Online Linear Discriminant Analysis for Data Streams with Concept Drift

Sarah Schnackenberg, Uwe Ligges and Claus Weihs

Abstract Various methods based on classical classification methods such as linear discriminant analysis (LDA) have been developed for working on data streams in situations with concept drift. Nevertheless, the updated classifiers of such methods may result in a bad prediction error rate in case the underlying distribution incrementally changes further on.

Therefore, we invented a rather general extension to such methods to improve the forecasting quality. Under some assumptions we estimate a model for the time-dependent concept drift that is used to predict the forthcoming distributions of the features. These predictions of distributions are finally used in the LDA to build the classification rule and hence for predicting new observations. In a simulation study we consider different kinds of concept drift and compare the new extended methods with the methods these are based on.

Sarah Schnackenberg · Uwe Ligges · Claus Weihs
Department of Statistics, TU Dortmund University, 44221 Dortmund, Germany
🖂 schnackenberg@statistik.tu-dortmund.de
🖂 ligges@statistik.tu-dortmund.de
🖂 weihs@statistik.tu-dortmund.de

Archives of Data Science, Series A
(Online First)
KIT Scientific Publishing
Vol. 5, No. 1, 2018

DOI 10.5445/KSP/1000087327/02
ISSN 2363-9881

1 Introduction and Motivation

Various methods based on classical classification methods have been developed for working on data streams in situations with concept drift. For a first explanation of the term *concept drift* along with examples see Hoens et al (2012). Webb et al (2016) present a comprehensive overview over the different types of concept drift and then try to formulate a standardized terminology by introducing formal definitions for the different types. We focus on the *linear discriminant analysis* (LDA) as a classification method in the context of data streams and concept drift.

Pang et al (2005) focus on Fisher's LDA (Fisher, 1936) and use standard formulas for updating the means and variances iteratively with new observations. The resulting classification rule is identical to that built on all observations at a time. Every data point has the same influence and is assigned the same weight, respectively (no adaptation to a possible concept drift).

However, observing data streams, the underlying distribution of features and/or the target variable may change over time. In such a case, exact updates and equal weighting of all observations in the data streams may lead to bad performance of the updated classifier. A weighting adapted to the currentness of new observations can show a better performance.

Kuncheva and Plumpton (2008) deal with an adaptive online version of the canonical LDA. They also use standard formulas for updating the relevant values. In addition, they extend the adaptive online version with an option to adjust to concept drift by introducing a fixed or adaptive learning rate. Hence, new observations are assigned different weights indirectly.

Anagnostopoulos et al (2012) use the idea of exponential forgetting by exponentially down-weighting the contribution of past observations in the updating steps for the classifier based on new observations in the data stream.

Nevertheless, the updated classifiers of such methods may result in a bad prediction error rate in case the underlying distribution incrementally changes further on and the forecasting quality can still be improved. Hence, in the following a framework is formulated that extends existing methods for online LDA. The idea is to estimate a model for the time-dependent concept drift under some assumptions. This model is used to predict the forthcoming distribution of the features. These predictions of distributions are finally used in the LDA to learn the classification rule and hence for predicting new observations.

2 Online Linear Discriminant Analysis

We focus on a data stream of p-dimensional vectors $\boldsymbol{x}_i$, $i = 1, 2, \ldots$, denoted by $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots\}$. The label of each observation arrives temporally delayed, more precisely, the label of observation $\boldsymbol{x}_t$ shows up at time $t + 1$.

At any time t the iteratively updated classifier is indirectly based on a set of t training vectors $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1,\ldots,t}$. The observations possess M different classes, where $g(\boldsymbol{x}_i) = c$, $c \in \{1, \ldots, M\}$, denotes the label of $\boldsymbol{x}_i$. $n_t^{(c)}$ indicates the number of observations with label c up to time t and n_t is the number of all observations in the data stream up to time t, $n_t = t$ in this contribution.

To build the classifier we need the class means, the priors as well as the inverse of the empirical covariance matrix, denoted by $\boldsymbol{m}^{(c)}_{n_t^{(c)}}$, $P_t^{(c)}$ and $\boldsymbol{S}_t^{-1}$. The subscript $n_t^{(c)}$ indicates on how many observations the class mean at time t is calculated on. The initialization can be performed on the first t observations of the data stream and by standard formulas: Class means, relative frequencies and inverse of the empirical covariance matrix as estimators for the expected values, the priors of the classes and the inverse covariance matrix.

2.1 Exemplary Method for Online Linear Discriminant Analysis

We concentrate exemplarily on an advanced online LDA, the online version of the canonical LDA as proposed by Kuncheva and Plumpton (2008). They use standard updating formulas: Mean vectors as maximum likelihood estimators for the expected values of the classes, the inverse of the empirical covariance matrix, and estimators for the priors. Thereby they iteratively update the classifier with every new observation in the data stream by updating all the needed measures.

In addition, they extend their adaptive online version with an option to adjust to concept drift through a weighting adapted to the currentness of new observations. This is done by introducing a fixed or adaptive learning rate λ, $0 < \lambda < 1$, that assigns different weights to new observations indirectly. For the adaptive case the authors introduce a heuristic approach (Kuncheva and Plumpton, 2008, p. 516f.). A large learning rate $\lambda \rightarrow 1$ can be interpreted as focusing only on most recent data, a small learning rate $\lambda \rightarrow 0$ is equivalent to not updating the classifier. For $\lambda = 0.5$ all observations are weighted equally.

The updated quantities at time $t+1$, based on the respective quantities at time t and the new observation $\boldsymbol{x}_{t+1}$ with label $g(\boldsymbol{x}_{t+1}) = k$, as well as the learning rate are given as follows (Kuncheva and Plumpton, 2008, p. 514):

$$\boldsymbol{m}^{(c)}_{n^{(c)}_{t+1}} = \begin{cases} \boldsymbol{m}^{(c)}_{n^{(c)}_t} & \text{if } g(\boldsymbol{x}_{t+1}) \neq c, \\ \dfrac{(1-\lambda)\cdot n^{(c)}_t \cdot \boldsymbol{m}^{(c)}_{n^{(c)}_t} + \lambda \cdot \boldsymbol{x}_{t+1}}{(1-\lambda)\cdot n^{(c)}_t + \lambda} & \text{if } g(\boldsymbol{x}_{t+1}) = c \in \{1,\ldots,M\}, \\ \boldsymbol{x}_{t+1} & \text{if } g(\boldsymbol{x}_{t+1}) = c = M+1, \end{cases} \tag{1}$$

