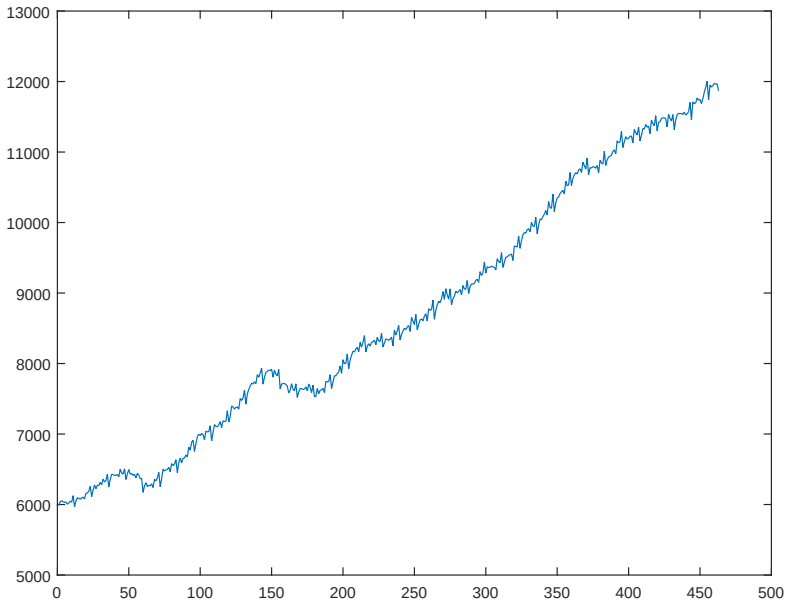
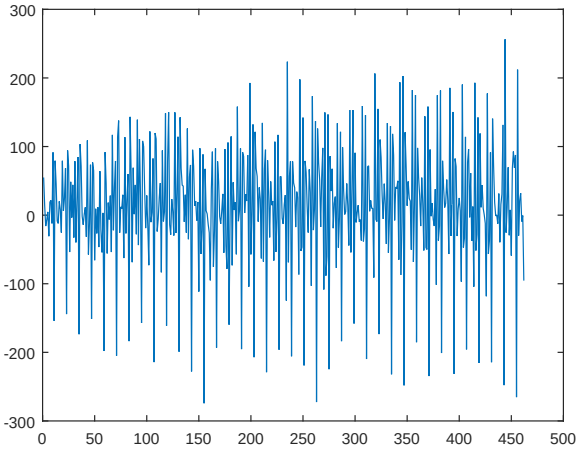
FORECASTING PROJECT: Australian labour force

Introduction

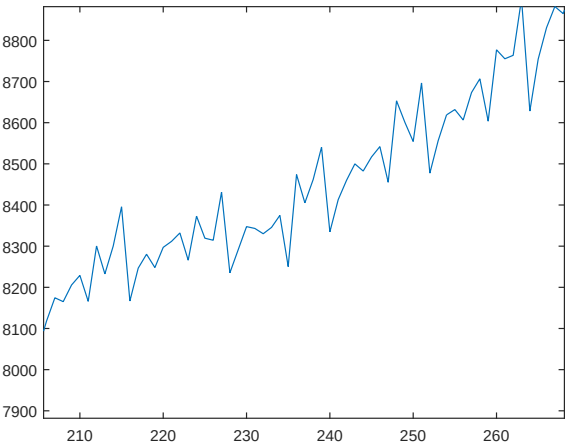
The variable I am forecasting is the Australian labour force which is the number of individuals in Australia who are able to work. The forecasts include forecast for all the months from September 2016 to August 2017 and density forecasts for the same months. The data for the month of September 2016 will be available on the website of the Australian Bureau of Statistics on October 20th. The graph below represents the evolution of the Australian labour force from February 1978 to August 2016, the y-axis represents the labour force (in thousands) and the x-axis represents the months.



