may lead to erroneous conclusions

Imbalanced Problem: misses 7 out of 24 examples (blue): 30%



Balanced Problem: misses 15 out of 130 examples (blue): 11%



Evaluation: Measuring performance in imbalanced domains

	Positive Prediction	Negative Prediction
Positive Class	True Positive (TP)	False Negative (FN)
Negative Class	False Positive (FP)	True Negative (TN)

Diagonal of True Hits

- Classical evaluation:
- $acc = \frac{TP+TN}{TP+TN+FP+FN}$
- It does not take into account the *"Individual Rates"*,
  - Very important in imbalanced problems

Sensitivity and Specificity

- Positive true ratio (*sensitivity*):
  - $TPR = \frac{TP}{TP+FN}$
- Negative true ratio (*specificity*):
  - $TNR = \frac{TN}{TN+FP}$
- *Geometric Mean:*
  - $GM = \sqrt{TPR \cdot TNR}$

- Single class metrics provide a **unique vision** of the performance
- **Sensitivity** must be stressed
- The fraction only considers same class samples
- **Aggregation** functions are important for a global vision

R. Barandela, J.S. Sánchez, V. García, E. Rangel. Strategies for learning in class imbalance problems. Pattern Recognition 36:3 (2003) 849-851

F-Measure

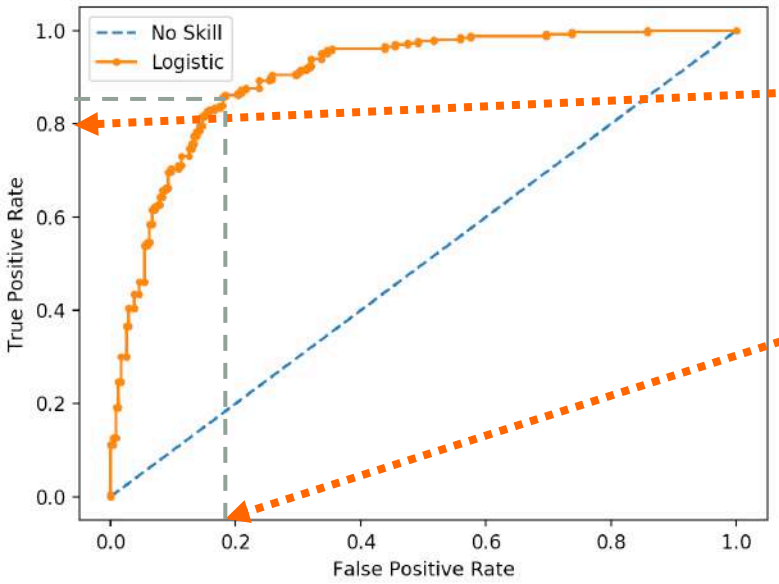
- F1 score is a harmonic mean between precision and recall:
  - Precision: number of correct positive results divided by the number of all positive results,
  - Recall / sensitivity: number of correct positive results divided by the number of positive results that should have been returned.

$$F_1 = 2 \cdot \frac{1}{\left(\frac{1}{recall} + \frac{1}{precision}\right)} = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

- General formula:

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall}$$

Area under ROC Curve (AUC): Scalar and graphical metric

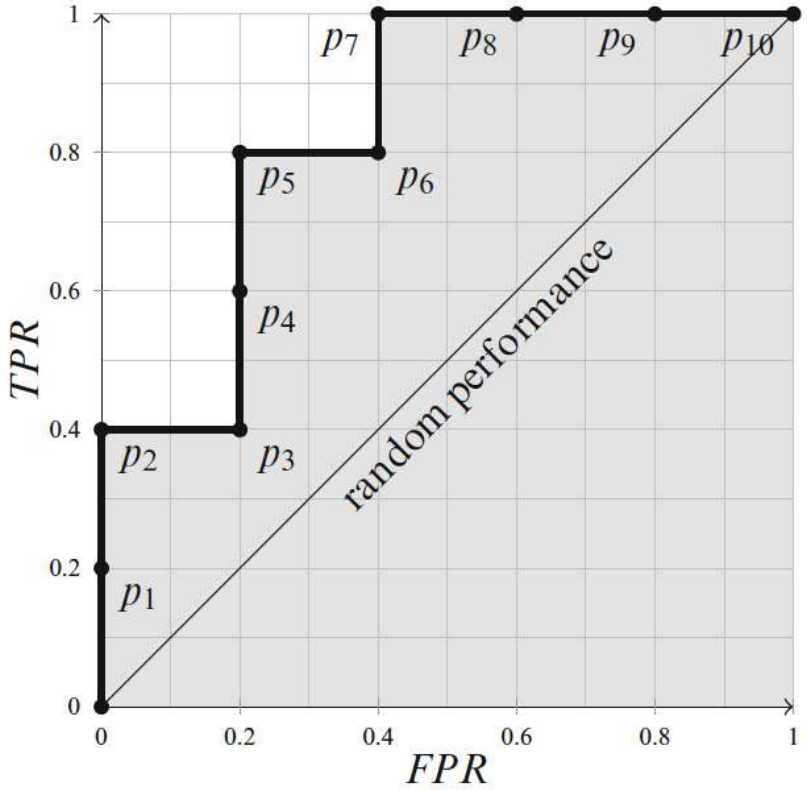


	Positive Prediction	Negative Prediction
Positive Class	0.82	0.1
Negative Class	0.18	0.9

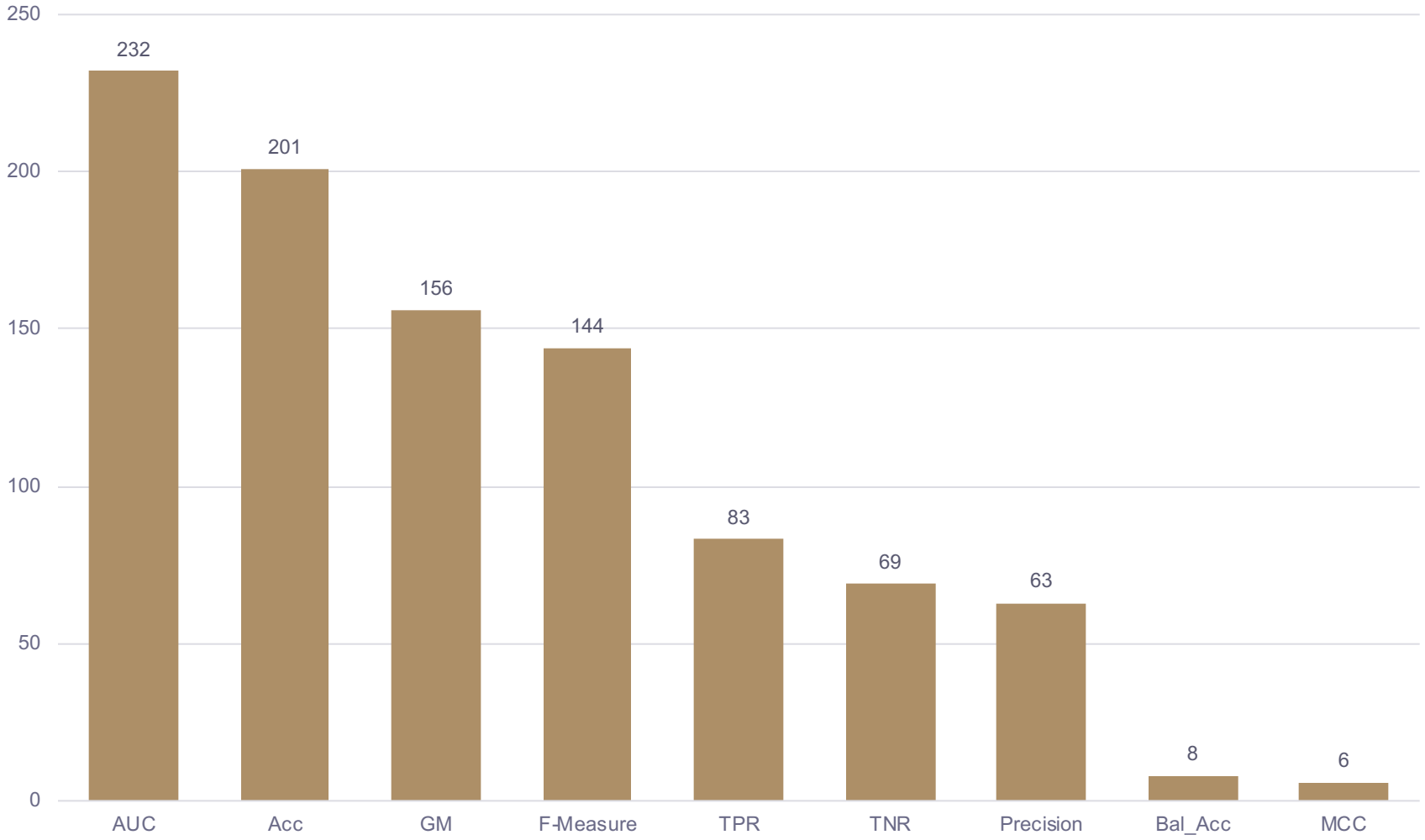
Default Probability (0.5): $AUC = \frac{1 + TPR - FPR}{2}$

Area under ROC Curve (AUC)

Rank	Score	Actual class	FPR	TPR	ROC point
#1	1.0	Positive	0.0	0.2	p_1
#2	0.9	Positive	0.0	0.4	p_2
#3	0.85	Negative	0.2	0.4	p_3
#4	0.7	Positive	0.2	0.6	p_4
#5	0.6	Positive	0.2	0.8	p_5
#6	0.45	Negative	0.4	0.8	p_6
#7	0.35	Positive	0.4	1.0	p_7
#8	0.3	Negative	0.6	1.0	p_8
#9	0.2	Negative	0.8	1.0	p_9
#10	0.05	Negative	1.0	1.0	p_{10}



Use of Metrics in the specialized literature



Taken from: G. Haixiang et al. ESwA 73 (2017), 220-239

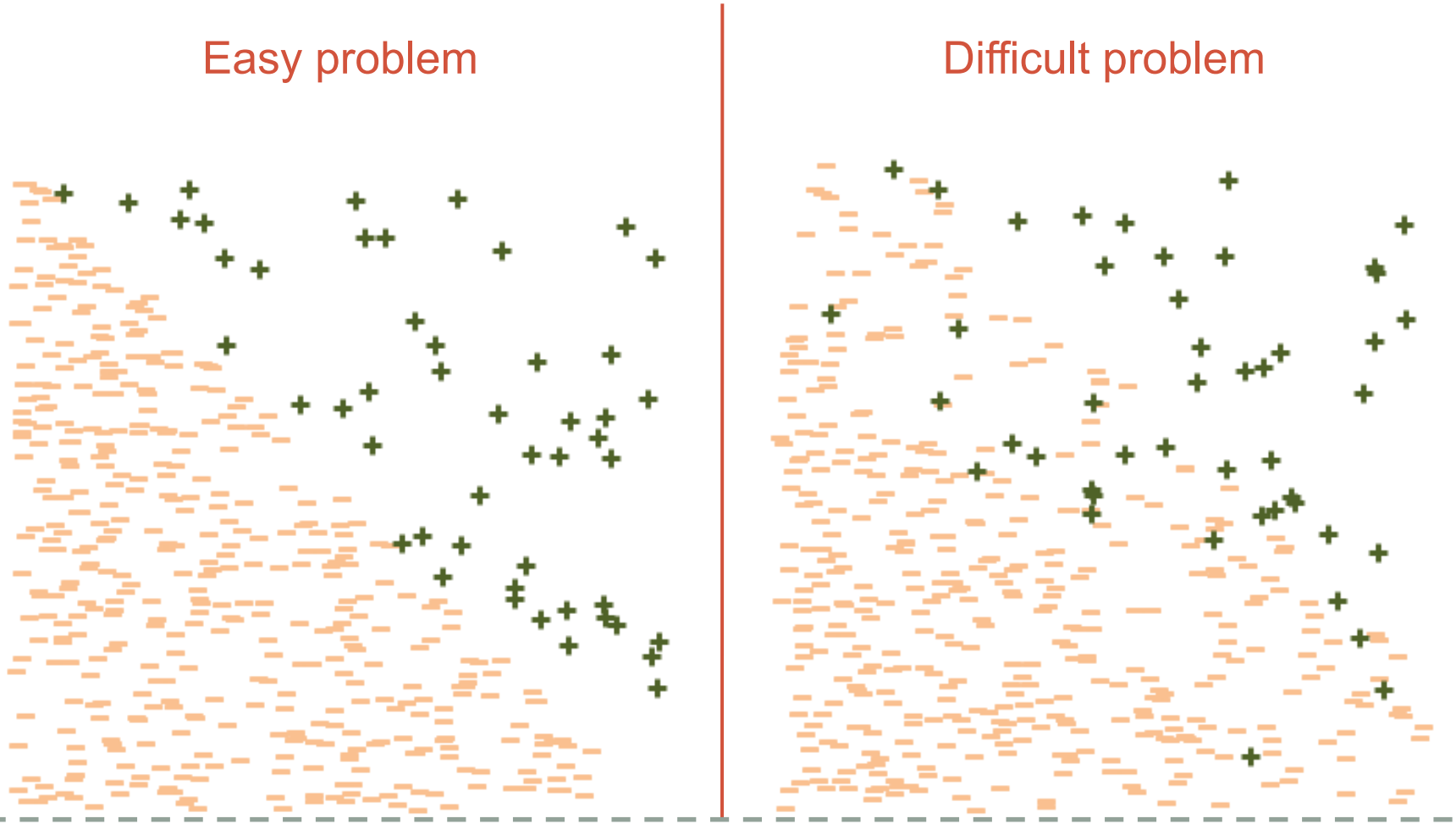
Outline

- Introduction: Definition, properties and difficulty
- Evaluation metrics
- Data Intrinsic Characteristics
- Addressing imbalanced datasets
- Software tools for classification with imbalanced data
- Final Comments
- Surveys for a deeper study

Data Intrinsic Characteristics in Imbalanced Classification

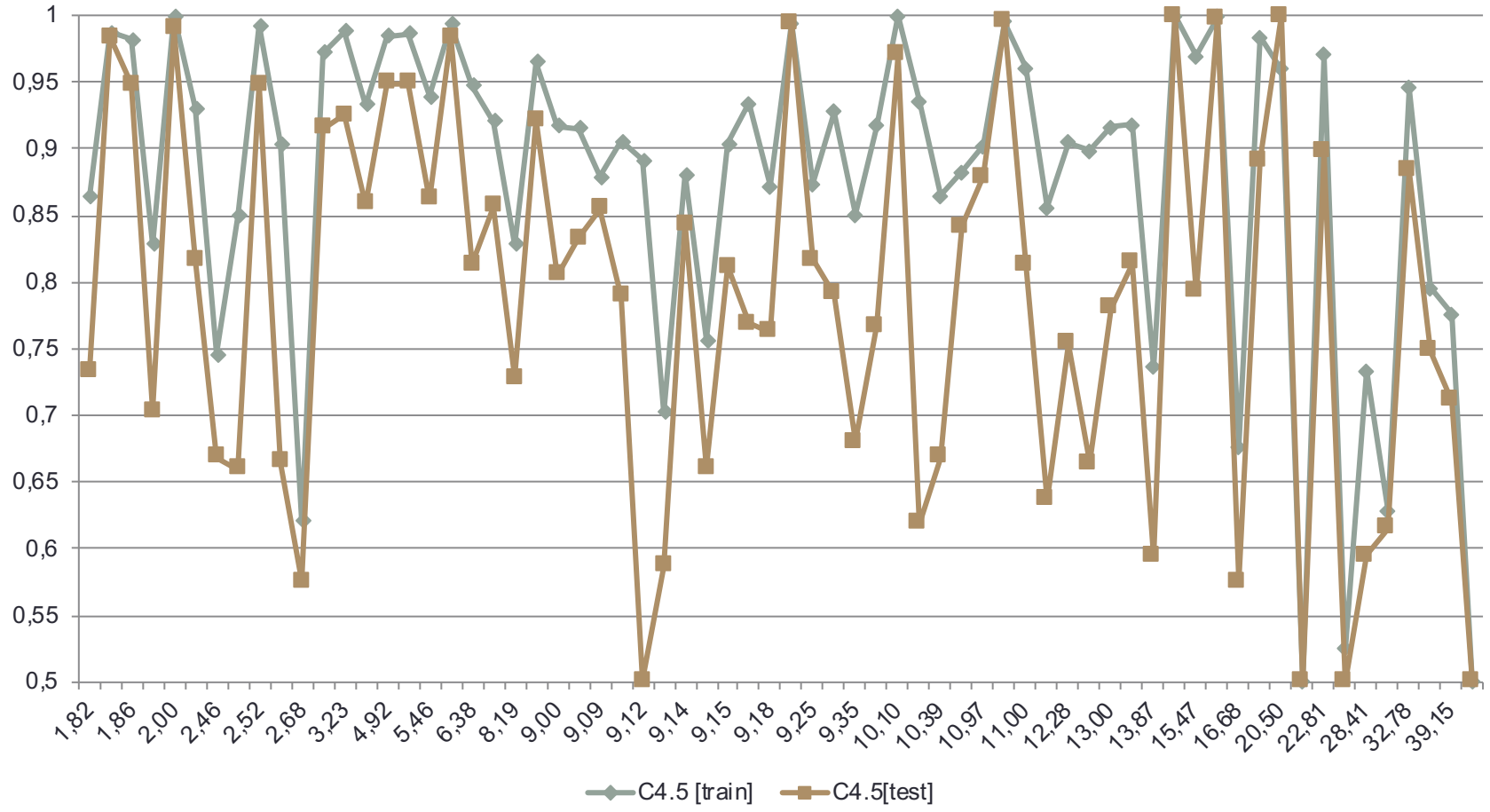
- Introduction
- Overlapping or class separability
- Small disjuncts
- Lack of density
- Noisy data and Borderline examples
- Dataset shift

