Sequential Dynamic Classification Using Latent Variable Models

Technical Report PARG 08-02

Seung Min Lee and Stephen J. Roberts

Robotics Research Group
Department of Engineering Science
University of Oxford

December 2008
Sequential Dynamic Classification Using Latent Variable Models

Summary

Adaptive classification is an important online problem in data analysis. Nonlinear and nonstationary nature of data makes standard static approaches unsuitable. We propose a sequential dynamic classification algorithm that employs a nonlinear filter named unscented Kalman filter. It is based on a method of unscented transformation, which enjoys several advantages over a linearisation method. Also, we suggest using the dynamic generalised linear model to solve the adaptive classification problem. An idea of using past observed input and label information is discussed. In practice we often face a situation in which information is incomplete: we receive missing inputs and/or target labels. We carry out experiments to see how various classifiers deal with this incomplete information problem.

Acknowledgements

This work was undertaken as part of the ALADDIN (Autonomous Learning Agents for Decentralised Data and Information Systems) project and is jointly funded by a BAE Systems and EPSRC (Engineering and Physical Research Council) strategic partnership.
1 Introduction

Decision processes can be broadly divided into two: (1) a retrospective decision process in which we reconsider what has happened in the past, decisions made and their consequences and expect to make best decisions out of the past experiences when we are in similar situations in the future; and (2) a prospective decision process in which we do not possess sufficient experiences to rely on, and thus with limited information we have to make decisions in real time, hoping that those decisions will produce good future results. In the machine learning literature the former is referred to as offline learning and, the latter online learning. Interpolation is a typical example of offline learning, in that it is a method of constructing new data points within the range of a set of known data points. In time series forecasting we make new future predictions in real time when a new set of observations are received, and hence is an example of online learning.

Many data analysis problems require online learning and an important online problem is that of adaptive classification. In practice data is by nature noisy, nonlinear and nonstationary; it changes constantly over time. Often components of the data are delayed and/or missing. In these uncertain and incomplete environments making a classification in an online manner represents a significant challenge.

To tackle this challenge, we use the framework of dynamic models. State-space models are corresponding dynamic models in the engineering literature. For a binary classification problem, a logistic regression model is trained to produce posterior class probabilities. By modelling the weights of the logistic regression classifier as time-varying parameters, the nonstationary nature of the problem can be captured. Penny and Roberts [1999] and Lowne et al. [2008] employed the extended Kalman filter to solve the nonlinear dynamic model. We shall demonstrate how an efficient nonlinear filter known as the unscented Kalman filter [Julier and Uhlmann, 1997, 2004] can be applied to the adaptive classification problem. Alternatively, this problem can be thought of as a special case of dynamic generalised linear models [West and Harrison, 1997]. Binary outcomes are known to have a binomial distribution, which is modelled by a generalised linear model. By allowing the weights of the model to evolve over time, we can solve the dynamic classification problem in the dynamic generalised linear model framework.

In Section 2 we briefly review the logistic regression model and its nonstationary version using the extended Kalman filter. In Section 3 we propose an online algorithm for training the nonstationary logistic regression model using the unscented Kalman filter. Section 4 reviews the dynamic generalised linear model and we examine how the dynamic classification can be dealt with in that context. In Section 5 we suggest variants of the classification models in which extra information about inputs and labels are exploited. In Section 6 we
apply the proposed models to synthetic and real data and carry out experiments to see how these models handle incomplete information cases in which part of information is missing, and results from numerical examples are presented. Section 7 concludes the report with discussions about future work.

2 Dynamic Logistic Regression

2.1 Logistic Regression

For a binary decision problem, target labels are represented by $y = 0$ and $y = 1$. We are interested in inferring a class probability defined by

$$\pi(h) \triangleq \Pr(y = 1|h),$$

where $h$ is a given input vector. By definition, $\Pr(y = 0|h) = 1 - \pi(h)$. Logistic regression models the class probability as follows:

$$\pi(h) = l(h^Tw),$$

where $w$ is a vector of weights and $l(\cdot)$ is a logistic function defined by

$$l(h^Tw) = \frac{\exp(h^Tw)}{1 + \exp(h^Tw)}.$$ (3)

Instead of using the input vector per se, we can consider its extended form:

$$\varphi(h) = \begin{cases} h \\ \phi(h) \\ 1 \end{cases},$$

where $\phi$ represents a set of nonlinear kernel functions such as Gaussian radial basis functions. Throughout this report we consider, unless otherwise stated, a logistic function with activation $a = \varphi(h)^Tw$, not $a = h^Tw$. With this kernel input vector it is possible to find a nonlinear boundary between two classes. The benefit of adding the unity component (i.e. 1) and the linear term (i.e. $h$) into the kernel vector $\varphi(h)$ is described in Lowne et al. [2008].

2.1.1 Bayesian approach

Suppose that the weight vector $w$ has a probability distribution $p(w)$. The class probability given in Equation (2) ignores any uncertainty associated with the weight vector. A Bayesian approach takes into account its distribution, and accordingly, moderates the class probability. If it is assumed that the distribution of $w$ is a multivariate Gaussian with mean $\hat{w}$ and covariance $P_w$, then the distribution of activation $a$ is also a Gaussian
with mean \( \hat{a} \) and variance \( c^2 \) defined by

\[
\hat{a} = \varphi(h)^T \hat{w},
\]

\[
c^2 = \varphi^T(h) P_w \varphi(h).
\]

By integrating \( a \) out, the posterior class probability \( \pi \) is moderated; that is,

\[
\pi = \int l(a) p(a) da.
\]

Since it is impossible to compute the probability analytically, it is approximated by [MacKay, 1992]:

\[
\pi \approx l(\kappa(c^2) \bar{a}),
\]

where

\[
\kappa(c^2) = \left(1 + \frac{\pi c^2}{8}\right)^{-1/2}.
\]

### 2.2 Nonstationary Logistic Regression

In the stationary logistic regression model given in Equation (2), the weight vector \( w \) is assumed to be static.

Penny and Roberts [1999] proposed the dynamic logistic regression in which the weight vector evolves according to the random walk. It has the following form:

\[
\pi_t = l(\varphi(h_t)^T w_t),
\]

\[
w_t = w_{t-1} + v_t,
\]

where \( v_t \) is a state noise variable assumed to be a Gaussian with mean 0 and covariance \( q_t I \). From now on we simplify the notation \( \varphi(h_t) \) to \( \varphi_t \), and the term ‘state variable’ is preferred to ‘weight vector’.

In the dynamic logistic regression the nonstationarity is captured by time-evolving state variable \( w_t \), and the binary decision is made according to the logistic function. In order to conduct adaptive classification processes, we have to estimate the state variable \( w_t \) in an online manner. Equation (11) can be thought of as a nonlinear dynamic model in which an observation process is the nonlinear logistic function and a state process is a linear first-order Markov process. Hence, instead of the standard Kalman filter, a nonlinear Kalman filter is more suitable for solving the adaptive classification problem.

#### 2.2.1 Dynamic Classification using Extended Kalman Filter

The extended Kalman filter (e.g. Jazwinski [1970]) is the most commonly used filter for handling nonlinear systems. It is based on linearising a nonlinear function; the nonlinear function is approximated by the first
two terms of its Taylor series expansion. We shall demonstrate how the extended Kalman filter is implemented in the dynamic logistic regression. Suppose that we have the posterior mean and covariance of state variable \( w_{t-1} \), which are represented by

\[
E(w_{t-1}|D_{t-1}) = \hat{w}_{t-1|t-1}, \quad \text{(12)}
\]

\[
\text{Cov}(w_{t-1}|D_{t-1}) = P_{t-1|t-1}, \quad \text{(13)}
\]

where \( D_{t-1} = \{y_1, y_2, \ldots, y_{t-1}\} \) is a set of observed labels up to time \( t - 1 \). Since the state process is linear, the two moments of the prior distribution of the state variable at time \( t \) can be easily computed as follows:

\[
\hat{w}_{t|t-1} = \hat{w}_{t-1|t-1},
\]

\[
P_{t|t-1} = P_{t-1|t-1} + q_t I. \quad \text{(14)}
\]

Owing to the nonlinear observation function, it is impossible to compute exactly the mean and variance of the one-step ahead forecast distribution. Thus the nonlinear function is approximated by

\[
\pi_t = l(\varphi_t^T w_t) \\ \approx l(\varphi_t^T \hat{w}_{t|t-1}) + A_t^T (w_t - \hat{w}_{t|t-1}), \quad \text{(15)}
\]

where

\[
A_t \triangleq \left. \frac{\partial l(w)}{\partial w} \right|_{w_{t|t-1}} \\ = l(\varphi_t^T \hat{w}_{t|t-1}) (1 - l(\varphi_t^T \hat{w}_{t|t-1})) \varphi_t. \quad \text{(16)}
\]