There is a clear upward trend in the data, continuously increasing until August 2016, going from around 6 million at the beginning of the period to around 12 million at the end of our period. Looking at the graph, it is clear that the data is non-stationary and this will cause problems in our forecasting models: the statistical properties of our time series such as mean, variance and autocorrelation are not constant over time. This will be a problem particularly in our AR models, we then have to take the first difference (period-period change) which gives us:



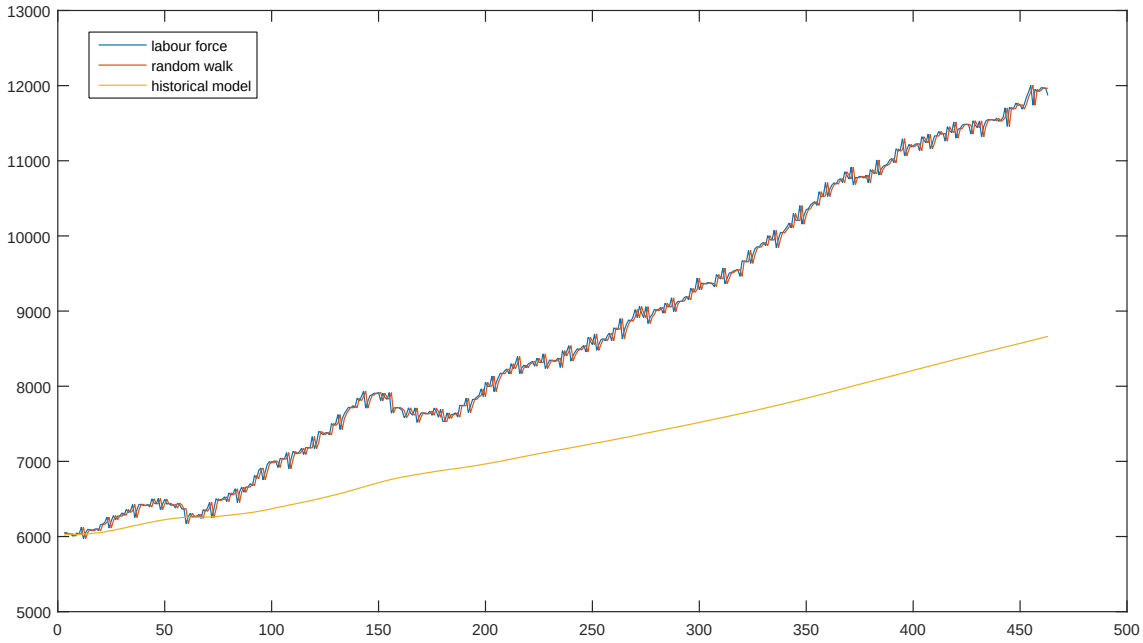
Now, we can use AR models on the first difference. Furthermore, our original data does not appear to have any monthly or seasonal effect, this is due to the scale of the graph. If we zoom in, we can indeed observe some monthly or quarterly effects:



It is clear from this graph that there are indeed seasonal effects, they appear to be quarterly effects. We will then include a model with monthly and quarterly dummies. Two predictor will also be taken into account in our model, namely the Australian GDP and the Australian population during the same period.

Benchmark model

The benchmark model will be the random walk model. The mean squared forecasting error for the random walk model is $8.9491e+03$ whereas for the historical model, the MSFE is $3.2598e+06$. The random walk model is a good benchmark to forecast our time series.



In yellow is the historical model, it is really far from the actual observations. The random walk model seems to forecast much better the labour force which is expected, the random walk model follows the actual observations with a lag of one period. Therefore, we will use the MSFE of the random walk model ($8.9491e+03$) as a benchmark for our relative error measures. The code can be found in the appendix under code 1.

