University of Southampton

Masters Thesis

Sequential Parameter and Model Estimation for Financial Timeseries.

Author:
Ajibola Loko

Supervisor:
Prof. Mahesan Niranjan

A thesis submitted in fulfilment of the requirements for the degree of Master of Science

in the

Control Systems and Signal Processing
Electronics and Computer Science

September 2015

Declaration of Authorship

I, Ajibola Loko, declare that this thesis titled, 'Sequential Parameter and Model Estimation for Financial Timeseries.' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: ____

Date: ____

“Thanks to the Holy Spirit, my strength, guide and all in all.”

Loko, Ajibola

UNIVERSITY OF SOUTHAMPTON

Abstract

Faculty of Physical Sciences and Engineering
Electronics and Computer Science

Master of Science

Sequential Parameter and Model Estimation for Financial Timeseries.

by Ajibola LOKO

This research reveals how financial options prices may be sequentially estimated by means of an Extended Kalman Filter(EKF) algorithm. The Black-Scholes model has been widely accepted and proven to yield quite accurate results if properly parameterized. Shortcomes in the Black-Scholes model estimation frequently occurs due to the various assumptions the model model is structured on. The Kalman filter is put to the test of overcoming the structural defects of the Black-Scholes model by sequentially estimating crucial parameters rather than obtaining the parameters based on the history of the dataset. Also, a Radial Basis Function(RBF) Network is used to model the non-linearity of the system and the EKF is put to the test of sequentially estimating RBF Network to adaptively capture the dynamics of the system.

Furthermore, the EKF is extended to the Second Order EKF and combined with the Black-Scholes and RBF models in order to capture the dynamics of the system when the non-linearity is high. Results of the simulation on FTSE100 Index options data showing dynamic RBF model outperforming the BS model is presented and discussed.

Acknowledgements

The completion of this dissertation would have been impossible but for the support and guidance of some special people. I would like to express my sincere gratitude to the following: Mahesan Niranjan, my supervisor and great mentor. For your patience and understanding in building me up from the little i knew to who i am today. For the encouraging words and stretching me to uncover my hidden abilities all through my Masters program.

Engr. Samuel Kayode Loko and Mrs Comfort Olubusola Loko, my loving parents. For your faith, support and prayers for me. For the guidance and immense love you showed me, throughout my entire life and especially in this times that i needed you the most.

Bing Chu, my program tutor. For the directions and advice that saw me come out strong through difficult times.

Contents

List of Figures

List of Tables

Abbreviations

BS	**B**lack **S**choles
EKF	**E**xtended **K**alman **F**ilter
SO-EKF	**S**econd **O**rder **E**xtended **K**alman **F**ilter
NN	**N**eural **N**etwork
RBF	**R**adial **B**asis **F**unction
RMSE	**R**oot **M**ean **S**quared **E**rror

To my parents, Engr. Samuel Kayode Loko & Mrs. Comfort Olubusola Loko

Chapter 1

Introduction

The signal processing community has in recent years drawn a lot of interest in time series analysis. Various real world problems having the nature of a time series include: weather forecasting, communication, pattern recognition, computational finance, astronomy, etc.

Time series simply describes a dataset whereby successive measurement or observations arrive sequentially at a fixed time interval. The dynamics of systems that can describe time series tend to be time varying, hence most time series require sequentially adjusting model parameters in order to enable accurate analysis the system.

Popular amongst time series is the financial time series. Financial time series analysis draws a lot of attention due to its direct impact on lives and the economy. Embedded in such time series is a high level of complexity due to the stochastic nature of market forces.

Various models have been developed over the years to analyse financial time series and quite notable amongst the pack is the Black-Scholes models used for describing financial derivatives.

Financial derivatives are also financial time series that have strong ties with the respective financial entities which they depend on. Such financial entities include equities, indices, commodities, etc. Financial derivatives are traded to reduce the risk on investments made on the underlying financial entity. Black-Scholes model is one that describes the nature of financial derivatives based on a set of parameters. Black-Scholes model has shown impressive accuracy over the years provided that this parameters abide by the set of assumptions made when the model was discovered

This set of assumptions poses as a defect to the Black-Scholes model. Signal processing techniques suitable for rectifying defects in model structure of this nature simply require the propagation of errors in parameters that make up the structure of the model.

Various methods of estimating the error and details of the error in statistical signal processing include: Minimum Variance Estimation, Least Square Error, Minimum Mean Square Error, Maximum A Posteriori, etc.

The Minimum Variance Estimator is of more interest in this project being the basis of the Kalman Filter. The Kalman filter perfectly fits in this project as an algorithm that sequentially propagates the error in the parameters that make up the model. The Kalman filter can be expanded by Taylor series to become the Extended Kalman Filter suitable for working with non-linear systems and further expansion of the Extended Kalman filter by Taylor series to the Second Order Extended Kalman Filter may enable the filter capture high levels of non-linearity in the system.

Situations whereby the defects in the Black-Scholes model cannot be circumvented by simply changing the parameters of the model may exist, and the the only solution to this is the implementation of a different model, better than the Black-Scholes model.

The rise in the use of Neural Networks as one of the best model estimators in the Machine learning environment provides us with a platform to fuse the Kalman filter and a Neural Network model into an algorithm to fix time series analysis problems.

This project focuses on analysis of the sequential estimation of parameters for the Black-Scholes model as well as sequential estimation of a Neural Network model on typical time series data.

1.1 Aims and Objectives

The aim of this project is to investigate the sequential estimation of parameters that describe a model compared to sequential estimation of a neural network model using Kalman Filters extended to both the first and second order of the Taylor series expansion of the model.

Achieving this aim is possible by first designing and analyzing the steady state response of both models using Kalman filter algorithms. This then enables us to tune parameters of the Kalman filter properly to reduce the settling time of the system as much as possible. With the algorithms and their respective suitable parameters in place, the simulation on real data is carried out. Conclusions can then be drawn based on inference from the results from the simulation.

1.2 Structure of the report

In chapter two, a review of the previous related work in sequential estimation as relating to non-linear systems, the Black-Scholes model and the Neural Network Models. A systematic progression is taken from the State space representation theorem and how this lays the foundation for the Kalman filter. merging the filter with the model based on parameter estimation and model estimation techniques then concludes this chapter.

In chapter three, the Kalman filter algorithms (Extended Kal;man Filter and the Second Oreder Extended Kalman Filter) are formulated. The Black-Scholes and Neural Networks are fitted into the algorithms for steady state analysis. Close attention is paid to tuning the various initial parameter of the filter to make the choice of initial parameters that best suite the form of real data intended to be analysed.

Chapter four shows the results of experiments of the developed algorithms on real data and discussed the implication of the performances recorded. An analysis time management of this project is shown in chapter five. Conclusions drawn from the entire project and discussion on ways to extend this project further is contained in chapter six.

Chapter 2

Literature Review

2.1 Estimation

In the control and signal processing environment, estimation is a broad topic that covers mainly the determination/prediction of values of parameter based on some form of data available but with the presence of inherent noise. This poses a huge problem in various field that involve data/signal processing such as communication, financial industry, engineering industry, etc. Various theorems have been discovered especially in the statistical signal processing community on mathematical methods of solving the problem of estimation [1]. In this section, we review the underlying theory behind the various estimation techniques and arrive at the estimator that best suites the nature of the data under research in this project.

2.1.1 Probabilistic-Based Estimation

In the statistical signal processing environment, probability based methods such as Maximum-a-Posteriori(MAP), Maximum Likelihood Estimate(MLE), Minimum Mean Squared Error(MMSE) have proven to be good estimators [2]. An underlying principle to determining the probabilities in these methods is the use of Bayesian solutions. The Bayesian solution becomes very useful when we can take advantage of some prior knowledge of the distribution of variables in the system. [3] did extensive research in this by embedding Bayesian probabilities within a Monte carlo simulation(popularly known as particle filters). There they showed how this method is well suited for systems having non-Gaussian and non-linear properties.

The core of this method is dependent on the number of samples used , since it involve the idea of monte carlo, whereby the more samples used, the higher the accuracy of the results gotten. This, most times, contributes to having to trade off some accuracy for computational cost by reducing the number of samples used during computation.

2.1.2 Deterministic-Based Estimation

The main deterministic based estimation in statistical signal processing is the Least Square Estimation(LSE). This approach simply attempts to minimise the squared difference between the true signal and the estimated signal. This difference is sometimes referred to as the cost or error function$J(\theta)$. The LSE may be mathematically defined as:

$$\hat{\theta} = \arg\min_{\theta} J(\theta)$$

$$J(\theta) = \sum_{n=1}^{N} (e^2)$$

where e is the error in the estimation and $J(\theta)$ may be seen as a function describing the covariance of the error. This invariably leads us to the foundation to help minimize the variation in the estimation error as initially intended.

An idea of the nature of the distribution of the states is also very important. The states in this system is solely a function of the of the randomness of the price of the underlying asset. Taking cues from the assumptions made by the Black-Scholes model, and actually carrying out a test on recent financial data confirms that the probability distribution of the price of the underlying asset for annually is log-normally distributed.

The validity of this assumption gives us an edge as to making a decision about the distribution of the states. We simply achieved this by generating random values from a log-normal distribution to represent synthetic price of underlying asset, then we take a look at how the states are distributed with this(see fig).

Based on the fact that the results shows some traces of the function of a Gaussian distribution, we therefore adopt sequential least square estimation suitable for Gaussian distributions.

2.2 State Space Representation

Taking cue from control engineering, the state space representation is an ideal starting point for sequential estimation problems. The state space equations are simply representation of a system using mathematical models which describe the relationship between the input, the output and the state variables of the system.

The state variables of the system is a set of the minimum number of independent variables that can be used to accurately describe the entire system. The state space representation can be divided into two: the state equation and the measurement equation.

The nature of financial derivatives(Options pricing being the case study for this project) as a non-linear system lead us to taking a good look at a parametric model approach and a non-parametric model approach.

Most popularly accepted amongst the parametric model approaches to options pricing is the Black-Scholes model. While a non-parametric model approach to the pricing of options will imply estimation of a model that estimates with a high degree of performance the options prices. [4] exhausted various neural network approaches and of interest is the RBF approach he proposed to be of the best accuracy in his literature.

In order to establish a sequential mode estimation technique using the Kalman filter, it is necessary to define the state space representation of whichever of the models is to be used as this helps define the relationship between the observation and the latent variables which serve as inputs to the filter.

Due to the stochastic nature of the system under consideration, the state parameters are totally random parameters and hence, the best relationship between the previous state and the next state is simply by adding a random number to the previous. This random number is bounded in variance. This variance is assumed to be possibly determined from the variation of the states from the past state data. The state equation adopted for this project shall be:

$$\theta(t+1) = \theta(t) + w(t) \tag{2.1}$$

where $\theta(t)$ is a vector containing the latent parameters to be estimated t is the time sequence and $w(t)$ is the random vector variable representing the stochastic nature of the state vector.

The observation equation is simply determined from the model (parametric or non-parametric) under observation. This is expressed as:

$$z(t+1) = h\big(\theta(t+1), U(t)\big) + v(t+1) \tag{2.2}$$

where $z(t+1)$ is the output or observation from the system. $U(t)$ contains the input variables to the system and $v(t)$ is the noise in the estimated measurement. the variation of this noise is assumed to be possibly determined from the variation of estimated observations using the model with respect to the true observation within the past data.

From the state space equations above, it is clear that the problem boils down to two main issues; being the accurate estimation of the initial state ($\theta(1)$) and every other state($\theta(t)$) and the estimation of the noise variables ($w(t)$ and $v(t+1)$). In order to arrive at something worthwhile within the time constraint of this research, we shall focus mainly on the estimation of the states and improvements that can be made on the estimation technique. A good amount of research has been carried in previous work([5], [6]) on the estimation of the initial states and noise variables using Expectation Maximization (EM) or adoption of Bayesian Models.

2.3 Sequential Estimation

Sequential mode estimation of non-linear systems with the use of Kalman filters has become popular over the years, [7] shows the step by step derivation of the Extended Kalman filter and Second order Extended Kalman Filter with respect to the Taylor series expansion of the non-linear equation that gives the state parameters and the non-linear equation that links the observations to the state parameters. [2]

Statistical signal processing time series (i.e. a system whereby the availability of data is in a sequential manner) usually involves either a batch mode estimation or an on-line mode(recursive) estimation technique. This is in order to avoid the problem of limit to storage or historical data as well as computational processing of a large amount of data each time a new data arrives. Batch mode estimation also has the inherent problem of computational cost especially if the optimal batch for achieving a good estimate is a lot. Recursive estimation has become popular nowadays with time series problems and has proven in most researches to be quite accurate.

A generic model for the recursive estimation techniques involves the use of prediction and update stages as shown in the figure below.

