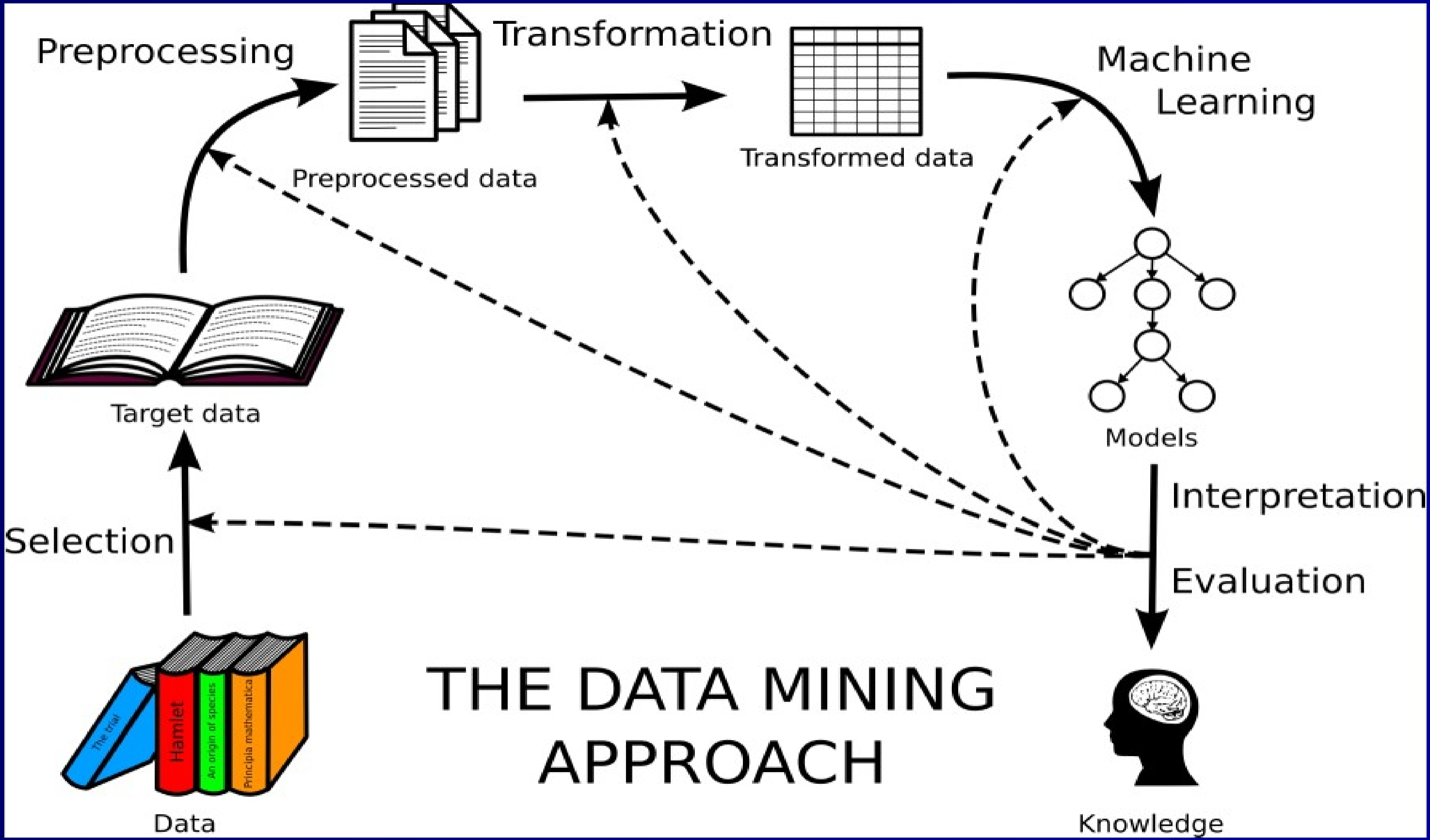
DATA PREPROCESSING FOR FURTHER ANALYSIS

mejor que duplicar muestras, interpolacion
Mejor que linear

DATA MINING PROCESS: THE PIPELINE
KDD: KNOWLEDGE DISCOVERY IN DATABASES



Summary

Good data preparation is key to producing valid and reliable models

Usama Fayyad, Ph.D.
Chief Data Officer & Executive VP
Yahoo! Inc.

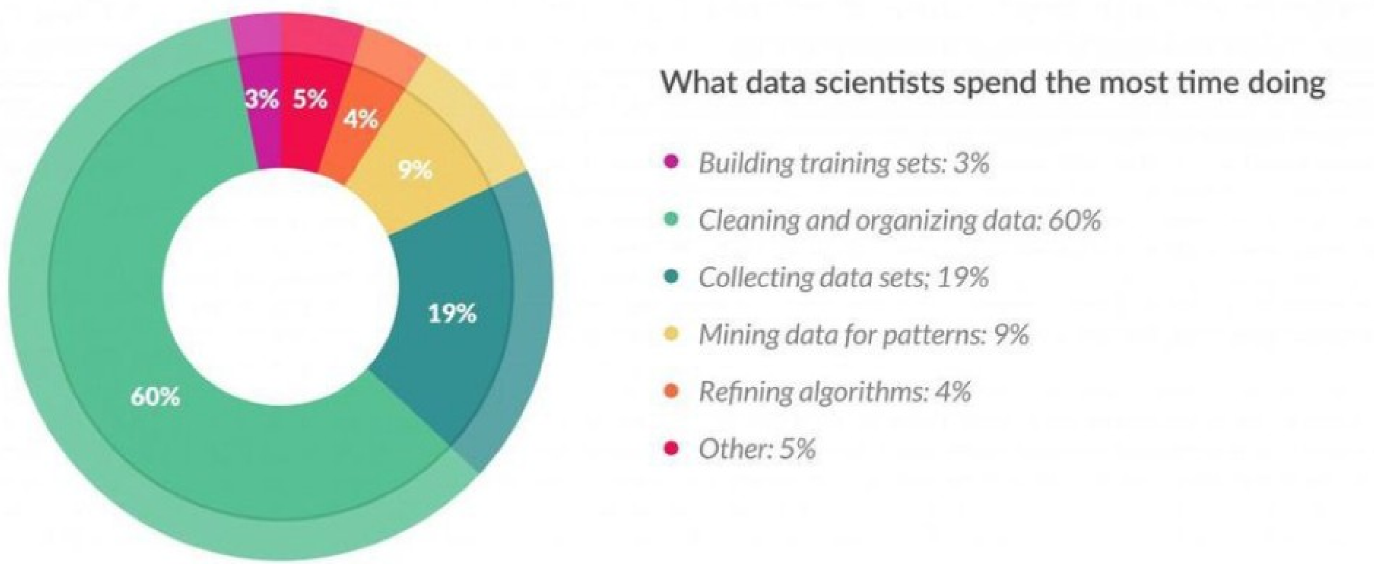
Y! Research

- We all worry about algorithms, they are fascinating
- Most of us know that data mining in practice is mostly data prep work

