EMET8012 Project

Chienhsiang Yeh — U6211003

October 21, 2018

1 Data Description and Examination

The variables I used for forecasting is "Index of Industrial Production" (INDPRO) which is the production index measures real output. It is expressed as a percentage of real output in a base year, currently 2012. This data releases at every mid month and are obtained from FRED (https://fred.stlouisfed.org/series/INDPRO). The original source of data is the Federal Reserve System's website (https://www.federalreserve.gov/releases/g17/Current/default.htm). This project intends to forecast the Index of Industrial Production in September. The data will be released at October 16 at the of the Federal Reserve System's website and can be downloaded from FRED.
Moreover, I consider some explanatory variables including Total Capacity Utilization(TCU), Civilian Unemployment Rate (UNRATE) and Consumer Price Index for All Urban Consumers: All Items (CPIAUCSL). TCU is released together with INDPRO and it estimates the sustainable potential output expressed as a percentage of actual output in 2012. These explanatory variables are also released every monthly.
Figure 1 shows the times series of above variables. We can know from the figure that y(INDPRO) has some co-movement with TCU and CPIAUCSL and negatively correlated with UNRATE. This is reasonable under macroeconomic intuition. More, series y is relatively smooth at first look. The reason behind may be that INDPRO is seasonal adjusted.
Furthermore, figure 2 shows the covariance of y is gradually but not significantly decreasing with lags. Thus, INDPRO may be nonstationary. I further examine the first difference($\Delta y = y_t - y_{t-1}$) and its covariance. From figure 3, I conclude that the first difference of y is stationary. Therefore, INDPRO may be an IMA(1) or ARIMA(p,1,q) process.
Since INDPRO is differencing stationary and the covariance is slightly decreasing with lags, I consider Holt-Winters smoothing, IMA(1), IAR(1), AR(p), ARMA(p,q) and VAR models in this project. I do not consider model with season dummies because INDPRO is seasonally adjusted.
Also, the time series I used are started from January 1981 to August 2018 ($T = 452$). I choose the forecasting T_0 around 120 so that there are 332 forecasters for calculating MSFE.

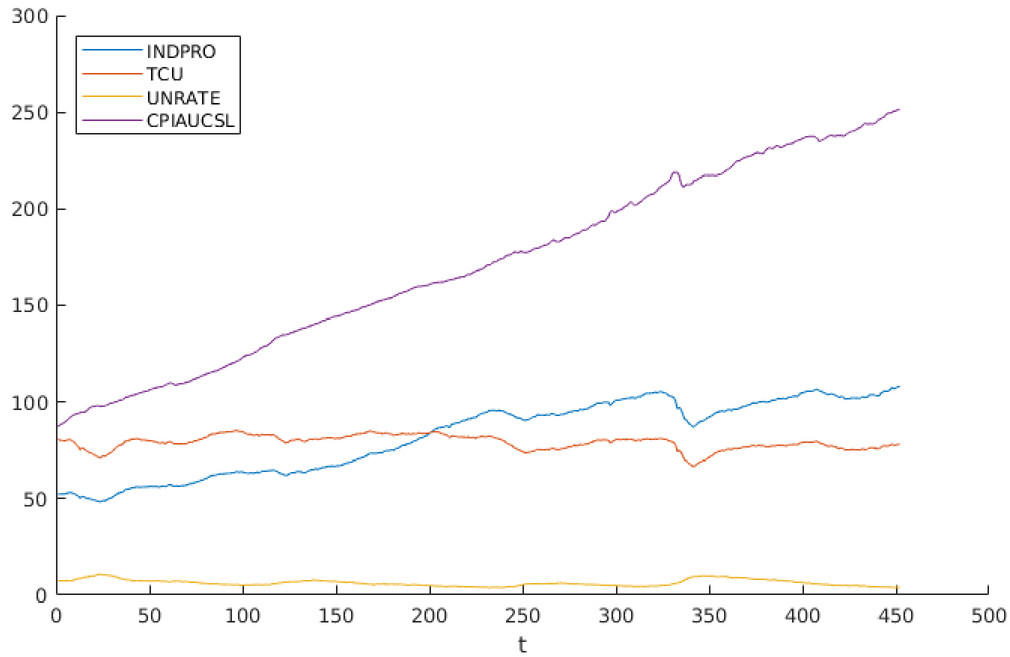

Figure 1: Time series of y and explanatory variables

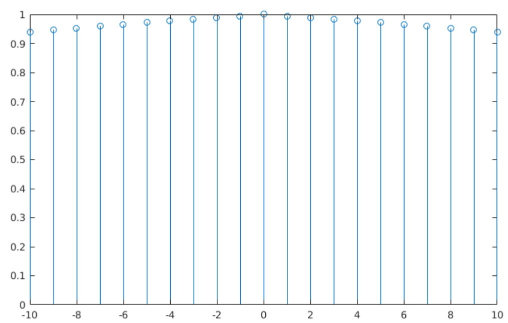

Figure 2: Covariance of y

2 Model Specification

2.1 Benchmark

I use the random walk as the benchmark model. The following table summaries the result. The MSFE is significantly small compared with the other models. Since y and also random walk $y_t = y_{t-1} + \epsilon$ are both difference stationary, random walk is a strong benchmark specification. Besides, y is seasonal adjusted, this also makes $\hat{y}_{t+1} = \mathbb{E}[y_{t+1}|y_t] = y_t$ be a good forecast.

model	AIC	BIC	MSFE
random walk	144.9271	157.2682	0.3652

Table 1: Random Walk

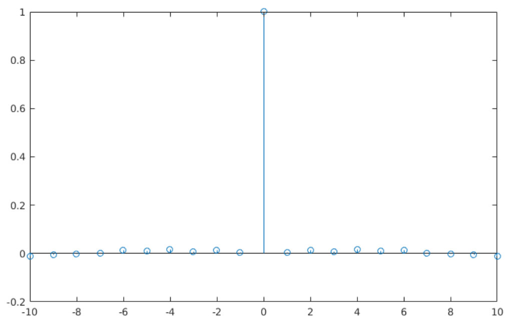

Figure 3: Covariance of first difference of y

2.2 Holt-Winters Smoothing

Since the series y does not have significant cycle in figure 1, I use Holt-Winters smoothing algorithm without seasonality. The algorithm simply follows the lecture note. By try and error, I get the best parameter are $\alpha = 0.6$ and $\beta = 0.5$. This specification is better than the benchmark. Since Holt-Winter algorithm weighted the previous level and slope, it can capture the trend and behaves like AR(1). I think this is why Holt-Winters smoothing is better than the benchmark. However, it only offers point forecasts.

