Class prediction for high-dimensional class-imbalanced data

Lara Lusa and Rok Blagus

Institute for Biostatistics and Medical Informatics – Faculty of Medicine
University of Ljubljana, Slovenia

Univerza v Ljubljani

Lara.Lusa at mf.uni-lj.si

ibmi

Class prediction for high-dimensional class-imbalanced data

Develop a rule that can be used to predict the class membership of new samples

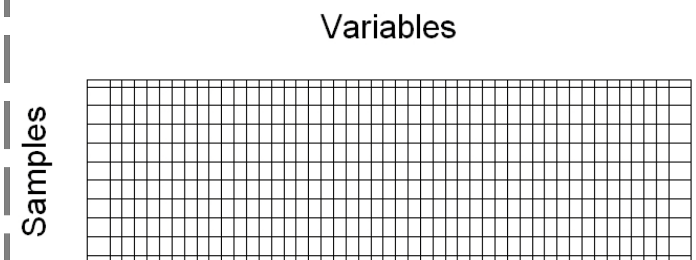
Variables

RULE

Class

p >> n

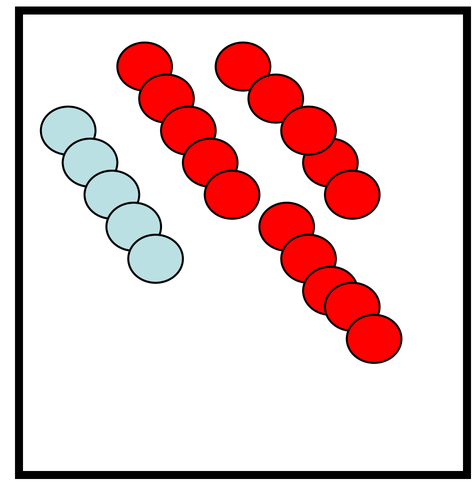
variables samples



$n_1 \neq n_2$

Class 1 samples

Class 2 samples



How to perform high-dimensional classification

- Design a **developmental study (training set)** to answer a specific clinical question
  - Select **n** subjects (patients, customers, ...)
  - Measure **p** variables (genes, SNPs, mRNAs, characteristics of a customer)
  - **High-dimensional data if p >> n :**
    - **Select a subset of variables (variable selection)**

Obtain a **RULE** based on the values of the variables for the classification of new samples

The RULE must be **completely specified**

- variables included
- how to combine them
- normalization of data
- thresholds for classification
- …

Predict class membership or calculate risk scores for new samples (**test set**)

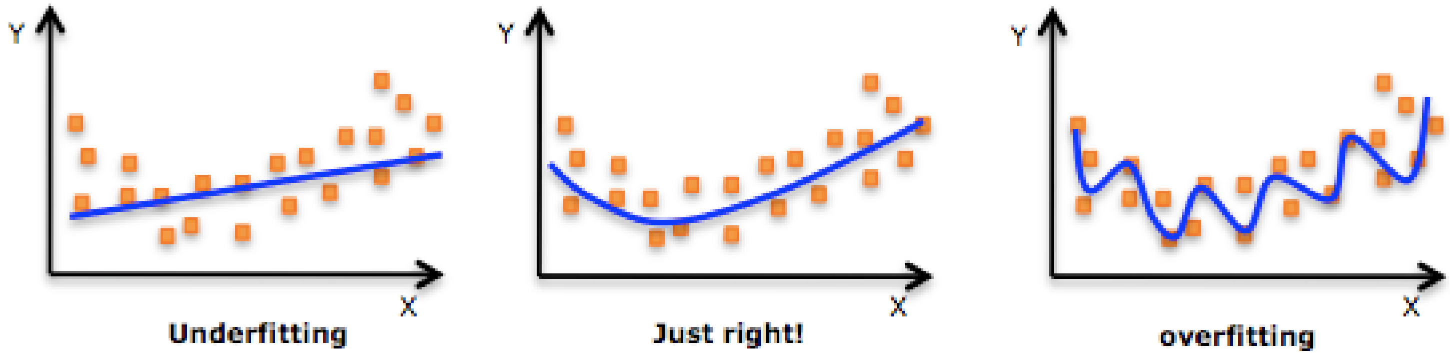
Class prediction rules

- Many different methods are available to specify the classification rule
  - Discriminant (linear diagonal) analysis (DLDA, DQDA)
    - Prediction analysis of Microarrays (PAM)
  - Support vector machines (SVM)
  - Penalized regression methods (PLR-L1, PLR-L2)
  - Classification and regression trees (CART) and random forests (RF)
  - k-nearest neighbor methods (k-NN)
  - …

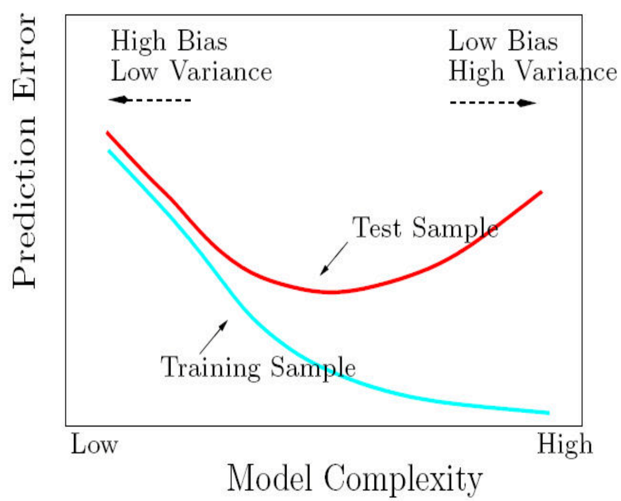
- Risk of ***overfitting*** and spurious findings
- ***Simple methods*** perform well with high-dimensional data (Dudoit et al, 2002)
- ***Variable selection*** - reducing the number of variables usually leads to more accurate classification

- No single method is optimal in every situation
  - *No Free Lunch Theorem*: in absence of assumptions we should not prefer any classification algorithm over another
  - *Ugly Ducking Theorem*: in absence of assumptions there is no "best" set of features

Overfitting with high-dimensional data



Can be easily obtained with high-dimensional data

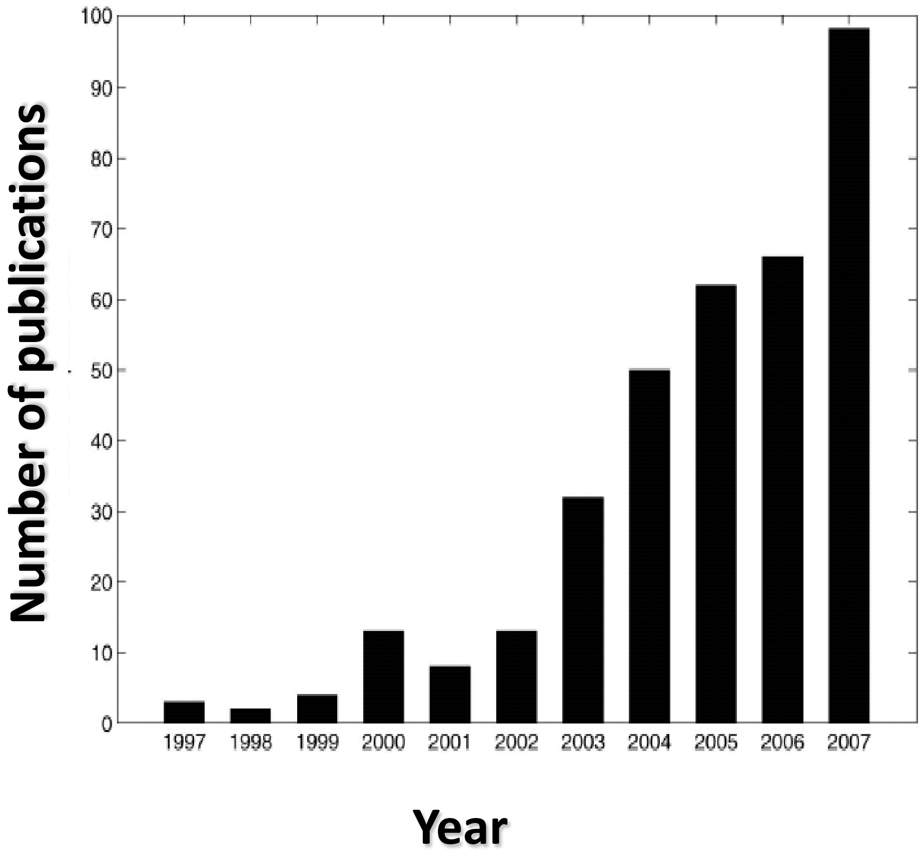


The model will not work well on new data

The problem is generally reduced by variable selection and use of simple methods

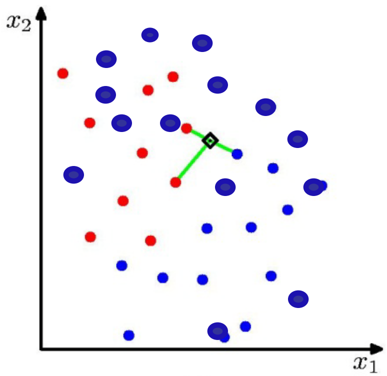
Class-imbalance: a "problem" for classification

- **Class-imbalanced data** = The number of samples in each class is not equal
- Most classifiers trained on imbalanced data
  - do not accurately predict the samples from the minority class
  - tend to classify all the samples in the majority class



Learning from Imbalanced Data – He and Garcia, 2009, *IEE Transactions on Knowledge and Data Engineering*

Example on the effect of class prevalence on the accuracy of a classifier (null case)



(a)

3-Nearest Neighbor classifier

No difference between classes (null case)

The classifier is <u>uninformative</u>

if $PA_1 = 1 - PA_0$

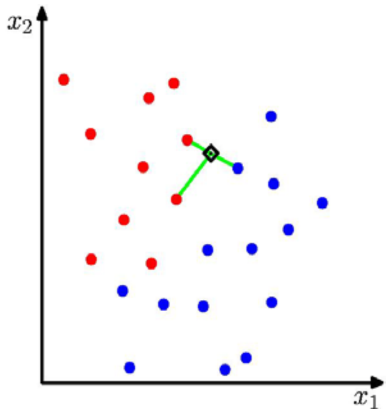
$P(\hat{Y}=1 \mid Y=1) = P(\hat{Y}=1 \mid Y=0)$

Test set

	Bad=80 Good=20
Training set: Bad=50 Good=50	
Training set: Bad=80 Good=20	

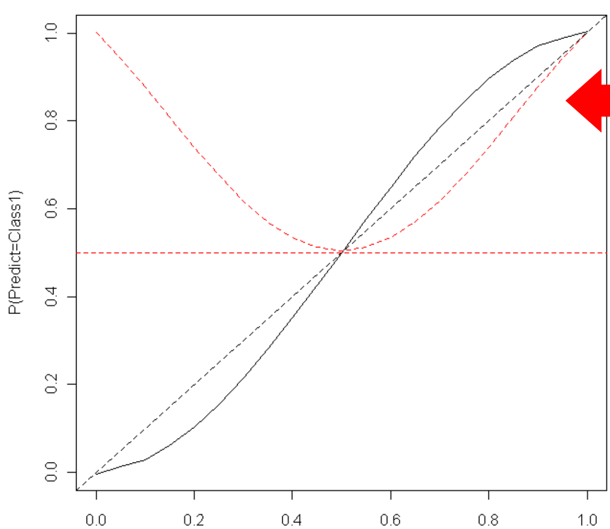
PA: predictive accuracy

Why does it happen?



