Table of Contents

Assignment 3 - Kalman filter for parameter estimation

Written by Peili Guo (peili.guo.7645@student.uu.se) and Francisco José Peralta Alguacil (francisco-jose.peraltaalguacil.0481@student.uu.se) This report is for Computational Finance: Calibration and Estimation Assignment 3.

In this project we apply Kalman filter to pricing models and optimise it to find the corresponding set of parameters.

First, we build the Kalman filter and observe how it works with prediction and data update and try to optimize its parameters with 10 interations of a "guessing optimizer".

In the second part, the maximum likelihood was computed and by calling fminsearch we find the set of parameters that maximize the likelihood of the filtering process. And this procedure was also applied to ABB stock data taken on 2015-02-05, the day when its Q4 report was released.

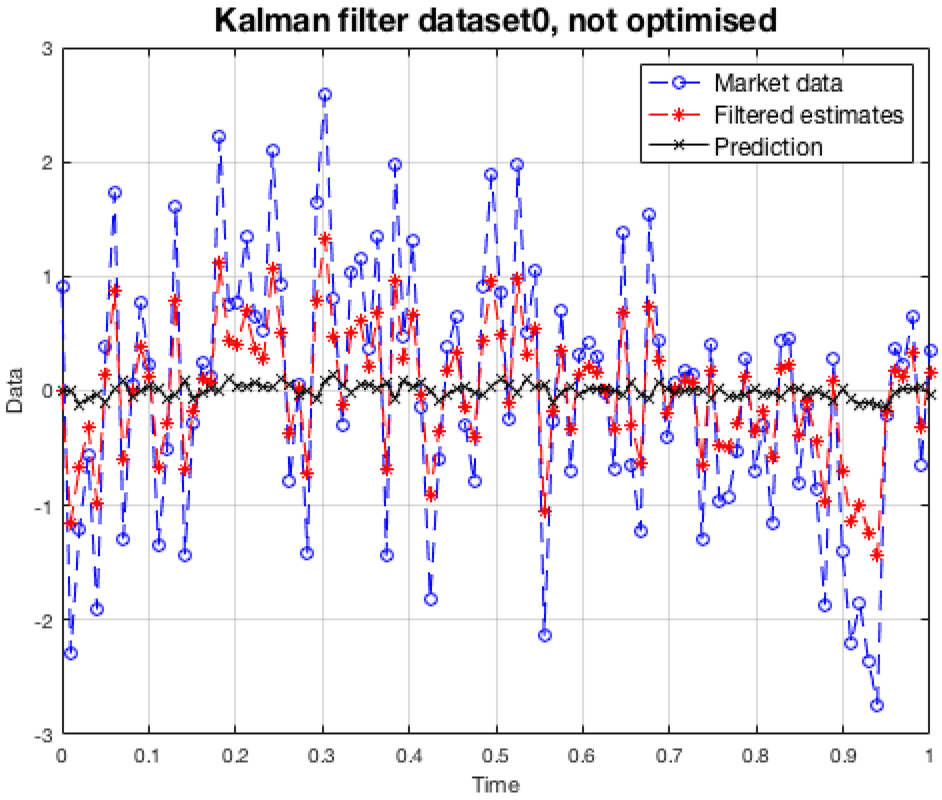
In the last part, we plot the residual and visualize it with histogram to check if it is normally distributed.

```
clear
close all
clc
```

Kalman filter creation

```
load('dataset0.mat');

[xUpd, xPred] = KalmanFilterFunc(0.1,1,1,1,data);

figure(1)
plot(dtime,data,'bo--');
hold on
plot(dtime,xUpd,'r*--');
plot(dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
title("Kalman filter dataset0, not optimised",'FontSize', 16);
```

```
grid on
```



