Variational Inference for Robust Sequential Learning of Multilayered Perceptron Neural Network

Najdan Vuković
Research Associate
University of Belgrade
Faculty of Mechanical Engineering
Innovation Center

Marko Mitić
Research Associate
University of Belgrade
Faculty of Mechanical Engineering
Production Engineering Department

Zoran Miljković
Full Professor
University of Belgrade
Faculty of Mechanical Engineering
Production Engineering Department

We derive a new sequential learning algorithm for Multilayered Perceptron (MLP) neural network robust to outliers. Presence of outliers in data results in failure of the model especially if data processing is performed on-line or in real time. Extended Kalman filter robust to outliers (EKF-OR) is probabilistic generative model in which measurement noise covariance is modeled as stochastic process over the set of symmetric positive-definite matrices in which prior is given as inverse Wishart distribution. Derivation of expressions comes straight form first principles, within Bayesian framework. Analytical intractability of Bayes' update step is solved using Variational Inference (VI). Experimental results obtained using real world stochastic data show that MLP network trained with proposed algorithm achieves low error and average improvement rate of 7% when compared directly to conventional EKF learning algorithm.

Keywords: *heavy-tailed noise, inverse Wishart distribution, extended Kalman filter, Bayesian learning, structured variational approximation.*

1. INTRODUCTION

Outliers have enormous importance when it comes to modelling engineering problems in which we have mathematical models of the physical system operating on-line. Outliers are defined as observations that significantly differ from the rest of the data [1], [2]. In engineering applications on-line processing of data is essential [2, 3, 4, 5, 6, 7, 8, 9, 10, 11] and failing to recognize, identify and process outliers may seriously jeopardize system's performance and eventually cause failure. Outliers may occur by chance, but more often, they may originate from temporary sensor failures, some unknown system anomalies or unmodeled reactions from the environment or some other disturbances [2].

We develop original sequential learning algorithm for Multilayered Perceptron neural network (MLP). To have system with this ability is of great importance for engineering because this approach bypasses off-line identification and removal of outliers. Furthermore, for sequential systems it is of extreme importance to process outliers as data arrives. Our algorithm is based on a conventional extended Kalman filter (EKF) but with the ability to process outliers during learning process as any other data point.

The structure of the paper is as follows. In the second part of the paper we provide brief survey of research efforts. In the third part we provide detailed and thorough derivation of EKF-OR. Experimental results are given in the fourth part, while concluding remarks in the last section of the paper.

Received: December 2014, Accepted: January 2015
Correspondence to: Dr Najdan Vuković
Innovation Center, Faculty of Mechanical Engineering,
Kraljice Marije 16, 11120 Belgrade 35, Serbia
E-mail: nvukovic@mas.bg.ac.rs
doi:10.5937/fmet1502123V

2. LITERATURE REVIEW AND CONTRIBUTIONS OF THE PAPER

Robust statistics is a broad research field and in this research we focus on robust sequential algorithms for neural network training. For wider perspective, additional information and concepts in terms of general robust statistics the reader is referred to [12, 13, 14 ,15, 16] and references therein.

The dominant approach in robust neural network training is to use robust cost function called M-estimator [16]; research community has proposed a number of M-estimators suited for this job: Hample [17], Welsch [18], and Tukey's biweight [19] (among others). All these robust cost function enable down-weighting of outliers.

Another approach is to identify and separate outliers before learning; then, one trains the model with data free of outliers [20, 21, 22, 23].

Finally, the third approach is to use hybrid algorithms and hope for the best. One may find hybrids of fuzzy and Radial Basis Function networks trained with Particle Swarm Optimization (PSO) [24], Support Vector Machines (SVM), RBF and fuzzy inference [25], Support Vector Regression and RBF networks [21].

The main features that set apart our paper from other research efforts are:

1. EKF-OR processes outliers as any other data point and naturally down-weights them within Bayesian framework.
2. Robustness to outliers is achieved using "uncertainty about uncertainty" approach [1]. The sequence of measurement noise covariance is modelled as stochastic process over the set of symmetric positive-definite matrices in which prior is given as inverse Wishart distribution;
3. Analytical intractability of update step is solved by applying structured variational approximation [26, 27, 28, 29] which puts tight lower bounds on the marginal likelihood of the data.

For additional information and deeper analysis of other research efforts, the reader is kindly referred to analysis given in [2] and references therein.

3. EXTENDED KALMAN FILTER ROBUST TO OUTLIERS

Let us define sequential learning problem of MLP network in the following form [2, 3, 4]:

$$p(\mathbf{w}_t \mid \mathbf{w}_{t-1}) \sim N(\mathbf{w}_{t-1}, \mathbf{Q}) \quad (1)$$

$$p(\mathbf{y}_t \mid \mathbf{w}_t) \sim N(\mathbf{H}_t \mathbf{w}_t, \mathbf{R}_t) \quad (2)$$

where $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$; $\mathbf{w}$ is vector of all network parameters (weights and biases), $\mathbf{Q}$ is a process covariance and $\mathbf{R_t}$ is observation noise covariance matrix. Finally, $\mathbf{y}_t$ is measurment while $\mathbf{H}_t$ is measurement Jacobian. Both distributions are conditionally Gaussian, while it is important to stress that measurement covariance is no longer fixed, i.e. $\mathbf{R_t}$ evolves over time and it is being estimated at each filter iteration. This is the first distinction that sets apart our algorithm from its predecessor Kalman filter.