p_i=proportion of samples in class i

$P(Pr_1|H_0)$=?



p_1 in training set

Overall PA if $p_{1\ Train} = p_{1\ Test}$

Overall Predictive Accuracy if $p_1 = p_2$ in test set

Theoretical values and expected behavior

Easy to calculate for 3-NN (using hypergeometric distribution)

Variable selection increases the bias towards the majority class

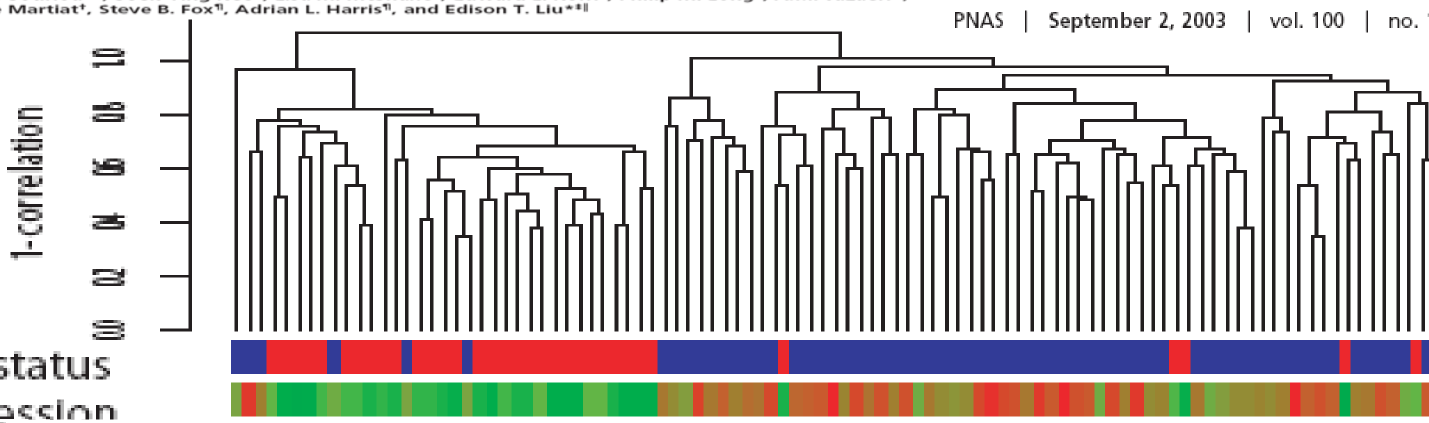
Consequences of class imbalance for other classifiers can be more difficult to understand …

Does it happen also when there are some differences between the classes and using real data?

Breast cancer classification and prognosis based on gene expression profiles from a population-based study

Christos Sotiriou*†, Soek-Ying Neo‡, Lisa M. McShane§, Edward L. Korn§, Philip M. Long‡, Amir Jazaeri*, Philippe Martiat†, Steve B. Fox¶, Adrian L. Harris¶, and Edison T. Liu*‡||

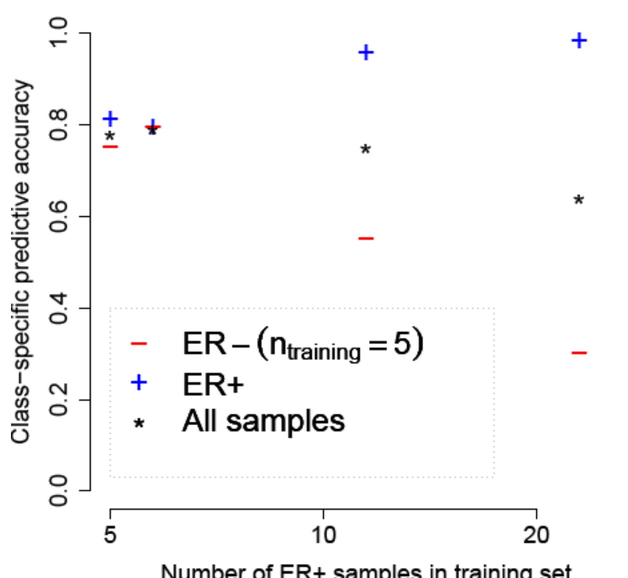
PNAS | September 2, 2003 | vol. 100 | no. 18 | 10393–10398



7650 clones
99 samples

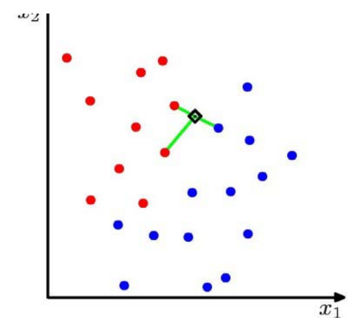
ER+ / ER-
65 / 34

Raw data



Normalized genes

5 ER+ / 5 ER- 10 ER+ / 5 ER- 20 ER+ / 5 ER-



3-NN classifier, 40 genes with largest t-statistic, missing values replaced by 0

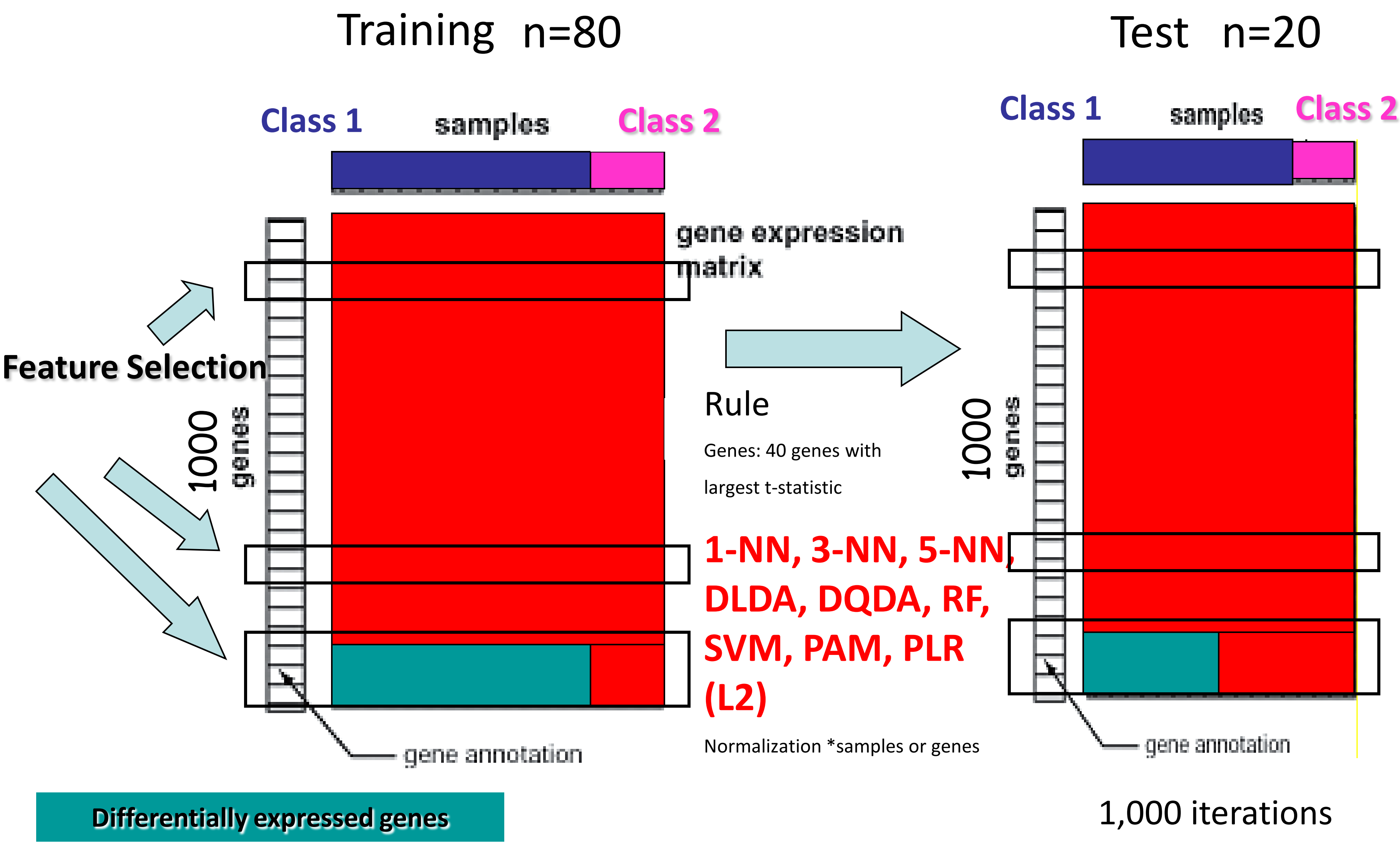
Test set: 20 ER+ / 20 ER-

Training set: 5 ER+ / 5 ER-, 10 ER+ / 5 ER-, 20 ER+ / 5 ER-

Our initial questions

- Does the high-dimensionality of the data further exacerbate the class-imbalance problem?
- Are there any classification methods that are more robust than others?
- Are the methods commonly used to deal with the class-imbalance problem effective if the data are high-dimensional?
- Can we get some theoretical insights on the nature of the class-imbalance problem?
- …