Intrinsic Data Issuess: Same class ratio



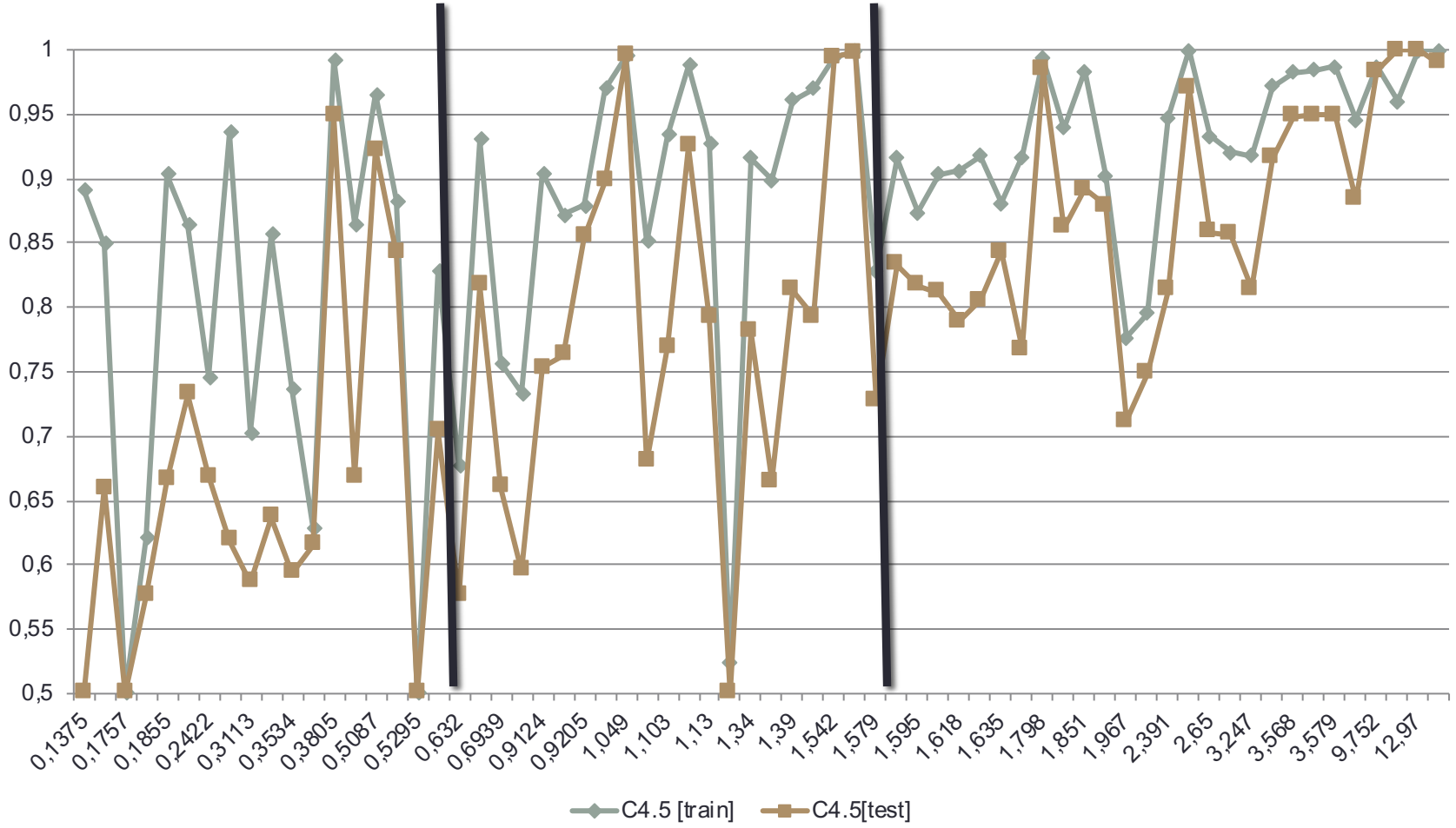
V. López, A. Fernandez, S. García, V. Palade, F. Herrera, **An Insight into Classification with Imbalanced Data: Empirical Results and Current Trends on Using Data Intrinsic Characteristics**. *Information Sciences* 250 (2013) 113-141

Data characterization: IR metric



J. Luengo, A. Fernandez, S. García, F. Herrera. **Addressing Data Complexity for Imbalanced Data Sets: Analysis of SMOTE-based Oversampling and Evolutionary Undersampling**. *Soft Computing*, 15 (10) (2011) 1909-1936

Data characterization: F1 metric

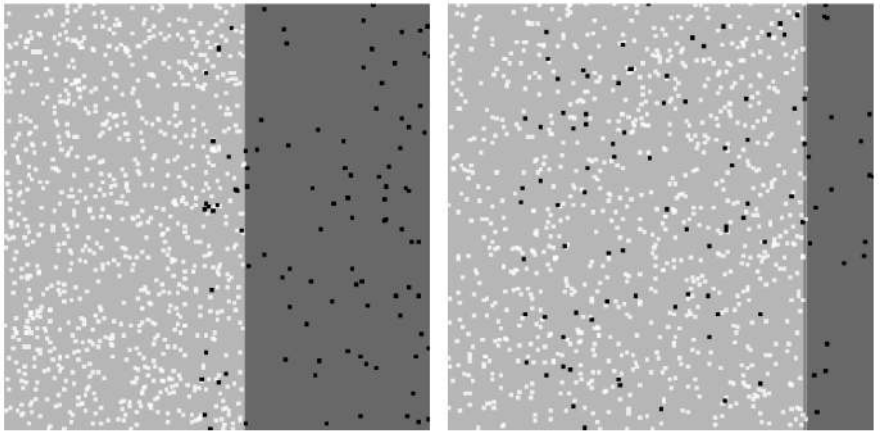


J. Luengo, A. Fernandez, S. García, F. Herrera. **Addressing Data Complexity for Imbalanced Data Sets: Analysis of SMOTE-based Oversampling and Evolutionary Undersampling**. *Soft Computing*, 15 (10) (2011) 1909-1936

Overlapping or class separability

- "Small" region of the data space is represented by a similar number of training data from both classes
- Inference mechanism with the same a priori probabilities:
  - Discrimination between classes becomes harder.
- "Linearly separable" problems solved by simple classifier
  - Regardless of class distribution

Overlap Degree	TP_{rate}	TN_{rate}	AUC
0 %	1.000	1.000	1.000
20 %	.79.00	1.000	.8950
40 %	.4900	1.000	.7450
50 %	.4700	1.000	.7350
60 %	.4200	1.000	.7100
80 %	.2100	.9989	.6044
100 %	.0000	1.000	.5000



(a) 20% of overlap (b) 80% of overlap

How to address overlapping?

- A dataset is represented by means of its attributes
- A large number of attributes may degrade the recognition of the borderline areas of the problem:
  - Some of these variables may be redundant
  - Some others may show a bad synergy among them
- The use of Feature Selection / Augmentation may allow to diminish the effect of overlapping
- Feature engineering itself does not solve the imbalanced classification problem
- It is mandatory to apply some of the standard solutions to address the problem.

Small Disjuncts

- The concepts are represented within small clusters
- Rare cases or Small disjuncts are those disjuncts in the learned classifier that cover few training examples.
  - Class A is the rare (minority) and B is the common (majority).
  - Subconcepts A2-A5 correspond to rare cases.
  - A1 corresponds to a fairly common case, covering a substantial portion of the instance space.
  - Subconcept B2 corresponds to a rare case, showing that common classes may contain rare cases.

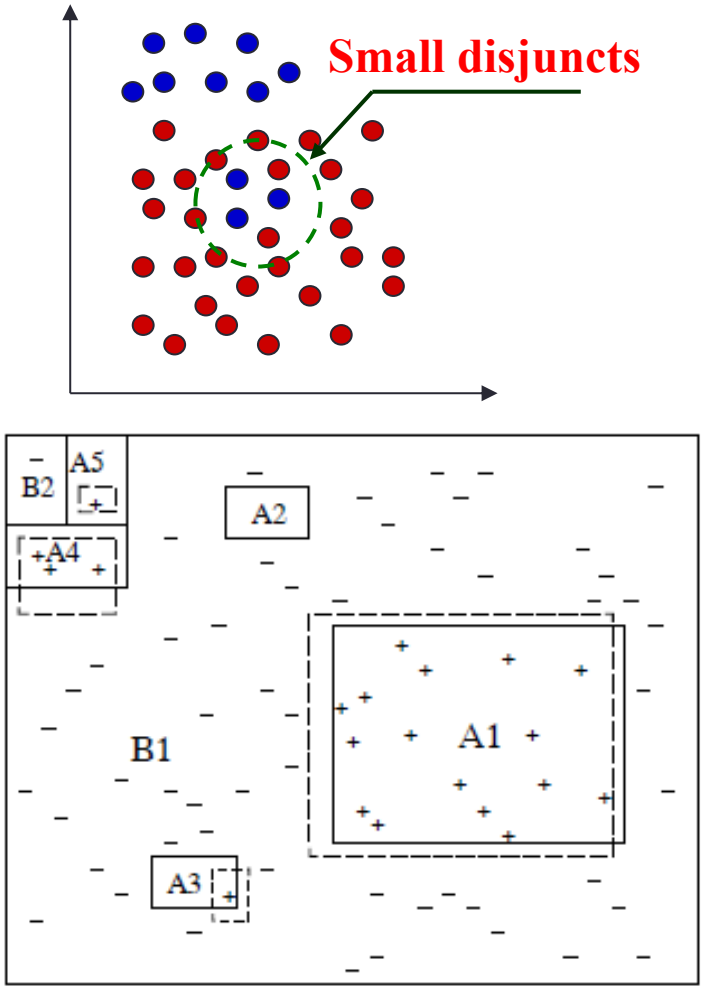


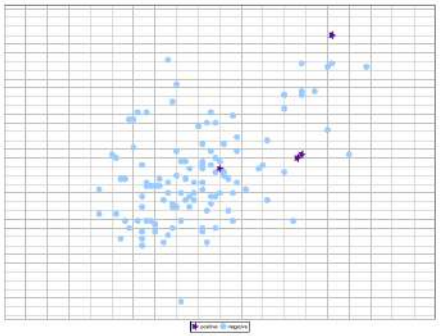
Figure 1: Graphical representation of a rare class and rare case

Small Disjuncts: How to address it?

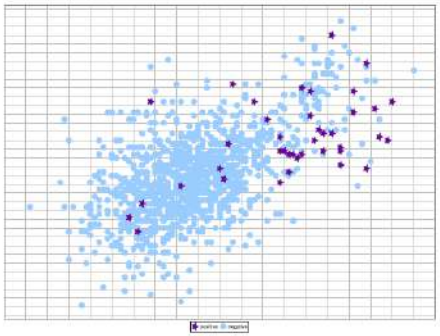
- Obtain additional training data.
- Use a more appropriate inductive bias (avoid Divide&Conquer).
- Apply overfitting management techniques (Disabling pruning)
- Employ boosting.
- SMOTE extensions that focus on density of data

Lack of density

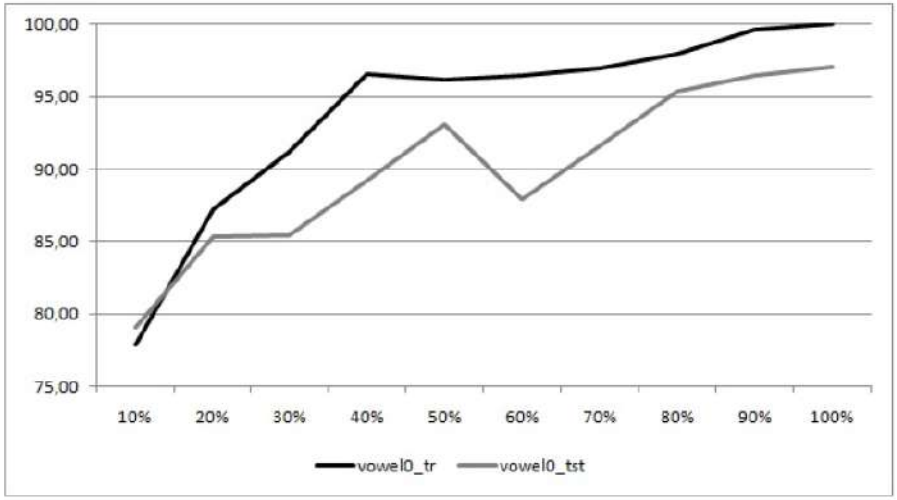
- Induction algorithms do not have enough data to generalise on sample distribution
- The knowledge model that learns this data space becomes too specific, leading to the overfitting problem.



(a) 10 % of training instances

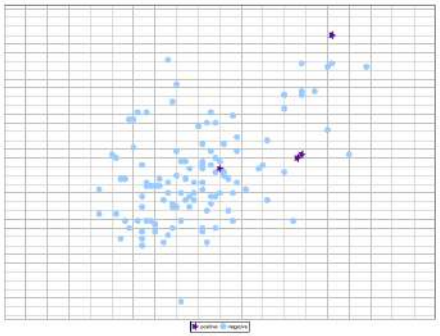


(b) 100 % of training instances

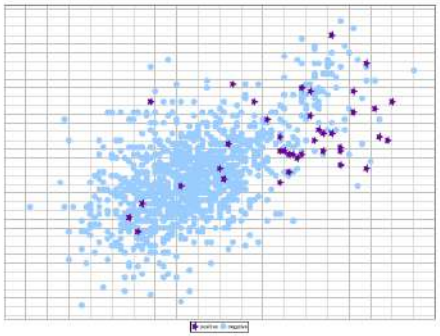


Lack of density (2)

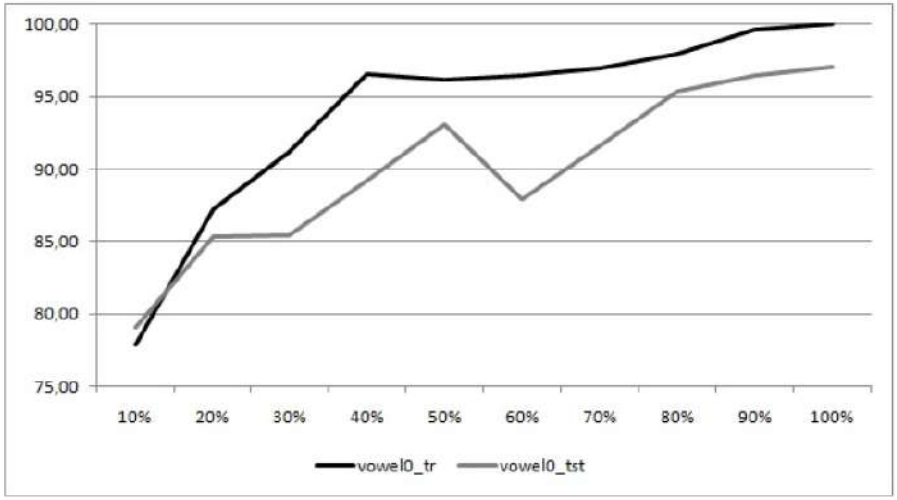
- The lack of density in the training data may also cause the small disjuncts
- Simple resampling mechanisms do not address the problem:
  - Collect more data!
  - Creating some synthetic instances could improve decision functions



(a) 10 % of training instances

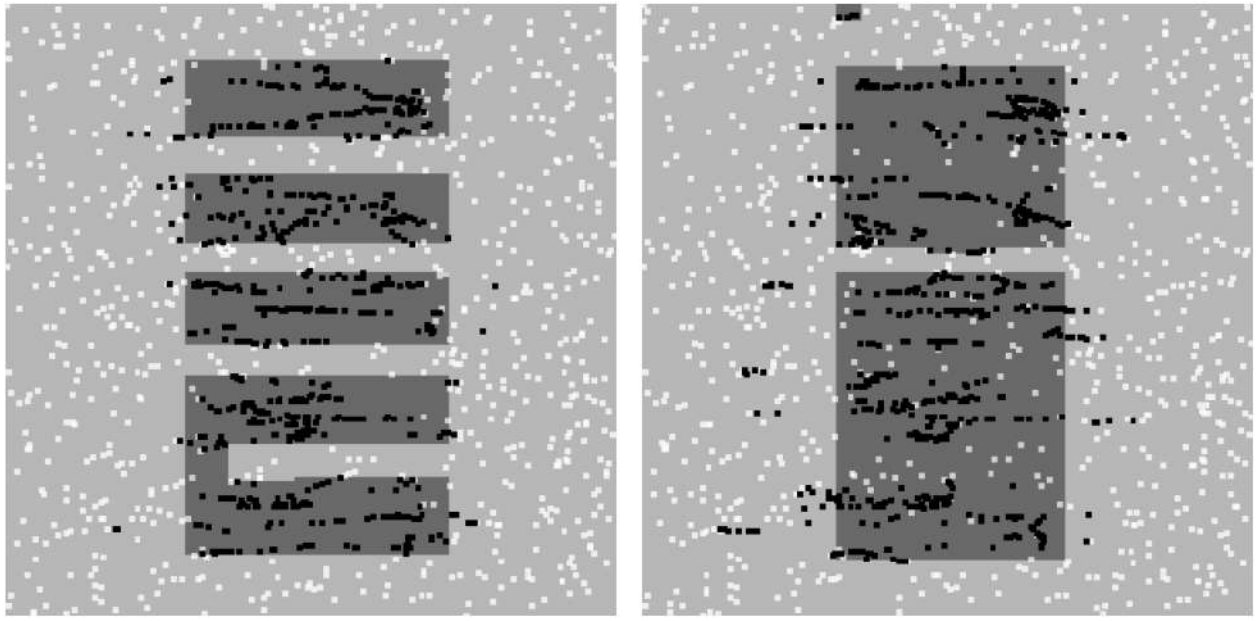


(b) 100 % of training instances



Noisy data

- Since the positive class have fewer examples to begin with, it will take fewer “noisy” examples to impact the learned subconcept.