The sequence $\{\mathbf{R}_t\}$ is modeled as stochastic process over the set of symmetric positive-definite matrices [1]. Let us define a prior distribution over $\mathbf{R_t}$ at each time step. In Bayesian statistical modeling, a conjugate distribution is distribution that generates the same functional form of posterior as prior [26, 27, 28]. In this research we assume prior distribution over $\mathbf{R_t}$ as inverse Wishart, i.e.

$$\mathbf{R} \sim W^{-1}(\nu\boldsymbol{\Omega}, \nu) \quad (3)$$

where $\boldsymbol{\Omega}$ and ν denote harmonic mean and degrees of freedom, respectively. We define inverse Wishart distribution as a probability distribution over convex cone of $d \times d$ symmetric positive-definite matrices, parameterized with harmonic mean $\boldsymbol{\Omega}$ and degrees of freedom ν. Now, suppose that at each time step the noise covariance matrix $\mathbf{R_t}$ is inverse Wishart, i.e.

$$\mathbf{R}_t \sim W^{-1}(\nu_t \boldsymbol{\Lambda}_t, \nu_t) \quad (4)$$

$\boldsymbol{\Lambda}_t$ and ν_t denote harmonic mean and degrees of freedom. If we multiply measurement (2) with (4), and marginalize out $\mathbf{R_t}$, we will come to the conclusion that observations $\mathbf{y}_t$ are Student *t*-distributed [2]:

$$p(\mathbf{y}_t \mid \mathbf{w}_t) = \int_{\mathbf{R}_t > \mathbf{0}} p(\mathbf{y}_t \mid \mathbf{w}_t, \mathbf{R}_t) p(\mathbf{R}_t) d\mathbf{R}_t$$

$$\propto \left(1 + \frac{(\mathbf{y}_t - \mathbf{H}_t \mathbf{w}_t)^T \boldsymbol{\Lambda}^{-1} (\mathbf{y}_t - \mathbf{H}_t \mathbf{w}_t)}{\nu_t}\right)^{-\frac{\nu_t + 1}{2}} \quad (5)$$

where $\boldsymbol{\Lambda}$ is scale matrix. Student *t* distribution has some very attractive properties when it comes to modeling of possible presence of outliers in data. Firstly, Student t distribution has heavier and longer tails then Gaussian, it decays at less then exponential rate, which actually tells our learner that probability mass is spread over wider region. Secondly, its influence function is less sensitive to infinitesimal changes in data. For additional information the reader is referred to [2].

3.1 Bayesian learning and Variational Inference

Let us suppose that data is given with the sequence $\mathbf{Y}_T = \{\mathbf{y}_t\}_{t=1}^T$, and let $\alpha(\mathbf{w}_t)$ be posterior of network parameters given all data, i.e.

$$\alpha(\mathbf{w}_t) = p(\mathbf{w}_t \mid \mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_T) = p(\mathbf{w}_t \mid \mathbf{Y}_T) \quad (6)$$

The state vector is not Gaussian due to the Student *t* distributed measurement vector. Bayesian learning is performed using the Bayes' rule [2]:

$$\underbrace{p(\mathbf{w}_t \mid \mathbf{y}_t)}_{\text{Posterior}} = \frac{\overbrace{p(\mathbf{y}_t \mid \mathbf{w}_t, \mathbf{R}_t)}^{\text{Likelihood}} \overbrace{p(\mathbf{w}_t)}^{\text{Prior}}}{\underbrace{p(\mathbf{y}_t)}_{\text{Evidence}}}$$

$$= \frac{p(\mathbf{y}_t \mid \mathbf{w}_t, \mathbf{R}_t) \overbrace{\int p(\mathbf{w}_t \mid \mathbf{w}_{t-1}) p(\mathbf{w}_{t-1}) d\mathbf{w}_{t-1}}^{p(\mathbf{w}_t \mid \mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{t-1})}}{p(\mathbf{y}_t)} \quad (7)$$

The main problem in derivation of EKF-OR (or in any other non Gaussian case), arises when likelihood $p(\mathbf{y}_t \mid \mathbf{w}_t, \mathbf{R}_t)$ is to be multiplied with posterior $p(\mathbf{w}_t \mid \mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{t-1})$; the non Gaussian noise model makes calculation of this analytically intractable. To overcome this problem, we apply Variational Inference (VI) approach in form of structured variational approximation [2, 26, 27, 28]. Variational methods have been in use in mathematics, physics and engineering since 17th century.

Let us define new sequence of random variables as $\mathbf{z}_t = \{\mathbf{w}_t, \mathbf{R}_t\}$, and let $q(\cdot)$ be an approximate posterior distribution over $\mathbf{z}_t$ given $\mathbf{y}_t$. Marginal log-likelihood of the data is given as [2]:

$$\ln p(\{\mathbf{y}_t\}) = L[q] + KL[q \,\|\, p] \quad (8)$$

where $L[q]$ denotes lower bound on the data marginal log-likelihood:

$$L[q] = \iint \ldots \int q(\{\mathbf{z}_t\}) \ln \frac{p(\{\mathbf{y}_t, \mathbf{z}_t\})}{q(\{\mathbf{z}_t\})} \prod_{t=1}^{T} d\mathbf{y}_t \quad (9)$$

and $KL[q \,\|\, p]$ is Kullback-Leibler (KL) divergence, known as relative entropy:

$$KL[q \,\|\, p] = \iint \ldots \int q(\{\mathbf{z}_t\}) \ln \frac{p(\{\mathbf{z}_t\} \mid \{\mathbf{y}_t\})}{q(\{\mathbf{z}_t\})} \prod_{t=1}^{T} d\mathbf{y}_t \quad (10)$$

We emphasize that KL is not symmetric hence it is not distance (metric) measure. It is used to quantify similarity between two distributions.

A perfect fit implies $KL[q \,\|\, p] = 0$. However, this is hard to achieve (if not impossible), thus knowing that

$KL[q \| p] \geq 0$, one may conclude that $\ln p(\{\mathbf{y}_t\}) > L[q]$, which is why $L[q]$ is known as lower bound on data log-likelihood [2]. Now, minimization of $KL[q \| p]$, which has to be performed to achieve good approximation of joint posterior with $q(\cdot)$, inevitable leads to maximization of $L[q]$. However, the main advantage is that $L[q]$ operates on complete data log-likelihood and does not involve operations on the true posterior.

Structured variational inference enables us to search for the solution among family of distributions $q(\cdot)$. In this paper we search for the solution of the problem by looking among family of distributions that factor as (see [2]):

$$q(\{\mathbf{w}_t, \mathbf{R}_t\}) = q(\{\mathbf{w}_t\}) q(\{\mathbf{R}_t\}) \tag{11}$$

By doing this, we preserve inner statistical dependencies between state and noises but we omit dependencies between them. In physics, this approach is called the mean field theory assumption [26, 27, 28].