Simulations

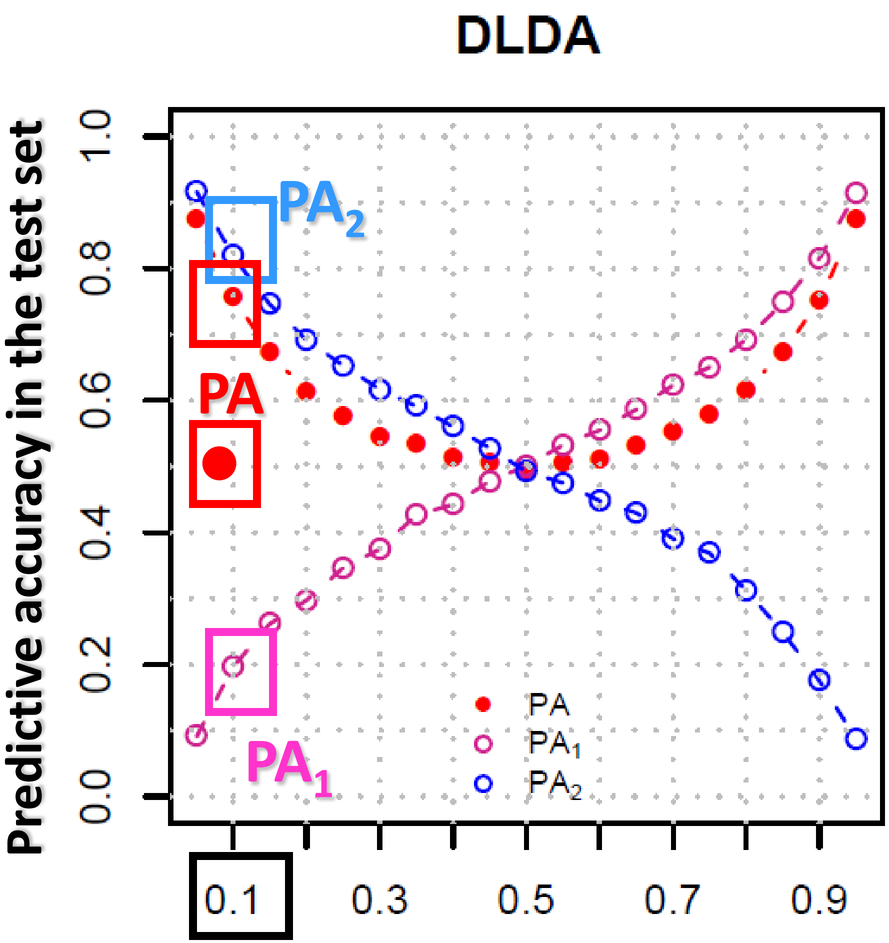


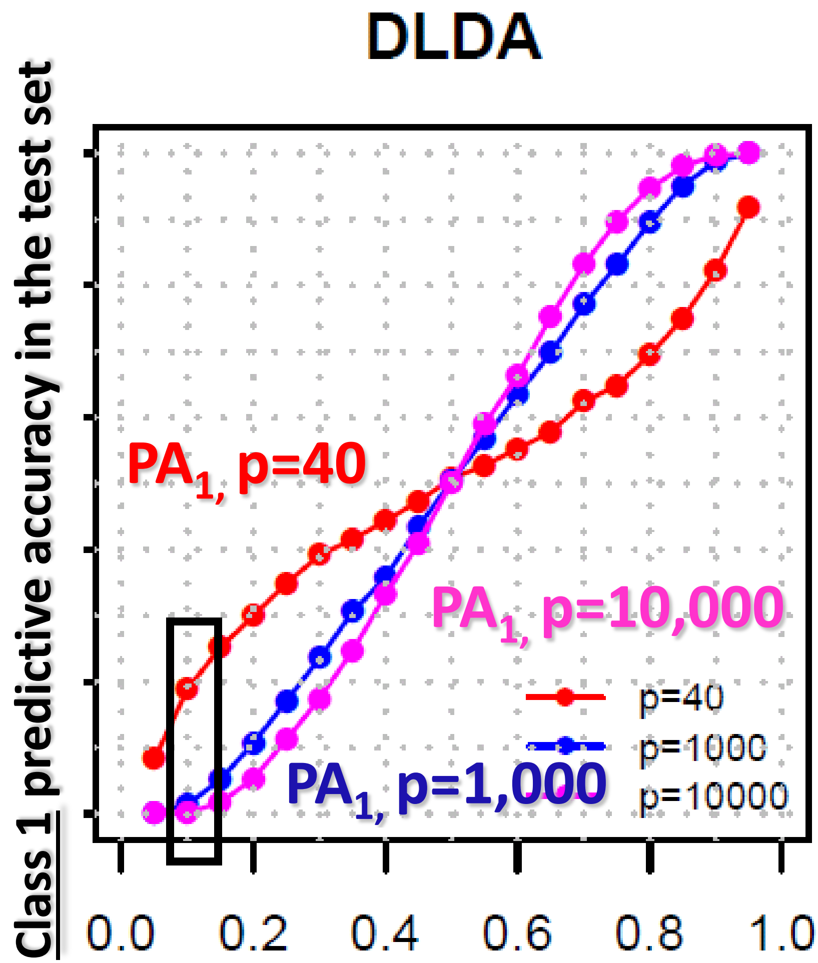
20 DE genes, magnitude of difference: g_{1i}~id N(1, 1), g_{2i}~id N(0, 1)

Null Hypothesis

$p=40$, $n_{train}=80$, same class imbalance in training and test set

What happens if the number of variables increases and we use the 40 most different variables for classification?