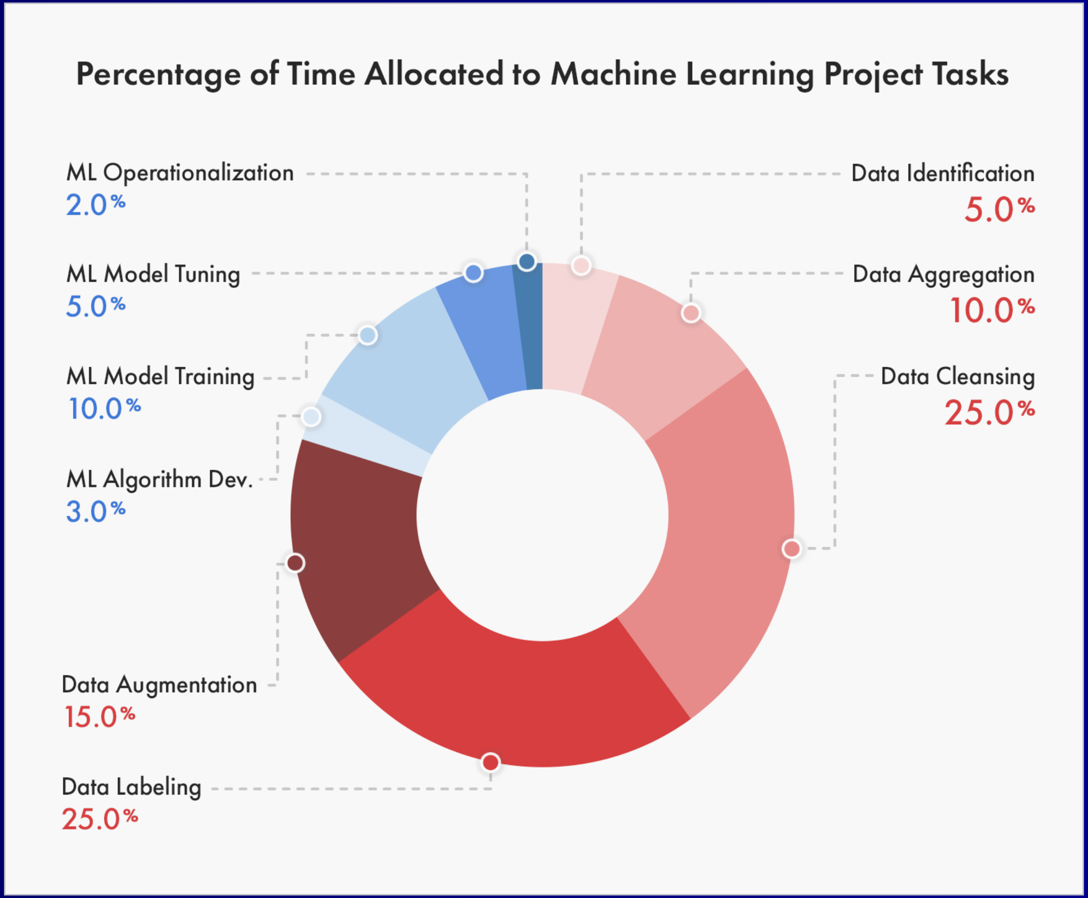
Furthermore, data need to be clean and formatted as requested by the mining tool. Data preprocessing can easily count for 70%–80% of the total KDD processing time. In the literature, very often, algorithms are compared on computation time (efficiency) without considering the time spent in data preprocessing.

Summary

According to a survey in Forbes, data scientists spend **80%** of their time on **data preparation:**



Summary



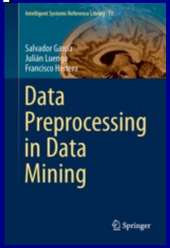
Percentage of Time Allocated to Machine Learning Project Tasks
ML Operationalization
2.0%
ML Model Tuning
5.0%
ML Model Training
10.0%
ML Algorithm Dev.
3.0%
Data Augmentation
15.0%
Data Labeling
25.0%
Data Identification
5.0%
Data Aggregation
10.0%
Data Cleansing
25.0%

Outline

- Data acquisition
- “Meta data”: attribute definition

- Numeric attribute normalization
- One-hot encoding → nominal features
- Dealing with missing values
- Discretization
- Unbalanced Target-Class Distribution

- More on “data preprocessing”:



Data acquisition

- Data in a flat file to be analyzed:
  - Fixed-column format
  - Delimited format: tab, comma "," [csv ";"], other
  - Use of "," to delimitate different variables (not decimals!!)
  - E.g. C4.5, Weka and most data analysis software use "arff"-like use comma-delimited data

- Trick --> use of Excel, Calc
- Special commands for reading data files, e.g. "read.table()"

- Specific preprocessing operations in each domain: NLP, images, gene expression – bioinformatics, voice-signal…

- Centered on "general" preprocessing filters for any kind of data

Metadata

```
@RELATION iris

@ATTRIBUTE sepallength  REAL
@ATTRIBUTE sepalwidth   REAL
@ATTRIBUTE petallength  REAL
@ATTRIBUTE petalwidth   REAL
@ATTRIBUTE class  {Iris-setosa,Iris-versicolor,

@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
```

- Field descriptions
- Field types:
  - binary, ***nominal (categorical), ordinal***, numeric, ...
  - discrete, continuous... ¡NO!
- Field role:
  - input, predictor, feature, variable... : inputs for modeling
  - target, class : output
  - id/auxiliary : keep, but not use for modeling
  - ...