$$P^{(c)}_{t+1} = \begin{cases} \dfrac{(1-\lambda)\cdot n^{(c)}_t}{(1-\lambda)\cdot n_t + \lambda} & \text{if } g(\boldsymbol{x}_{t+1}) \neq c, \\ \dfrac{(1-\lambda)\cdot n^{(c)}_t + \lambda}{(1-\lambda)\cdot n_t + \lambda} & \text{if } g(\boldsymbol{x}_{t+1}) = c \in \{1,\ldots,M\}, \\ \dfrac{1}{n_t + 1} & \text{if } g(\boldsymbol{x}_{t+1}) = c = M+1, \end{cases} \tag{2}$$

$$\boldsymbol{S}^{-1}_{t+1} = \frac{(1-\lambda)\cdot n_t + \lambda}{(1-\lambda)\cdot n_t}\left(\boldsymbol{S}^{-1}_t - \frac{\boldsymbol{S}^{-1}_t \boldsymbol{v}\boldsymbol{v}^T \boldsymbol{S}^{-1}_t}{\frac{(1-\lambda)\cdot n_t \cdot (n^{(k)}_t + 1)}{\lambda \cdot n^{(k)}_t} + \boldsymbol{v}^T \boldsymbol{S}^{-1}_t \boldsymbol{v}}\right), \; 0 < \lambda < 1, \tag{3}$$

where

$$\boldsymbol{v} = \boldsymbol{x}_{t+1} - \frac{\left(n^{(k)}_t + 1\right)\cdot \boldsymbol{m}^{(k)}_{n^{(k)}_{t+1}} - \boldsymbol{x}_{t+1}}{n^{(k)}_t}. \tag{4}$$

For more details on this updating method see Kuncheva and Plumpton (2008). Note that the updated inverse covariance matrix from Equation (3) (and Equation (4)) has been corrected compared to Kuncheva and Plumpton (2008, p. 514) so that it represents the properties for $\lambda = 0.5$, $\lambda \rightarrow 0$ and $\lambda \rightarrow 1$.

In the canonical LDA the classification rule is based on M different discriminant functions, that are calculated based on the Gaussian distribution. A new observation $\boldsymbol{x}$ is assigned to the class c with the highest value (e.g. Kuncheva and Plumpton, 2008, p. 512):

$$\underset{c\in\{1,\ldots,M\}}{\operatorname{argmax}} \; g_c(\boldsymbol{x}) = \log P^{(c)} - \frac{1}{2}\cdot \left(\boldsymbol{m}^{(c)}\right)^T \boldsymbol{S}^{-1}\boldsymbol{m}^{(c)} + \left(\boldsymbol{m}^{(c)}\right)^T \boldsymbol{S}^{-1}\boldsymbol{x}. \tag{5}$$

3 Extension of Existing Methods

We propose the following framework that extends existing methods for online LDA with the aim to build better online classifiers concerning the forecasting quality by, first of all, estimating the time-dependent concept drift.

3.1 Estimation of time-dependent Concept Drift

As a first step, we assume the expected values of all classes c to be subject to a linear trend:

$$\boldsymbol{\mu}_i^{(c)} = \boldsymbol{\beta}_0^{(c)} + \boldsymbol{\beta}_1^{(c)} \cdot i, \quad i = 1, \dots. \tag{6}$$

As a result the expected values of one class c differ for different times $t \neq s$ and $\boldsymbol{\beta}_1^{(c)} \neq \mathbf{0}$: $\boldsymbol{\mu}_t^{(c)} \neq \boldsymbol{\mu}_s^{(c)}$, $t \neq s$. Furthermore, we assume the covariance matrices of all classes are unchanged over time and identical for all classes, which is the standard assumption for the LDA. For a two-dimensional distribution of one class this linear shift of the expected value over time, yet unchanged covariance matrix, is exemplarily illustrated in Figure 1.

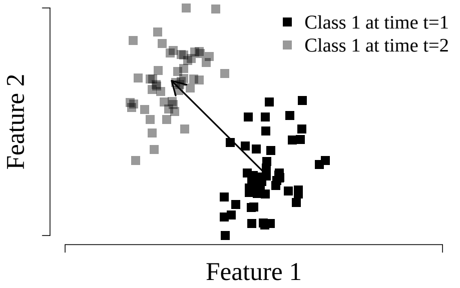

Figure 1: Example for a linear shift of the expected value of a two-dimensional distribution of one class from one time to the next. The covariance matrix is unchanged over time.

3.1.1 Local Linear Regression Model

The overall idea in order to improve the forecasting quality of the running classifier is to replace each maximum likelihood estimator $\boldsymbol{m}_{n_t^{(c)}}^{(c)}$ in the prediction rule from Equation (5) of the LDA by an estimator based on a local linear regression model of the class means, hence a model for the time-dependent concept drift.

We will exemplarily focus on the extension of the method proposed by Kuncheva and Plumpton (2008) (see Section 2.1). In the case of a fixed learning rate $\lambda = 0.5$ the resulting model from the updating steps in Equations (1)–(3) at any time t is the same as that trained on all data up to time t at once. Hence, the class mean at time t can also be written in a static way based on all observations:

$$\boldsymbol{y}_t^{(c)} := \frac{1}{n_t^{(c)}} \sum_{\substack{i:\, g(\boldsymbol{x}_i)=c \\ i \leq t}} \boldsymbol{x}_i = \frac{1}{\sum_{j=1}^{t} \mathbb{1}_{\{g(\boldsymbol{x}_j)=c\}}} \sum_{i=1}^{t} \left(\boldsymbol{x}_i \cdot \mathbb{1}_{\{g(\boldsymbol{x}_i)=c\}}\right), \tag{7}$$

where $\mathbb{1}$ is the indicator function and $\boldsymbol{y}_t^{(c)} = \boldsymbol{m}_{n_t^{(c)}}^{(c)}$ holds for $\lambda = 0.5$. The class means as unbiased maximum likelihood estimators can be considered as estimates for the expected values of the features of all classes. In order to model the linear trend of the expected values and predict the forthcoming expected values of the features at time t a local linear regression model is fitted to n_{trend} recent updated class means from the interval

$$I = \underbrace{\{j : g(\boldsymbol{x}_j) = c,\ j = t - n_{\text{trend}} + 1, \dots, t\}}_{n_{\text{trend}} \text{ time points}} \tag{8}$$

using the shifted times $z_i^{(c)}$:

$$\boldsymbol{y}_i^{(c)} = \boldsymbol{\alpha}_{0t}^{(c)} + \boldsymbol{\alpha}_{1t}^{(c)} \cdot z_i^{(c)} + \boldsymbol{\epsilon}_i, \qquad i \in I, \tag{9}$$

$$z_i^{(c)} = \frac{1}{n_i^{(c)}} \sum_{\substack{j:\, g(\boldsymbol{x}_j)=c \\ j \leq i}} j = \frac{1}{\sum_{k=1}^{i} \mathbb{1}_{\{g(\boldsymbol{x}_k)=c\}}} \sum_{j=1}^{i} \left(j \cdot \mathbb{1}_{\{g(\boldsymbol{x}_j)=c\}}\right). \tag{10}$$

Note that if $\lambda \neq 0.5$, the class means $\boldsymbol{y}_i^{(c)}$ in Equation (9) have to be the streaming counterparts from Equation (1), even for formal notation of the linear regression model.