The one-step ahead prediction for the class probability is computed according to

\[
\hat{\pi}_{t|t-1} = l(\varphi_t^T \hat{w}_{t|t-1}), \quad \text{(17)}
\]

and its variance is

\[
\text{Var}(\pi_t|D_{t-1}) = A_t^T P_{t|t-1} A_t + r_t, \quad \text{(18)}
\]

where \( r_t \) is a prior observation noise variance for the observed label \( y_t \). From the fact that a binary label variable has a Bernoulli distribution, we can estimate the observation noise variance according to

\[
r_t = \hat{\pi}_{t|t-1} (1 - \hat{\pi}_{t|t-1}). \quad \text{(19)}
\]

Alternatively, we can set \( r_t \) to a fixed value. So long as \( r_t > 0 \), this added observation noise variance prevents singularities in the inverse observation covariance, which is needed shortly in computing the Kalman gain.
After observing a new label $y_t$, we update the prior distribution of the state variable. The posterior mean and covariance of the state variable are computed according to

$$
\hat{w}_{t|t} = \hat{w}_{t|t-1} + K_t \left( y_t - \tilde{y}_{t|t-1} \right), \tag{20}
$$

$$
P_{t|t} = P_{t|t-1} - K_t \left( A_t^T P_{t|t-1} A_t + r_t \right) K_t^T. \tag{21}
$$

Kalman gain $K_t$ is computed as follows:

$$
K_t = P_{t|t-1} A_t \left( A_t^T P_{t|t-1} A_t + r_t \right)^{-1}. \tag{22}
$$

In the prediction context Jazwinski [1970] proposed an idea for adaptively estimating noise variances by maximising the ‘evidence’ of observations. Penny and Roberts [1999] and Lowne et al. [2008] modified the idea to make it suitable for the classification context by maximising the ‘evidence’ of model predictions instead. For example, Lowne et al. [2008] suggested updating state noise variance $q_t$ according to

$$
q_t = \max \left\{ u_{t|t} - u_{t|t-1}, 0 \right\}, \tag{23}
$$

where

$$
u_{t|t} = \tilde{y}_{t|t} \left( 1 - \tilde{\pi}_{t|t} \right). \tag{24}
$$

In this report we do not cover this issue of the adaptive state noise estimation.

## 3 Dynamic Classification using Unscented Kalman Filter

### 3.1 Unscented Kalman Filter

Julier and Uhlmann [1997, 2004] proposed a filter named unscented Kalman filter. It is based on the unscented transformation, which is a method for calculating statistics of a random variable that undergoes a nonlinear transformation and is founded on the principle that it is easier to approximate a probability distribution than an arbitrary nonlinear function [Julier and Uhlmann, 1997]. The unscented transformation resembles Monte Carlo sampling methods in that a number of points are selected and the points are propagated through a nonlinear function. However, the unscented transformation selects a set of points (referred to as sigma points) not randomly but deterministically, so that they preserve the statistics (e.g. mean and variance) of their underlying distribution. We first describe the unscented transformation.

Suppose we have two random variables $a$ and $c$ and a nonlinear function $b(\cdot)$. We would like to infer the probability density function for $c = b(a)$; we denote the means and covariances of $a$ and $c$ by $\bar{a}$, $\bar{c}$, $P_a$ and
\( \mathbf{P}_c \) respectively. We first select \( 2L + 1 \) \((L \text{ is a dimension of } a)\) sigma vectors \( \mathcal{A}_i \) in the following deterministic way:

\[
\begin{align*}
\mathcal{A}_0 &= \bar{a} \\
\mathcal{A}_i &= \bar{a} + \left( \sqrt{(L + \lambda) \mathbf{P}_a} \right)_i, \quad i = 1, \ldots, L \\
\mathcal{A}_i &= \bar{a} - \left( \sqrt{(L + \lambda) \mathbf{P}_a} \right)_i, \quad i = L + 1, \ldots, 2L
\end{align*}
\]

(25)

where \( \lambda = \alpha^2(L + \kappa) - L. \quad \left( \sqrt{(L + \lambda) \mathbf{P}_a} \right)_i \) represents the \( i \)th column of the matrix square root. \( \alpha \) and \( \kappa \) are parameters of the method; the former is a scaling parameter that determines the spread of the sigma points around \( \bar{a} \) and is normally set to a small positive value; and the latter is a secondary scaling parameter and is normally set to 0 [Wan and van der Merwe, 2001].

In order to compute the statistics of the transformed vector \( c \), the sigma vectors \( \mathcal{A}_i \) are propagated through the nonlinear function and we have another set of sigma vectors:

\[
\mathcal{C}_i = b(\mathcal{A}_i), \quad i = 0, 1, \ldots, 2L.
\]

(26)

The unscented transformation approximates the mean and covariance of \( c \) as weighted sums of the propagated sigma vectors according to

\[
\begin{align*}
\bar{c} &\approx \sum_{i=0}^{2L} \omega_i^{(m)} \mathcal{C}_i, \\
\mathbf{P}_c &\approx \sum_{i=0}^{2L} \omega_i^{(c)} (\mathcal{C}_i - \bar{c})(\mathcal{C}_i - \bar{c})^T.
\end{align*}
\]

(27)

(28)

The weights \( \omega_i \) are given by

\[
\begin{align*}
\omega_0^{(m)} &= \frac{\lambda}{L + \lambda}, \\
\omega_0^{(c)} &= \frac{\lambda}{L + \lambda} + 1 - \alpha^2 + \beta, \\
\omega_i^{(m)} &= \omega_i^{(c)} = \frac{1}{2(L + \lambda)}, \quad i = 1, \ldots, 2L
\end{align*}
\]

(29)

where \( \beta \) is another parameter that incorporates prior knowledge of the distribution of \( a \). For Gaussian distributions, \( \beta = 2 \) is optimal [Wan and van der Merwe, 2001].

In the unscented Kalman filter, the prior and posterior means and covariances of the state variable of a dynamic nonlinear model are approximated by the unscented transformation. There are two major advantages of the unscented Kalman filter over the extended Kalman filter. Firstly, whereas the linearisation method of the extended Kalman filter approximates true mean and covariance up to the first order, the mean and covariance
obtained by the unscented Kalman filter for Gaussians are accurate up to the third order for all nonlinearities, and for non-Gaussians an accuracy up to at least the second order is guaranteed [Julier and Uhlmann, 1996]. Secondly, despite the improvement of the approximation accuracy, the computational cost required by the unscented transformation is equal to the linearisation method. A comparison between the linearisation approach, the unscented transformation and a sampling method is excellently illustrated in Wan and van der Merwe [2001].

3.2 Dynamic Logistic Regression using Unscented Kalman Filter

We repeat the dynamic logistic regression model:

\[ \pi_t = l(\varphi(h_t)^T w_t), \]
\[ w_t = w_{t-1} + v_t, \]

where \( v_t \) is a Gaussian state noise with mean 0 and covariance \( q_t I \). In section 2.2.1 we derived an algorithm for solving this nonlinear system using the extended Kalman filter. We here present an algorithm in which the unscented Kalman filter is applied to the dynamic classification problem. Assuming that the posterior mean and covariance of state variable \( w_{t-1} \) are \( \hat{w}_{t-1} | t-1 \) and \( P_{t-1 | t-1} \) respectively, the two moments of the prior state distribution at time \( t \) are

\[ \hat{w}_{t | t-1} = \hat{w}_{t-1 | t-1}, \]
\[ P_{t | t-1} = P_{t-1 | t-1} + q_t I. \]

In order to compute the mean and variance of the one-step ahead prediction distribution, the unscented transformation method is employed. We first draw \((2L + 1)\) sigma vectors for the distribution of \( w_t \) from its estimated mean and covariance, \( \hat{w}_{t | t-1} \) and \( P_{t | t-1} \):

\[ \mathcal{W}_{t | t-1} = \left[ \mathcal{W}_{t-1}^0, \ldots, \mathcal{W}_{t-1}^i, \ldots, \mathcal{W}_{t-1}^{2L} \right] \]
\[ = \left[ \hat{w}_{t | t-1}, \hat{w}_{t | t-1} + \sqrt{(L + \lambda)P_{t | t-1}}_i, \hat{w}_{t | t-1} - \sqrt{(L + \lambda)P_{t | t-1}}_i \right], \]

where \( L \) is the dimension of \( w \), \( \lambda = \alpha^2(L + \kappa) - L \), and \( \sqrt{(L + \lambda)P_{t | t-1}}_i \) is the \( i \)th column of the matrix square root. The sigma vectors are propagated through the logistic function \( l(\cdot); \)

\[ \Pi_{t | t-1}^i = l \left( \varphi^T \left( \mathcal{W}_{t | t-1}^i \right) \right), \quad i = 0, 1, \ldots, 2L \]
Using these propagated sigma vectors we can compute a one-step ahead predicted class probability and its variance, which are given by

\[
\hat{\pi}_{t|t-1} = \sum_{i=0}^{2L} C^{(m)}_i \Pi^i_{t|t-1}
\]

\[
P_{yy} = \sum_{i=0}^{2L} C^{(c)}_i (\Pi^i_{t|t-1} - \hat{\pi}_{t|t-1})^2 + r_t.
\]

The prior observation noise variance \(r_t\) can be either fixed to a static small value or estimated according to

\[
r_t = \hat{\pi}_{t|t-1}(1 - \hat{\pi}_{t|t-1}).
\]