(a) Original problem and decision functions

(b) Noisy instances and new undesirable decision functions

Noisy data (2)

- Classifiers are more sensitive to noise than imbalance.
- As imbalance increases in severity, it plays a larger role in the performance of classifiers and sampling techniques.
- Most robust classifiers tested over imbalanced and noisy data are bayesian and SVMs, better on average than rule induction algorithms or instance based learning.
- Simple undersampling techniques performed the best overall at all levels of noise and imbalance.

C. Seiffert, T. M. Khoshgoftaar, J. V. Hulse, A. Folleco, **An empirical study of the classification performance of learners on imbalanced and noisy software quality data**, *Information Sciences* 259 (2014) 571–595

Noisy and Borderline examples: SMOTE+IPF Solution

- SMOTE algorithm:
  - Balances the class distribution
  - Helps to fill in the interior of subparts of the minority class
- IPF filter:
  - Removes the noisy examples originally present in the dataset and also those created by SMOTE.
  - Cleans up the boundaries of the classes, making them more regular.

Table 6
AUC results obtained by C4.5 on real-world datasets with noisy and borderline examples.

Dataset	None	SMOTE	SMOTE-ENN	SMOTE-TL	SL-SMOTE	B1-SMOTE	B2-SMOTE	SMOTE–IPF
acl	88.75	86.75	86.75	88.00	85.25	**89.00**	88.00	88.50
breast	61.73	60.56	63.70	62.01	**64.72**	63.31	63.58	64.40
bupa	64.40	66.88	61.46	60.18	66.84	**68.60**	63.61	67.53
cleveland	52.58	54.85	57.22	**64.33**	60.07	54.75	56.66	62.82
ecoli	72.46	82.16	**89.97**	82.33	84.52	79.55	79.37	86.55
haberman	57.57	65.41	64.68	62.03	**67.07**	61.40	60.23	66.76
hepatitis	67.66	71.38	71.90	71.15	68.53	66.39	62.70	**72.25**
newthyroid	90.87	96.35	94.64	94.37	90.95	93.45	**97.18**	96.63
pima	70.12	71.29	71.40	69.48	**73.97**	70.94	73.77	73.58

The best case for each dataset is highlighted in bold.

J.A. Sáez, J. Luengo, J. Stefanowski, F. Herrera. **SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering**. *Information Sciences* 291 (2015) 184-203.

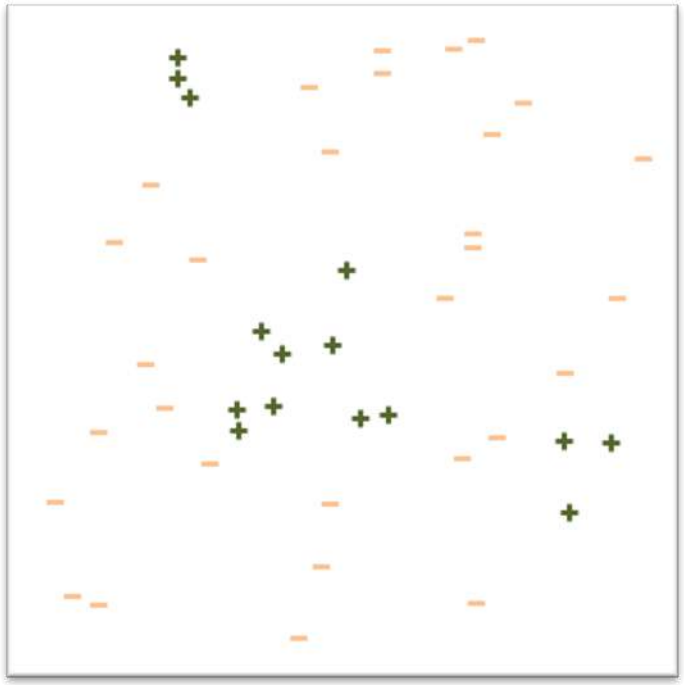
Dataset shift

- Training and test data partitions follow different distributions.
- Particularly sensitive in imbalanced domains due to the minority class examples.
- Potential approaches:
  - **Intrinsic dataset shift**: develop techniques to discover and measure the presence of dataset shift focusing on minority class
  - **Induced dataset shift**: a suitable validation technique needs to be developed to avoid introducing dataset shift issues artificially

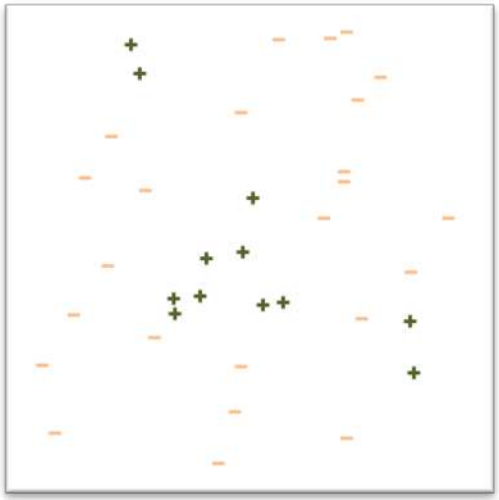
J. G. Moreno-Torres, T. Raeder, R. Aláiz-Rodríguez, N. V. Chawla, F. Herrera, **A unifying view on dataset shift in classification,** *Pattern Recognition* 45 (1) (2012) 521–530

Dataset Shift: Training and Test with the same data distribution

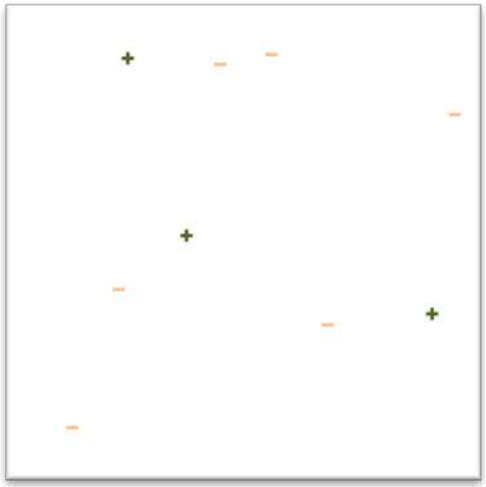
Original Data



Training Data

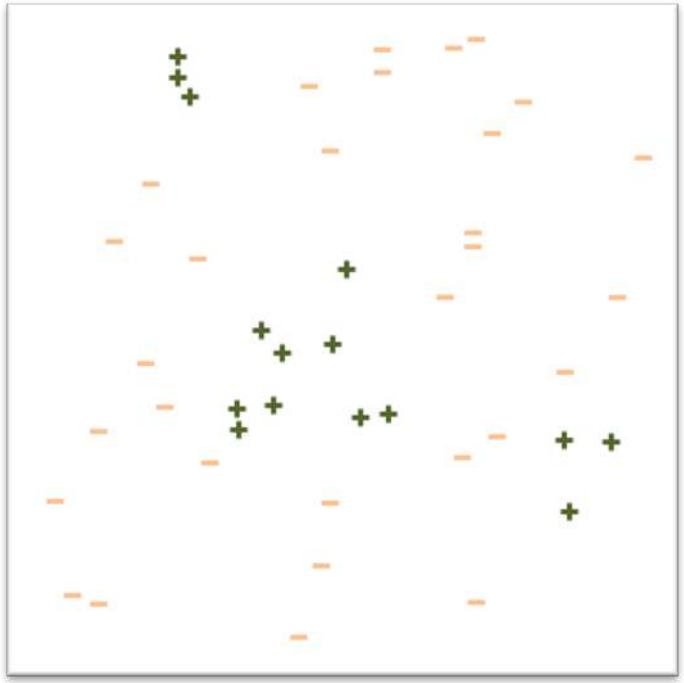


Test Data

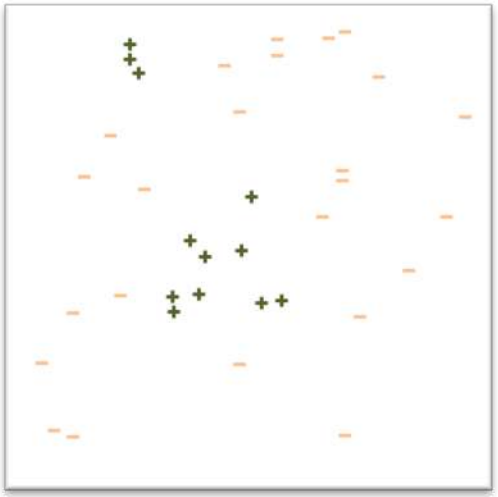


Dataset Shift: Training and Test with different data distribution

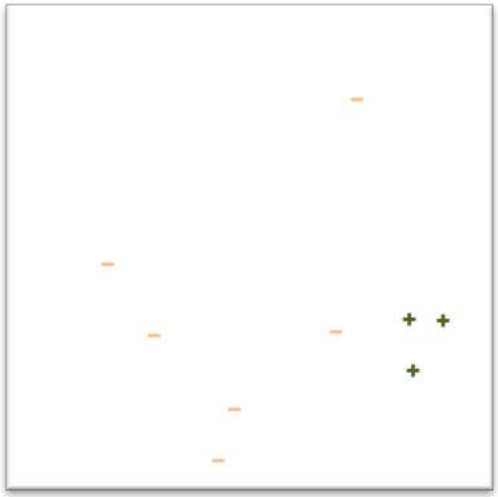
Original Data



Training Data



Test Data



Dataset Shift: DOB Partitioning Method

- Misclassifications for the positive class hinder average precision.
- Avoid those errors due to a “random clustering” of the classes, i.e. generating outliers.
- Keeping data distribution as similar as possible between train and test folds while maximizing diversity.

Algorithm 1 DOB-SCV Partitioning Method

for each class $c_j \in C$ **do**
  while count$(c_j) > 0$ **do**
    $e_0 \leftarrow$ randomly select an example of class c_j from D
    $e_i \leftarrow i$th closest example to e_0 of class c_j from D $(i = 1, \ldots, k-1)$
    $F_i \leftarrow F_i \bigcup e_i (i = 0, \ldots, k-1)$
    $D \leftarrow D \setminus e_i (i = 0, \ldots, k-1)$
  end while
end for

V. López, A. Fernandez, F. Herrera, **On the Importance of the Validation Technique for Classification with Imbalanced Datasets: Addressing Covariate Shift when Data is Skewed**. *Information Sciences* 257 (2014) 1-13

Outline

- Introduction: Definition, properties and difficulty
- Evaluation metrics
- Data Intrinsic Characteristics
- Addressing imbalanced datasets
- Software tools for classification with imbalanced data
- Final Comments
- Surveys for a deeper study

Strategies for imbalanced datasets

Data level:

- Instance preprocessing
- Re-sampling.

Algorithm level:

- change the behaviour of the algorithm itself.

Cost-Sensitive learning:

- Varying costs of different misclassification types.
- Mix of strategies

Ensemble methods:

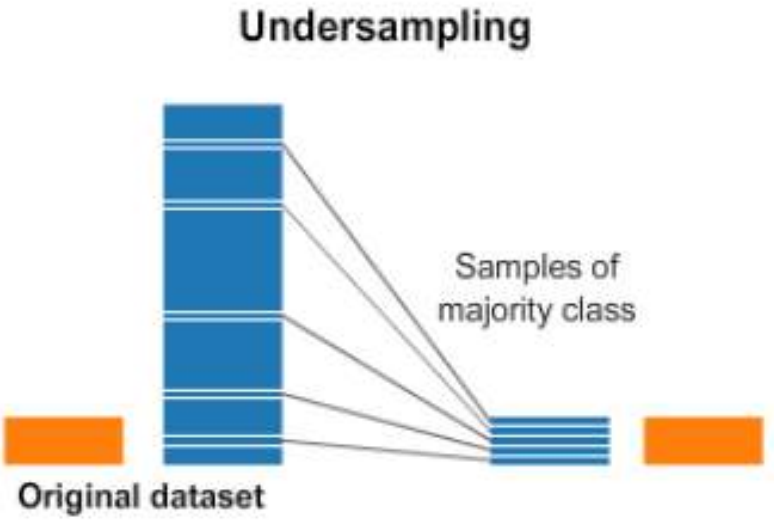
- Multiple classifier systems embedding any of the former approaches

PREPROCESSING

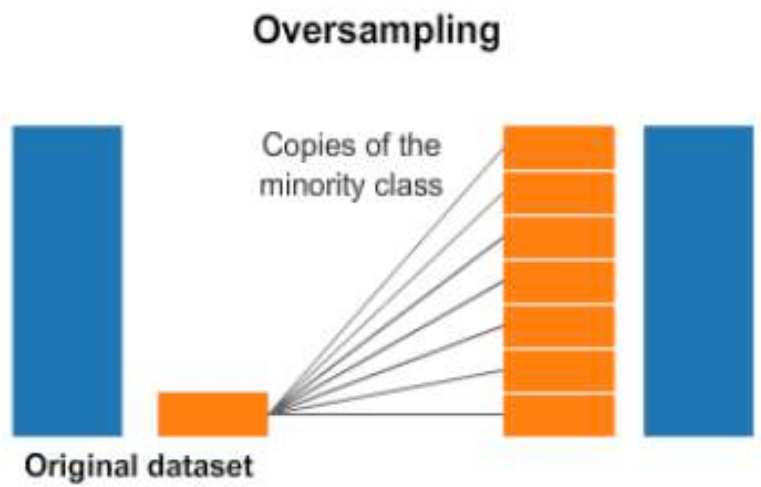
Balancing the training set prior learning

Preprocessing algorithms

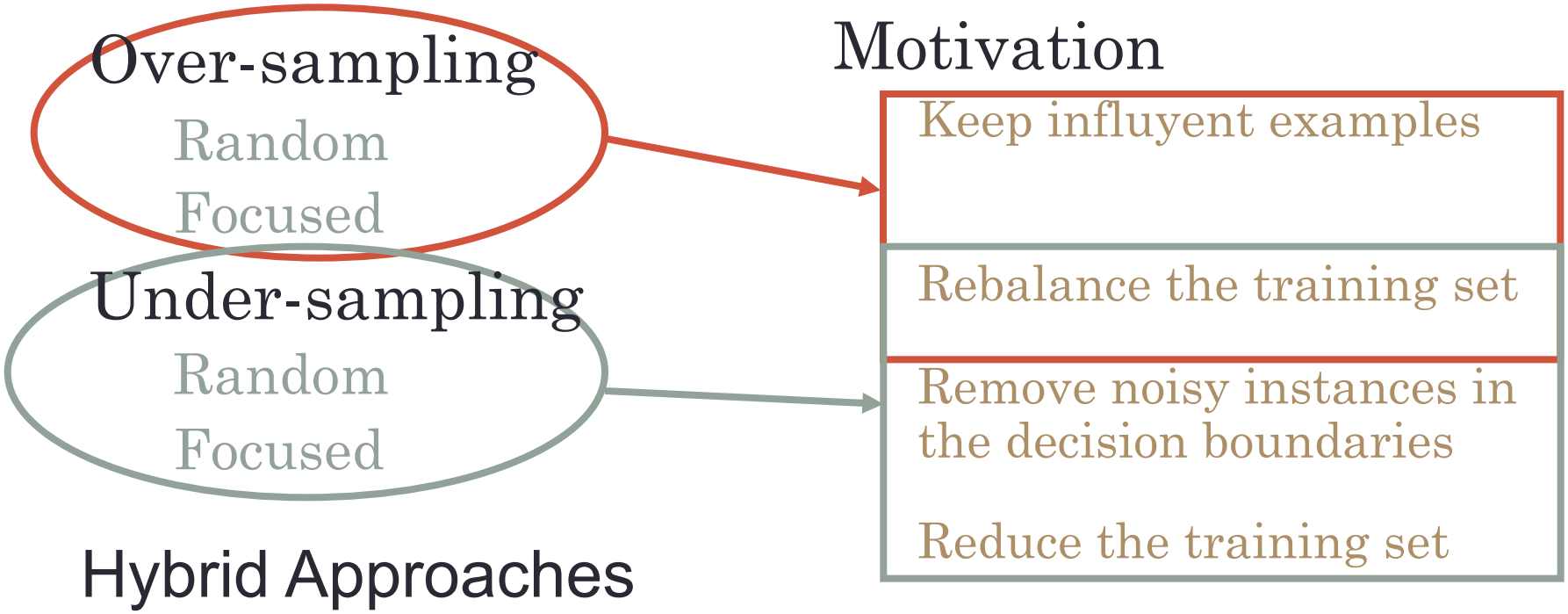
Removing majority class samples



Adding minority class samples



Preprocessing algorithms (2)