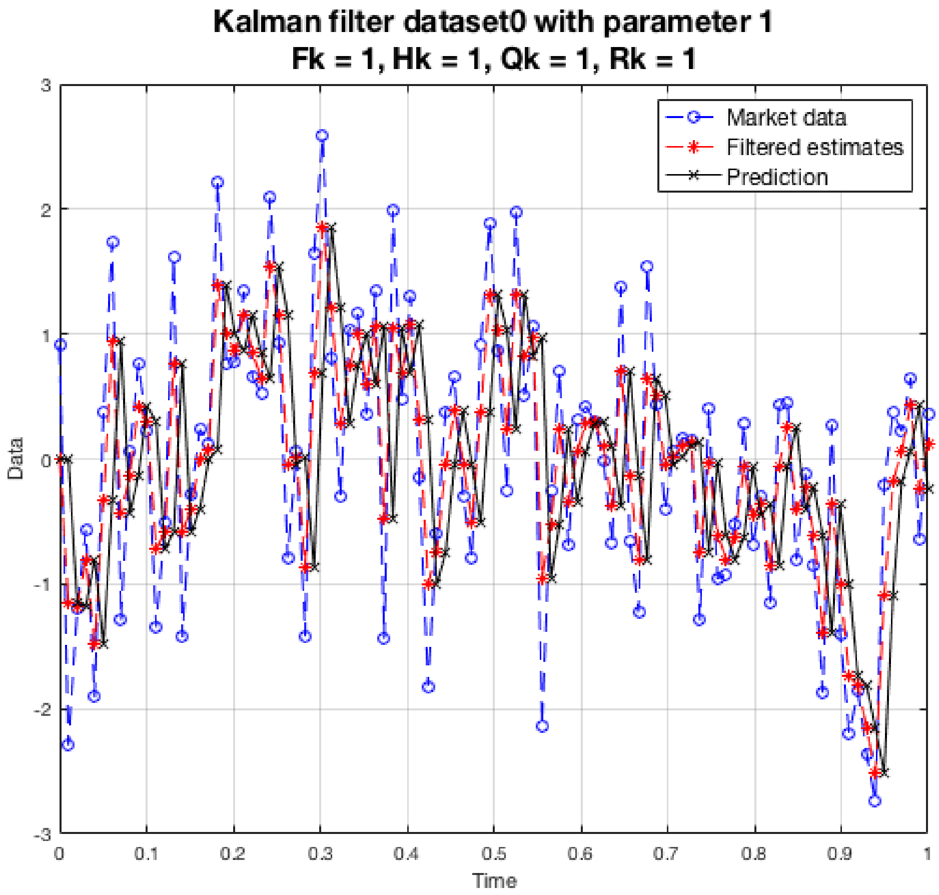
10 interations for guessing optimizer

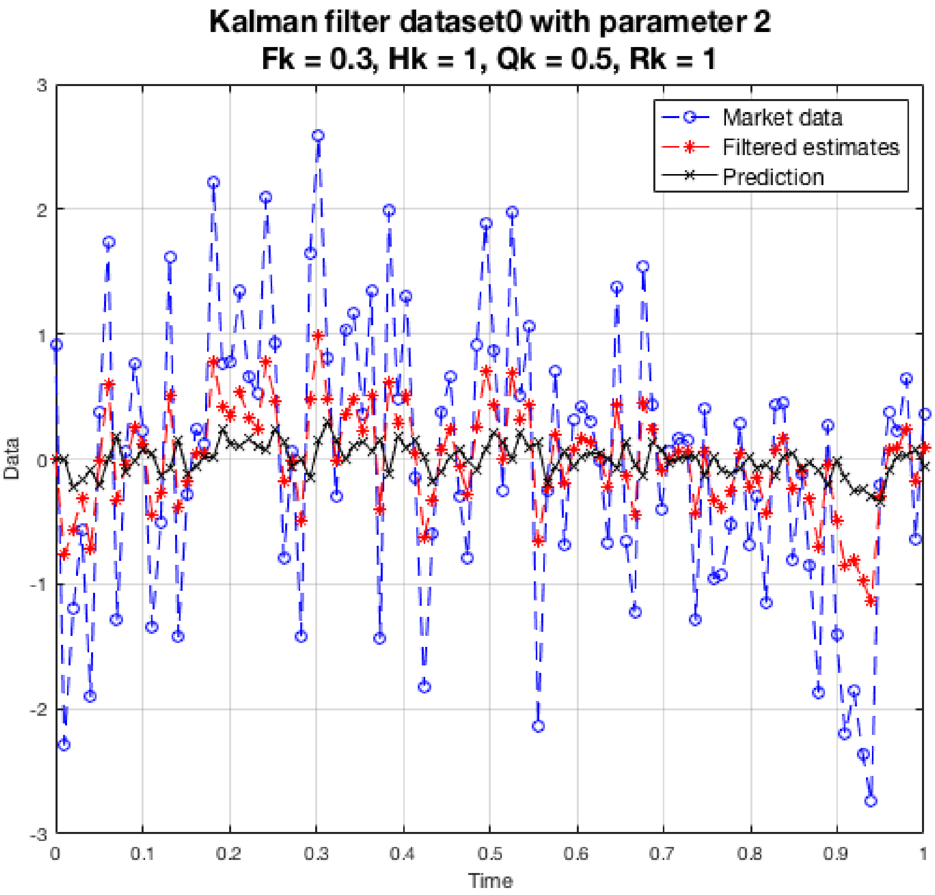
```
load('dataset0.mat');

%iteration 1
ite = [1 1 1 1];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f1 = figure('position', [0, 0, 600, 500]);
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 1";str},'FontSize', 16);
grid on

%iteration 2
ite = [0.3 1 0.5 1];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f2 = figure('position', [0, 0, 600, 500]);
```

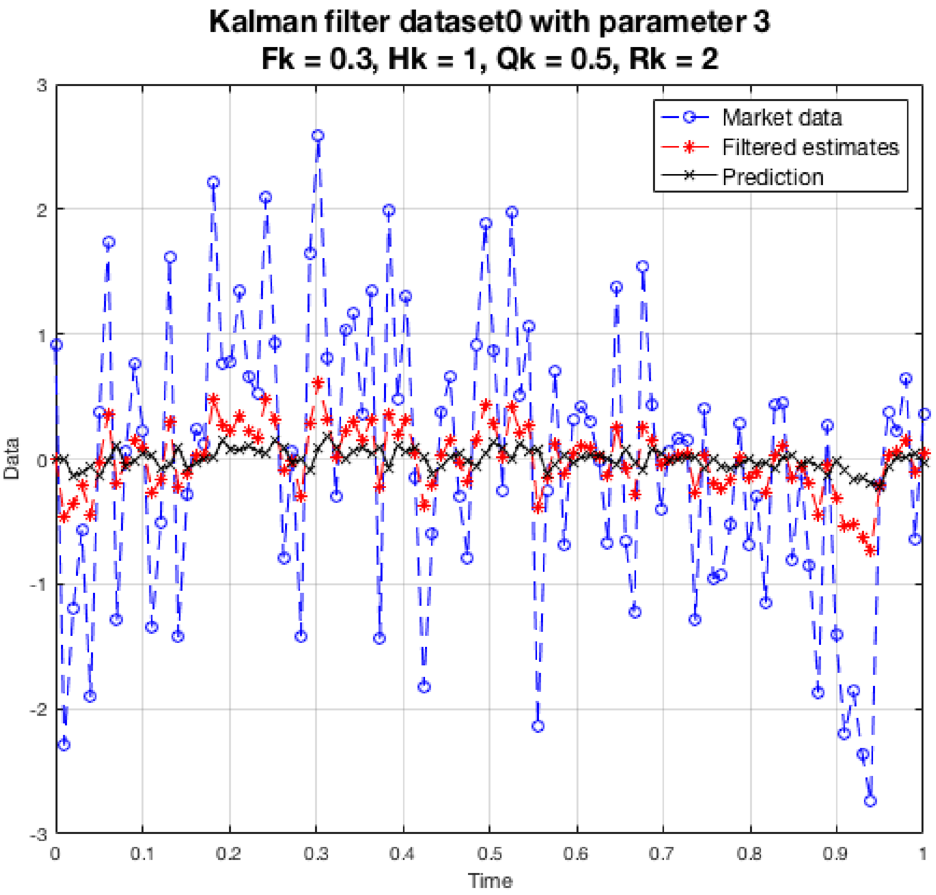
```
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 2";str},'FontSize', 16);
grid on

%iteration 3
ite = [0.3 1 0.5 2];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f3 = figure('position', [0, 0, 600, 500]);
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 3";str},'FontSize', 16);
grid on

%iteration 4
ite = [0.3 5 0.5 2];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f4 = figure('position', [0, 0, 600, 500]);
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 4";str},'FontSize', 16);
grid on

%iteration 5
ite = [2 5 2.5 2];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f5 = figure('position', [0, 0, 600, 500]);
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 5";str},'FontSize', 16);
grid on
```