In addition, we compute the covariance between the state variable and the class probability according to

\[
P_{wy} = \sum_{i=0}^{2L} C^{(c)}_i (W^i_{t|t-1} - \hat{w}_{t|t-1})(\Pi^i_{t|t-1} - \hat{\pi}_{t|t-1})^T.
\]

Weights \(C^{(m)}_i\) and \(C^{(c)}_i\) are given by

\[
C^{(m)}_0 = \frac{\lambda}{L + \lambda},
\]

\[
C^{(c)}_0 = \frac{\lambda}{L + \lambda} + 1 - \alpha^2 + \beta,
\]

\[
C^{(m)}_i = C^{(c)}_i = \frac{1}{2(L + \lambda)}, \quad i = 1, \ldots, 2L.
\]

After observing a new label \(y_t\), the state variable distribution is updated. The posterior mean and covariance of \(w_t\) are computed according to

\[
\hat{w}_{t|t} = \hat{w}_{t|t-1} + K_t (y_t - \hat{\pi}_{t|t-1})
\]

\[
P_{t|t} = P_{t|t-1} - K_t P_{yy} K_t^T,
\]

where the Kalman filter is

\[
K_t = P_{wy} P_{yy}^{-1}.
\]

In Section 2.1.1 we introduced the Bayesian approach to the logistic regression model in order to take into account the uncertainty associated with the state variable \(w\). Note that the unscented Kalman filter concerns the distribution of \(w\) by selecting sigma points from the distribution.

## 4 Dynamic Generalised Linear Model

The dynamic classification problem can be viewed from a different perspective. Generalised linear models are a powerful method for analysing any distributions in the exponential family. They can be generalised to capture nonstationary nature of data by allowing model parameters to evolve over time, which is known as dynamic.
4 Dynamic Generalised Linear Model

generalised linear models. Since the binary label follows a Bernoulli distribution, which is a special case of a binomial distribution, the dynamic classification problem can be solved in the framework of the dynamic generalised linear model. In this section we review the generalised linear model and the dynamic generalised linear model in the Bayesian perspective. Then, we look in more detail at the dynamic binomial model that provides a solution to the dynamic classification problem.

4.1 Generalised Linear Models

In an ordinary linear regression model an observation random variable $y$ is modelled by

$$ y = \mu + n, \quad (43) $$

where $\mu = h^T w$. Note that $h$ represents a known input variable (referred to as predictor variable) and $w$ is a model parameter vector. In addition, $n$ is a Gaussian error variable with mean 0 and variance $\sigma^2$. The observation variable is also a Gaussian distribution with the following mean and variance

$$ E(Y) = \mu, \quad \text{Var}(Y) = \sigma^2. \quad (44) $$

Note that the mean is a linear function of the predictor variable, $h$.

In a generalised linear model two generalisations are made from the standard linear model above: (1) the probability distribution of the observation variable is not restricted to be a Gaussian but is allowed to have any distribution in the exponential family; (2) it is not the mean of the variable but a function of the mean that is linearly related to the predictor variable.

An observation random variable $y$ that has a distribution in the exponential family takes the form,

$$ p(y|\theta, \phi) = \exp \left( \frac{d(y)\theta - b(\theta)}{a(\phi)} + c(y, \phi) \right), \quad (45) $$

where $\theta$ is called canonical parameter and $\phi$ dispersion parameter, and $a, b, c, d$ are all known functions. The mean and variance of the variable are related to the canonical and dispersion parameters through the known functions as follows:

$$ E(d(y)|\theta, \phi) = \mu = b'(\theta), \quad (46) $$

$$ \text{Var}(d(y)|\theta, \phi) = b''(\theta)a(\phi). \quad (47) $$

Members of the exponential family involve Gaussian, binomial, Poisson, exponential, gamma, inverse Gaussian distributions, etc.
In the ordinary Gaussian linear model the mean is linearly related to the predictor variable, $h$; that is,

$$\mu = h^T w.$$  \hfill (48)

The right-hand side of the equation is denoted by $\eta \triangleq h^T w$, and $\eta$ is referred to as linear predictor. Note that the linear predictor and the mean of a Gaussian variable can take any real values. However, this may or may not be true for other distributions in the exponential family; for instance, the mean of a Poisson distribution must be only positive. Hence, we need a way of relating the linear predictor to the mean, which is given by

$$g(\mu) = \eta = h^T w.$$  \hfill (49)

The function $g$ is referred to as link function. The mean can be computed simply by inverting the link function:

$$\mu = g^{-1}(\eta) = g^{-1}(h^T w).$$  \hfill (50)

Comparing Equation (48) with Equation (49), we can see that the link function for the standard Gaussian linear model is an identity function. A particular form of the link function is of importance. A link function that makes the linear predictor $\eta$ equal to the canonical parameter $\theta$ is referred to as canonical link:

$$g(\mu) = \eta = \theta.$$  \hfill (51)

The canonical link is important due to the existence of minimal sufficient statistics for the model parameter [McCullagh and Nelder, 1989], thereby being used in most cases.

### 4.2 Exponential Family Dynamic Models

Summing up the generalised linear model, there are two key characteristics: (1) the non-Gaussianity of an observation variable and (2) the nonlinear relation of the mean to a predictor variable. We can view a dynamic generalised linear model in two different ways. Firstly, it is a “generalised” version of the dynamic linear model that allows an observation variable to be non-Gaussian. Secondly, it is a “dynamic” version of the generalised linear model that allows model parameters to vary over time. We here introduce the dynamic generalised linear model for a scalar observation variable.

Suppose that an observation variable $y$ that has a distribution in the exponential family is observed over time. The observation model of a dynamic generalised linear model is of form

$$p(y_t|\theta_t, \phi_t) = \exp \left( \frac{d(y_t)\theta_t - b(\theta_t)}{a(\phi_t)} + c(y_t, \phi_t) \right),$$  \hfill (52)

with the following link equation

$$g(\mu_t) = \eta_t = h_t^T w_t,$$  \hfill (53)
where $\mu_t = E(y_t)$. If $y_t$ has a Gaussian distribution, then the observation model and link equation can be merged into a single observation equation as follows:

$$y_t = h_t^T w_t + n_t,$$

where $n_t \sim N(0, r_t)$ and $\mu_t = h_t^T w_t$. This is equivalent to an observation equation of a dynamic linear model.

The state variable (or model parameter vector) $w_t$ is assumed not to be static but to evolve over time; the state process is

$$w_t = F_t w_{t-1} + v_t,$$  \hfill (55)

where $v_t \sim (0, Q_t)$. The notation ‘$a \sim (b, c)$’ represents that the mean and variance of $a$ are $b$ and $c$ respectively without specifying the probability density function of the variable $a$.

We review a sequential inference method for the state variable in the Bayesian setting [West and Harrison, 1997]. It is an approximate method in a sense that distributions are only specified in terms of their first and second moments without full distributional information. Suppose that, given $D_{t-1}$, state variable $w_{t-1}$ has mean $\hat{w}_{t-1} | t-1$ and covariance $P_{t-1 | t-1}$; that is,

$$w_{t-1} | D_{t-1} \sim (\hat{w}_{t-1} | t-1, P_{t-1 | t-1}).$$  \hfill (56)

Since the state process of Equation (55) is linear, the prior distribution of the state variable at $t$ can be easily computed and it is of form

$$w_t | D_{t-1} \sim (\hat{w}_t | t-1, P_{t | t-1}),$$  \hfill (57)

where

$$\hat{w}_{t | t-1} = F_t \hat{w}_{t-1 | t-1};$$  \hfill (58)

$$P_{t | t-1} = F_t P_{t-1 | t-1} F_t^T + Q_t.$$  \hfill (59)

Remind that what is linearly related to the state variable is not the mean of the observation variable but the linear predictor (i.e. $\eta_t = g(\mu_t) = h_t^T w_t$). Owing to the linearity, we can readily compute the prior distribution of $\eta_t$; given $D_{t-1}$ the joint distribution of $\eta_t$ and $w_t$ is

$$\begin{bmatrix} \eta_t \\ w_t \end{bmatrix} | D_{t-1} \sim \begin{pmatrix} \hat{\eta}_{t | t-1} \\ \hat{w}_{t | t-1} \end{pmatrix}, \begin{pmatrix} S_{t | t-1} & h_t^T P_{t | t-1} \\ P_{t-1 | t-1} & P_{t | t-1} \end{pmatrix},$$  \hfill (60)

where

$$\hat{\eta}_{t | t-1} = h_t^T \hat{w}_{t-1 | t-1},$$  \hfill (61)

$$S_{t | t-1} = h_t^T P_{t-1 | t-1} h_t.$$  \hfill (62)
The one-step ahead forecast distribution (i.e. the distribution of $y_t$ given $D_{t-1}$) can be obtained by marginalising over the canonical parameter assuming that the dispersion parameter is known:

$$p(y_t|D_{t-1}) = \int p(y_t|\theta_t)p(\theta_t|D_{t-1})d\theta_t.$$  \hfill (63)

To solve the equation we need to specify the prior distribution of the canonical parameter, $p(\theta_t|D_{t-1})$. A conjugate prior is widely used because it allows us to derive the forecast distribution analytically. The conjugate prior distribution is of form

$$p(\theta_t|D_{t-1}) = \omega(k_t, m_t) \exp(k_t \theta_t - m_t b(\theta_t)),$$  \hfill (64)

where $\omega$ is a known function providing the normalising constant, and $k_t$ and $m_t$ are hyperparameters of the conjugate prior.