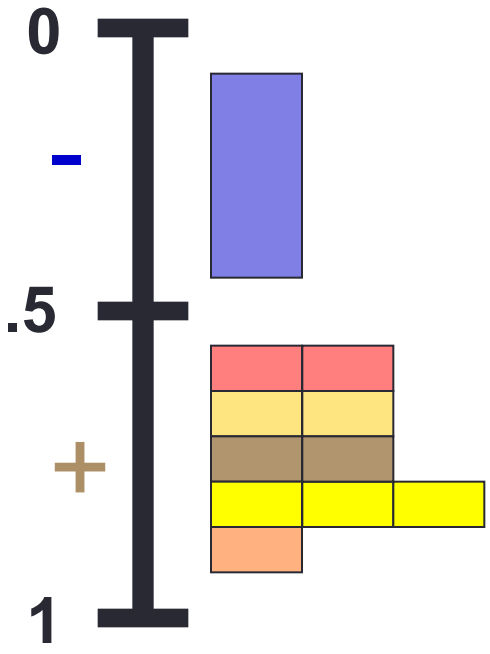


Resampling the original data sets: US vs. OS

Over Sampling
- Random
- Focused

Under Sampling
- Random
- Focused

Cost Modifying



examples of -

examples of +

Resampling the original data sets: US vs. OS

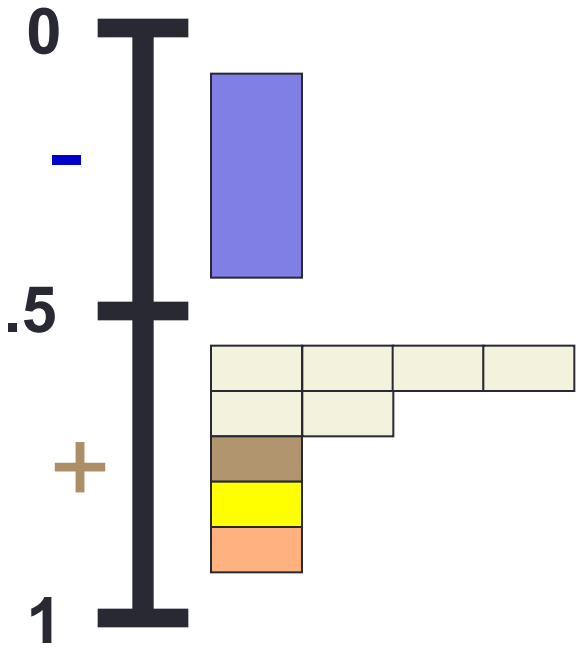
Over Sampling

- Random
- Focused

Under Sampling

- Random
- Focused

Cost Modifying



examples of -

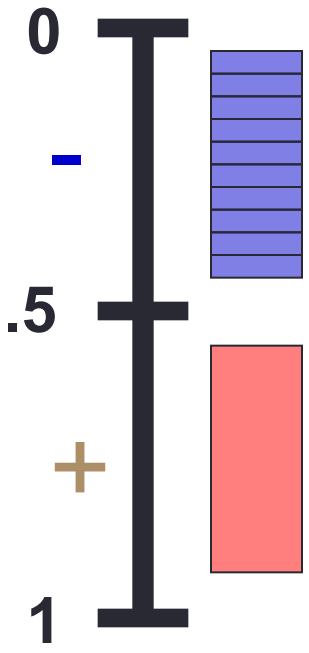
examples of +

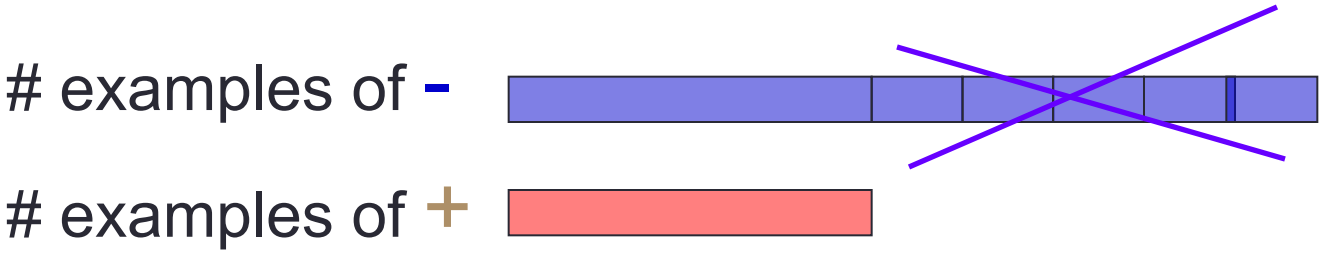
Resampling the original data sets: US vs. OS

Over Sampling
- Random
- Focused

Under Sampling
- Random
- Focused

Cost Modifying





Resampling the original data sets: US vs. OS

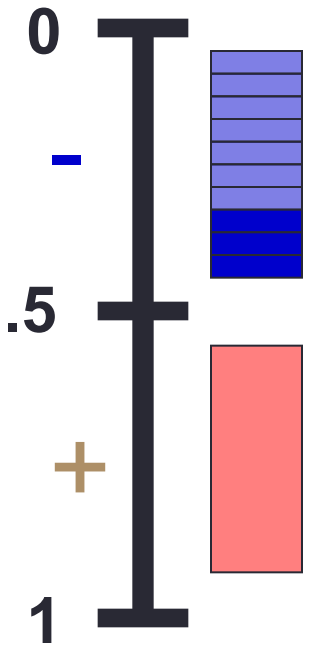
Over Sampling

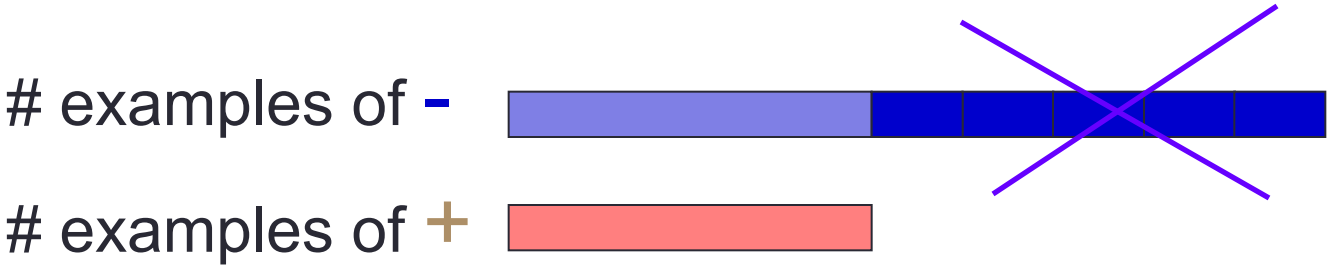
- Random
- Focused

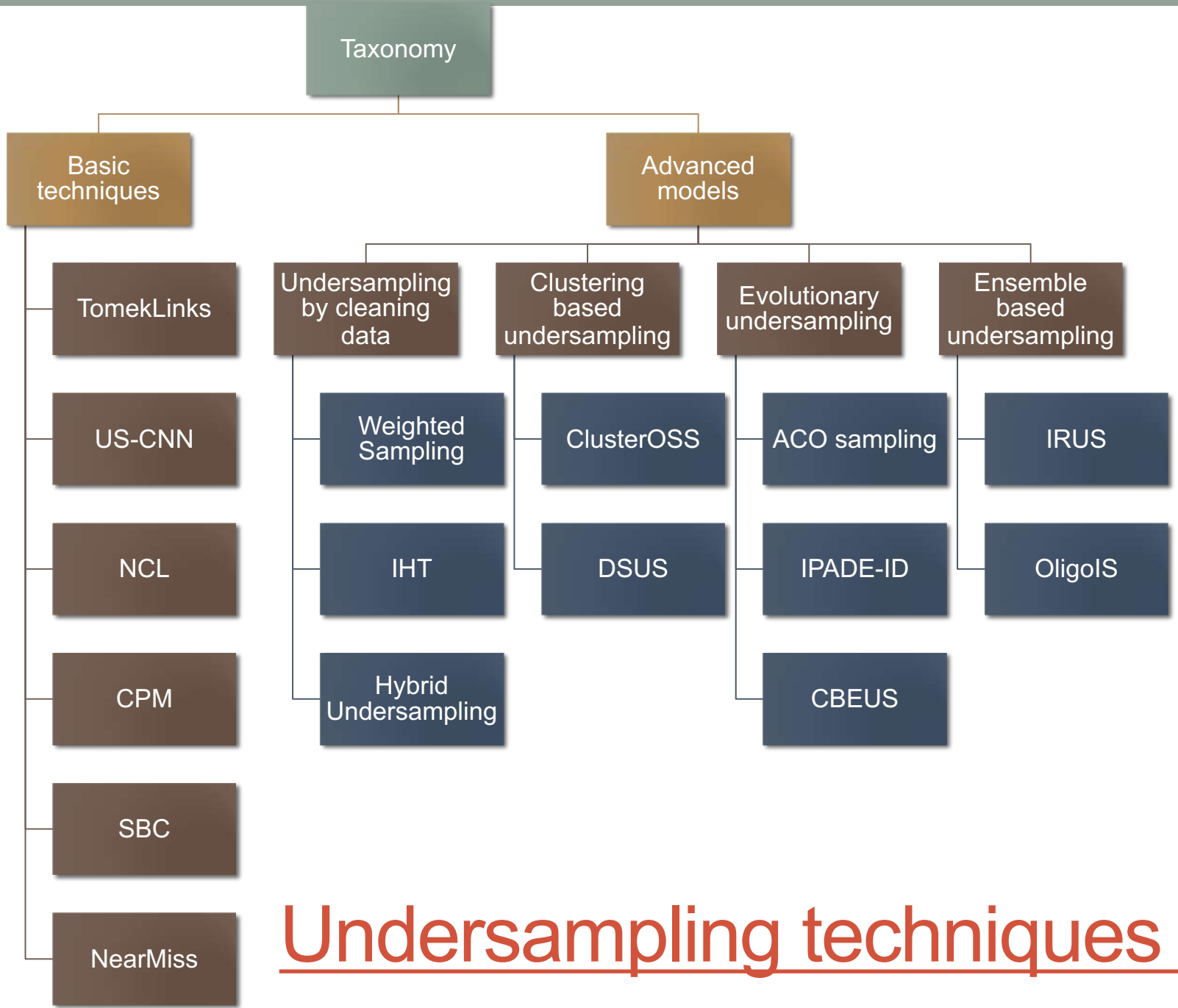
Under Sampling

- Random
- Focused

Cost Modifying



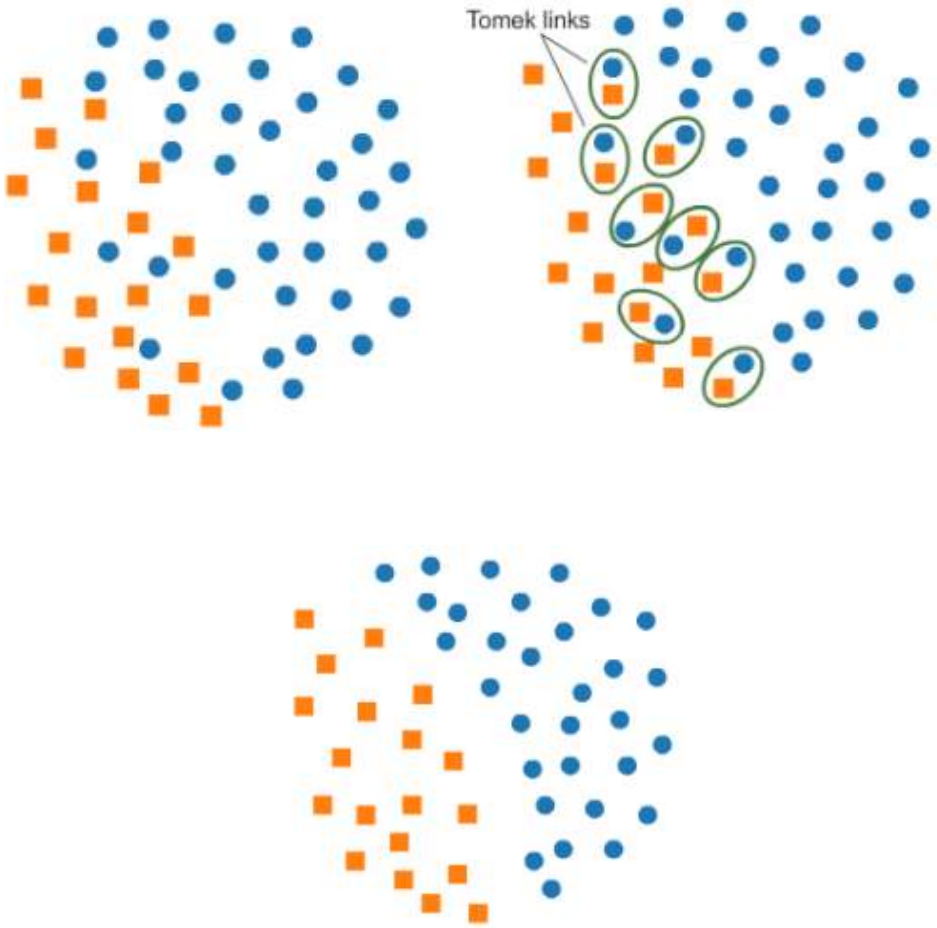




Undersampling techniques

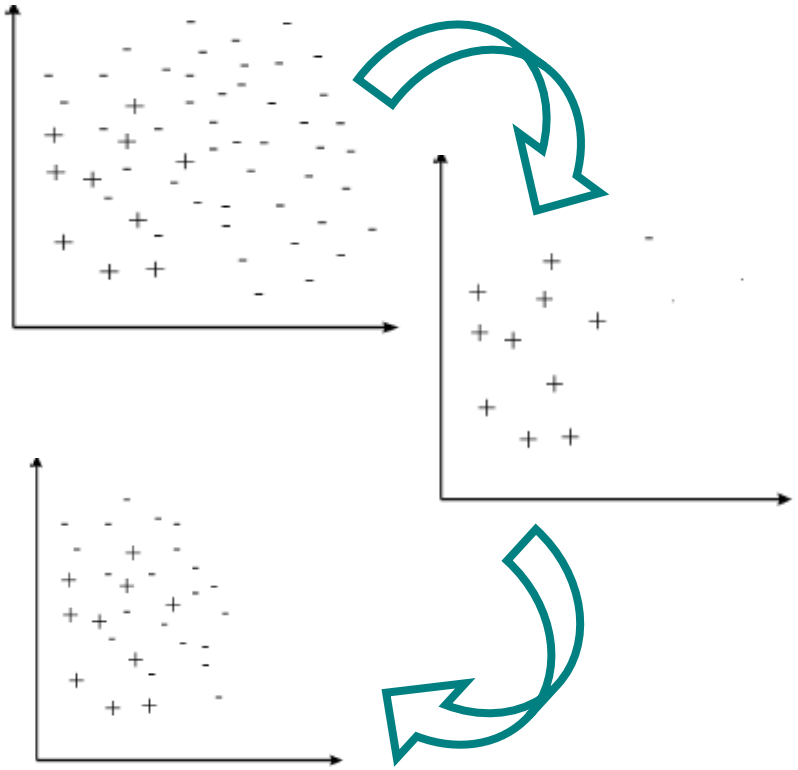
UnderSampling: Tomek Links (Cleaning)

- Remove both noise and borderline examples (majority class)
  - E_i, E_j belong to different classes,
  - d (E_i, E_j): distance
  - A (E_i, E_j) pair is called a Tomek link if there is no example E_l, such that
    - $d(E_i, E_l) < d(E_i, E_j)$ or
    - $d(E_j, E_l) < d(E_i, E_j)$.



