On the Window Size for Classification in Changing Environments

Ludmila I. Kuncheva[1] and Indrė Žliobaitė[2]
[1]School of Computer Science, Bangor University, UK
e-mail: l.i.kuncheva@bangor.ac.uk
[2]Faculty of Mathematics and Informatics, Vilnius University, Lithuania
e-mail: indre.zliobaite@mif.vu.lt

Abstract

Classification in changing environments (commonly known as concept drift) requires adaptation of the classifier to accommodate the changes. One approach is to keep a moving window on the streaming data and constantly update the classifier on it. Here we consider an abrupt change scenario where one set of probability distributions of the classes is instantly replaced with another. For a fixed 'transition period' around the change, we derive a generic relationship between the size of the moving window and the classification error rate. We derive expressions for the error in the transition period and for the optimal window size for the case of two Gaussian classes where the concept change is a geometrical displacement of the whole class configuration in the space. A simple window resize strategy based on the derived relationship is proposed and compared with fixed-size windows on a real benchmark data set data set (Electricity Market).

Keywords: concept drift; streaming data; training sample size; moving window size

1 Introduction

Concept change (concept drift) is a major setback in real-life pattern recognition problems. A classifier trained on the currently available data set may fail if data distributions change abruptly or migrate gradually in the course if its use. The solution is to keep updating the classifier with the new-coming data, provided that the true labels of the data become available after the classification. Various aspects of classification in the presence of concept drift have been discussed over the years nesting within different research disciplines and using different terminologies. In a quest to establish and streamline the concept drift research, special issues of reputable journals as well as international research workshops have been devoted to this theme [5], [11], [18].

There are two general approaches to updating the classifier when changes in the probability distributions are expected:

- *Detect and Retrain.* Train a classifier on an initial training sample and use it for the streaming data. Monitor the classification performance and the data stream. As soon as change is detected, retrain the classifier on a new set of training data that represents the current distribution.
- *Constant Updates.* Use a moving window containing the latest N observations in the data stream. Retrain the classifier using the latest window as the training data. The window can be of fixed or variable size. The window size is crucial for this approach to work. With a small window, the classifier will be responsive to the changes but may be chronically undertrained. On the other hand, a classifier using a large window will be inert but well suited to static bouts in the probability distributions.

Window size has been discussed at length in relation to change detection [1], [2], [6], [8]–[10], [13], [16], [17]. The size reduction of the window after the change detection is typically guided by heuristics, varying from exhaustive backward search through a host of past of data [2], [8], golden section search through possible cut-off points within the current window [10], pre-defined reduction rate [9], to constructing a window that starts at the suspected onset of the change [6]. Theoretical results have been derived for splitting a *change-detection window* [2], in the form of bounds on the false positive and false negative detections. On the other hand, theoretical results that relate the *training* window size with the online classification accuracy in the presence of concept drift are scarce [19].

This paper can be regarded as a step in the direction of determining an optimal window size in relation to the chosen classifier model and the properties of the streaming data.

The rest of the paper is organized as follows. Section 2 explains the relationship between training sample size and classifier accuracy. The general set-up for our study and the derivation of the optimal window size are given in Section 3. Section 4 illustrates the relationships found on a special case of two Gaussian classes and one feature.

Simulation results are also presented there. In Section 5 we propose a simple window resizing procedure and compare it with fixed-size windows using the benchmark Electricity Market dataset. Section 6 concludes the study.

2 Design sample size and classification accuracy

The relationship between the design sample size and the accuracy/error of the classifier has been a topic of interest for decades [4], [14], [15]. Its relevance to our study lies in the fact that the classifier in the online classification of streaming data is trained on a fixed, possibly quite small, training window. To achieve a responsive and at the same time reasonably accurate classifier, a compromise with the window size has to be sought.