The shifted times are used, because the class mean with index i is not an adequate estimator for the expected value at time i in the case of a linear trend of the expected values. That is because the class means are updated with each new observation in the data stream, hence the class mean at time i is based on all observations in class c from time 1 to i. The idea of shifted times is graphically visualized in Figure 2. You can see that the class means $\boldsymbol{y}_i^{(1)}$ clearly do not represent the expected values $\boldsymbol{\mu}_i^{(1)}$ at time i. Instead they represent the respective expected values of class 1 at time $z_i^{(1)}$.

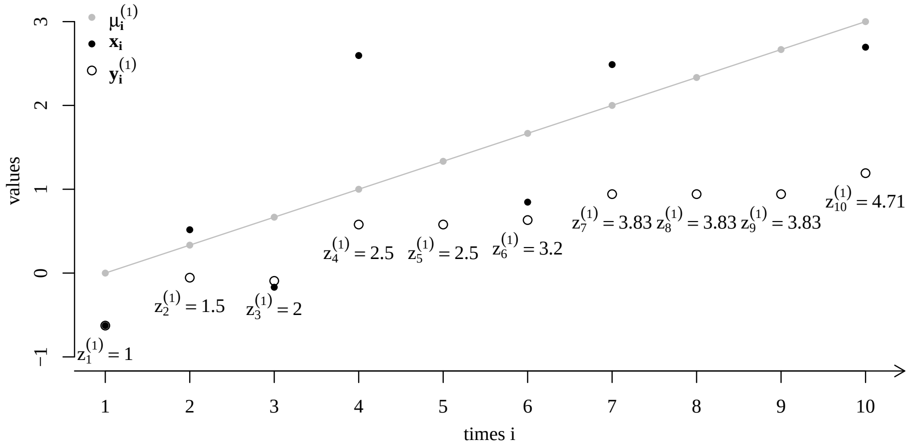

Figure 2: Example for the determination of shifted times from Equation (10) for one class $c = 1$. The gray points represent the expected values $\boldsymbol{\mu}_i^{(c)}$ for each time i. This expected value is subject to a linear trend. The black points illustrate observations $\boldsymbol{x}_i$ that show up in class 1 at time i (here $i \in \{1, 2, 3, 4, 6, 7, 10\}$). These observations follow the distribution with expected value $\boldsymbol{\mu}_i^{(c)}$. The circles represent the class means calculated by Equation (7). These are adequate estimators for the expected values at time $z_i^{(1)}$ instead of time i.

Simplified, consider observations $\boldsymbol{x}_i$ for one class for each time $i = 1, \ldots, t$ and $\lambda = 0.5$ with a linearly changing expectation. Then $\boldsymbol{m}_{n_t} = \bar{\boldsymbol{x}}$ is a good estimator for the expectation of $\boldsymbol{x}_{\frac{t+1}{2}}$ rather than for the desired $\boldsymbol{x}_{t+1}$.

3.1.2 Prediction of Forthcoming Distribution of the Features

This linear regression model for the time-dependent concept drift is used to predict the forthcoming distribution of the features:

$$\hat{\boldsymbol{y}}_{t+1}^{(c)} = \hat{\boldsymbol{\alpha}}_{0t}^{(c)} + \hat{\boldsymbol{\alpha}}_{1t}^{(c)} \cdot (t+1), \qquad c = 1, \ldots, M. \tag{11}$$

Because we focus on the linear regression model, closed forms for least squares estimators of linear regression can be used, leading to the following estimators:

$$\begin{aligned}
\hat{\boldsymbol{\alpha}}_{1t}^{(c)} &= \frac{\sum_{i\in I} (z_i^{(c)} - \bar{z}^{(c)}) \left(\boldsymbol{y}_i^{(c)} - \bar{\boldsymbol{y}}^{(c)}\right)}{\sum_{i\in I} (z_i^{(c)} - \bar{z}^{(c)})^2}, \\
\hat{\boldsymbol{\alpha}}_{0t}^{(c)} &= \bar{\boldsymbol{y}}^{(c)} - \hat{\boldsymbol{\alpha}}_{1t}^{(c)} \cdot \bar{z}^{(c)}, \qquad \text{where} \qquad & \bar{\boldsymbol{y}}^{(c)} &= \frac{1}{n_{\text{trend}}} \sum_{i\in I} \boldsymbol{y}_i^{(c)}, \\
& & \bar{z}^{(c)} &= \frac{1}{n_{\text{trend}}} \sum_{i\in I} z_i^{(c)}. \qquad (12)
\end{aligned}$$

These predictions $\hat{\boldsymbol{y}}_{t+1}^{(c)}$ are then used to build the classification rule for the LDA. The former maximum likelihood estimator $\boldsymbol{m}_{n_t^{(c)}}^{(c)}$ is replaced by the estimated parameter $\hat{\boldsymbol{y}}_{t+1}^{(c)}$ in the prediction rule of Equation (5). New observations in the data stream can then be predicted by:

$$\begin{aligned}
\underset{c\in\{1,\ldots,M\}}{\text{argmax}}\; g_c(\boldsymbol{x}_{t+1}) = \; & \log P_t^{(c)} - \frac{1}{2} \cdot \left(\hat{\boldsymbol{y}}_{t+1}^{(c)}\right)^T \boldsymbol{S}_t^{-1} \hat{\boldsymbol{y}}_{t+1}^{(c)} \\
& + \left(\hat{\boldsymbol{y}}_{t+1}^{(c)}\right)^T \boldsymbol{S}_t^{-1} \boldsymbol{x}_{t+1}. \qquad (13)
\end{aligned}$$

3.2 Algorithm

The extension of methods for online LDA for data streams with concept drift, e.g. a linear drift of the expected values of the classes, is summarized in the following Algorithm 1:

Algorithm 1 Extension for online LDA for data streams with concept drift

```
Require: Data stream $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots\}$, $t$, $n_{\text{trend}}$.
1: Initialization of LDA based on $t$ observations (8) by e.g. (1).
2: When new observation $\boldsymbol{x}_{t+1}$ arrives → Update of measures (1), (2) and (3) for LDA.
3: Use class means of $n_{\text{trend}}$ recent updates of the interval (8) to fit a model, e.g. (9)
   (for the first time after $n_{\text{trend}}$ updates of measures for LDA).
4: Estimation of forthcoming expected values of features by means of prediction of the
   forthcoming class means (11).
5: Use estimated parameters of classes in the LDA to build the classification rule and
   predict the new observation by (13).
6: Repeat 2–5 for data stream.
```

As we propose a general procedure for extending methods for online linear discriminant analysis the training complexity of such an extended method at any time t in the data stream overall consists of the complexity of the specific method for online LDA and in addition the training complexity of the linear regression based on n_{trend} observations.