Numeric attribute normalization

- When computing distances between pairs of instances:
  - Are all numeric attributes in the same range of values (i.e. max-min)?
- Normalize all numeric attributes to [0,1] interval
- Indispensable for computing distances (e.g. K-NN)

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{p} (x_{ir} - x_{jr})^2}$$

- Automatically done by many data analysis software tools

Nominal features
One hot encoding

id	color
1	red
2	blue
3	green
4	blue

One Hot Encoding

id	color_red	color_blue	color_green
1	1	0	0
2	0	1	0
3	0	0	1
4	0	1	0

Human-Readable

Pet
Cat
Dog
Turtle
Fish
Cat

Machine-Readable

Cat	Dog	Turtle	Fish
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1
1	0	0	0

Missing Values

- Original missing data can appear in several forms:
  - ‹empty field› “0” “.” “999” “NA” “?” …
- Standardize missing value code(s)
- How can we deal with missing values?
  - Missing value imputation discipline

Missing Values

- Dealing with missing values:
  - ignore records with missing values
  - treat missing value as a separate value
  - Imputation: fill in with mean or median values (class conditioned conditional mean or median)

    o la media condicionada a la clase
  - Advanced imputation techniques: K-NN neighbours, EM algorithm (“Expectation and Maximization”) [Dempster et al.’ 77]

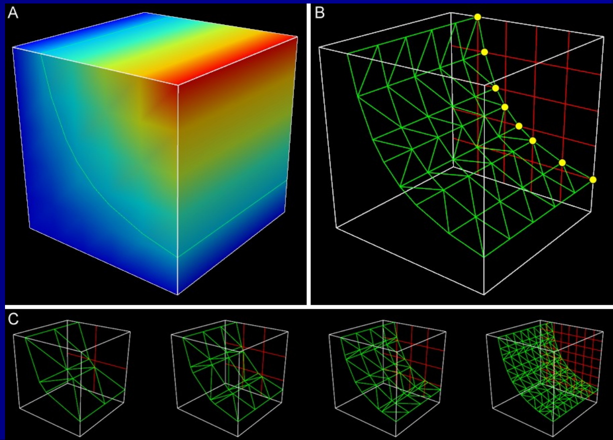
Relation: iris

No.	sepallength Numeric	sepalwidth Numeric	petallength Numeric	petalwidth Numeric	class Nominal
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3.0	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	6.3	2.5	4.9	1.5	Iris-versicolor
5	6.1	?	4.7	1.2	Iris-versicolor
6	6.4	2.9	4.3	1.3	Iris-versicolor
7	6.5	3.0	5.2	2.0	Iris-virginica
8	6.2	3.4	5.4	2.3	Iris-virginica
9	5.9	3.0	5.1	1.8	Iris-virginica

Quitar toda la columna si mas del 30" de los datos son nulos

Discretization

- Goal: reduce the number of values of a continuous attribute by GROUPING them into a number of INTERVALS (bins)
- Some methods require discrete values, e.g. most versions of naïve Bayes and Bayesian networks, several decision tree algorithms...



Discretization

- Better results than dealing with continuous data?

- Decision (in naïve Bayes):
  - When discretizing... multinomial feature (probabilities)

  versus
  - When not discretizing... assuming a density function

- Decision (in K-NN classifiers):
  - When not discretizing... Euclidean distance

  versus
  - When discretizing... overlap distance (a=a distance=0, b≠a distance=1, b≠c distance=1)

Types of discretization

- Unsupervised *versus* supervised → use class info?

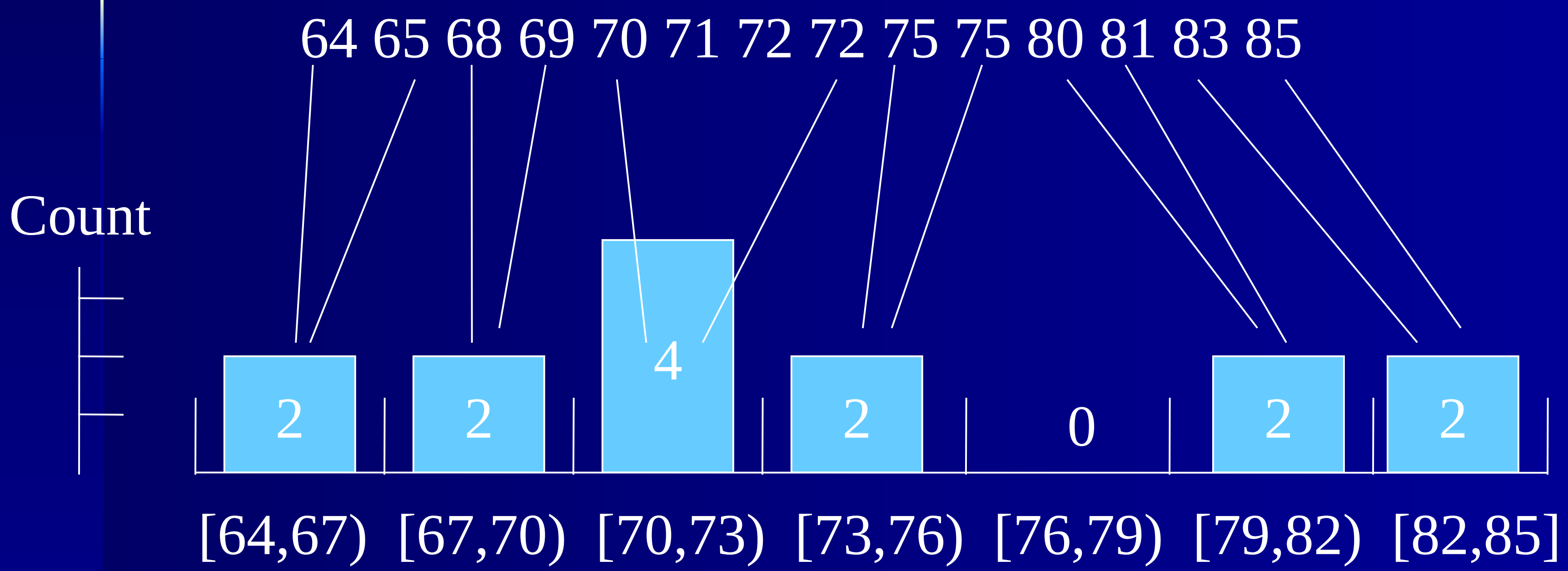
- Static – one attribute at a time –
  versus
- Dynamic – searching for combinations of intervals in all the features simultaneously –

IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 4, APRIL 2013

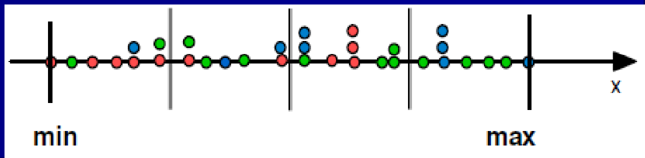
A Survey of Discretization Techniques: Taxonomy and Empirical Analysis in Supervised Learning

Salvador García, Julián Luengo, José Antonio Sáez, Victoria López, and Francisco Herrera

