Class imbalance problem

- "A study of the behavior of several methods for balancing machine learning training data." *ACM Sigkdd Explorations Newsletter* 6.1 (2004): 20-29.
- "Classification of imbalanced data by combining the complementary neural network and SMOTE algorithm." *International Conference on Neural Information Processing*. Springer Berlin Heidelberg, 2010.
- "Class imbalances versus class overlapping: an analysis of a learning system behavior." *Mexican international conference on artificial intelligence*. Springer Berlin Heidelberg, 2004.

20 Oct, 2016

Hunsik Shin

hunsik@dm.snu.ac.kr

Introduction

- **Class imbalance**
  - Occurs in which examples in training data belonging to **one class heavily outnumber** the examples in the other class
  - In real data, minority class describes an **infrequent** but **important event**(e.g. fraud, disease)
  - **Major obstacle** in inducing classifiers in imbalanced domain

- **Two main approaches to deal with imbalanced data**
  - ➢ **Data-level approach** : re-balancing the class distribution before a classifier is trained
    - Under-sampling : random under sampling, Tom다 links, Wilson's Edited nearest neighbor rule(ENN)
    - Over-sampling : random over sampling, Synthetic Minority Over-sampling(SMOTE)
    - Combined method : under-sampling+ over-sampling
  - ➢ **Algorithm-level approach** : strengthening the existing classifier by adjusting algorithms to recognize the smaller classes
    - Cost-sensitive learning

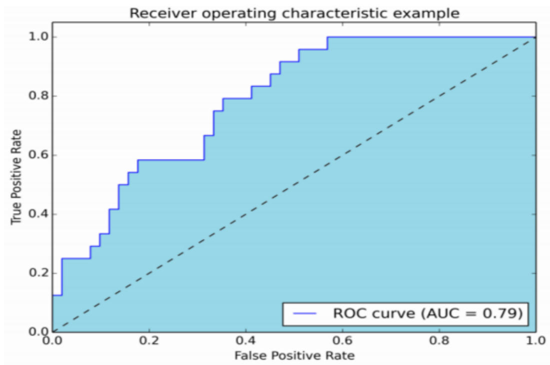
Introduction

- **Evaluation**
  - The most straightforward way to evaluate the performance → confusion matrix
  - Common metric is '**Error rate**' or '**Accuracy**'
    - Err(error rate) = $\frac{FP+FN}{TP+FN+FP+TN}$
    - Accuracy = $\frac{TP+TN}{TP+FN+FP+TN}$ = 1 - Err
  - If proportion of majority class = 99%, then accuracy becomes 99%

    (by simply forecasting every new example as the majority)
  - The area under the ROC curve(**AUC**) or **Geometric mean** are used for performance evaluation
  - AUC represents the expected performance as a single scalar

	p' (Predicted)	n' (Predicted)
p (Actual)	True Positive (TP)	False Negative (FN)
n (Actual)	False Positive (FP)	True Negative (TN)

<confusion matrix>



<area under the curve(AUC)>

Introduction

- **Data-level approach**
  1. **Under-sampling**
     - Tomek-links
     - Condensed Nearest Neighbor(CNN) Rule
  2. **Over-sampling**
     - Synthetic Minority Over-sampling Technique(SMOTE)
  3. **Combined method**
     - Complementary Neural Network(CMTNN) with SMOTE