Consider a classification problem in the n-dimensional real space $\Re^n$. Let C be the chosen classifier whose parameters are calculated from a sample of size N[1]. Denote by $\epsilon^N(C)$ the theoretical error achievable by C. Let $\epsilon(C)$ be the asymptotic error rate of C obtained as $\epsilon(C) = \lim_{N\to\infty} \epsilon^N(C)$. Fukunaga and Hayes [4] show that, for any parametric classifier C with a set of parameters $\boldsymbol{\theta}$, regardless of the types of the probability density functions (pdfs), the classification error can be expressed approximately as

$$\epsilon^N(C) \approx \epsilon(C) + \frac{1}{2}\mathrm{tr}\left\{\frac{\partial^2\epsilon(C)}{\partial\boldsymbol{\theta}^2} E\left[\Delta(\boldsymbol{\theta})\Delta(\boldsymbol{\theta})^T\right]\right\}, \tag{1}$$

where $\Delta(\boldsymbol{\theta}) = \boldsymbol{\theta} - \boldsymbol{\theta}^{(N)}$, with $\boldsymbol{\theta}^{(N)}$ being the estimate of $\boldsymbol{\theta}$ from a data sample of size N. The authors state that for many estimators, the second term can be factorised so that

$$\epsilon^N(C) \approx \epsilon(C) + g(N)h(C), \tag{2}$$

where $h(C)$ is a function that depends on the classifier type and the true distributions but not on N, and $g(N)$ is a function of the sample size N. The expression is derived by expanding $\epsilon^N(C)$ in a Taylor series about the true values of $\boldsymbol{\theta}$, up to second order terms. Equation (2) comes as the expectation of $\epsilon^N(C)$ across all samples of size N. Then $h(C) = \frac{1}{2}\mathrm{tr}\left\{\frac{\partial^2\epsilon(C)}{\partial\boldsymbol{\theta}^2}K(\boldsymbol{\theta})\right\}$, where the function K depends on the way $\boldsymbol{\theta}^{(N)}$ are computed. For two Gaussian classes, (2) reduces to

$$\epsilon^N(C) \approx \epsilon(C) + \frac{1}{N}f(C), \tag{3}$$

Consider a problem with two n-dimensional Gaussian classes with identical covariance matrices. Denote by δ the Mahalanobis distance between the class means. Assume that the means and the covariance matrix are calculated from N samples from each of the two classes. The *linear discriminant classifier* (LDC) is Bayes-optimal for this case. Its error can be expressed as [4]

$$\epsilon^N(\mathrm{LDC}) \approx \Phi\left(-\frac{\delta}{2}\right) + \frac{1}{N\sqrt{8\pi}\,\delta}\left[\left(1+\frac{\delta^2}{4}\right)n - 1\right]\exp\left(-\frac{\delta^2}{8}\right), \tag{4}$$

where Φ is the cumulative distribution function of the standard normal distribution. When the covariance matrices are the identity matrices, (4) gives the error of the *nearest mean classifier* (NMC), where δ is just the Euclidean distance between the two class means.

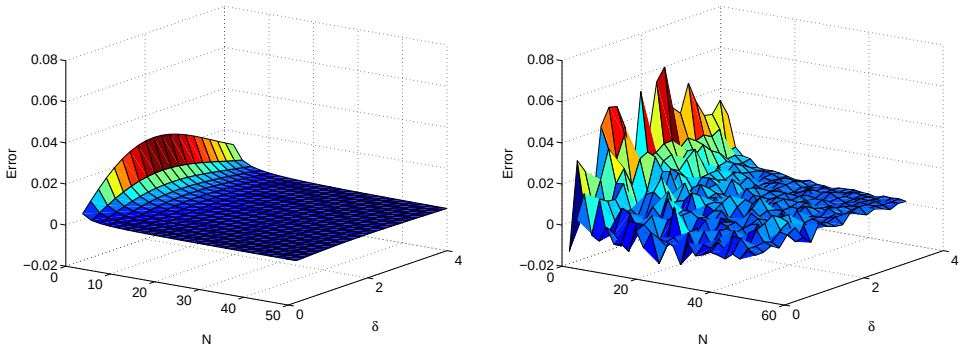
Figure 1 (a) shows the calculated surface of $f(C)/N$ as a function of δ and N, and Figure 1 (b) shows the result from a simulation of two 1-d Gaussians with distance between the means δ. The plots match in shape and position, with the calculated error being slightly lower than the simulated error. Interestingly, the adverse effect of small sample sizes peaks at medium values of δ. Highly overlapping or highly separable classes receive smaller addition to the error for the same training size N.