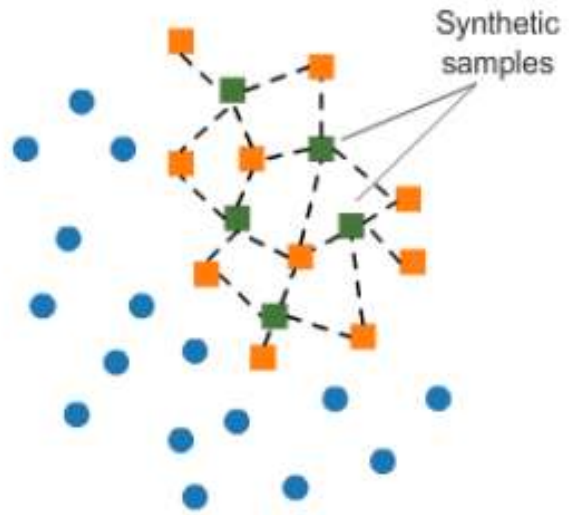
UnderSampling: CNN (Cleaning)

- Remove noise and borderline
  - Let E be the original training set
  - Let E' contains all positive examples from S and one randomly selected negative example
  - Classify E with the 1-NN rule using the examples in E'
  - Move all misclassified example from E to E'



Preprocessing algorithms: SMOTE

- Oversampling: Simply replicating examples
- Synthetic Minority Over-sampling Technique (**SMOTE**):
  - **Generation** of new minority class examples
  - Interpolation among several minority class instances that lie together



For each minority sample

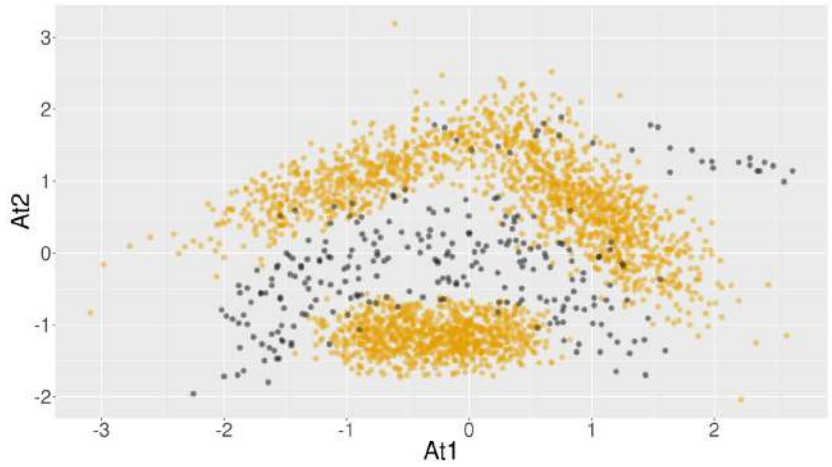
- Find its k-nearest minority neighbours
- Randomly select j neighbours
- Randomly generate synthetic samples along the lines joining the minority sample selected and its j neighbours

(j depends on the amount of oversampling desired)

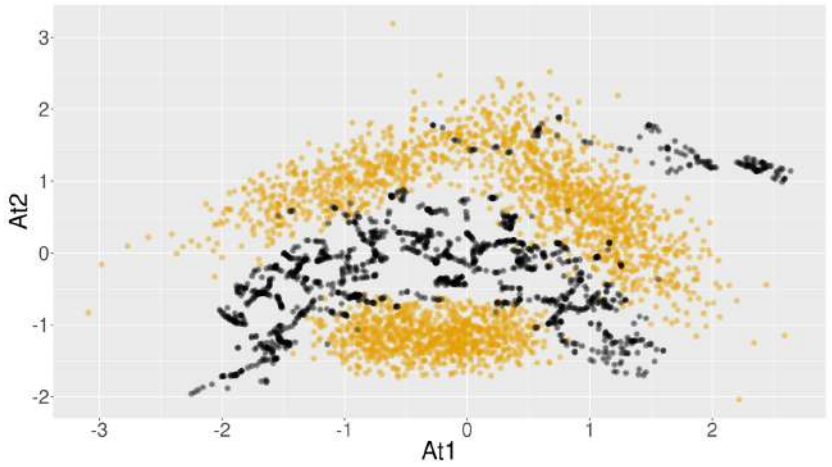
N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer. **SMOTE: synthetic minority over-sampling technique**. *Journal of Artificial Intelligence Research* 16 (2002) 321-357

Solutions: Preprocessing

Original



SMOTE



SMOTE vs Random Oversampling

- **Random Oversampling** (with replacement) of the minority class:
  - Making the decision region for the minority class very specific.
  - In a decision tree, it would cause a new split and lead to overfitting.
- **SMOTE's** informed oversampling
  - It generalizes the decision region for the minority class.
  - Larger and less specific regions are learned.
  - Paying attention to minority class samples without causing overfitting.

Overgeneralization

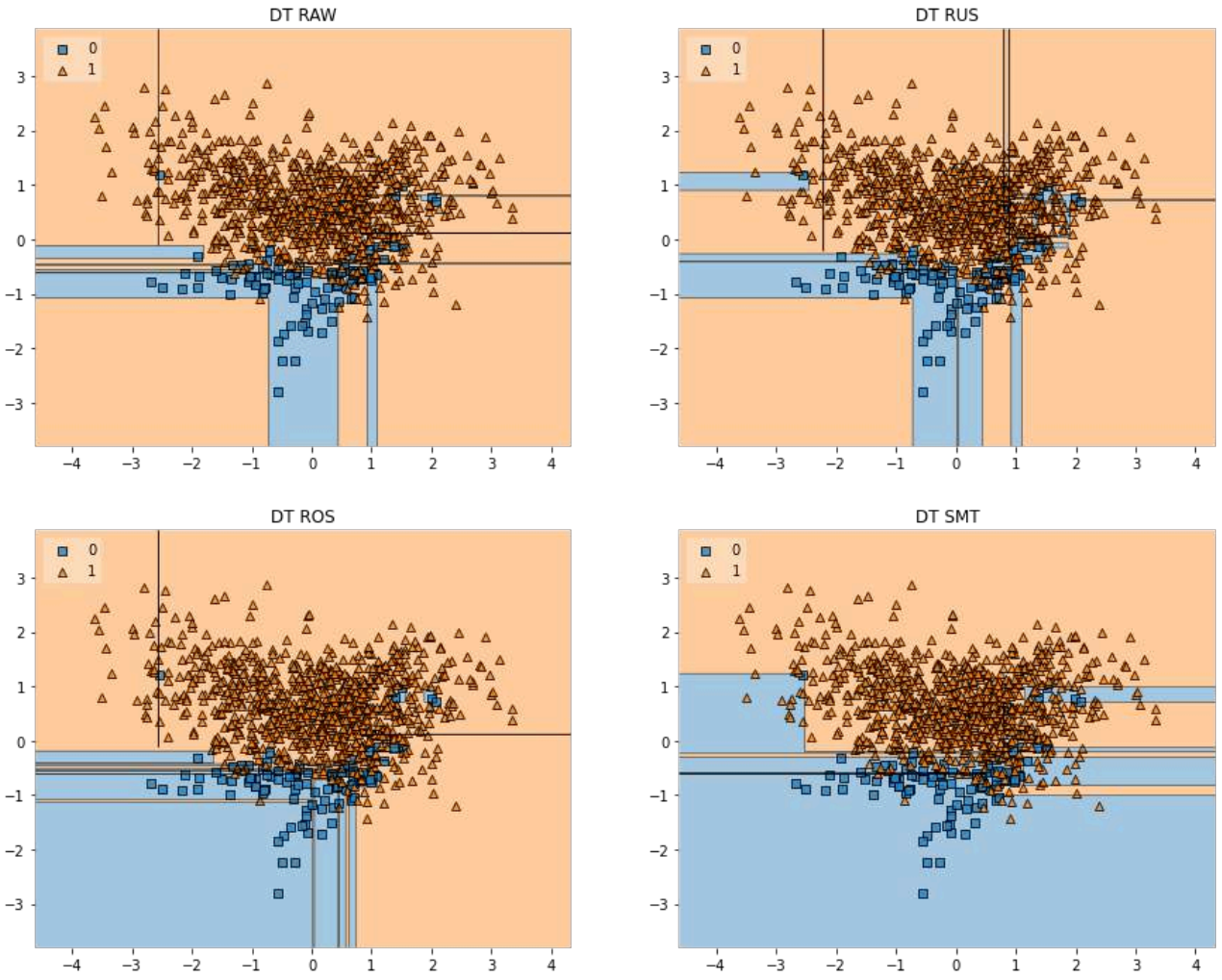
- SMOTE's is inherently dangerous since it **blindly generalizes** the minority area disregard majority class.
- This strategy is particularly problematic in the case of highly skewed class distributions:
  - Minority class is very sparse w.r.t. the majority class.
  - Results in a greater chance of class mixture.

Lack of Flexibility

- The number of synthetic samples generated by SMOTE is fixed in advance,
- This does not allow for any flexibility in the re-balancing rate.

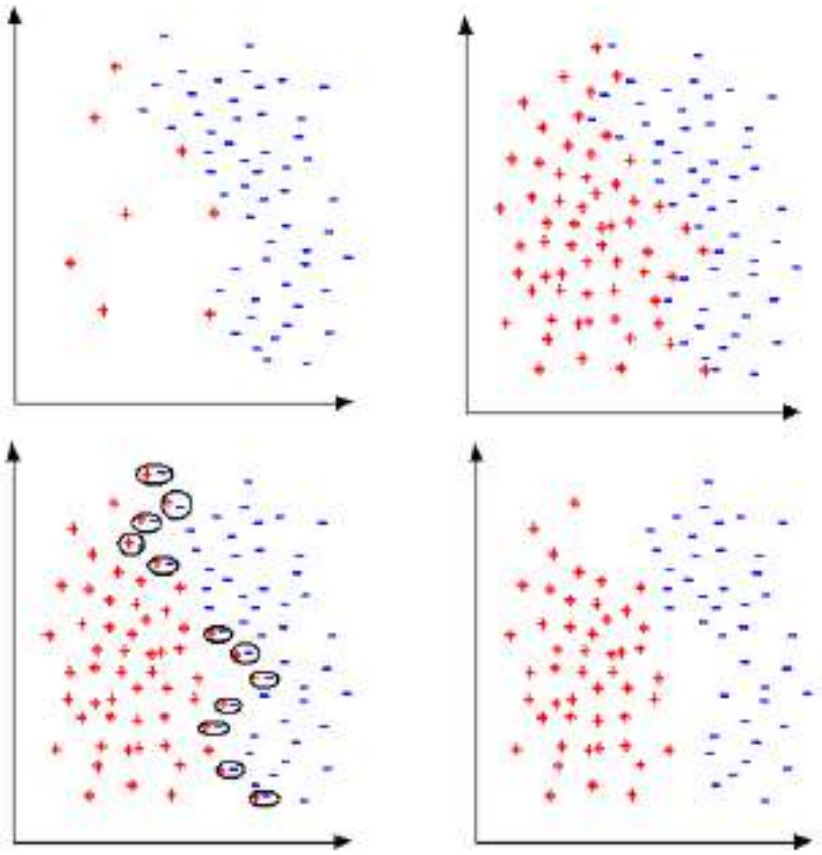
SMOTE shortcomings

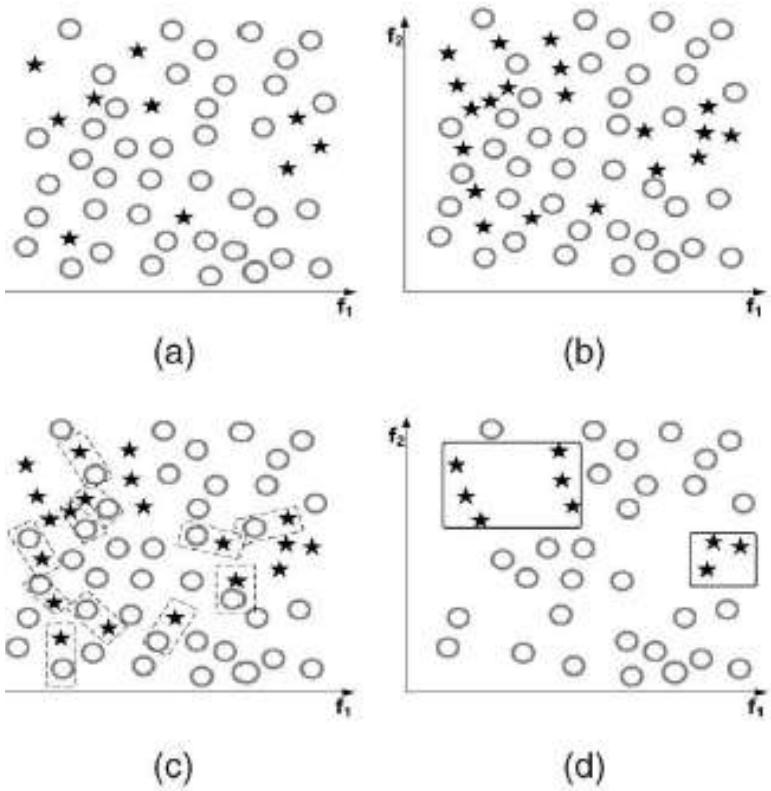
The 3 state-of-the-art resampling (DT)



SMOTE Hybridization: SMOTE + Tomek Links. Data Cleaning in original and synthetic samples

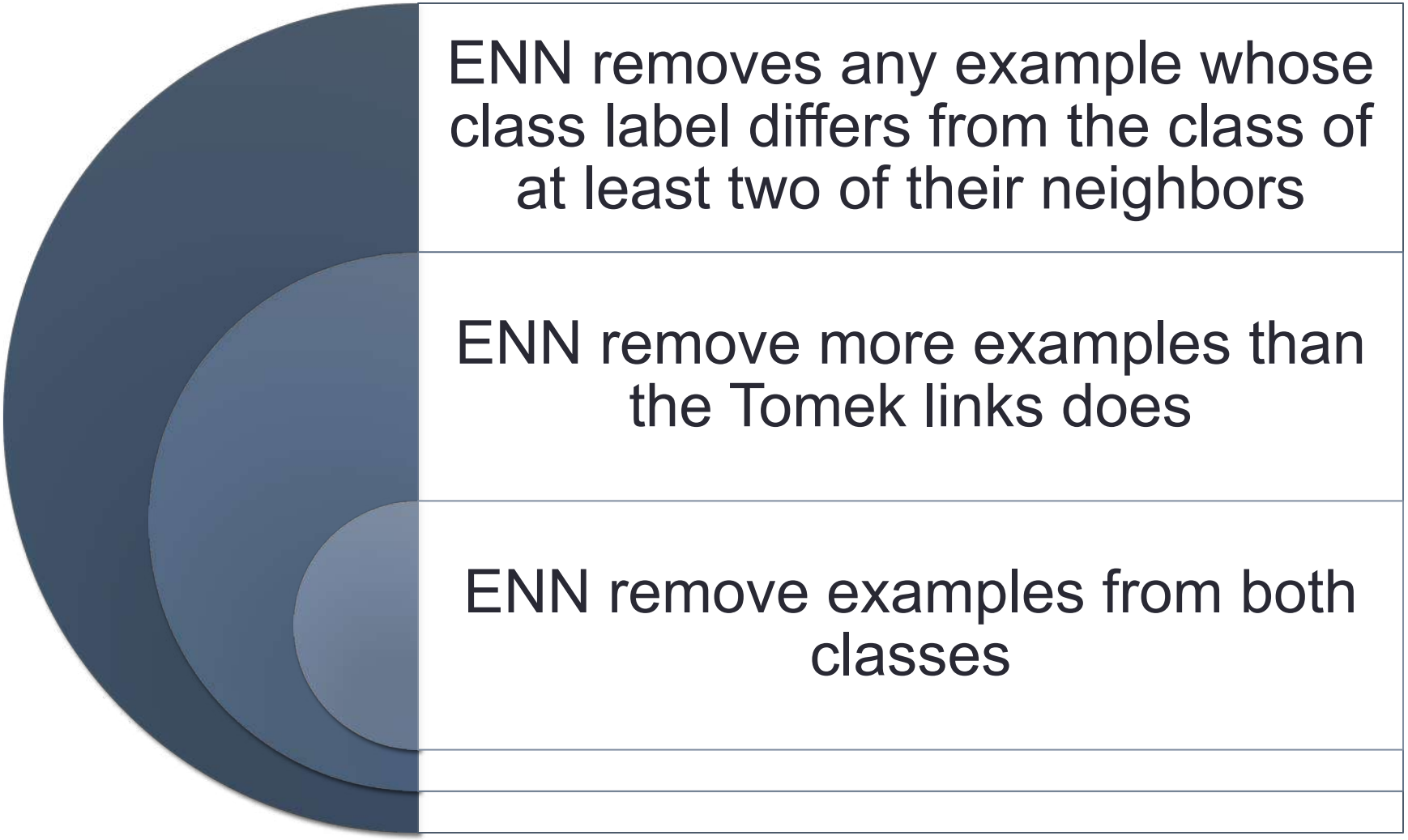
Figure: SMOTE+TomekLink



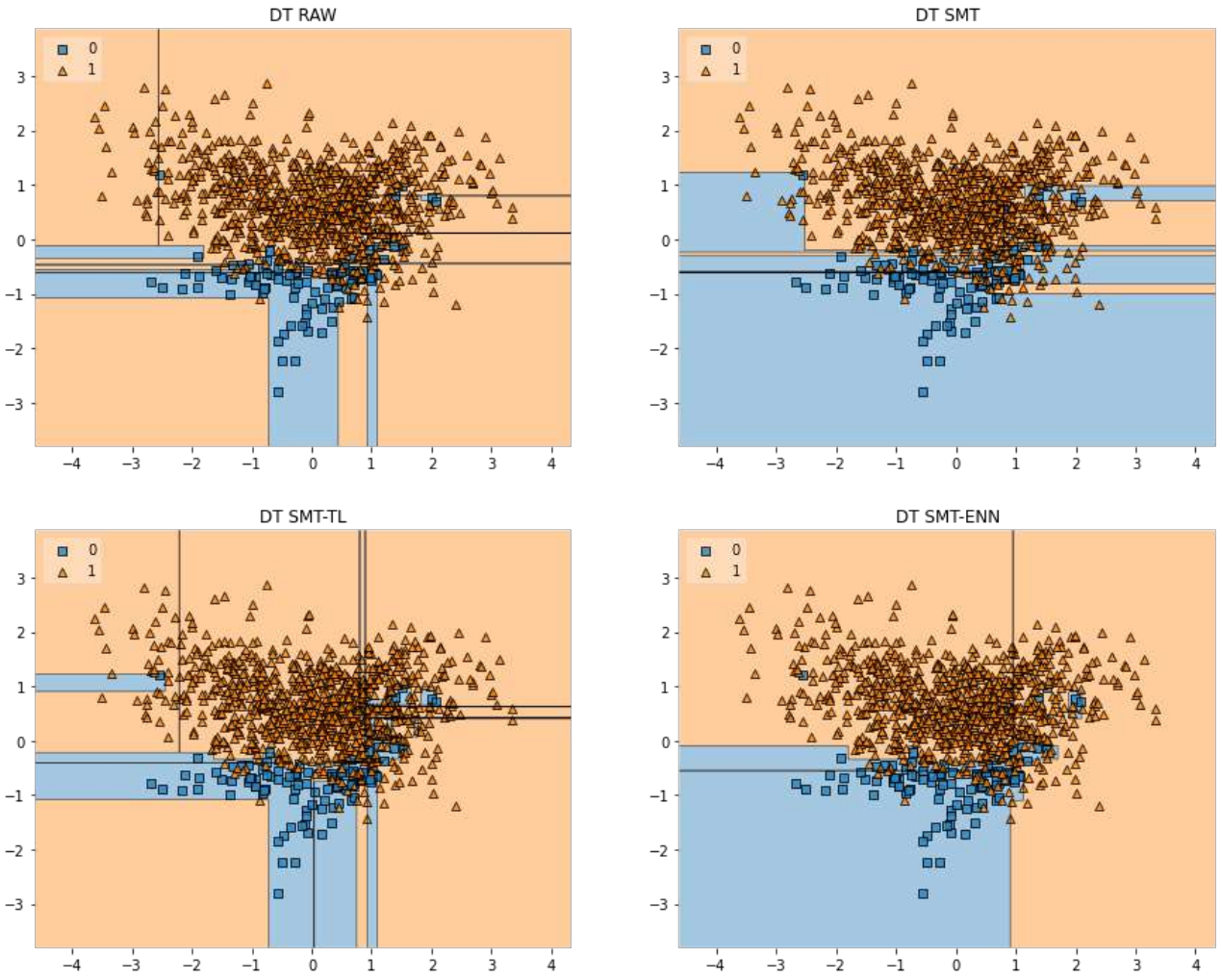


ure 17: (a) Original data-set distribution. (b) Post-SMOTE da
(c) The identified Tomek Links. (d) The data-set after removi
nek links