The general aim of a recursive estimator is to be able to propagate as much as possible, the properties of the error in each estimate. On successfully doing this, one can effectively reduce the error made on subsequent estimates. One approach to ensuring we get minimum error is by getting a weighted error which when summed up to the estimated state gets us as close as possible to the true state.

Also noteworthy is the fact that all time series problems tend to have data driven approaches as the best solutions. This leads us to the need to consider statistical models suited to minimizing the error in estimations. A key indicator of the level of certainty of the estimation is embedded in the variance of the error in estimation. Statistical model approaches are usually based on either some probabilistic distribution or it could be deterministic.

There are different methods for parameter estimation generally but most useful for time series is the batch mode estimation and the sequential mode estimation given that the data arrives in bits for processing to be done. [4] made an in-depth analysis on the RBF using the batch mode to estimate the Black-Scholes equation and showed that reasonable results can be achieved doing this. Taking a cue from [8]'s claim that the RBF has the best approximation property and combining this with the notion that generally sequential mode estimation often gives better performance than the batch mode method especially on large scale, non-convex problems (see [9], [10], and [11]), we therefore take a critical look at sequential mode estimation using the Kalman Filter.

2.4 Kalman Filter

The Kalman filter is a data processing algorithm that bases its estimation on the recursively optimizing the least squared error. This filter's optimality encompasses most criteria that may be set aside as a measure of performance of the filter. The Kalman filter is designed in such a way as to incorporate [12], [13]:

- knowledge of the dynamics of the system by the use of the model describing the relationship between the states.
- knowledge of the dynamics of the observation from the system by the use of the model defining the relationship between the states and the observation
- statistical distribution of the noise inherent in the states model and the observation's model
- prior knowledge of the initial values or conditions of the state of the system and the variance of the error of this initial state.

A main advantage of the Kalman filter lies in the fact that it requires only the the immediate previous measurement to be stored in order to make an estimate of the next measurement. The ability of Kalman filter to repeatedly combine all this information to produce an accurate estimate of the desired measurement a step ahead is mainly anchored on the way the algorithm is structured to pay close attention to the error made on immediate past estimations and in-turn make the right adjustment to the system to reduce the error made in consequent estimations to the barest minimum.

This filter is based on a the Minimum Mean Square Error Estimation (MMSE) technique. The algorithm simply propagates the mean and covariance of the of the states every period of the sequence while updating these values to ensure that the error made in the previous estimation is embedded in the model for the estimation of the measurement for the next time instance.

The normal Kalman filter is well suited for Linear systems and has successfully been put to use frequently in recent years [14]. An extension to the normal Kalman filter is the Extended Kalman filter(EKF). The EKF algorithm is based on extension of the approximation of models used to describe the state space equations. These approximate models are not just linear versions of the models, as obtained in the normal Kalman filter, but expanded using the Taylor series generic formula:

$$f(x) = f(a) + \frac{\partial f}{\partial a}(x - a) + H.O.T$$

where a is a point where $f(x)$ is indefinitely differentiable, $f(x)$ represents the model for estimation and $H.O.T$ are the Higher Order terms which are ignored in this case.

2.5 Models

The Model constitutes a very crucial aspect of the working of the sequential estimation algorithm. A model simply refers to a mathematical formula or function that accurately defines the relationship between the input and output of a particular system. A model can be seen as the aspect of the system where the data that comes into the system gets processed to give meaningful output from the system. In this section, attention is given to the use of as parametric and a non-parametric model. The parametric model being the Black-Scholes Equation is a popularly accepted model in the financial industry that has proven to produce highly accurate results in defining the dynamics of the financial derivatives system [15]. The non-parametric model is drawn from the machine learning community in the area of Neural networks. This is further discussed in subsequent sections.

2.5.1 Black Scholes Model

The Black-Scholes Model developed by Fischer Black and Myron Scholes in 1973 [16] is popularly accepted as one of the most accurate models for rightly pricing financial derivatives, hence, it has drawn so much interest over the years to improve on its accuracy even more. [15] exhaustively gave a breakdown of the working principle behind the model and how it can be applied to the financial market. In the case of European options pricing, the Black-Scholes model narrows down to:

$$C = SN(d_1) - X\exp(-rt_m)N(d_2) \tag{2.3}$$

$$P = -SN(-d_1) + X\exp(-rt_m)N(-d_2) \tag{2.4}$$

where

$$d_1 = \frac{ln(S/X) + (r + \frac{\sigma^2}{2})\sqrt{t_m}}{\sigma\sqrt{t_m}}$$
$$d_2 = d_1 - \sigma\sqrt{t_m}$$

$N(x)$ represents the cumulative probability distribution function of x, for a standardized normal distribution, C is the European Call option price while P is the European Put option price.

The variables: Price of the underlying asset(S), the strike price (X), Time to maturity (t_m), volatility (σ) i.e. the variation of the returns on the asset and continuously compounded risk-free interest rate (r) are the main parameters needed to compute the option price using this formula.

S, X and t_m are readily available as data from the financial market or can easily be determined. r is also available but has no fixed universal value at any given time. σ is unavailable and is quite difficult to rightly estimate since it depends on the returns.

2.5.1.1 Volatility

The volatility is usually estimated from the historical price of the underlying asset. One of the simplest methods for estimation of the volatility is by simply computing:

$$\hat{\sigma} = \frac{s}{\sqrt{\tau}} s = \sqrt{\tfrac{1}{n-1}\textstyle\sum_{i=1}^{n}(u_i - \bar{u})^2}$$

where s is the standard deviation, $\hat{\sigma}$ is the estimated volatility and τ is the change in time or the number of days for which to estimate the volatility. More complex methods for estimating the volatility exist such as the GARCH model [15]. [**?**] also suggested estimating volatility as:

$$\hat{\sigma} = \frac{s}{\sqrt{60}}$$

where s is the standard deviation for the compounded returns for the past 60 days.

In order to obtain the real volatility of the the underlying asset as of a particular time, the use of the Black-Scholes model has proven to be a useful tool. The Black-Scholes model is usually used to obtain the volatility after the price of the option has been determined by the market. The nature of the market gives room for the price of the option to be influenced by various economic parameters which are outside the scope of most parametric models, hence, getting a good estimate of the volatility becomes really difficult when the market forces become really strong. However, the Black-Scholes model gives us an idea of what the true volatility should have been and this is termed "Implied volatility". Estimation as close as possible to this implied volatility is of utmost importance to the aspect of this research that involves the Black-Scholes model. [17]

2.5.1.2 Risk-free Interest Rates

Estimation of the value of the risk-free interest rate is quite important to accurately estimate the options price. Most widely accepted measures of risk-free interest rates are the rates on Treasury Bonds(also called Gilts in United Kingdom) issued by the government or the interest rates on the Treasury bills. Other accepted short-term risk-free interest measures include the London Interbank Offered Rate(LIBOR) and Repurchase agreement (Repo) rate.[4]

Getting the value of any of these interest rates usually comes at a cost especially in a case whereby it has to be gotten in time enough to be useful for estimating the options price for business deals. Also, an assumption made in the derivation of the Black-Scholes equation is that the risk-free interest rate remains the same for different lengths of time to maturity. This is not obtainable in real life though the difference may be quite little.

2.5.2 Radial Basis Function Network

RBF is a function that depends on a distance of data points from a particular point C (this could be the center point). Several distance functions are possible in order to make this calculation. One commonly used distance function is the euclidean distance which is the norm of the difference between each data point and a particular center point

$$\phi(x, C) = \phi(x - c) \tag{2.5}$$

considering the core aim of the project is to map multiple input (x) variables to a univariate output, the use of RBF has been proven by Poggio and Girosi (1990) to be a good tool for this approximation task. They derived the RBF by regularization. Given an unknown function $y = f(x)$ where y is a dependent scalar to be approximated and x is a multivariate independent dataset. The approximated function is expected to make errors so that our new function i:

$$y_t = f(x_t) + \epsilon$$

It is intended that ϵ is as close to zero as possible.In the best case scenerios, we have

$$E[\epsilon_t|x_t] = 0$$

Applying regularization in order to minimize ϵ to the barest minimum, we achieve the following equation:

$$H(\hat{f}) = \sum_{t=1}^{T}(y_t - \hat{f}(x_t)^2 + P[\hat{f}(x_t)]^2)$$

where the first term of the summation gives the square of the error while the second term in the summation is a penalty function with being the weight of the penalty and P[.] being a derivative of the approximated function.

A general form of a solution giving the approximate function (x) is:

$$\hat{f}(x_t) = \sum_{i=1}^{k} c_i h_i(x - z) + p[x]$$

where z are center vector of the the dataset (x_t), c_i is a scalar weighting coefficient, h_i is a set of radial basis functions and p(.) is a polynomial dependent on x

Hutchinson's research which yielded good results suggested the radial basis function being the square of the weighted difference between the datapoint and the mean while the polynomial function is simply a linear function of x and a constant:

$$\hat{f}(x_t) = \sum_{i=1}^{k} c_i h_i\big((x - z_i)^T W^T W (x - z_i)\big) + w_1^T x + w_0 \tag{2.6}$$

Hutchinson suggests that this form of a RBF network reflect some prior knowledge of the nature of the data on the output and proves to be computationally efficient compared to the usual Gaussian radial basis function.

Chapter 3

Design and Implementation

3.1 Sequential Estimation Algorithm

The solution to the sequential estimation problem is finally arriving at the corrected State estimate, $\theta(t+1|t+1)$ and the updated state covariance estimate $P(t+1|t+1)$. In this situation whereby we are simply required to track a particular value, recursively estimating the states and its covariance allows us avoid keeping record of the history of all the previous states and their respective covariances. Now, provided we can accurately estimate the states and the covariance, we can easily arrive at accurate price measurements using the right model. This section shows how the states and its covariance may be estimated using the Kalman filter algorithm.

Estimation of $\theta(t{+}1|t{+}1)$ and $P(t{+}1|t{+}1)$ is computed in two stages: The prediction and the correction/update stage. During the prediction stage, information about the present state of the system is propagated into the future using an function/model, $f\big(\theta(\mathbf{t})\big)$, that rightly estimates the next state vector, we can then predict a state estimate as:

$$\hat{\theta}(t+1|t) = f\big(\theta(\mathbf{t})\big) + \mathbf{w}(\mathbf{t})$$

where $w(t)$ is a noise vector having the same size as the state. The covariance of the estimated state is then computed based on some information contained in the model used for estimation of the state. The partial derivative of the model with respect to each of the parameters of the state vector gives us an idea of the gradient and is useful in computing the covariance of the error made in the estimation of the state vector. This may be obtained as:

$$\mathbf{P}(t+1|t) = \mathbf{J}(t)\mathbf{P}(t|t)\mathbf{J}(t)^T + \mathbf{Q}(t)$$

where $\mathbf{J}(t) = \frac{\partial f}{\partial x}$ described the gradient of the state estimation model with respect to the each parameter contained in the state vector and is normally described as the Jacobian. $\mathbf{Q}(t)$ is the state noise covariance. $w(t)$ is usually randomly selected from a zero-mean Gaussian distribution with a covariance $\mathbf{Q}(t)$

Next, the correction/update stage involves the application of the minimum variance estimation technique being the core of the Kalman filter. The major focus is to minimize the variance of the error in the estimation of the state parameters. The first step in the correction of the predicted state and state-error covariance is to compute the error made on the measurement, $e(t)$ and also compute its measurement error covariance $S(t)$ as:

$$e(t) = z(t) - h\big(\hat{\theta}(t+1|t)\big) + v(t)$$

$$\mathbf{S}(t) = \mathbf{H}(t+1)\mathbf{P}(t+1|t)\mathbf{H}(t+1)^T + R(t)$$

where $\mathbf{H}(t+1) = \frac{\partial h}{\partial x}$ described the gradient of the measurement estimation model with respect to the each parameter contained in the state vector and is normally described as a Jacobian. $\mathbf{R}(t)$ is the measurement noise covariance. The measurement error,$v(t)$ is usually randomly selected from a zero-mean Gaussian distribution with a covariance $\mathbf{R}(t)$

Next, the Kalman gain, a factor derived from the minimum state and measurement error covariances (detailed proof shown in Appendix B) is obtained by:

$$\mathbf{K}(t) = \mathbf{P}(t+1|t)\mathbf{H}(t+1)^T\mathbf{S}(t)^{-1}$$

The state estimates and the state-error covariance are now corrected using the Kalman gain as:

$$\hat{\theta}(t+1|t+1) = \hat{\theta}(t+1|t) + \mathbf{K}(t)e(t)$$

$$\mathbf{P}(t+1|t+1) = \big(\mathbf{I} - \mathbf{K}(t)\mathbf{H}(t+1)\big)\mathbf{P}(t+1|t)$$

This corrected state and state-error covariance are ready to be used for the next prediction of the state and eventually a measurement estimate. The right choice of $\theta(1|1)$, $P(1|1)$, Q and R arise as the next question. This is a general issue that relates to sequential estimation problems. tuning this parameters based on prior knowledge of the data is a widely used approach that has been used to obtain good results.