3 Optimal window size

3.1 General problem set-up

Consider two sets of probability distributions, which we will call "sources", S_1 and S_2. Since we suspect that concept drift will occur, the chosen classifier model, C, is trained online using the window containing the latest N streaming observations. At a designated time, say T_1, source S_1 is replaced by source S_2 leading to a sudden change in the classification error. The moving window along the next N observations will be sampled from a mixture of distributions. The mixing proportions at time step T_1+i will be i/N from S_2 and $1-i/N$ from S_1, where $1 \le i < N$. A data set of size N will be used to train C, making progressive steps towards adapting the classifier to the distribution of source S_2. At time $T_1 + N$, the whole training sample would have come from S_2. Figure 2 displays the error progression of the classifier along the time axis for two values of N. The classifier with the larger window takes more time to recover from the change but the error will be smaller outside the transition period.

1. An important assumption here is that each parameter of C is calculated using N observations. In reality, the parameters may be calculated from training samples of different length, for example the class means.



(a) Calculated through equation (4) (b) Simulated

Fig. 1. Surface plot of the addition to the classification error as a function of the training set size, N, and the distance δ between the centres of the two Gaussian classes.

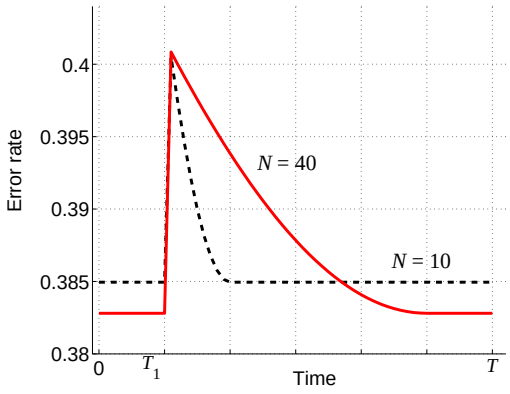


Fig. 2. Error progression for two values of the window size N and change at time T_1 (swap of sources S_1 and S_2).

We are interested in finding the best window size for a certain transition period containing the onset of the concept shift. Figure 3 shows the error curve during a transition period of length T with T_1 observations before the change and $T - N - T_1$ observations after the whole window has slid in source S_2. The area under the curve is a measure of the error during that period. We use the following notations

C_i	classifier C trained on data coming from source S_i, $i = 1, 2$
C_α^m,	classifier C trained on data coming from a mixture of source S_1 (mixing coefficient $1 - \alpha$) and S_2 (mixing coefficient α)
$\epsilon_i(C)$	error of classifier C if the class pdfs came from source S_i, $i = 1, 2$
$f_i(C)$	the numerator of the additional term in equation (3) for classifier C and class pdfs from source S_i, $i = 1, 2$

The mean error of a classifier C trained on a moving window of size N in a transition period of length T, where the change comes after T_1 observations, is

$$e(T, N, T_1) = \frac{1}{T}\left(T_1\,\epsilon_1^N(C_1) + (T - N - T_1)\,\epsilon_2^N(C_2) + \sum_{i=1}^{N}\epsilon_2^N\left(C_{\frac{i}{N}}^m\right)\right) \tag{5}$$

All throughout the transition period the classifier will be trained on a sample of size N. The term $f(C)$ in equation (3) depends on the underlying probability distribution and the classifier model. Hence for the first T_1 observations the multiplier will be $f_1(C)$ and from that point onwards, the multiplier will be $f_2(C)$. Putting together (5) and (3), the error in the transition period can be calculated as a function of N. The third term in (5) is not easy to explicate as a function of N, hence we propose an approximation below.

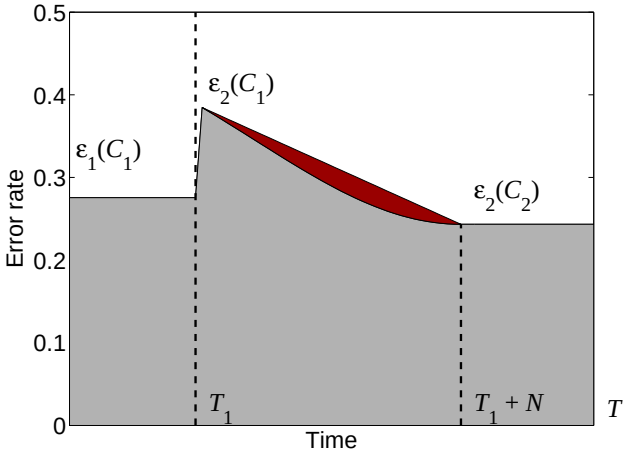


Fig. 3. Error of the online classifier in the transition period. The time of the concept shift where source S_1 is substituted by source S_2 is indicated by a vertical dashed line. The dark-shaded area is the addition to the error due to approximation of the slope with a line.