A complete form of the one-step ahead forecast distribution is obtained in the following way:

$$p(y_t|D_{t-1}) = \int \omega(k_t, m_t) \exp(k_t \theta_t - m_t b(\theta_t))$$

$$\int \exp \left( \frac{d(y_t) \theta_t - b(\theta_t)}{a(\phi_t)} + c(y_t, \phi_t) \right) d\theta_t$$

$$= \omega(k_t, m_t) \exp(c(y_t, \phi_t))$$

$$\int \exp \left( \theta_t - \left( m_t + \frac{1}{a(\phi_t)} \right) b(\theta_t) \right) d\theta_t$$

$$= \omega \left( k_t + \frac{d(y_t)}{a(\phi_t)}, m_t + \frac{1}{a(\phi_t)} \right) \exp \left( c(y_t, \phi_t) \right).$$  \hfill (65)

In order to employ the one-step ahead forecast distribution we need to determine the values of the hyperparameters of the conjugate prior, $k_t$ and $m_t$. If a canonical link is used, which ensures $\theta_t = \eta_t$, then we know that

$$E(\theta_t|D_{t-1}) = E(\eta_t|D_{t-1}) = \hat{\eta}|t-1, \hfill (66)$$

$$\text{Var}(\theta_t|D_{t-1}) = \text{Var}(\eta_t|D_{t-1}) = S|t-1. \hfill (67)$$

From the form of the conjugate prior in Equation (64), we can obtain $E(\theta_t|D_{t-1})$ and $\text{Var}(\theta_t|D_{t-1})$ in terms of the hyperparameters. In addition, we know the values of $\hat{\eta}|t-1$ and $S|t-1$, which are given in Equation (62). We can, therefore, calculate the values of the hyperparameters.

According to the Bayes’ theorem, the posterior distribution of the canonical parameter $\theta_t$ can be represented by

$$p(\theta_t|D_t) \propto p(\theta_t|D_{t-1})p(y_t|\theta_t).$$  \hfill (68)
Its complete distributional form is
\[
p(\theta_t|D_t) = \omega \left( k_t + \frac{d(y_t)}{a(\phi_t)}, m_t + \frac{1}{a(\phi_t)} \right) \exp \left( \left( k_t + \frac{d(y_t)}{a(\phi_t)} \right) \theta_t - \left( m_t + \frac{1}{a(\phi_t)} \right) b(\theta_t) \right).
\] (69)

Since using the canonical link makes the posterior distribution of \( \eta_t \) equivalent to that of \( \theta_t \), we have the posterior distribution of \( \eta_t \) of from
\[
\eta_t|D_t \sim (\hat{\eta}_t, S_{t|t}),
\] (70)
where
\[
\hat{\eta}_t = E(\eta_t|D_t) = E(\theta_t|D_t),
\] (71)
\[
S_{t|t} = \text{Var}(\eta_t|D_t) = \text{Var}(\theta_t|D_t).
\] (72)

Note that \( E(\theta_t|D_t) \) and \( \text{Var}(\theta_t|D_t) \) can be computed from Equation (69).

The posterior distribution of the state variable is rewritten as
\[
p(w_t|D_t) = \int p(\eta_t, w_t|D_t) d\eta_t
\]
\[
\propto \int p(\eta_t, w_t|D_{t-1}) p(y_t|\eta_t) d\eta_t
\]
\[
\propto \int p(w_t|\eta_t, D_{t-1}) p(\eta_t|D_{t-1}) p(y_t|\eta_t) d\eta_t
\]
\[
\propto \int p(w_t|\eta_t, D_{t-1}) p(\eta_t|D_t) d\eta_t
\] (73)

The first two moments of \( p(\eta_t|D_t) \) are given in Equation (72). The mean and covariance of \( p(w_t|\eta_t, D_{t-1}) \) are estimated using the linear Bayesian estimation [West and Harrison, 1997]. They are given by
\[
\hat{E}(w_t|\eta_t, D_{t-1}) = \tilde{w}_{t|t-1} + \frac{P_{t|t-1}^T h_t}{S_{t|t-1}} (\eta_t - \hat{\eta}_{t|t-1}),
\] (74)
\[
\text{Var}(w_t|\eta_t, D_{t-1}) = P_{t|t-1} - \frac{P_{t|t-1}^T h_t}{S_{t|t-1}} h_t^T P_{t|t-1}.
\] (75)

Using the two moments of \( p(w_t|\eta_t, D_{t-1}) \) and \( p(\eta_t|D_t) \), we can compute the posterior mean and covariance of \( w_t \) as follows:
\[
\tilde{w}_{t|t} = E(w_t|D_t) = E[E(w_t|\eta_t, D_{t-1})|D_t]
\]
\[
= E \left[ \tilde{w}_{t|t-1} + \frac{P_{t|t-1}^T h_t}{S_{t|t-1}} (\eta_t - \hat{\eta}_{t|t-1}) \Big| D_t \right]
\]
\[
= \tilde{w}_{t|t-1} + \frac{P_{t|t-1}^T h_t}{S_{t|t-1}} (E(\eta_t|D_t) - \hat{\eta}_{t|t-1})
\]
\[
= \tilde{w}_{t|t-1} + \frac{P_{t|t-1}^T h_t}{S_{t|t-1}} (\hat{\eta}_t - \hat{\eta}_{t|t-1}),
\] (76)
and
\[
P_{t|t} = \text{Var}(w_t|D_t) = \text{Var}[E(w_t|\eta_t, D_{t-1})|D_t] + E[\text{Var}(w_t|\eta_t, D_{t-1})|D_t]
\]
\[
= \text{Var} \left[ \hat{w}_{t|t-1} + \frac{P_{t|t-1}^T h_t}{S_{t|t-1}} (\eta_t - \hat{\eta}_{t-1}) \right] D_t
\]
\[
+ E \left[ P_{t|t-1} - \frac{P_{t|t-1}^T h_t}{S_{t|t-1}} h_t^T P_{t|t-1} \right] D_t
\]
\[
= \frac{P_{t|t-1}^T h_t}{(S_{t|t-1})^2} h_t^T P_{t|t-1} \text{Var}(\eta_t|D_t) + P_{t|t-1} - \frac{P_{t|t-1}^T h_t}{S_{t|t-1}} h_t^T P_{t|t-1}
\]
\[
= P_{t|t-1} - \frac{P_{t|t-1}^T h_t}{S_{t|t-1}} h_t^T P_{t|t-1} \left( 1 - \frac{S_{t|t}}{S_{t|t-1}} \right). \quad (77)
\]

### 4.3 Dynamic Classification using the Dynamic Binomial Model

The total number of ‘successes’ \( y \) in \( n \) independent experiments follows a binomial distribution with probability of success \( \pi \). The binomial distribution is a member of the exponential family and it is defined by
\[
p(y|n, \pi) = \binom{n}{y} \pi^y (1 - \pi)^{n-y}, \quad (78)
\]
where \( 0 \leq \pi \leq 1 \), \( n = \{0, 1, 2, \ldots\} \) and \( y = \{0, 1, 2, \ldots, n\} \). We can rearrange the above density function in the following way:
\[
p(y|n, \pi) = \exp \left( n \left[ \frac{y}{n} \log \left( \frac{\pi}{1 - \pi} \right) - \log \left( \frac{1}{1 - \pi} \right) \right] + \log \left( \frac{n}{y} \right) \right), \quad (79)
\]
Comparing this with the general form of distributions in the exponential family,
\[
p(y|\theta, \phi) = \exp \left( \frac{d(y)\theta - b(\theta)}{a(\phi)} + c(y, \phi) \right), \quad (80)
\]
we can find that \( \theta = \log \left( \frac{\pi}{1 - \pi} \right), a(\phi) = \frac{1}{n}, b(\theta) = \log(1 + \exp(\theta)), c(y, \phi) = \log \left( \frac{n}{y} \right) \) and \( d(y) = \frac{y}{n} \). The mean and variance of the binomial variable are
\[
E \left( \frac{y}{n} | n, \pi \right) = \mu = \pi, \quad (81)
\]
\[
\text{Var} \left( \frac{y}{n} | n, \pi \right) = \frac{1}{n} \pi (1 - \pi). \quad (82)
\]
For the binomial distribution, the canonical link, which makes the linear predictor \( \eta \) equal to the canonical parameter \( \theta \) is the logit function (the inverse of the logistic function).

To solve the dynamic binary classification problem we consider the dynamic binomial model, in which time index \( t \) is added to \( y \) and \( \pi \) in Equation (78). The link equation is defined by
\[
g(\mu_t) = \eta_t = h_t^T w_t. \quad (83)
\]
In addition, we assume that the state variable $w_t$ evolves according to
\[ w_t = w_{t-1} + v_t, \]  
(84)
where $v_t \sim (0, q_t I)$. The following derivation of the dynamic binomial model may be better understood with reference to the previous section.