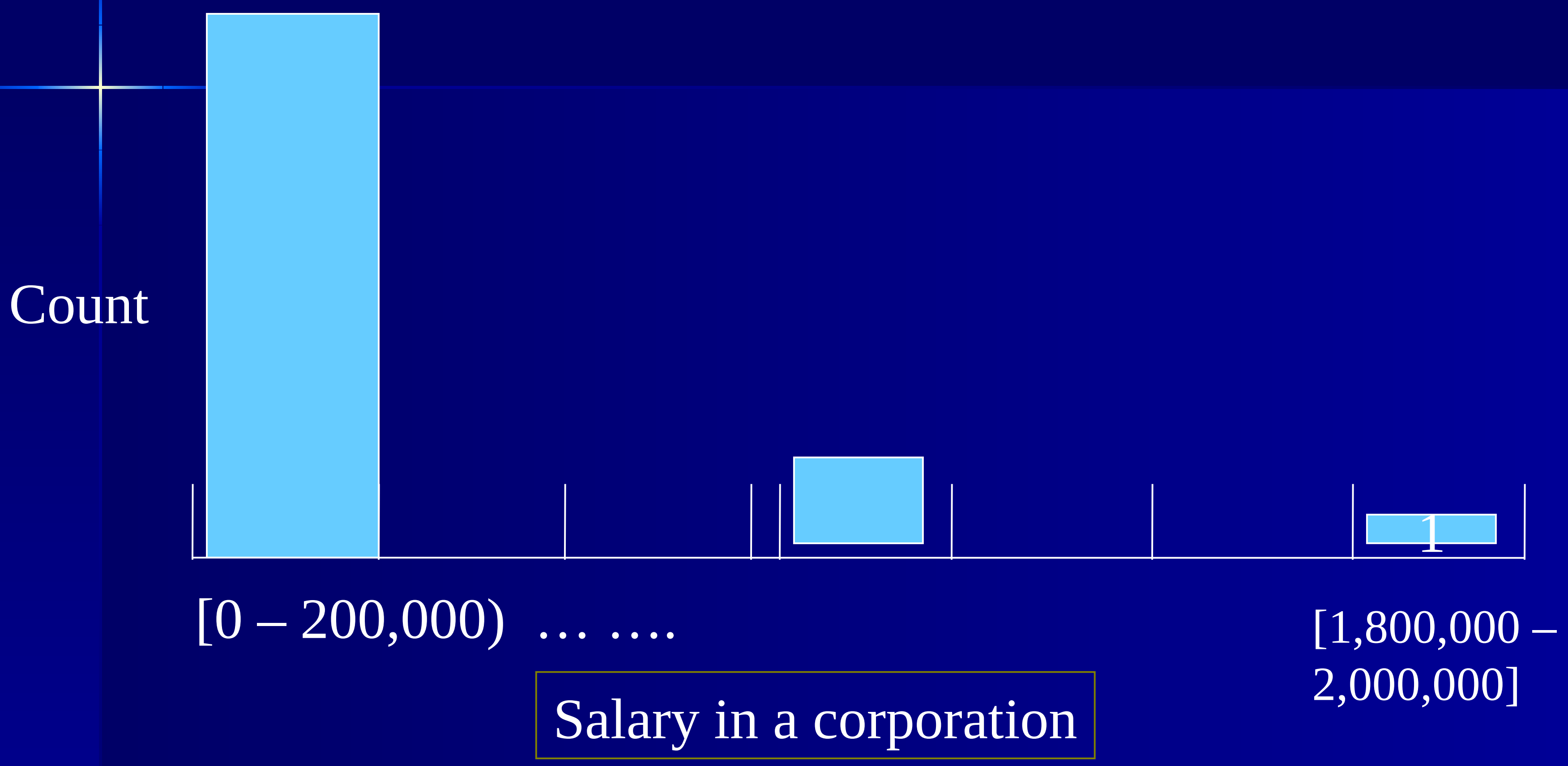
Discretization: Equal-width



- Deciding the number of bins before hand (7)
- Dividing the *[max_value – min_value]* range in 7 equal-width ranges



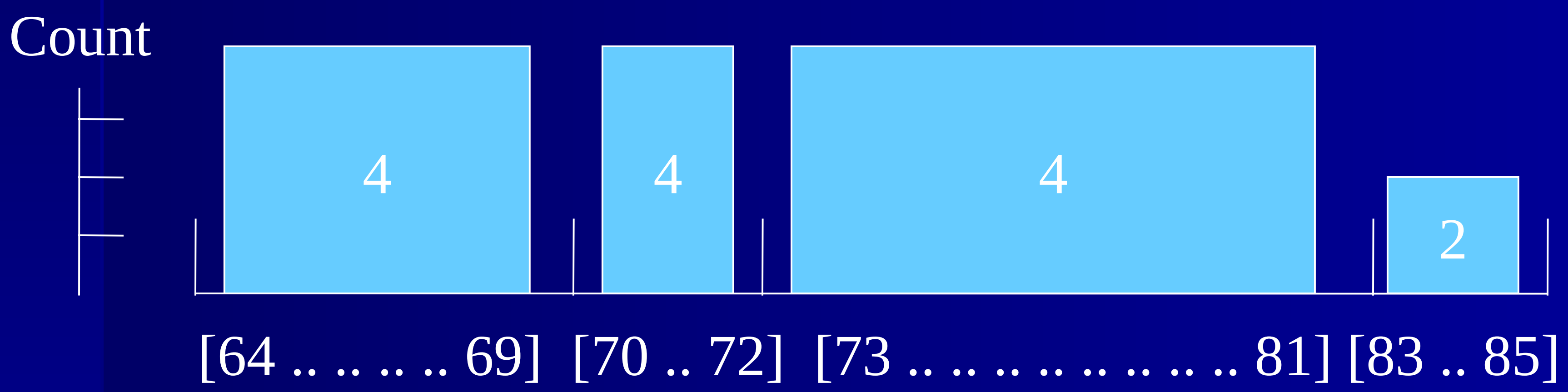
Equal-width may produce clumping
Uneven distribution



Value unbalance, with respect to the number of cases...
What can we do to get a more even distribution? Not easy task...

Discretization: Equal-frequency

Temperature values:
64 65 68 69 70 71 72 72 75 75 80 81 83 85



– Fixing the number of bins before hand (4)

Discretization number of intervals

mejor que duplicar muestras, interpolacion
Mejor que lineal, con normales multivariantes

- Several discretization methods require deciding the number of intervals before-hand

- A large number of intervals:
  - Much of the original info is retained, but...
  - Intervals may not have enough samples to calculate needed statistics for learning, i.e. $p(c|x_i)=3/10=30/100$?...
  - → Unreliable statistics estimation

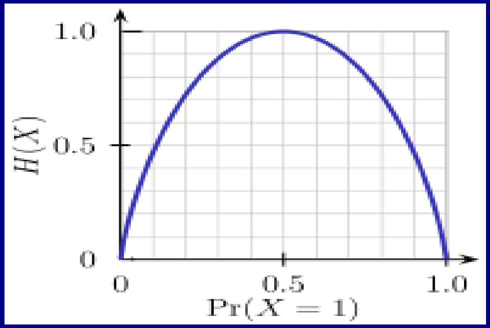
- Guessing the number of intervals, literature heuristics:
  - Number_samples / (3 x numer_class_values)
  - Square_root (non_missing_values)