3.2 Optimal window size

The dark-shaded area in Figure 3 shows the additional area if the true slope of the error is replaced by a straight line. The starting point has coordinates $(T_1, \epsilon_2(C_1))$ as the distribution has just changed, and the classifier trained hitherto, C_1, has only seen observations from source S_1. The end point of the line segment has coordinates $(T_1 + N, \epsilon_2(C_2))$ because at this point the classifier is trained completely on observations coming from source S_2. We shall assume that the classes are equiprobable and that the steam of the data is i.i.d. This assumption is needed in order to be able to use $N/2$ in (3) as the sample size form which the class means are calculated. The i.i.d. assumption ensures that there are roughly $N/2$ observations from each class within a window of size N. The error approximation, taking into account (5) and (3), reduces to

$$\begin{aligned} e(T, N, T_1) &= \frac{1}{T}\Big(T_1\, \epsilon_1(C_1) + (T - T_1)\, \epsilon_2(C_2) \\ &+ \frac{2}{N}(f_1(C)T_1 + f_2(C)(T - T_1)) + \frac{1}{2}N(\epsilon_2(C_1) - \epsilon_2(C_2)) \\ &+ f_1(C) - f_2(C)\Big). \end{aligned} \tag{6}$$

This error is now easy to handle in order to derive an optimal value of the window size N. Taking the first derivative of (6), putting it to zero and solving for N, the optimal window size is

$$N_{\text{opt}} = 2\sqrt{\frac{T_1 f_1(C) + (T - T_1) f_2(C)}{(\epsilon_2(C_1) - \epsilon_2(C_2))}}. \tag{7}$$

As $\frac{\partial^2 e(T,N,T_1)}{\partial N^2} > 0$, the extremum is a minimum of the classification error for transition period of length T, with onset of the change after observation T_1. When there is no change in the distributions, $\epsilon_2(C_1) = \epsilon_2(C_2)$ and $N_{\text{opt}} \to \infty$, as should be expected.

3.3 Optimal window size for rotation and translation

Suppose that S_2 has the same mutual class configuration and distributions but the whole "structure" is displaced in $\Re^n$ (translated, rotated or both). The training of some classifier models, e.g., the linear and the quadratic discriminant classifiers, the k-nearest neighbour, SVM, etc. [3], makes them invariant to such changes. In other words, the errors for the original and for the displaced case are the same provided the same training procedure is used. Using the above notations, $\epsilon_1(C_1) = \epsilon_2(C_2) = \epsilon$. Also, $f_1(C) = f_2(C) = f(C)$. For this set-up, T_1 would not have any effect because the error will be the same before and after the N observations where the window slides over from S_1 to S_2. Therefore

$$e(T, N) = \epsilon + \frac{2f(C)}{N} + \frac{N}{2T}\left(\epsilon_2(C_1) - \epsilon\right) \tag{8}$$

and

$$N_{\text{opt}} = 2\sqrt{\frac{T\ f(C)}{(\epsilon_2(C_1) - \epsilon)}}. \tag{9}$$

Expressions for $f(C)$ are tabulated for some popular classifier models [4], [14], and $(\epsilon_2(C_1) - \epsilon)$ can be estimated empirically as the jump in the error immediately after the change.

4 An example of two Gaussian classes and the Nearest Mean Classifier (NMC)

Consider two Gaussian classes with equal prior probabilities of $1/2$ and identity covariance matrices. Let δ be the distance between the two means. For this case the nearest mean classifier (NMC) is Bayes-optimal and its error is $\epsilon = \Phi\left(-\frac{\delta}{2}\right)$. Following [4],

$$f(C) = \frac{1}{2\delta\sqrt{2\pi}}\ e^{-\frac{\delta^2}{8}}\ \left[\left(1 + \frac{\delta^2}{4}\right) n - 1\right]. \tag{10}$$