In the case of using normal equations to solve the least-squares problem the different steps of calculation have a complexity of $\mathcal{O}(n_{\text{trend}} \cdot p^2)$ (matrix multiplication of $p \times n_{\text{trend}}$- and $n_{\text{trend}} \times p$-matrix), $\mathcal{O}(n_{\text{trend}} \cdot p)$ (matrix multiplication of $p \times n_{\text{trend}}$-matrix and n_{trend}-dimensional vector) and $\mathcal{O}(p^3)$ (inversion and matrix multiplication of $p \times p$-matrices) when using standard matrix multiplication and matrix inversion via Gaussian elimination (Golub and Loan, 1996), where p is the dimension of the feature space. If $n_{\text{trend}} > p$, then the training complexity simplifies to $\mathcal{O}(n_{\text{trend}} \cdot p^2)$.

4 Simulation Results

All implementations and analyses of results were performed in R (R Core Team, 2018), using the standard packages as well as the additional packages `MASS` (Venables and Ripley, 2002), `mvtnorm` (Genz and Bretz, 2009; Genz et al, 2019), `BatchJobs` (Bischl et al, 2015), and `biglm` (Lumley, 2013). We compare the advanced online LDA by Kuncheva and Plumpton (2008) using a fixed learning rate (see Section 2.1) $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with the new introduced extension on a range of different data situations representing different characteristics of both incremental and sudden concept drift. These data situations are schematically visualized and explained in Figure 3.

For the simulation of each of the four data situations we focus on 4000 data points in two classes. The distributions in both classes are two-dimensional Gaussians with equal covariance matrix (assumptions in LDA), but differing expected values. The covariance matrices, we use

$$\mathbf{\Sigma} = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}, \tag{14}$$

are even unchanged over time t, while the expected values of the classes change, representing a concept drift.

In the case of an incremental drift (Figure 3 (a)–(c)) the distribution changes for every new observation in the data stream, while in the case of a sudden drift (d) each 1000 successive observations follow one underlying distribution.

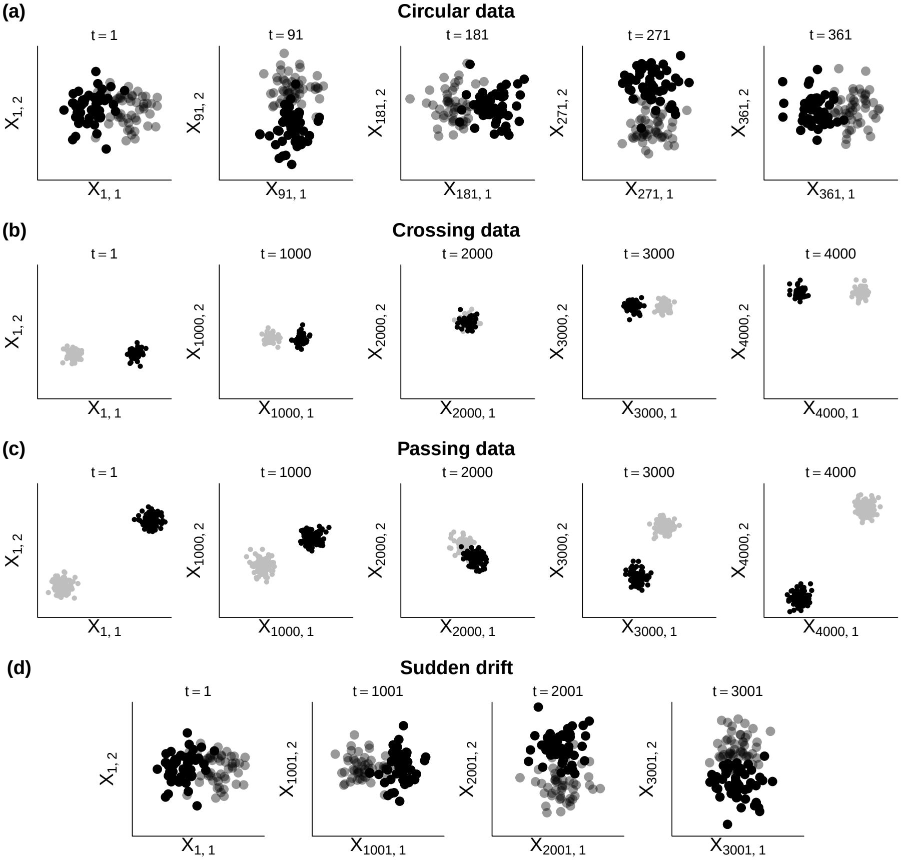

Figure 3: Visualization of binary (two classes) two-dimensional distributions of the four different data situations (rows (a)–(d)) for four or five different times, respectively (columns).

The following steps are performed for all four data situations: For each distribution of features at each time we simulated 100 additional observations for testing the prediction accuracy. The initialization is performed on the first $t = 10$ observations respectively. The initialization of the linear regression model for class means is based on the class means of the first n_{trend} update steps of the LDA. The steps 2–5 of Algorithm 1 are repeated for all 4000 observations. The prediction (step 5) is performed for the test data of the next time point, respectively. The simulation of the whole data situation and execution of Algorithm 1 is repeated 100 times to be able to produce a mean prediction error for each time.

In the following sections the four data situations and their simulations are explained in detail followed by a discussion of the results on these simulated data.

4.1 "Circular" data (a)

For the "circular data" the expected values of the two classes geometrically lie on a circle with radius 2 on contrary sides. We focus on a shift of the expected values of one degree at each time. That means a full rotation of the distribution of features after 360 data points in the data stream (see Figure 3 (a)).

As our proposal assumes a linear trend of the expected values while the example shows a circular one, the latter can be approximated by a local version of the former, i.e. modeling a *local* linear trend. Table 1 summarizes the results by means and standard deviations of the mean predictions errors as well as means of standard deviations of prediction errors for the whole data stream.

