Probabilistic kernel least mean squares algorithms

Citation for published version:
Park, IM, Seth, S & Vaerenbergh, SV 2014, Probabilistic kernel least mean squares algorithms. in Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. Institute of Electrical and Electronics Engineers (IEEE), pp. 8272-8276. https://doi.org/10.1109/ICASSP.2014.6855214

Digital Object Identifier (DOI):
10.1109/ICASSP.2014.6855214

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.
PROBABILISTIC KERNEL LEAST MEAN SQUARES ALGORITHMS

Il Memming Park*, Sohan Seth†, Steven Van Vaerenbergh*

* Institute for Neuroscience, University of Texas at Austin, USA
† Helsinki Institute for Information Technology HIIT, Aalto University, Finland
* Department of Communications Engineering, University of Cantabria, Spain

ABSTRACT
The kernel least mean squares (KLMS) algorithm is a computation ally efficient nonlinear adaptive filtering method that “kernelizes” the celebrated (linear) least mean squares algorithm. We demonstrate that the least mean squares algorithm is closely related to the Kalman filtering, and thus, the KLMS can be interpreted as an approximate Bayesian filtering method. This allows us to systematically develop extensions of the KLMS by modifying the underlying state-space and observation models. The resulting extensions introduce many desirable properties such as “forgetting”, and the ability to learn from discrete data, while retaining the computational simplicity and time complexity of the original algorithm.

Index Terms— kernel adaptive filtering, KLMS, sequential Bayesian learning, state-space model

1. INTRODUCTION
Adaptive filtering algorithms deal with real-time learning scenarios, in which the environment is often nonstationary. In general, these algorithms need to fulfill three basic requirements: 1) to sequentially learn from each observation; 2) to be adaptive to changing environments; and 3) to be computationally efficient. Among many existing algorithms that fulfill these requirements, one that has stood the test of time is the celebrated least mean squares (LMS) algorithm. This algorithm has several interesting properties, in particular its inherent computational simplicity, and its implicit tracking ability despite its assumption of stationarity.

Inspired by the success of the LMS algorithm, a “kernelization” has been recently proposed under the name kernel least mean squares (KLMS) algorithm [1]. The KLMS inherits many desirable properties of LMS and extends it to a large class of nonlinear filtering algorithms. Nevertheless, it has certain limitations that arise from its formulation as an adaptive filter in a possibly infinite dimensional feature space. Specifically, if implemented naively, the representation of the filter grows linearly with the number of data samples processed. Moreover, both the LMS and the KLMS explicitly minimize the squared error between the desired and the estimated observation values, hence, they cannot be naturally applied to problems with discrete observations, e.g., class labels or neural spike counts.

In this paper, we show that LMS (and KLMS) can indeed be derived as an approximation of a state-space based Bayesian filtering (section 3). In order to achieve a low computational complexity, only the mode of the posterior distribution can be estimated and retained for each sample. This new interpretation allows us to derive extensions of the KLMS algorithm by tweaking the underlying state-space and observation models (section 4.1 and 4.2). Here, we extend the KLMS algorithm to integer-valued observations, and also introduce a forgetting factor in order to improve its tracking ability.

2. LMS DERIVATION REVISITED
Both Widrow & Hoff’s LMS, and KLMS are derived from mean squared error cost function [2, 1, 3] which is prevalent in traditional signal processing. The filtering setting assumes a linear model

\[ f(x; w) = w^T \phi(x) \]

in the feature space where \( \phi(x) = x \in \mathbb{R}^d \), while KLMS uses a (potentially) infinite dimensional (reproducing kernel) Hilbert space induced by a positive definite kernel \( k: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R} \) where \( k(x, y) = \phi(x)^T \phi(y) \) [4]. The mean squared error is,

\[ \mathcal{L}(w) = \frac{1}{2} E[(f(x; w) - y)^2], \]

where \( x \) and \( y \) are the random vector and variable for the input signal and the desired output, respectively. The basic steepest descent learning rule has the form

\[ \Delta w \leftarrow -\eta \frac{\partial \mathcal{L}(w)}{\partial w} = -\eta E \left[ (w^T \phi(x) - y) \phi(x) \right]. \quad (1) \]

1Extended version of this paper is available as http://arxiv.org/abs/1310.5347 [stat.ML]
To make an online learning rule, a stochastic gradient descent is used in practice. In particular, the learning rule of the LMS algorithm is obtained by dropping the expectation from (1), which yields

\[ w_{i+1} \leftarrow w_i - \eta_i (w_i^T \phi(x_i) - y_i) \phi(x_i). \]  

(2)

Hence, after processing \( i \) samples, the prediction for the next sample \( y_{i+1} \) is given by,

\[ \hat{y}_{i+1} = w_i^T \phi(x_{i+1}) = \sum_{k=1}^{i} \eta e_k \phi(x_k) \phi(x_{i+1}) \]

where \( e_k = y_k - w_k^T \phi(x_k) \) is the error for each sample. For KLMS, the prediction can be directly computed from the samples despite \( f \in \mathcal{H} \), since \( \phi(x_i) \phi(x_k) = k(x_i, x_k) \).