Supervised discretization

minimizo la entropia de la clase en cada intervalo

```
Predictor:      {64  65 68  69  70  71  72 72  75  75  80  81  83  85}
Class variable: {Yes No Yes Yes Yes No  No No  Yes Yes No Yes Yes Yes}
```

$$H(X) = -\sum_{i=1}^{n} p(x_i) \cdot \log_2 p(x_i)$$



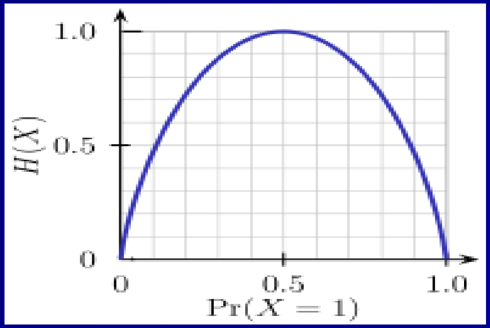
Fayyad and Irani (1993). Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning. IJCAI, 1022-1029

Supervised discretization

mejor que duplicar muestras, interpolacion
Mejor que lineal, con normales multivariantes

```
Predictor:      {64  65 68  69  70 | 71  72 72 | 75  75  80  81  83  85}
Class variable: {Yes No Yes Yes Yes | No  No No | Yes Yes No Yes Yes Yes}
```

$$H(X) = -\sum_{i=1}^{n} p(x_i) \cdot \log_2 p(x_i)$$



Fayyad and Irani (1993). Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning. IJCAI, 1022-1029

Discretization: considerations

- Equal Width → simplest, good for many classes
  – can fail for unequal distributions

- Equal Frequency → usually gives better results

- Class-dependent can be better for classification
  – Discretizes in a single bin features that do not change over class values – constant → removal

- Decision trees → build discretization on the fly

- Many other methods exist ...

IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 4, APRIL 2013

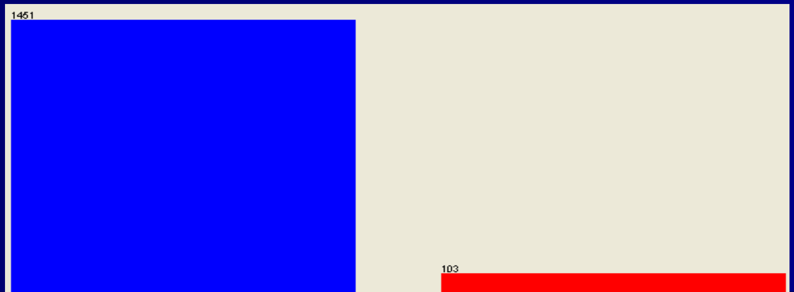
A Survey of Discretization Techniques: Taxonomy and Empirical Analysis in Supervised Learning

Salvador García, Julián Luengo, José Antonio Sáez, Victoria López, and Francisco Herrera

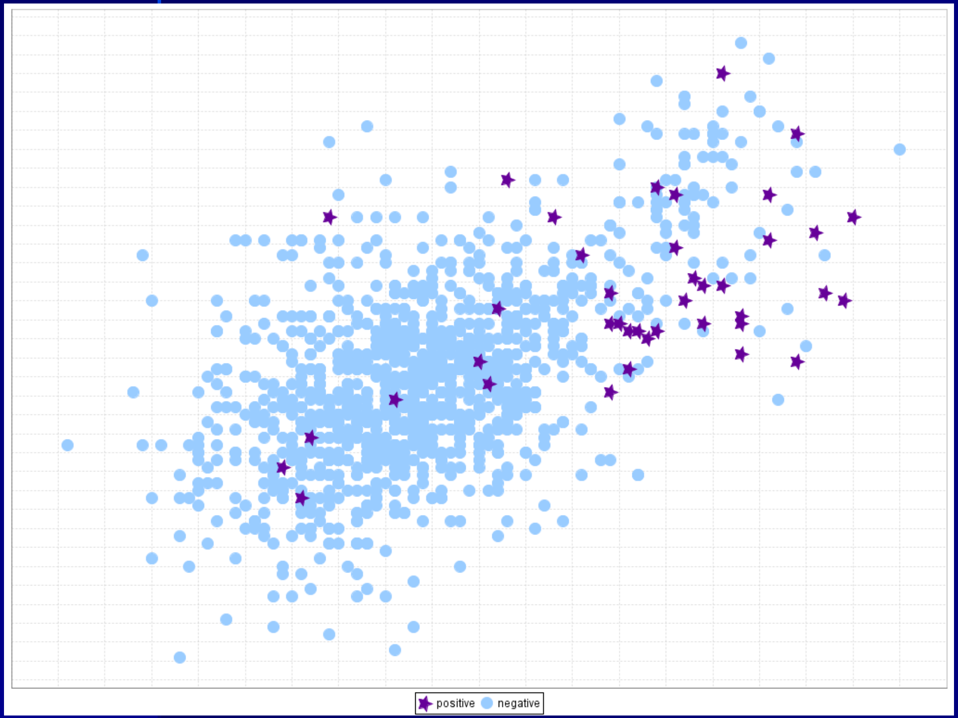
Unbalanced Target Distribution

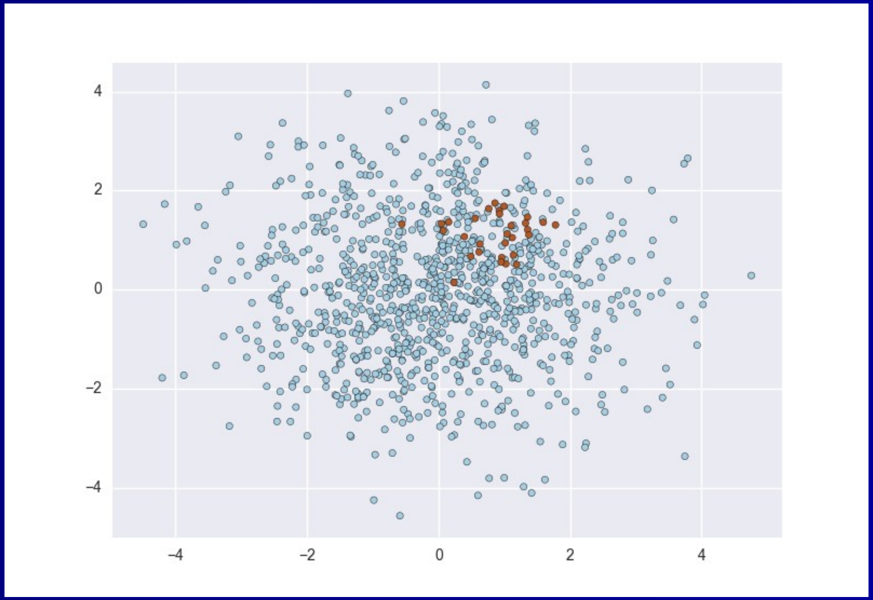
- Sometimes, classes have very unequal frequency
  - Fraud detection: 98% non-fraudulent, 2% fraudulent
  - medical diagnosis: 90% healthy, 10% disease
  - eCommerce: 99% don't buy, 1% buy
  - spam e-mails: 95% non-spam, 5% spam