Table 1: Results of simulation study for "circular" data. Means and standard deviations (in parentheses) of mean prediction errors over time (curves in Figure 4; first row resp.) and means of standard deviations of prediction errors over time (second row resp.).

	base model (Section 2.1)	extension with			
λ	no extension	$n_{\text{trend}} = 10$	$n_{\text{trend}} = 20$	$n_{\text{trend}} = 50$	$n_{\text{trend}} = 200$
0.1	0.4979 (0.308)	0.1111 (0.007)	0.0955 (0.005)	0.1077 (0.011)	0.5995 (0.088)
	0.0737	0.0506	0.0338	0.0368	0.0866
0.3	0.4974 (0.313)	0.1083 (0.006)	0.0932 (0.043)	0.1044 (0.007)	0.5946 (0.080)
	0.0584	0.0468	0.0324	0.0335	0.0641
0.5	0.4976 (0.190)	0.1080 (0.006)	0.0928 (0.004)	0.1032 (0.004)	0.5832 (0.084)
	0.1483	0.0464	0.0322	0.0331	0.0611
0.7	0.4695 (0.078)	0.1083 (0.006)	0.0928 (0.004)	0.1024 (0.004)	0.5627 (0.104)
	0.1338	0.0468	0.0323	0.033	0.0599
0.9	0.3672 (0.107)	0.1110 (0.009)	0.0941 (0.007)	0.1010 (0.004)	0.4897 (0.137)
	0.0817	0.0492	0.0335	0.0331	0.0584

The results are visualized in Figure 4 for fixed learning rates $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with the proposed extension by a local linear regression model based on a window size of $n_{\text{trend}} \in \{10, 20, 50, 200\}$ recent class means and without our extension, respectively.

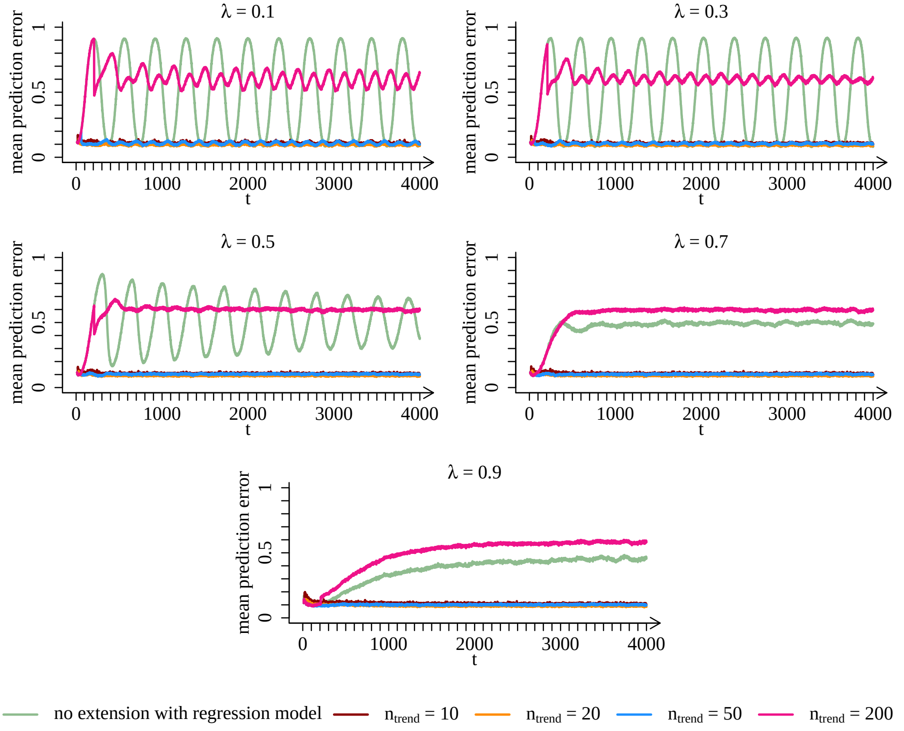

Figure 4: Exemplary mean prediction errors for fixed learning rates $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with the extension by a local linear regression model based on an interval of $n_{\text{trend}} \in \{10, 20, 50, 200\}$ recent class means and without our extension for the simulated “circular” data (see Figure 3 (a)).

For this data situation the method for online LDA can be drastically improved concerning the forecasting quality by extending it with a local linear regression model to estimate the concept drift. For example, we see the proposed extension can improve the mean prediction error by a factor of about 4–5 for $\lambda = 0.5$ (cp. Table 1). Without the extension the mean prediction error rises to a value of about 0.9 over time (green curve in Figure 4) in some cases. It declines to almost the value from the beginning for some λ after 360 points or rather oscillates because the distribution of features has fully rotated once on the circle, leading to the same distribution as the one in the beginning. Furthermore, the results strongly depend on the choice of λ. The oscillation declines with rising value of λ. Still, the decrease of the mean prediction error is much lower. The minimal mean prediction error is not less than 0.35 for $\lambda = 0.9$. In addition, note that the optimal setting of λ is typically unknown for new applications. With the introduction of a local linear regression model these problems can partly be solved. The prediction error settles at a level of around 0.1 or even a little less with small standard deviation, where both mean and standard deviation are almost independent of λ. In this case the results are more dependent only on the choice of n_{trend}. Until a certain value an increasing value of n_{trend} (higher smoothing) leads to a lower mean prediction error for all λ. Hence, a too high value of n_{trend} degrades the results concerning the forecasting quality (see Table 1) in this case. This result is due to a non-linear trend of the expected values and hence opposition to the assumptions of the model. Still, we see that the trend can be approximated by a *local* linear trend leading to an improvement compared to the method without extension (here for $n_{\text{trend}} \leq 50$; cp. Table 1 and Figure 4).

All in all, although the assumption of a linear trend of the expected values is not fulfilled in this data situation the approximation by a *local* linear trend is sufficient to drastically improve the results.

4.2 “Crossing” data (b)

In the case of “crossing” data (cp. Figure 3 (b)) the assumption of a time-dependent linear trend of the expected values of the classes in Equation (6) is fulfilled. The two-dimensional distributions change over time as the expected values smoothly shift along a hyperplane (here straight line). The speed is identical for both classes and the hyperplanes intersect at time $t = 2001$, the expectation of both classes being

$$\boldsymbol{\mu}_{2001} = \begin{pmatrix} 10 \\ 10 \end{pmatrix} \tag{15}$$

so that the distributions fully overlap, before moving apart again. In detail the expected values follow the following linear trend:

$$\begin{aligned}\boldsymbol{\mu}_i^{(1)} &= \begin{pmatrix}-0.005\\-0.005\end{pmatrix} + \begin{pmatrix}0.005\\0.005\end{pmatrix}\cdot i, \quad i = 1,\dots,4000,\\ \boldsymbol{\mu}_i^{(2)} &= \begin{pmatrix}20.005\\-0.005\end{pmatrix} + \begin{pmatrix}-0.005\\0.005\end{pmatrix}\cdot i, \quad i = 1,\dots,4000.\end{aligned} \tag{16}$$

The results for $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ and $n_{\text{trend}} \in \{10, 20, 50, 200\}$ are summarized in Table 2 and visualized in Figure 5.