The full algorithm for estimation of the states and consequently the observations is described mathematically as:

Prediction:

$$\text{State Estimate} \qquad \hat{\theta}(t+1|t) = f\big(\theta(\mathbf{t})\big) + \mathbf{w}(\mathbf{t}) \tag{3.1}$$

$$\text{State-error covariance estimate} \qquad \mathbf{P}(t+1|t) = \mathbf{J}(t)\mathbf{P}(t|t)\mathbf{J}(t)^T + \mathbf{Q}(t) \tag{3.2}$$

Correction:

$$\text{Measurement error} \qquad e(t) = z(t) - h\big(\hat{\theta}(t+1|t)\big) \tag{3.3}$$

$$\text{Measurement error covariance} \qquad \mathbf{S}(t) = \mathbf{H}(t+1)\mathbf{P}(t+1|t)\mathbf{H}(t+1)^T + R(t) \tag{3.4}$$

$$\text{Kalman Gain} \qquad \mathbf{K}(t) = \mathbf{P}(t+1|t)\mathbf{H}(t+1)^T\mathbf{S}(t)^{-1} \tag{3.5}$$

$$\text{Updated state-error covariance} \qquad \mathbf{P}(t+1|t+1) = \big(\mathbf{I} - \mathbf{K}(t)\mathbf{H}(t+1)\big)\mathbf{P}(t+1|t) \tag{3.6}$$

$$\text{Updated State estimate} \qquad \hat{\theta}(t+1|t+1) = \hat{\theta}(t+1|t) + \mathbf{K}(t)e(t) \tag{3.7}$$

where

$$\mathbf{J}(t) = \frac{\partial f}{\partial x} \tag{3.8}$$

$$\mathbf{H}(t+1) = \frac{\partial h}{\partial x} \tag{3.9}$$

both indicate the Jacobian matrices of the partial derivatives of the state equation and measurement equations respectively.

3.2 Estimation of the Black-Scholes Model Parameters

According to Merton(1990), the Black-Scholes formula is homogenous of degree one in both the price of the underlying asset S(t) and the strike (X) price. For this reason, the network only needs to learn a function that depends on two input variable; the ratio - $S(t)/X$ and the time to maturity(t_m)

Utilizing the sequential estimation based on the Kalman Filter algorithms in order to estimate the random variable that are not readily available(r and σ) narrows down to the having a state space equation as:

$$\text{State Equation} \qquad \theta(\mathbf{t+1}) = \theta(\mathbf{t}) + \mathbf{w}(\mathbf{t}) \tag{3.10}$$

$$\text{Observation Equation} \qquad z(t) = h\big(\theta(\mathbf{t+1}), \mathbf{U}(\mathbf{t})\big) + \mathbf{v}(\mathbf{t+1}) \tag{3.11}$$

where

$$\theta = [\sigma \ r]^T \tag{3.12}$$

$$U(t) = S, \ X, \ t_m \tag{3.13}$$

and $h(.)$ is the Black-Scholes equation described in section 2.5.1
Considering the prediction phase, the stochastic nature of the states chosen can best be represented by a random walk, i.e. addition of random noise added to the previous state.

$$\theta(\mathbf{t+1}) = \theta(\mathbf{t}) + \mathbf{w}(\mathbf{t})$$

where

$$w(t) \sim N(0, Q(t))$$

The noise covariance $\mathbf{Q(t)}$ is set to an identity matrix with diagonal values that suites the random nature of the states from historical data.
The Estimated state-error covariance $P(t+1|t)$is simply set as:

$$P(t+1|t) = P(t|t) + Q(t)$$

since the partial derivatives of the random walk of the states is simply equates to 1.

On arrival of the real measurement, i.e. the option price, we begin the correction stage by the estimation of the error made from the prediction stage as:

$$e(t) = z(t) - h\big(\theta(\mathbf{t+1}), \mathbf{U(t)}\big)$$

The covariance of the error made on the prediction of the options price is computed as shown in equation 3.4. The first order derivative of the Black-Scholes Model is computed at this point. This derivatives are readily available formulas used by financial analyst in devicing strategies to hedge the risk on investments. They are collectively termed Financial Greeks or Risk sensitivities and the proof of the derivation is shown is Appendix A. From the formula of the Jacobian in 3.9, we have:

$$\frac{\partial h}{\partial x} = \begin{bmatrix} \frac{\partial C}{\partial \sigma} & \frac{\partial C}{\partial r} \\ \frac{\partial P}{\partial \sigma} & \frac{\partial P}{\partial r} \end{bmatrix} \tag{3.14}$$

where

$$\frac{\partial C}{\partial \sigma} = \frac{\partial P}{\partial \sigma} = \sqrt{t_m} S N'(d_1) \tag{3.15}$$

$$\frac{\partial C}{\partial r} = t_m X \exp(-r t_m) N(d_2) \tag{3.16}$$

$$\frac{\partial P}{\partial r} = -t_m X \exp(-r t_m) N(-d_2) \tag{3.17}$$

$\mathbf{R}(\mathbf{t})$ of the noise, $v(t+1)$, added to when estimating the option price in in the observation equation. Next the Kalman gain may be computed as stated in equation 3.5.

This Kalman gain factor has both the actual error covariances from the state prediction as well as the observation prediction. This is then used as a weight factor on the actual error in measurement prediction, $e(t)$.

Next the predicted state , $\theta(t+1|t)$ and state-error covariance, $P(t+1|t)$ are then updated to $\theta(t+1|t+1)$ and $P(t+1|t+1)$ as shown in equation 3.7 and equation 3.6 respectively. This entire cycle is then repeated each time a new measurement arrives.

3.3 Radial Basis Function Network Estimation

Generally, learning networks have been proven to be a good tool for estimation of complex non-linear models if properly trained. A typical example of such complex model is the Black-Scholes formula. [4] critically looked into how this formula may be learnt using the RBF network. He generated synthesized data by setting two of the input parameters of the Black-Scholes formula constant. The network was able to learn and approximate the formula with a good degree of accuracy within a span of 6 months duration of data.

The RBF proposed in section 2.6 may be re-written as:

$$\frac{\hat{C}}{X} = \sum_{i=1}^{k} \lambda_i \phi_i(\mathbf{x}, \mathbf{\Sigma}, \mathbf{m}) + \mathbf{w_1^T x} + \mathbf{w_0} \tag{3.18}$$

where $\phi_i(.)$ and x are:

$$\phi_i = \sqrt{(\mathbf{x} - \mathbf{m_i})^{\mathbf{T}} \mathbf{\Sigma^{-1}} (\mathbf{x} - \mathbf{m_i})} \tag{3.19}$$

$$\mathbf{x} = [\frac{S}{X} \; t_m] \tag{3.20}$$

λ and $\mathbf{w_1}$ are linear weight vectors, w_0 is set aside to be a bias while $\mathbf{m}$ and $\mathbf{\Sigma}$ stand for mean vector and covariance matrices respectively

The main idea behind this model is to track the non-linearity in the system by tracking the mean of the input vector $\mathbf{x}$. Bearing this in mind, we preset the remaining variables based on test carried out on historical data and focus on the sequential estimation of the mean as state variable.

The same state space representations described in equation 3.10 and equation 3.11 are adopted for this algorithm. Also, the sequential estimation procedure described in section 3.2 remains the same for the RBF model estimation with exception to a few equations. Here,

$$\theta = \mathbf{m}$$

$$U(t) = S/X,\ t_m$$

and $h(.)$ is the RBF model described in equation 3.18. Since the mean is dependent on the price of the underlying asset, we assume the best prediction of the mean can also simply be represented by a random walk.

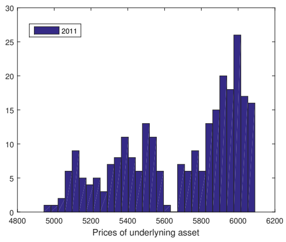
3.4 Synthesizing Data for Simulation

In order to test if the algorithm will give reasonable results when used on real data, it is necessary to synthesize data for simulation. This gives us an idea of the dynamics to expect from the various elements used for the build-up of this algorithm and eases tracking down errors in the simulation. Taking cue from **??**, we simply set the states to constant values and generate data using this state provided other variables are set to meet the conditions and/or be as close as possible to what is obtained in real situation which includes the log-normally distributed price of the underlying asset, the strike price close to the price of the underlying asset and a gradually reducing time to maturity.

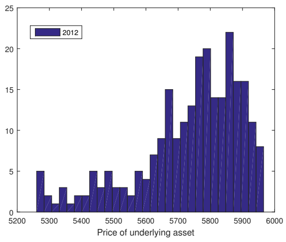
3.4.1 Black-Scholes Synthetic Data and Simulation

Since the state vector has been set to $[\sigma\ r]^T$, we decided to choose arbitrary values close to the average of what is obtainable for real data as $[0.1\ 0.05]^T$. This values will represent the volatility and risk-free interest rates throughout the life of the options contracct we wish to synthesize. We chose a time period of 200 days. Hence the time to maturity, t_m, is represented by a normalised value of the days such that 200 days to maturity is represented as 1.

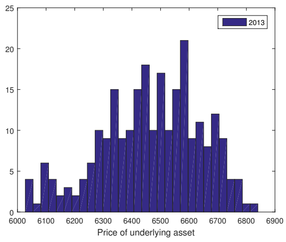
With reference to Black-Scholes assumptions on the probability distribution of the price of the underlying asset(S) as observation from distribution of recent financial data (see



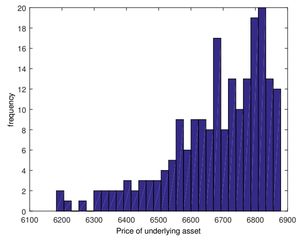
(a) 2011 FTSE100 data distribution



(b) 2012 FTSE100 data distribution



(c) 2013 FTSE100 data distribution



(d) 2014 FTSE100 data distribution

Figure 3.1: **Annual price distribution of the underlying asset.** Illustration of a distribution of the price of underlying asset in line with Black-Scholes assumptions

figure 4.3), $S(t)$ usually has a log-normal distribution [16]. I chose Strike prices(X) to be values close to the price of the underlying asset at the beginning of the life of the contract. With these, we generated the data(Call options prices) using the Black-Scholes Model.

The filter algorithm is setup as described in section 3.2 and the filter is put to test with the results shown in figure 3.2 and **??** below.

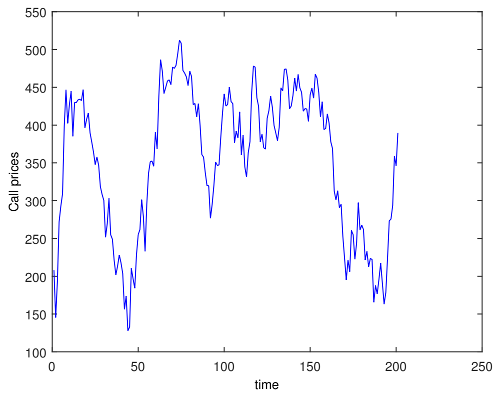
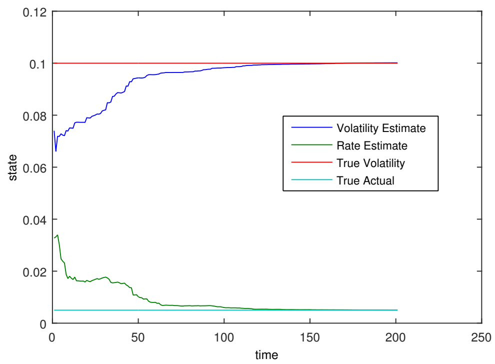
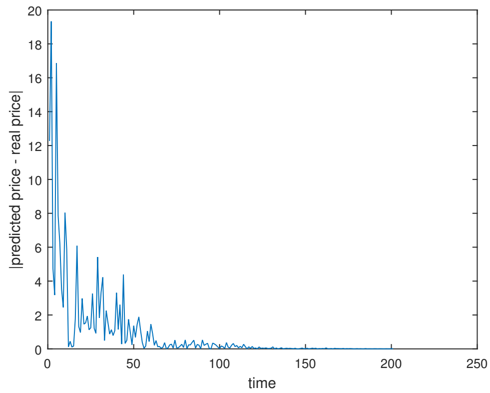


Figure 3.2: **The synthesized timeseries.** Data generated from steady-state shows a similar measure of non-linearity comparable to real data and is therefore suitable to run tests on



(a) Evolution of state vector estimation



(b) Absolute error on the One step ahead prediction

Figure 3.3: **EKF steady-state response using the Black-Scholes model.** Evolution of the state estimation from the EKF gradually converges to the true states in (a) and is reflected in the measurement error in (b)

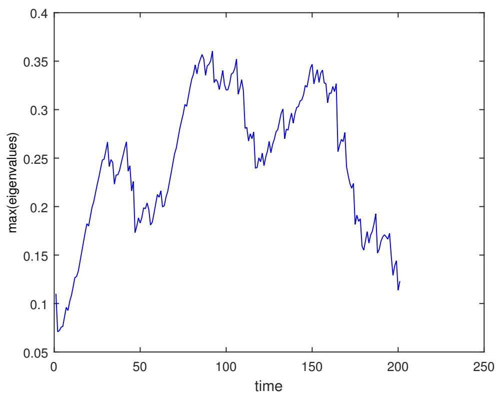


Figure 3.4: **Measure of uncertainty of the estimated states.** The covariance of the error initially increases as the EKF adapts to the non-linearity on the system. The drop in uncertainty towards the end reflects high accuracy in the estimation of the states

In order to ensure and test for the limits of the filter's ability to converge to the correct states, the initial state parameters, $\theta(\mathbf{1})$ are tuned appropriately (see fig 3.5). This gives us an idea of the settling time for various levels of step responses of the filter to the system.