Seasonal effect

The excel file has been modified by adding a column with the monthly dummies and column with the quarterly dummies. Using a model with monthly and quarterly dummies:

$$y_t = \alpha t + \sum_{i=1}^{4} \beta_i Q_{it} + \epsilon_t$$

and

$$y_t = \alpha t + \sum_{i=1}^{12} \theta_i M_{it} + \epsilon_t$$

where Q_{it} and M_{it} are respectively the quarterly dummies and the monthly dummies. Here, we assume that T0=50, i.e. we use the first 50 months to forecast the remaining

months. For the model including an effect for each quarter, we find a MSFE of 6.9203e+04 while for the monthly model, the MSFE is 7.1473e+04. There is more a quarterly effect in our time series than a monthly effect but both models have a higher error than the error in our benchmark model. We will then use the quarterly model and try to improve it to beat our benchmark. The code can be found in the appendix under code 2.

Predictors

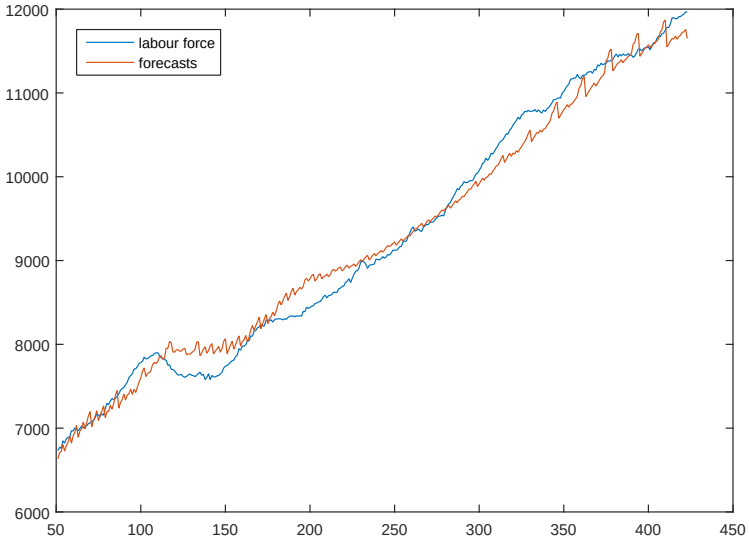
It seems relevant to include the increase of the GDP and the increase of the population during the period studied in our model. With the quarterly model, we will now introduce a predictor which is the Australian GDP for the same period. There is only quarterly data for the GDP so four consecutive months in our labour force time series will correspond to the same GDP for this quarter. The model now is:

$$y_t = \alpha t + \sum_{i=1}^{4} \beta_i Q_{it} + \gamma x_{t-h} \epsilon_t$$

where x_{t-h} is the GDP at time $t-h$. (code 3 in the appendix). The MSFE becomes 4.5383e+04, lower than the previous 6.9203e+04 but the relative mean squared forecasting error is 5.0713 which means that the forecasting error is five times that of the random walk model. Now, we add another predictor (the Australian population) to improve the model.

$$y_t = \alpha t + \sum_{i=1}^{4} \beta_i Q_{it} + \gamma x_{t-h} + \delta z_{t-h} + \epsilon_t$$

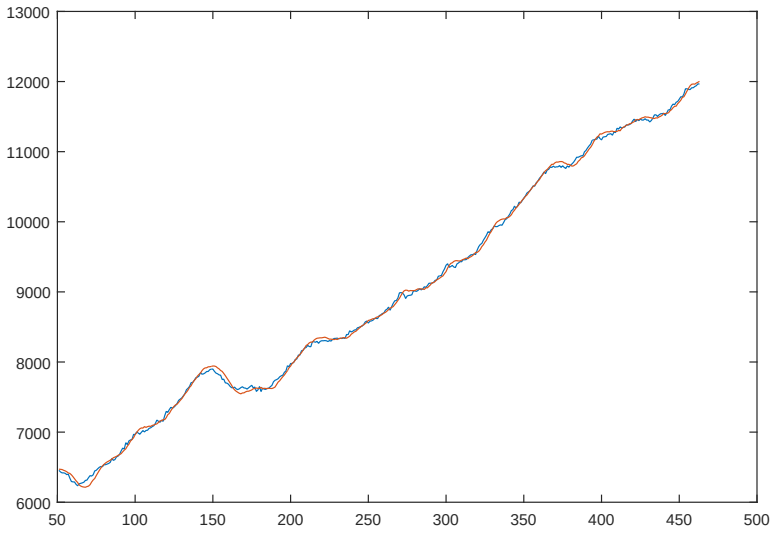
where z_{t-h} is the Australian population at time $t-h$. However, there is only data on the Australian population from June 1981 so we are cutting the first 40 months of our labour force time series and our gdp time series to get the same vector length as the population time series. Now, the MSFE becomes 3.6841e+04, it still gets lower but the relMSFE is still superior than 1 (=4.1167): our model still performs worse than a simple random walk.