Suppose that we have the posterior mean and covariance of $w_{t-1}$, given a set of observations $D_{t-1} = \{y_1, \ldots, y_{t-1}\}$. They are denoted by $\hat{w}_{t-1|t-1}$ and $P_{t-1|t-1}$ respectively. Owing to the linear state process, the prior distribution of the state variable is
\[ w_t | D_{t-1} \sim (\hat{w}_{t-1|t-1}, P_{t|t-1}), \]  
(85)
where \( \hat{w}_{t-1|t-1} = \hat{w}_{t-1|t-1} \), \( P_{t|t-1} = P_{t|t-1} + q_t I \).

The mean and variance of the prior distribution of the linear predictor can be easily computed by
\[ \eta_{t|t-1} = h_T \hat{w}_{t|t-1}, \]  
(88)
\[ r_{t|t-1} = h_T P_{t|t-1} h. \]  
(89)

To derive the one-step ahead forecast distribution given by
\[ p(y_t | D_{t-1}) = \int p(y_t | \theta_t) p(\theta_t | D_{t-1}) d\theta_t, \]  
(90)
we need to specify the conjugate prior, $p(\theta_t | D_{t-1})$. According to the derivation given in Appendix A.1, we can find that the prior conjugate distribution is
\[ p(\theta_t | D_{t-1}) = \frac{\Gamma(m_t)}{\Gamma(k_t)} \frac{\exp(k_t \theta_t)}{(1 + \exp(\theta_t))^{m_t}}, \]  
(91)
and its first two moments are approximately
\[ E(\theta_t | D_{t-1}) \approx \log \left( \frac{k_t}{m_t - k_t} \right), \]  
(92)
\[ \text{Var}(\theta_t | D_{t-1}) \approx \frac{1}{k_t} + \frac{1}{m_t - k_t}. \]  
(93)

We know that the mean and variance of the canonical parameter are equivalent to that of the linear predictor because a canonical link is used. Hence, we can calculate the values of the hyperparameters as follows:
\[ k_t = \frac{1}{r_{t|t-1}} (1 + \exp(\hat{\eta}_{t|t-1})), \]  
(94)
\[ m_t = \frac{1}{r_{t|t-1}} (1 + \exp(-\hat{\eta}_{t|t-1}))(1 + \exp(\hat{\eta}_{t|t-1})). \]  
(95)
With these hyperparameter values, the one-step ahead forecast distribution can be analytically obtained as

\[ p(y_t|D_{t-1}) = \int_{-\infty}^{\infty} p(y_t|\theta_t)p(\theta_t|D_{t-1})d\theta_t, \]

\[ = \frac{\Gamma(m_t)}{\Gamma(k_t)\Gamma(m_t-k_t)} \frac{\Gamma(k_t+y_t)\Gamma(m_t-k_t+n-y_t)}{\Gamma(m_t+n)} \binom{n}{y_t}. \]  

(96)

Its derivation is given in Appendix A.2. This is a beta-binomial distribution with parameters \( k_t \) and \( m_t - k_t \), and accordingly the mean and variance of the forecast distribution are

\[ E(y_t|D_{t-1}) = \frac{nk_t}{m_t}, \]

(97)

\[ \text{Var}(y_t|D_{t-1}) = \frac{nk_t(m_t-k_t)(m_t+n)}{m_t^2(1+m_t)}. \]  

(98)

After observing a new datum \( y_t \), we update the distributions of \( \theta_t \) and \( \eta_t \). The posterior distribution of \( \theta_t \) is of form

\[ p(\theta_t|D_t) \propto p(\theta_t|D_{t-1})p(y_t|D_t) \]

\[ = \exp\left((k_t + y_t)\theta_t - (m_t + n)\log(1 + \exp(\theta_t))\right). \]  

(99)

The approximate values of the mean and variance of the posterior distribution are

\[ E(\theta_t|D_t) \approx \log\left(\frac{k_t + y_t}{m_t - k_t + n - y_t}\right), \]

(100)

\[ \text{Var}(\theta_t|D_t) \approx \frac{1}{k_t + y_t} + \frac{1}{m_t - k_t + n - y_t}. \]  

(101)

They are equivalent to the mean and variance of the posterior distribution of \( \eta_t \), which are represented by \( \hat{\eta}_{t|t} \) and \( r_{t|t} \) respectively. Finally, the posterior mean and covariance of \( w_t \) are

\[ \hat{w}_{t|t} = \hat{w}_{t|t-1} + \frac{P_{t|t-1}^T h_t}{r_{t|t-1}}(\hat{\eta}_{t|t} - \hat{\eta}_{t|t-1}), \]

(102)

\[ P_{t|t} = P_{t|t-1} - \frac{P_{t|t-1}^T h_t}{r_{t|t-1}} h_t^T P_{t|t-1} \left(1 - \frac{r_{t|t}}{r_{t|t-1}}\right). \]  

(103)

The dynamic classification problem can be solved by a special case of the dynamic binomial model in which the number of trials, \( n_t \), is set to 1 and the number of success, \( y_t \), is either 0 or 1. Hence, the expected value of the observation, \( E(y_t) \), is equal to the success probability, \( \pi_t \). In the approach of using the nonlinear Kalman filters discussed in Sections 2.2.1 and 3.2, the class (or success) probability \( \pi_t \) is directly predicted from inputs via the logistic function. However, in the dynamic binomial model approach it is predicted in a full Bayesian manner, allowing for the explicit distribution form of the class variable and its conjugate prior distribution. We shall compare performances of these methods in a real-world dynamic classification problem later in the report.
5 Making Use of Past Information As Inputs

The dynamic classification models that have so far been studied consider a one-to-one mapping between the current input vector and the current class observation. We suggest using past information (i.e. past inputs and past labels) to improve the classification performance of the dynamic models. Firstly, in the prediction context it is not uncommon to use past inputs in predicting the current output. This can be thought of as a “wide-sense” Markov process. The new input vector consisting of the current input and $p$ previously observed inputs is of form

$$i_{t,p} = [h_t^T, \ldots, h_{t-p}^T]^T.$$  \hspace{1cm} (104)

We refer to $p$ as look-back parameter. Hence, the observation process of the dynamic logistic regression has the form

$$\pi_t = l \left( \varphi(i_{t,p})^T w_t \right),$$  \hspace{1cm} (105)

and for the dynamic binomial model the link function is

$$\eta_t = \varphi(i_{t,p})^T w_t.$$  \hspace{1cm} (106)

The mapping of multiple quantities of input information onto the current output may reveal hidden patterns between past inputs and the current label and consequently produce better classifications.

Secondly, we can think of incorporating previously observed labels as well as past inputs into the current input vector; the new input vector is of form

$$d_{t,p} = [h_t^T, \ldots, h_{t-p}^T, y_{t-1}, \ldots, y_{t-p}]^T.$$  \hspace{1cm} (107)

By allowing for past labels any unknown patterns between the past labels and the current label could be detected via the new input variable. The dynamic classification models can be readily employed with kernel vectors of the new inputs, $\varphi(i_{t,p})$ and $\varphi(d_{t,p})$.

6 Results

We experiment three dynamic classification models: (1) the dynamic logistic regression using the extended Kalman filter, (2) the dynamic logistic regression using the unscented Kalman filter, and (3) the dynamic binomial model. We refer to them as ‘DLR-E’, ‘DLR-U’ and ‘DBM’ respectively. We shall be interested in making classifications sequentially not only when input and label information is completely given, but also when part of the information is missing or incomplete.
6 Results

6.1 Synthetic Data

We consider two overlapping Gaussian distributions rotating in a circular fashion around a central point at $[0, 0]$, with the two distributions out of phase by $\pi$ radians. Target labels are interleaved, i.e. $\{0, 1, 0, 1, \ldots\}$ (Figure 1). Three data sets are constructed with different Bayes error (Figure 2). The data is as presented in Lowne et al. [2008]. Owing to the nonstationary nature of the data any static classifier would not be suitable, whereas our dynamic classifiers adapt the change of the environment and produce a dynamic decision boundary. In the experiments the kernel input vector is set to

$$\varphi(h_t) = \begin{bmatrix} 1 \\ h_t \end{bmatrix},$$

(108)

i.e. with no basis functions, because we know that the underlying boundary is linear. The vector $h_t$ represents an input at time $t$. The state evolution noise variance $q_t$ is set to 0.1, and for DLR-E and DLR-U the observation noise variance $r_t$ is adaptively estimated.

![Figure 1: Two Gaussian distributions rotating in a circular fashion over time.](image1)

![Figure 2: Three data sets of two rotating Gaussians with different Bayes error.](image2)

When presented with a full set of inputs and labels, we computed one-step ahead predictions of the
class probability (i.e. $\hat{\pi}_{t|t-1}$) with the adaptive classifiers. The predicted label $\hat{y}_t$ is determined with a decision threshold set to 0.5: if $\hat{\pi}_{t|t-1} > 0.5$, then $\hat{y}_t$ is 1, or 0 otherwise. Comparing the predicted labels with the true labels, the classifiers achieved performances of near the Bayes errors.