The performance of the filter was evaluated by computing the Root Mean Squared Error(RMSE) of the estimated prices for cases with strike prices close that of the underlying price at the beginning of the life of the options contract. The results are shown in table 3.1 and 3.2.

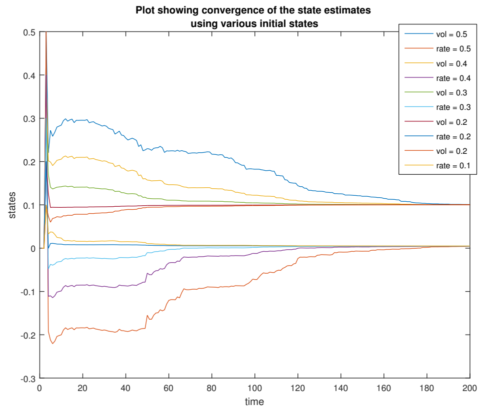


Figure 3.5: **EKF steady-state response with various initial state parameters.** Various levels of the step response of the system displays the EKF's ability/limits and expected settling time when adapting to changes in the system states

A major indicator of the dynamics of the filter is the state-error covariance matrix. This is very important because it gives us an idea of the certainty/confidence of predictions made by the filter. Under ideal filtering situations, the certainty of the prediction made by the filter should gradually increase as the filter estimation error reduces.

In order to display the extent of the covariance of the state-error, we compute the first principal component by calculating the eigenvalues of the covariance, $\mathbf{P}(\mathbf{t}+\mathbf{1}|\mathbf{t})$ and picking the maximum eigenvalue, this gives us an idea of the width of the variance along the axis of maximum spread [18]. A figure showing this dynamics for a sample run on the synthesized data is shown in figure 3.4

It is also necessary at this point to take a look at the relevance of the volatility on the options price outcome. This enables us visualize the impact of the volatility on the dynamics of the system contrary to presumption made originally by Black and Scholes [16] that the volatility is meant to be remain constant throughout the life of the options contract. Figure 3.6 reveals a tangible effect of the variation in volatility in about the first 80 percent of the life of the options contract and points out where to focus when tracking the volatility.

3.4.2 Radial Basis Function Synthesized Data and Simulation

The proposed RBF network may be divided into two: the linear and non-linear parts. The linear part simply involves scaling the input vector with a set of predefined weights

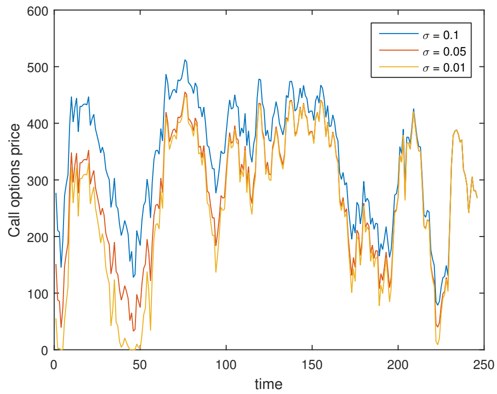


Figure 3.6: **Impact of volatility variations on the options price.** Volatility variations shows important effect on the options price especially in the initial 80% of the life of the contract contrary to Black-Scholes assumption of a constant volatility throughout the life of the option. This illustrates the necessity of accurate estimations of the volatility during the period of the life of the option

which can easily be determined based on ones discretion from historical data-set. The major focus is on the non-linear part, $\phi(.)$ defined by equation 3.21.

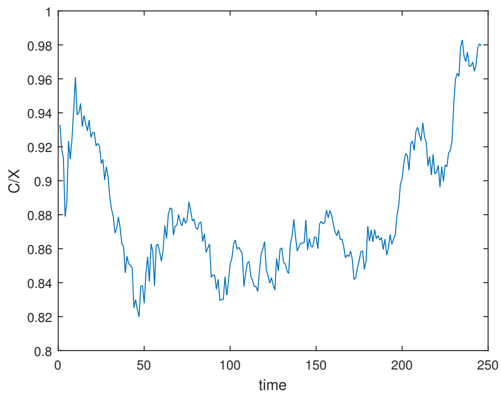


Figure 3.7: **Synthesized timeseries using the RBF network with steady-states.**

The RBF is trained by recursively using the Extended Kalman Filter. First, the covariance, Σ is set to be a unit identity matrix and is constant throughout the simulation. S, X and t_m are synthesized as described in section 3.4. k is fixed to 1 and the linear parts of the equation is neglected for now. Hence normalized options prices are synthesized using:

$$\frac{C}{X} = \sqrt{(\mathbf{x}-\mathbf{m})^{\mathbf{T}}\mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{m})} \tag{3.21}$$

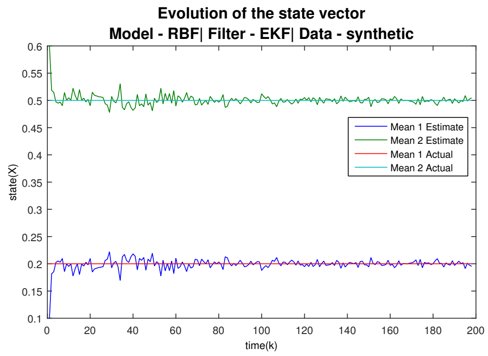


Figure 3.8: **EKF estimation of the steady-state using the RBF network.** Estimated states converges to the actual state values when the states are initialized close to the true states

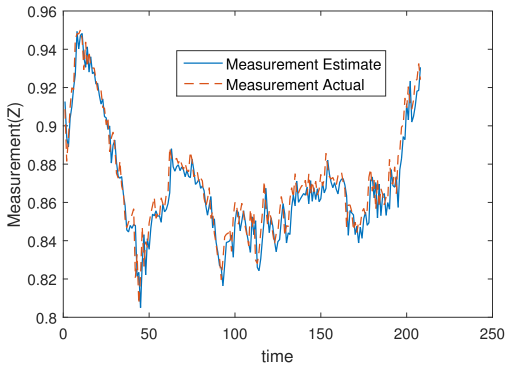


Figure 3.9: **One-step ahead measurement estimate tracking.**

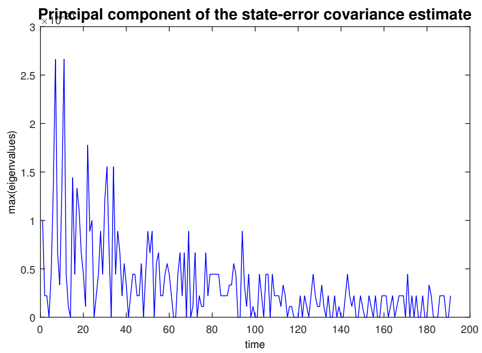


Figure 3.10: **Measure of uncertainty of the estimated state vector.** Gradual reduction in the state-error covariance reflected from the high accuracy of the state estimates

where

$$\mathbf{x} = [\frac{S}{X}\ t_m] \tag{3.22}$$

The results from sample runs showing the generated synthetic data, the progress of the sequential state estimation and the sequence of the first principal component of the state-error covariance.

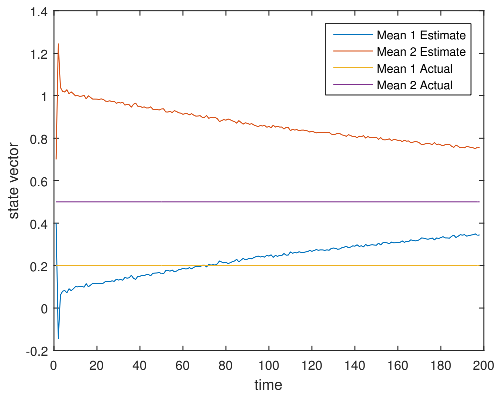


Figure 3.11: Steady-State estimation for EKF states initialized far from the true states. The state converging to values other than the real state vector set to $[0.2\ 0.5]^T$ when the initial state vector deviates too far from the true states. An illustration of the flexibility of the model to different state vectors as good solutions

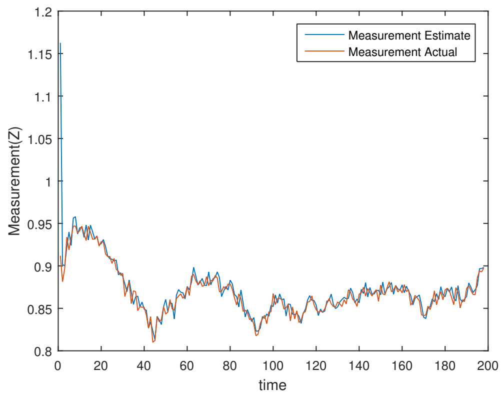


Figure 3.12: The corresponding measurement tracking when the state is initialized far away from the true initial state

Slight distortions in the initial states parameters reveal that the state tends to converge to some other arbitrary mean values (figure 3.11) but the observation converges to the true observation. Since the dynamics of the state-error covariance (figure 3.12) still remains as expected i.e. reduction in the spread of variance, it is right to assume that the model is flexible enough to values it deems are d best which are a function of the initial state.

3.5 Second Order Extended Kalman Filter

Bearing in my that the degree of non-linearity of the system may be quite high, implies there is a possibility of omitting useful information by terminating the Taylor series expansion of the filter after the first order derivative and neglecting higher order terms. This is usually avoided because the cost of computing the the second order partial derivative i.e. the Hessian Matrix may be extremely high.

Hence, we decided to extend the expansion of the Taylor series to the second order and study how much of a difference this makes.

In the light of Control Systems Theory, this is seen as extending the system to a type 2 controller system and the expected improvement compared to a type 1 controller system is mainly a shorter settling time, i.e. faster steady state stability.

For the Black-Scholes Model, the expressions for the Second order derivatives are readily available and makes it quite easy to implement the Second order Extended Kalman Filter(SO-EKF). In the case of the RBF model, we utilized the symbolic toolbox in MATLAB to feed in the model to the system and made use of MATLAB in-built functions to compute the derivatives.