Substituting in (9) and guessing $\epsilon_2(C_1)$, we can calculate the optimal window size N_{opt} for a rotation-translation type of change. The value of $\epsilon_2(C_1)$ depends on the actual change - its direction and magnitude. The larger the change, the less adequate C_1 will be for the distribution from source S_2. To be able to derive an expression for N consider the following special case. Let $n = 1$, and the two sources define the following distributions

$$\begin{aligned} S_1 \quad &: \quad P(\omega_1) = P(\omega_2) = \frac{1}{2},\ \ p(x|\omega_1) \sim \mathcal{N}(0, 1),\ \ p(x|\omega_2) \sim \mathcal{N}(\delta, 1) \\ S_2 \quad &: \quad P(\omega_1) = P(\omega_2) = \frac{1}{2},\ \ p(x|\omega_1) \sim \mathcal{N}(-\theta\delta, 1),\ \ p(x|\omega_2) \sim \mathcal{N}(\delta(1-\theta), 1) \end{aligned}$$

Source S_2 simply translates the class configuration in S_1 to the left by offest $\theta\delta$. The optimal boundary for S_1 is at $b_1 = \frac{\delta}{2}$, and for S_2, $b_2 = \frac{\delta}{2} - \delta\theta$. Then $\epsilon_2(C_1)$ would be the error incurred within S_2 for classification boundary $b_1 = \frac{\delta}{2}$

$$\epsilon_2(C_1) = \frac{1}{2}\left[\Phi\left(-\frac{\delta}{2} - \theta\delta\right) + \Phi\left(-\frac{\delta}{2} + \theta\delta\right)\right]. \tag{11}$$

Using the approximation (9) and the expression for $f(C)$ (setting $n = 1$),

$$N_{\text{opt}} = (2\pi)^{-\frac{1}{4}} \sqrt{\frac{T\ \delta\ e^{-\frac{\delta^2}{8}}}{\Phi\left(-\frac{\delta}{2} - \theta\delta\right) + \Phi\left(-\frac{\delta}{2} + \theta\delta\right) - 2\Phi\left(-\frac{\delta}{2}\right)}}\ . \tag{12}$$

For this special case we can calculate the accurate value of the error (5), without having to resort to the approximation of the slope with a straight line. The boundary found through NMC is the middle of the segment between the two means. For i observations from source S_2 and $N - i$ from source S_1, the means for the two classes are

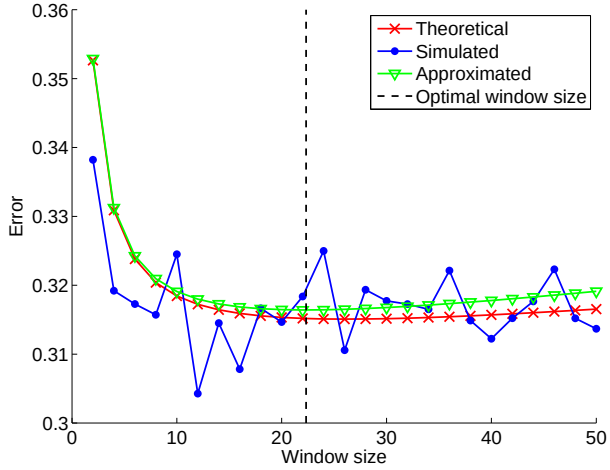
$$\begin{aligned} m_1 &= -\theta\ \delta\ \frac{i}{N} \\ m_2 &= (1-\theta)\ \delta\ \frac{i}{N} + \left(1 - \frac{i}{N}\right)\ \delta \end{aligned}$$

Hence the boundary at observation i is $b = \frac{\delta}{2} - \theta\delta\frac{i}{N}$, and the error is

$$\begin{aligned} e'(T, N) &= \frac{T-N}{T}\ \epsilon + \frac{2f(C)}{N} + \sum_{i=1}^{N} \epsilon\left(C^m_{\frac{i}{N}}\right) \\ &= \frac{T-N}{T}\ \Phi\left(-\frac{\delta}{2}\right) + \frac{\delta\ e^{-\frac{\delta^2}{8}}}{4N\sqrt{2\pi}} \\ &+ \frac{1}{2T}\sum_{i=1}^{N}\left[\Phi\left(-\frac{\delta}{2} - \theta\delta\left(1 - \frac{i}{N}\right)\right) + \Phi\left(-\frac{\delta}{2} + \theta\delta\left(1 - \frac{i}{N}\right)\right)\right] \end{aligned} \tag{13}$$

However, finding an accurate optimal value of N is not straightforward even for this simple scenario.