SMOTE Hybridization: SMOTE + ENN



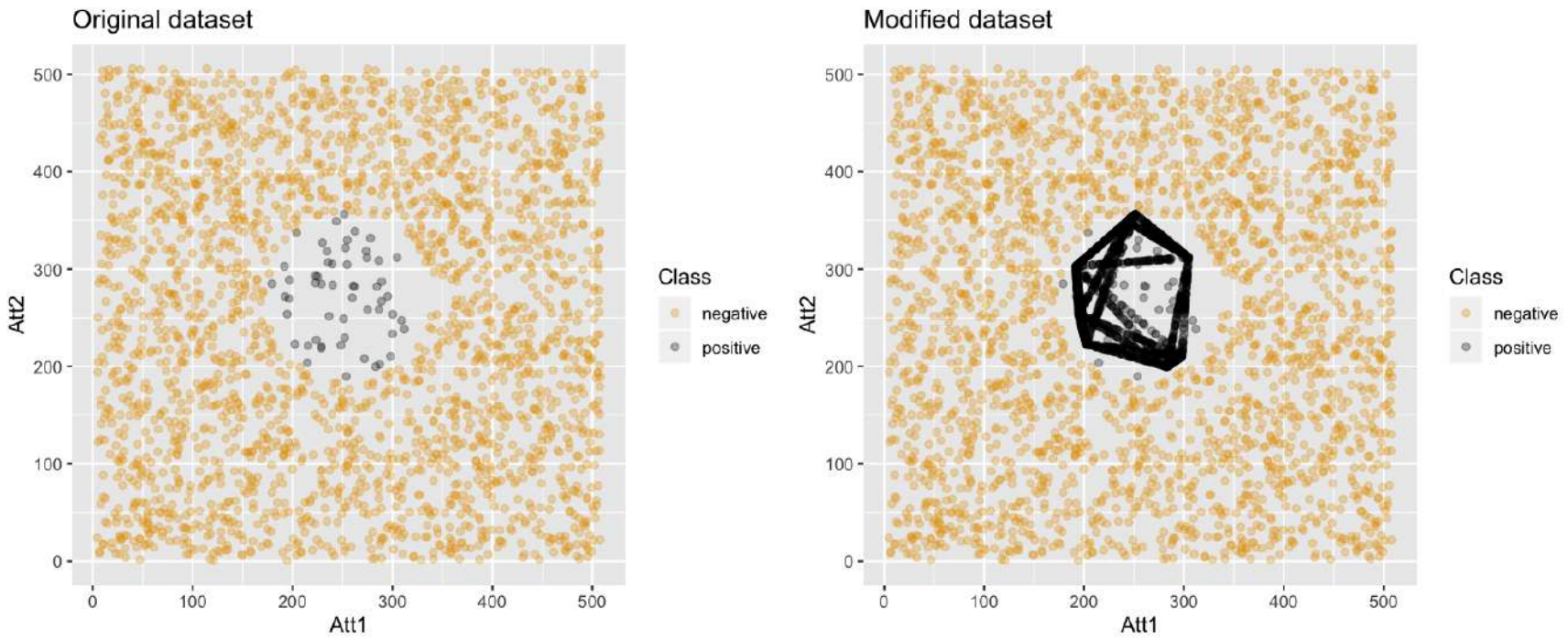
Using different SMOTE Hybridization



Other SMOTE Hybridizations

Safe_Level_SMOTE:	• C. Bunkhumpornpat, K. Sinapiromsaran, C. Lursinsap. Safe-level-SMOTE: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD-09). LNAI 5476, Springer-Verlag 2005, Bangkok (Thailand, 2009) 475-482
Borderline_SMOTE:	• H. Han, W.Y. Wang, B.H. Mao. Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. International Conference on Intelligent Computing (ICIC'05). Lecture Notes in Computer Science 3644, Springer-Verlag 2005, Hefei (China, 2005) 878-887
SMOTE-RSB:	• E. Ramentol, Y. Caballero, R. Bello, F. Herrera, SMOTE-RSB*: A Hybrid Preprocessing Approach based on Oversampling and Undersampling for High Imbalanced Data-Sets using SMOTE and Rough Sets Theory. *Knowledge and Information Systems 33:2 (2012) 245-265.*
SMOTE-IPF:	• Sáez, J.A., Luengo, J., Stefanowski, J., Herrera, F.: SMOTE-IPF: addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering. *Information Sciences* 291, 184–203 (2015)

SMOTE Hybridization: SMOTE-Borderline



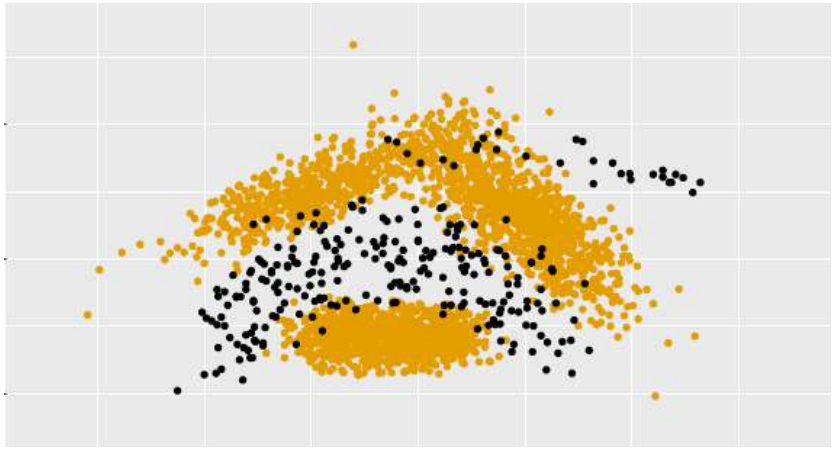
H. Han, W. Wang, B. Mao. **Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning**. *In: ICIC 2005*. LNCS 3644 (2005) 878-887.

Other Oversampling algorithms

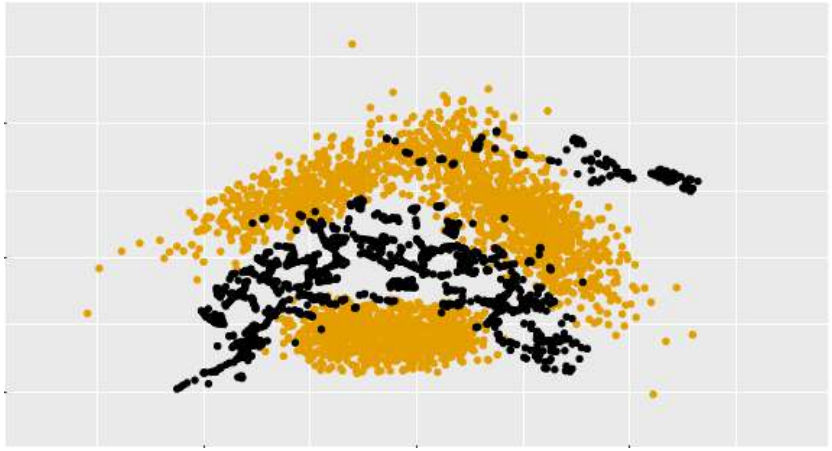
Barua et al. (2014) IEEE Transactions on Knowledge and Data Engineering	**MWMOTE**—Majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning.
Huaxiang Zhang, Mingfang Li (2014). Information Fusion	**RWO-Sampling**: A random walk over-sampling approach to imbalanced data classification.
M. Gao, X. Hong, S. Chen, C.J. Harris, E. Khalaf (2014) Neurocomputing	**PDFOS**: PDF estimation based over-sampling for imbalanced two-class problems
Lida Abdi and Sattar Hashemi (2016). IEEE Transactions on Knowledge and Data Engineering	**MDO:** To Combat Multi-Class Imbalanced Problems by Means of Over-Sampling Techniques.
Chumphol Bunkhumpornpat, Krung Sinapiromsaran (2016). Knowledge and Information Systems	**DBMUTE**: density-based majority under-sampling technique.

Comparison of approaches

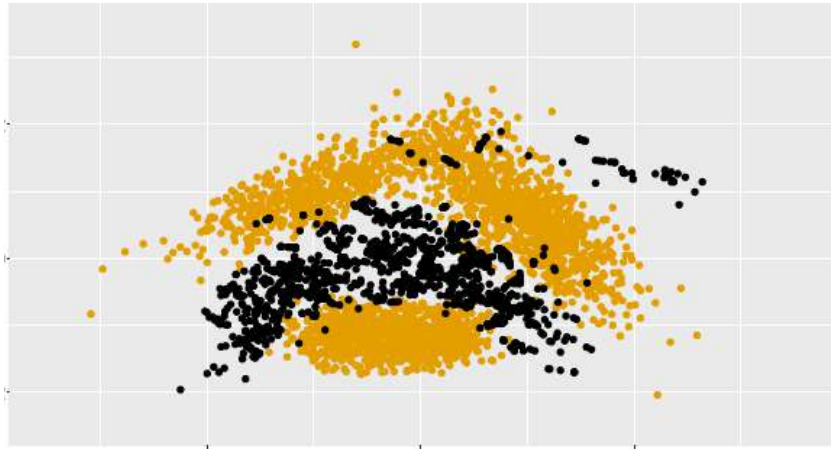
Imbalanced banana data



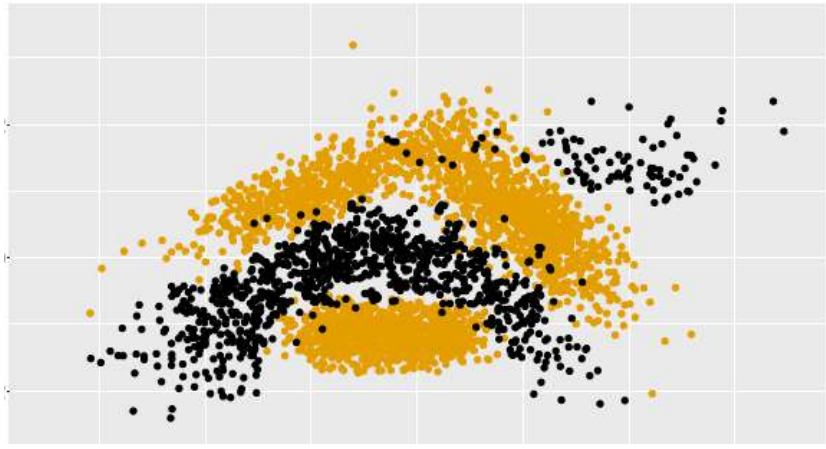
SMOTE applied to banana



MWMOTE applied to banana



PDFOS applied to banana



Some few extensions of SMOTE

Ref.	Algorithm name	Initial selection	Integration US	Type of Interpolation	Dimensionality change	Adaptive generation	Relabeling	Filtering
(Batista et al., 2004)	SMOTE+TomekLinks							✓
(Batista et al., 2004)	SMOTE+ENN							✓
(Han et al., 2005)	Borderline-SMOTE	✓		Range restricted				
(Cohen, Hilario, Sax, Hugonnet, & Geissbühler, 2006)	AHC		✓	Clustering				
(Wang, Xu, Wang, & Zhang, 2006)	LLE-SMOTE				LLE			
(de la Calleja & Fuentes, 2007)	Distance-SMOTE			Multiple				
(de la Calleja et al., 2008)	SMMO	✓		Without-Gaussian				
(Gazzah & Amara, 2008)	Polynom-Fit-OS			Topologies				
(He et al., 2008)	ADASYN					✓		
(Stefanowski & Wilk, 2008)	< *no name* >	✓		Without-Copy			✓	
(Tang & Chen, 2008)	ADOMS			With PCA	PCA			
(Bunkhumpornpat et al., 2009)	Safe-Level-SMOTE	✓		Range restricted		✓		
(Gu et al., 2009)	Isomap-Hybrid		✓		MDS			
(Liang, Hu, Ma, & He, 2009)	MSMOTE	✓						
(Chen, Cai, Chen, & Gu, 2010a)	DE-Oversampling		✓	DE operators				
(Chen, Guo, & Chen, 2010c)	CE-SMOTE	✓						
(Kang & Won, 2010)	Edge-Det-SMOTE	✓						
(Barua et al., 2011)	CBSO			Clustering		✓		
(Cao & Wang, 2011)	SMOBD	✓				✓		
(Cateni et al., 2011)	SUNDO	✓	✓	Gaussian+Cov.				
(Deepa & Punithavalli, 2011)	E-SMOTE				FS with GA			
(Dong & Wang, 2011)	Random-SMOTE			Multiple				
(Fan, Tang, & Weise, 2011)	MSYN					✓		✓
(Fernández-Navarro, Hervás-Martínez, & Gutiérrez, 2011)	DSRBF					✓		
(Maciejewski & Stefanowski, 2011)	LN-SMOTE	✓		Range restricted				
(Zhang & Wang, 2011b)	Distribution-SMOTE	✓						
(Zhang & Wang, 2011a)	NDO-Sampling			Without-Gaussian				
(Bunkhumpornpat et al., 2012)	DBSMOTE	✓		Graph based				
(Farquad & Bose, 2012)	SVM-Balance						✓	
(Puntumapon & Waiyamai, 2012)	TRIM-SMOTE					✓		✓
(Ramentol et al., 2012)	SMOTE-RSB*							✓
(Wang et al., 2012)	ASMOBD	✓		Smoothing		✓		
(Barua, Islam, & Murase, 2013)	ProWSyn			Clustering		✓		
(Bunkhumpornpat & Subpaiboonkit, 2013)	SL-Graph-SMOTE	✓		Range restricted		✓		
(Hu & Li, 2013)	NRSBoundary-SMOTE							✓
(Li, Zou, Wang, & Xia, 2013b)	ISMOTE		✓					
(Nakamura et al., 2013)	LVQ-SMOTE	✓(LVQ)			FS			
(Pérez-Ortiz, Gutiérrez, & Hervás-Martínez, 2013)	BKS	✓		Range restricted	Kernels			

A. Fernández, S. García, F. Herrera, N.V. Chawla. **SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary** *Journal of Artificial Intelligence Research* 61 (2018) 863-905