This data was further analysed by augmenting the input vector at time $t$ with past inputs and past labels, as described in Section 5. Instead of the current input vector $h_t$, we used $i_{t,p} = [h_t^T, \ldots, h_{t-p}^T]^T$ or $d_{t,p} = [h_t^T, \ldots, h_{t-p}^T, y_{t-1}, \ldots, y_{t-p}]^T$, where $y_{t-k}$ represents an observed label at time $t - k$. As the value of the look-back parameter $p$ we chose that which is the smallest among values that performed best. Comparing the classification performances we found that for the data with 4% Bayes error, classifiers using $i_{t,p}$ achieved the perfect classification, and for the data with 22% Bayes error, an accuracy improvement of $13.4\% \sim 15.9\%$.

In addition, when the classifiers used $d_{t,p}$ as inputs, classification error was uniformly zero (Table 1). These performance improvements are not surprising if we consider how the data is created: the input values rotate slowly over time and the true labels alternate between 0 and 1 every time step. The patterns within the inputs or the labels are implicitly captured by the augmented input vectors.

<table>
<thead>
<tr>
<th>Input vector</th>
<th>Classifier</th>
<th>0% Bayes error</th>
<th>4% Bayes error</th>
<th>22% Bayes error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_t$</td>
<td>DLR-E</td>
<td>0.9980</td>
<td>0.9750</td>
<td>0.7740</td>
</tr>
<tr>
<td></td>
<td>DLR-U</td>
<td>0.9990</td>
<td>0.9680</td>
<td>0.7930</td>
</tr>
<tr>
<td></td>
<td>DBM</td>
<td>0.9970</td>
<td>0.9650</td>
<td>0.7720</td>
</tr>
<tr>
<td>$i_{t,{4/6/3}}$</td>
<td>DLR-E</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9198</td>
</tr>
<tr>
<td></td>
<td>DLR-U</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9268</td>
</tr>
<tr>
<td></td>
<td>DBM</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9308</td>
</tr>
<tr>
<td>$d_{t,{2/4/8}}$</td>
<td>DLR-E</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td></td>
<td>DLR-U</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td></td>
<td>DBM</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
</tbody>
</table>

Table 1: Proportion of correct classification; $h_t$ is the current input vector; $i_{t,p} = [h_t^T, h_{t-1}^T, \ldots, h_{t-p}^T]^T$ and $d_{t,p} = [h_t^T, h_{t-1}^T, \ldots, h_{t-p}^T, y_{t-1}, \ldots, y_{t-p}]^T$ are augmented input vectors. For instance, $d_{t,\{2/4/8\}}$ denotes that the value of the look-back parameter $p$ for the data sets with 0%, 4% and 22% Bayes errors is 2, 4 and 8 respectively.

### 6.1.1 Missing Labels

In practice it is common that we receive label information sparsely, i.e. labels are missing. We are required nevertheless to make a decision. An important feature of a Bayesian dynamic model is that it can cope easily with missing observations. When class labels are not known, we may infer the missing label via the predictive
distribution of the dynamic model. In this case we have

\[ \tilde{y}_t = \Pr(y_t = 1|\mathbf{h}_t) = \hat{\pi}_{t|t-1} \]  

(109)

as a “quasi-target” in the place of a true target \( y_t \). This approach allows us not to discard information contained within the input vector \( \mathbf{h}_t \) such as slow drift. However, as Lowne et al. [2008] pointed out, treating the quasi-targets as if they were true causes a risk of a classifier becoming excessively confident in predictions. To compensate for this illusory knowledge when lacking feedback, they proposed folding the uncertainty in the class probability predicted from the input alone into the state noise. We therefore update \( q_t \) using

\[ q_t = q + \hat{\pi}_{t|t-1}(1 - \hat{\pi}_{t|t-1}), \]  

(110) 

for the dynamic logistic regression based algorithms, and

\[ q_t = q + \text{Var}(y_t|D_{t-1}), \]  

(111)

for the dynamic binomial model. The quantity \( q \) is a pre-set value for the stationary noise variance and \( \text{Var}(y_t|D_{t-1}) \) is the variance of the one-step ahead forecasting distribution given in Equation (98). We thereby ensure that the classifier does not become overly confident too quickly based on fictitious feedback.

We carried out experiments in which label information is successively randomly removed from 0% to 100% within each data stream. The performances of the classifiers were evaluated over 100 runs (Figure 3). We can see that the classification performances of the models did not worsen in proportion to the number of unobserved labels. The performances with 50% labelling were degraded by 0.13% ~ 0.31% for the data with zero Bayes error, by 0.99% ~ 1.17% for the data with 4% Bayes error, and by 0.6% ~ 3.8% for the data with 22% Bayes error; with only 20% class labels available the classifiers’ performances were degraded by only 1.5% ~ 5.2% in comparison with 100% labelling. This result indicates that an adaptive classifier can deal with sparsely observed labels and maintain its classification performance. This leads the way to enabling the adaptive classifier to decide for itself whether or not to request a label. This problem will be dealt with later in this section when we consider active label requesting.

### 6.2 Mountain Fire Scenario

We consider a fire in a wooded mountain with several villages (denoted by red dots) in the nearby region (Figure 4). In order to allocate limited resources effectively we must classify if each of the villages is in potential danger according to local weather conditions measured by weather sensors. For example, the stronger the wind speed is and the higher the air temperature is, the more likely it is for a village on the bottom left corner (denoted by
a light blue ring) to be in danger. In addition, unless the wind blows in a northerly direction, the village is more vulnerable to the spread of fire. With weather data collected by a network of weather sensors [Chan, 2000], we created a data set consisting of three input variables (wind speed, wind direction and air temperature) and a binary class variable (‘danger’ or ‘no danger’).

The manner in which inputs and target labels of the data are distributed means that no static classifier is able to separate the two groups (the left panel of Figure 5). The true labels alternate between ‘danger’ and ‘no danger’ at irregular intervals as the weather variables change (the right panel of Figure 5). For the sake of comparison, we ran a standard logistic regression on the data set; 100 input-output pairs were randomly taken from the full data set to use as a test set. The performance of this static model was averaged over 100 runs. When the original input vector, \( h_t = [(\text{wind speed})_t; (\text{wind direction})_t; (\text{air temperature})_t] \), is used, our three adaptive classifiers significantly outperformed the static one, achieving an improvement of between 19.2% and 22.5% (Table 2). When the latest input and label were incorporated into an augmented input vector, i.e. \( d_{t,1} = [h^T_t, h^T_{t-1}, y_{t-1}]^T \) (the look-back parameter \( p = 1 \) was chosen because it achieved the lowest classification error), an improvement of classification accuracy was not significant for DLR-E and DBM. However, for DLR-U it achieved an improvement of 3.9%. As found in the experiment with the synthetic data in the previous section, when label information is successively randomly removed from 0% to 100% within the data stream, the classification performances of the models were not downgraded in proportion to the number of unobserved labels (Figure 6).
Figure 4: Mountain fire scenario: a fire starts in a wooded mountain with five villages (denoted by red dots), which are in potential danger according to local weather conditions. The satellite image of the mountain is extracted from Google Map.

![Satellite image of mountain with villages](image)

Figure 5: (Left panel) A 3-D plot of the three input variables and the class variable. (Right panel) A time plot of the class variable.

![3D plot and time plot](image)

Table 2: Mountain fire scenario: proportion of correct classification. $h_t$ is the current input vector, and $d_{t,1} = [h_t^T, h_{t-1}^T, y_{t-1}]^T$ is an augmented input vector.

<table>
<thead>
<tr>
<th>Input vector</th>
<th>Classifier</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_t$</td>
<td>Logistic Regression</td>
<td>0.7390</td>
</tr>
<tr>
<td></td>
<td>DLR-E</td>
<td>0.9636</td>
</tr>
<tr>
<td></td>
<td>DLR-U</td>
<td>0.9313</td>
</tr>
<tr>
<td></td>
<td>DBM</td>
<td>0.9623</td>
</tr>
<tr>
<td>$d_{t,1}$</td>
<td>DLR-E</td>
<td>0.9649</td>
</tr>
<tr>
<td></td>
<td>DLR-U</td>
<td>0.9703</td>
</tr>
<tr>
<td></td>
<td>DBM</td>
<td><strong>0.9784</strong></td>
</tr>
</tbody>
</table>
6.2.1 Active Label Requesting

We consider in this section the process of active label requesting. In practice observing labels may be expensive. Therefore, we can think of actively requesting a label; in other words, an adaptive classifier decides for itself when to request a label without severely deteriorating its classification performance. This problem is closely related to the problem of active data selection [Osborne, 2008], as both methods use uncertainty as a guide to request information. At each time step a classifier makes a one-step ahead label prediction and if uncertainty associated with the prediction is higher than a threshold, then it requests a label, otherwise, it proceeds without labelling. For dynamic logistic classifiers, the prediction uncertainty is \( \hat{\pi}_{t|t-1} (1 - \hat{\pi}_{t|t-1}) \) because a predicted label is assumed to have a Bernoulli distribution, and for the dynamic binomial model it is the variance of a beta-binomial predictive distribution, \( \text{Var}(y_t|D_{t-1}) \), given in Equation (98).