We ran simulations to gauge the adequacy of the estimates of the optimal window size. The empirical boundary was calculated from 300 random samples drawn from the distribution (or mixture of distributions) at each observation. Figure 4 shows the classification error as a function of the window size N for $\delta = 1$, $\theta = 1$ and transition period



$T = 200$, $\delta = 1$, $\theta = 1$
$N_{\text{opt}} = 22$, $N'_{\text{opt}} = 26$

Fig. 4. Error of the online classifier as a function of the window size N

$T = 200$. The simulated curve is plotted together with the "theoretical"[2] curve $e'(T, N)$ (13) and the approximation $e(T, N)$ (8). We show N_{opt} (12) and also

$$N'_{\text{opt}} = \arg\min_N \ e'(T, N).$$

The two values do not differ by much, which justifies the approximation of the slope of the error curve in the transition period. The curve for the approximated error is slightly higher than the "theoretical" curve because of the added surface. The difference causes a minor displacement of N_{opt} which is still in the area of the fairly flat minima of both curves. The simulated values of the error also agree with the prediction of the optimal window size.

5 An Experiment with the Electricity Market Data Set

5.1 Data

This data set is one of the few publicly available benchmark data sets for changing environments [7]. The version named Elec 2.2. was used here. It consists of 45 312 data points, each represented by three features: day of the week, time of the day and electricity demand of New South Wales, Australia at the time. The data set is a collection of successive measurements at every 30 minutes, spanning the period from May 1996 to December 1998. The class label of each point is either UP or DOWN, referring to whether the electricity price at the specified time is higher or lower than the average price of the preceding 24 hours. The classes are approximately equiprobable (58%/42%). In our experiments we used the error of the new data point in the sequence as the testing error before retrieving the correct label and deciding on the new training window. Thus the overall error from an experiment is the average of correct/wrong predictions on the whole data set.

For this data set, the label at time t is highly correlated with the label at time $t-1$, and also with the few preceding labels. This suggests a trivial solution whereby the label at t is taken to be the same as the label at $t-1$. In order to break this relationship and make the problem more difficult, we sub-sampled the data at regular offset, taking every K-th observation. We ran experiments with 4 versions of the original data set: for $K = 10$ (10% of the data), $K = 7$ (14.29%), $K = 4$ (25%) and $K = 2$ (50%).

5.2 Method

Equation (9) was used to design a window resizing algorithm for the nearest mean classifier (NMC). The initial window size was set to $N = 20$ and was re-calculated at each observation. The following choices and assumptions were made:

- Perfect classification (zero error) was possible before and after the (hypothetical) change, i.e., $\epsilon = 0$.
- The period of interest was the current window, i.e., $T = N$.

2. This "theoretical" curve is also an *approximation* of the error. In both (13) and (8), the effect of N is included by an approximate calculation [4]; however (8) introduces an additional inaccuracy by replacing the slope with a straight line.

TABLE 1
Error rates for the NMC with the Electricity Market data set (Elec2.2) for different window sizes, N and different percentages of sub-sampled data.

N	10%	14%	25%	50%
1	44.26 •	38.90 •	28.96 ○	20.68 ○
2	42.97 •	38.55 •	29.28 ○	20.99 ○
5	36.99 –	40.95 •	32.41 ○	27.13 ○
20	38.10 –	37.14 •	34.04 –	33.27 •
50	36.65 ○	34.72 ○	34.60 •	33.17 •
100	36.18 ○	35.27 –	34.80 •	34.80 •
'Resize'	38.05	35.95	33.36	30.85

Error of the 'largest prior' classifier = 42.45%
'•' 'Resize' is significantly better than this method
'○' 'Resize' is significantly worse than this method
(McNemar test for equivalence between the mean errors, $\alpha = 0.05$)

- The error "jump" was taken to be the error of the latest window.

Given the current window of size N, the "optimal" window size was calculated as

$$N^* = 2\sqrt{\frac{N f(C)}{\hat{e}_N}},$$

where $\hat{e}_N$ is the error of the current window and $f(C)$ is as in (10). If $N^* > N$, the new observation was added to the current window; otherwise, the current window was shrunk to the latest N^* observations. The value of N was reset to the modified length of the current window.