3.2 Derivation of the EKF-OR learning algorithm

Firstly, we need to specify the complete data likelihood, defined as the following product:

$$p(\{\mathbf{w}_t, \mathbf{R}_t, \mathbf{y}_t\}) = p(\{\mathbf{w}_1\}) \prod_{t=2}^{T} p(\mathbf{w}_t \mid \mathbf{w}_{t-1}) \prod_{t=1}^{T} p(\mathbf{z}_t \mid \mathbf{w}_t, \mathbf{R}_t) p(\{\mathbf{R}_t\}) \tag{12}$$

Now, approximate posteriors of noise and state vector that maximize (9) are given with the following expressions [2]:

$$\ln q(\{\mathbf{w}_t\}) = \mathrm{E}_{q(\{\mathbf{R}_t\})}\left[\ln p(\{\mathbf{w}_t, \mathbf{R}_t, \mathbf{y}_t\})\right] + \ldots \tag{13}$$

$$\ln q(\{\mathbf{R}_t\}) = \mathrm{E}_{q(\{\mathbf{w}_t\})}\left[\ln p(\{\mathbf{w}_t, \mathbf{R}_t, \mathbf{y}_t\})\right] + \ldots \tag{14}$$

where $\mathrm{E}_{q(\cdot)}\left[\ln p(\{\mathbf{w}_t, \mathbf{R}_t, \mathbf{y}_t\})\right]$ stands for expectation of compete data log-likelihood $\ln p(\{\mathbf{w}_t, \mathbf{R}_t, \mathbf{y}_t\})$ calculated with respect to the distribution $q(\cdot)$. Taking the logarithm of (12) and expectation over $q(\{\mathbf{R}_t\})$ we may formulate the expression for the state vector, which is given as:

$$\ln q(\{\mathbf{w}_t\}) = \ln p(\{\mathbf{w}_1\}) + \sum_{t=2}^{T} \ln p(\mathbf{w}_t \mid \mathbf{w}_{t-1}) + \sum_{t=1}^{T} \mathrm{E}_{q(\{\mathbf{R}_t\})}\left[\ln p(\mathbf{z}_t \mid \mathbf{w}_t, \mathbf{R}_t)\right] + \ldots \tag{15}$$

One may notice that the last term of (15) is quadratic with respect to the state vector $\mathbf{w}_t$. This update equation for $q(\{\mathbf{w}_t\})$ has the same functional form as standard EKF recursion [2, 3, 4, 30, 31], in which one starts with initial estimate of the state $\mathbf{w}_1$ and iteratively propagates it via state transition model (1) and updates it using newest observation $\mathbf{y}_t$ via measurement update equations (2). The difference is that in EKF-OR we use expected value of the measurement covariance, i.e. $\mathrm{E}\left[\mathbf{R}_t^{-1}\right] = \mathbf{\Omega}_t^{-1}$.

Having found the expression for distribution of the state vector, it remains to derive expression for measurement covariance. To do that, we need to specify a model for measurement noise. In this research we shall focus on the case of independent identically distributed (IID) noise, characterized by the following prior at each time stamp

$$\mathbf{R}_t \sim W^{-1}(s\mathbf{\Omega}, s) \tag{16}$$

Taking expectation of complete data log-likelihood with respect to distribution $q(\{\mathbf{w}_t\})$ results in expression for measurement noise covariance:

$$\ln q(\{\mathbf{R}_t\}) = \sum_{t=1}^{T} \mathrm{E}_{q(\{\mathbf{w}_t\})}\left[\ln p(\mathbf{y}_t \mid \mathbf{w}_t, \mathbf{R}_t)\right] + \sum_{t=1}^{T} \ln p(\mathbf{R}_t) \tag{17}$$

Where the last term in (17) implies that noise $q(\{\mathbf{R}_t\})$ further factors as (see [2]):

$$q(\{\mathbf{R}_t\}) = \prod_{t=1}^{T} q(\mathbf{R}_t) \tag{18}$$

This result comes as consequence of (11) and IID assumption. (15) shows that marginal of state vector given observations is Gaussian $\mathbf{w}_t \mid \{\mathbf{y}_t\} \sim N(\boldsymbol{\mu}_t, \mathbf{\Sigma}_t)$. It remains to define distribution of $\mathbf{R}_t \mid \{\mathbf{y}_t\}$. Knowing that (16) is the conjugate prior for (2), it is easy to show the following proportionality

$$\mathbf{R}_t \mid \{\mathbf{y}_t\} \sim W^{-1}(\nu_t \mathbf{\Omega}_t, \nu_t) \tag{19}$$

In other words, measurement covariance $\mathbf{R}_t$ is distributed according to inverse Wishart, given observations $\{\mathbf{y}_t\}$. Harmonic mean $\mathbf{\Omega}_t$ is given as:

$$\mathbf{\Omega}_t = \frac{s\mathbf{R} + \mathbf{S}_t}{s+1} \tag{20}$$

with $\nu_t = s+1$ degrees of freedom. Finally, harmonic mean $\mathbf{\Omega}_t$ is a convex combination of the nominal noise $\mathbf{R}$ and the expected sufficient statistics matrix $\mathbf{S}_t$, calculated as (see [2]):

$$\mathbf{S}_t = (\mathbf{y}_t - \mathbf{g}_t(\boldsymbol{\mu}_t, \mathbf{x}_t))(\mathbf{y}_t - \mathbf{g}_t(\boldsymbol{\mu}_t, \mathbf{x}_t))^T + \mathbf{H}\mathbf{\Sigma}_t\mathbf{H}^T \tag{21}$$

where $\boldsymbol{\mu}_t = \hat{\mathbf{w}}_t$ is expected value of the state vector $\mathbf{w}_t$. As given by (20), in the limit when $s \to \infty$, harmonic mean $\mathbf{\Omega}_t$ reduces to nominal noise covariance $\mathbf{R}$. We emphasize another interpretation of (21). When difference between predicted output of the network $\mathbf{g}_t(\hat{\mathbf{w}}_t, \mathbf{x}_t)$ and current measurement (data point) $\mathbf{y}_t$ is

small, approximate noise remains the same as the nominal **R**; on the other hand, if there is big difference between predicted output $\mathbf{g}_t(\hat{\mathbf{w}}_t, \mathbf{x}_t)$ and current measurement $\mathbf{y}_t$, matrix $\mathbf{S}_t$ will be larger than nominal measurement covariance **R**, and it will dominate in (20). As a consequence, harmonic mean $\mathbf{\Omega}_t$ will be larger than **R**, which results in down-weighting of the measurement $\mathbf{y}_t$ since it is being treated as an outlier.