	Holt-Winters	Random Walk
MSFE	0.3173	0.3652

Table 2: Holt-Winters Smoothing

2.3 Simple OLS

I consider the following independent variables in the OLS: constant, trend, TCU, UNRATE, CPIAUCSL. I find some best models are:

1. y regress on constant, trend, UNRATE,
2. y regress onconstant, trend, TCU, UNRATE,
3. y regress onconstant, trend, TCU, UNRATE, CPIAUCSL

The following summarizes the result of MSFE. These models are all worse than the benchmark. Nevertheless, we may include them in AR model to improve the forecast.

	model 1	model 2	model3	Random Walk
MSFE	28.6174	29.1927	26.9512	0.3652

Table 3: Simple OLS

2.4 Extracting Factors

I use the 80 monthly data obtained from FRED to extract factors. To begin, I consider the simple OLS with k factors and without lags.

$$y_t = a_0 + a_1 t + \beta_1 \hat{f}_{1,t-1} + \beta_2 \hat{f}_{2,t-1} + \beta_3 \hat{f}_{3,t-1} + ... + \beta_k \hat{f}_{k,t-1} + u_t$$

	$k=2$	$k=3$	$k=5$	$k=10$	$k=20$	$k=30$	$k=50$	Random Walk
MSFE	19.1891	17.6510	14.2600	5.1093	2.2331	0.7962	0.5724	0.3652

Table 4: Extracting Factors

Table 4 shows that the MSFE is comparable with the benchmark as $k \geq 30$ and generally better than the model in section 2.3. Furthermore, I consider the above model with AR(1):

$$y_t = a_0 + a_1 y_{t-1} + \beta_1 \hat{f}_{1,t-1} + \beta_2 \hat{f}_{2,t-1} + \beta_3 \hat{f}_{3,t-1} + ... + \beta_k \hat{f}_{k,t-1} + u_t$$

The MSFE is much smaller and comparable to the benchmark and AR(1). However, only the

	$k=1$	$k=2$	$k=3$	$k=5$	$k=10$	$k=15$	AR(1)	Random Walk
MSFE	0.3721	0.3769	0.3768	0.3980	0.3104	0.3459	0.3512	0.3652

Table 5: Extracting Factors

model with 10 or above factors is better than random walk and AR(1). When k is small, the factors even make the AR(1) worse (see the next section for AR(1)). I also try to include the lag on factors, but it even decreases the MSFE. Consider ARX(3) with factors:

$$y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \beta_1 \hat{f}_{1,t-1} + \beta_2 \hat{f}_{2,t-1} + \beta_3 \hat{f}_{3,t-1} + ... + \beta_k \hat{f}_{k,t-1} + u_t$$

Comparing the benchmark, AR(3) is a good model. If I add the forecast into AR(3), the

	$k=1$	$k=2$	$k=3$	$k=5$	$k=10$	$k=15$	AR(3)	Random Walk
MSFE	0.3555	0.3566	0.3585	0.3927	0.3286	0.3630	0.3319	0.3652

Table 6: Extracting Factors

forecasting is worse (table 6) even though they are better than the benchmark. It seems that the factors are more irrelevant to y than y_{t-1} so that it increases the MSFE. Since most economic data are quarterly, I only includes 10 categories and 80 data series. This database may be insufficient to capture the economic factor to explain the independent variable y.

2.5 AR(p)

Since the covariance of y is gradually decreasing, I further consider AR and ARMA. Here, i first consider AR(1),..., AR(4), ARX(1), ARX(2) and AR(1) with factors. I use UNRATE for ARX(1) and ARX(2), that is, with UNRATE regressor in AR model. I also consider ARX(1) with UNRATE, TCU and CPIAUCSL, but it has the least MSFE when there is only one explanatory variable X = UNRATE in the model. In table 7, ARF(1) and ARF(3) are AR(1) and AR(3) respectively with one factor in the right hand side of equation. These ARF specifications are as same as those in section 2.4.
In these specifications, AR(4) is the best. Besides, including the factor or other explanatory makes the MSFE larger. While, most of them are better than the benchmark.
Since AR(4) has the smallest MSFE value, I further examine it for forecasting.

$$y_{T+1} = 0.1513 + 1.1064 y_T + 0.0809 y_{T-1} + 0.0514 y_{T-2} - 0.2397 y_{T-3}, \ \ \sigma^2 = 0.2504$$

	AR(1)	AR(2)	AR(3)	AR(4)	Random Walk
MSFE	0.3512	0.3442	0.3319	0.3176	0.3652
	ARX(1)	ARX(2)	ARF(1)	ARF(3)	ARF(4)
MSFE	0.3551	0.3473	0.3721	0.3555	0.3353

Table 7: Extracting Factors

	MSFE	MSE	AIC	BIC
AR(4)	0.3176	0.2504	122.4221	142.6957
random walk	0.3652	0.3074	144.9271	157.2682

Table 8: AIC, BIC for AR(4)

2.6 IMA(1)

Series u is difference stationary, so I consider IMA(1): $\Delta y_t = u_t + u_{t-1}$. The result is slightly better than the benchmark. We know that the first difference is stationary, so the random walk and IMA(1) are all good specifications for forecasting. However, they are not as well as AR(4).

	IMA(1)	AR(4)	Random Walk
MSFE	0.3539	0.3176	0.3652

Table 9: Extracting Factors

2.7 IAR(1)

Next, I consider the IAR(1). I simply perform the AR(1) algorithm on Δy_t.

$$\Delta y_t = \mu + \psi \Delta y_{t-1} + u_t$$

The result shows that it improves the forecast significantly. IAR(1) is even better than AR(4).

	IAR(1)	AR(4)	Random Walk
MSFE	0.2291	0.3176	0.3652

Table 10: Extracting Factors

Further, MSE, AIC and BIC of the regression $\Delta y_t = \mu + \psi \Delta y_{t-1} + u_t$ can be obtained with $k = 2$. The following table summarizes these indicators which are all smaller than those of benchmark.

	MSFE	MSE	AIC	BIC
IAR(1)	0.2291	0.2797	129.8728	138.0912
random walk	0.3652	0.3074	144.9271	157.2682

Table 11: AIC, BIC for IAR(1)

IAR(1) can also be written as

$$y_t = \mu + \psi \Delta y_{t-1} + y_t + u_t = 0.0988 + 1.2086 y_t - 0.2086 y_{t-1} + u_t, \ \ \sigma^2 = 0.2797$$

2.8 ARMA(p,q)

For ARMA, I consider ARMA(1,1), ARMA(2,1), ARMA(3,1), ARMA(3,1), ARMA(1,2) and ARMA(2,2). In these models, ARMA(2,2) and ARMA(3,1) have the best forecasts. Most of them beats the benchmark.