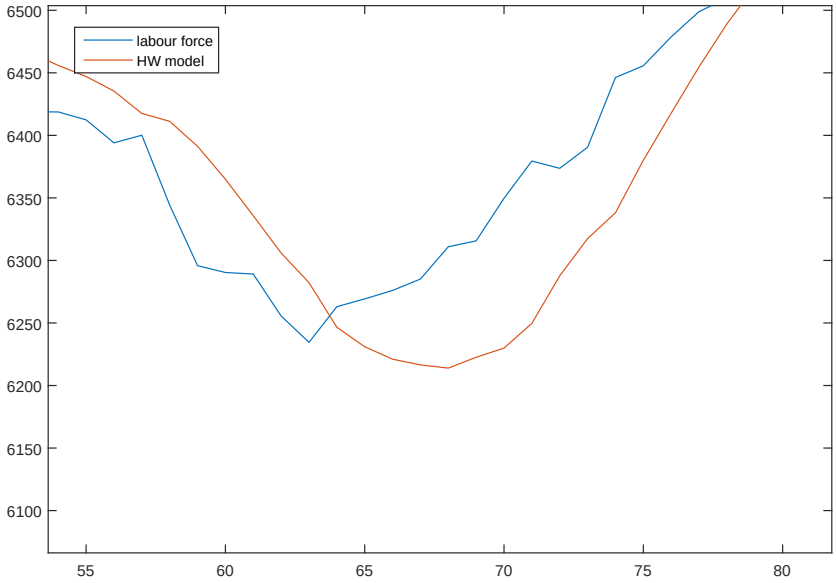
(code 4 in the appendix)

Holt-Winters Smoothing

We now introduce a more sophisticated model. With the Holt-Winters Smoothing, we find a MSFE of 1.9810e+03 which is lower than our benchmark's, the relMSFE is 0.2214, the error in this model is 0.2214 times the error in the random walk model. We have improved the precision of our model.



From the graph, we can see that the Holt-Winters model (in orange) forecasts well the time series (in blue) but if we zoom in, we still have some significant differences:



(code 5 from the appendix)

AR model

From the data of the labour force, we can see that the values are closely related to past value, there is not a large difference between the present value and close past values. Past values might have an effect on current values. It might then be relevant to use an autoregressive model (AR) to capture this effect. The form of an AR(p) model with p lags is:

$$y_t = c + \phi_1 y_{t-1} + ... + \phi_p y_{t-p} + \epsilon_t$$

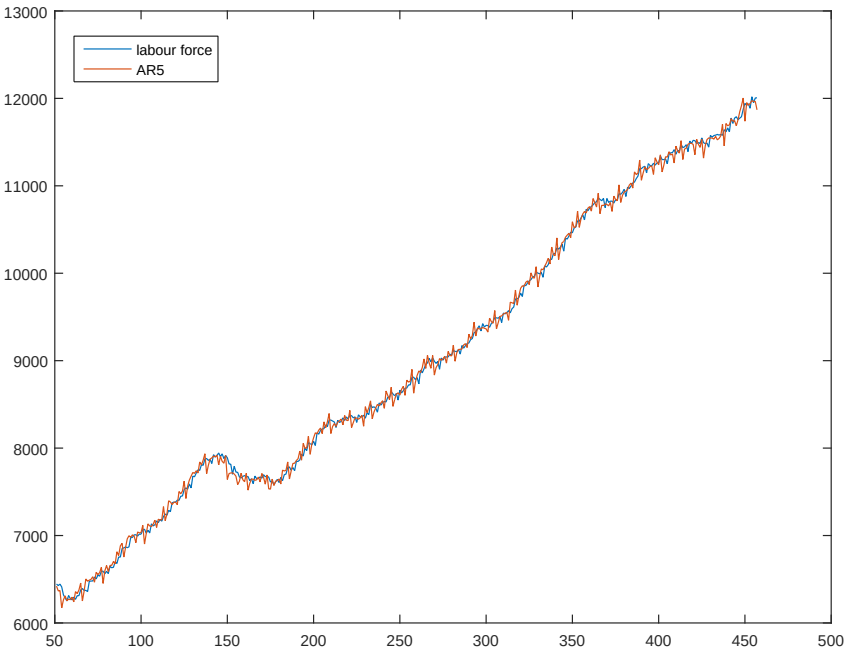
where ϵ_t is an uncorrelated innovation process with mean zero.

We now have to determine the best AR model in terms of lags.

Table 1: AR process

	AR1	AR2	AR3	AR4	AR5	AR6
MSFE	9.4032e+03	6.6201e+03	5.4193e+03	5.4143e+03	5.3204e+03	5.3380e+03
relMSFE	1.0507	0.7398	0.6056	0.6050	0.5945	0.5965

We see that after one lag, we get a model that beats our benchmark model, the relMSFE is lower than 1. The best model is the AR5 process which means that after 5 lags, the past starts losing its predictive power: the error for the AR6 process is higher than for the AR5 process. (code 6 in the appendix).



This model still performs worse than the Holt-Winters model for one-step ahead forecasts. Now, we have to remember that our time series is not stationary so it is more relevant to study the first difference of our time series than the actual observations.

Table 2: AR process on first diff

	AR1_firstdiff	AR2_firstdiff	AR3_firstdiff	AR4_firstdiff
MSFE	6.6110e+03	6.2173e+03	5.9776e+03	6.0108e+03
relMSFE	0.7387	0.6947	0.6680	0.6717

Now, the best model is the AR3 model (code 7 in the appendix). For one lag, the AR model already performs better than the benchmark. Although in this case, the error is larger than for the best AR model with our original time series.

Forecasts

We now have two competing models which are the Holt-winters model and the AR3 model on the first difference of our time series. We have to remember that the Holt-Winters model loses its precision as the number of steps ahead forecasted, h, increases. The Holt-Winters model might not be very accurate in forecasting the labour force in the next twelve months, we then have to compare the errors of the two models as the number of steps ahead forecasted increases.

Table 3: Comparison between AR3 and HW

relMSFE	AR3_firstdiff	Holt-Winters
h=1	0.6680	0.2214
h=2	0.9152	0.3197
h=3	0.9275	0.4466
h=4	0.9646	0.6043
h=5	0.8756	0.7932
h=6	0.9423	1.0168
h=7	0.9057	1.2743
h=8	0.9038	1.5669
h=9	0.6836	1.8924
h=10	0.2118	2.2581
h=11	0.2108	2.6577
h=12	0.2137	3.0947

We observe that the Holt-Winters model performs better than the AR3 process until h reaches 6. For $h = 6$, the Holt-Winters model performs worse than the benchmark (i.e. $relMSFE > 1$). Therefore, to forecast the Australian labour force, we will use the Holt-Winters model to forecast the first 5 months from September 2016 to January 2017 and then use the AR3 process to forecast the remaining months until August 2018. The point forecasts are then:

Table 4: Point forecasts for the coming year

	Point forecasts
Sep 2016	1.2011e+04 (HW model)
Oct 2016	1.2018e+04 (HW model)
Nov 2016	1.2047e+04 (HW model)
Dec 2016	1.2057e+04 (HW model)
Jan 2017	1.2071e+04 (HW model)
Feb 2017	-2.0295 (AR3 on first diff)
March 2017	-15.0879 (AR3 on first diff)
April 2017	16.4162 (AR3 on first diff)
May 2017	-15.4250 (AR3 on first diff)
June 2017	-3.1912 (AR3 on first diff)
July 2017	5.1112 (AR3 on first diff)
August 2017	-83.0737 (AR3 on first diff)

which gives us the following forecasts:

Table 5: Points forecasts for the following 12 months starting in Sept 2016

	Point forecasts (in thousands)
Sep 2016	12011.189
Oct 2016	12018.175
Nov 2016	12047.355
Dec 2016	12057.371
Jan 2017	12070.916
Feb 2017	12068.887
March 2017	12053.799
April 2017	12070.215
May 2017	12054.790
June 2017	12051.599
July 2017	12056.710
August 2017	11973.636

The density forecasts are (code 8 in the appendix):

Table 6: Density forecasts for the following 12 months starting in September 2016

	90% confidence interval	
Sep 2016	-61.3510	177.6884
Oct 2016	-155.3112	124.1457
Nov 2016	-164.1685	116.8099
Dec 2016	-116.5341	169.6654
Jan 2017	-184.1954	88.1459
Feb 2017	-142.6665	138.6076
March 2017	-152.7846	122.6087
April 2017	-121.0130	153.8454
May 2017	-134.8624	104.0125
June 2017	-68.9598	62.5774
July 2017	-60.4403	70.6626
August 2017	-148.6281	-17.5193

For the density forecasts, the AR3 process on the first difference of the time series is used so the values represent the difference from the previous months' level: it gives us an interval of variation of the level of labour force from one month to the next.

Appendix

a) Code 1: random walk

```
data=xlsread('labor','Data1');
data=data(6:end,3);
T=length(data);

%one-step ahead forecasts historical mean and random walk
h=1; T0=2;
datahat_h=zeros(T-h-T0+1,1);
datahat_r=zeros(T-h-T0+1,1);
data_obs=data(T0+h:end);
for t=T0:T-h
    datat=data(1:t);
    datahat_r(t-T0+h,:)=datat(end);
    datahat_h(t-T0+h,:)=mean(datat);
end
MSFE_RW=mean((data_obs-datahat_r).^2)
MSFE_h=mean((data_obs-datahat_h).^2)

plot(1:T,data);
print -depsc myfig.eps

plot(T0+h:T,data_obs,T0+h:T,datahat_r,T0+h:T,datahat_h);
```

b) Code 2: seasonal effect

```
data=xlsread('data');
y=data(:,1);
T=length(y);
t=(1:T)';

Q=data(:,4);
R=data(:,2);
T0=50;

%% construct 4 dummy vairables
D1=(Q==1); D2=(Q==2);
D3=(Q==3); D4= (Q==4);
%% construct 12 dummy variables
B1=(R==1); B2=(R==2); B3=(R==3);
B4=(R==4); B5=(R==5); B6=(R==6);
B7=(R==7); B8=(R==8); B9=(R==9);
```

```
B10=(R==10); B11=(R==11); B12=(R==12);

h=1;
syhat=zeros(T-h-T0+1, 1);
ytph=y(T0+h:end);%observed y_{t+h}
for t=T0:T-h
    yt=y(1:t);
    D1t=D1(1:t); D2t=D2(1:t);
    D3t=D3(1:t); D4t=D4(1:t);
    Xt1=[(1:t)' D1t D2t D3t D4t];
    beta=(Xt1'*Xt1)\(Xt1'*yt);
    yhat=[t+h D1(t+h) D2(t+h) D3(t+h) D4(t+h)]*beta;
    syhat(t-T0+1)=yhat;
end
MSFE=mean((ytph-syhat).^2)

h=1;
syhat1=zeros(T-h-T0+1, 1);
ytph1=y(T0+h:end);%observed y_{t+h}
for t=T0:T-h
    y1t=y(1:t);
    B1t=B1(1:t); B2t=B2(1:t); B3t=B3(1:t);
    B4t=B4(1:t); B5t=B5(1:t); B6t=B6(1:t);
    B7t=B7(1:t); B8t=B8(1:t); B9t=B9(1:t);
    B10t=B10(1:t); B11t=B11(1:t);B12t=B12(1:t);
    Xt2=[(1:t)' B1t B2t B3t B4t B5t B6t B7t B8t B9t B10t B11t
        B12t];
    beta1=(Xt2'*Xt2)\(Xt2'*y1t);
    yhat1=[t+h B1(t+h) B2(t+h) B3(t+h) B4(t+h) B5(t+h) B6(t+h)
        B7(t+h) B8(t+h) B9(t+h) B10(t+h) B11(t+h) B12(t+h)]*
        beta1;
    syhat1(t-T0+1)=yhat1;
end
MSFE1=mean((ytph-syhat1).^2)
```