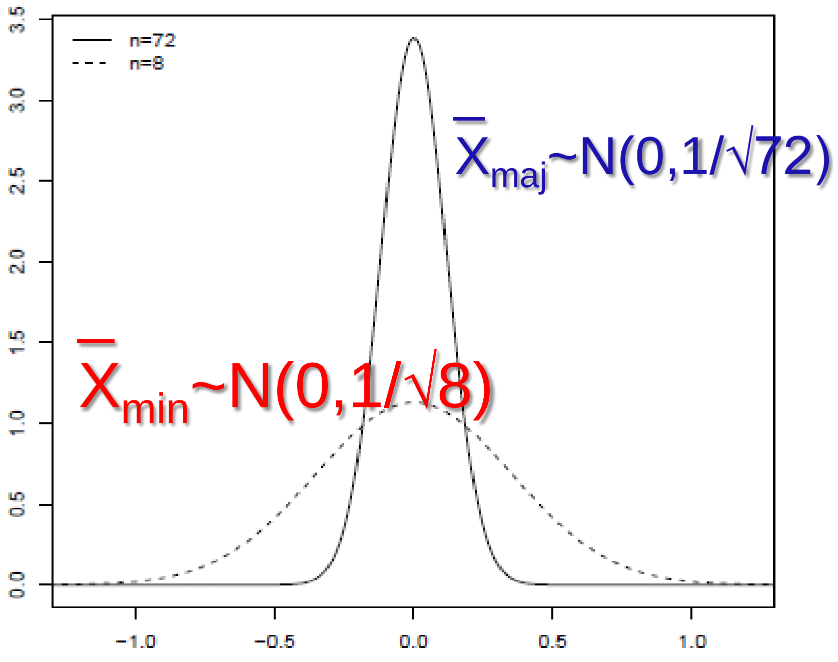




Proportion of samples from Class 1 in the training set

Sampling variability and class imbalance

Null case: distribution of the sample means

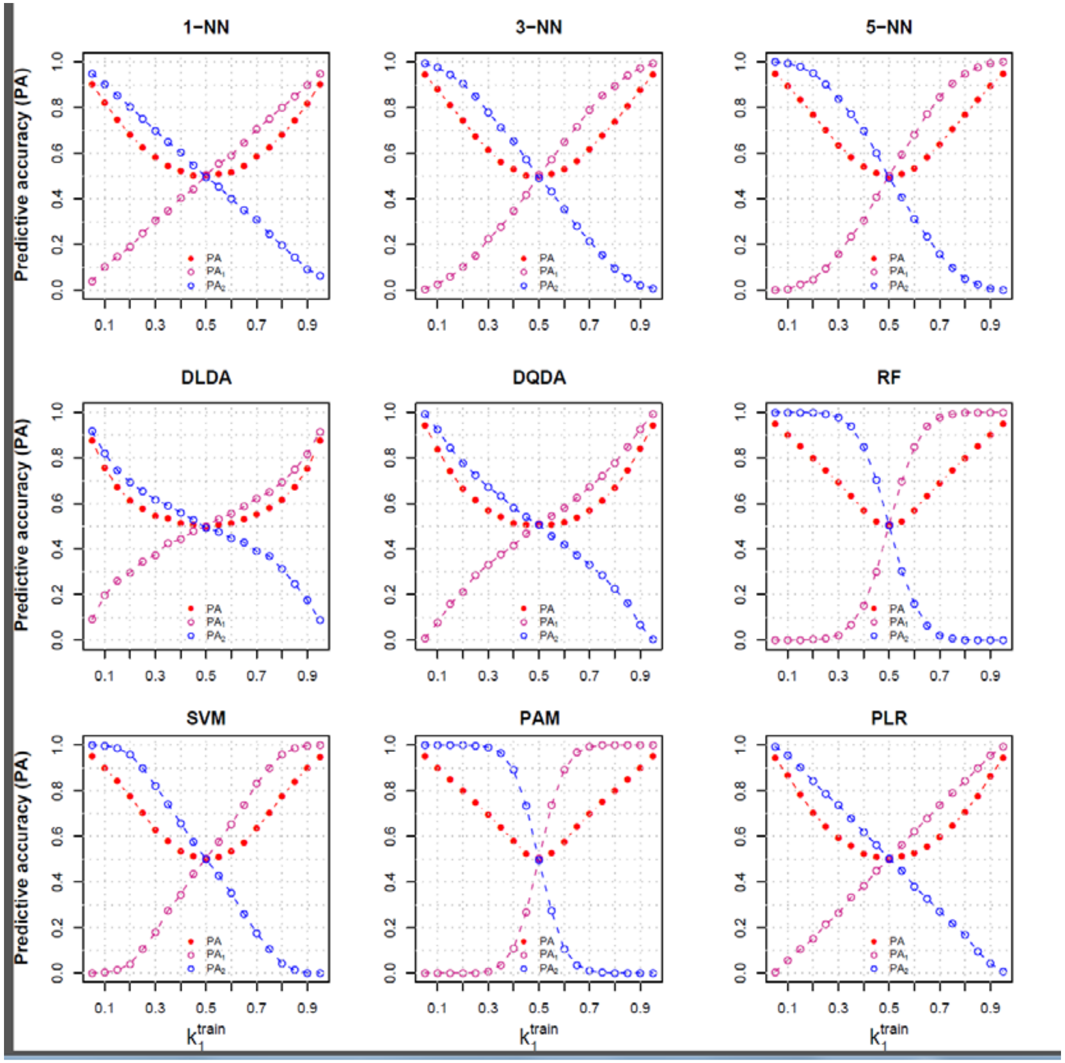


$X_{maj} \sim N(0,1)$; n=72

$X_{min} \sim N(0,1)$; n=8

Null Case Results

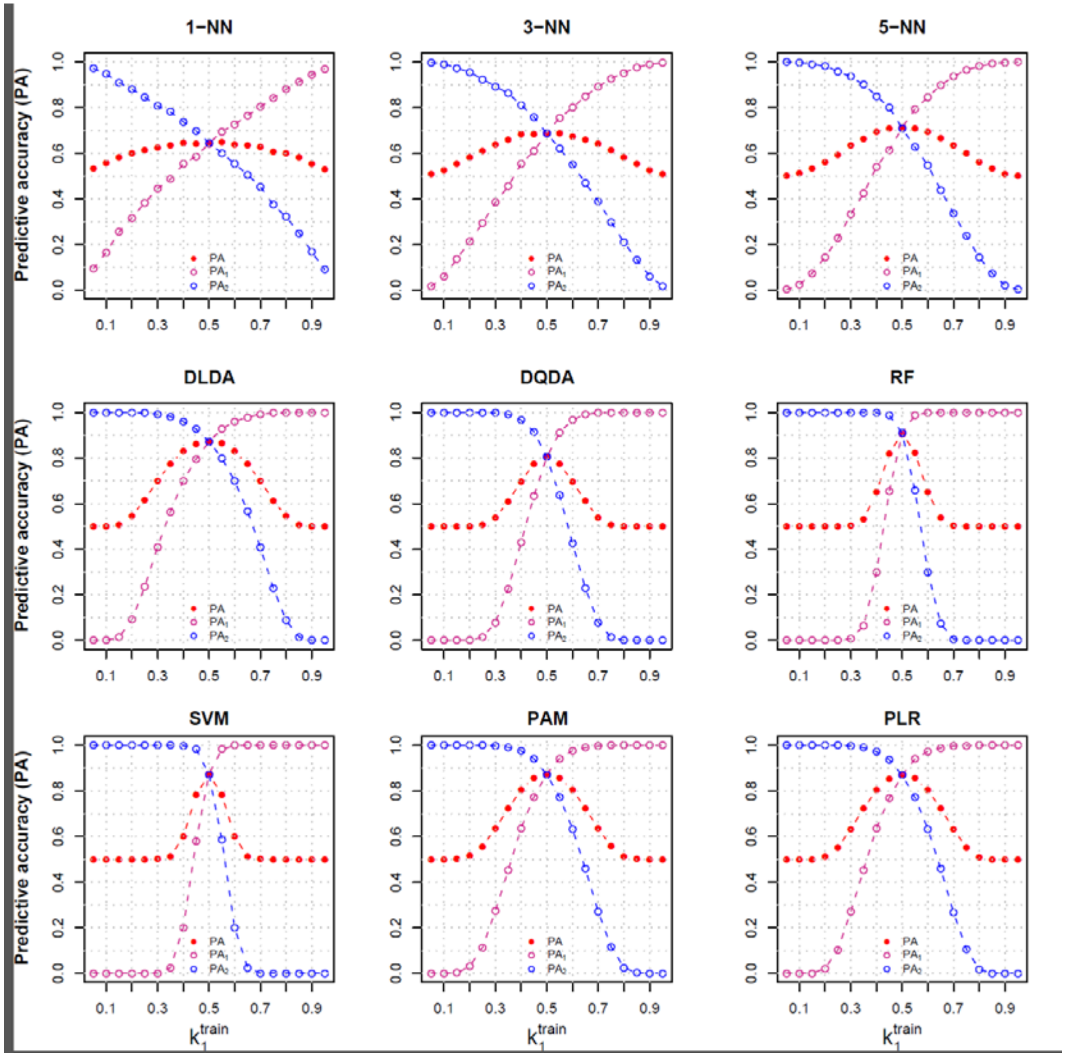
(**p=1000, G=40**, n_{train}=80, same class imbalance in training and test set)



All the classifiers are non-informative (PA_1=1-PA_0)

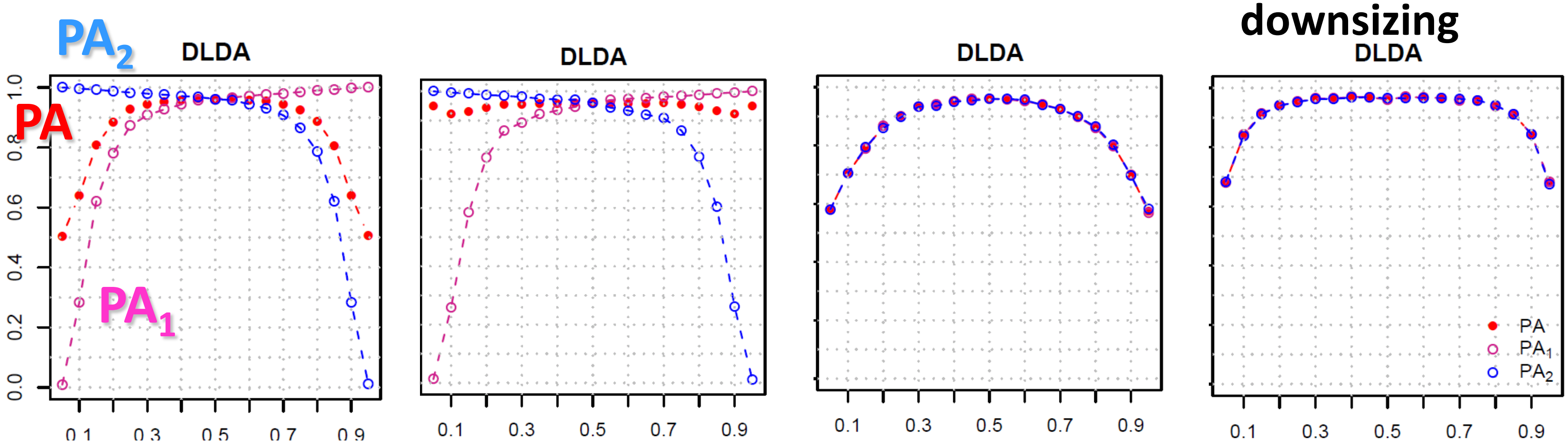
Alternative Case Results – moderate class differences

(**p=1000, G=40**, delta=0.7, n_{train}=80, same class imbalance in training and test set)

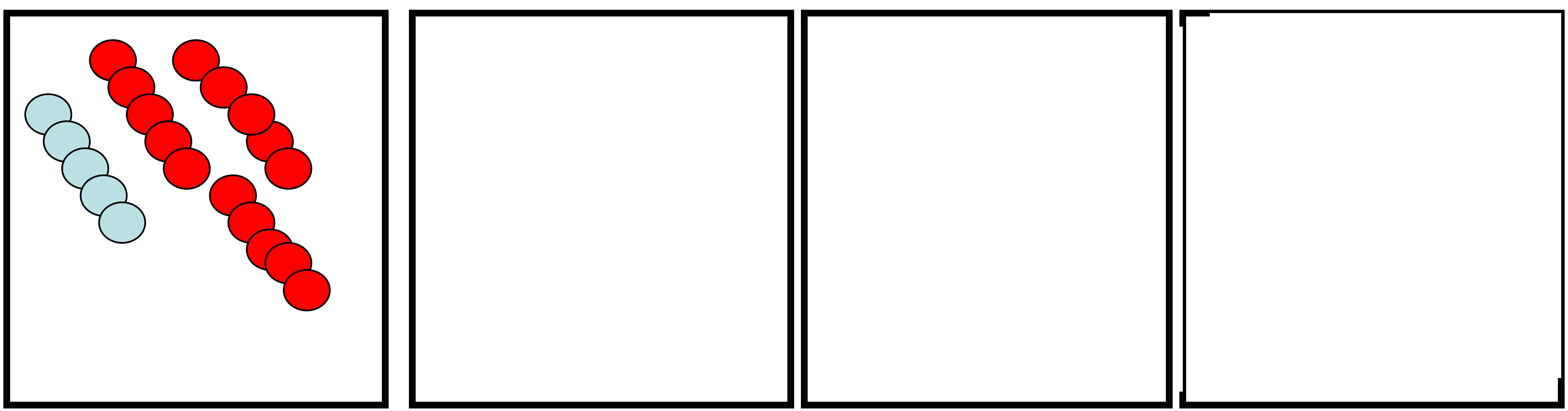


High-dimensionality	☹Additionally biases classification towards majority class for most classifiers. Mostly due to large sampling variability of the minority class
Variable selection	☺ Improves the PA ☹Additionally biases classification towards majority class for some classifiers (k-NN) 😐 Most variable selection methods share the same problems seen here (using t-test)
Classification methods	😐 DLDA, DQDA, PLR, RF ☹ k-NN, PAM, SVM
Matching the prevalence in training and test set	😐 Does not remove the problem
Variable normalization	☹ Further increases the bias if $p_{train} \neq p_{test}$ 😐 $p_{train} \neq p_{test}$

No correction **Oversampling** **Downsizing** **Multiple downsizing**

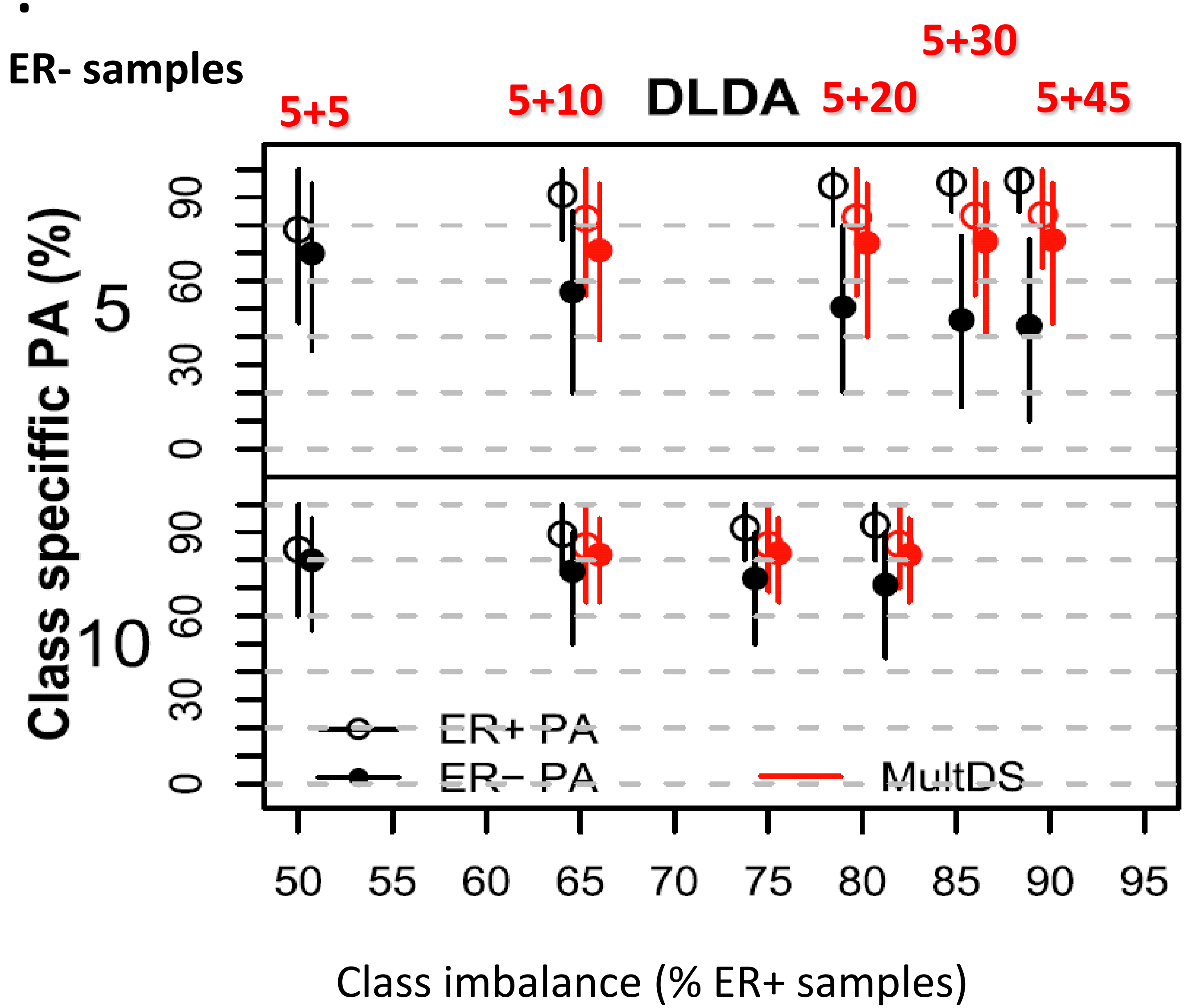


Proportion of samples from Class 1 in the training set



Alternative hypothesis: the classes are different, and prediction using the balanced training set is easy

Does multiple downsizing work also on real data?