The stochastic gradient descent algorithm is guaranteed to convergence (almost surely) to the global optimal solution under stationary and ergodic observations if a proper step size scheduling is used (e.g., \( \sum_{i=1}^{\infty} \eta_i^2 < \infty \) and \( \sum_{i=1}^{\infty} \eta_i = \infty \)). However, the tracking capability of LMS/KLMS is dependent on the step size; if the step-size were annealed, it would be tracking less efficiently as more samples are seen. Therefore, to have constant tracking, step size is not annealed in practice, that is, \( \forall i \eta_i = \eta \). Then, for the price of non-zero misadjustment, the algorithms can surprisingly learn continuously from new samples, and overwrite what was learned before. Note that this “hack” disconnects the algorithm from the graphical model Fig. 1A which inherently assumes a stationary data generation process. In the following section, we show how this tracking ability can be derived from first principles.

### 3. BAYESIAN INTERPRETATION

A slowly changing system can be explicitly described by a probabilistic model with latent dynamics on the parameter. In such a model, each parameter associated with each sample or time is considered as an interdependent random (latent) variable, as illustrated in Fig. 1B. Our goal is to show that the KLMS is an approximate sequential inference for \( w_i \).

We start with a diffusion process as a reasonable model for non-stationarity:

\[ P(w_{k+1}|w_k) = \mathcal{N}(w_{k+1}; w_k, \sigma^2_d I), \]

(3)

where \( \mathcal{N} \) denotes a Gaussian distribution in the feature space, and \( \sigma^2_d \) is the variance of diffusion on each direction. The likelihood model is assumed to be a linear-Gaussian model, similar to the stationary case,

\[ P(y_k|x_k, w_k) = \mathcal{N}(y_k; w_k^T \phi(x_k), \sigma^2_n) \]

(4)

where \( \sigma^2_n \) is the observation noise variance. We remark that the conditional distributions (3) and (4) for the finite dimensional feature space is a special case of the Kalman filter model with linear dynamics.

The posterior weight distribution given data \( p(w_k|D_k) \) is fully described by its mean and covariance, since we assume Gaussianity in this model [5]. Let \( P(w_{k-1}|D_{k-1}) = \mathcal{N}(\mu_{k-1}, \Sigma_{k-1}) \), then a single linear Gaussian observation results in a one step evolution of the posterior as another Gaussian \( P(w_k|D_k) = \mathcal{N}(\mu_k, \Sigma_k) \), with

\[ \Sigma_k^{-1} = \Sigma_{k-1}^{-1} + \frac{1}{\sigma_n^2} \phi(x_k) \phi(x_k)^T \]

\[ \mu_k = \Sigma_{k-1} \left( \frac{1}{\sigma_n^2} y_k \phi(x_k) + \Sigma_{k-1}^{-1} \mu_{k-1} \right) \]

This recursion can be solved efficiently, and the solution is known as the extended recursive least squares algorithm [6]. However, it requires a quadratic number of operations in terms of the dimension of the feature vector for updating the (inverse) covariance matrix. In case of an infinite-dimensional feature space, the feature vector dimension grows linearly with the number of observations, rendering this approach prohibitive. Therefore, in order to obtain a linear time complexity algorithm, we assume the posterior to be concentrated around the maximum. In other words, we approximate the posterior as a delta function at the maximum a posteriori (MAP) estimate \( P(w_k|D_k) \cong \delta_{w_k} \) before inferring \( P(w_{k+1}|D_k) \). Below, we show the steps for online inference rules using this approximation.

First, note that the approximation is equivalent to assuming an isotropic Gaussian around the MAP estimate for the previous sample.

\[ P(w_{k+1}|D_k) = \int P(w_{k+1}|w_k) P(w_k|D_k) dw_k \]
Fig. 2. Tracking example of the Poisson extension of KLMS algorithm. The observations model a slowly drifting tuning curve of a simple cell in V1. The tuning curve is modeled as an exponentiated cosine \( \lambda(x) = \exp(4 \cos(x - \mu) - 0.1) \) where \( \mu \) constantly drifted 100 degrees during the 1000 iterations. We measure the normalized estimation error between the true tuning curve and the estimated curve. Insets show the actual function estimate at 25, 50, 100, 500 time steps. Gray lines show 11 repeats of the experiment, and the dark curves correspond to their average. The kernel was \( k(x, y) = \exp(-(x - y)^2/100) \) and \( \sigma^2 = 0.1. \)

\[
\begin{align*}
\zeta & \simeq \int P(w_{k+1}|w_k)\delta(w_{MAP}^{k})dw_k \\
& = P(w_{k+1}|w_k^{MAP}) = N(w_k^{MAP}, \sigma^2 I). \quad (5)
\end{align*}
\]