c) Code 3: GDP predictor

```
data=xlsread('data');
y=data(:,1);
g=data(:,3);
T=length(y);
t=(1:T)';

```

```
Q=data(:,4);
R=data(:,2);
T0=50;

%% construct 4 dummy vairables
D1=(Q==1); D2=(Q==2);
D3=(Q==3); D4= (Q==4);
%% construct 12 dummy variables
B1=(R==1); B2=(R==2); B3=(R==3);
B4=(R==4); B5=(R==5); B6=(R==6);
B7=(R==7); B8=(R==8); B9=(R==9);
B10=(R==10); B11=(R==11); B12=(R==12);

h=1;
syhat=zeros(T-h-T0+1, 1);
ytph=y(T0+h:end);%observed y_{t+h}
for t=T0:T-h
    yt=y(1:t);
    yt=yt(h+1:t, :);
    D1t=D1(1:t); D2t=D2(1:t);
    D3t=D3(1:t); D4t=D4(1:t);
    gdp=lagmatrix(g(1:t),h);
    Xt=[lagmatrix((1:t),h) D1t D2t D3t D4t gdp];
    Xt=Xt(h+1:t, :);
    beta=(Xt'*Xt)\(Xt'*yt);
    yhat=[t+h D1(t+h) D2(t+h) D3(t+h) D4(t+h) gdp(t)]*beta;
    syhat(t-T0+1)=yhat;
end
MSFE1=mean((ytph-syhat).^2)
```

d) Code 4: GDP and population predictors

```
data=xlsread('data');
y=data(41:end,1);
p=data(41:end,5);
g=data(41:end,3);
T=length(y);
t=(1:T)';

Q=data(:,4);
R=data(:,2);
T0=50;

```

```
%% construct 4 dummy vairables
D1=(Q==1); D2=(Q==2);
D3=(Q==3); D4= (Q==4);
%% construct 12 dummy variables
B1=(R==1); B2=(R==2); B3=(R==3);
B4=(R==4); B5=(R==5); B6=(R==6);
B7=(R==7); B8=(R==8); B9=(R==9);
B10=(R==10); B11=(R==11); B12=(R==12);

h=1;
syhat=zeros(T-h-T0+1, 1);
ytph=y(T0+h:end);%observed y_{t+h}
for t=T0:T-h
    yt=y(1:t);
    yt=yt(h+1:t, :);
    D1t=D1(1:t); D2t=D2(1:t);
    D3t=D3(1:t); D4t=D4(1:t);
    gdp=lagmatrix(g(1:t),h);
    p1=lagmatrix(p(1:t),h);
    Xt=[lagmatrix((1:t),h) D1t D2t D3t D4t gdp p1];
    Xt=Xt(h+1:t, :);
    beta=(Xt'*Xt)\(Xt'*yt);
    yhat=[t+h D1(t+h) D2(t+h) D3(t+h) D4(t+h) gdp(t) p1(t)]*
        beta;
    syhat(t-T0+1)=yhat;
end

MSFE1=mean((ytph-syhat).^2)
MSFE_RW= 8.949053318210024e+03;
RelMSE=MSFE1/MSFE_RW

plot(T0+h:T,ytph,T0+h:T,syhat)
```