As a main result, pre-processing or labelling of outliers is eliminated; processing of outliers is carried out sequentially within learning algorithm. EKF-OR learning algorithm is given in Algorithm 1.

Algorithm 1. Learning algorithm based on extended Kalman filter Robust to Outliers (EKF-OR) for Multilayered Perceptron Neural Network (IID noise case)

input $\left(\mathbf{g}(\cdot,\cdot), p_0, q_0, r_0\right)$
1. **For each observation** $(\mathbf{x}_t, \mathbf{y}_t)$, **t = 1,…,n do**
1.1 Predict state vector and covariance $\boldsymbol{\mu}_{t\|t-1} = \mathbf{w}_{t-1}$; $\mathbf{P}_{t\|t-1} = \mathbf{P}_{t-1} + \mathbf{Q}$
1.2 Set initial values for iterative process $\mathbf{m}_t \leftarrow \mu_{t\|t-1}$; $\mathbf{M}_t \leftarrow \mathbf{P}_{t\|t-1}$
2. **While**
2.1 Update noise covariance given state 2.2 $\mathbf{S}_t \leftarrow \left(\mathbf{y}_t - \mathbf{g}_t(\boldsymbol{\mu}_t, \mathbf{x}_t)\right)\left(\mathbf{y}_t - \mathbf{g}_t(\boldsymbol{\mu}_t, \mathbf{x}_t)\right)^T + \mathbf{H}\mathbf{P}_t\mathbf{H}^T$ 2.3 $\mathbf{\Omega}_t \leftarrow \frac{s\mathbf{R} + \mathbf{S}_t}{s+1}$
2.4 Update state given noise 2.5 $\mathbf{K}_t \leftarrow \mathbf{M}_t\mathbf{H}^T\left(\mathbf{H}\mathbf{M}_t\mathbf{H}^T + \mathbf{\Omega}_t\right)^{-1}$ 2.6 $\boldsymbol{\mu}_t \leftarrow \mathbf{m}_t + \mathbf{K}_t\left(\mathbf{y}_t - \mathbf{g}(\mathbf{m}_t, \mathbf{x}_t)\right)$ 2.7 $\mathbf{P}_t \leftarrow \left(\mathbf{I} - \mathbf{K}_t\mathbf{H}\right)\mathbf{M}_t\mathbf{H}$
EndWhile
3. **EndFor** **output**

In the very first step of EKF-OR the user defines type of activatoin function (hyperbolic tangent or sigmoid), architecture of MLP neural network (number of hidden layers and neurons in them) and parameters needed for filter deployment: initial state uncertainty $\mathbf{P}_0 = p_0\mathbf{I}_\mathbf{P}$, state transition uncertainty $\mathbf{Q} = q_0\mathbf{I}_\mathbf{Q}$, and measurement uncertainty $\mathbf{R} = r_0\mathbf{I}_\mathbf{R}$, where $\mathbf{I}_\mathbf{P}$, $\mathbf{I}_\mathbf{Q}$, and $\mathbf{I}_\mathbf{R}$ are the identity matrices of appropriate dimensions. The learning process starts with introduction of the first data point $(\mathbf{x}_1, \mathbf{y}_1)$. The first step in Kalman filtering theory is the prediction of the state and covariance; this is performed in 1.1. In 1.2 the algorithm assigns new parameters needed for EKF-OR iterative procedure. In the second step of the EKF-OR learning algorithm, we introduce new measurement $\mathbf{y}_t$ and generate estimate of the state vector $\mathbf{w}_t$. To solve (13) and (14) (i.e. (15) and (17)), we have to apply iterative procedure because no closed form solution exist. Convergence is controled by monitoring innovation likelihood declaring that change in two consecutive iterations has to be greater than previously defined tolerance. In the first step we estimate $q(\{\mathbf{R}_t\})$ given our best current estimate of the state vector; in 2.2 expected sufficient statistics matrix $\mathbf{S}_t$ is calculated. Then, in 2.3, we update noise with newly estimated $\mathbf{S}_t$. The second step 2.4 implies update of our current best estimate of the state vector parameters given new estimate of the noise covariance $\mathbf{\Omega}_t$. In 2.5 we calculate the Kalman gain. Lines 2.6 and 2.7 perform update of $q(\{\mathbf{w}_t\})$ parameters.

It is important to stress that measurement Jacobian **H**, which is defined as $\mathbf{H} = \left[\frac{\partial \mathbf{g}(\cdot)}{\partial \mathbf{w}}\right]$, is iteratively calculated (within the While loop of the Algorithm 1-step #2). We point out that other nonlinear Kalman filters may be used. For example, unscented Kalman Filter (UKF) or extended information filter (EIF) may be applied as learning algorithm of neural network. These algorithms are extensively tested for RBF networks training and they have proven their performance in [4, 32, 33]. Instead of Taylor series linearization these algorithms may be used in our robust sequential learning algorithm without necessity to change the basic idea of the algorithm.

4. EXPERIMENTAL RESULTS

To fully assess perfomance of MLP network trained with EKF-OR sequential learning algorithm we setup the following experiment. MLP network is to learn highly nonlinear stochastic functions such as values of different stock indexes on the financial market. We have chosen values of actions for three consecutive years (1st January 2006 -31st December 2008) for the following indexes of companies in IT sector *Microsoft*, *Apple*, *Google*, *Red Hat*, *Intel*, *Yahoo*, *IBM*, and *Oracle* and stock index of 500 largest companies *S&P500*. Data are downloaded from [34]. These functions are highly nonlinear and stochastic, which makes them hard to model. However, when outliers are added to these functions they become even harder for modeling. The identical MLP network is trained with EKF and EKF-OR and EKF-OR's performance is compared to that of EKF.

Artificial outliers are added to each training set. For all experiments the following procedure is adopted:

1. Add certain number of outliers generated according to predefined mechanism;
2. Train MLP neural network using EKF-OR with training set contaminated by outliers;
3. Test performance of optimized MLP neural network using test set free of outliers.