Table 2: Results of simulation study for "crossing" data. Means and standard deviations (in parentheses) of mean prediction errors over time (curves in Figure 5; first row resp.) and means of standard deviations of prediction errors over time (second row resp.).

	base model (Section 2.1)		extension with		
λ	no extension	$n_{\text{trend}} = 10$	$n_{\text{trend}} = 20$	$n_{\text{trend}} = 50$	$n_{\text{trend}} = 200$
0.1	0.5015 (0.455)	0.0756 (0.133)	0.0628 (0.126)	0.0596 (0.121)	0.0579 (0.118)
	0.0136	0.0506	0.0144	0.0123	0.0116
0.3	0.5008 (0.458)	0.0749 (0.134)	0.0630 (0.127)	0.0598 (0.122)	0.0582 (0.119)
	0.0117	0.0467	0.0143	0.0123	0.0117
0.5	0.4965 (0.454)	0.0723 (0.135)	0.0629 (0.127)	0.0598 (0.122)	0.0582 (0.119)
	0.0160	0.0318	0.0142	0.0123	0.0117
0.7	0.2136 (0.349)	0.0703 (0.135)	0.0627 (0.126)	0.0596 (0.122)	0.0579 (0.119)
	0.0182	0.0222	0.0142	0.0123	0.0117
0.9	0.0736 (0.163)	0.0692 (0.134)	0.0629 (0.126)	0.0597 (0.122)	0.0580 (0.119)
	0.0133	0.0199	0.0144	0.0124	0.0117

In this situation the prediction error rises to a value of about one over time for the online LDA without extension for different learning rates (green curves). Extending the method with a local linear regression model to estimate the concept drift leads to a huge reduction of the maximal prediction error over time. The error does not exceed the Bayes error of 0.5 at $t = 2001$, when both distributions fully overlap. After $t = 2001$, the mean prediction error decreases to a value close to zero over time for all values of λ and n_{trend} contrary to the advanced online LDA without extension. For a lower value of n_{trend} the prediction error decreases for higher learning rates. For higher values of n_{trend} (higher smoothing) the value of λ does not strongly affect the results anymore. Comparing the results for varying magnitudes of smoothing the mean prediction error as well as the standard deviations of prediction error decrease for increasing n_{trend} for all learning rates λ (cp. Table 2). However, the mean prediction errors differ just slightly which is evident from the fact that the curves for the extension in Figure 5 strongly resemble each other for all λ and n_{trend}.

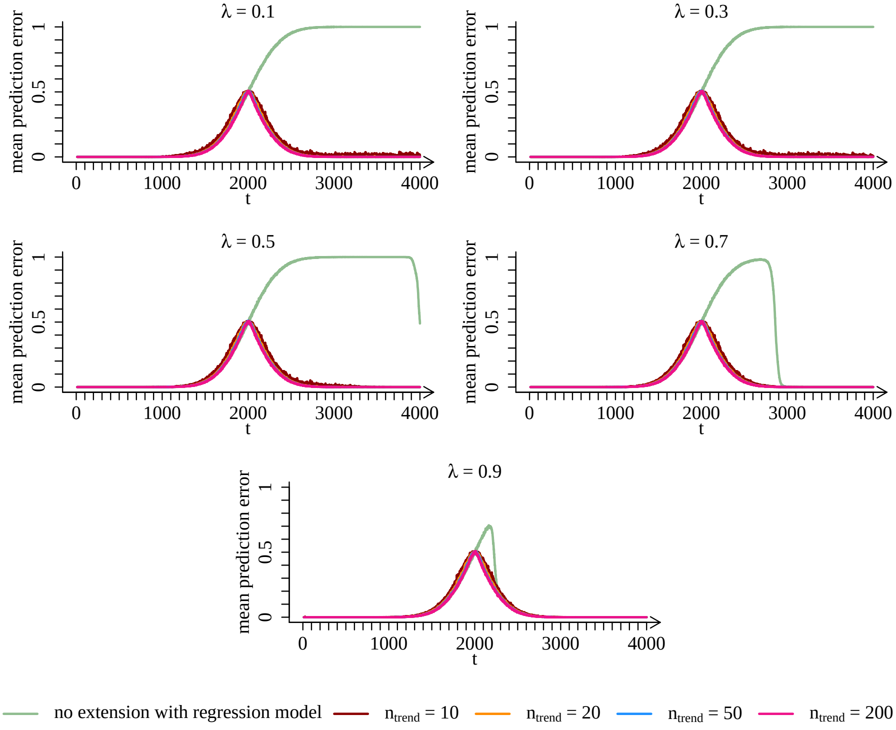

Figure 5: Exemplary mean prediction errors for fixed learning rates $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with the extension by a local linear regression model based on an interval of $n_{\text{trend}} \in \{10, 20, 50, 200\}$ recent class means and without our extension for the simulated "crossing" data (see Figure 3 (b)).

4.3 "Passing" data (c)

In the case of "passing" data (see Figure 3 (c)) the two expected values smoothly shift along two parallel hyperplanes (here straight lines) in opposite directions:

$$\begin{aligned} \boldsymbol{\mu}_i^{(1)} &= \begin{pmatrix} -0.005 \\ -0.005 \end{pmatrix} + \begin{pmatrix} 0.005 \\ 0.005 \end{pmatrix} \cdot i, \quad i = 1, \ldots, 4000, \\ \boldsymbol{\mu}_i^{(2)} &= \begin{pmatrix} 23 \\ 17 \end{pmatrix} + \begin{pmatrix} -0.005 \\ -0.005 \end{pmatrix} \cdot i, \quad i = 1, \ldots, 4000. \end{aligned} \tag{17}$$

In this case the advanced online LDA by Kuncheva and Plumpton (2008) works well for large learning rates, especially $\lambda = 0.9$ (see Figure 6).

Table 3: Results of simulation study for “passing” data. Means and standard deviations (in parentheses) of mean prediction errors over time (curves in Figure 6; first row resp.) and means of standard deviations of prediction errors over time (second row resp.).

	base model (Section 2.1)		extension with		
λ	no extension	$n_{\text{trend}} = 10$	$n_{\text{trend}} = 20$	$n_{\text{trend}} = 50$	$n_{\text{trend}} = 200$
0.1	0.3467 (0.361)	0.0202 (0.028)	0.0147 (0.023)	0.0137 (0.022)	0.0133 (0.021)
	0.0640	0.0239	0.0086	0.0077	0.0075
0.3	0.1206 (0.118)	0.0229 (0.028)	0.0192 (0.025)	0.0179 (0.023)	0.0173 (0.023)
	0.0247	0.0155	0.0106	0.0096	0.0093
0.5	0.0640 (0.060)	0.0222 (0.028)	0.0189 (0.025)	0.0176 (0.023)	0.0170 (0.023)
	0.0184	0.0143	0.0104	0.0095	0.0091
0.7	0.0316 (0.039)	0.0174 (0.026)	0.0149 (0.023)	0.0139 (0.022)	0.0134 (0.022)
	0.0125	0.0112	0.0082	0.0074	0.0071
0.9	0.0114 (0.024)	0.0115 (0.023)	0.0097 (0.020)	0.0089 (0.019)	0.0085 (0.018)
	0.0054	0.0070	0.0052	0.0047	0.0045