We carried out a set of experiments in order to obtain classification performances for different proportions of labelling from 100% to 30% (Table 3). With 50% labelling classification accuracy was degraded by 1.2% \( \sim 4.5% \) (Figure 7). In the figure the grey regions represent time steps when a classifier requested a label. We can see that when the models were confident enough about their predictions they did not request a label. Note that despite low classification error, the dynamic binomial model failed to detect the change of label in two occasions, and the dynamic logistic regression using the EKF in one occasion.

6.2.2 Missing Inputs

It is not only a label but also an input that can be missing, for example, as a result of failure of weather sensors or failure of transmitting, weather condition data can be occasionally unavailable. Our decision models are
### Table 3: Mountain fire scenario: proportion of correct classification when a classifier actively requests labels.

<table>
<thead>
<tr>
<th>Labelling</th>
<th>DLR-E</th>
<th>DLR-U</th>
<th>DBM</th>
</tr>
</thead>
<tbody>
<tr>
<td>100% Labelling</td>
<td>0.9636</td>
<td>0.9313</td>
<td>0.9623</td>
</tr>
<tr>
<td>90% Labelling</td>
<td>0.9528</td>
<td>0.9299</td>
<td>0.9407</td>
</tr>
<tr>
<td>70% Labelling</td>
<td>0.9528</td>
<td>0.9259</td>
<td>0.9407</td>
</tr>
<tr>
<td><strong>50% Labelling</strong></td>
<td><strong>0.9394</strong></td>
<td><strong>0.9191</strong></td>
<td><strong>0.9178</strong></td>
</tr>
<tr>
<td>30% Labelling</td>
<td>0.9272</td>
<td>0.8935</td>
<td>0.7278</td>
</tr>
</tbody>
</table>

Table 3: Mountain fire scenario: proportion of correct classification when a classifier actively requests labels.

![Dynamic Logistic Regression using EKF](image1)

![Dynamic Logistic Regression using UKF](image2)

![Dynamic Binomial Model](image3)

Figure 7: Mountain fire scenario: classification predictions when 50% of total labels is requested; the one-step ahead predictions for the class probability (blue), the label predictions (green), and the observed labels (red). The grey regions represent moments when the classifiers request a label.
defined by

\[ \pi_t = l(\varphi(h_t)^T w_t), \]  

(112)

for the DLR models, or

\[ y_t \sim \text{Binomial with } \eta_t = \varphi(h_t)^T w_t, \]  

(113)

for the DBM. Here \( l(\cdot) \) is the logistic function given in Equation (3) and \( \eta_t \) is the link equation given in Equation (53). When the input, \( h_t \), is not available, we can predict it as \( \hat{h}_t \). With a set of observed inputs up to time \( t - 1 \), i.e. \( \{h_1, \ldots, h_{t-1}\} \), we can predict \( h_t \) using, for example, a Gaussian Process model \([Rasmussen and Williams, 2006]\) or a nonstationary autoregressive model \([Lee and Roberts, 2008]\).

The data streams of the three inputs in the mountain fire scenario (i.e. wind speed, wind direction, air temperature) are illustrated in the left panel of Figure 8. We removed at random 70% of the input data (the middle panel of Figure 8) and predicted the missing input values with a nonstationary multivariate autoregressive model of order 2 (the right panel of Figure 8). We can see that the predicted inputs are very similar to the true ones except for a drastic drop of the air temperature predictions (coloured red) near at \( t = 100 \). To evaluate how adaptive classifiers perform when missing inputs, we randomly removed input information successively from 0% to 100% within the data. The performances of the classifiers were evaluated over 100 runs (Figure 9). The results are surprisingly good, in particular for DLR-E; with only 20% of inputs available it classified as correctly as when given complete input information. Note that for all the classifiers the variation in performance is very robust to large numbers of unobserved inputs.

Combined with the active label requesting, it is possible for a classifier to decide for itself which label to request even when input information is not available. We carried out a set of experiments in which 50% of inputs are missing and a classifier requests 50% of labelling. We computed the average proportion of correct classification for the three adaptive classifiers over 10 runs: 0.9279 (DLR-E), 0.9151 (DLR-U) and 0.8694 (DBM). In comparison to corresponding results when given complete input information provided in Table 3, we can see that for the DLR models, classification accuracy was degraded by only 0.4% or 1.1%.

6.3 Experimental Data

We applied the adaptive classifiers detailed here to an online brain-computer interface experiment whose goal was to classify electroencephalogram (EEG) activity into a movement / non-movement label. Data used in this experiment consisted of two channels of EEG, recorded at 256Hz placed over the central portion of the head and one channel of muscle electrical activity (EMG), recorded at 1024Hz over the muscles of the right fore-arm. The EMG was then down-sampled to 256Hz and muscle contraction strength for movement and
Figure 8: A time plot of (a) true inputs, (b) given inputs (70% missing) and (c) predicted inputs.

Figure 9: Mountain fire scenario: missing input experiment with variation in fractions of observed inputs from 0% to 100%.
non-movement detection was evaluated via a simple windowed peak and trough detection; this then formed a movement / non-movement label. The second reflection coefficient of a second-order autoregressive model [Pardey et al., 1996] were calculated over each EEG signal once every 78ms using a sliding one-second-long window, forming a set of feature vectors $h_t$. This data description is extracted from Lowne et al. [2008].

We ran a static logistic classifier to compare with the adaptive ones. It significantly underperformed the dynamic classifiers; DLR-E produced the best performance (Table 4). The one-step ahead predictions of the class probability and the label predictions are illustrated in Figure 10. They are coloured blue and green respectively. The observed labels (coloured red) are also drawn for reference. As discovered in the previous subsection, when previously observed inputs and labels were incorporated into the current input variable (i.e. the look-back parameter $p = 1$), all the classifiers produced almost perfect performances. In addition, when we successively removed label information, the performances of the dynamic classifiers were not degraded in proportion to the number of unobserved labels (Figure 11). In the figure we can see that the variation in performances is robust to large numbers of unobserved labels; with only 50% class labels, DLR-E achieved a 96.6% accuracy, DLR-U 92.4%, and DBM 91.9%.

<table>
<thead>
<tr>
<th>Input vector</th>
<th>Classifier</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_t$</td>
<td>Logistic Regression</td>
<td>0.7550</td>
</tr>
<tr>
<td></td>
<td>DLR-E</td>
<td><strong>0.9742</strong></td>
</tr>
<tr>
<td></td>
<td>DLR-U</td>
<td>0.9532</td>
</tr>
<tr>
<td></td>
<td>DBM</td>
<td>0.9348</td>
</tr>
<tr>
<td>$d_{t,1}$</td>
<td>DLR-E</td>
<td><strong>0.9922</strong></td>
</tr>
<tr>
<td></td>
<td>DLR-U</td>
<td>0.9916</td>
</tr>
<tr>
<td></td>
<td>DBM</td>
<td>0.9920</td>
</tr>
</tbody>
</table>

Table 4: EEG data: proportion of correct classification. $h_t$ is the current input vector, and $d_{t,1} = [h^T_t, h^T_{t-1}, y_{t-1}]^T$ is an augmented input vector.

We consider how an classification behaves when requesting a label actively. We varied the value of the threshold in order to compute classification error for different proportions of labelling from 100% to 30% (Table 5). With 50% labelling our classifiers’ performances were degraded by only $0.7\% \sim 1.6\%$. A more detailed description of this case is illustrated in Figure 12. Note that although the models decided for themselves when to request a label, they correctly detected irregular changes of labels and achieved high classification accuracies.

In addition, we carried out an experiment for missing input information. As presented in the previous section, input information is successively randomly removed from 0% to 100% (Figure 13). We can see that the
Figure 10: EEG data: classification predictions for the three adaptive models; the one-step ahead predictions of the class probability (blue), the label predictions (green), and the observed labels (red).

Figure 11: EEG data: missing label experiment with variation in fractions of observed labels from 0% to 100%.
performs of our adaptive classifiers reached close to optimal even with a small fraction of given inputs; for example, DLR-E achieved more than a 90% accuracy with 90% absence of input information. We experimented a case in which a classifier actively requests 50% of labelling with 50% of inputs available. To evaluate how adaptive classifiers perform in this environment, we computed the average proportion of correct classification over 10 runs: 0.9554 (DLR-E), 0.9381 (DLR-U) and 0.9107 (DBM). Comparing these values with results given on the fourth row of Table 5, we can find that classification accuracy was degraded by only 0.3% $\sim$ 1.7%.

Table 5: EEG data: proportion of correct classification when a classifier actively requests labels.