Using Bayes’ rule, the posterior weight distribution is,

\[
P(w_{k+1}|D_{k+1}) \propto P(y_{k+1}|x_{k+1}, w_k)P(w_{k+1}|D_k) = N(y_{k+1}; w_k^{\top}\phi(x_{k+1}), \sigma^2_n) \cdot N(w_{k+1}; w_k^{MAP}, \sigma^2 I) = N(w_{k+1}; w_k^{MAP}, \Sigma_{k+1})
\]

where the parameters for the posterior are,

\[
\Sigma_{k+1}^{-1} = \frac{1}{\sigma^2_d} I + \frac{1}{\sigma^2_n} \phi(x_{k+1})\phi(x_{k+1})^\top,
\]

\[
w_{k+1}^{MAP} = w_{k}^{MAP} + \frac{\eta(y_k - \lambda w_k^{MAP}\phi(x_k))}{1 + \eta^2\phi(x_k)^2} \phi(x_k). \quad (9)
\]

This learning rule (9) is very similar to that of NORMA [2]. Note that the learning rule (9) can be rewritten as,

\[
w_{k}^{MAP} = \sum_{i=1}^{k} \lambda^{k-1} \beta_i \phi(x_i), \quad (10)
\]

where \( \beta_i \) is a scalar corresponding to the coefficient at the learning step. We can see that each effective coefficient \( \lambda^{k-1} \beta_i \) for each \( \phi(x_i) \) shrinks geometrically over time. Thus, the effect of older observation to the current weight estimate is small in general. Note, however, that the algorithm forgets not by making the covariance larger as in the Kalman filter, but by changing the mean, as discussed in [7, 8].

4.2. Novel observations models for KLMS

Poisson likelihood is widely used when the observations are natural numbers: 0, 1, 2, \ldots. For example, in neuroscience, neural response is often quantified by the number of spikes, and tracking how the neural code changes during experiment is of great importance [9]. We use the canonical inverse link function (exponential) for the Poisson distribution to map the linear (or nonlinear) function from the input to the non-negative rate parameter, i.e.,

\[
P(y_k|x_k, w_k) = \text{Poisson}(y_k; \exp(w_k^\top \phi(x_k))). \quad (11)
\]
posterior as,

where irrelevant constants are omitted. We need to maximize

\( \log(5) \), for which the log prior is,

proximate the current state given the previous observations as

To derive the adaptive

system operates almost linearly.

properties of a 4-channel MIMO system. Real part of the lin-

varying channels. One-step ahead prediction error and aver-

channel, using data measured on a test bed with fast time-

Fig. 3

To test tracking, we acquired data from a wireless commu-

cation test bed that is used to evaluate the performance of
digital communication systems in realistic indoor environ-
ments. The platform is composed of several transmit and
receive nodes, each one including a radio-frequency front-
end and baseband hardware for signal generation and acqui-
sition. The front-end also incorporates a programmable vari-
able attenuator that causes signal saturation (see [10] for de-
tails). Using the hardware platform, we transmitted clipped
orthogonal frequency-division multiplexing (OFDM) signals
centered at 5.4 GHz over real frequency-selective and time-
varying channels, with normalized Doppler frequency around

\( 10^{-3} \). The transmit amplifier was operated close to saturation.

In this experiment the transmitted and received signals are
used to track the variations of the nonlinear channel. We com-
pare 4 algorithms with hyperparameters tuned using the first
500 samples \([2, 1, 11]\). Fig. 3 displays the one-step ahead pre-
diction normalized mean squared error (NMSE) of the track-
ing experiment. Quantized KLMS [11] and KLMS extended
with forgetful dynamics have almost identical performance.

5. NONLINEAR AND NONSTATIONARY CHANNEL

In this paper, we derived a family of linear time and space
complexity kernel adaptive filtering algorithms from Bayesian
filtering by maintaining only the mode of the posterior at each
iteration and discarding the covariance. One of the basic re-
sulting algorithms is the original KLMS. The tracking ability
of LMS/KLMS is usually understood by its stochastic nature
that allows it to continually adjust itself to the non-stationary
environment. We provide an alternate explanation of this
mechanism by showing the KLMS can also be seen as an
approximation to state-space modeling which possesses ex-

cplicit tracking abilities. Our framework allows flexibility in

the state-space models which can be used to induce forgetting
behavior, as well as novel observation noise models, such as
Poisson and Bernoulli.

6. CONCLUSION


7. REFERENCES