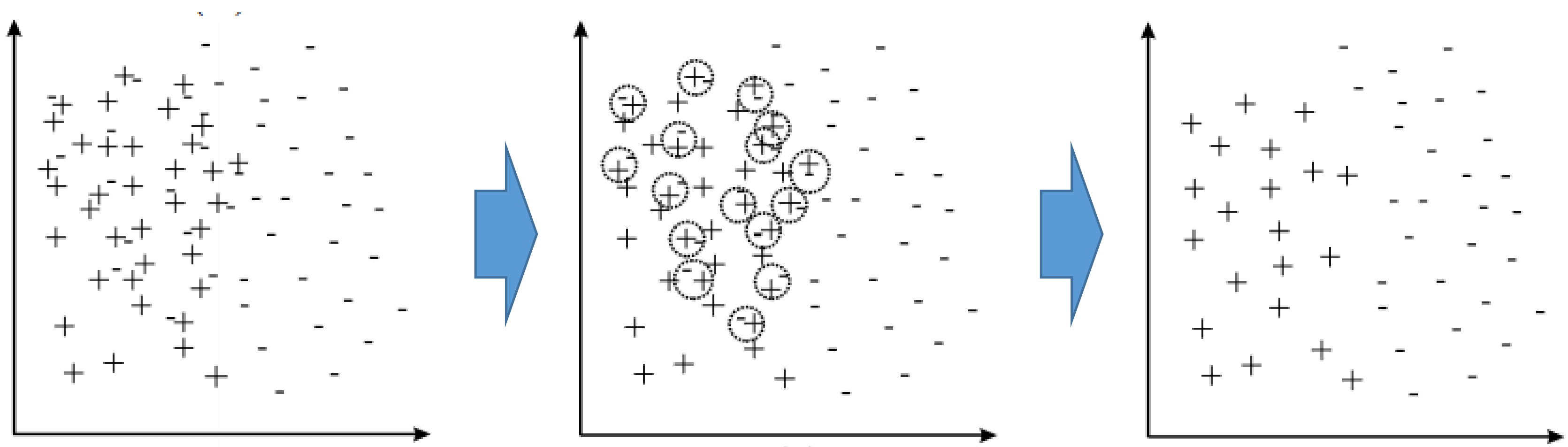
Under-sampling – “Tomek-link”

- **Tomek-link**
  - A pair of two examples belonging to different classes
  - If two examples form Tomek-link, then either one of these examples is **noise** or both are **borderline**
  - Tomek-link can be used as an **under-sampling** or **data cleaning method**

Def) A (E_i, E_j) *pair is called a Tomek link,*

$$if\ there\ is\ not\ an\ example\ \ E_l\ such\ that\ d(E_i, E_l) < d(E_i, E_j)\ \ or\ \ d(E_j, E_l) < d(E_i, E_j).$$

* E_i, E_j *belongs to different classes*



<Over-sampled data> <Tomek-link identification> <noise and borderline removal>

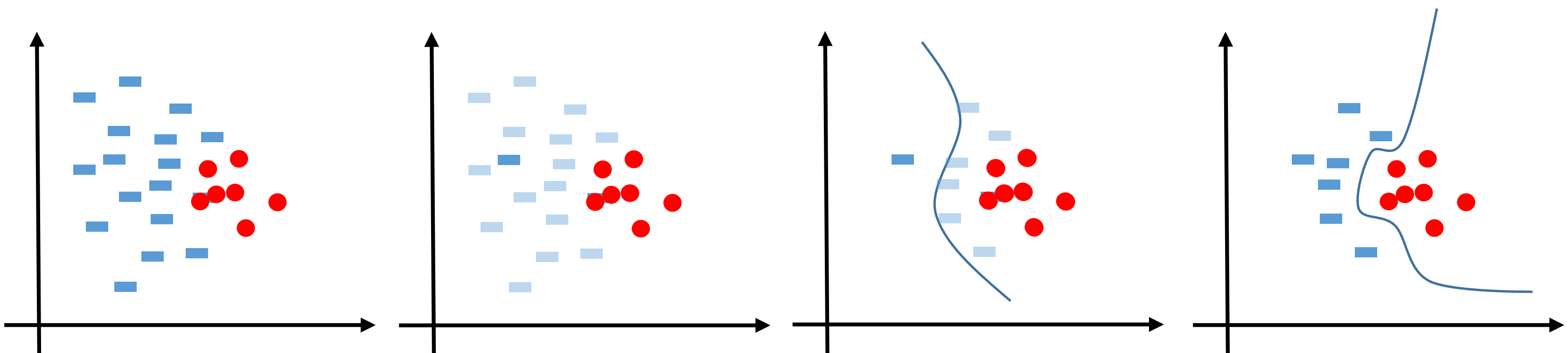
Under-sampling – "Condensed Nearest Neighbor"