The simulated noise sequence is burst noise, following bistable regime:

$$b_t = \begin{cases} 0, & \text{nominal noise} \\ 1, & \text{burst of outliers} \end{cases} \tag{22}$$

b_t is sampled from Markov process with p=5% probability of transition. The probabilty that two

consecutive elements are equal is given with probability 1-p=95%. If b_t=0 the noise is samlped from inverse Wishart distribution with unit covariance and 10 degrees of freedom; otherwise b_t=1 it is sampled from Gaussian distribution $N\left(0,\beta^2\right), \beta^2 = 0.01$. During the burst, measurements are quickly becoming larger and more volatile than in nominal conditions. We emphasize that this noise sequence does not obey IID noise case which may jeoperdize performance of both EKF-OR and generic EKF. However, although IID noise case is deeply embedded into roots of EKF-OR, as we will see, EKF-OR can easily overcome this problem due to VI employment.

MLP network is to predict the next value in the series given three previous values/measurements, i.e. $\hat{\mathbf{y}}_{t+1} = \mathbf{g}\left(\mathbf{y}_t, \mathbf{y}_{t-1}, \mathbf{y}_{t-2}\right)$. The accuracy is measured with root mean square error (RMSE) defined as:

$$\text{RMSE} = \sqrt{\frac{1}{N_{test}} \sum_{n=1}^{N_{test}} \left(\mathbf{y}_t - \hat{\mathbf{y}}_t\right)^T \left(\mathbf{y}_t - \hat{\mathbf{y}}_t\right)} \qquad (23)$$

where N_{test} denotes total number of available testing samples. The lower numerical value of RMSE for testing set free of outliers implies better generalization. All codes are written and run in Matlab 7.12 programing environment; experiments are conducted on laptop computer with Intel(R) Core™ i5-4200U CPU @ 1.6GHz (2.3GHz) with 6GB of RAM, running on 64-bit Windows 7.0.

Five different MLP architectures are tested: (1) 3-10-1; (2) 3-20-1; (3) 3-5-5-1; (4) 3-10-5-1; (5) 3-10-10-1. Each experiment is repeated 30 times; each time the new initial values of weights and biases are generated and entire learning process is performed. The results averaged over 30 independent trials are given in Tables 1-5. Results in tables show average, maximum and minimal RMSE for test set free of outliers, as well as the improvement rate (IR) of EKF-OR when compared directly to EKF.

As experimental results given in Tables 1-5 show, EKF-OR outperforms EKF in terms of accuracy, where accuracy is measured by RMSE calculated for test set free of outliers. Furthermore, the average maximum RMSE of EKF-OR is lower than average maximum RMSE of EKF; similarly, the average minimum RMSE value of EKF-OR is higher than average minimum value of RMSE for EKF. The average improvement rate is 7%, where for some experiments the highest IR value reaches 21%. In Figure 1 one may see *S&P500* stock index. The upper part of the figure shows the nominal value of the stock (blue solid line) and values polluted by outliers (black dotted line). The sudden and wild bursts of heavy tailed noise (outliers) are easily noticed. The lower part of Figure 1 depicts test set for *S&P500* stock index. As mentioned, the test set is free of outliers. One may see that MLP trained with EKF-OR is able to reconstruct original signal/data regardless of outliers' presence. Similarly, in upper left corner of Figure 2 one may see nominal time series (solid blue) and time series polluted with outliers (dotted black) for *Apple* stock index. Lower left in Figure 2 depicts reconstructed time series plotted versus nominal free of outliers. The right part of Figure 2 shows box plot for EKF and EKF-OR for 30 independent experimental runs.

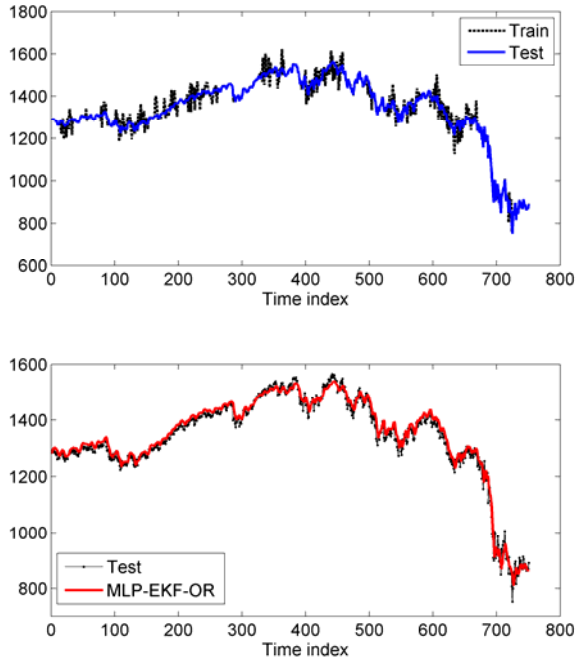

Figure 1. S&P500 stock index in a given time frame.

5. CONCLUSION

In real world applications of neural networks designers/engineers have to deal with presence of outliers in data. For on-line and especially real time implementations, it is essential to have learning model able to tackle possible outliers in stream of data.

To enable real world implementation of neural networks, in this paper we have derived a new sequential algorithm for robust learning of Multilayered Perceptron (MLP) neural network in presence of outliers. Extended Kalman Filter robust to outliers (EKF-OR) is based on simple intuition of "uncertainty about uncertainty" [1, 2]; in EKF-OR we allow measurement covariance matrix to evolve over time and model this process as stochastic process in which prior is modelled as inverse Wishart distribution. EKF-OR sequentially processes all data points, regardless if data point is outlier or not. In EKF-OR outliers are "naturally" down-weighted within learning setup. To solve the problem of analytical intractability of update step in Bayesian framework we applied Variational Inference (VI) in form of structured variational approximation. This enables the algorithm to operate on complete data log-likelihood and to iteratively improve estimates of state and noise.

Experimental results on real world data (real world time series polluted with burst of noise) demonstrate effectiveness and good generalization ability of derived learning algorithm, and together with developed theoretical concept provide strong foundations for future research.