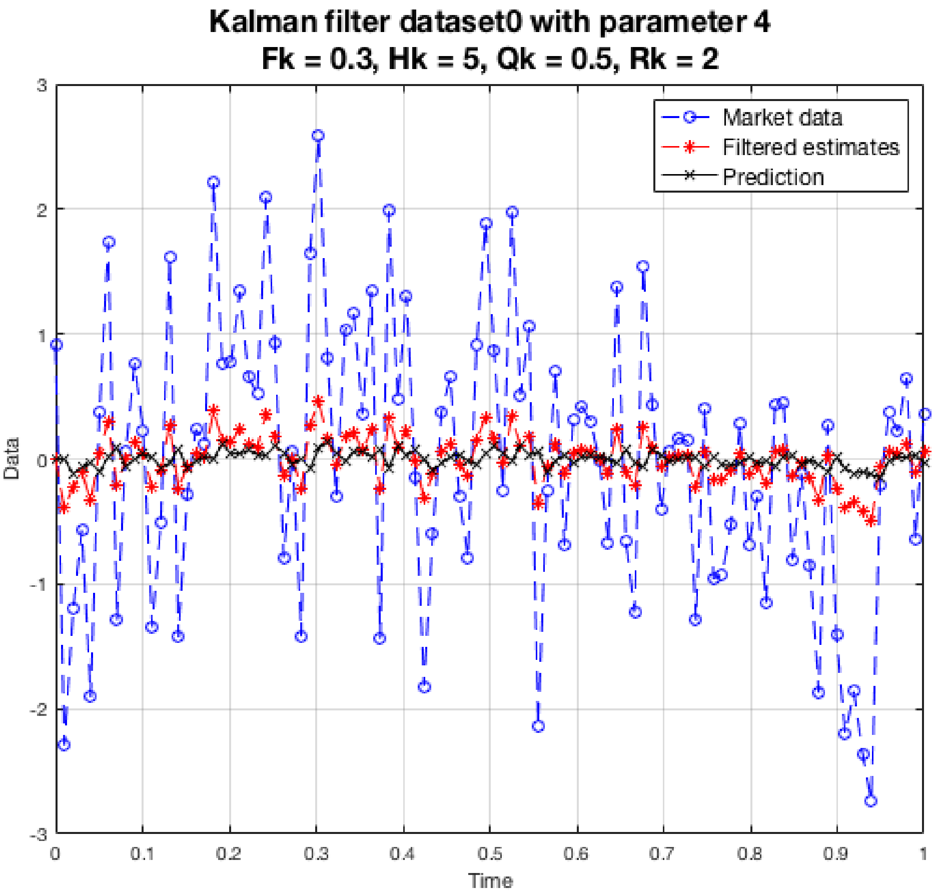
```
%iteration 6
ite = [2 5 2.5 0.25];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f6 = figure('position', [0, 0, 600, 500]);
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 6";str},'FontSize', 16);
grid on


%iteration 7
ite = [2 1 0.5 0.25];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f7 = figure('position', [0, 0, 600, 500]);
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 7";str},'FontSize', 16);
grid on

%iteration 8
ite = [1.2 1.3 0.5 0.25];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f8 = figure('position', [0, 0, 600, 500]);
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 8";str},'FontSize', 16);
grid on


%iteration 9
ite = [0.65 1.15 0.5 0.25];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f9 = figure('position', [0, 0, 600, 500]);
```

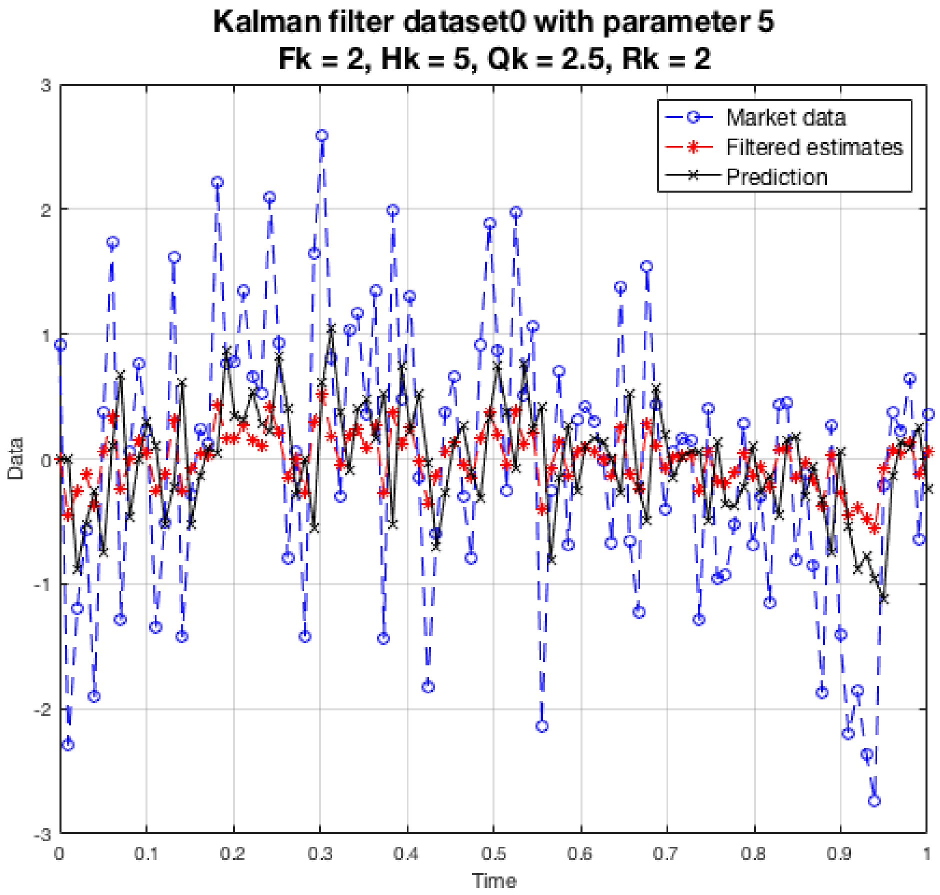
```
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 9";str},'FontSize', 16);
grid on



%iteration 10
ite = [0.65 1.15 0.65 0.05];
[xUpd, xPred] = KalmanFilterFunc(ite(1),ite(2),ite(3),ite(4),data);

f10 = figure('position', [0, 0, 600, 500]);
plot(dtime,data,'bo--',dtime,xUpd,'r*--',dtime,xPred,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
str = sprintf('Fk = %s, Hk = %s, Qk = %s, Rk =
 %s',string(ite(1)),string(ite(2)),string(ite(3)),string(ite(4)));
title({"Kalman filter dataset0 with parameter 10";str},'FontSize',
 16);
grid on
```

