School of Electronics and Computer Science
University of Southampton

COMP6247(2020/21): Reinforcement and Online Learning

Kalman and Particle Filters; Online PCA

Issue	
Deadline	

Introduction

The aim of this assignment (worth 30% of your assessment for this module) is to study online learning with supervised and unsupervised learning problems. You will implement an Extended Kalman Filter (EKF), and Monte Carlo methods, including a particle filter.

Generating Data for Logistic Regression Classifier

If we consider a two-class classification problem in which the class conditional likelihoods are Gaussian distributed with distinct means and a common covariance matrix, the functional form of the posterior probability of class membership, computed from Bayes formula as

$$P\left[\omega_i \,|\, \boldsymbol{x}\right] \;=\; \frac{P\left[\omega_i\right]\, p\left(\boldsymbol{x} \,|\, \omega_i\right)}{\sum_{j=1}^{K} p\left(\boldsymbol{x} \,|\, \omega_j\right)},$$

where $\omega_j,\ j = 1, 2, ..., K$ denote K classes and $\boldsymbol{x}$ is the feature vector. The posterior has a sigmoidal form:

$$P\left[\omega_i \,|\, \boldsymbol{x}\right] \;=\; \frac{1}{1 + \exp(-\boldsymbol{\theta}^T \boldsymbol{x})}.$$

We will generate data from a two class problem in two dimensions with means set at $\left[\begin{array}{cc} -\alpha & \alpha \end{array}\right]$ and $\left[\begin{array}{cc} \alpha & -\alpha \end{array}\right]$ with common covariance marix $\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$.
This is set symmetric so there is no need to include a bias term and α could be tuned to simulate different slopes of the sigmoidal posterior. If α is set high, the means are far apart, data from the problem is linearly separable and easy to learn. Hence this could give us the flexibility of observing convergence of an extended Kalman filter at different problem complexities.
Please see Lab Sheets of Foundations of Machine Learning for simulating such data.

Extended Kalman Filter

We know that the Kalman filter is optimal in estimating $\boldsymbol{x}(n)$ when the models of dynamics and observations are linear and the noise processes are both Gaussian. When faced with nonlinear systems, the natural approach is to locally linearize by truncating the Taylor series expansion of the nonlinearities about the operating point. This leads to the extended Kalman filter (EKF) algorithm. In the machine learning literature, such approaches to online learning have been used effectively, for example, in training radial basis function and multi-layer perceptrons sequentially [1, 2].
The simple dynamical system we will consider to illustrate EKF algorithm for online estimation of parameters will again have a random walk state dynamics and Gaussian noise processes, but

we will make the observation model nonlinear.

$$\begin{aligned}\boldsymbol{\theta}(n) &= \boldsymbol{\theta}(n-1) + \boldsymbol{w}(n) \\ y(n) &= f\left(\boldsymbol{\theta}, \boldsymbol{x}_n\right) + v(n),\end{aligned}$$

where $f(.,.)$ is a nonlinear function. With random walk dynamics imposed on $\boldsymbol{\theta}$, the unknown parameters of a model we wish to estimate, $\boldsymbol{x}(n)$ is now the input to the model.

Sequential Estimation of a Logistic Regression Model

We will use extended Kalman filter (EKF) to estimate parameters $\boldsymbol{\theta}$ of a logistic regression model defined as

$$y = \frac{1}{1 + \exp\left(-\boldsymbol{\theta}^T \boldsymbol{x}\right)}$$

in a sequential setting by propagating $\boldsymbol{\theta}$ and an error covariance matrix on its estimate sequentially (online).

Particle Filters

Sequential Monte Carlo methods are a family of online Bayesian inference tools in which a non-parametric density represented by samples drawn from it is propagated over time along with importance weights. When applied to filtering problems, Sequential Monte Carlo methods are often referred to as 'particle filters'. Within the context of particle filters, the importance samples are typically referred to as 'particles', and the importance weights are known as the 'particle weights'. In addition to several tracking and robot localization problems, e.g., [3], these techniques have also been used in optimizing neural networks [4].
We will consider Algorithms 1 and 2 given in [5] for this exercise.

Tasks

10 Marks Apply Sequential Importance Sampling (SIS) (Algorithm 1 in [5]) for the synthetic time varying AR problem in the Kalman Filter Assignment. Show that sequential importance sampling suffers from weight degeneracy. Implement a particle filter on the synthetic time varying AR problem by extending SIS to include a resampling step (Algorithm 2 in [5]). Investigate the impact of resampling on the importance weights. Compare your results to Kalman filter solution and discuss your findings.

10 Marks Write out the extended Kalman filter (EKF) algorithm for the logistic regression problem described above and present the algorithm as pseudocode. You need to differentiate the logistic function to do this step.

Generate synthetic data from a two-dimensional two-class distinct mean, common covariance problem as outlined in the section above and implement an EKF for the estimation of its parameters.

Show in a graph how the class boundary converges to the true solution, starting from a random guess.

10 Marks Design a particle filter to solve the sequential logistic regression problem. Provide the corresponding pseudocode and justify your assumptions. Implement your proposed approach and compare your results to the EKF solution. Discuss your findings.

Report

Upload a report of **no more than six pages** describing your work. Please make sure you include your name and/or student number in the report.

References

[1] Kadirkamanathan, V. and Niranjan, M. A Function Estimation Approach to Sequential Learning with Neural Networks,*Neural Computation* **5**(6): 954-975, 1993.

[2] Puskorius, G.V. and Feldkamp L.A., Decoupled extended Kalman filter training of feed-forward layered networks, *International Joint Conference on Neural Networks (IJCNN91):* 771-777, 1991.

[3] Evers, C. and Naylor, P. N., Acoustic SLAM, *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **26**(9): 1484-1498, 2018.

[4] Freitas, JFG de and Niranjan, M. and Gee, A.H. and Doucet, A., Sequential Monte Carlo methods to train neural network models, *Neural computation* **12**(4): 955-993, 2000.

[5] Arulampalam, MS, Maskell, S, Gordon, N &Tim Clapp, T. A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking, *IEEE Trans. Signal Processing* **50**(2), 174-188, 2002.