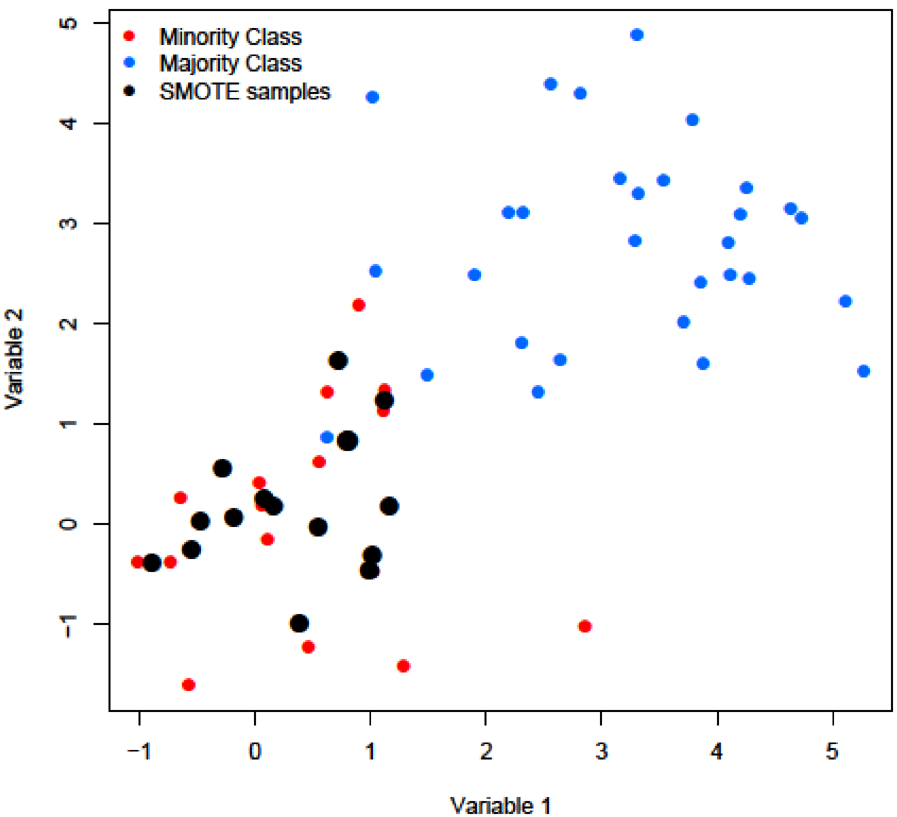


Variable selection	☺ Improves the PA ☹Additionally biases classification towards majority class for some classifiers 😐 Most variable selection methods share the same problems seen here (t-test)
Classification methods	😐 DLDA, DQDA, PLR, RF ☹ k-NN, PAM, SVM
Matching the prevalence in training and test set	😐 Does not remove the problem
Variable normalization	☹ Further increases the bias if $p_{train} \neq p_{test}$ 😐 $p_{train} \neq p_{test}$
Solutions	**😐/☹ Built-in solutions (SVM, RF, PAM)** **😐 Downsizing** **☹ Oversampling** **☺ Multiple downsizing**

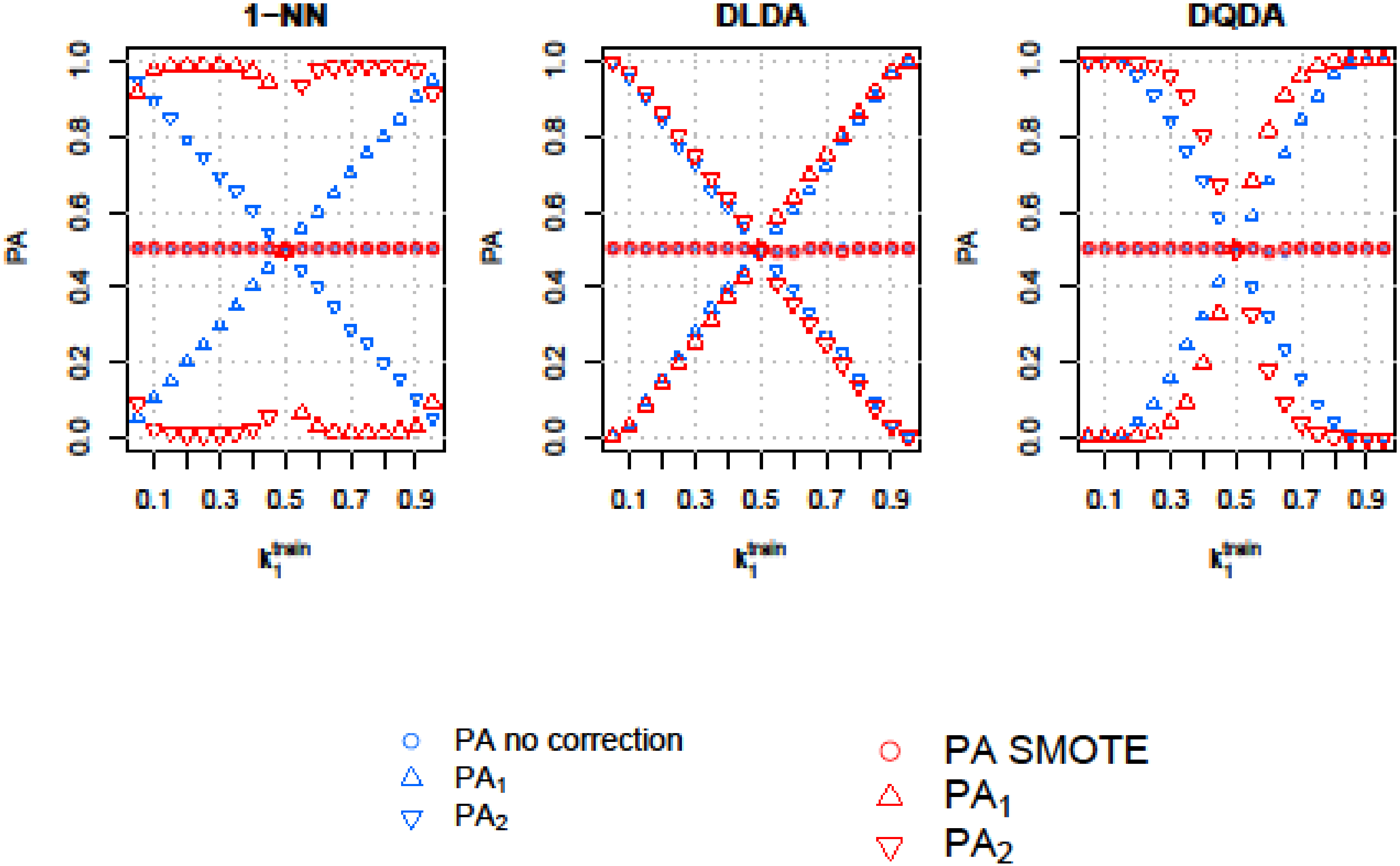
SMOTE

Synthetic Minority Over-sampling TEchnique

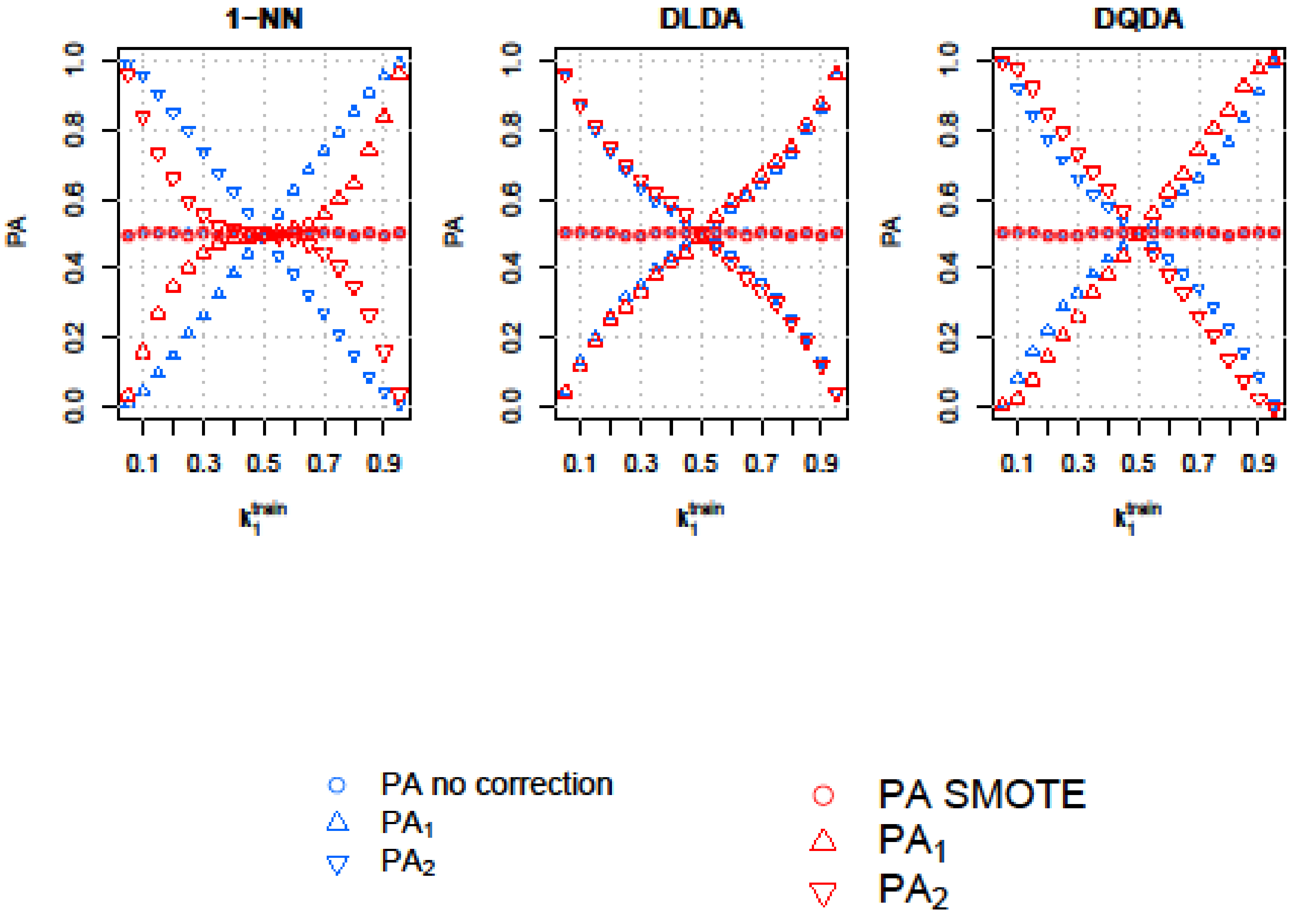
$$\mathbf{x}^{SMOTE} = \mathbf{x} + u * (\mathbf{x}^{NN} - \mathbf{x})$$