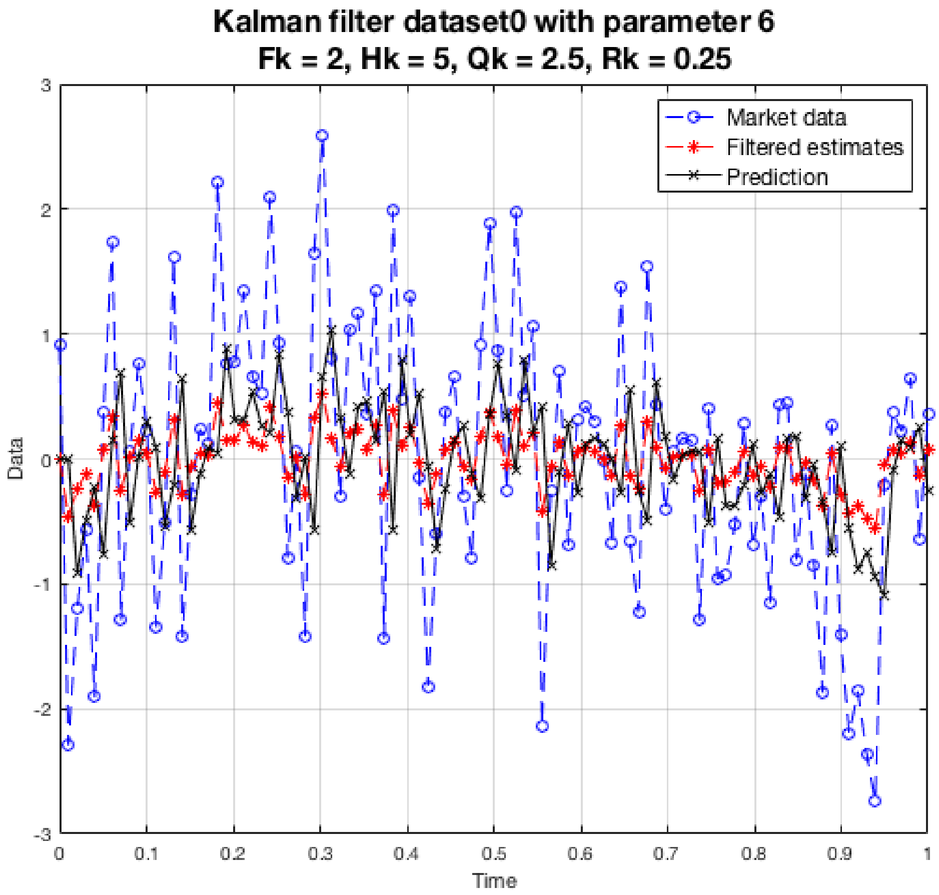

Kalman filter dataset0 with parameter 1
Fk = 1, Hk = 1, Qk = 1, Rk = 1
Market data
Filtered estimates
Prediction
Data
Time
3
2
1
0
-1
-2
-3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

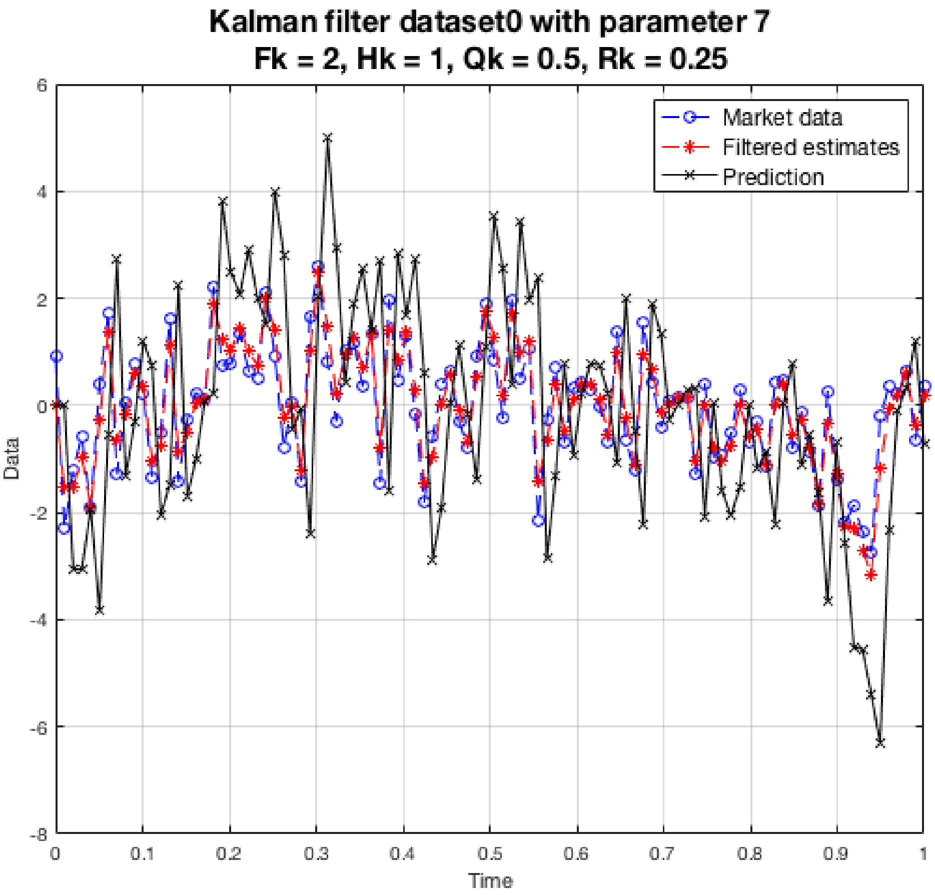

Kalman filter dataset0 with parameter 2
Fk = 0.3, Hk = 1, Qk = 0.5, Rk = 1
Market data
Filtered estimates
Prediction
Data
Time
3
2
1
0
-1
-2
-3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

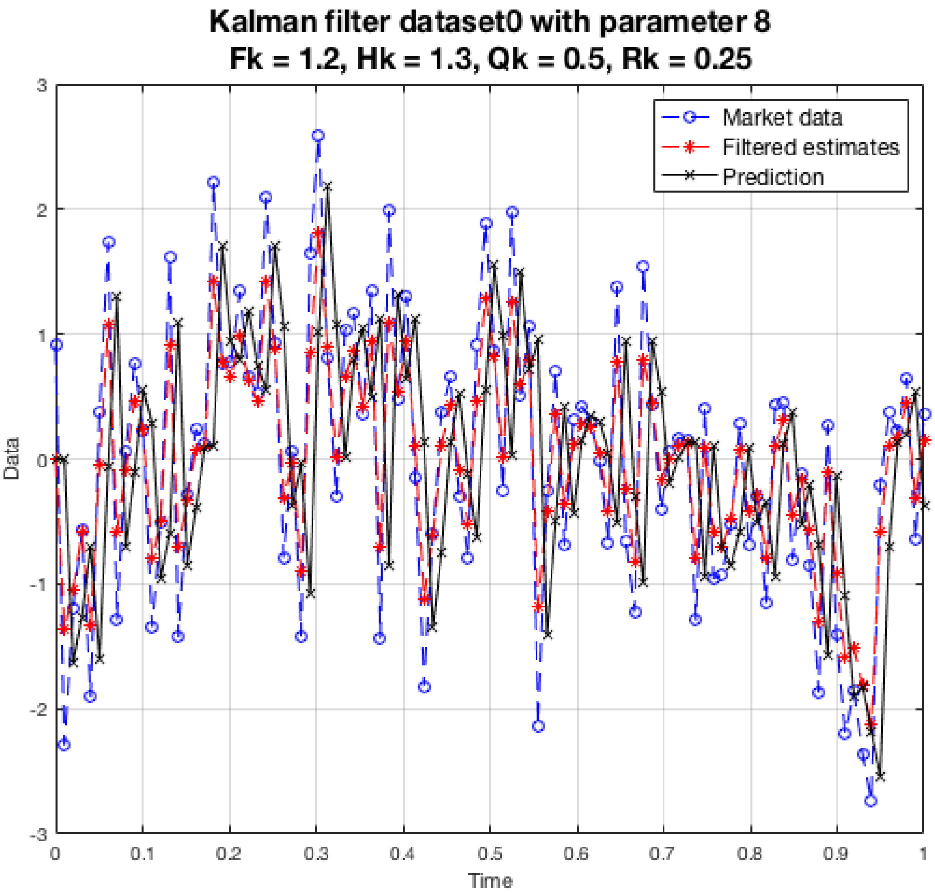

Kalman filter dataset0 with parameter 3
Fk = 0.3, Hk = 1, Qk = 0.5, Rk = 2
Market data
Filtered estimates
Prediction
Data
Time
3
2
1
0
-1
-2
-3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