- Majority class classifier → 97% accuracy → but useless
- Interested in increasing the TPR in the minority class



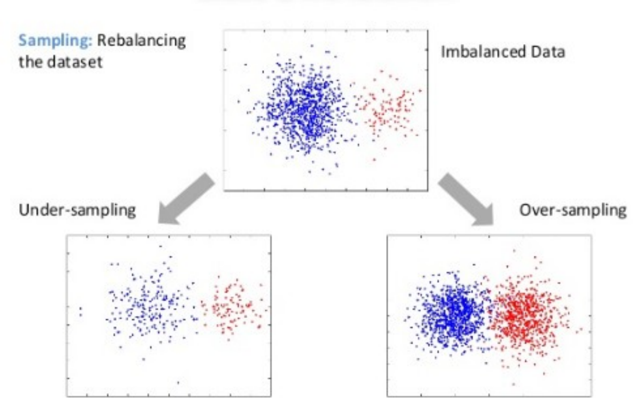
Unbalanced Target Distribution
Difficult task!!





Unbalanced Target Distribution

- Large set of techniques → balance training sets, stratified sampling...
- Training with different misclassification costs
  - Usually, minority-class samples → repeated-reweighted in the training phase
  - However → majority-class samples usually tend to increase their misclassification level
- WEKA: CostSensitiveClassifier + CostMatrix

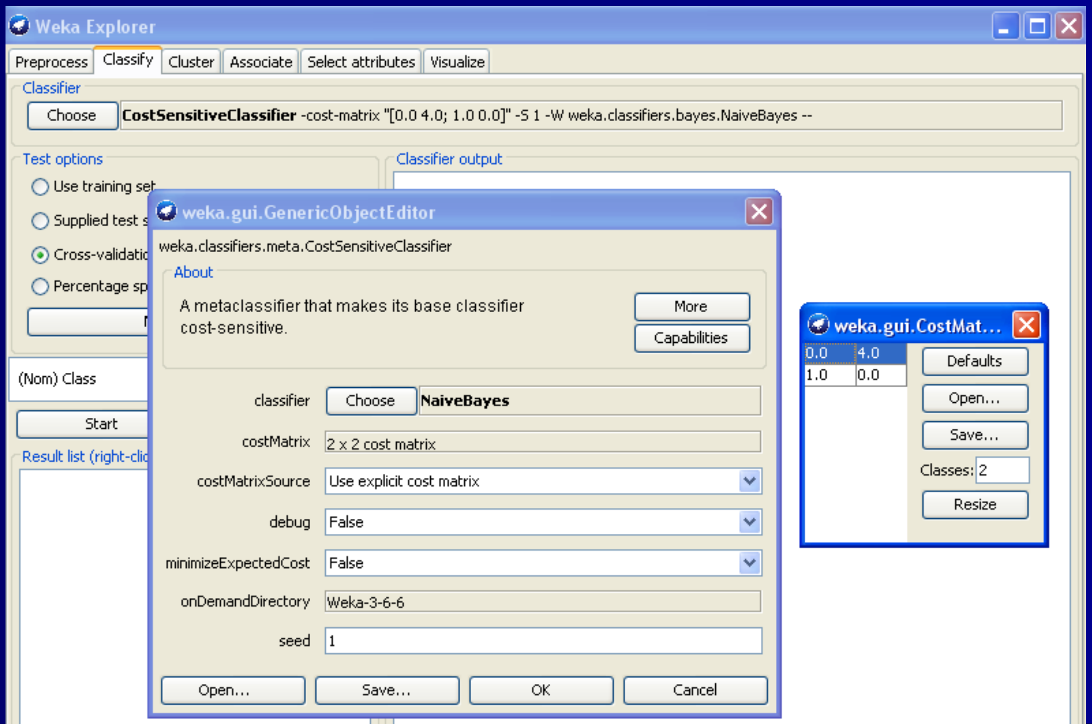


el generar muestras nuevas siempre
en training, nunca en test

Handling Unbalanced Data

WEKA: CostSensitiveClassifier + CostMatrix

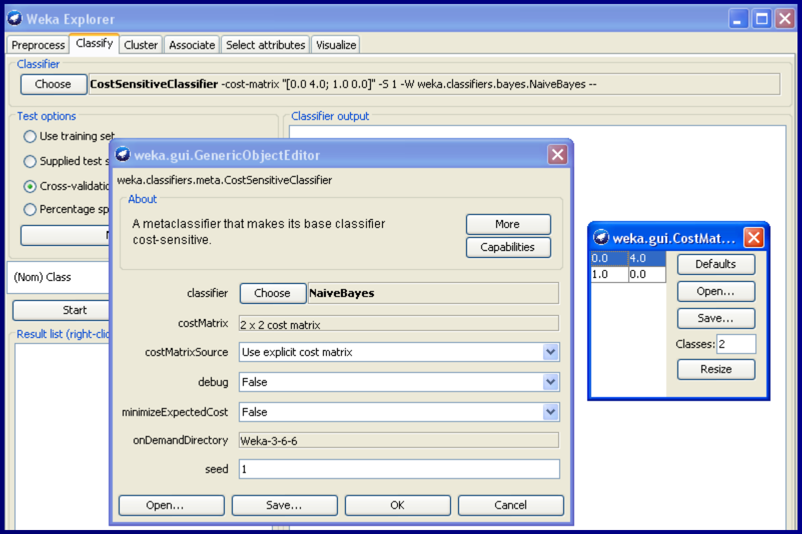
- Explicit introduction of “costs” per class. For example:
  - Training → reweighted minority class samples → “cost matrix”
  - Training → majority class samples not reweighted
  - Test → no repetition of the samples!