Does SMOTE work? (null case, p=1000)



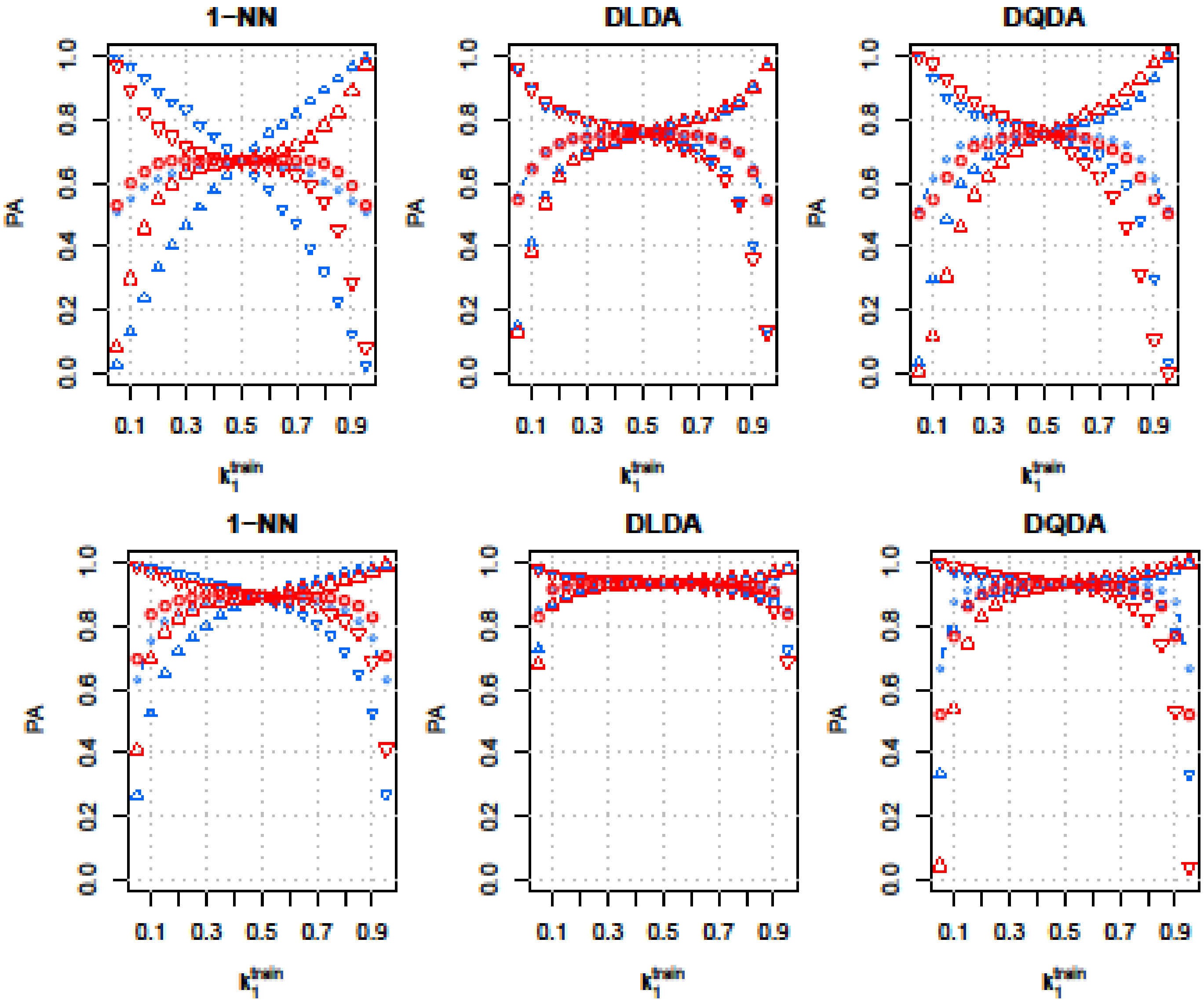
Does SMOTE work if we perform variable selection? (null case, p=1000, 40 selected)



Does SMOTE work if there is a difference between classes? (alternative case, p=1000, 40 selected)

Small differences

Large differences



Could we expect it?

Theoretical results

- SMOTE does not change the expected value of the minority class

$$E(X^{SMOTE}) = E(X)$$

- The variance of the minority class is reduced if SMOTE is used to balance the class-distribution

$$var(X^{SMOTE}) = \frac{2}{3} var(X)$$

- When p is large new samples are expected to be closer (in terms of Euclidean distance) to SMOTE samples than to original samples

$$E(d(X^{test}, X^{original})) > E(d(X^{test}, X^{SMOTE}))$$

$$2p \cdot var(X) > 2p\frac{5}{6} \cdot var(X)$$

It does not affect much the classifiers that rely on mean values (DLDA)

It negatively affects the classifiers that use class-specific variances (DQDA). Care with variable selection methods!

If data are high-dimensional: classifiers that base their classification rule on the Euclidean distance tend to classify most samples in the MINORITY class (k-NN without variable selection)

Other theoretical results

- SMOTE introduces correlation between some samples
- $$\rho\left(X_j^{SMOTE_1}, X_j^{SMOTE_2}\right) = \begin{cases} 3/4 & \text{If they "share" two original samples} \\ 3/8 & \text{If they "share" one of the original samples} \\ 0 & \text{Otherwise} \end{cases}$$

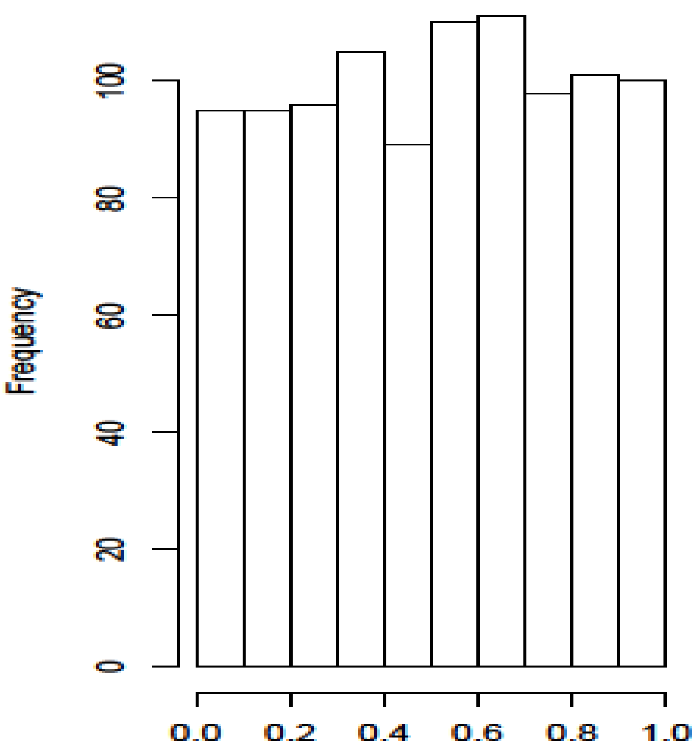
$$\rho(X_j^{SMOTE_1}, X_j^{o}) = \begin{cases} \frac{\sqrt{3}}{2\sqrt{2}} & \text{If the original sample was used to generate the SMOTE sample} \\ 0 & \text{Otherwise} \end{cases}$$

Can we still reliably use classification methods and variable selection methods that <u>assume independence between samples</u> (discriminant analysis methods, PLR, two sample t-test, …) ?