<table>
<thead>
<tr>
<th>Labelling</th>
<th>DLR-E</th>
<th>DLR-U</th>
<th>DBM</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td>0.9742</td>
<td>0.9532</td>
<td>0.9348</td>
</tr>
<tr>
<td>90%</td>
<td>0.9685</td>
<td>0.9526</td>
<td>0.9306</td>
</tr>
<tr>
<td>70%</td>
<td>0.9639</td>
<td>0.9486</td>
<td>0.9276</td>
</tr>
<tr>
<td><strong>50%</strong></td>
<td><strong>0.9585</strong></td>
<td><strong>0.9455</strong></td>
<td><strong>0.9279</strong></td>
</tr>
<tr>
<td>30%</td>
<td>0.9518</td>
<td>0.9367</td>
<td>0.8438</td>
</tr>
</tbody>
</table>

Figure 12: EEG data: classification predictions when 50% of total labels is requested: the one-step ahead predictions for the class probability (blue), the label predictions (green), and the observed labels (red). The grey regions represent moments when the classifiers request a label.
7 Conclusions and Future Work

7.1 Conclusions

We have demonstrated how a classification decision can be made sequentially in highly nonlinear, nonstationary and incomplete environments. We have studied two different approaches to the problem: (1) dynamic logistic regression using nonlinear Kalman filters and (2) a dynamic generalised linear model. The dynamic generalised linear model provides a mathematically elegant Bayesian framework for dynamic models so long as the distribution of an observation variable is a member of the exponential family. Since in the binary classification an observation variable has a Bernoulli distribution, the dynamic binomial model has been used as an adaptive classifier. In addition, we have suggested augmenting the current input variable with previously observed inputs and labels in order to capture any hidden patterns within the inputs and/or the labels.

In order to evaluate performances of the various models we have conducted a set of experiments. Firstly, the complete information scenario is considered, in which all necessary information including input values and class labels is available. We find that the adaptive classifiers outperformed static classifier; validating the usefulness of the adaptive approach in nonstationary classification problems. We find also that when a classifier uses an augmented input variable, its classification performance significantly improves; in particular if there is a dynamic pattern within inputs and/or labels. The second scenario we have examined is when information is incomplete; target labels are missing. When missing labels occurred randomly, we have found that classification accuracy is not linearly related to the number of unobserved labels. This is closely related to the problem of active label requesting, in which an adaptive classifier decides for itself which label to observe. This approach allows us to save costs associated with requesting a label. This is useful in application domains where feedback

Figure 13: EEG data: missing input experiment with variation in fractions of observed inputs from 0% to 100%. 
is expensive to acquire. We have also examined the case in which input information is missing. The adaptive classifiers still produced a robust performance when missing values were forecasted.

### 7.2 Outline of Future Work

Potential future work is twofold: (1) classifying multi-class labels (2) forecasting future decisions. Firstly, we can think of extending our adaptive classifiers to enable classification in a multi-class environment. Lowne et al. [2008] suggested an algorithm whereby, given a set of $K$ classes, we evaluate $(K - 1)$ two-class models which successively evaluate the probability that the true label is less than label index $k$. However, in the context of the dynamic generalised linear model this problem could be solved straightforwardly with a multinomial distribution. By allowing parameters of the multinomial distribution to vary over time and using a Dirichlet distribution as a conjugate prior, we may have an adaptive model for multi-class labels.

Secondly, in this research we have mainly focused on making binary classification decisions at one-step prediction horizons. A classification decision at a future time step, say $t + T$, could be made according to the dynamic logistic model,

$$\pi_{t+T} = l(\varphi_{t+T}^T \hat{w}_{t+T}),$$  \hspace{1cm} (114)

or alternatively the dynamic binomial model,

$$y_t \sim \text{Binomial with } \hat{\eta}_{t+T} = \varphi_{t+T}^T \hat{w}_{t+T},$$  \hspace{1cm} (115)

To solve the problem of making future classifications we need to predict unknown future inputs $h_{t+T}$ from the observed inputs to date, i.e. $\{h_1, \ldots, h_t\}$. In the missing input experiments of Section 6 this forecasting problem was solved by using Gaussian Process models [Rasmussen and Williams, 2006] or nonstationary autoregressive models [Lee and Roberts, 2008]. Incorporating these forecasting methods into the dynamic classification framework may allow us to forecast future decisions over longer time horizons.

Often we are not only interested in predicting a decision at a particular time in the future, but also making predictions sequentially over the entire time span between the present and the future time step. The decision made in Equations (114) or (115) is solely dependent upon an estimate at a single time step, $\hat{a}_t$. However, we would like to make a decision allowing for its future consequences. This can be formulated in the following way:

$$y_t = d\left(\sum_{i=0}^{T} \hat{a}_{t+i}\right),$$  \hspace{1cm} (116)

where $d(\cdot)$ is a decision function (such as the logistic function for binary classification), and $\hat{a}_{t+i} = \varphi_{t+i}^T \hat{w}_{t+i}$. Note that this has similarities with the sequential decision making method found in reinforcement learning.
In addition, we can consider a decision that takes into account uncertainties associated with the estimates \( \hat{a}_{t+i} \). In other words, a decision made with consideration for the trade-off between high return and low volatility may be more beneficial.

### A Derivations

#### A.1 Conjugate prior distribution for the binomial distribution

The conjugate prior for the binomial distribution is of form

\[
p(\theta_t | D_{t-1}) = \omega(k_t, m_t) \exp(k_t \theta_t - m_t \log(1 + \exp(\theta_t))). \tag{A.117}
\]

Using \( \theta_t = \log \left( \frac{\pi_t}{1 - \pi_t} \right) \), the probability density function of \( \pi_t \) is of form

\[
p(\pi_t | D_{t-1}) = p_{\theta_t | D_{t-1}}(\pi_t) \left| \frac{d}{d\pi_t} \log \left( \frac{\pi_t}{1 - \pi_t} \right) \right| = \omega(k_t, m_t) \pi_t^{k_t-1}(1 - \pi_t)^{m_t-k_t-1}. \tag{A.118}
\]

This is a beta distribution with parameters \( k_t \) and \( m_t - k_t \), thereby we know that

\[
\omega(k_t, m_t) = \frac{\Gamma(m_t)}{\Gamma(k_t)\Gamma(m_t - k_t)}.
\]

The mean and variance of the conjugate prior are computed in the following way. The moment generating function of \( \theta_t \) is

\[
M_{\theta_t}(z) = E(\exp(z\theta_t)) = \int_{-\infty}^{\infty} \exp(z\theta_t)p(\theta_t | D_{t-1})d\theta_t = \frac{\Gamma(k_t + z)\Gamma(m_t - k_t - z)}{\Gamma(k_t)\Gamma(m_t - k_t)}, \tag{A.119}
\]

and thus the cumulant generating function is

\[
C_{\theta_t}(z) = \log(M_{\theta_t}(z)) = \log(\Gamma(k_t + z) - \log(\Gamma(m_t - k_t - z)) - \Gamma(k_t) - \Gamma(m_t - k_t) \tag{A.120}
\]

From the cumulant generating function the two moments of the conjugate prior can be computed:

\[
E(\theta_t | D_{t-1}) = \left. \frac{d}{dz} C_{\theta_t}(z) \right|_{z=0} = \psi(k_t) - \psi(m_t - k_t) \approx \log \left( \frac{k_t}{m_t - k_t} \right), \tag{A.121}
\]

\[
\text{Var}(\theta_t | D_{t-1}) = \left. \frac{d^2}{dz^2} C_{\theta_t}(z) \right|_{z=0} = \psi'(k_t) + \psi'(m_t - k_t) \approx 1/k_t + 1/m_t - k_t. \tag{A.122}
\]

The function \( \psi(\cdot) \) denotes the digamma function defined by

\[
\psi(x) = \frac{d}{dx} \log(\Gamma(x)) = \frac{\Gamma'(x)}{\Gamma(x)}. \tag{A.123}
\]

The digamma function and its derivative are approximated by \( \log(x) \) and \( 1/x \) respectively [Abramovitz and Stegun, 1965].
A.2 One-step forecast distribution for the binomial distribution

When the observation variable $Y_t$ is a binomial, the one-step ahead forecast distribution is computed as follows:

$$p(y_t|D_{t-1}) = \int_{-\infty}^{\infty} p(y_t|\theta_t)p(\theta_t|D_{t-1})d\theta_t,$$

$$= \int_{-\infty}^{\infty} \exp \left( n \left[ \frac{y_t}{n} - \log(1 + \exp(\theta_t)) \right] + \log \left( \frac{n}{y_t} \right) \right) \frac{\Gamma(m_t)}{\Gamma(k_t)\Gamma(m_t - k_t)} \frac{\exp(k_t\theta_t)}{(1 + \exp(\theta_t))^{m_t}}d\theta_t,$$

$$= \frac{\Gamma(m_t)}{\Gamma(k_t)\Gamma(m_t - k_t)} \left( \frac{n}{y_t} \right) \int_{-\infty}^{\infty} \exp((k_t + y_t)\theta_t) \frac{\Gamma((m_t + n) - y_t)}{\Gamma(m_t + n)}d\theta_t.$$

$$= \frac{\Gamma(m_t)}{\Gamma(k_t)\Gamma(m_t - k_t)} \frac{\Gamma(k_t + y_t)\Gamma((m_t + n) - y_t)}{\Gamma(m_t + n)} \left( \frac{n}{y_t} \right). \quad (A.124)$$

References