- **Condensed Nearest Neighbor Rule**
  - To find a consistent subset of examples.
    - Subset that correctly classifies the whole examples by 1-nearest neighbor
  - Condensed Nearest Neighbor(CNN) can be used as an under-sampling method

Algorithm)

1. Randomly draw one majority class example and all examples from minority class
2. Using this subset of examples, classify all examples by 1-nearest neighbor
3. Misclassified examples from majority class are moved to the existed subset
4. This subset becomes a consistent subset of all examples

→ **Not** guarantee to find the **smallest consistent subset.**



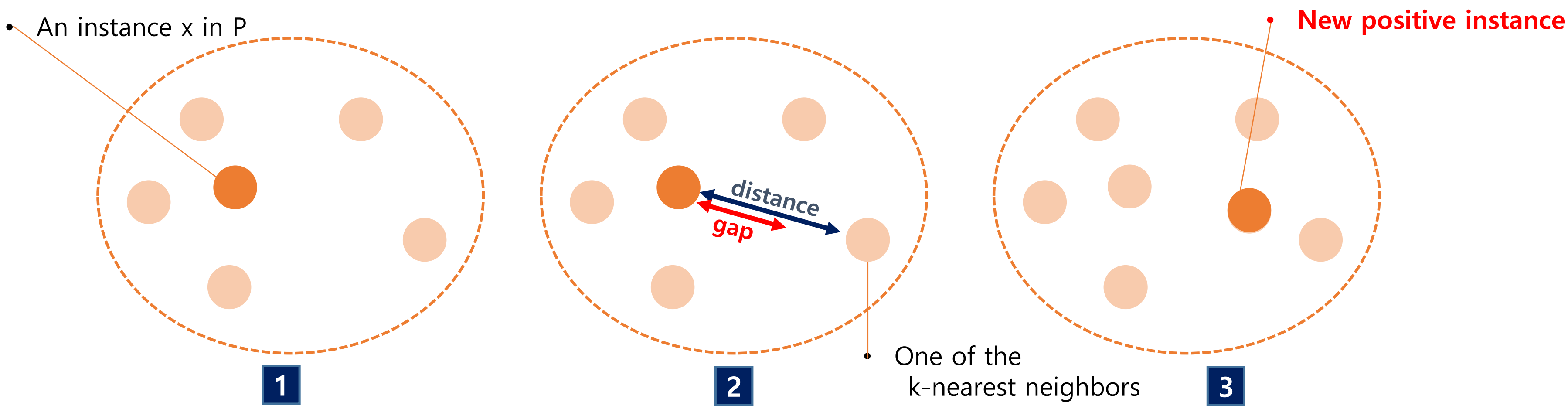
<All examples> **<Algorithm 1 ~ 3>** **<Consistent subset>**

Over-sampling – "SMOTE"

- **Synthetic Minority Over-sampling(SMOTE)**
  - SMOTE is an over-sampling technique that increases a number of new minority class instances by interpolation method.

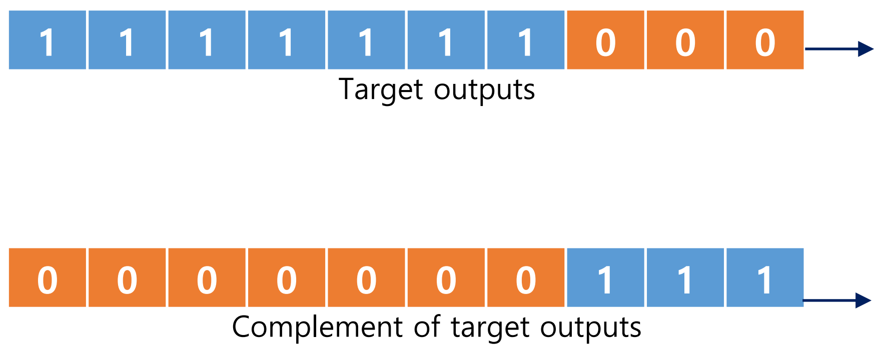
O is the original data set
P is the set of positive instances (minority class instances)
For each instance x in P
  1 Find the k-nearest neighbors (minority class instances) to x in P
    Obtain y by randomizing one from k instances
  2 $difference = x - y$
    gap = random number between 0 and 1
    $n = x + difference * gap$
  3 Add n to O
End for