ARMA(1,1)	ARMA(2,1)	ARMA(3,1)	ARMA(4,1)	ARMA(1,2)	ARMA(2,2)	R.W.
0.3451	0.3121	0.3097	0.4167	0.3429	0.3063	0.3652

Table 12: Extracting Factors

I further use ARMA(2,1), ARMA(3,1) and ARMA(2,2) to forecast.

- ARMA(2,1)

$$y_{T+1} = -0.0051 + 1.8998 y_T + -0.8996 y_{T-1} + u_{T+1} + -0.7165 u_T, \quad \sigma^2 = 0.2523$$

- ARMA(3,1)

$$y_{T+1} = 0.0050 + 0.6019 y_T + 0.6369 y_{T-1} + -0.2373 y_{T-2} + u_{T+1} + 0.5953 u_T, \quad \sigma^2 = 0.2766$$

- ARMA(2,2)

$$y_{T+1} = 0.0041 + 0.4919 y_T + 0.5101 y_{T-1} + u_{T+1} + 0.6627 u_T + 0.1493 u_{T-1}, \quad \sigma^2 = 0.2816$$

	MSFE	MSE	AIC	BIC
ARMA(2,1)	0.3121	0.2523	121.5314	137.9684
ARMA(3,1)	0.3097	0.2762	134.0231	154.5582
ARMA(2,2)	0.3063	0.2816	136.7115	157.2577
random walk	0.3652	0.3074	144.9271	157.2682

Table 13: AIC, BIC for ARMA

Table 13 shows that ARMA(2,1) is better than the others under AIC and BIC. The parsimonious trade-off makes it a nicer choice. I further try to include structural dummy about global financial crisis in AR(4) and ARMA(2,1), but they become slightly worse under MSFE criterion. Then, I stop here and avoid making the model too complicated.

2.9 VAR

Next, consider VAR(1) with n=2, 3 or 4 series.

$$\begin{aligned}
y_{1,t} &= b_1 + B_{11} y_{1,t-1} + B_{12} y_{2,t-1} + B_{13} y_{3,t-1} + B_{14} y_{4,t-1} + u_1 \\
y_{2,t} &= b_2 + B_{21} y_{2,t-1} + B_{22} y_{2,t-1} + B_{23} y_{3,t-1} + B_{24} y_{4,t-1} + u_2 \\
y_{3,t} &= b_3 + B_{31} y_{2,t-1} + B_{32} y_{2,t-1} + B_{33} y_{3,t-1} + B_{34} y_{4,t-1} + u_2 \\
y_{4,t} &= b_4 + B_{41} y_{2,t-1} + B_{42} y_{2,t-1} + B_{43} y_{3,t-1} + B_{44} y_{4,t-1} + u_2
\end{aligned}$$

where, $y_{1,t}$, $y_{2,t}$, $y_{3,t}$ and $y_{4,t}$ are INDPRO, TCU, UNRATE and CPIAUCSL, respectively. When n=2, $\text{VAR}_2(1)$ only includes INDPRO and TCU.
Table 14 illustrates that these VAR(1) models are all worse than the random walk, although they do better than those in section 2.1. These models can be also used as benchmarks.

	$VAR_2(1)$	$VAR_3(1)$	$VAR_4(1)$	R.W.
MSFE	3.3036	2.1959	2.2108	0.3652

Table 14: Extracting Factors

3 Conclusion

1. INDPRO series is difference stationary, so the random walk is a good specification.
2. All of the AR(p), ARMA(p,q), IMA(1) or IAR(1) I considered are all better than the $VAR_n(1)$. Since series y is difference stationary, including a lag can improve the forecast significantly. If I try to add additional explanatory, it usually aggravates the forecast. This is also the case of ARX(1) or ARF(1).
3. Among the discussed models, IAR(1), AR(4), ARMA(2,1), ARMA(3,1) and ARMA(2,2) are the best specification compared to Holt-Winters smoothing algorithm under the loss function of MSFE.
4. MSFE selects IAR(1) for the best forecasting model, while both AIC and BIC select ARMA(2,1) under parsimonious criterion.

	MSFE	MSE	AIC	BIC
random walk	0.3652	0.3074	144.9271	157.2682
AR(4)	0.3176	0.2504	122.4221	142.6957
IAR(1)	0.2291	0.2797	129.8728	138.0912
ARMA(2,1)	0.3121	0.2523	121.5314	137.9684
ARMA(3,1)	0.3097	0.2762	134.0231	154.5582
ARMA(2,2)	0.3063	0.2816	136.7115	157.2577

Table 15: MSFE,AIC, BIC summary

5. The factor methodology is worse than I expected. In ARX(p) model, using one explanatory variable like UNRATE or TCU is better than using the factors.
6. Next period's confidence intervals of prediction, $\hat{y}_{T+1} \pm 1.96\hat{\sigma}$ at 5% significance level:

	IAR(1)	AR(4)	ARMA(2,1)
σ^2	0.2797	0.2504	0.2523
confidence interval	(107.3866,109.4598)	(107.5584,109.5200)	(107.5386,109.5076)
	ARMA(3,1)	ARMA(2,2)	
σ^2	0.2766	0.2816	
confidence interval	(107.3235, 109.3873)	(107.3282, 109.4084)	

Table 16: Confidence interval

7. Point forecast for the following year given $y_T = 108.2317$, $y_{T-1} = 107.7875$, $y_{T-1} = 107.3999$, $y_{T-1} = 106.7486$. ARMA(2,1) has the most optimistic forecast of y_{T+12}, while IAR(1) is more conservative (see Table 17).