5.3 Experimental protocol

We ran experiments with the 5 versions of the data set for fixed windows of sizes of 1, 2, 5, 20, 50 and 100. In addition to NMC, we apply the linear discriminant classifier (LDC) [3] (the online version [12]), the nearest neighbour classifier, and a decision tree classifier (with no pruning). The proposed method, called 'Resize', is compared with the fixed window method for the chosen window sizes.

We also considered two window resizing methods from the literature: Drift detection [6] and ADWIN2 [2]. Neither of these methods detected a change in the error rate of the classifier for any of the 4 versions of the data set. This means that the training data for the classifier grew incrementally with each new observation. The classification results were identical and inferior to those of 'Resize' as well as the fixed windows.

5.4 Results and discussion

Table 1 contains the final error rate for the Nearest Mean Classifier for the 4 data versions. To quantify the significance of the differences between Resize and the fixed window sizes, we used the McNemar test since the same testing data has been classified by the all methods. For level of significance 0.05 we marked in the table all results that were significantly worse than 'Resize' using the symbol '•'. All results that were significantly better than 'Resize' are marked with '○'.

Figure 5 interpolates the error rates across the percentage of the data retained. The progression of the errors demonstrates the tendency of the nearest neighbour becoming the best window when data becomes more dense. It starts with the highest error rate when the data is sparse but then becomes the favourite from a certain percentage onwards. Larger windows fare better for sparse data but lose to the smaller windows when large percentage of data is retained. The 'Resize' method behaves reasonably for different data densities.

The error rates for the 'Resize' method with the different classifiers are shown in Table 2. Table 3 displays the statistical significance of the differences between the 'Resize' method and the fixed-window methods. Symbol '□' indicates that the respective method is significantly better than 'Resize', and symbol '■' indicates that the respective method is significantly worse than 'Resize'.

The results suggest that 'Resize' is fairly robust in regard to different classifier methods even though $f(C)$ is estimated as in (10) with the Euclidean distance as δ, and therefore applies to NMC. Table 3 shows also the Win-Draw-Loss numbers of 'Resize' compared to each fixed window size across all classifiers and data versions. All results are in favour of 'Resize'. Note that 'Resize' does not require tuning of any parameters whereas the optimal size of the fixed window is not known in advance.

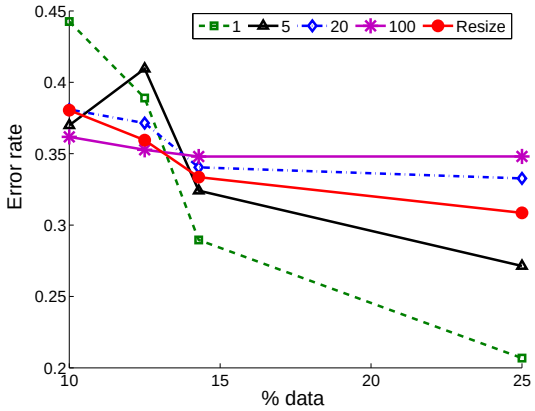


Fig. 5. Error of NMC for different window sizes as a function of the percentage of data retained using regularly subsampled Elec2.2 dataset

TABLE 2
Error rates for the 'Resize' method with the four classifiers for the 4 sub-sampled versions of the data set.

Data set →	10%	14%	25%	50%
NMC	38.05	35.95	33.36	30.85
LDC	35.88	32.64	29.98	27.30
1-nn	36.54	31.90	26.51	21.57
tree	35.55	32.78	27.68	22.60

6 Conclusions

The choice of the size of the moving window for classification of streaming data typically relies on heuristics. The intention of this study was to provide a stepping stone towards a theoretical basis for the choice of window size. Here we offer a framework for relating the running classification error to the window size. We derive a generic relationship for a fixed transition period containing a single abrupt concept change. Sudden displacement of the whole class structure (translation+rotation) is considered as a special case and illustrated on two Gaussian classes. We further devise a simple window resizing method ('Resize') and compare it with fixed-size windows on a benchmark data set. The results indicate that the 'Resize' is robust and accurate.

Further research directions include relaxing the assumptions and deriving theoretical relationships for different and/or more general cases. In particular, different types of concept change and different classifier models will be examined.

TABLE 3
Statistical significance of the differences between the 'Resize' method and the fixed window methods.