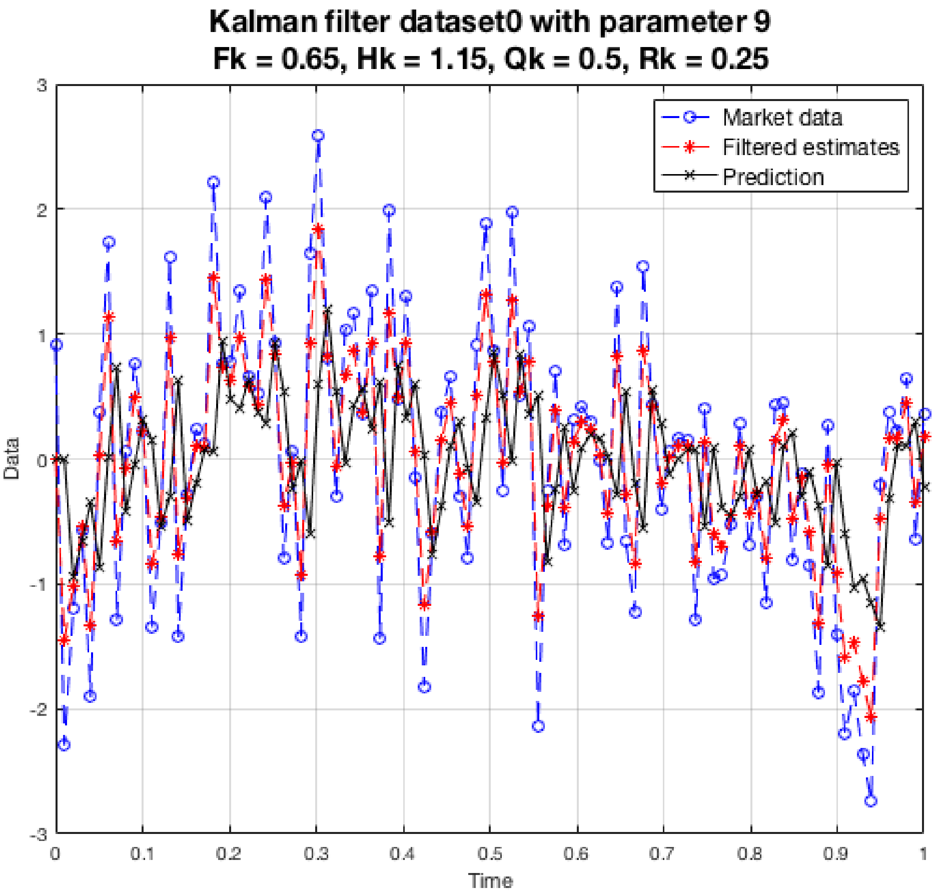

Kalman filter dataset0 with parameter 4
Fk = 0.3, Hk = 5, Qk = 0.5, Rk = 2
Market data
Filtered estimates
Prediction
Data
Time
3
2
1
0
-1
-2
-3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

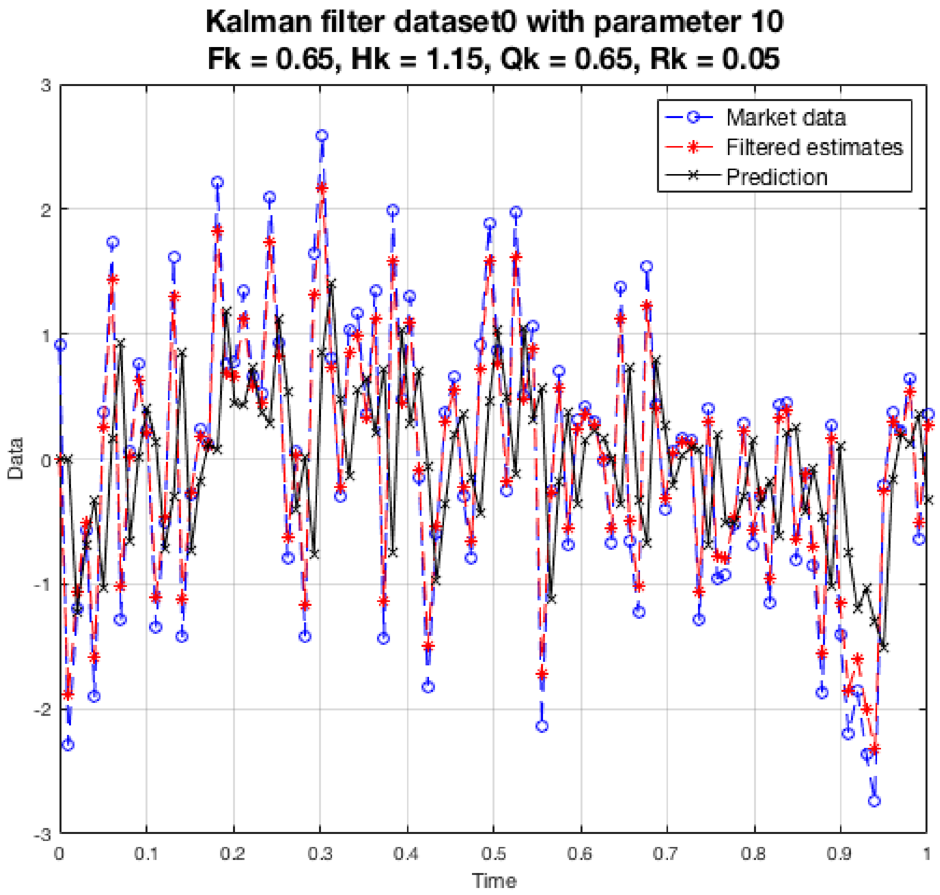

Kalman filter dataset0 with parameter 5
Fk = 2, Hk = 5, Qk = 2.5, Rk = 2
Market data
Filtered estimates
Prediction
Data
Time
3
2
1
0
-1
-2
-3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Kalman filter dataset0 with parameter 6
Fk = 2, Hk = 5, Qk = 2.5, Rk = 0.25
Market data
Filtered estimates
Prediction
Data
Time
3
2
1
0
-1
-2
-3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Kalman filter dataset0 with parameter 7
Fk = 2, Hk = 1, Qk = 0.5, Rk = 0.25
Market data
Filtered estimates
Prediction
Data
Time
6
4
2
0
-2
-4
-6
-8
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Kalman filter dataset0 with parameter 8
Fk = 1.2, Hk = 1.3, Qk = 0.5, Rk = 0.25
Market data
Filtered estimates
Prediction
Data
Time
3
2
1
0
-1
-2
-3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Kalman filter dataset0 with parameter 9
Fk = 0.65, Hk = 1.15, Qk = 0.5, Rk = 0.25
Market data
Filtered estimates
Prediction
Data
Time
3
2
1
0
-1
-2
-3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