Month	IAR(1)	AR(4)	ARMA(2,1)	ARMA(3,1)	ARMA(2,2)
Sep 2018	108.4232	108.5392	108.5231	108.3565	108.3683
Oct 2018	108.5619	108.7792	108.8012	108.5824	108.5448
Nov 2018	108.6897	108.9994	109.0675	108.6924	108.6786
Dec 2018	108.8152	109.1718	109.3232	108.8729	108.8345
Jan 2019	108.9402	109.3189	109.5694	108.9980	108.9795
Feb 2019	109.0651	109.4494	109.8071	109.1621	109.1303
Mar 2019	109.1899	109.5617	110.0372	109.2978	109.2784
Apr 2019	109.3148	109.6628	110.2605	109.4543	109.4282
May 2019	109.4397	109.75527	110.4777	109.5960	109.5774
Jun 2019	109.5646	109.8400	110.6895	109.7487	109.7273
Jul 2019	109.6894	109.9196	110.8965	109.8938	109.8771
Aug 2019	109.8143	109.9951	111.0992	110.0447	110.0272

Table 17: Forecast the following year

4 Appendix

4.1 Database

Production and Capacity Industrial Production Index (INDPRO)
Industrial Production: Manufacturing (NAICS) (IPMAN)
Industrial Production: Mining (IPMINE)
Industrial Production: Final Products (Market Group) (IPFINAL)
Industrial Production: Manufacturing (SIC) (IPMANSICN)
Industrial Production: Consumer Goods (IPCONGD)
Industrial Production: Durable Manufacturing (NAICS) (IPDMAN)
Industrial Production: Nondurable Manufacturing (NAICS) (IPNMAN)
Industrial Production: Business Equipment (IPBUSEQ)
Industrial Production: Construction supplies (IPB54100S)
Industrial Production: Nondurable Consumer Goods (IPNCONGD)
Industrial Production: Durable Consumer Goods (IPDCONGD)
Industrial Production: Materials (IPMAT)
Industrial Production: Nondurable Materials (IPNMAT)
Industrial Production:Computers,communications equipment, and semiconductors(IPHITEK2S)
Industrial Production: Final Products and Nonindustrial Supplies (IPFPNSS)
Industrial Production: Construction supplies (IPB54100S)
Total Capacity Utilization(TCU)
Capacity Utilization: Manufacturing (SIC) (CUMFNS)
Capacity Utilization: Mining (CAPUTLG21S)
Capacity Utilization: Electric and gas utilities (CAPUTLG2211A2S)
Capacity Utilization: Manufacturing (NAICS) (MCUMFN)
Capacity Utilization: Durable manufacturing (CAPUTLGMFDS)
Capacity Utilization: Nondurable manufacturing (CAPUTLGMFNS)
Capacity Utilization: Computers, communications equipment, and semiconductors (CAPUTL-HITEK2S)
Price
Consumer Price Index for All Urban Consumers: All Items (CPIAUCSL)
Consumer Price Index for All Urban Consumers: All Items Less Food and Energy (CPILFESL)

Consumer Price Index for All Urban Consumers: Rent of primary residence (CUUR0000SEHA)
Consumer Price Index for All Urban Consumers: Food and Beverages (CPIFABSL)
Producer Price Index for All Commodities (PPIACO)
Domestic Capital Account
Personal Income (PI)
Personal Consumption Expenditures (PCE)
Personal Saving (PMSAVE)
Disposable Personal Income (DSPI)
Personal Consumption Expenditures: Chain-type Price Index (PCEPI)
Personal Consumption Expenditures: Durable Goods (PCEDG)
Employment
Unemployment Level (UNEMPLOY)
Civilian Unemployment Rate (UNRATE)
Unemployment Rate: 20 years and over (LNS14000024)
Unemployment Rate: Aged 15-64: All Persons for the United States (LRUN64TTUSM156S)
Civilian Labor Force (CLF16OV)
Civilian Labor Force Participation Rate (CIVPART)
All Employees: Total Nonfarm Payrolls (PAYEMS)
Civilian Employment Level (CE16OV)
Civilian Employment-Population Ratio (EMRATIO)
Employment Population Ratio: 25 - 54 years (LNS12300060)
Employment Level: Part-Time for Economic Reasons, All Industries (LNS12032194)
Exchange rate
Real Trade Weighted U.S. Dollar Index: Broad (TWEXBPA)
Trade Weighted U.S. Dollar Index: Broad (TWEXBMTH)
Trade Weighted U.S. Dollar Index: Major Currencies (TWEXMMTH)
China / U.S. Foreign Exchange Rate (EXCHUS)
Japan / U.S. Foreign Exchange Rate (EXJPUS)
U.S. / U.K. Foreign Exchange Rate (EXUSUK)
U.S. / Australia Foreign Exchange Rate (EXUSAL)
Interest rate
Effective Federal Funds Rate (FEDFUNDS)
1-Year Treasury Constant Maturity Rate (GS1)
2-Year Treasury Constant Maturity Rate (GS2)
5-Year Treasury Constant Maturity Rate (GS5)
10-Year Treasury Constant Maturity Rate (GS10)
3-Month Treasury Bill: Secondary Market Rate (TB3MS)
10-Year Treasury Constant Maturity Minus 2-Year Treasury Constant Maturity (T10Y2YM)
Moody's Seasoned Aaa Corporate Bond Yield (AAA)
Federal Government Debt
Federal Surplus or Deficit [-] (MTSDS133FMS)
Market Value of Gross Federal Debt (MVGFD027MNFRBDAL)
Market Value of Marketable Treasury Debt (MVMTD027MNFRBDAL)
Monetary
M1 Money Stock (M1SL)
M2 Money Stock (M2SL)
Monetary Base; Total (BOGMBASE)
Currency Component of M1 (CURRSL)
St. Lou Total Checkable Deposits (TCDSL)is Adjusted Monetary Base (AMBSL)
Banking
Total Assets, All Commercial Banks (TLAACBM027SBOG)

Total Consumer Credit Owned and Securitized, Outstanding (TOTALSL)
Commercial and Industrial Loans, All Commercial Banks (BUSLOANS)
Stock
S&P 500 (∧GSPC)
NASDAQ Composite (∧IXIC)
NYSE COMPOSITE (DJ) (∧NYA)
Else
Total Vehicle Sales (TOTALSA)
Spot Crude Oil Price: West Texas Intermediate (WTI) (WTISPLC)
New Private Housing Units Authorized by Building Permits (PERMIT)
Housing Starts: Total: New Privately Owned Housing Units Started (HOUST)
New Privately-Owned Housing Units Completed: Total (COMPUTSA)