Revisiting the state space equations, let us take a look at where to focus our attention when making adjustments to the filter algorithm. The state space equations remain the same as:

$$\theta(\mathbf{t+1}) = f\big(\theta(\mathbf{t})\big) + \mathbf{w}(\mathbf{t}) \tag{3.23}$$

$$z(t) = h\big(\theta(\mathbf{t+1}), \mathbf{U}(\mathbf{t})\big) + \mathbf{v}(\mathbf{t+1}) \tag{3.24}$$

where

$$\tag{3.25}$$

$$f\big(\theta(\mathbf{t})\big) = \mathbf{f}\big(\theta(\mathbf{t}|\mathbf{t})\big) + \mathbf{J}_\theta(\mathbf{t})(\theta(\mathbf{t}) - \theta(\mathbf{t}|\mathbf{t})) + \frac{\mathbf{1}}{\mathbf{2}}\sum_{\mathbf{i=1}}^{\mathbf{2}} \mathbf{m_i}(\theta(\mathbf{t}) - \theta(\mathbf{t}|\mathbf{t}))^{\mathbf{T}}\mathbf{H}_\theta(\mathbf{t})(\theta(\mathbf{t}) - \theta(\mathbf{t}|\mathbf{t})) + H.O.T \tag{3.26}$$

$$h(\theta(\mathbf{t+1})) = \mathbf{h}(\theta(\mathbf{t+1}|\mathbf{t})) + \mathbf{J_z}(\mathbf{t+1})(\theta(\mathbf{t+1}) - \theta(\mathbf{t+1}|\mathbf{t})) + \frac{1}{2}\sum_{i=1}^{2} m_i(t(t+1) - \theta(\mathbf{t+1}|\mathbf{t}))^{\mathbf{T}}\mathbf{H_z}(\mathbf{t+1})(\theta(\mathbf{t+1}) - \mathbf{x}(\mathbf{k+1}|\mathbf{k})) + H.O.T \tag{3.27}$$

and

$$\mathbf{J}_\theta(\mathbf{t}) = \frac{\partial f}{\partial \theta} \tag{3.28}$$

$$\mathbf{H}_\theta(\mathbf{t}) = \frac{\partial^2 h}{\partial \theta^2} \tag{3.29}$$

The implication this has on the general EKF algorithm is the addition of terms to the state-error covariance estimate, $P(t+1|t)$, and the measure-error covariance, $S(t)$, which is;

$$P(t+1|t) = J_\theta(t)P(t|t)J_\theta(t)^T + \frac{1}{2}\sum_{i=1}^{2}\sum_{j=1}^{2} m_i m_j^T tr\{H_\theta^i(t)P(t|t)H_\theta^j(t)P(t|t)\} + Q(t+1) \tag{3.30}$$

$$S(t) = J_z(t+1)P(t+1|t)J_z(t+1)^T + \frac{1}{2}\sum_{i=1}^{2}\sum_{j=1}^{2} m_i m_j^T tr\{H_z^i(t+1)P(t+1|t)H_z^j(t+1)P(t+1|t)\} + R(t+1) \tag{3.31}$$

Given that for both models implemented, we assumed prediction of the state to simply be a random walk, therefore the only changes made to the algorithm will be on $S(t)$ and $P(t+1|t)$. The introduction of the second order derivative, $H_z^i(t+1)$ necessitate the computation of the Hessian matrix of the models used. In the case of the RBF Network, this will be difficult to derive by hand but can be easily computed using the symbolic toolbox on MATLAB. On the other hand, the second derivative of the Black-Scholes are also readily available formulas used by financial analyst in strategizing methods of investment. The Hessian matrix is as follows:

$$H_z^1(t+1) = \begin{bmatrix} \frac{\partial^2 C}{\partial \sigma^2} & \frac{\partial^2 C}{\partial \sigma \partial \sigma} \\ \frac{\partial^2 C}{\partial \sigma \partial \sigma} & \frac{\partial^2 C}{\partial r^2} \end{bmatrix} \tag{3.32}$$

$$H_z^2(t+1) = \begin{bmatrix} \frac{\partial^2 P}{\partial \sigma^2} & \frac{\partial^2 P}{\partial \sigma \partial \sigma} \\ \frac{\partial^2 P}{\partial \sigma \partial \sigma} & \frac{\partial^2 P}{\partial r^2} \end{bmatrix} \tag{3.33}$$

where

$$\frac{\partial^2 C}{\partial \sigma^2} = \frac{\partial^2 P}{\partial \sigma^2} = \sqrt{t_m} S \frac{d_1 d_2}{\sigma} N'(d_1) \tag{3.34}$$

$$\frac{\partial^2 C}{\partial r^2} = -X t_m^2 \exp(-r t_m) \Big(t_m N(d_2) - \frac{d_2 \sqrt{t_m}}{\sigma} N'(d_2) \Big) \tag{3.35}$$

$$\frac{\partial^2 P}{\partial r^2} = X t_m^2 \exp(-r t_m) \Big(t_m N(-d_2) - \frac{d_2 \sqrt{t_m}}{\sigma} N'(-d_2) \Big) \tag{3.36}$$

$$\frac{\partial^2 C}{\partial \sigma \partial r} = \frac{\partial^2 P}{\partial \sigma \partial r} = -\frac{S d_1 \sqrt{t_m}}{\sigma} N'(d_1) \tag{3.37}$$

3.6 Degree of Non-linearity

[19] and [20] identified the various methods for the determination of the degree of non-linearity of a system. The curvature based degree of non-linearity, seen in [19], which is the ratio of the determinant of the hessian to the square of the determinant of the Jacobian, was chosen due to its close ties in line with this project. We can rightly observe that the major difference between the EKF and SO-EKF is the addition of terms that are majorly dependent on the Hessian matrix. Since this additions are made to covariance matrices, our interest lies in knowing the extent the addition of the Hessian dependent term increases the error covariance and can be found in the first principal component i.e. the maximum eigenvalue of the Hessian matrix.

A simulation is then carried out on steady-state data in order to visualize the areas where high level of non-linearity is expected in the system using both Black-Scholes and the RBF models.

Visualization of the dynamics of the states and error-covariance also indicate interesting results as seen in the shorter settling time of the SO-EKF compared to the EKF (see figure 3.15). This simply indicates that the SO-EKF captures the higher level of non-linearity in the system and responds faster to this than what is obtainable using the EKF algorithm. This also indicates that the SO-EKF has the ability to stabilize faster on events of distortions from the normal system dynamics

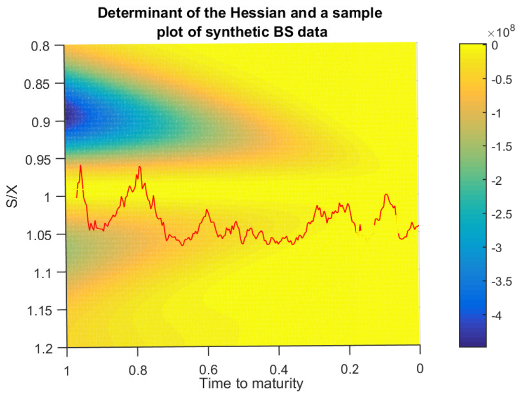


Figure 3.13: Measure of intensity of the non-linearity in the system on progression through the life of a call options contract

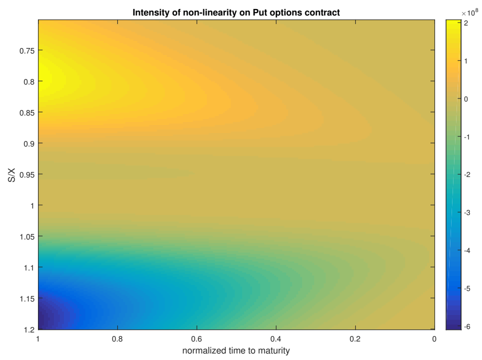
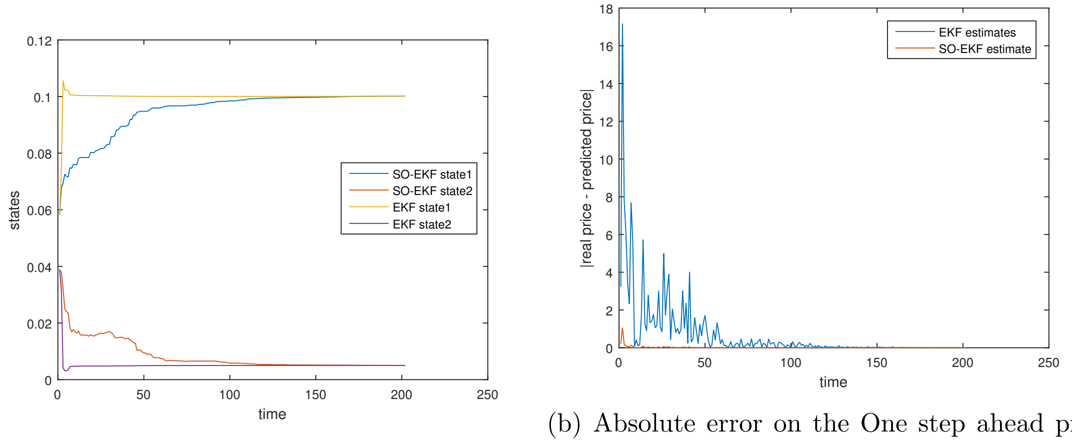


Figure 3.14: Measure of intensity of the non-linearity in the system on progression through the life of a put options contract

3.7 Tuning the Noise variance parameters

Determination of the best noise parameter for a particular data set is a difficult issue to contain [2] lays emphasis on tuning of noise parameters to help determine the noise variance that suites a dataset. Due to the nature of this research, we simply resorted to the same trial and error method of making a choice for the noise variance. Here, simply tuning the noise variance gives us an idea of the filters response to the system and our



(a) State measurement evolution

(b) Absolute error on the One step ahead prediction

Figure 3.15: **Comparison between the steady state responses of the EKF and SO-EKF.** Noise covariances Q and R were set to $10 \exp -12$

choice of the right noise was based on the noise variance that resulted in the fastest settling time in the steady state analysis (see figure 5.1)

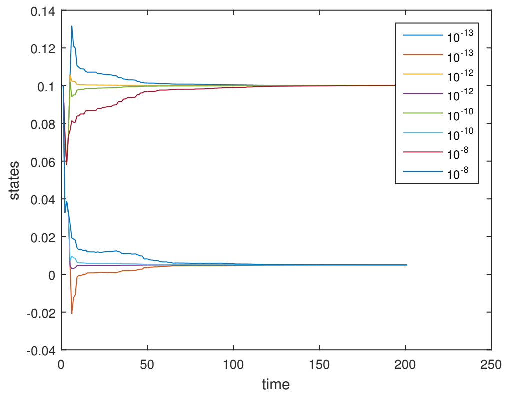


Figure 3.16: **Effect of tuning the noise variance of the system on the settling time of the filter**

3.8 Performance Measures

On completion of test on the synthesized data, results are computed based on the Root Mean Squared Error(RMSE) in the one step ahead prediction of the the prices of the two types of options contract. A comparison is made by matching each model to the two types of filters implemented. The separate tables portray information about the model filter combination that performs best in regions of low likelihood of non-linearity and high likelihood of non-linearity respectively.

Table 3.1: RMSE table for regions with a low likelihood of non-linearity

-	BS		RBF	
Option Type	EKF	SOKF	EKF	SOKF
Call	3.19exp-4	1.06exp-4	0.0032	0.2032
Put	5.9exp-5	6.35exp-5	0.0031	0.9926

Table 3.2: RMSE table for regions with a high likelihood of non-linearity

-	BS		RBF	
Option Type	EKF	SOKF	EKF	SOKF
Call	0.0022	5.72exp-4	0.0041	0.0066
Put	0.0022	0.0021	0.0042	0.0409

Chapter 4

Application to Real Data

4.1 Experiment for Parameter Estimation

Pricing options contract is quite difficult in the world today especially due to the amount of instability in the world markets in recent years. Market forces all around the world now contribute in one way or the other to the movement of the prices on the market. Most affected by this global market forces are top global brands that serve as the highest contributors to the stock market and the economy of big countries. An example of an index made to track the joint movement of this global brands is the FTSE100 in the United Kingdom. Hence, the reason for choosing it as the real data to experiment with. We envisage that the highest level of the dynamics of global market forces on financial market is well represented in this data.

Consequently, tracking the options contracts solely dependent on the FTSE100 index shall be the major focus.

Hutchinson did extensive work tracking the price of financial derivatives similar to the financial options. He, then, proposed a static RBF which produced quite good results. He made use of the SP 500 data of the United States to evaluate the performance of the RBF.

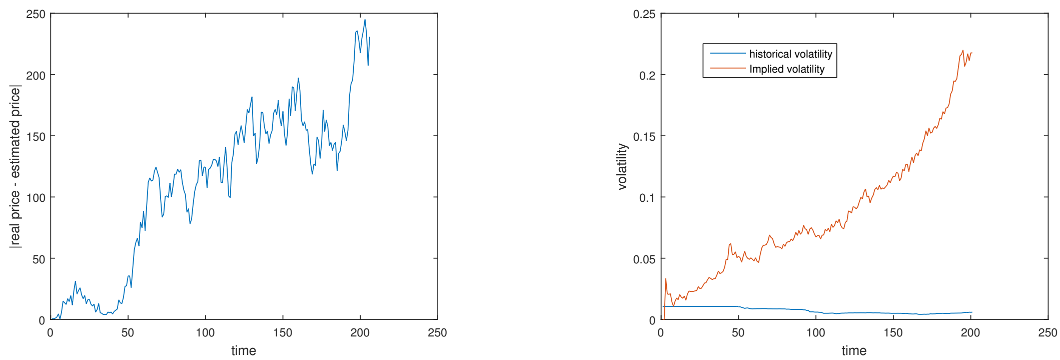
[2] envisaged that a sequential approach to the estimation of the options prices might be a better way to capture the non-linearity of the market forces. He addressed the problem using the EKF to estimate the volatility and the risk-free interest rate for the FTSE100(1994) data. [2] later used a Bayesian probability approach to the sequential estimation problem by adopting Monte Carlo method in the estimation of the linear weight of various neural networks.