The Synthetic Minority Oversampling Technique(SMTOE) algorithm



Combined method – "CMTNN + SMOTE"

- **Complementary Neural Network(CMTNN) : Under-sampling**
  - ➢ CMTNN is a technique using a pair of complementary feedforward neural networks
    - Truth Neural Network(Truth NN) : trained to predict the degree of the truth membership
    - Falsity Neural Network(Falsity NN) : trained to predict the degree of the false membership



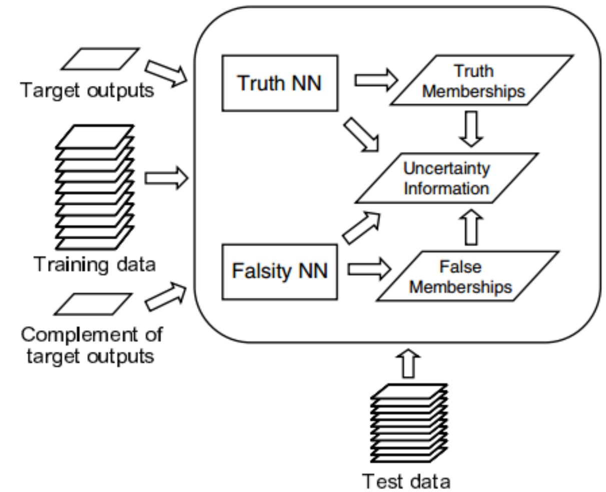


Fig. 1. Complementary Neural Network

- ➢ Under-sampling technique

For Truth NN : If $Y_{Truth\ i} \neq O_{Truth\ i}$ then $M_{Truth} \leftarrow M_{Truth}\ U\ \{T_i\}$

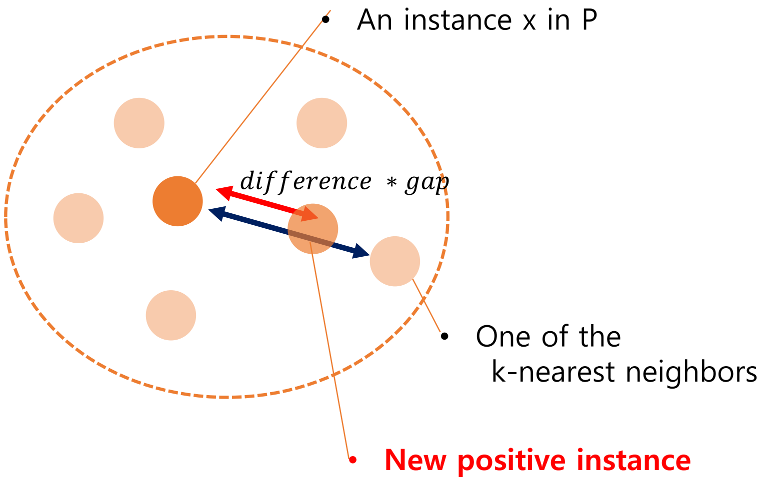
For Falsity NN : If $Y_{Falsity\ i} \neq O_{Falsity\ i}$ then $M_{Falsity} \leftarrow M_{Falsity}\ U\ \{T_i\}$

1. $T_c \leftarrow T - (M_{Truth} \cap M_{Falsity})$
2. $T_c \leftarrow T - (M_{Truth} \cup M_{Falsity})$

- $M_{Truth} = the\ misclassification\ patterns\ of\ Truth\ NN$
- $M_{Falsity} = the\ misclassification\ patterns\ of\ Falsity\ NN$
- $Y = the\ predction\ ouputs$
- $T = the\ training\ data$
- $O = the\ actual\ data$

Combined method – "CMTNN + SMOTE"

- **Synthetic Minority(SMOTE) : Over-sampling**
  - SMOTE is an over-sampling technique that increases a number of new minority class instances by interpolation method.