4.2 Code

4.2.1 Extracting Factors, AR, IMA, IAR, ARMA

```
%%
load 'INDPRO.csv'; % industrial production index  2012=100
load 'TCU.csv';% x1 tocal capacity utilization
load 'UNRATE.csv' % x2 unemployment rate
load 'PPIACO.csv';% x3 producer price index  1982=100
load 'CPIAUCSL.csv'; % x4 CPI for urban consumer
y = INDPRO(:,2);
x1 = TCU(:,2); x2 = UNRATE(:,2); x3 = PPIACO(:,2); x4=CPIAUCSL(:,2);
T = length(y);
T0 =120;
t=(1:T)';
%% covariance
[cov_y,lags_y] = xcov(y,10,'coeff');
stem(lags_y,cov_y);
autocorr(y);
L = spdiags(ones(T-1,1),-1,T,T);
I = speye(T);
D = I - L;
[cov_y,lags_y] = xcov(D*y,10,'coeff');  % Dy is covariance stationary.
stem(lags_y,cov_y);

[cov_x1,lags_x1] = xcov(x1,10,'coeff');
stem(lags_x1,cov_x1);

%% graphs
hold on
plot(y);
plot(x1); plot(x2); plot(x4);

legend({'INDPRO','TCU', 'UNRATE','CPIAUCSL'}, 'Location','northwest');
xlabel('t')
hold off


%% Benchmark: random walk
h=1; y = INDPRO(:,2);T = length(y);
ytph = y(T0+h:end);
syhat = y(T0: T-h);
MSE_rw = mean((y(2:end)-y(1:T-1)).^2);
AIC_rw = MSE_rw*T + k*2;
BIC_rw = MSE_rw*T + k*log(T);
MSFE_rw = mean((ytph - syhat).^2);
```

```
%% Holt-Winters smoothing   one step
y = INDPRO(:,2);T = length(y);
alpha = 0.6; beta = 0.5;
ytph = y(T0+h:end);
syhat = zeros(T-h-T0+1, 1);
hold on
plot(y);
Lt = y(1); bt = y(2) - y(1);
for t = 2: T-h
% Lt = alpha yt +(1-alpha)(L_t-1 + b_t-1)
newLt = alpha*y(t) + (1 - alpha)*(Lt+bt);
% bt = beta(Lt - L_t-1) + (1-beta)b_t-1
newbt = beta*(newLt - Lt) + (1 - beta)*bt;
% yhat_t+h = Lt + h bt
yhat = newLt + h*newbt ;
Lt = newLt; bt = newbt; % update Lt and bt
if t>= T0 % store the forecasts for t >= T0
syhat(t-T0+1,:) = yhat;
end
end
time = [T0+h:T]';
plot(time, syhat);
MSFE_HW = mean((ytph - syhat).^2);
hold off

%% X: constant, trend, unemployment rate
ytph = y(T0+h:end);
syhat = zeros(T-h-T0+1, 1);
hold on
plot(y)
for i = T0:T-h
yt = y(1:i);
%D_gfc_t = D_gfc(1:i);
%D_dot_t = D_dot(1:i);
Xt = [ones(i,1) (1:i)' x2(1:i)];
beta = (Xt'*Xt)\ (Xt'*yt);
yhat = [1 i+h x2(i+h)] * beta;
syhat(i-T0+1) = yhat;
end
plot(time, syhat)
hold off
MSFE_2 = mean((ytph - syhat).^2);

%% X: constant, trend, capacity
ytph = y(T0+h:end);
syhat = zeros(T-h-T0+1, 1);
hold on
plot(y)
for i = T0:T-h
yt = y(1:i);
%D_gfc_t = D_gfc(1:i);
%D_dot_t = D_dot(1:i);
Xt = [ones(i,1) (1:i)' x1(1:i)];
beta = (Xt'*Xt)\ (Xt'*yt);
yhat = [1 i+h x1(i+h)] * beta;
syhat(i-T0+1) = yhat;
end
plot(time, syhat)
hold off
MSFE_3 = mean((ytph - syhat).^2);

%% X: constant, trend, unemployment, capacity
```

```
ytph = y(T0+h:end);
syhat = zeros(T-h-T0+1, 1);
hold on
plot(y)
for i = T0:T-h
yt = y(1:i);
%D_gfc_t = D_gfc(1:i);
%D_dot_t = D_dot(1:i);
Xt = [ones(i,1) (1:i)' x1(1:i) x2(1:i)];
beta = (Xt'*Xt)\ (Xt'*yt);
yhat = [1 i+h x1(i+h) x2(i+h)] * beta;
syhat(i-T0+1) = yhat;
end
plot(time, syhat)
hold off
MSFE_4 = mean((ytph - syhat).^2);

%% X: constant, trend, unemployment, capacity, CPI
ytph = y(T0+h:end);
syhat = zeros(T-h-T0+1, 1);
hold on
plot(y)
for i = T0:T-h
yt = y(1:i);
%D_gfc_t = D_gfc(1:i);
%D_dot_t = D_dot(1:i);
Xt = [ones(i,1) (1:i)' x1(1:i) x2(1:i) x4(1:i)];
beta = (Xt'*Xt)\ (Xt'*yt);
yhat = [1 i+h x1(i+h) x2(i+h) x4(i+h)] * beta;
syhat(i-T0+1) = yhat;
end
plot(time, syhat)
hold off
MSFE_5 = mean((ytph - syhat).^2);

%% Extract factor without lags
load 'database.csv'
%database = readtable('database.csv');
y = INDPRO(:,2); T = length(y);
k=10; % number of factors
%data=table2array(database);
X = database; N = size(X,2);
[Lambda, D] = eigs(X'*X, k); %extract the largest k eigenvalues
fhat = X*Lambda./N;
Xt=[ones(T,1) fhat];
beta = (Xt'*Xt)\ (Xt'*y);
yhat_fac = [ones(T,1) fhat(:,:)] * beta;
MSE_fac = mean((y-yhat_fac).^2);

hold on
plot(y);
m=4;
T0 =120-m; h=1;
y0 = y(1:m); y = y(m+1:end); T = length(y);
ytph = y(T0+h:end);
syhat = zeros(T-h-T0+1, 1);
for i = T0:T-h
X = database(1:i+m,:);
[Lambda, D] = eigs(X'*X, k); %extract the largest k eigenvalues
fhat = X*Lambda./N;
yt = y(h:i);
fhat0 = fhat(1:m,:); fhat = fhat(m+1:end,:);
zt = [ones(i-h+1,1) (1:i)' [fhat0(m,:);fhat(1:i-h,:)]];
```

```
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 i+h fhat(i,:)] * beta;
syhat(i-T0+1) = yhat;
end
MSFE_factor = mean((ytph - syhat).^2) %332