Now, [2] and [4]'s work is taken even a step further by sequentially estimating of the volatility and risk-free interest rate. Also, the linear weights of the RBF are fixed to reasonable values and then directly tracking the non-linearity in the system by assigning the mean vectors in the non-linear aspect of the non-parametric(RBF) model as the state vectors of the Kalman filter. All this is implemented on more recent data being the FTSE100 annual data (2012 - 2014) for both Call and Put options contracts.

Now a comparison of the accuracy of the estimation's capabilities of various approached to estimation of options prices is made in order to evaluate the performance. This models include:

4.1.1 The Regular Black-Scholes(BS-R)

This represents the usual Black-Scholes Model described in section 2.5.1. The unknown parameters remain the risk-free interest rates and the volatility. The risk- free interest rate may be estimated using the interest rates quoted for the UK Goverment Bonds called the UK Giltz. Daily estimates of the UK Giltz was sourced for from Bloomberg Terminal. An estimate of the volatility also can be derived from the daily historical prices using a 50-day timing window(see 4.1). This method of estimation of the options prices is useful for financial data analyst to get a rough and quick estimate of the option's price.



(a) Absolute error on one step ahead estimation of the options price

(b) Comparison between the Implied volatility and the historical volatility

Figure 4.1: Results from the use of historical volatility estimated from a moving window on estimation of the options price

4.1.2 Black-Scholes with Extended Kalman Filter(BS-EKF)

The model in this method is the usual Black-Scholes Model described in section 2.5.1. The unknown parameters remain the risk-free interest rate(r) and the volatility(σ). This parameters are sequentially estimated using the EKF algorithm described in section 3.1. Hence,the state vector, $\theta(t|t)$ is set to $[log(\sigma)\ log(r)]^T$ rather than $[\sigma\ r]^T$ in order to avoid having negative as estimates of the state vector which is impossible in real life. At time, $t = 1$, the following initializations take place for the EKF algorithm to work:

$$\begin{aligned} \text{State Estimate} \quad & \theta(1|1) = [log(0.1)\ log(0.1)]^T \\ \text{State Error Covariance} \quad & P(1|1) = 0.1I \\ \text{State error noise covariance} \quad & Q = 0.001I \\ \text{Measurement error covariance} \quad & R = 0.1 \end{aligned}$$

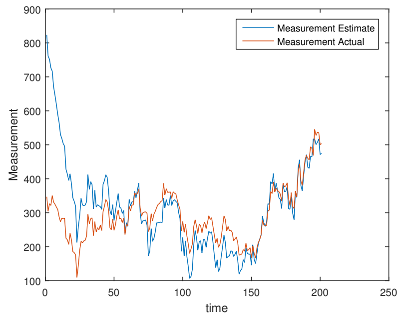
where I is a 2 X 2 identity matrix

4.1.3 Black-Scholes with Second Order Extended Kalman Filter(BS-SOEKF)

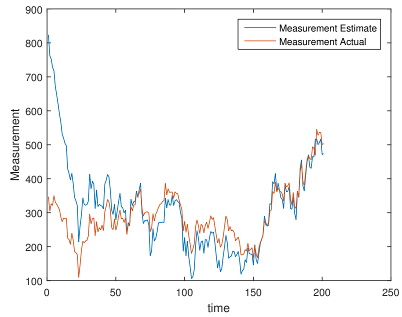
The model in this method is the usual Black-Scholes Model described in section 2.5.1. The unknown parameters remain the risk-free interest rate(r) and the volatility(σ). This parameters are sequentially estimated using the SO-EKF algorithm described in section 3.5. Hence,the state vector, $\theta(t|t)$ is set to $[log(\sigma)\ log(r)]^T$ rather than $[\sigma\ r]^T$ in order to avoid having negative as estimates of the state vector which is impossible in real life. At time, $t = 1$, the following initializations take place for the SO-EKF algorithm to work:

$$\begin{aligned} \text{State Estimate} \quad & \theta(1|1) = [log(0.1)\ log(0.1)]^T \\ \text{State Error Covariance} \quad & P(1|1) = 0.1I \\ \text{State error noise covariance} \quad & Q = 0.001I \\ \text{Measurement error covariance} \quad & R = 0.1 \end{aligned}$$

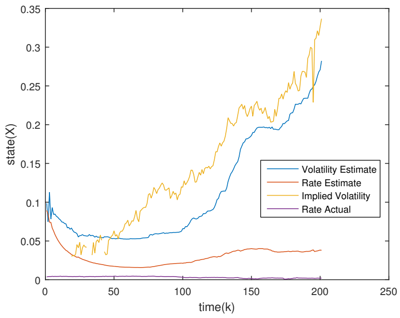
where I is a 2 X 2 identity matrix



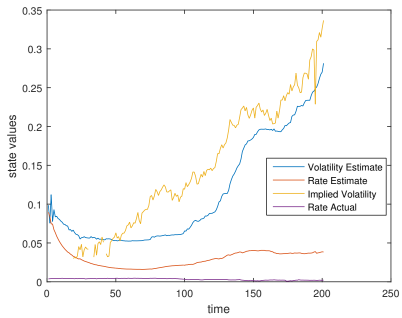
(a) Comparing real and estimated price estimation for the EKF



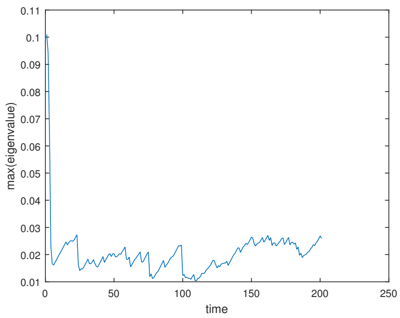
(b) Comparing real and estimated price estimation for the SO-EKF



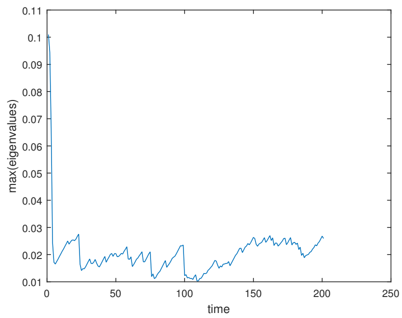
(c) Evolution of the states estimates compared to the real states for EKF



(d) Evolution of the states estimates compared to the real states for SO-EKF



(e) Measure of the uncertainty of the state estimates for the EKF



(f) Measure of the uncertainty of the state estimates for the SO-EKF

Figure 4.2: **Comparison between the EKF and SO-EKF with Black-Scholes model on the real data.** (a) and (b) one-step ahead tracking of the options price over the life-time of the contract, (c) and (d) shows corresponding tracking of the volatility and the interest rates, (e) and (f) shows the measure of uncertainty of each the estimations of the states

4.2 Experiments for Model Estimation

4.2.1 Radial Basis Function with Extended Kalman Filter (RBF-EKF)

The model adopted for sequential estimation using the EKF remains as described in equation 3.18. The variables therein are set as follows:

$$k = 4$$
$$\lambda_i = 0.1i$$
$$w_1 = [-0.1 \;\; -0.1]^T$$
$$w_0 = -0.2$$
$$\Sigma = I$$

where I is a 2 X 2 identity matrix.

The choice of k is very important as it gives us an idea of the number of clusters we envisage in the data set. It is set to 4 based on cues from [4] and the reasons for this is also detailed in section 2.5.2

The Linear weights, w^T, w_0 and λ_i are set as the average of the best linear weights that fit into the model for the training dataset. I made use of a MATLAB in-built function called 'fminunc'. It is used to find unconstrained variables for a function with the least error. fminunc arrives at this values by combining Gaussian mixture models with expectation maximization algorithms. The unknown parameter in this RBF model is the vector m. This parameter is sequentially estimated using the EKF algorithm. At time, $t = 1$, the following initializations take place for the EKF algorithm to work:

$$\begin{array}{rr} \text{State Estimate} & \theta(1|1) = [1 \; 1]^T \\ \text{State Error Covariance} & P(1|1) = I \\ \text{State error noise covariance} & Q = 10^{-7}I \\ \text{Measurement error covariance} & R = 10^{-8} \end{array}$$

where I is a 2 X 2 identity matrix

4.2.2 Radial Basis Function with Second Order Extended Kalman Filter (RBF-SOEKF)

The model adopted for this method is described in 3.5. The variables therein are set as follows:

$$
\begin{aligned}
k &= 4 \\
\lambda_i &= 0.1i \\
w_1 &= [-0.1 \ \ -0.1]^T \\
w_0 &= -0.2 \\
\Sigma &= I
\end{aligned}
$$

where I is a 2 X 2 identity matrix.

The choice of k is very important and is set to 4 based on cues from **??** and the reasons for this is also detailed in section **??**

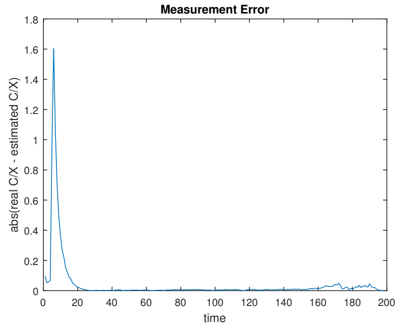
The Linear weights, w^T, w_0 and λ are set as the average of the best linear weights that fit into the model for the training dataset. I made use of a MATLAB in-built function called 'fminunc'. It is used to find unconstrained variables for a function with the least error. fminunc arrives at this values by combining Gaussian mixture models with expectation maximization algorithms. The unknown parameter in this RBF model is the vector m. This parameter is sequentially estimated using the EKF algorithm. At time, $t = 1$, the following initializations take place for the EKF algorithm to work:

$$
\begin{aligned}
\text{State Estimate} &\qquad \theta(2|1) = [1 \ 1]^T \\
\text{State Error Covariance} &\qquad P(2|1) = I \\
\text{State error noise covariance} &\qquad Q = 10^{-7}I \\
\text{Measurement error covariance} &\qquad R = 10^{-8}
\end{aligned}
$$

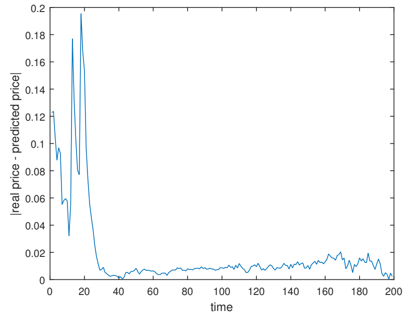
where I is a 2 X 2 identity matrix

4.3 Evaluation of Results

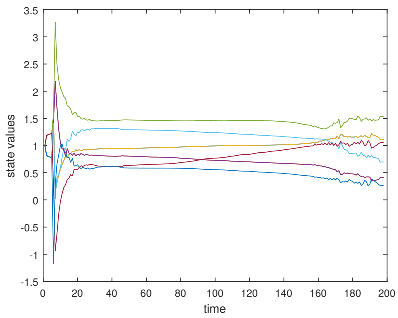
The Black-Scholes Model has shown remarkable performance using the sequential estimation algorithms. This validates the Kalman filter's ability to estimate, with high accuracy, the implied volatility and true risk-free interest rate of the financial market.



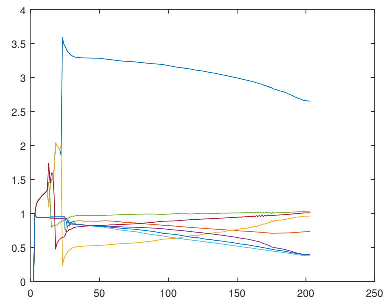
(a) Absolute error on one-step ahead estimated price estimation for the SO-EKF



(b) Absolute error on one-step ahead estimated price estimation for the EKF



(c) Evolution of the states estimates for SO-EKF



(d) Evolution of the states estimates for EKF

Figure 4.3: **Visualization of comparison between the EKF and SO-EKF using the RBF Network.**

Table 4.1: RMSE on one-step ahead predictions for regions with a low likelihood of non-linearity for Call options contract

-	-	BS		RBF	
Year	Strike(X)	EKF	SOKF	EKF	SOKF
2014	6700	0.0050	0.0050	0.0124	0.2638
2014	6800	0.0049	0.0052	0.0197	0.2406
2013	6000	0.0029	0.0030	0.0226	0.1601
2013	6100	0.0035	0.0035	0.0209	0.1791
2012	5500	0.0055	0.0056	0.0199	0.1420
2012	5600	0.0051	0.0051	0.0133	0.1823

Advantages in the use of a sequential estimator that tracks the error in the estimated state parameters over the regular method of estimating the volatility and interest rates from historical data is revealed at instances when unusual/unexpected dynamics occur in the system, which could not be contained by the Black-Scholes formula. Here, the Kalman filter is quick to discover unusual dynamics in the system and rightly adjust to the situation in the market faster than any historical based estimation of the volatility or interest rates can do.