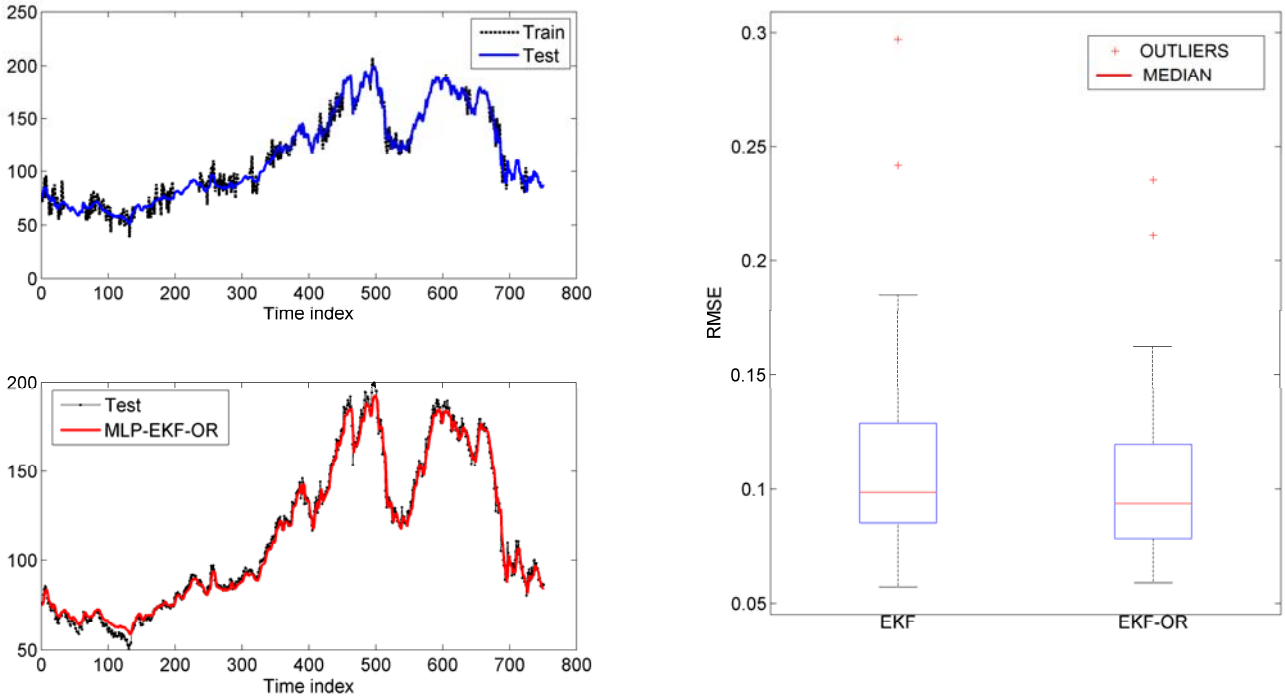

Figure 2. Apple stock index in a given period. Right part shows box plot of EKF and EKF-OR for 30 independent trials (3-20-1).

Table 1. Experimental results for 3-10-1 MLP network architecture. Results are averaged over 30 independent trials.

	Avg. RMSE test		Max RMSE test		Min RMSE test		IR
	EKF-OR	EKF	EKF-OR	EKF	EKF-OR	EKF	%
Microsoft	**0.1130±0.0464**	0.1226±0.0595	**0.2377**	0.2773	0.0639	0.0631	**7.825**
Apple	**0.1066±0.0411**	0.1170±0.0535	**0.2351**	0.2971	0.0590	0.0572	**8.8820**
Google	**0.0910±0.0205**	0.0952±0.0244	**0.1357**	0.1696	**0.0574**	0.0608	**4.4214**
Red Hat	**0.1102±0.0533**	0.1146±0.0519	**0.2872**	0.2903	**0.0602**	0.0606	**3.9084**
Intel	**0.1087±0.0356**	0.1189±0.0476	**0.2155**	0.2397	**0.0642**	0.0657	**8.5683**
Yahoo	**0.0898±0.0346**	0.0938±0.0297	**0.1991**	0.2033	**0.0538**	0.0541	**4.2414**
IBM	**0.0951±0.0261**	0.1047±0.0361	**0.1648**	0.2034	**0.0591**	0.0651	**9.2441**
Oracle	**0.1228±0.0590**	0.1334±0.0669	**0.3477**	0.3711	**0.0804**	0.0832	**7.9646**
S&P500	**0.0953±0.0381**	0.1006±0.0397	0.2244	**0.2046**	0.0500	**0.0475**	**5.2508**

Table 2. Experimental results for 3-20-1 MLP network architecture. Results are averaged over 30 independent trials.

	Avg. RMSE test		Max RMSE test		Min RMSE test		IR
	EKF-OR	EKF	EKF-OR	EKF	EKF-OR	EKF	%
Microsoft	**0.0926±0.0235**	0.0964±0.0243	0.1454	0.1380	**0.0624**	0.0627	**3.8733**
Apple	**0.0921±0.0327**	0.0987±0.0343	**0.1869**	0.2226	**0.0526**	0.0577	**6.7474**
Google	**0.1173±0.0806**	0.1238±0.0799	0.4609	0.4446	**0.0520**	0.0540	**5.1982**
Red Hat	**0.1083±0.0491**	0.1192±0.0654	**0.2694**	0.3299	**0.0605**	0.0611	**9.1485**
Intel	**0.0920±0.0222**	0.1007±0.0355	**0.1767**	0.2147	**0.0658**	0.0686	**8.7191**
Yahoo	**0.0982±0.0410**	0.1008±0.0443	0.2493	0.2211	**0.0558**	0.0561	**2.6347**
IBM	**0.1035±0.0346**	0.1150±0.0468	**0.1997**	0.2520	**0.0576**	0.0574	**10.0422**
Oracle	**0.1060±0.0231**	0.1096±0.0263	**0.1594**	0.1756	**0.0759**	0.0799	**3.2856**
S&P500	**0.0922±0.0355**	0.0950±0.0370	**0.1952**	0.1989	**0.0486**	0.0488	**2.8612**

Table 3. Experimental results for 3-5-5-1 MLP network architecture. Results are averaged over 30 independent trials.