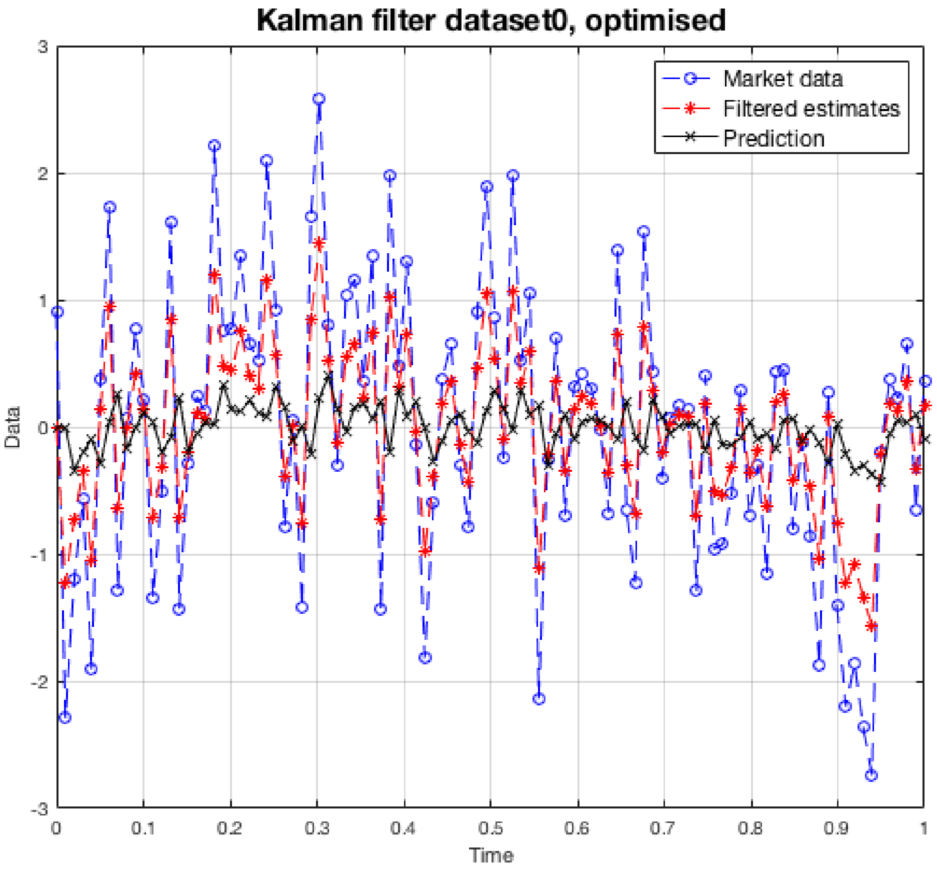


Parameter estimation dataset0

```
clear

load("dataset0.mat");

options = optimset('MaxFunEvals',2000);

theta0 = [0.1 5 5 0.1];
x0 = 0;
p0 = 0;

fun = @(theta)max_like1(theta, x0, p0);

optParameters0 = fminsearch(fun,theta0,options)

optArguments0 = num2cell(optParameters0);

[xUpdOpt0, xPredOpt0] = KalmanFilterFunc(optArguments0{:},data);

figure(2)
plot(dtime,data,'bo--');
```

```
hold on
plot(dtime,xUpdOpt0,'r*--');
plot(dtime,xPredOpt0,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
title("Kalman filter dataset0, optimised",'FontSize',16);
grid on
```

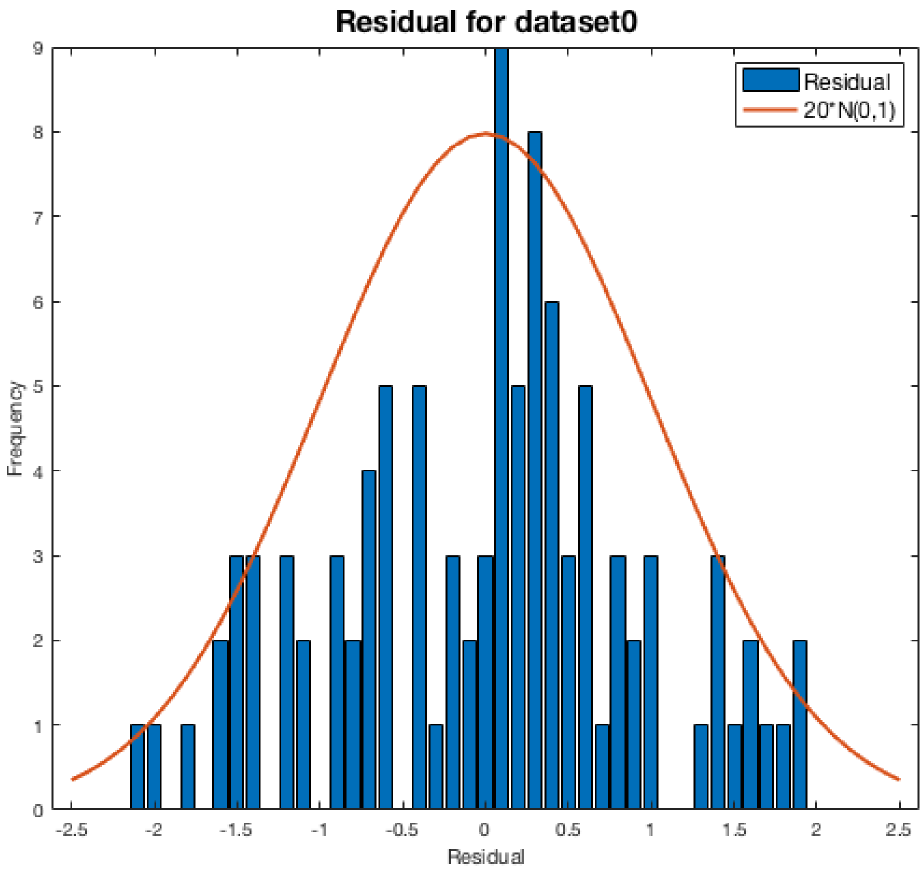
```
optParameters0 =

    0.2752    1.4364    0.4473    0.2721
```