Handling Unbalanced Data

WEKA: CostSensitiveClassifier + CostMatrix

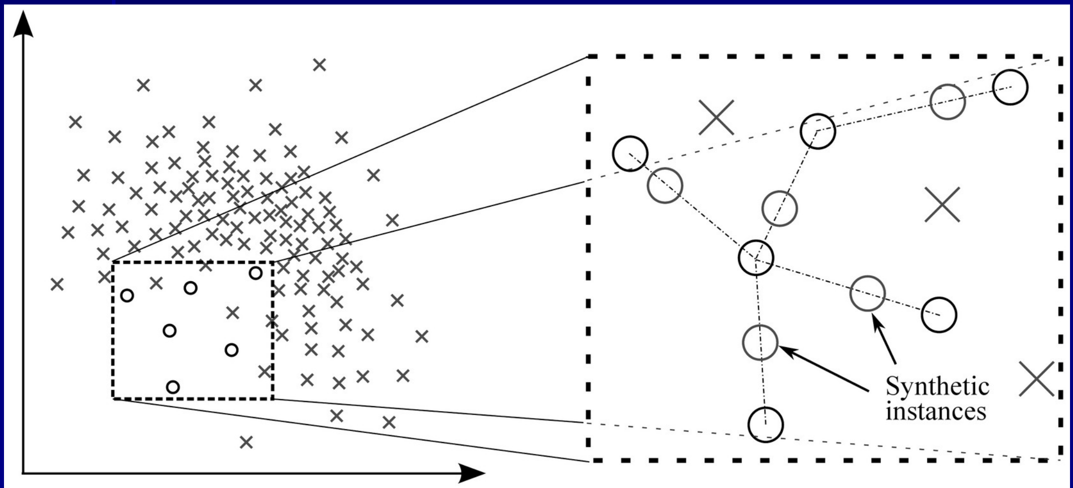
- Bank-marketing dataset [link] → load in WEKA
- Check class imbalance
- Learn naive Bayes + 10-fold cross-validate:
  - with equal missclassification costs
  - enlarging misclassification cost of minority class
  - check confusion matrix → recall of minority class? recall of majority class?



Handling Unbalanced Data

WEKA: Filter – Supervised – instances - SMOTE

mejor que duplicar muestras, interpolacion
Mejor que lineal, con normales multivariantes



ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning

Haibo He, Yang Bai, Edwardo A. Garcia, and Shutao Li

Journal of Artificial Intelligence Research 16 (2002) 321–357 Submitted 09/01; published 06/02

SMOTE: Synthetic Minority Over-sampling Technique

Nitesh V. Chawla CHAWLA@CSEE.USF.EDU
Department of Computer Science and Engineering, ENB 118
University of South Florida
4202 E. Fowler Ave.
Tampa, FL 33620-5399, USA

Kevin W. Bowyer KWB@CSE.ND.EDU
Department of Computer Science and Engineering
384 Fitzpatrick Hall
University of Notre Dame
Notre Dame, IN 46556, USA

Lawrence O. Hall HALL@CSEE.USF.EDU
Department of Computer Science and Engineering, ENB 118
University of South Florida
4202 E. Fowler Ave.
Tampa, FL 33620-5399, USA

W. Philip Kegelmeyer WPK@CALIFORNIA.SANDIA.GOV
Sandia National Laboratories
Biosystems Research Department, P.O. Box 969, MS 9951
Livermore, CA, 94551-0969, USA

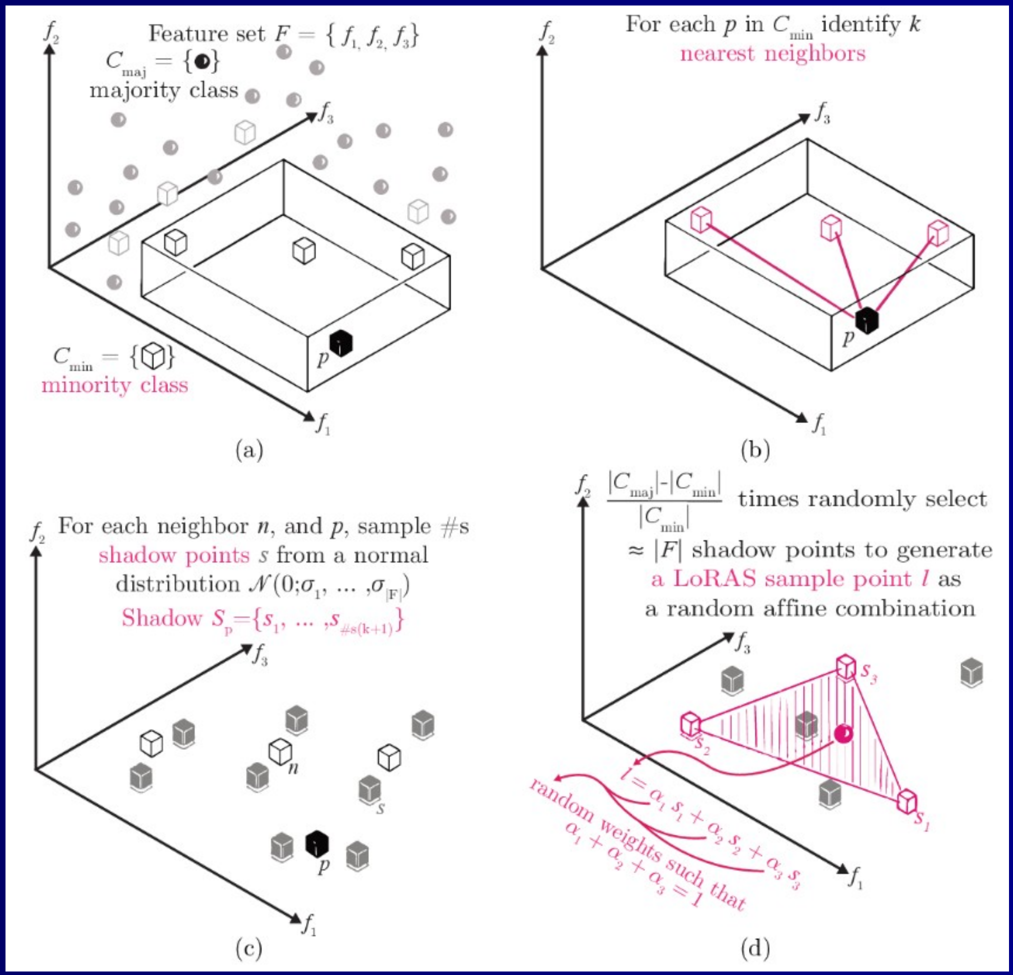
Abstract

An approach to the construction of classifiers from imbalanced datasets is described. A dataset is imbalanced if the classification categories are not approximately equally represented. Often real-world data sets are predominately composed of "normal" examples with only a small percentage of "abnormal" or "interesting" examples. It is also the case that the cost of misclassifying an abnormal (interesting) example as a normal example is often much higher than the cost of the reverse error. Under-sampling of the majority (normal) class has been proposed as a good means of increasing the sensitivity of a classifier to the minority class. This paper shows that a combination of our method of over-sampling the minority (abnormal) class and under-sampling the majority (normal) class can achieve better classifier performance (in ROC space) than only under-sampling the majority class. This paper also shows that a combination of our method of over-sampling the minority class and under-sampling the majority class can achieve better classifier performance (in ROC space) than varying the loss ratios in Ripper or class priors in Naive Bayes. Our method of over-sampling the minority class involves creating synthetic minority class examples. Experiments are performed using C4.5, Ripper and a Naive Bayes classifier. The method is evaluated using the area under the Receiver Operating Characteristic curve (AUC) and the ROC convex hull strategy.