e) Code 5: Holt-Winters smoothing

```
data=xlsread('data');
y=data(:,1);

T=length(data);
t=(1:T)';

T0 = 50; h = 1; s=12;
syhat = zeros(T-h-T0+1,1);
```

```
ytph = y(T0+h:end); % observed y {t+h}
alpha = .2; beta = .2; gamma=.2;% smoothing parameters
St=zeros(T-h,1);
Lt = mean(y(1:s)); bt = 0; St(1:12)=y(1:s)/Lt
for t = s+1:T-h
newLt = alpha*(y(t)-St(t-s)) + (1-alpha)*(Lt+bt);
newbt = beta*(newLt-Lt) + (1-beta)*bt;
St(t)=gamma*(y(t)-newLt) + (1-gamma)*St(t-s);
yhat = newLt + h*newbt + St(t+h-s);
Lt = newLt; bt = newbt; % update Lt and bt
if t>= T0 % store the forecasts for t >= T0
syhat(t-T0+1,:) = yhat;
end
end

MSFE1 = mean((ytph-syhat).^2)
MSFE_RW= 8.949053318210024e+03;
RelMSE=MSFE1/MSFE_RW

plot(T0+h:T,ytph,T0+h:T,syhat)
```

f) Code 6: AR5 process

```
data=xlsread('labor','Data1');
data=data(6:end,3);

m=6;
data0=data(1:m);
data=data(m+1:end);
T=length(data);
T0=50; h=1;
data_AR5=zeros(T-h-T0+1,1);

for t=T0:T-h
    datat=data(h:t);
    yt=[[data0(m);data(1:t-h)] [data0(m-1:end);data(1:t-h-1)]
        [data0(m-2:end);data(1:t-h-2)]...
          [data0(m-3:end);data(1:t-h-3)] [data0(m-4:end);data(1:
              t-h-4)]];
    betahat5 =(yt'*yt)\(yt'*datat);
    data_AR5(t-T0+1,:)=[data(t) data(t-1) data(t-2) data(t-3)
        data(t-4)]*betahat5;
end
```

```
data_obs=data(T0+h:end);

MSFE_AR5=mean((data_obs-data_AR5).^2)
MSFE_RW= 8.949053318210024e+03;
RelMSE=MSFE_AR5/MSFE_RW

plot(T0+h:T,data_AR5,T0+h:T,data_obs);
```

g) Code 7: AR3 process on first diff

```
clear all
clc

data=xlsread('labor','Data1');
data=data(6:end,3);
T=length(data);

Newdata=data(2:end)-data(1:end-1);

m=4;
Newdata0=Newdata(1:m);
Newdata= Newdata(m+1:end);
T=length(Newdata);
T0=50;h=1;
Newdata_AR=zeros(T-h-T0+1,1);
for t=T0:T-h
    Newdatat=Newdata(h:t);
    yt=[[Newdata0(m);Newdata(1:t-h)] [Newdata0(m-1:end);
        Newdata(1:t-h-1)]...
          [Newdata0(m-2:end);Newdata(1:t-h-2)]];
    betahat1=(yt'*yt)\(yt'*Newdatat);
    Newdata_AR(t-T0+1,:)=[Newdata(t) Newdata(t-1) Newdata(t-2)
        ]*betahat1;
end

dataobs=Newdata(T0+h:end);
MSFE_AR=mean((dataobs-Newdata_AR).^2)
MSFE_RW= 8.949053318210024e+03;
RelMSE=MSFE_AR/MSFE_RW
```

h) Code 8: Density forecasts

```
sigmahat=(1/(length(data)-1))*sum((dataobs-Newdata_AR).^2); %
    where dataobs are the actual observations, Newdata_AR are
```

```
the forecasts using an AR3 process on the first difference

IntForecast1=[Newdata_AR1-1.645*sqrt(sigmahat) Newdata_AR1
    +1.645*sqrt(sigmahat) ] %where Newdata_AR1 is the point
    forecast for a particular month
```