plot((T0:T-h)', syhat);
hold off
%% Factor + AR(1)
k=15;
m=4;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
hold on
plot(y);
T0=120-4 ;h=1;
ytph = y(T0+h: end);
yhatAR = zeros(T-h-T0+1,1);
for i = T0:T-h
X = database(1:i+m,:);
[Lambda, D] = eigs(X'*X, k); %extract the largest k eigenvalues
fhat = X*Lambda./N;
yt = y(h:i);
fhat0 = fhat(1:m,:); fhat = fhat(m+1:end,:);
zt = [ones(i-h+1,1) [y0(m);y(1:i-h)] [fhat0(m,:);fhat(1:i-h,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) fhat(i,:)] * beta;
yhatAR(i-T0+1) = yhat;
end
MSFE_AR_factor = mean((ytph - yhatAR).^2)
plot((T0:T-h)', yhatAR);
hold off
%% Factor with one lag+ AR(1)
k=30;
m=4;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
hold on
plot(y);
T0=120-4 ;h=1;
ytph = y(T0+h: end);
yhatAR = zeros(T-h-T0+1,1);
for i = T0:T-h
X = database(1:i+m,:);
[Lambda, D] = eigs(X'*X, k); %extract the largest k eigenvalues
fhat = X*Lambda./N;
yt = y(h:i);
fhat0 = fhat(1:m,:); fhat = fhat(m+1:end,:);
zt = [ones(i-h+1,1) [y0(m);y(1:i-h)]  [fhat0(m,:);fhat(1:i-h,:)] ...
[fhat0(m-1:end,:);fhat(1:i-h-1,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) fhat(i,:) fhat(i-1,:)] * beta;
yhatAR(i-T0+1) = yhat;
end
MSFE_AR_factor1 = mean((ytph - yhatAR).^2)
plot((T0:T-h)', yhatAR);
hold off
%% Factor + AR(2)
k=1;
m=4;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);

T0=120-4 ;h=1;
```

```
ytph = y(T0+h:  end);
yhatAR = zeros(T−h−T0+1,1);
for i = T0:T−h
X = database(1:i+m,:);
[Lambda, D] = eigs(X'*X, k); %extract the largest k eigenvalues
fhat = X*Lambda./N;
yt = y(h:i);
fhat0 = fhat(1:m,:); fhat = fhat(m+1:end,:);
zt = [ones(i−h+1,1) [y0(m);y(1:i−h)] [y0(m−1:end);y(1:i−h−1)]...
 [fhat0(m,:);fhat(1:i−h,:)]];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) y(i−1) fhat(i,:)] * beta;
yhatAR(i−T0+1) = yhat;
end
MSFE_AR2_factor = mean((ytph − yhatAR).^2)

%% Factor + AR(3)
k=15;
m=4;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);

T0=120−4 ;h=1;
ytph = y(T0+h:  end);
yhatAR = zeros(T−h−T0+1,1);
for i = T0:T−h
X = database(1:i+m,:);
[Lambda, D] = eigs(X'*X, k); %extract the largest k eigenvalues
fhat = X*Lambda./N;
yt = y(h:i);
fhat0 = fhat(1:m,:); fhat = fhat(m+1:end,:);
zt = [ones(i−h+1,1) [y0(m);y(1:i−h)] [y0(m−1:end);y(1:i−h−1)]...
 [y0(m−2:end);y(1:i−h−2)] [fhat0(m,:);fhat(1:i−h,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) y(i−1) y(i−2) fhat(i,:)] * beta;
yhatAR(i−T0+1) = yhat;
end
MSFE_AR3_factor = mean((ytph − yhatAR).^2)

%% Factor + AR(3)
k=1;
m=4;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);

T0=120−4 ;h=1;
ytph = y(T0+h:  end);
yhatAR = zeros(T−h−T0+1,1);
for i = T0:T−h
X = database(1:i+m,:);
[Lambda, D] = eigs(X'*X, k); %extract the largest k eigenvalues
fhat = X*Lambda./N;
yt = y(h:i);
fhat0 = fhat(1:m,:); fhat = fhat(m+1:end,:);
zt = [ones(i−h+1,1) [y0(m);y(1:i−h)] [y0(m−1:end);y(1:i−h−1)]...
 [y0(m−2:end);y(1:i−h−2)] [fhat0(m,:);fhat(1:i−h,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) y(i−1) y(i−2) fhat(i,:)] * beta;
yhatAR(i−T0+1) = yhat;
end
MSFE_AR3_factor = mean((ytph − yhatAR).^2);

%% Factor + AR(4)
```

```
k=1;
m=5;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);

T0=120-m ;h=1;
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
for i = T0:T-h
X = database(1:i+m,:);
[Lambda, D] = eigs(X'*X, k); %extract the largest k eigenvalues
fhat = X*Lambda./N;
yt = y(h:i);
fhat0 = fhat(1:m,:); fhat = fhat(m+1:end,:);
zt = [ones(i-h+1,1) [y0(m);y(1:i-h)] [y0(m-1:end);y(1:i-h-1)]...
 [y0(m-2:end);y(1:i-h-2)] [y0(m-3:end);y(1:i-h-3)]...
  [fhat0(m,:);fhat(1:i-h,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) y(i-1) y(i-2) y(i-3) fhat(i,:)] * beta;
yhatAR(i-T0+1) = yhat;
end
MSFE_AR4_factor = mean((ytph - yhatAR).^2);
%% AR(1) X: urate
m=4;
y = INDPRO(:,2); x2 = UNRATE(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
x20 = x2(1:m); x2 = x2(m+1:end);
T0=120-m ;h=1;
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
for i = T0:T-h
yt = y(h:i);
zt = [ones(i-h+1,1) [y0(m);y(1:i-h)] [x20(m,:);x2(1:i-h,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) x2(i)] * beta;
yhatAR(i-T0+1) = yhat;
end
MSFE_AR1_urate = mean((ytph - yhatAR).^2);
%% AR(1) X: TCU
m=4;