Effect of SMOTE on two-sample t-test P-values

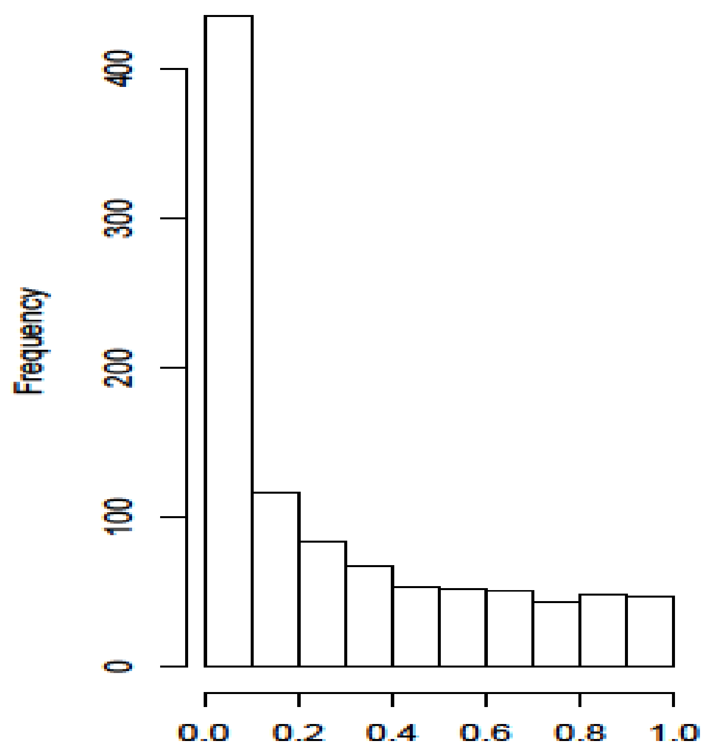
Null *case*, p=1000, n=10+90

Original data



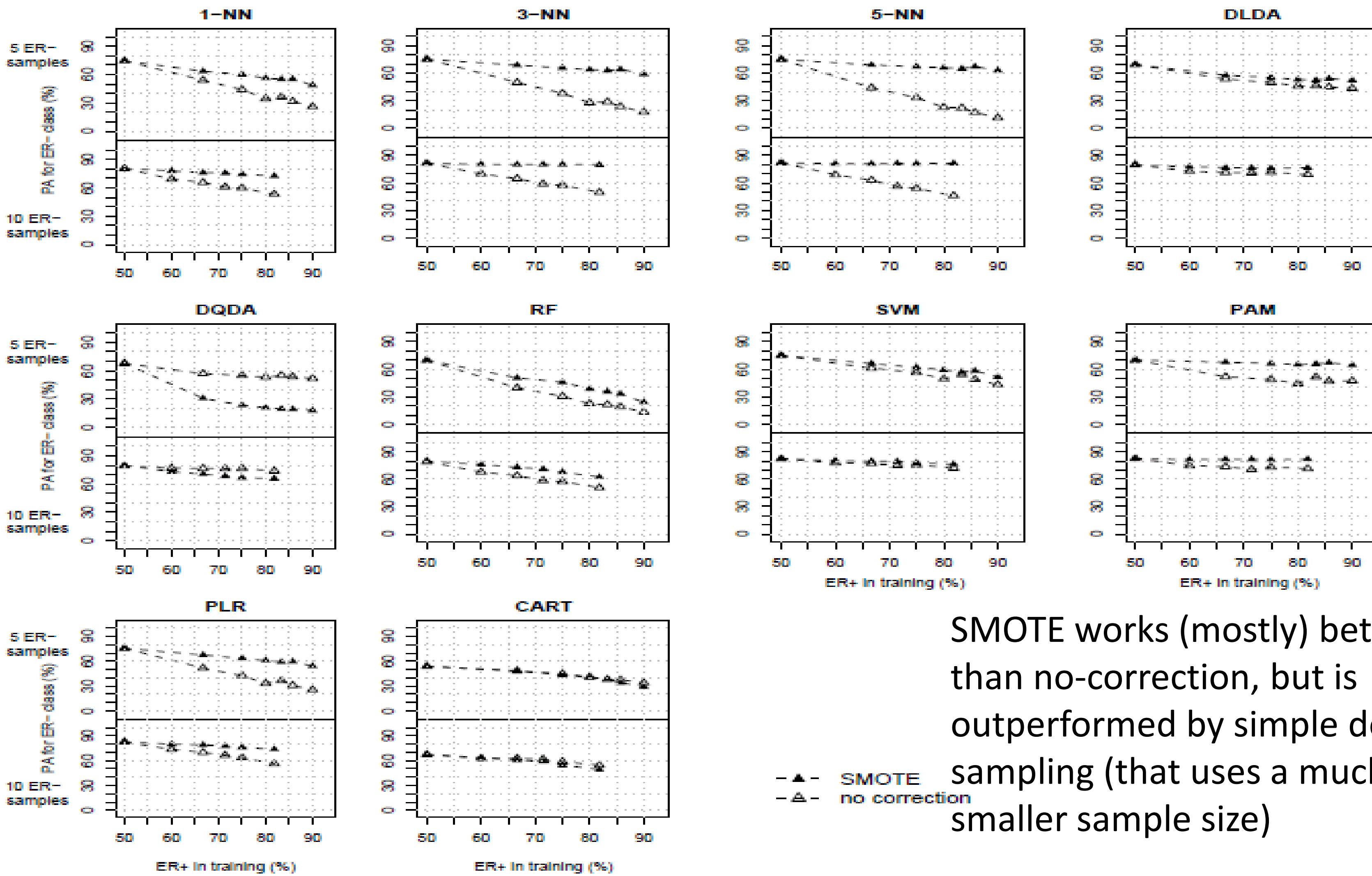
P-values

After “SMOTE-augmenting” the data



P-values

Do we see the same things on real data?



SMOTE works (mostly) better than no-correction, but is outperformed by simple down-sampling (that uses a much smaller sample size)

Summary of the performance of SMOTE on low and high-dimensional data

	low-dimensional data		high-dimensional data, without variable selection			high-dimensional data, with variable selection		
Classifier	NC	SMOTE	NC	CO	SMOTE	NC	CO	SMOTE
1-NN		↑		≈	↓		≈	↑
5-NN		↑		↑	↓		↑	↑
DLDA		≈		≈	≈		≈	≈
DQDA		≈		≈	↓		≈	↓
RF		↑		↑	↑		↑	≈
SVM		↑		↑	≈		↑	≈
PAM		↑		↑	≈		↑	↑
PLR-L1		↑		↑	≈		↑	≈
PLR-L2		↑		↑	≈		↑	≈
CART		↑		≈	≈		≈	≈

- NC: no correction
- CO: classification cut-off calibration

Conclusions

- High-dimensionality of data exacerbates the class imbalance problem
- Some classifiers are less sensitive than others to class imbalance
  - DLDA seems to work well also in this setting if the class-imbalance is not too extreme
- Solutions
  - Down-sizing works surprisingly well (fewer data better than imbalanced data!) but wastes a lot of data
    - Combination of down-sized classifiers works very well
  - Over-sampling is generally a bad idea, SMOTE is not improving over simple downsampling
  - Change
  - Cut-off calibration (ongoing work)
    - Worked reasonably well for RF, PLR and k-NN

- Normalization of variables exacerbated the problems

Ongoing work

- Further explorations for the penalized logistic regression methods (PLR-L1 and PLR-L2)
- Evaluation and adaptation of boosting methods for high-dimensional (and class-imbalanced) data
- cartHD: R package that includes functions to fit classification trees with binary outcomes and to correct the analysis for the class-imbalance problem
  - Multiple downsizing, boosting, cross-validation, … with fast implementation for high-dimensional data and repeated estimation

Some references

- Japkowicz N, Stephen S: **The class imbalance problem: A systematic study**. *Intell Data Anal 2002*, **6**(5):429-449.
- He H, Garcia EA: **Learning from imbalanced data**. *IEEE Trans Knowledge and Data Eng 2009,* **21**(9):1263-1284.

- Our published work on the topic
  - **Class prediction for high-dimensional class-imbalanced data.** *BMC Bioinformatics* 2010, **11**:253
  - **Impact of Class-Imbalance on Multi-Class High-Dimensional Class Prediction.** *Metodoloski zvezki – Advances in Methodology and Statistics 2012,* **9:** 25-45.
  - **SMOTE for high-dimensional class-imbalanced data.** *BMC Bioinformatics* 2013, **14**:106
  - **Improved shrunken centroid classifiers for high-dimensional class-imbalanced data.** *BMC Bioinformatics* 2013

Threshold calibration

Develop the rule

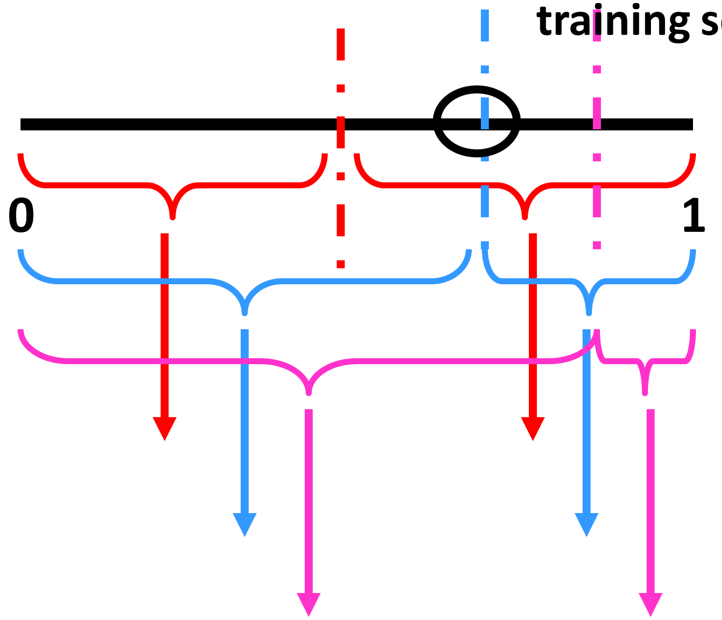
Number

Estimated on the training set (by CV)

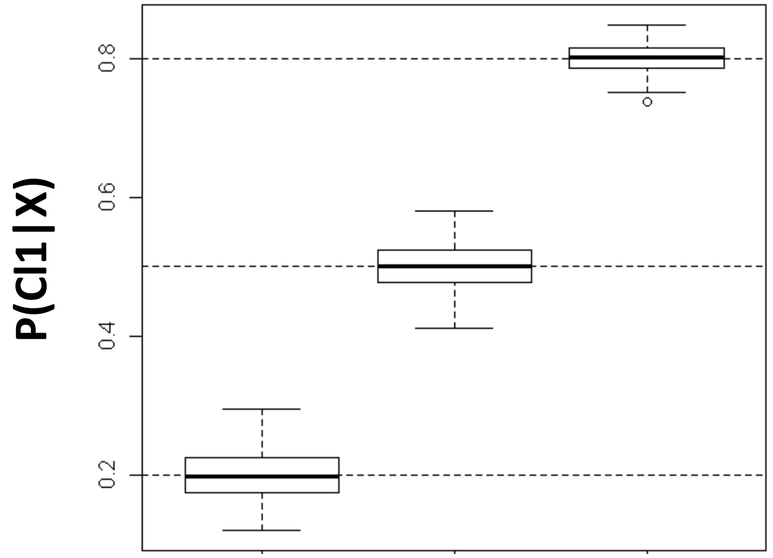
PLR: P(Class 1| X)