Some few extensions of SMOTE

(Sánchez, Morales, & Gonzalez, 2013)	SOI-CJ	✓		Clustering+Jittering				
(Wang et al., 2013a)	TSMOTE+AB			Range restricted	Bagging	✓		
(Wang, Yao, Zhou, Leng, & Chen, 2013b)	MST-SMOTE			Graph based				
(Zhou, Yang, Guo, & Hu, 2013)	Assembled-SMOTE	✓						
(Menardi & Torelli, 2014)	ROSE	✓	✓	Without-Smoothing	Kernels			
(Barua, Islam, Yao, & Murase, 2014)	MWMOTE	✓		Clustering				
(Gao et al., 2014b)	PDFOS			PDF+Gaussian				
(Koto, 2014)	SMOTE-Out			Range restricted				
(Koto, 2014)	SMOTE-Cosine	✓						
(Koto, 2014)	Selected-SMOTE				FS			
(Li, Zhang, Lu, & Fang, 2014)	SDSMOTE	✓				✓		
(López et al., 2014)	IPADE-ID		✓			✓		✓
(Mahmoudi, Moradi, Ahklaghian, & Moradi, 2014)	DSMOTE	✓						
(Rong et al., 2014)	SSO			Gaussian+Q-union				
(Sandhan & Choi, 2014)	G-SMOTE			Gaussian+Non-linear				
(Xu, Le, & Tian, 2014)	NT-SMOTE			Multiple				
(Zhang & Li, 2014)	RWO-Sampling			Without-Gaussian				
(Lee, Kim, & Lee, 2015)	*< no name >*	✓						
(Almogahed & Kakadiaris, 2015)	NEATER							✓
(Alejo et al., 2015)	MSEBPOS					✓		
(Bellinger et al., 2015)	DEAGO			Without	Auto-Encoder			
(Dang et al., 2015)	SPY						✓	
(Das et al., 2015)	wRACOG	✓		Without-Markov				
(Gazzah, Hechkel, & Amara, 2015)	*< no name >*		✓	Topologies	PCA			
(Jiang, Qiu, & Li, 2015)	MCT			Without-Copy				
(Li, Fong, & Zhuang, 2015)	SMOTE-PSO/BAT							
(Mao, Wang, & Wang, 2015)	MinorityDegree-SMOTE		✓					✓
(Mathew et al., 2015)	K-SMOTE				Kernels			
(Pourhabib, Mallick, & Ding, 2015)	ADG	✓		Without-Gaussian	Kernels			
(Sáez et al., 2015)	SMOTE-IPF							✓
(Tang & He, 2015)	KernelADASYN				Kernels	✓		
(Xie et al., 2015)	MOT2LD	✓		Clustering	t-SNE			
(Young et al., 2015)	V-synth	✓		Voronoi				
(Zieba et al., 2015)	RBM-SMOTE					✓		✓
(Abdi & Hashemi, 2016)	MDO	✓		Ellipse	PCA	✓		
(Bellinger et al., 2016)	DAE			Without	PCA+Auto-Encoder			
(Borowska & Stepaniuk, 2016)	VIS-RST	✓					✓	✓
(Gong & Gu, 2016)	DGSMOTE		✓	Clustering				✓
(Jiang et al., 2016)	GASMOTE					✓		
(Nekooeimehr & Lai-Yuen, 2016)	A-SUWO	✓		Clustering				
(Peng, Zhang, Yang, Chen, & Zhou, 2016)	SMOTE-DGC					✓		✓
(Pérez-Ortiz et al., 2016)	OEFS				Kernels			
(Ramentol et al., 2016)	SMOTE-FRST-2T							✓
(Rivera & Xanthopoulos, 2016)	OUPS	✓						
(Torres, Carrasco-Ochoa, & Martínez Trinidad, 2016)	SMOTE-D			Range restricted		✓		
(Yun, Ha, & Lee, 2016)	AND-SMOTE					✓		
(Cervantes et al., 2017)	SMOTE-PSO	✓(SVs)				✓		✓
(Ma & Fan, 2017)	CURE-SMOTE	✓		Clustering				
(Rivera, 2017)	NRAS					✓		✓
(Cao, Liu, Zhang, Zhao, Huang, & Zaïane, 2017b)	MKOS				FS + Kernels			
(Douzas & Bacao, 2017)	SOMO			Clustering	SOM	✓		
(Li, Fong, Wong, & Chu, 201)	AMSCO	✓	✓			✓		✓

A. Fernández, S. García, F. Herrera, N.V. Chawla. **SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary** *Journal of Artificial Intelligence Research* 61 (2018) 863-905

ALGORITHMIC AND COST-SENSITIVE SOLUTIONS

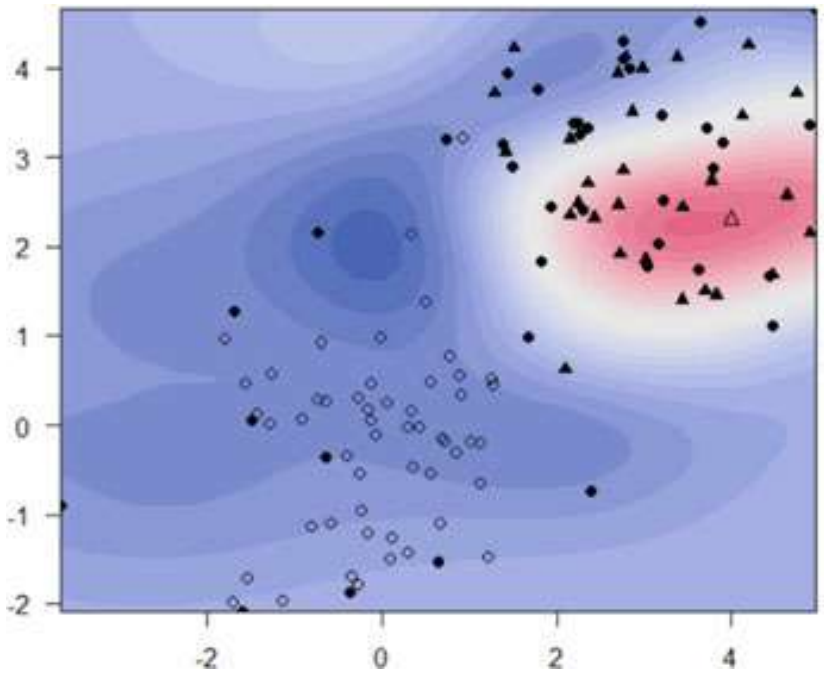
Acting over the raw data

Algorithmic modifications in imbalanced

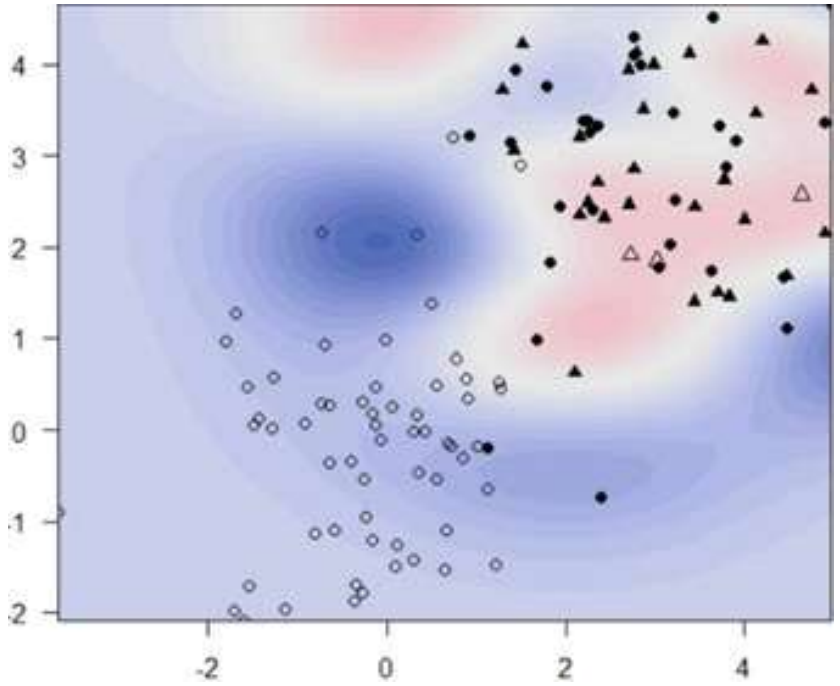
- Concentrate on modifying existing learners to alleviate their bias towards majority class instead on altering the training set
- This requires a good insight into the modified learning algorithm and a precise identification of reasons for its failure in mining skewed distributions
- This reduces their flexibility, but offers higher specialization potential in tuning the method to the problem at hand

Different decision boundaries

SVM standard approach



SVM with instance level sampling



Taxonomy of algorithmic approaches

Support Vector Machines

- Kernel modification
- Weighted approaches
- Active learning

Decision Trees

- Hellinger distance for splitting

K-Nearest Neighbours

- Gravitation based computation
- Weighted prototypes
- Fuzzy OWA K-NN

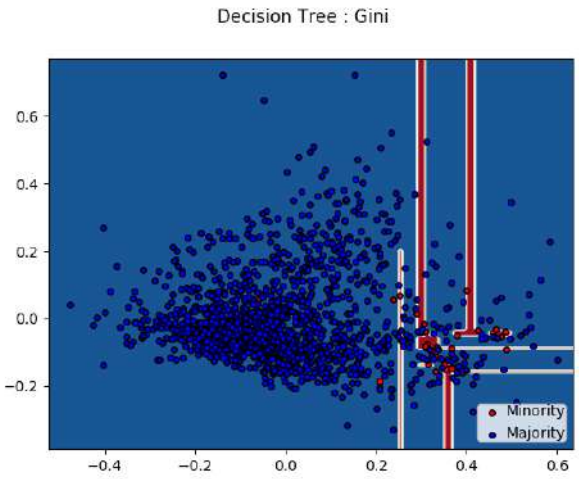
Bayesian Classifiers

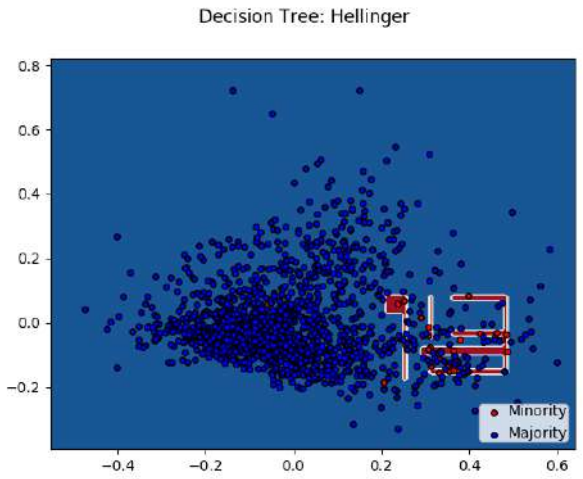
- Locally weighted NB

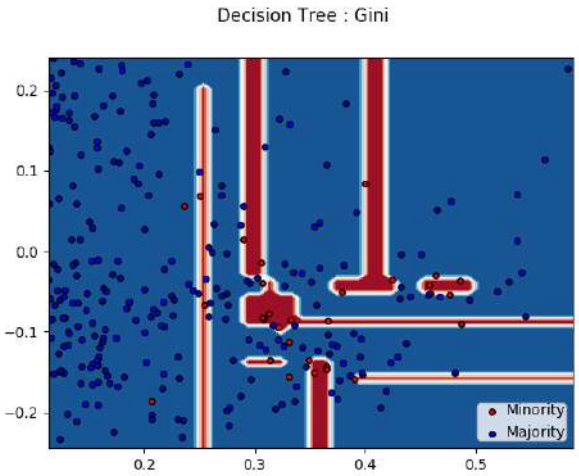
One-Class Classifiers

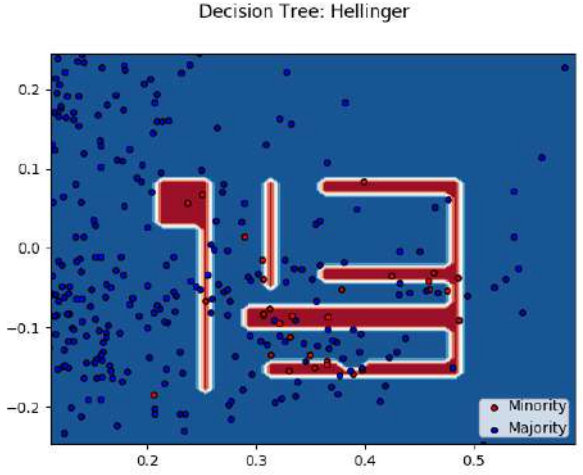
- Training a one-class classifier on the majority class;
- Training a well-tuned one-class classifier on the minority class;
- Training one-class classifiers on both classes and combining their outputs.

Example of Decision Tree modification









Cost-sensitive learning

- Weighting errors made on minority class examples higher than those of the majority class in computing training error:
  - C(+, -) > C(-, +)
  - C(+, +) = C(-, -) = 0
- Needs a cost matrix, which encodes misclassification penalty.
- Consider the cost-matrix throughout the building of the model for achieving the lowest cost.
- However, the cost matrix is often unavalaible

	actual negative	actual positive
predict negative	$C(0,0) = c_{00}$	$C(0,1) = c_{01}$
predict positive	$C(1,0) = c_{10}$	$C(1,1) = c_{11}$

	fraudulent	legitimate
refuse	\$20	$-\$20$
approve	$-x$	$0.02x$

C. Elkan, **The foundations of cost-sensitive learning**, in: *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, 2001, pp. 973–978

How to obtain cost-matrix

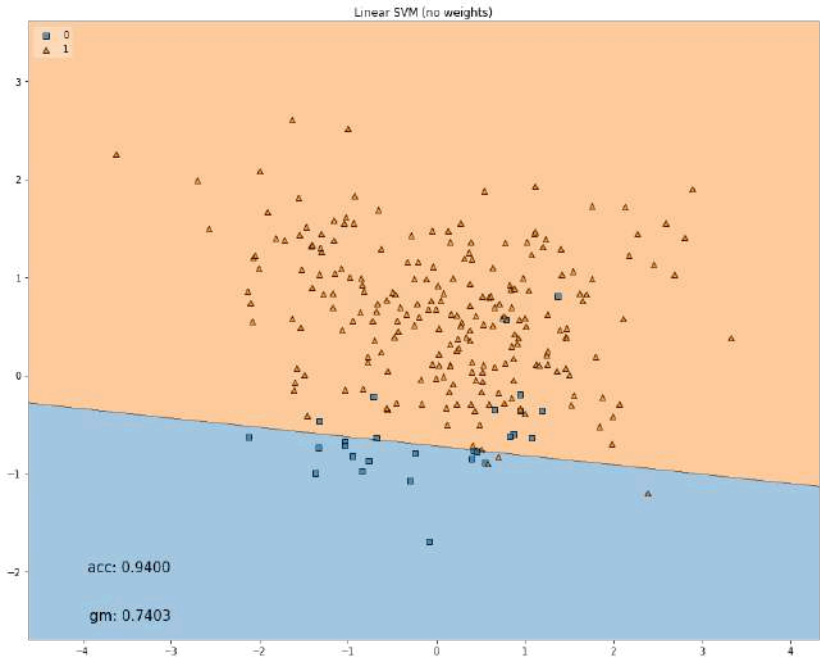
- **Provided by an expert.**
  - Supplied data is accompanied by the cost matrix that comes directly from the nature of a problem.
  - This usually requires an access to a domain expert that can assess the most realistic cost values, i.e. credit card fraud detection
- **Estimated using training data**.
  - No information on cost matrix available during training:
    - Heuristic setting of cost values: IR for cost estimation
    - Learning from training data: Thresholding via validation set

Cost-sensitive learning

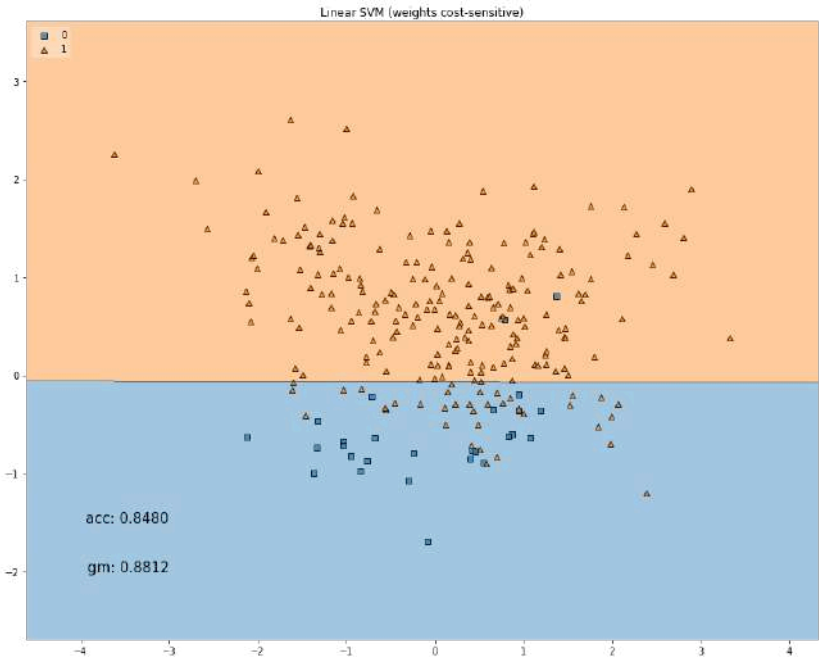
- **Direct methods:**
  - Introduce and utilize misclassification costs into the learning algorithms.
- **Meta-learning:**
  - "*Preprocessing*" mechanism for the training data or a "*post-processing*" of the output.
  - The original learning algorithm is not modified:
    - Sampling: assigning instance weights
    - Thresholding based on the Bayes decision theory: assign instances to class with minimum expected cost.