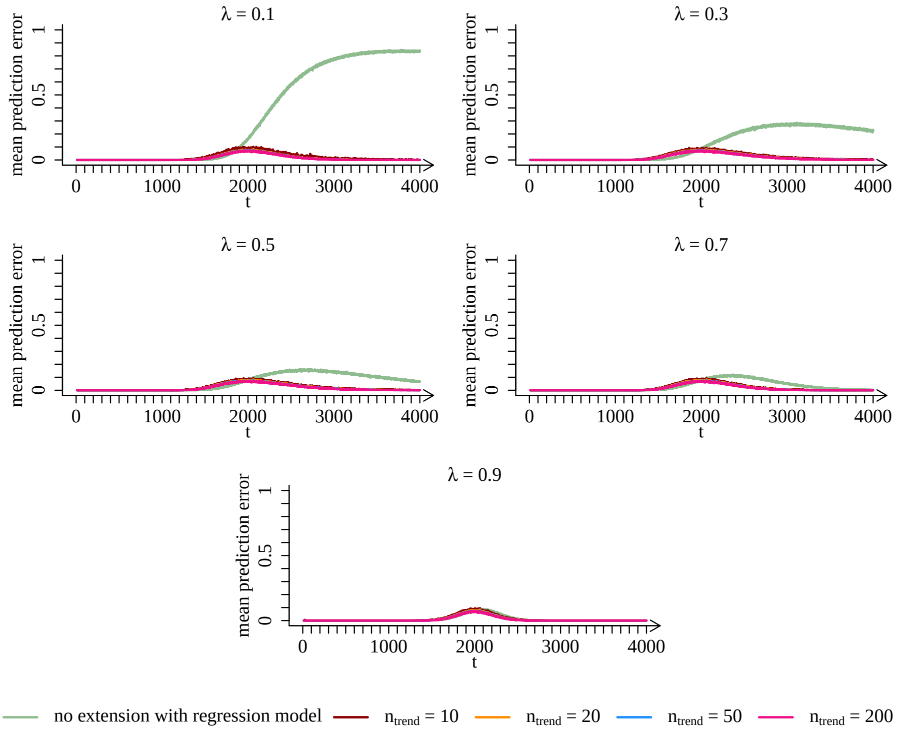

Figure 6: Exemplary mean prediction errors for fixed learning rates $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with the extension by a local linear regression model based on an interval of $n_{\text{trend}} \in \{10, 20, 50, 200\}$ recent class means and without our extension for the simulated “passing” data (see Figure 3 (c)).

However, the extension of the method for online LDA with a local linear regression model produces low mean prediction errors for all values of λ (cp. Table 3). More precisely, a higher smoothing (large n_{trend}) leads to lower mean prediction errors over time (cp. first rows in Table 3: decreasing means of mean prediction errors with decreasing standard deviations of mean prediction errors) with lower standard deviation (cp. second rows in Table 3: means of standard deviations of prediction errors over time) for all learning rates λ.

All in all the error rates just slightly vary for different values of λ and n_{trend} in this data situation. The curves (except the green ones) are very similar for all parameters. As a result, finding a good value for the learning rate is no longer crucial when the extension is applied, because the result in form of the prediction error does not that much depend on the learning rate λ anymore.

4.4 Sudden drift (d)

The data for the situation of a sudden drift is simulated analogous to the idea of drifting expectations on a circle (cp. "circular" data (a) in Section 4.1) with a difference in frequency and magnitude of drift. Contrary to the "circular" data we only focus on three changes of the underlying distribution at $t = 1000, 2000, 3000$ leading to four different concepts for the data stream. At these times the expectations not only shift one degree, but 180, 90, and at last once again 180 degrees (see Figure 3 (d)). Note that a shift of 180 degrees of both classes represents a change of class affiliation, because the expectations are interchanged.

The results in form of mean and standard deviation of the mean prediction errors over time as well as mean of standard deviations of prediction errors over time are summarized in Table 4. The mean prediction error over time for all parameters $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ and $n_{\text{trend}} \in \{10, 20, 50, 200\}$ is visualized in Figure 7.

Our proposed general extension for online discriminant analysis method for data streams with concept drift has not been constructed for situations with a sudden concept drift or structural break. Yet, we can see that the method performs quite well when including the extension of a local linear regression model to estimate the "trend". It is not necessary to "detect" the sudden drift as the extended method is able to very quickly adapt to such situations. The results in Table 4 indicate that the mean prediction error can be reduced by estimating the underlying "trend" of the expectations.

Table 4: Results of simulation study for sudden drift. Means and standard deviations (in parentheses) of mean prediction errors over time (curves in Figure 7; first row resp.) and means of standard deviations of prediction errors over time (second row resp.).

	base model (Section 2.1)		extension with		
λ	no extension	$n_{\text{trend}} = 10$	$n_{\text{trend}} = 20$	$n_{\text{trend}} = 50$	$n_{\text{trend}} = 200$
0.1	0.5037 (0.291)	0.1118 (0.036)	0.0964 (0.055)	0.0972 (0.090)	0.1311 (0.181)
	0.0808	0.0533	0.0332	0.0308	0.0327
0.3	0.5009 (0.334)	0.1113 (0.037)	0.0953 (0.054)	0.0950 (0.090)	0.1263 (0.180)
	0.0445	0.0504	0.0329	0.0297	0.0302
0.5	0.4896 (0.405)	0.1148 (0.039)	0.0973 (0.054)	0.0956 (0.089)	0.1251 (0.178)
	0.0417	0.0516	0.0343	0.0301	0.0301
0.7	0.3134 (0.314)	0.1200 (0.037)	0.1002 (0.054)	0.0973 (0.088)	0.1244 (0.173)
	0.0385	0.0549	0.0356	0.0304	0.0301
0.9	0.1581 (0.217)	0.1209 (0.037)	0.1017 (0.053)	0.0982 (0.085)	0.1187 (0.156)
	0.0326	0.0556	0.0362	0.0308	0.0303