	10%				14%				25%				50%				
N	A	B	C	D	A	B	C	D	A	B	C	D	A	B	C	D	Win-Draw-Loss
1	■	■	■	■	■	■	■	■	□	□	■	■	□	□	□	□	10-0-6
2	■	■	■	■	■	■	■	■	□	–	■	■	□	□	□	□	10-1-5
5	–	■	–	–	■	■	■	■	□	■	■	■	□	–	–	□	8-5-3
20	–	–	–	–	■	–	–	■	–	□	□	–	■	□	■	■	5-8-3
50	□	–	–	–	□	–	–	–	■	■	□	■	■	■	□	■	6-6-4
100	□	–	–	□	–	–	–	–	■	■	■	■	■	■	□	■	7-6-3

Notes:
(A) NMC; (B) LDC; (C) 1nn; (D) decision tree
N is the fixed window size
'■' 'Resize' is significantly better
'□' 'Resize' is significantly worse

REFERENCES

[1] M. Baena-García, J. Campo-Ávila, R. Fidalgo, A. Bifet, R. Gavaldà, and R. Morales-Bueno. Early drift detection method. In *Fourth International Workshop on Knowledge Discovery from Data Streams*, pages 77–86, 2006.

[2] A. Bifet and R. Gavaldà. Learning from time-changing data with adaptive windowing. In *Proceedings of the Seventh SIAM International Conference on Data Mining*, pages 443 – 448, Minneapolis, Minnesota, USA, 2007.

[3] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, NY, second edition, 2001.

[4] K. Fukunaga and R. R. Hayes. Effects of sample size in classifier design. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(8):873–885, 1989.

[5] J. Gama, R. Klinkenberg, and J. Aguilar, editors. *The Fourth International Workshop on Knowledge Discovery from Data Streams*, Berlin, Germany, 2006.

[6] J. Gama, P. Medas, G. Castillo, and P. Rodrigues. Learning with drift detection. In *Advances in Artificial Intelligence - SBIA 2004, 17th Brazilian Symposium on Artificial Intelligence*, volume 3171 of *Lecture Notes in Computer Science*, pages 286–295. Springer Verlag, 2004.

[7] M. Harries. Splice-2 comparative evaluation: Electricity pricing, 1999.

[8] R. Klinkenberg and T. Joachims. Detecting concept drift with support vector machines. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML)*, pages 487–494, San Francisco, CA, USA, 2000. Morgan Kaufmann.

[9] R. Klinkenberg and I. Renz. Adaptive information filtering: Learning in the presence of concept drifts. In *AAAI-98/ICML-98 workshop Learning for Text Categorization*, Menlo Park,CA, 1998.

[10] I. Koychev and R. Lothian. Tracking drifting concepts by time window optimisation. In *Proceedings the 25th SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence, AI-2005*, pages 46–59, London, 2005. Springer.

[11] M. Kubat, J. Gama, and P. Utgoff. Incremental learning and concept drift: Editor's introduction. *Intelligent Data Analysis*, 8(3):211–212, 2004.

[12] L. I. Kuncheva and C.O. Plumpton. Adaptive learning rate for online linear discriminant classifiers. In *Proc. Joint IAPR International Workshops on Structural and Syntactic Pattern Recognition and Statistical Techniques in Pattern Recognition S+SSPR*, pages 510–519, Orlando, Florida, USA, 2008.

[13] M.M. Lazarescu and S. Venkatesh. Using selective memory to track concept drift effectively. In *Intelligent Systems and Control*, volume 388, Salzburg, Austria, 2003. ACTA Press.

[14] S. J. Raudys and A. K. Jain. Small sample size effects in statistical pattern recognition: Recommendations for practitioners. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):252–264, 1991.

[15] S. J. Raudys and V. Pikelis. On dimensionality, sample size, classification error and complexity of classification algorithm in pattern recognition. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 2:242–252, 1980.

[16] M. Scholz and R. Klinkenberg. An ensemble classifier for drifting concepts. In *Proceedings of the 2nd Workshop on Knowledge Discovery from Data Streams*, pages 53–64, Porto, Portugal, 2005.

[17] G. Widmer and M. Kubat. Learning in the presence of concept drift and hidden contexts. *Machine Learning*, 23:69–101, 1996.

[18] G. Widmer and M. Kubat. Special Issue on Context Sensitivity and Concept Drift. *Machine Learning*, 32, 1998.

[19] I. Zliobaite. Expected classification error of the Euclidean linear classifier under sudden concept drift. In *Proc. 5th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD'08)*, Jinan, China, 2008.