Table 4.2: RMSE on one-step ahead predictions for regions with a high likelihood of non-linearity for call options contract

-	-	BS		RBF	
Year	Strike(X)	EKF	SOKF	EKF	SOKF
2014	6700	0.0112	0.0113	0.0413	0.0238
2014	6800	0.0099	0.0099	0.0081	0.1700
2013	6000	0.0170	0.0172	0.6230	0.0113
2013	6100	0.0162	0.0164	0.6059	0.0183
2012	5500	0.0134	0.0136	0.4505	0.0038
2012	5600	0.0121	0.0123	0.0296	0.3144

Table 4.3: RMSE on one-step ahead predictions for regions with a low likelihood of non-linearity for Put options contract

-	-	BS		RBF	
Year	Strike(X)	EKF	SOKF	EKF	SOKF
2014	6700	0.0047	0.0056	0.0118	0.0104
2014	6800	0.0062	0.0082	0.0154	0.0075
2013	6000	0.0028	0.0347	0.6230	0.0253
2013	6100	0.0026	0.0398	0.0186	0.0171
2012	5500	0.0622	0.0622	0.0157	0.0128
2012	5600	0.0682	0.0132	0.0296	0.0131

Table 4.4: RMSE on one-step ahead predictions for regions with a high likelihood of non-linearity for put options contract

-	-	BS		RBF	
Year	Strike(X)	EKF	SOKF	EKF	SOKF
2014	6700	0.0121	0.0130	6.6801	0.0126
2014	6800	0.0141	0.0144	0.1862	0.2582
2013	6000	0.0022	0.0070	0.0068	7.9038
2013	6100	0.0023	0.0087	0.0090	2.8586
2012	5500	0.0133	0.0124	0.0084	16.9127
2012	5600	0.0165	0.0064	0.0058	2.3015

Another test of the accuracy of the results can be shown from the convergence of the states for each years data. It is expected that the volatility and interest rates are independent of the strike price, hence, this values must remain the same irrespective of the strike price chosen. The simulations using the the Black-Scholes Model clearly illustrates this.

Introduction of the Second Order Extended Kalman Filter (SOEKF) to the Black-Scholes model made little or no difference in most cases as regards the accuracy of the estimation. Since the reason for extending the EKF to SOEKF was to enable the filter capture details of non-linearity that is beyond the first order and the results reveal little or no difference. It is right to infer at this point that the Black-Scholes model on its own has been able to capture most of the non-linearity in the system, Hence, further knowledge of the gradient(first order derivative) alone will be sufficient to have accurate estimations of the states.

In the case of the model estimation of the RBF, it is observed that slight deviations from the true best states results in the EKF estimating parameters that are different from the true parameters. This gives a clear indication that the RBF is quite dependent on the initial parameters as seen from the steady state response diagrams in section **??**. The fact that the resulting estimates are quite accurate indicates to us the flexibility of adopting model estimation. A major advantage to this can be seen when the system deviates from the expected dynamics, the RBF does not have to take time trying to settle back into accurate estimation when the system stabilizes. The RBF simply settles into the closest model parameters that results in good estimation of the option prices.

This is evident in the estimated state values not having any major jump or difference in values from immediate previous values. This ability of the model to be flexible by adopting the closest best model is a major advantage of the model estimation method.

Chapter 5

Risk Analysis and Project Management

5.1 Project Management

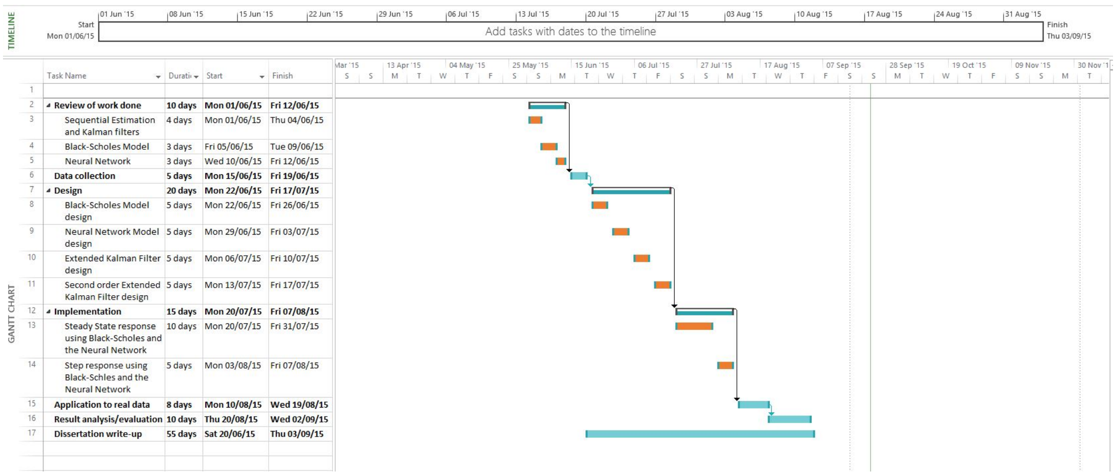


Figure 5.1: Gantt Chart showing the Project management timeline

A total of about fifteen weeks was spent to fully expend the entire project. About 50% of the entire time and effort put into this work was spent of the design and implementation of the EKF and the SO-EKF algorithms. The design required ensuring the algorithm was correctly setup on the MATLAB software for the purpose of simulation of the system in view. The implementation process simply required getting the models coded into the simulator rightly and merging these models correctly with the filter algorithms. The project kicked off with a review of work done relating generally to sequential estimation of non-linear systems especially with the use of Kalaman filters. This period of review lasted for about two weeks. The design and implementation was the next in line with major focus on synthetic data to ensure that the parameters in my correct were rightly tuned. Consequently, experimentation on real data was next. This required first gathering and processing the data to rightly fit into the algorithm and then the experimentation procedures were carried out for about three week. The respective results from the experiments were gathered and compilation of the report simultaneously took place.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this research, the Extended Kalman Filter(EKF) approach as a dynamic learning technique has been used for the sequential estimation of Financial options prices. The EKF was capable of accurately estimate the parameters (volatility and interest rates) of the Black-Scholes(BS) model sequentially under conditions whereby the initial parameters of the filter such as the initial prior state vector, initial prior state-error covariance and the noise parameters are properly set based on historical knowledge of the dataset. We also explored the EKF approach to sequential estimation of the a non-parametric model(a Radial Basis Function(RBF) Network proposed by Hutchinson et al **??**).

A high level of non-linearity of the system was envisaged and the EKF was adjusted to capture this by implementing a Second Order Extended Kalman Filter to estimate parameter of the Black-Scholes model as well as the RBF network.

The key idea here is to study how sequentially propagating the states and error-covariance can lead to accurate estimation of a stochastic and severely non-linear system output such as financial options prices.

Emphasis was placed on the idea of a dynamic model with the ability to adapt to changes in the dynamics of the financial market with time. This dynamic model may be interpreted as one that sequentially adapts to the stochasticity inherent in parameters influencing the price of options with respect to the price of the underlying asset(S) and the time to maturity.

The little or no difference between the results of the EKF and SOKF using the BS model indicates to us that the BS model captures most of the non-linearity of the system without a need for exploring the information in the second order derivative of the model, hence not much difference is made when extension to the second order is implemented. On the other hand, the large difference in the result of the EKF and SOKF using the RBF model indicates the inadequacy of the RBF model in capturing high levels of non-linearity in the system.

Experiments with the real annual financial data(FTSE100 [2012-2014]) reveals that the RBF-SO-EKF outperforms the BS model, within the region of expected high non-linearity, for some cases. As observed, high non-linearity occurs mostly at the initial stage of the options life and gives us a fair conclusion that the RBF converges to the true options price faster that the BS model. This accuracy of the RBF is mainly reflected in the performance of the Second order EKF and can be attributed to the fact that this filter captures more of the non-linearity in the system.

Regions of expected low non-linearity(later part of the life of the option) reflect better performance measures using the BS-EKF especially for the Call options prices.

6.2 Future Work

Further directions for likely improvement include design of algorithm to sequentially estimate the covariance used in the RBF and methods of matching the network structure, especially the number of means and covariances to be used, to different datasets.

Other methods of determining the optimal noise parameters can also be explored rather than tuning the noise parameter to fit the data as a choice of the optimal noise.

Due to the positive prospects the RBF has shown on this particular dataset, other forms of RBF techniques such as the Gaussian RBF may be explored. Notwithstanding, the use of other forms of Neural Networks such as the Multi-Layer perdceptron and Deep Learning are also likely to give good results if properly sequentially estimated.

Appendix A

Appendix Title Here

A.1 Proof of the derivatives of the Black-Scholes Model

Given the Black-Scholes equations as:

$$C = SN(d_1) - X \exp(-rt_m)N(d_2) \tag{A.1}$$

$$d_1 = \frac{ln(S/X) + (r + \frac{\sigma^2}{2})\sqrt{t_m}}{\sigma\sqrt{t_m}} \tag{A.2}$$

$$d_2 = d_1 - \sigma\sqrt{t_m} \tag{A.3}$$

The First order derivative with respect to σ and r

$$\begin{aligned}\frac{\partial C}{\partial \sigma} &= SN'(d_1)\frac{\partial d_1}{\partial \sigma} - X \exp(-rt_m)N'(d_2)\frac{\partial(d_2)}{\partial \sigma} \\ &= SN'(d_1)\frac{\partial(d_1)}{\partial \sigma} - X \exp(-rt_m)N'(d_2) \\ &\Big(\frac{\partial(d_1)}{\partial \sigma} - \sqrt{t_m}\Big) \\ &= \Big(SN'(d_1) - X \exp(-rt_m)N'(d_2)\Big)\frac{\partial(d_1)}{\partial \sigma} + \\ &\sqrt{t_m}X \exp(-rt_m)N(d_2)\end{aligned} \tag{A.4}$$

But,

$$SN'(d_1) = X \exp(-rt_m)N'(d_2) \tag{A.5}$$

This is proven by solving exp $\Big(ln(\frac{N(d_1)}{N(d_2)})\Big)$ Therefore, substituting for (5) in (4)

$$\boxed{\frac{\partial C}{\partial \sigma} = \sqrt{t_m} S N'(d_1)} \tag{A.6}$$

This is normally referred to as vega

$$\begin{aligned} \frac{\partial C}{\partial r} &= SN(d_1)\frac{\partial(d_1)}{\partial r} - X\exp(-rt_m) \\ &\big(N(d_2)\frac{\partial(d_2)}{\partial r} - t_m N(d_2)\big) \end{aligned} \tag{A.7}$$

Applying the same approach of factorizing and substituting for $SN'(d_1)$ from (5) into (7) as shown above, we have:

$$\boxed{\frac{\partial C}{\partial r} = t_m X \exp(-rt_m) N(d_2)} \tag{A.8}$$

This is normally referred to as rho(ρ)

Now, the second order derivative with respect to σ and r are as follows

$$\frac{\partial^2 C}{\partial \sigma^2} = \sqrt{t_m} S \frac{\partial(N'(d_1))}{\partial(d_1)} . \frac{\partial(d_1)}{\partial \sigma} \tag{A.9}$$

Now,

$$\frac{\partial(N'(d_1))}{\partial(d_1)} = -d_1 N'(d_1) \tag{A.10}$$

$$\frac{\partial(d_1)}{\partial \sigma} = \frac{\partial(d_2)}{\partial \sigma} + \sqrt{t_m} \tag{A.11}$$

$$= -\frac{d_1}{\sigma} + \sqrt{t_m} \tag{A.12}$$

$$= -\frac{d_2}{\sigma} \tag{A.13}$$

Now substituting from (10) and (13) into (9), we have:

$$\frac{\partial^2 C}{\partial \sigma^2} = \sqrt{t_m} S\Big(- d_1 N'(d_1)\big(- \frac{d_2}{\sigma})\Big) \tag{A.14}$$

$$\boxed{\frac{\partial^2 C}{\partial \sigma^2} = \sqrt{t_m} S \frac{d_1 d_2}{\sigma} N'(d_1)} \tag{A.15}$$

$$\frac{\partial^2 C}{\partial r^2} = \frac{\partial\big(Xt_m \exp(-rt_m)N(d_2)\big)}{\partial r}$$

Applying quotient rule to this:

$$\begin{aligned}\frac{\partial^2 C}{\partial r^2} &= \frac{\partial\big(Xt_m \exp(-rt_m)\big)}{\partial r}N(d_2)+ \\ &\big(Xt_m \exp(-rt_m)\big)N'(d_2)\frac{\partial(d_2)}{\partial r}\end{aligned} \tag{A.16}$$

$$\frac{\partial\big(Xt_m \exp(-rt_m)\big)}{\partial r} = \big(-Xt_m^2 \exp(-rt_m)\big) \tag{A.17}$$

$$\frac{\partial(d_2)}{\partial r} = \frac{1}{\sigma} \tag{A.18}$$

Hence, substituting back from (17) and (18) into the (16) equation, we get:

$$\boxed{\frac{\partial^2 C}{\partial r^2} = -Xt_m^2 \exp(-rt_m)\Big(t_m N(d_2) - \frac{d_2\sqrt{t_m}}{\sigma}N'(d_2)\Big)} \tag{A.19}$$

$$\frac{\partial^2 C}{\partial \sigma \partial r} = \frac{\partial\big(Ssqrtt_m N'(d_1)}{\partial r}$$

$$\frac{\partial^2 C}{\partial \sigma \partial r} = S\sqrt{t_m}\frac{\partial(N'(d_1))}{\partial(d_1)}.\frac{\partial(d_1)}{\partial r} \tag{A.20}$$

$$\frac{\partial(d_1)}{\partial r} = \frac{1}{\sigma} \tag{A.21}$$

Substituting from (10) and (21) into (20), we have:

$$\boxed{\frac{\partial^2 C}{\partial \sigma \partial r} = -\frac{Sd_1\sqrt{t_m}}{\sigma}N'(d_1)} \tag{A.22}$$

Appendix B

Appendix Title Here

The objective of the Kalman filter is to find the unbiased minimum variance estimator of the state at time k+1. The state and measurement equations are defined in (1) and (2) below. The aim is to minimize a loss or error function:

$$f(e_k) = f(x_k - \hat{x}_k)$$

This has to be a positive and increasing value, hence the need to find the square:

$$f(e_k) = (x_k - \hat{x}_k)^2$$

Since, the prediction is made over a period of time with different values, a better error function would be the Mean Square Error (MSE):

$$e(t) = E[f(e_k)] = E(e_k^2)$$

The expression above is the trace of the covariance of e_k. This is solved for in the set of equations below.