SMOTE: synthetic minority over-sampling technique
NV Chawla, KW Bowyer, LO Hall... - Journal of artificial ..., 2002 - jair.org
An approach to the construction of classifiers from imbalanced datasets is described. A dataset is imbalanced if the classification categories are not approximately equally represented. Often real-world data sets are predominately composed of ``normal''examples ...

Handling Unbalanced Data

"SMOTE variants"



Machine Learning (2021) 110:279–301
https://doi.org/10.1007/s10994-020-05913-4

LoRAS: an oversampling approach for imbalanced datasets

Saptarshi Bej[1] · Narek Davtyan[1] · Markus Wolfien[1] · Mariam Nassar[1] · Olaf Wolkenhauer[1]

Handling Unbalanced Data
"SMOTE variants"

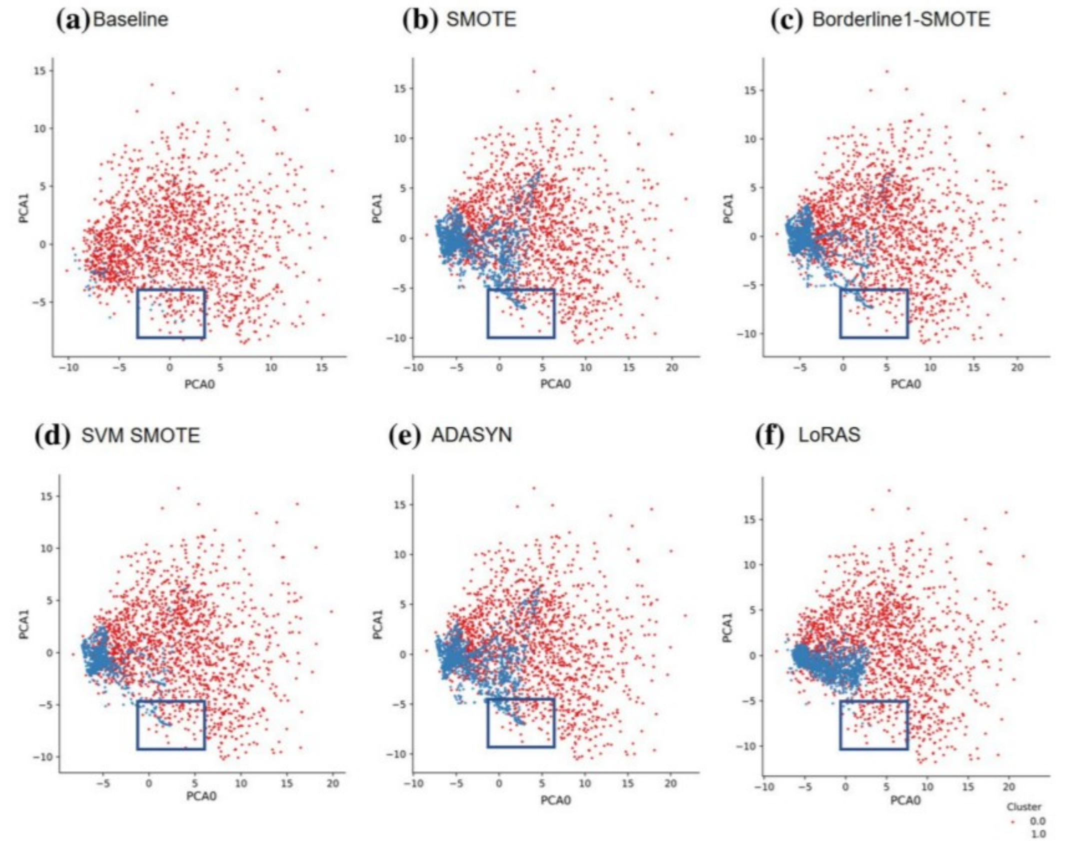


Fig. 2 Figure showing for principal component analysis plot of ozone dataset for baseline data and over-sampled data with several oversampling strategies for the ozone_level dataset. The boxed region in each subplot shows a neighbourhood of outliers and how each oversampling stategy generates synthetic samples in that neighbourhood

- PC1 + PC2 visualization
- Box for outlier samples

SOFTWARE - PACKAGES

The `caret` Package

11 Subsampling For Class Imbalances

Contents

In classification problems, a disparity in the frequencies of the observed classes can have a significant negative impact on model fitting. One technique for resolving such a class imbalance is to subsample the training data in a manner that mitigates the issues. Examples of sampling methods for this purpose are:

- *down-sampling*: randomly subset all the classes in the training set so that their class frequencies match the least prevalent class. For example, suppose that 80% of the training set samples are the first class and the remaining 20% are in the second class. Down-sampling would randomly sample the first class to be the same size as the second class (so that only 40% of the total training set is used to fit the model). **caret** contains a function (`downSample`) to do this.
- *up-sampling*: randomly sample (with replacement) the minority class to be the same size as the majority class. **caret** contains a function (`upSample`) to do this.
- *hybrid methods*: techniques such as SMOTE and ROSE down-sample the majority class and synthesize new data points in the minority class. There are two packages (**DMwR** and **ROSE**) that implement these procedures.

Imbalanced learn | Getting Started | User Guide | API reference | Examples | Release history | More

imbalanced-learn documentation

API reference > Over-sampling methods > SMOTE

SMOTE

class **imblearn.over_sampling.SMOTE**(**, sampling_strategy='auto', random_state=None, k_neighbors=5, n_jobs=None*) [source]

Class to perform over-sampling using SMOTE.

This object is an implementation of SMOTE - Synthetic Minority Over-sampling Technique as presented in [1].

Other interesting filters

- A huge variety of data filters exists

- Depending on the analysis goals, they may be useful:
  - Standardize numeric attributes
  - Outlier value detection
  - Transformations: numeric to binary, numeric to nominal...
  - Add noise to an attribute values
  - ...

- Shown filters: “general” filters for any dataset
- Specialized filters and data-cleaning depending on the application: NLP, images, biological data, html data...

Summary

"Good data preparation is key to producing valid and reliable models"

Usama Fayyad, Ph.D.
Chief Data Officer & Executive VP
Yahoo! Inc.

Y!Research

- We all worry about algorithms, they are fascinating
- Most of us know that data mining in practice is mostly data prep work

Furthermore, data need to be clean and formatted as requested by the mining tool. Data preprocessing can easily count for 70%–80% of the total KDD processing time. In the literature, very often, algorithms are compared on computation time (efficiency) without considering the time spent in data preprocessing.