Name of data set	No. of instances	No. of attributes	Minority class (%)	Majority class (%)
Pima Indians Diabetes data	768	8	34.90	65.10
German Credit data	1000	20	30.00	70.00
Haberman's Survival data	306	3	26.47	73.53
SPECT Heart data	267	22	20.60	79.40

Table 1. Characteristics of data sets used in the experiment

- **The Proposed Combined Techniques**
  1. Under-sampling only the majority using **CMTNN 1** → over-sampling the minority class using **SMOTE**
  2. Under-sampling only the majority using **CMTNN 2** → over-sampling the minority class using **SMOTE**
  3. Over-sampling the minority class using **SMOTE** → under-sampling only the majority using **CMTNN 1**
  4. Over-sampling the minority class using **SMOTE** → under-sampling only the majority using **CMTNN 2**

→ Training **ANN, kNN, SVM** classifiers

Combined method – "CMTNN + SMOTE"

- **Experiment result : AUC, G-mean**

Techniques	Pima Indian Diabetes data		German Credit data		Haberman's Survival data		SPECT Heart data	
	GM	AUC	GM	AUC	GM	AUC	GM	AUC
Original Data	70.12	0.8276	63.92	0.7723	33.11	0.5885	64.05	0.7590
a. ENN	72.64	0.8298	70.74	0.7794	50.45	0.6305	71.80	0.7895
b. Tomek links	73.11	0.8288	70.48	0.7793	51.88	0.6323	72.88	0.8178
c. SMOTE	74.30	0.8281	71.48	0.7777	58.60	0.6345	73.59	0.8241
d. Technique I (Majority) + SMOTE	75.55	0.8332	72.03	0.7855	60.00	0.6452	73.86	0.8374
e. Technique II (Majority) + SMOTE	74.53	0.8300	73.32	0.7873	62.78	0.6770	74.32	0.8273
f. SMOTE + Technique I	75.00	0.8285	71.52	0.7844	61.41	0.6653	73.00	0.8264
g. SMOTE + Technique II	74.96	0.8300	72.07	0.7860	58.59	0.6248	74.04	0.8373
Best technique	d	d	e	e	e	e	e	d
Second best	f	e & g	g	g	f	f	g	g

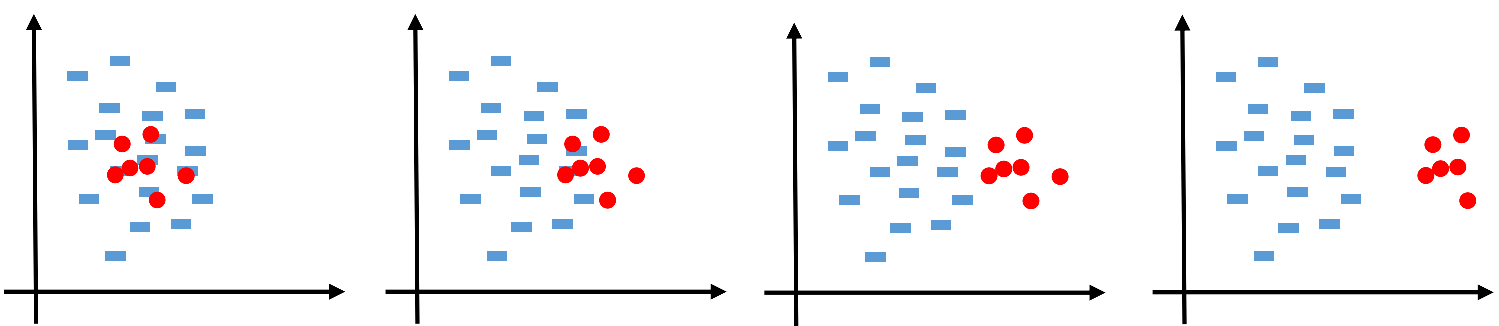
Table. 2. The results of G-Mean and AUC for each data set classified by ANN