B.1 Extended Kalman Filter

$$x(k+1) = f(x(k)) + w(k) \tag{B.1}$$

$$z(k+1) = h(x(k+1)) + v(k+1) \tag{B.2}$$

Now setting the covariances expressions

$$Q(k) = E[w(k).w(k)^T] \tag{B.3}$$

$$R(k) = E[v(k).v(k)^T] \tag{B.4}$$

$$P(k) = E[e(k).e(k)^T] \tag{B.5}$$

$$e(k) = x(k) - \hat{x}(k) \tag{B.6}$$

where e(k) is error in state estimation and **hat** denotes estimate

We wish to perform a state estimate update that is a function of the previous state estimate and the error in the predicted measurement as:

$$x(k+1|k+1) = x(k+1|k) + K(k)(Z(k) - h(x(k+1|k))) \tag{B.7}$$

Now expanding the $f(x(k))$ in Taylor series about $x(k|k)$ implies:

$$f(x(k)) = f(x(k|k)) + J_x(k)(x(k) - x(k|k)) + H.O.T \tag{B.8}$$

$$f(x(k)) = f(x(k|k)) + J_x(k)e(k) + H.O.T \tag{B.9}$$

$$x(k+1|k) = f(x(k|k)) + J_x(k)e(k) \tag{B.10}$$

$$E[f(x(k))|Z(k)] = f(x(k|k)) + J_x(k)E[e(k)] \tag{B.11}$$

$$E[e(k)] = 0 \tag{B.12}$$

$$x(k+1|k) \approx f(x(k|k)) \tag{B.13}$$

$$J_x(k) = \frac{\delta f}{\delta x} \tag{B.14}$$

where H.O.T are the Higher Order Terms. We choose to ignore them in this first order filter, $J_x(k)$ is the Jacobian

Now we need to predict the covariance to carry along the main features of the probability density of the distribution being the mean and variance.

The estimator error is:

$$e(k+1) = x(k) - x(k+1|k) + k(K)(z(k) - h(x(k+1|k))) \tag{B.15}$$

$$\tag{B.16}$$

From (2), now taking Taylor series expansion of $h(x(k+1))$ about $x(k+1|k)$, we get:

$$\hat{z}(k+1) = h(x(k+1)) = h(x(k+1|k)) + J_z(k+1)(x(k+1) - x(k+1|k)) + H.O.T \tag{B.17}$$

$$z(k+1) \approx h(x(k+1|k)) + J_z(k+1)(x(k+1) - x(k+1|k)) + v(k+1) \tag{B.18}$$

Substituting from (1) and (10)

$$\hat{e}(k+1) = x(k+1) - x(k+1|k) = J_x(k)e(k) + w(k) \tag{B.19}$$

$$e(k+1) = \hat{e}(k+1) + K(k)(z(k+1) - \hat{z}(k+1)) \tag{B.20}$$

$$e(k+1) \approx J_x(k)e(k) + w(k) - K(k)(J_z(k+1)(J_x(k)e(k) + w(k)) + v(k+1)) \tag{B.21}$$

$$\approx (I - K(k)J_z(k+1))J_x(k)e(k) + (I - K(k)J_z(k+1))w(k) - K(k)v \tag{B.22}$$

Hence the predicted covariance is:

$$P(k+1|k) = E[\hat{e}(k+1).\hat{e}(k+1)^T] \tag{B.23}$$

$$P(k+1|k) = J_x(k)P(k|k)J_x(k)^T + Q(k) \tag{B.24}$$

$$\tag{B.25}$$

In order to obtain the Kalman gain, we differentiate the trace of the predicted error estimate covariance wrt $K(k)$ and equate to 0. With this we achieve the minimum variance estimator. Then, an expression for $K(k)$ can be determined as shown below:

$$P(k+1|k+1) = E[e(k+1).e(k+1)^T] \tag{B.26}$$

$$P(k+1|k+1) = (I - K(k)J_z(k+1))P(k+1|k)(IK(k)J_z(k+1))^T + K(k)R(k+1)K(k)^T \tag{B.27}$$

$$\frac{\delta tr(P(k+1|k))}{\delta K(k)} = -(J_z(k+1)P(k+1|k))^T - P(k+1|k)J_z(k+1)^T + \\ 2K(k)J_z(k+1)P(k+1|k)J_z(k+1) + 2K(k)R(k+1) = 0 \tag{B.28}$$

$$K(k) = P(k+1|k)J_z(k+1)^T(J_z(k+1)P(k+1|k)J_z(k+1)^T + R(k+1))^{-1} \tag{B.29}$$

$$S(k) = J_z(k+1)P(k+1|k)J_z(k+1)^T + R(k+1) \tag{B.30}$$

substitute back for $K(k)$ in $P(k+1|k+1)$ in (27)

$$P(k+1|k+1) = (I - K(k)J_z(k+1))P(k+1|k) \tag{B.31}$$

B.2 Second Order Kalman Filter

This filter is similar to the extended kalman filter derivation but for the Taylor series expansion of $f(x(k))$ and $h(x(k+1))$ from (1) and (2) that extends a step into the H.O.T that was previously ignored. Now, equations (1) to (7) remain exactly the same here but (8) and (17) is changed to:

$$f(x(k)) = f(x(k|k)) + J_x(k)(x(k) - x(k|k)) + \frac{1}{2}\sum_{i=1}^{2} m_i(x(k) - x(k|k))^T H_x(k)(x(k) - x(k|k)) \tag{B.32}$$

$$+ H.O.T \tag{B.33}$$

$$h(x(k+1)) = h(x(k+1|k)) + J_z(k+1)(x(k+1) - x(k+1|k)) + \tag{B.34}$$

$$\frac{1}{2}\sum_{i=1}^{2} m_i(x(k+1) - x(k+1|k))^T H_z(k+1)(x(k+1) - x(k+1|k)) + H.O.T \tag{B.35}$$

Hence the estimated state update remains the same as (7) provided (33) is used to sub for $h(x(k+1))$. Now, the estimated state predictions conditioned on z(k) and estimate of z(k+1) is:

$$x(k+1|k) = f(x(x(k|k))) + \frac{1}{2}\sum_{i=1}^{2} m_i tr\{H_x(k)P(k|k)\} \tag{B.36}$$

$$\hat{z}(k+1) = h(x(k+1|k)) + \sum_{i=1}^{2} m_i tr\{H_x(k)P(k|k)\} \tag{B.37}$$

The state prediction error $\hat{e}(k+1) = x(k+1) - x(k+1|k)$ is:

$$\hat{e}(k+1) = J_x(k) + \frac{1}{2}\sum_{i=1}^{2} m_i\{x(k|k)^T H_z(k)x(k|k) - tr\{H_x(k)P(k|k)\} + v(k+1) \tag{B.38}$$

$$H_x(k) = \frac{\delta^2 h}{\delta x^2} \tag{B.39}$$

where $H_x(k)$ is the Hessian function now the state prediction error covariance is determined as:

$$P(k+1|k) = J_x(k)P(k|k)J_x(k)^T + \frac{1}{2}\sum_{i=1}^{2}\sum_{j=1}^{2} m_i m_j^T tr\{H_x^i(k)P(k|k)H_x^j(k)P(k|k)\} + Q(k+1) \tag{B.40}$$

Now the S(k) term in the Kalman gain expression is replaced by:

$$S(k) = J_z(k+1)P(k+1|k)J_z(k+1)^T \tag{B.41}$$

$$+ \frac{1}{2}\sum_{i=1}^{2}\sum_{j=1}^{2} m_i m_j^T tr\{H_z^i(k+1)P(k+1|k)H_z^j(k+1)P(k+1|k)\} \tag{B.42}$$

References

[1] E. Lehmann. *Theory of point estimation.* Wadsworth, Pacific Grove, 2nd edition edition, 1991.

[2] Mahesan Niranjan. Sequential tracking in pricing financial options using model based and neural network approaches. *Advances in Neural Information Processing Systems 9*, pages 960–966, 1997.

[3] Nando de Freitas, Mahesan Niranjan, Andrew Gee, and Arnaud Doucet. Sequential monte carlo methods for optimisation of neural network models, 1998.

[4] J M Hutchinson, a W Lo, and T Poggio. A nonparametric approach to pricing and hedging derivative securities via lerning networks. *The Journal of Finance*, 49(3): 771–818, 1994.

[5] Nando de Freitas, Mahesan Niranjan, and Andrew Gee. Hierarchical bayesian-kalman models for regularisation and ard in sequential learning. Technical report, DEPARTMENT OF ENGINEERING, CAMBRIDGE UNIVERSITY, 1998.

[6] Nando de Freitas, Mahesan Niranjan, Andrew Gee, Jfg De Freitas, M Niranjan, and Ah Gee. The em algorithm and neural networks for nonlinear state space estimation. Technical report, 1998.

[7] Y. Bar-Shalom and X.R. Li. *Estimation and Tracking: Principles, Techniques, and Software.* A Wiley-Interscience publication. The Author, 1998. ISBN 9780964831216.

[8] F. Girosi, M. Jones, and T. Poggio. Priors stabilizers and basis functions: From regularization to radial, tensor and additive splines. *MIT AI Lab Memo 1430*, 1993.

[9] H. Robbins and S. Monro. A stochastic approximation model. 1951.

[10] *Theory and Practice of Recursive Identification.* MIT Press, Cambridge, Massachusetts, 1986.

[11] Bernard Widrow, Michael Lehr, Franqoise Beaufays, Eric Wan, and Michel Bilello. Learning algorithms for adaptive signal processing and control.

[12] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian\nBayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002. ISSN 1053587X. doi: 10.1109/78.978374.

[13] Peter S. Maybeck. Stochastic models estimation and control. vol. 2, 1982. URL http://opac.inria.fr/record=b1089399.

[14] R. E. Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 1960.

[15] John C. Hull. *Options, futures, and other derivatives*. Pearson, Prentice Hall, Upper Saddle River (N.J.), 2006. ISBN 0-13-149908-4. URL http://opac.inria.fr/record=b1120280.

[16] Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3):pp. 637–654, 1973. ISSN 00223808. URL http://www.jstor.org/stable/1831029.

[17] Torben G. Andersen, Tim Bollerslev, Francis X. Diebold, and Paul Labys. Modeling and forecasting realized volatility. Working Paper 8160, National Bureau of Economic Research, March 2001. URL http://www.nber.org/papers/w8160.

[18]

[19] Douglas M. Bates and Donald G. Watts. *Nonlinear regression analysis and its applications*. Wiley series in probability and mathematical statistics. Wiley, New York [u.a.], 1988. ISBN 0471816434.

[20] Ruixin Niu, Pramod K. Varshney, Mark G. Alford, Adnan Bubalo, Eric K. Jones, and Maria Scalzo. Curvature nonlinearity measure and filter divergence detector for nonlinear tracking problems. In *FUSION*, pages 1–8. IEEE, 2008. ISBN 978-3-00-024883-2.