y = INDPRO(:,2); x2 = UNRATE(:,2); x1 = TCU(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
x20 = x2(1:m); x2 = x2(m+1:end); x10 = x1(1:m); x1 = x1(m+1:end);
T0=120-m ;h=1;
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
for i = T0:T-h
yt = y(h:i);
zt = [ones(i-h+1,1) [y0(m);y(1:i-h)] [x10(m,:);x1(1:i-h,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) x1(i)] * beta;
yhatAR(i-T0+1) = yhat;
end
MSFE_AR1_capacity = mean((ytph - yhatAR).^2);

%% AR(1) X: TCU UNRATE
m=4;
y = INDPRO(:,2); x2 = UNRATE(:,2); x1 = TCU(:,2); x3=CPIAUCSL(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
x20 = x2(1:m); x2 = x2(m+1:end); x10 = x1(1:m); x1 = x1(m+1:end);
x30 = x3(1:m); x3 = x3(m+1:end);
T0=120-m ;h=1;
ytph = y(T0+h:  end);
```

```
yhatAR = zeros(T-h-T0+1,1);
for i = T0:T-h
yt = y(h:i);
zt = [ones(i-h+1,1) [y0(m);y(1:i-h)] ...
[x10(m,:);x1(1:i-h,:)] [x20(m,:);x2(1:i-h,:)] [x30(m,:);x3(1:i-h,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) x1(i) x2(i) x3(i)] * beta;
yhatAR(i-T0+1) = yhat;
end
MSFE_AR1_X = mean((ytph - yhatAR).^2);

%%  AR(4) X: unrate
m=5;
y = INDPRO(:,2); x2 = UNRATE(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
x20 = x2(1:m); x2 = x2(m+1:end);
T0=120-m ;h=1;
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
for i = T0:T-h
yt = y(h:i);
zt = [ones(i-h+1,1) [y0(m);y(1:i-h)] ...
[y0(m-1:end);y(1:i-h-1)] [y0(m-2:end);y(1:i-h-2)] ...
[y0(m-3:end);y(1:i-h-3)] [x20(m,:);x2(1:i-h,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) y(i-1) y(i-2) y(i-3) x2(i)] * beta;
yhatAR(i-T0+1) = yhat;
end
MSFE_AR4_urate = mean((ytph - yhatAR).^2);

%%  AR(4) X: unrate, capacity

m=5;
y = INDPRO(:,2); x1 = TCU(:,2); x2 = UNRATE(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
x20 = x2(1:m); x2 = x2(m+1:end); x10 = x1(1:m); x1 = x1(m+1:end);
T0=120-m ;h=1;
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
for i = T0:T-h
yt = y(h:i);
zt = [ones(i-h+1,1) [y0(m);y(1:i-h)] [y0(m-1:end);y(1:i-h-1)]...
 [y0(m-2:end);y(1:i-h-2)] [y0(m-3:end);y(1:i-h-3)] ...
 [x10(m,:);x1(1:i-h,:)] [x20(m,:);x2(1:i-h,:)] ];
beta = (zt'*zt)\ (zt'*yt);
yhat = [1 y(i) y(i-1) y(i-2) y(i-3) x1(i) x2(i)] * beta;
yhatAR(i-T0+1) = yhat;
end
MSFE_AR4_urate_cpcty = mean((ytph - yhatAR).^2);


%% AR(1)
m=4;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
T0 = 120-4;
h=1;
for t = T0:T-h
yt = y(h:t);
zt = [ones(t-h+1,1) [y0(m); y(1:t-h)]];
betahat = (zt'*zt)\(zt'*yt);
```

```
yhatAR(t-T0+1,:)= [1 y(t) ]*betahat;
end
MSFE_AR1 = mean((ytph - yhatAR).^2);  %332

%% AR(1) + X: unrate
m=4;
y = INDPRO(:,2);x2 = UNRATE(:,2);
x20 = x2(1:m); x2 = x2(m+1:end);
y0 = y(1:m); y = y(m+1:end); T = length(y);
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
T0 = 120-4;
h=1;
for t = T0:T-h
yt = y(h:t);
zt = [ones(t-h+1,1) [y0(m); y(1:t-h)] [x20(m,:);x2(1:t-h,:)]];
betahat = (zt'*zt)\(zt'*yt);
yhatAR(t-T0+1,:)= [1 y(t) x2(t)]*betahat;
end
MSFE_AR1 = mean((ytph - yhatAR).^2);  %332
%% AR(2) + X: unrate
m=4;
y = INDPRO(:,2);x2 = UNRATE(:,2);
x20 = x2(1:m); x2 = x2(m+1:end);
y0 = y(1:m); y = y(m+1:end); T = length(y);
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
T0 = 120-4;
h=1;
for t = T0:T-h
yt = y(h:t);
zt = [ones(t-h+1,1) [y0(m); y(1:t-h)]...
 [y0(m-1:end); y(1:t-h-1)] [x20(m,:);x2(1:t-h,:)]];
betahat = (zt'*zt)\(zt'*yt);
yhatAR(t-T0+1,:)= [1 y(t) y(t-1) x2(t)]*betahat;
end
MSFE_ARX2 = mean((ytph - yhatAR).^2);  %332

%% AR(2)
m=4;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
T0 = 120-4;
h=1;
for t = T0:T-h
yt = y(h:t);
zt = [ones(t-h+1,1) [y0(m); y(1:t-h)] [y0(m-1:end); y(1:t-h-1)]];
betahat = (zt'*zt)\(zt'*yt);
yhatAR(t-T0+1,:)= [1 y(t) y(t-1)]*betahat;
end
MSFE_AR2 = mean((ytph - yhatAR).^2);  %332

%% AR(3)
m=4;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
T0 = 120-4;
h=1;
for t = T0:T-h
```

```
yt = y(h:t);
zt = [ones(t-h+1,1) [y0(m); y(1:t-h)] [y0(m-1:end); ...
y(1:t-h-1)] [y0(m-2:end); y(1:t-h-2)]];
betahat = (zt'*zt)\(zt'*yt);
yhatAR(t-T0+1,:)= [1 y(t) y(t-1) y(t-2)]*betahat;
end
MSFE_AR3 = mean((ytph - yhatAR).^2);  %332

%% AR(4)
m=5;
y = INDPRO(:,2);
y0 = y(1:m); y = y(m+1:end); T = length(y);
ytph = y(T0+h:  end);
yhatAR = zeros(T-h-T0+1,1);
T0 = 120-m;
h=1;
for t = T0:T-h
yt = y(h:t);