	Avg. RMSE test		Max RMSE test		Min RMSE test		IR
	EKF-OR	EKF	EKF-OR	EKF	EKF-OR	EKF	%
Microsoft	**0.1042±0.0276**	0.1256±0.0472	**0.1747**	0.2742	**0.0643**	0.0742	**17.0475**
Apple	**0.1269±0.0599**	0.1325±0.0704	**0.2714**	0.3850	0.0614	0.0603	**4.2196**
Google	**0.1335±0.0644**	0.1553±0.0646	**0.3366**	0.3582	**0.0648**	0.0748	**14.0552**
Red Hat	**0.1308±0.0467**	0.1437±0.0727	**0.2487**	0.4330	**0.0800**	0.0827	**9.0313**
Intel	**0.1496±0.0549**	0.1796±0.0771	**0.2776**	0.3333	**0.0770**	0.0848	**18.2190**
Yahoo	0.1183±0.0422	0.1179±0.0341	0.2225	0.1930	**0.0701**	0.0764	-0.3817
IBM	0.1345±0.0599	0.1237±0.0388	0.2771	0.2567	**0.0680**	0.0741	-8.7127
Oracle	**0.1676±0.0812**	0.1746±0.0781	0.4601	0.4161	0.0834	0.0828	**4.0233**
S&P500	0.1564±0.1016	0.1553±0.0677	0.5071	0.2873	0.0656	0.0588	-0.7397

Table 4. Experimental results for 3-10-5-1 MLP network architecture. Results are averaged over 30 independent trials.

	Avg. RMSE test		Max RMSE test		Min RMSE test		IR
	EKF-OR	EKF	EKF-OR	EKF	EKF-OR	EKF	%
Microsoft	**0.1135±0.0323**	0.1272±0.0291	**0.2020**	0.2177	**0.0697**	0.0809	**10.7343**
Apple	**0.1127±0.0273**	0.1253±0.0475	**0.1678**	0.2738	0.0743	0.0622	**10.0915**
Google	**0.1148±0.0448**	0.1171±0.0412	0.2632	0.2273	0.0612	0.0585	**1.9532**
Red Hat	0.1220±0.0406	0.1192±0.0358	0.2290	0.1936	**0.0674**	0.0706	-2.4020
Intel	**0.1094±0.0325**	0.1143±0.0356	**0.2136**	0.2333	0.0758	0.0675	**4.2718**
Yahoo	**0.1013±0.0538**	0.1136±0.0398	**0.2778**	0.3909	**0.0590**	0.0596	**10.8682**
IBM	0.1105±0.0383	0.1071±0.0368	**0.2372**	0.2516	**0.0659**	0.0668	-3.2058
Oracle	**0.1192±0.0340**	0.1256±0.0274	0.2163	0.1764	**0.0796**	0.0820	**5.0593**
S&P500	**0.0882±0.0319**	0.1119±0.0655	0.3980	0.1544	0.0578	0.0548	**21.1564**

Table 5. Experimental results for 3-10-10-1 MLP network architecture. Results are averaged over 30 independent trials.

	Avg. RMSE test		Max RMSE test		Min RMSE test		IR
	EKF-OR	EKF	EKF-OR	EKF	EKF-OR	EKF	%
Microsoft	**0.0936±0.0233**	0.1118±0.0328	**0.1710**	0.1982	**0.0684**	0.0712	**16.2928**
Apple	**0.0882±0.0280**	0.1052±0.0505	**0.1630**	0.2372	0.0554	0.0520	**16.1275**
Google	**0.0891±0.0290**	0.0958±0.0331	**0.1656**	0.1676	0.0547	0.0523	**7.0683**
Red Hat	**0.1119±0.0384**	0.1163±0.0605	**0.1956**	0.3223	0.0626	0.0594	**3.7394**
Intel	**0.1134±0.0277**	0.1250±0.0402	**0.2001**	0.2834	**0.0683**	0.0705	**9.2816**
Yahoo	**0.0896±0.0354**	0.1139±0.0602	**0.2093**	0.2709	0.0606	0.0544	**21.3280**
IBM	**0.1063±0.0306**	0.1119**±0.0379**	**0.2026**	0.2503	**0.0623**	0.0700	**5.0531**
Oracle	**0.1105±0.0295**	0.1129±0.0335	0.2005	0.1870	0.0775	0.0768	**2.0899**
S&P500	**0.0962±0.0358**	0.1008±0.0270	0.2111	0.1675	**0.0508**	0.0614	**4.5604**

ACKNOWLEDGMENT

This work is supported by the Serbian Government-the Ministry of Education, Science and Technological Development-Project title: An innovative, ecologically based approach to the implementation of intelligent manufacturing systems for the production of sheet metal parts (2011–2015) under grant TR35004.

REFERENCES

[1] Agamennoni, G., Nieto, J. and Nebot, E.: Approximate Inference in State-Space Models with Heavy-Tailed Noise. IEEE Transactions on Signal Processing, Vol. 60, No. 10, pp. 5024-5037, 2012.

[2] Vuković, N. and Miljković, Z.: Robust Sequential Learning of Feedforward Neural Networks in Presence of Heavy Tailed Noise, Neural Networks, Vol. 63, pp. 31-47, 2015.

[3] Vuković, N. and Miljković, Z.: A growing and pruning sequential learning algorithm of hyper basis function neural network for function approximation, Neural Networks, Vol. 46, pp. 210-226, 2013.

[4] Vuković, N.: *Machine Learning of Intelligent Mobile Robot Based on Artificial Neural Networks*, PhD thesis (in Serbian). DOI: 10.2298/BG20120928VUKOVIC. University of Belgrade – Faculty of Mechanical Engineering, 2012.

[5] Miljković, Z., Vuković, N., Mitić, M. and Babić, B.: New hybrid vision-based control approach for automated guided vehicles, The International Journal of Advanced Manufacturing Technology, Vol. 66, No 1-4, pp. 231-249, 2013.

[6] Babić, B., Miljković, Z., Vuković, N. and Antić, V.: Towards Implementation and Autonomous Navigation of an Intelligent Automated Guided Vehicle in Material Handling Systems, Iranian Journal of Science and Technology – Transaction B: Engineering, Vol. 36, No. M1, pp. 25-40, 2012.

[7] Miljković, Z., Mitić, M., Lazarević, M. and Babić, B.: Neural Network Reinforcement Learning for Visual Control of Robot Manipulators, Expert Systems with Applications, Vol. 40, No. 5, pp. 1721-1736, 2013.

[8] Mitić, M. and Miljković, Z.: Neural Network Learning from Demonstration and Epipolar Geometry for Visual Control of a Nonholonomic Mobile Robot, Soft Computing, Vol. 18, No. 5, pp. 1011-1025, 2014.

[9] Mitić, M. and Miljković, Z.: Bio-inspired Approach to Learning Robot Motion Trajectories and Visual Control Commands, Expert Systems with Applications, Vol. 42, No. 5, pp. 2624-2637, 2015.