Residual dataset0

```
theta = optParameters0;
load('dataset0.mat');

x0 = 0;
p0 = 0;
```

```
x1(1) = theta(1)*x0;
p1(1) = theta(1)*p0*theta(1)'+theta(3);

sk(1) = theta(2)*p1(1)*theta(2)'+theta(4);
L(1) =-0.5*log(2*pi)-0.5*log(abs(sk(1)))-0.5*(data(1)-
theta(2)*x1(1))'*sk(1)^-1*(data(1)-theta(2)*x1(1));
res(1) = (data(1)-theta(2)*x1(1))/sk(1);

for i=1:1:length(data)-1
    K(i) = p1(i)*theta(2)'*((theta(2)*p1(i)*theta(2)'+theta(4))^(-1));

    xm(i) = x1(i)+K(i)*(data(i)-theta(2)*x1(i));
    pm(i) = (1-K(i)*theta(2))*p1(i);

    x1(i+1) = theta(1)*xm(i);
    p1(i+1) = theta(1)*pm(i)*theta(1)'+theta(3);
    sk(i+1) = theta(2)*p1(i+1)*theta(2)'+theta(4);
    res(i+1) = (data(i+1)-theta(2)*x1(i+1))/sk(i+1);
end

figure(3)
x = -2.5:0.1:2.5;
histu = hist(res,x);
bar(x,histu);
hold on
norm = 20*normpdf(x,0,1);
plot(x,norm,'LineWidth',2);
xlabel('Residual');
ylabel('Frequency');
legend({"Residual", "20*N(0,1)"},'FontSize',12);
title("Residual for dataset0",'FontSize',16);
```

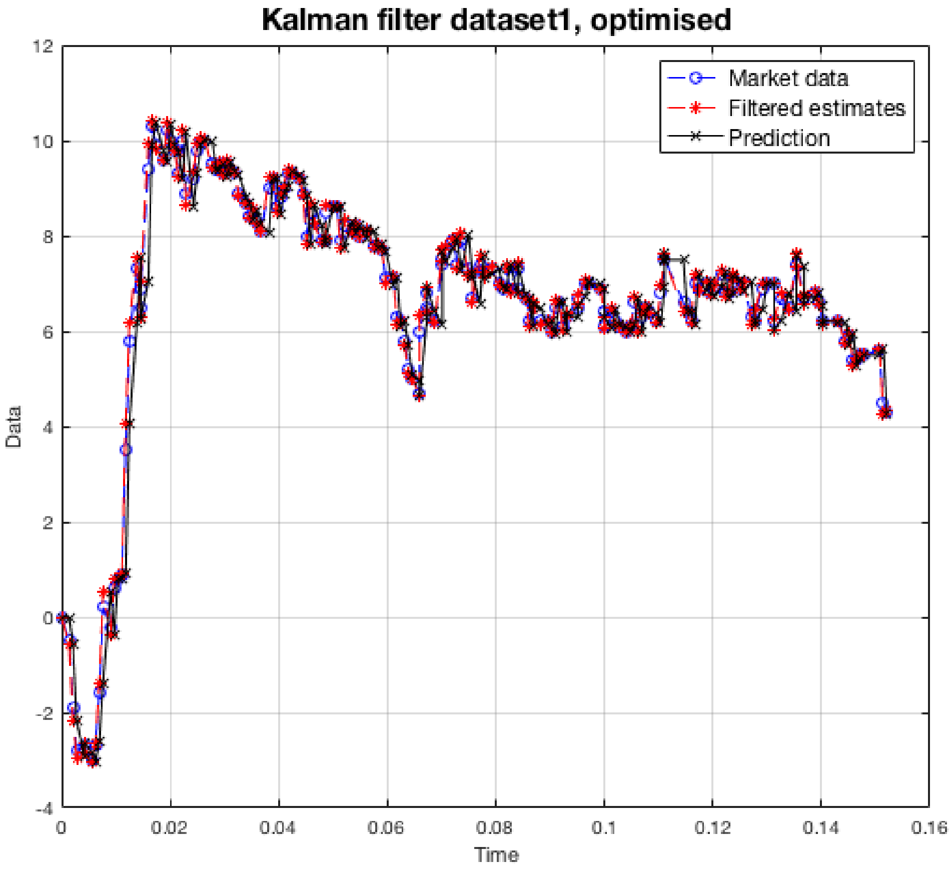


Parameter estimation dataset1

```
clear

load("dataset1.mat");
t = datenum(dtime);
T0 = t(1);
t = t-T0;

options = optimset('MaxFunEvals',2000);

theta0 = [2 1 0.2 0.2];
x0 = 0;
p0 = 0;

fun = @(theta)max_like2(theta, x0, p0);

optParameters1 = fminsearch(fun,theta0,options)

optArguments1 = num2cell(optParameters1);

[xUpdOpt1, xPredOpt1] = KalmanFilterFunc(optArguments1{:},data);
```

```
figure(4)
plot(t,data,'bo--');
hold on
plot(t,xUpdOpt1,'r*--');
plot(t,xPredOpt1,'kx-');
xlabel("Time");
ylabel("Data");
legend({"Market data", "Filtered
 estimates", "Prediction"},'FontSize',12);
title("Kalman filter dataset1, optimised",'FontSize',16);
grid on
```

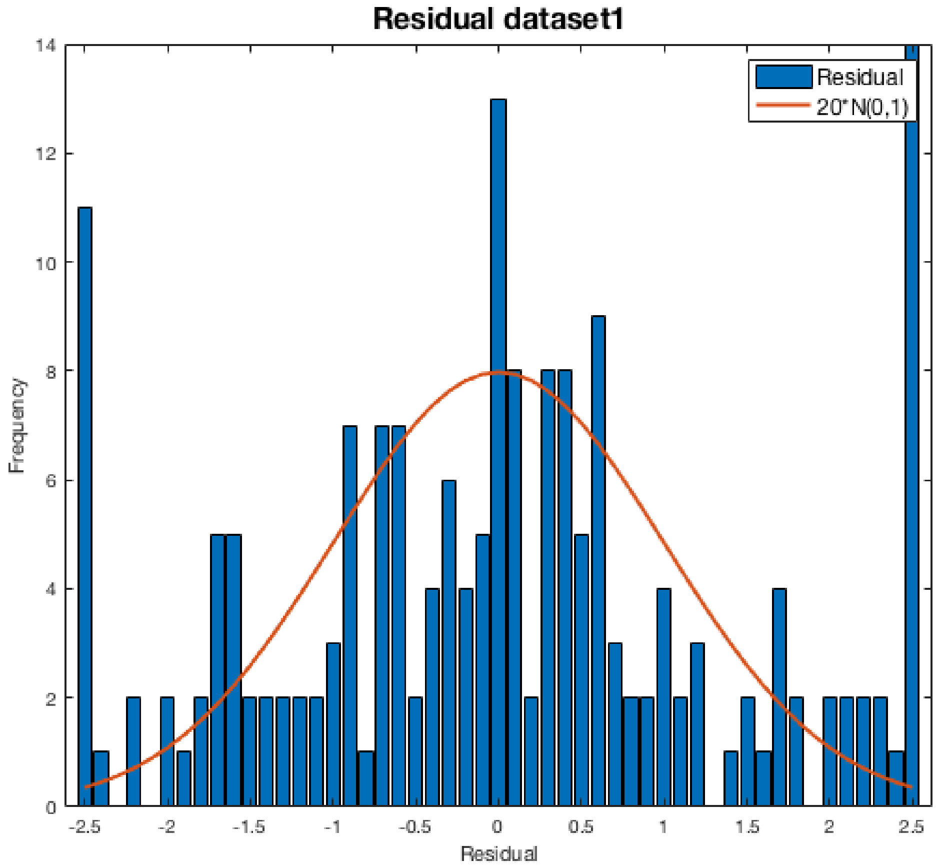
```
optParameters1 =

    0.9968    0.9982    0.4302   -0.0635
```