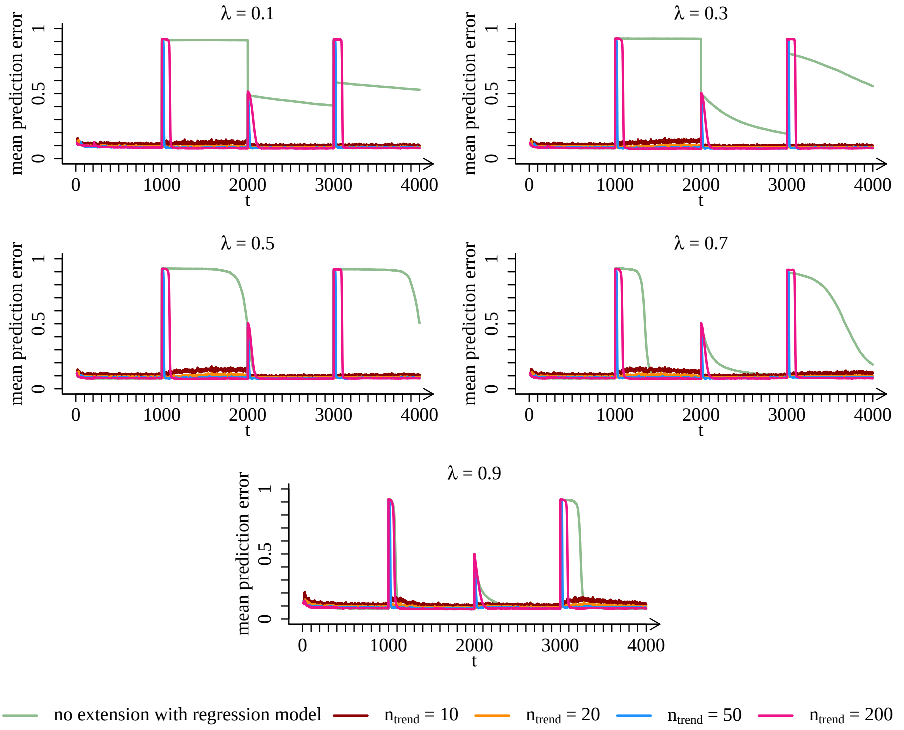

Figure 7: Exemplary mean prediction errors for fixed learning rates $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with the extension by a local linear regression model based on an interval of $n_{\text{trend}} \in \{10, 20, 50, 200\}$ recent class means and without our extension for the simulated data with sudden concept drift (see Figure 3 (d)).

Focusing on the green curves in Figure 7 it becomes clear that for most learning rates λ the method for online LDA is not able to adapt quickly to the new distribution following the sudden concept drift. The mean prediction error increases abruptly after the change of concept and decreases slowly afterwards. Only for a large learning rate of $\lambda = 0.9$ this effect is reduced.

By including the estimation of the trend of the expectations the situation can be improved. The adaption to the new distribution can be strongly accelerated. The prediction error jumps to a higher level when the distribution changes but then declines very quickly again for all learning rates λ. A higher smoothing through a higher value of n_{trend} (bigger interval) is only improving the results up to a certain point because the assumption of a linear trend is not fulfilled. Yet, up to an interval of $n_{\text{trend}} = 50$ observations the mean prediction error can be improved by a factor of 1.5–5.0, depending on the learning rate used in the method for online LDA (cp. Table 4).

5 Conclusion

We propose a general procedure for extending methods for online linear discriminant analysis (LDA) in order to improve the forecasting quality in situations where concept drift has to be considered. We try to track and forecast changing expectations of the distributions of features by a local approximation via a linear model. The maximum likelihood estimator used for the LDA model is replaced by the predicted value from the linear model. As an example, we applied this procedure to the advanced online LDA method with fixed learning rate by Kuncheva and Plumpton (2008).

Knowing that the space of possible concept drift situations is infinitely huge, we have chosen a set of very different and difficult situations (sudden and incremental drift, rotating and straight movement of class expectations, etc.). The simulation shows that even if assumptions do not hold at all, namely a violation of our assumption of a linear trend of the expectations, our extension is still capable of reducing the prediction error drastically. Because we focus on a *local* linear regression model, various types of trends can be approximated by a linear one. Last but not least, finding a good value for the learning rate is no longer crucial when the extension is applied.

References

Anagnostopoulos C, Tasoulis DK, Adams NM, Pavlidis NG, Hand DJ (2012) Online Linear and Quadratic Discriminant Analysis with Adaptive Forgetting for Streaming Classification. Statistical Analysis and Data Mining: The ASA Data Science Journal 5(2):139–166, John Wiley & Sons, Inc., New York. DOI: 10.1002/sam.10151.

Bischl B, Lang M, Mersmann O, Rahnenführer J, Weihs C (2015) BatchJobs and BatchExperiments: Abstraction Mechanisms for Using R in Batch Environments. Journal of Statistical Software 64(11):1–25. DOI: 10.18637/jss.v064.i11.

Fisher RA (1936) The Use of Multiple Measurements in Taxonomic Problems. Annals of Eugenics 7(7):179–188, John Wiley & Sons, Inc., New York. DOI: 10.1111/j.1469-1809.1936.tb02137.x.

Genz A, Bretz F (2009) Computation of Multivariate Normal and t Probabilities, 1st edn. Lecture Notes in Statistics, Springer, Berlin / Heidelberg. ISBN: 978-3-642016-88-2, DOI: 10.1007/978-3-642-01689-9.

Genz A, Bretz F, Miwa T, Mi X, Leisch F, Scheipl F, Hothorn T (2019) mvtnorm: Multivariate Normal and t Distributions. URL: `https://CRAN.R-project.org/package=mvtnorm`. R package version 1.0-10.

Golub GH, Loan CFV (1996) Matrix Computations, 3rd edn. The Johns Hopkins University Press, Baltimore, London. ISBN: 08-0185-413-X.

Hoens TR, Polikar R, Chawla NV (2012) Learning from Streaming Data with Concept Drift and Imbalance: An Overview. Progress in Artificial Intelligence 1(1):89–101, Springer. DOI: 10.1007/s13748-011-0008-0.

Kuncheva LI, Plumpton CO (2008) Adaptive Learning Rate for Online Linear Discriminant Classifiers, Lecture Notes in Computer Science, vol. 5342, Springer, Berlin, Heidelberg, pp. 510–519. DOI: 10.1007/978-3-540-89689-0_55.

Lumley T (2013) biglm: bounded memory linear and generalized linear models. URL: `https://CRAN.R-project.org/package=biglm`. R package version 0.9-1.

Pang S, Ozawa S, Kasabov N (2005) Incremental Linear Discriminant Analysis for Classification of Data Streams. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 35(5):905–914, IEEE Press, Piscataway, NJ, USA. DOI: 10.1109/TSMCB.2005.847744.

R Core Team (2018) R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. URL: `https://www.R-project.org/`.

Venables WN, Ripley BD (2002) Modern Applied Statistics with S, 4th edn. Springer, New York. DOI: 10.1007/978-0-387-21706-2.

Webb GI, Hyde R, Cao H, Nguyen HL, Petitjean F (2016) Characterizing Concept Drift. Data Mining and Knowledge Discovery 30(4):964–994, Springer US. DOI: 10.1007/s10618-015-0448-4.