RF: proportion of trees that classify the sample in Class 1

k-NN: proportion of NN from Class 1



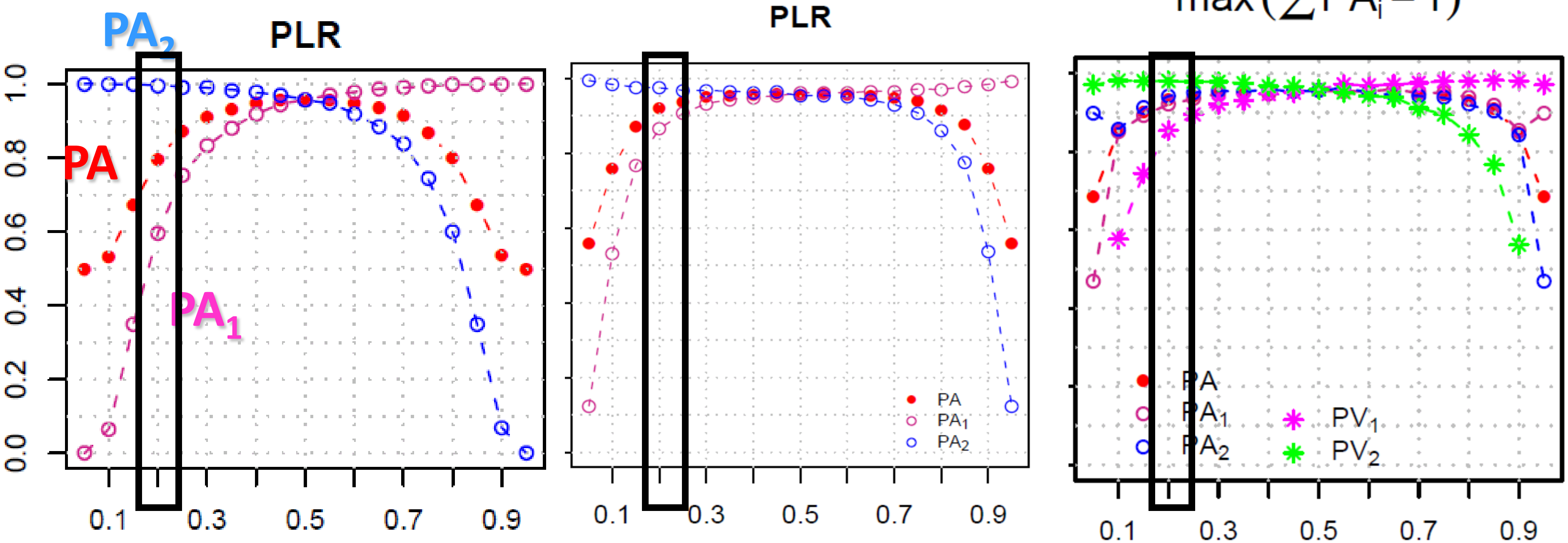
Class 2 Class 1



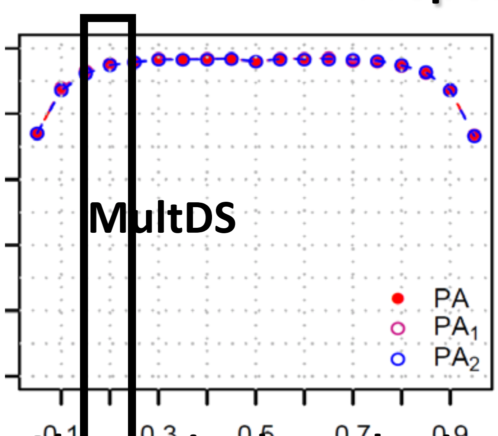
Cl1 20% 50% 80%

Thr=0.50 **Thr=proportion of samples from Class 1** **Estimated Thr**

$$\max\left(\sum PA_i - 1\right)$$



Proportion of samples from Class 1 in the training set



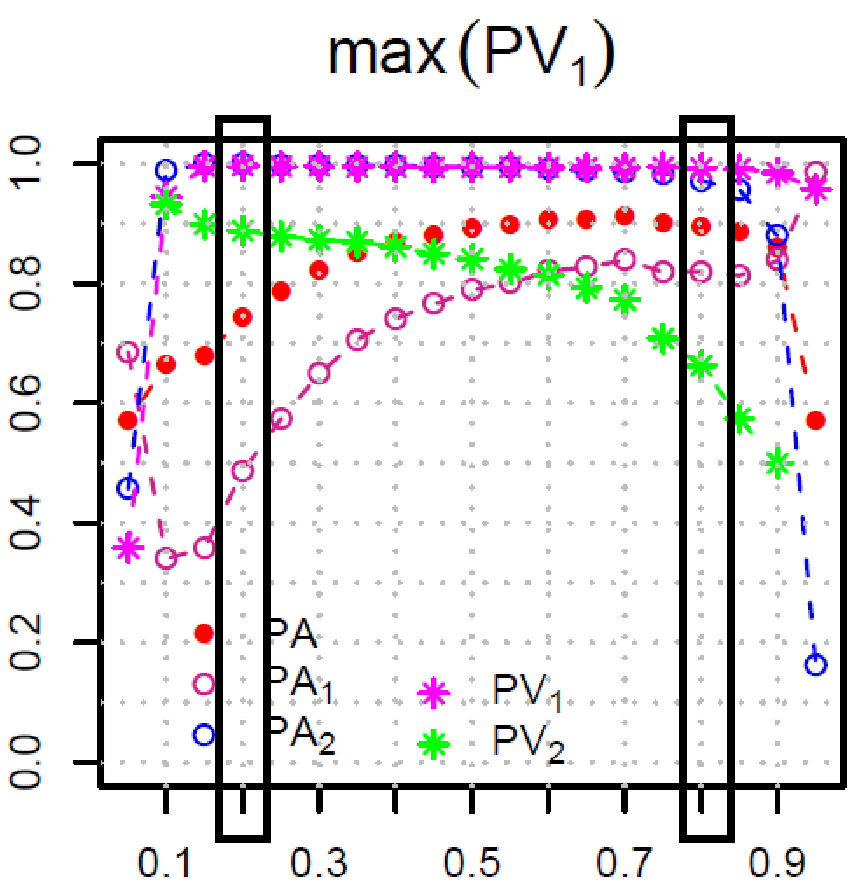
Predictive value vs predictive accuracy

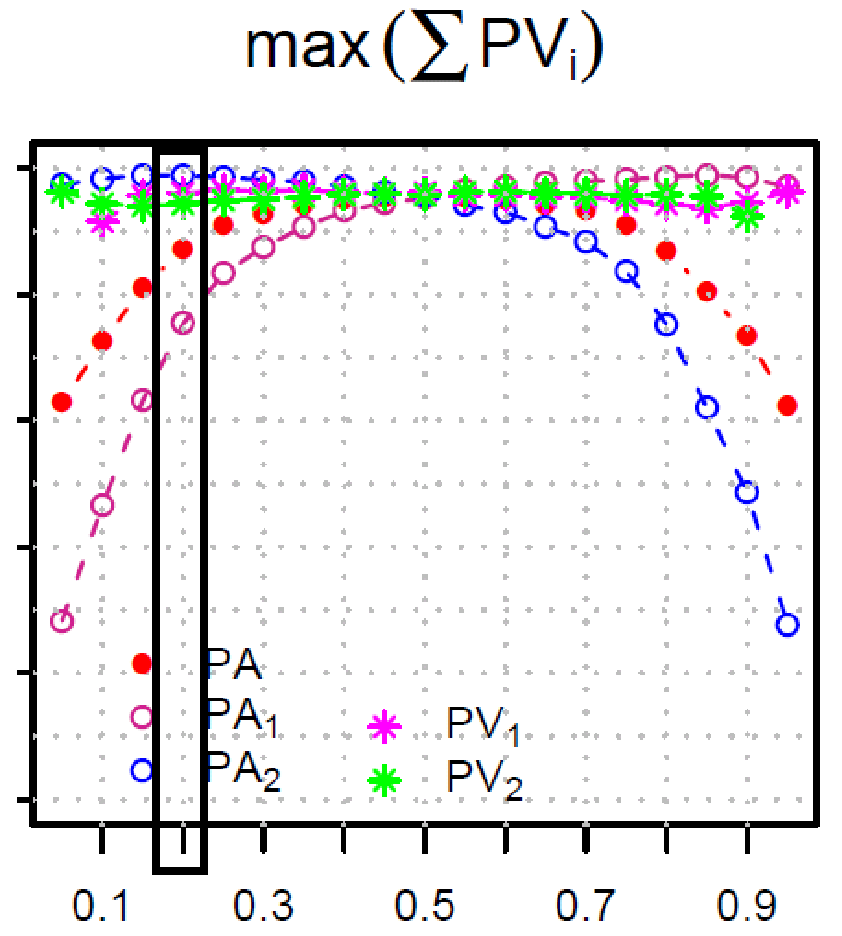
PA_1=P(Predicted $Class_1$| True $Class_1$)

PV_1=P(True $Class_1$| Predicted $Class_1$)

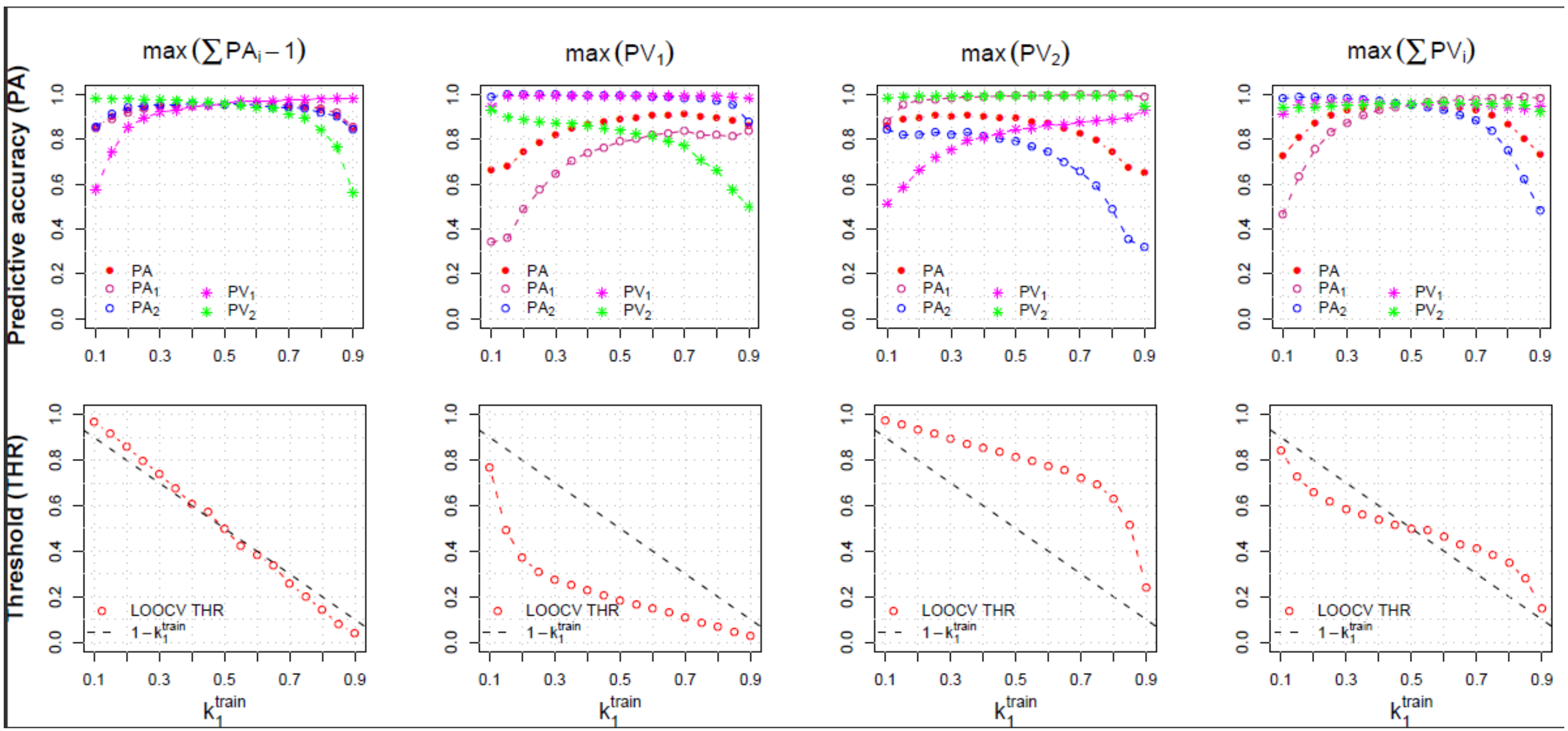
Alternative hypothesis: the classes are different, and prediction using the balanced training set is easy

Other functions that can be used to estimate the threshold





Proportion of samples from Class 1 in the training set



max (∑PAᵢ – 1)
max (PV₁)
max (PV₂)
max (∑PVᵢ)
Predictive accuracy (PA)
Threshold (THR)
PA
PA₁
PA₂
PV₁
PV₂
LOOCV THR
1 – k₁^train
k₁^train

How to evaluate the performance of a classifier

- Classification error
  - A sample is classified in a class to which it does not belong
    - $g(X) \neq Y$
    - Predictive accuracy (PA)=% of correctly classified samples
      - careful interpretation!
  - In a two-class problem, using the terminology from diagnostic tests ("+"=diseased, "-"=healthy)
    - PA_+ =Sensitivity = P(classified +| true +)
    - PA_- =Specificity = P(classified -| true -)

      } Class-specific predictive accuracies
    - Positive predictive value = P(true +| classified +)
    - Negative predictive value = P(true - | classified -)
  - It is important to report all 4 when classes are imbalanced