[10] Vuković, N. and Miljković, Z.: New Hybrid Control Architecture for Intelligent Mobile Robot Navigation in a Manufacturing Environment, FME Transactions, Vol.37 No.1, pp. 9-18, 2009.

[11] Miljković, Z and Aleksendrić, D.: *Artificial neural networks—solved examples with theoretical background* (In Serbian), University of Belgrade-Faculty of Mechanical Engineering, 2009.

[12] Stanković, S.S. and Kovačević, B.D.: Analysis of robust stochastic approximation algorithms for process identification, Automatica, Vol. 22, No. 4, pp. 483-488, 1986.

[13] Đurović, Ž.M. and Kovačević, B.D.: Robust estimation with unknown noise statistics, IEEE

Transactions on Automatic Control, Vol. 44, No. 6, pp. 1292-1296, 1999.

[14] Markou, M. and Singh, S.: Novelty detection: a review—part 1: statistical approaches, Signal processing, Vol. 83, No. 12, 2481-2497, 2003.

[15] Markou, M. and Singh, S.: Novelty detection: a review—part 1: statistical approaches, Signal processing, Vol. 83, No. 12, 2499-2521, 2003.

[16] Huber, P. J.: *Robust statistics*. Springer, Berlin Heidelberg, 2011.

[17] Lee, C.C., Chung, P.C., Tsai, J.R. and Chang, C.I.: Robust radial basis function neural networks, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, Vol. 29, No. 6, pp. 674-685, 1999.

[18] Lee, C.C., Chiang, Y.C., Shih, C.Y. and Tsai, C.L.: Noisy time series prediction using M-estimator based robust radial basis function neural networks with growing and pruning techniques, Expert Systems with Applications, Vol. 36, No. 3, pp. 4717-4724, 2009.

[19] Chuang, C.C., Su, S.F. and Chen, S.S.: Robust TSK fuzzy modeling for function approximation with outliers, IEEE Transactions on Fuzzy Systems, Vol. 9, No. 6, pp. 810-821, 2001.

[20] Chuang, C.C., Su, S.F., Jeng, J.T. and Hsiao, C.C.: Robust support vector regression networks for function approximation with outliers, IEEE Transactions on Neural Networks, Vol. 13, No. 6, pp. 1322-1330, 2002.

[21] Chuang, C.C., Jeng, J.T. and Lin, P.T.: Annealing robust radial basis function networks for function approximation with outliers, Neurocomputing, Vol. 56, pp. 123-139, 2004.

[22] Chuang, C.C. and Jeng, J.T.: CPBUM neural networks for modeling with outliers and noise, Applied Soft Computing, Vol. 7, No. 3, pp. 957-967, 2007.

[23] Chuang, C.C. and Lee, Z.J.: Hybrid robust support vector machines for regression with outliers, Applied Soft Computing, Vol. 11, No. 1, pp. 64-72, 2011.

[24] Yang, Y.K., Sun, T.Y., Huo, C.L., Yu, Y.H., Liu, C.C. and Tsai, C.H.: A novel self-constructing Radial Basis Function Neural-Fuzzy System, Applied Soft Computing, Vol. 13, No. 5, pp. 2390-2404, 2013.

[25] Fu, Y.Y., Wu, C.J., Jeng, J.T. and Ko, C.N.: ARFNNs with SVR for prediction of chaotic time series with outliers, Expert Systems with Applications, Vol. 37, No. 6, pp. 4441-4451, 2010.

[26] Beal, M. J.: *Variational algorithms for approximate Bayesian inference*, PhD thesis. University of London, 2003.

[27] Bishop, C.: *Pattern recognition and machine learning*, Springer, Berlin, 2006.

[28] Barber, D.: *Bayesian reasoning and machine learning*, Cambridge University Press, 2012.

[29] Tzikas, D.G., Likas, C.L. and Galatsanos, N.P.: The variational approximation for Bayesian inference, IEEE Signal Processing Magazine, Vol. 25, No. 6, pp. 131-146, 2008.

[30] Simon, D.: Training radial basis neural networks with the extended Kalman filter, Neurocomputing, Vol. 48, No. 1, pp. 455-475, 2002.

[31] Andrieu, C., De Freitas, N. and Doucet, A.: Robust full Bayesian learning for radial basis networks, Neural Computation, Vol. 13, No. 10, pp. 2359-2407, 2001.

[32] Vuković,N., and Miljković,Z.: Machine Learning of Radial Basis Function Neural Networks with Gaussian Processing Units Using Kalman filtering– Introduction, TEHNIKA. Vol. LXIX, No. 4, pp. 613-620, 2014.

[33] Vuković,N., Miljković,Z.: Machine Learning of Radial Basis Function Neural Networks with Gaussian Processing Units Using Kalman filtering – Experimental Results (in serbian), TEHNIKA, Vol. LXIX, No. 4, pp. 621-628, 2014.

[34] http://finance.yahoo.com/ (last date of access: 23rd November 2014)

ВАРИЈАЦИОНИ ПРИСТУП РОБУСТНОМ ОБУЧАВАЊУ ВИШЕСЛОЈНОГ ПЕРЦЕПТРОНА НА БАЗИ БАЈЕСОВСКЕ МЕТОДОЛОГИЈЕ

Најдан Вуковић, Марко Митић, Зоран Миљковић

У раду је приказан и изведен нови секвенцијални алгоритам за обучавање вишеслојног перцептрона у присуству аутлајера. Аутлајери представљају значајан проблем, посебно уколико спроводимо секвенцијално обучавање или обучавање у реалном времену. Линеаризовани Калманов филтар робустан на аутлајере (ЛКФ-РА), је статистички генеративни модел у коме је матрица коваријанси шума мерења моделована као стохастички процес, а априорна информација усвојена као инверзна Вишартова расподела. Извођење свих једнакости је базирано на првим принципима Бајесовске методологије. Да би се решио корак модификације примењен је варијациони метод, у коме решење проблема тражимо у фамилији расподела одговарајуће функционалне форме. Експериментални резултати примене ЛКФ-РА, добијени коришћењем стварних временских серија, показују да је ЛКФ-РА бољи од конвенционалног линеаризованог Калмановог филтра у смислу генерисања ниже грешке на тест скупу података. Просечна вредност побољшања одређена у експерименталном процесу је 7%.