Class imbalance vs overlapping

- **Class overlapping**
  - Several works point out 'class imbalance' as an obstacle on applying ML algorithm
  - In some cases, learning algorithms perform well on several imbalanced domain
    → Not solely caused by **class imbalance**, but is also related to the degree of **data overlapping**
    → The degree of class overlapping has a **strong correlation** with class imbalance

Positive instances	Distance of Class Centroids				
	0	1	2	3	9
1%	50.00% (0.00%)	64.95% (9.13%)	90.87% (6.65%)	98.45% (2.44%)	99.99% (0.02%)
2.5%	50.00% (0.00%)	76.01% (6.41%)	95.82% (3.11%)	97.95% (2.12%)	99.99% (0.02%)
5%	50.00% (0.00%)	81.00% (2.86%)	98.25% (1.45%)	98.95% (1.11%)	100.00% (0.00%)
10%	50.00% (0.00%)	86.69% (2.11%)	98.22% (1.14%)	99.61% (0.55%)	99.99% (0.02%)
15%	50.00% (0.00%)	88.41% (2.37%)	98.92% (0.75%)	99.68% (0.49%)	99.99% (0.02%)
20%	50.00% (0.00%)	90.62% (1.44%)	99.08% (0.42%)	99.90% (0.21%)	99.99% (0.02%)
25%	50.00% (0.00%)	90.88% (1.18%)	99.33% (0.32%)	99.90% (0.14%)	99.98% (0.03%)
30%	50.00% (0.00%)	90.75% (0.81%)	99.24% (0.29%)	99.86% (0.14%)	99.99% (0.02%)
35%	50.00% (0.00%)	91.19% (0.94%)	99.36% (0.43%)	99.91% (0.08%)	99.99% (0.02%)
40%	50.00% (0.00%)	90.91% (0.99%)	99.46% (0.10%)	99.90% (0.13%)	99.99% (0.03%)
45%	50.00% (0.00%)	91.73% (0.79%)	99.44% (0.22%)	99.90% (0.09%)	99.98% (0.04%)
50%	50.00% (0.00%)	91.32% (0.68%)	99.33% (0.19%)	99.87% (0.13%)	99.99% (0.03%)

<Mean AUC obtained from classifiers varying class priors and class overlapping, (standard dev)>



<Pictorial representation of artificial data>

Application?

- **Limitations**
  1. **Under-sampling**
     - Information loss of majority class examples
  2. **Over-sampling**
     - Over-fitting to minority class(or over-generalization)


рахмат
danke
謝謝
spas
ngiyabonga
teşekkür ederim
Баярлалаа
mersi
kia ora
barka
welalin
tack
mahalo
tapadh leat
спасибо
faafetai lava
vinaka
blagodaram
dank je
misaotra
paldies
grazzi
matondo
спасибі
kiitos
dankie
nanni
nandri
dhanyavad
thank you
gracias
хвала
asante
manana
bedankt
bayarlalaa
gràçie
hvala
mauruuru
köszönöm
akun
dankon
aciū
obrigada
tenki
enkosi
djiere dieuf
tau
mochchakkeram
murakoze
дякую
mamnun
dziękuję
chnorakaloutioun
gratias ago
gràcies
sulpáy
taiku
go raibh maith agat
chokrane
sobodi
dëkuji
sukriya
kop khun krap
obrigado
mèsi
didi madloba
sağolun
ありがとう
tanemirt
grazie
arigatô
takk
dakujem
trugarez
najis tuke
terima kasih
rahmet
ευχαριστώ
diolch
dhanyavadagalu
shukriya
mercê
мерси
kam sah hamnida
rahmat
তোমাকে ধন্যবাদ
감사합니다
xièxie
merci