Cost-Sensitive Solutions: Meta-Learning

Linear SVM (No weights)



Linear SVM (Weights – CS)



Outline

- Introduction: Definition, properties and difficulty
- Evaluation metrics
- Data Intrinsic Characteristics
- Addressing imbalanced datasets
- Software tools for classification with imbalanced data
- Final Comments
- Surveys for a deeper study

R Packages for imbalanced classification

- Imbalance
  - Cordón, I., García, S., Fernández, A. & Herrera, F. (2018). Imbalance: Oversampling algorithms for imbalanced classification in R.. Knowl.-Based Syst., 161, 329-341.
- unbalanced: Racing for Unbalanced Methods Selection
  - Dal Pozzolo, Andrea, et al. "Racing for unbalanced methods selection. IDEAL 2013. Springer Berlin Heidelberg, 2013. 24-31.
- ROSE (Random Over Sampling Examples):
  - Menardi, G. and Torelli, N. (2014). Training and assessing classification rules with imbalanced data. Data Mining and Knowledge Discovery, 28:92–122.
- SmoteFamily
  - No associated paper

Imbalance R package: Oversampling Methods

The *Majority Weighted Minority Oversampling Technique* (MWMOTE):	• assigns higher weight to borderline instances, undersized minority clusters and examples near the borderline.
Rapidly Converging Gibbs (RACOG) and *wrapper-based RACOG* (wRACOG),	• work for discrete attributes.
Random Walk Oversampling (RWO)	• generates synthetic instances so that mean and deviation of numerical attributes remain close to the original ones.
Probability Distribution density Function estimation based Oversampling (PDFOS)	• uses multivariate Gaussian kernel methods to locally approximate the minority class.

Imbalance R package

Methods included (`filtering`):

- The *filteriNg of ovErsampled dAta us- ing non cooperaTive gamE theoRy* (NEATER): Highly based on game theory. It discards the instances with higher probability of belonging to the opposite class

Method `oversample`:

- *Wrapper* that eases calls to the described and already existing methods.

Visual method: `plotComparison`

- Pairwise comparative grid of a selected set of attributes, both in the original dataset and the oversampled one.

It includes some datasets from the KEEL repository

- Additional datasets can be easily imported: they must contain a class column having two different values.

Package unbalanced

- Oversampling:
  - ROS & SMOTE
- Undersampling:
  - RUS + Cleaning (CNN, TL,…)
- "**Racing**" algorithm:
  - Select strategy for a given unbalanced task (*ubRacing*).
  - This function compares the 8 preprocessing algorithms,
  - Plus applying the learning over the original dataset.

```
#use Racing to select the best technique for an imbalanced dataset

#configure sampling parameters
ubConf <- list(type="ubUnder", percOver=200, percUnder=200, k=2, perc=50,
               method="percPos", w=NULL)
#load the classification algorithm that you intend to use inside the Race
#see 'mlr' package for supported algorithms

results <- ubRacing(Class ~., ubIonosphere, "randomForest",
                    positive=1, ubConf=ubConf, ntree=10)
```

```
##
##  Racing for unbalanced methods selection in 10 fold CV
##  Number of candidates.....................................9
##  Max number of folds in the CV...........................10
##  Max number of experiments...............................100
##  Statistical test.....................Friedman test
##
##                          Markers:
##                             x No test is performed.
##                             - The test is performed and
##                               some candidates are discarded.
##                             = The test is performed but
##                               no candidate is discarded.
##
##  +-+-----------+-----------+-----------+-----------+-----------+
##  | |       Fold|      Alive|       Best|  Mean best| Exp so far|
##  +-+-----------+-----------+-----------+-----------+-----------+
##  |x|          1|          9|          2|     0.7745|          9|
##  |=|          2|          9|          4|     0.7473|         18|
##  |-|          3|          5|          4|     0.7585|         27|
##  |=|          4|          5|          1|     0.7668|         32|
##  |=|          5|          5|          4|     0.7585|         37|
##  |=|          6|          5|          1|      0.761|         42|
##  |=|          7|          5|          1|     0.7748|         47|
##  |=|          8|          5|          1|     0.7633|         52|
##  |=|          9|          5|          1|     0.7585|         57|
##  |=|         10|          5|          2|     0.7572|         62|
##  +-+-----------+-----------+-----------+-----------+-----------+
## Selected candidate: ubOver    metric: f1      mean value: 0.7572
```

Package smotefamily

- `smotefamily`: SMOTE oversampling extensions
  - SMOTE,
  - ADASYN,
  - ANS,
  - Borderline-SMOTE,
  - SafeLevels-SMOTE
  - Relocating Safe-level SMOTE (RSLS)

```
data_example = sample_generator(10000,ratio = 0.80)
genData = SMOTE(data_example[,-3],data_example[,3])
genData_2 = ADAS(data_example[,-3],data_example[,3],K=7)
genData_3 = BLSMOTE(data_example[,-3],data_example[,3],K=7,
                    C=5, method = "type2")

## [1] "Borderline-SMOTE done"
genData_4 = SLS(data_example[,-3],data_example[,3],K=7, C=5)

## [1] "SLS done"
```

https://github.com/cran/smotefamily

https://cran.r-project.org/src/contrib/Archive/smotefamily/

Comparison imbalanced packages R

Property	Imbalance	Unbalanced	Smotefamily	Rose
Version	1.0.0	2.0	1.2	0.0-3
Date	2018-02-18	2015-06-26	2018-01-30	2014-07-15
#Techniques	12	9	6	1
Undersampling	✗	√	✗	✗
Oversampling	√	√	√	√
SMOTE (& var.)	√	√	√	✗
Advanced OverS.	√	✗	✗	✗
Filtering	√	√	✗	✗
Wrapper	√	√	✗	✗
Visualization	√	✗	✗	✗

Python libraries: imbalanced-learn

- Dependant of `Scikit-Learn`
- A large number of preprocessing techniques
- Include **ensemble learning**
- Specific performance **metrics**
- Imbalanced **Datasets**

Preprocessing	*Technique*
Under-Sampling	Random majority under-sampling with replacement Extraction of majority-minority Tomek links Under-sampling with Cluster Centroids NearMiss-(1 & 2 & 3) Condensend Nearest Neighbour One-Sided Selection Neighboorhood Cleaning Rule Edited Nearest Neighbours Instance Hardness Threshold Repeated Edited Nearest Neighbours AllKNN
Over-Sampling	Random majority over-sampling with replacement SMOTE - Synthetic Minority Over-sampling Technique bSMOTE(1 & 2) - Borderline SMOTE of types 1 and 2 SVM SMOTE - Support Vectors SMOTE ADASYN - Adaptive synthetic sampling approach for imbalanced learning
Hybrid sampling	SMOTE + TomekLinks SMOTE + ENN
Ensemble sampling	EasyEnsemble BalanceCascade

Python libraries: imbalanced-learn

- Sampler class implements 3 main methods from the API:
  - **`fit`** computes statistics needed to resample the data;
  - **`resample`** performs the sampling with the desired balancing ratio;
  - **`fit_resample`** is equivalent to calling both methods directly.
- Input data must be in `dataFrame` or `numpy` structure.

```
from imblearn.over_sampling import RandomOverSampler


ros = RandomOverSampler(random_state=0)
X_resampled, y_resampled = ros.fit_resample(X, y)
```

Python libraries: imbalanced-learn

- Hybridizations in preprocessing are also included:

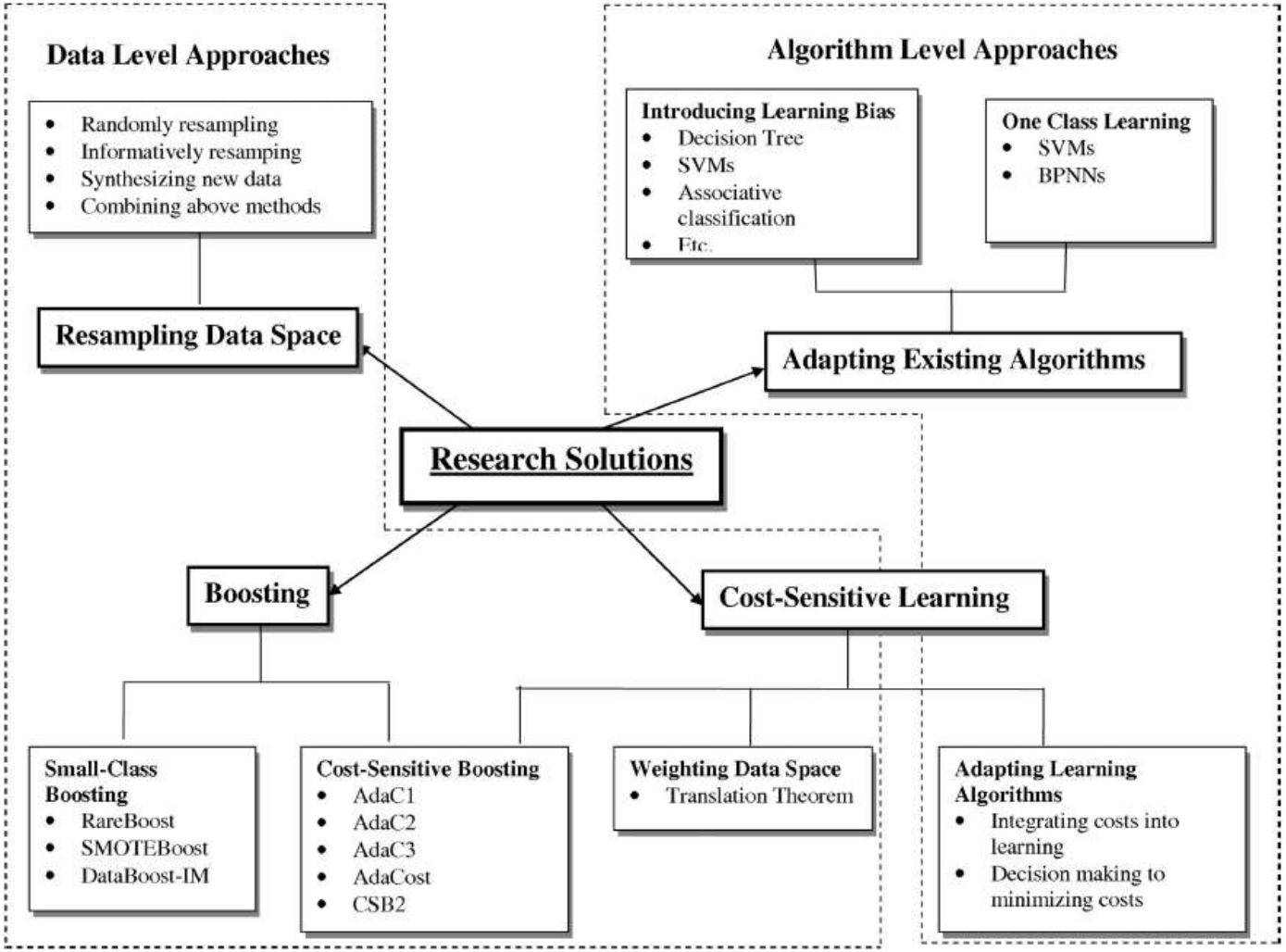
```
from imblearn.combine import SMOTEENN
from imblearn.combine import SMOTETomek
smote_enn = SMOTEENN(random_state=0)
X_new, y_new = smote_enn.fit_resample(X, y)
```

- Class **Pipeline**:
  - Inherited from the `scikit-learn` toolbox to automatically combine samplers, transformers, and estimators.
- State-of-the-art metrics to evaluate the imbalanced learning problem: `module imblearn.metrics`
  - Recall, specificity, f-measure (F1), geometric mean, Index of Balanced Accuracy (IBA), and support.

Outline

- Introduction: Definition, properties and difficulty
- Evaluation metrics
- Data Intrinsic Characteristics
- Addressing imbalanced datasets
- Software tools for classification with imbalanced data
- Final Comments
- Surveys for a deeper study

Summary and Discussion



Y. Sun, A. K. C. Wong and M. S. Kamel. Classification of imbalanced data: A review. International Journal of Pattern Recognition 23:4 (2009) 687-719.

Imbalanced Learning: CheatSheet

- Apply a standard battery of algorithms to raw problem, in order to know the base behaviour: kNN, DT, SVM.
- Observe the ROC curve (AUC) in case there is some threshold that allows a balance between positive and negative hits, within the requirements of the case study.
- In case the values obtained by the quality metrics are not sufficient, apply one of the following solutions:
  - Undersampling
  - Oversampling
  - Cost-Sensitive

CheatSheeet: Solutions

Undersampling applied in case:

- High sensitivity is desired: **RUS**
- There may be noise in the set: **RUS, CNN, TL**
- There are a high number of negative examples: **RUS**
- There is a need to reduce the learning time: **RUS, Class Weights**
- High Dimensionality: **RUS, Class Weights**

Oversampling applied in case:

- A good balance to be kept for TPR, TNR: **ROS**
- There may be subclusters of the positive class: **SMOTE**
- Positive class reinforcement needed in overlapping areas: **SMOTE**
- Few data samples: **SMOTE**
- High Dimensionality: **ROS, Class Weights**

CheatSheet: Final comments

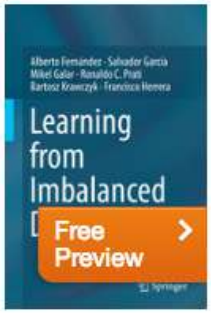
- It is quite convenient to analyze the ROC curve to find adequate probability thresholds: a posteriori approach
- Hyperparametrization: find optimal values
  - the number of neighbors (k) in kNN,
  - the pruning in a decision tree,
  - the kernel values in an SVM…
- Ensembles-based techniques are very powerful, best if these are used in synergy with preprocessing:
  - RUSBoost
  - SMOTEBagging.

Outline

- Introduction: Definition, properties and difficulty
- Evaluation metrics
- Data Intrinsic Characteristics
- Addressing imbalanced datasets
- Software tools for classification with imbalanced data
- Final Comments
- Surveys for a deeper study

The Imbalanced Learning Book

» Computer Science » Artificial Intelligence



© 2018

Learning from Imbalanced Data Sets

Authors: **Fernández**, A., **García**, S., **Galar**, M., **Prati**, R.C., **Krawczyk**, B., **Herrera**, F.

Offers a comprehensive review of imbalanced learning widely used worldwide in many real applications, such as fraud detection, disease diagnosis, etc

» see more benefits

Buy this book

eBook $109.00

price for USA in USD (gross)

Buy eBook

- ISBN 978-3-319-98074-4
- Digitally watermarked, DRM-free
- Included format: EPUB, PDF
- ebooks can be used on all reading devices
- Immediate eBook download after purchase

Hardcover $149.99

» FAQ » Policy

About this book

This book provides a general and comprehensible overview of imbalanced learning. It contains a formal description of a problem, and focuses on its main features, and the most relevant proposed solutions. Additionally, it considers the different scenarios in Data Science for which the imbalanced classification can create a real challenge.

Book Metrics

Readers 4

Downloads 4344

Provided by **Book**metrix

Reviews on Imbalanced Classification

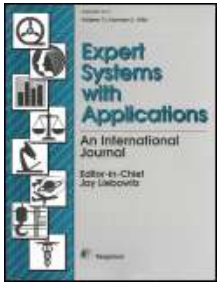
- 2004:
  - A Study of the Behavior of Several Methods for **Balancing** Machine Learning Training Data (Gustavo E. A. P. A. Batista; Ronaldo C. Prati; Maria Carolina Monard)

- 2009:
  - **Learning from Imbalanced D**ata (Haibo He, and Edwardo A. Garcia)

  - **Classification of Imbalanced Data: A Review** (Yanmin Sun, Andrew K. C. Wong, Mohamed S. Kamel)

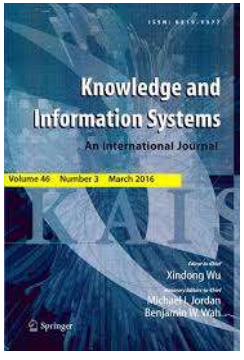
Reviews on Imbalanced Classification

- 2012:
  - Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics (Victoria López, Alberto Fernández, Jose G. Moreno-Torres, Francisco Herrera)



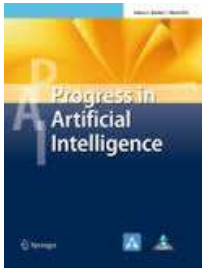
- 2013:
  - An insight into classification with imbalanced data: Empirical results and current trends on using **data intrinsic characteristics** (Victoria López, Alberto Fernández, Salvador García, Vasile Palade, Francisco Herrera)

- 2015:
  - Class imbalance revisited: a new experimental setup to assess the performance of treatment methods (Ronaldo C. Prati, Gustavo E. A. P. A. Batista, Diego F. Silva)



Reviews on Imbalanced Classification

- 2016:
  - A **Survey** of Predictive Modeling on Imbalanced Domains (Paula Branco, Luis Torgo, Rita P. Ribeiro)
  - Dealing with Data Difficulty Factors While Learning from Imbalanced Data (Jerzy Stefanowski)
  - Learning from class-imbalanced data: **Review of methods and applications** (Guo Haixiang, Li Yijing, Jennifer Shang , Gu Mingyun, Huang Yuanyue, Gong Bing)
  - Learning from imbalanced data: **open challenges and future directions** (Bartosz Krawczyk)



Seminario Permanente de Formación
en Inteligencia Artificial
Aplicada a la Defensa

Clasificación en conjuntos de datos con clases no balanceadas

Alberto Fernández

Instituto Andaluz de Investigación en Data Science and Computational Intelligence (DaSCI)

Dpto. Ciencias de la Computación e I.A.
Universidad de Granada

alberto@decsai.ugr.es
https://www.dasci.es

UNIVERSIDAD DE GRANADA

DaSCI