Residual dataset1

```
theta = optParameters1;
load('dataset1.mat');
```

```
t = datenum(dtime);
T0 = t(1);
t = t-T0;

x0 = 0;
p0 = 0;

x1(1) = theta(1)*x0;
p1(1) = theta(1)*p0*theta(1)'+theta(3);

sk(1) = theta(2)*p1(1)*theta(2)'+theta(4);
L(1) =-0.5*log(2*pi)-0.5*log(abs(sk(1)))-0.5*(data(1)-
theta(2)*x1(1))'*sk(1)^-1*(data(1)-theta(2)*x1(1));
res(1) = (data(1)-theta(2)*x1(1))/sk(1);

for i=1:1:length(data)-1
    K(i) = p1(i)*theta(2)'*((theta(2)*p1(i)*theta(2)'+theta(4))^(-1));

    xm(i) = x1(i)+K(i)*(data(i)-theta(2)*x1(i));
    pm(i) = (1-K(i)*theta(2))*p1(i);

    x1(i+1) = theta(1)*xm(i);
    p1(i+1) = theta(1)*pm(i)*theta(1)'+theta(3);
    sk(i+1) = theta(2)*p1(i+1)*theta(2)'+theta(4);
    res(i+1) = (data(i+1)-theta(2)*x1(i+1))/sk(i+1);
end

figure(5)
x = -2.5:0.1:2.5;
histu = hist(res,x);
bar(x,histu);
hold on
norm = 20*normpdf(x,0,1);
plot(x,norm,'LineWidth',2);
xlabel('Residual');
ylabel('Frequency');
legend({"Residual", "20*N(0,1)"},'FontSize',12);
title("Residual dataset1",'FontSize',16);
```



Reflections

```
% In the parameter optimisation process, the function fminsearch is
 likely
% to find a local minimunm of the objective function. That is why the
% obtained results might be susceptible to slight changes in the
 initial
% guesses (specially for the dataset1, where we had to find the
 initial
% guess ourselves).
% It is interesting to mention that, according to the obtained plots,
 it
% seems that the optimization process worked better on dataset1, which
% correspond to the ABB stock prices. However, when computing the
 residual
% for it, we could observe that it was not proportional to N(0,1),
 which
% could be because of the fact that the Q4 report was released on that
% date.
```

Functions called in the scripts

```
dbtype('KalmanFilterFunc.m');
dbtype('max_like1.m');
dbtype('max_like2.m');
```

```
function [xUpd,xPred] = KalmanFilterFunc(Fk,Hk,Qk,Rk,data)
    x0 = 0;
    P0 = 0;

    xPred = zeros(length(data),1);
    PPred = zeros(length(data),1);

    xUpd = zeros(length(data),1);
    PUpd = zeros(length(data),1);

    xPred(1) = x0;
    PPred(1) = P0;

    K = PPred(1)*Hk/(Hk*PPred(1)*Hk+Rk);

    xUpd(1) = xUpd(1)+K*(data(1)-Hk*xUpd(1));
    PUpd(1) = (1-K*Hk)*PUpd(1);

    for i=1:length(data)-1
        xPred(i+1) = Fk*xUpd(i);
        PPred(i+1) = Fk*PUpd(i)*Fk+Qk;

        K = PPred(i+1)*Hk/(Hk*PPred(i+1)*Hk+Rk);

        xUpd(i+1) = xPred(i+1)+K*(data(i+1)-Hk*xPred(i+1));
        PUpd(i+1) = (1-K*Hk)*PPred(i+1);
    end
end
```

```
function y = max_like1(theta,x0,p0)

    load('dataset0.mat');

    x1(1) = theta(1)*x0;
    p1(1) = theta(1)*p0*theta(1)'+theta(3);

    sk(1) = theta(2)*p1(1)*theta(2)'+theta(4);
    L(1) = -0.5*log(2*pi)-0.5*log(abs(sk(1)))-0.5*(data(1)-theta(2)*x1(1))'*sk(1)^-1*(data(1)-theta(2)*x1(1));

    for i=1:1:length(data)-1
        K(i)=p1(i)*theta(2)'*((theta(2)*p1(i)*theta(2)'+theta(4))^(-1));

        xm(i) = x1(i)+K(i)*(data(i)-theta(2)*x1(i));
        pm(i) = (1-K(i)*theta(2))*p1(i);
        x1(i+1) = theta(1)*xm(i);
        p1(i+1) = theta(1)*pm(i)*theta(1)'+theta(3);

        sk(i+1) = theta(2)*p1(i+1)*theta(2)'+theta(4);
        L(i+1) = -0.5*log(2*pi)-0.5*log(abs(sk(i+1)))-0.5*(data(i+1)-theta(2)*x1(i+1))'*(sk(i+1)^(-1))*(data(i+1)-theta(2)*x1(i+1));
    end

    y = -sum(L);
end

```

```
function y = max_like2(theta,x0,p0)
    load('dataset1.mat');

    t = datenum(dtime);
    T0 = t(1);
    t = t-T0;


    x1(1) = theta(1)*x0;
    p1(1) = theta(1)*p0*theta(1)'+theta(3);


    sk(1) = theta(2)*p1(1)*theta(2)'+theta(4);
    L(1) = -0.5*log(2*pi)-0.5*log(abs(sk(1)))-0.5*(data(1)-theta(2)*x1(1))'*sk(1)^-1*(data(1)-theta(2)*x1(1));

    for i=1:1:length(data)-1

        K(i) = p1(i)*theta(2)'*((theta(2)*p1(i)*theta(2)'+theta(4))^(-1));

        xm(i) = x1(i)+K(i)*(data(i)-theta(2)*x1(i));
        pm(i) = (1-K(i)*theta(2))*p1(i);

        x1(i+1) = theta(1)*xm(i);
        p1(i+1) = theta(1)*pm(i)*theta(1)'+theta(3);
        sk(i+1) = theta(2)*p1(i+1)*theta(2)'+theta(4);
        L(i+1) = -0.5*log(2*pi)-0.5*log(abs(sk(i+1)))-0.5*(data(i+1)-theta(2)*x1(i+1))'*(sk(i+1)^(-1))*(data(i+1)-theta(2)*x1(i+1));
    end

    y = -sum(L);
end

```

Published with MATLAB® R2018a