zt = [ones(t-h+1,1) [y0(m); y(1:t-h)] [y0(m-1:end); y(1:t-h-1)]...
 [y0(m-2:end); y(1:t-h-2)] [y0(m-3:end); y(1:t-h-3)]];
betahat = (zt'*zt)\(zt'*yt);
yhatAR(t-T0+1,:)= [1 y(t) y(t-1) y(t-2) y(t-3)]*betahat;
end
MSFE_AR4 = mean((ytph - yhatAR).^2);  %332

y = INDPRO(:,2); T=length(y);
Z = [ones(T-4,1) y(4:T-1) y(3:T-2) y(2:T-3) y(1:T-4)];
betahat = (Z'*Z)\(Z'*y(5:end));
yhatAR4= Z*betahat;
MSE_AR4 = mean((yhatAR4-y(5:end)).^2);
k=5;
AIC_AR4 = MSE_AR4*(T-3) + k*2;
BIC_AR4 = MSE_AR4*(T-4)+ k*log(T-4);

yhatAR4_T=zeros(12,1);
yhatAR4_T(1) = [1 y(end) y(end-1) y(end-2) y(end-3)]*betahat;
yhatAR4_T(2) = [1 yhatAR4_T(1) y(end) y(end-1) y(end-2)]*betahat;
yhatAR4_T(3) = [1 yhatAR4_T(2) yhatAR4_T(1) y(end) y(end-1)]*betahat;
yhatAR4_T(4) = [1 yhatAR4_T(3) yhatAR4_T(2) yhatAR4_T(1) y(end)]*betahat;
for i =5:12
yhatAR4_T(i) = [1 yhatAR4_T(i-1) yhatAR4_T(i-2)...
 yhatAR4_T(i-3) yhatAR4_T(i-4)]*betahat;
end
%% IMA(1)
h = 1; m = 4; y = INDPRO(:,2);
dely = y(m+h:end) - y(m:end-h);
y0 = y(1:m); y = y(m+1:end);T = length(y);
T0 = 120-m ;
yhatMA = zeros(T-h-T0+1,1);
ytph = y(T0+h:end);
f = @(z) loglike_MA1(z, dely(1:T0));
thetaMLE = fminsearch(f, [.5,.5]);
% guess [sig, psi] = [.5 .5], thetaMLE = [sig psi]

for t = T0: T-h
delyt = dely(1:t);
% find the MLE
f = @(z) loglike_MA1(z, delyt);
thetaMLE = fminsearch(f, thetaMLE);
% prediction
H = speye(t) + spdiags(ones(t-1,1),[-1], t,t)*thetaMLE(2);
uhat = H\delyt;
% store
```

```
yhatMA(t-T0+1,:) = y(t) + uhat(end)*thetaMLE(2);
end
MSFE_IMA = mean((ytph - yhatMA).^2);

%% IAR(1): AR process on del y
y = INDPRO(:,2);
dely = y(2:end) - y(1:end-1);
m=3; T0 = 120-m;  h=1;
% y0 = y(1:m); y = y(m+1:end);
N = length(y);
dely0 = dely(1:m); dely = dely(m+1:end); T = length(dely);
ydelhat = zeros(T-h-T0+1,1);
yhat_IAR = zeros(T-h-T0+1,1);
ytph = y(m+T0+1:  end);
for t = T0:T
delyt = dely(h:t);
zt = [ones(t-h+1,1) [dely0(m); dely(1:t-h)]];
betahat = (zt'*zt)\(zt'*delyt);
ydelhat(t-T0+1,:) = [1 dely(t) ]*betahat;
yhat_IAR(t-T0+1,:) = ydelhat(t-T0+1,:) + y(m+t);
end
MSFE_IAR1 = mean((ytph - yhat_IAR).^2);

%
y = INDPRO(:,2);
dely = y(2:end) - y(1:end-1); T = length(dely);
Y = dely(2:end); X=[ones(T-1,1) dely(1:end-1)];
betahat_I = (X'*X)\X'*Y;
delyhat = X*betahat_I;
MSE_I = mean((delyhat-Y).^2);
k=4;
AIC_I = MSE_I*450 + k*2;
BIC_I = MSE_I*450 + k*log(450);

yhat_I = y(2:end-1)+ delyhat;
sigma2_I = mean((y(3:end)-yhat_I).^2);
yhat_I_T=zeros(12,1);
yhat_I_T(1)= betahat_I(1)+betahat_I(2)*dely(end)+y(end);
yhat_I_T(2)= betahat_I(1)+betahat_I(2)*(yhat_I_T(1)-y(end))+yhat_I_T(1);
for i=2:11
yhat_I_T(i+1) =  betahat_I(1)+betahat_I(2)*(yhat_I_T(i)-yhat_I_T(i-1))+yhat_I_T(i);
end

%% ARMA(1,1)
m = 4;
y = INDPRO(:,2);
hold on
plot(y);
y0 = y(1:m); y = y(m+1:end); T = length(y);
T0 = 120-m; h = 1;
yhatARMA11 = zeros(T-h-T0+1,1);
% theta = [phi1 phi2 mu psi];
f = @(theta) loglike_ARMA11(theta,y0,y(1:T0));
thetahat = fminsearch(f, [.5;0;.5]);

for t = T0:T-h
yt = y(1:t);
% MLE
f = @(theta) loglike_ARMA11(theta, y0, yt);
thetahat = fminsearch(f, thetahat);
% make uhat
H = speye(t) + spdiags(ones(t-1,1),[-1],t,t)*thetahat(3);
X = [[y0(m); y(1:t-1)] ones(t,1)];
```

```
uhat = H\(yt − X*[thetahat(1) thetahat(2)]');
% store
yhatARMA11(t−T0+1,:) = thetahat(2) + thetahat(1)*y(t) + thetahat(3)*uhat(end);
end
plot((T0+h:T), yhatARMA11);
ytph = y(T0+h:end);
MSFA_ARMA11 = mean((ytph−yhatARMA11).^2);
%% ARMA(2,1)
m = 4;
y = INDPRO(:,2);
hold on
plot(y);
y0 = y(1:m); y = y(m+1:end); T = length(y);
T0 = 120−m; h = 1;
yhatARMA21 = zeros(T−h−T0+1,1);
% theta = [phi1 phi2 mu psi];
f = @(theta) loglike_ARMA21(theta,y0,y(1:T0));
thetahat = fminsearch(f, [.5;.5;0;.5]);

for t = T0:T−h
yt = y(1:t);
% MLE
f = @(theta) loglike_ARMA21(theta, y0, yt);
thetahat = fminsearch(f, thetahat);
% make uhat
H = speye(t) + spdiags(ones(t−1,1),[−1],t,t)*thetahat(4);
X = [[y0(m); y(1:t−1)] [y0(m−1:end); y(1:t−2)] ones(t,1)];
uhat = H\(yt − X*[thetahat(1) thetahat(2) thetahat(3)]');
% store
yhatARMA21(t−T0+1,:) = thetahat(3) + thetahat(1)*y(t)...
 + thetahat(2)*y(t−1) + thetahat(4)*uhat(end);
end
plot((T0+h:T), yhatARMA21);
ytph = y(T0+h:end);
MSFA_ARMA21 = mean((ytph−yhatARMA21).^2);

y = INDPRO(:,2); T = length(y);
f = @(theta) loglike_ARMA21_sig(theta,y(1:2),y(3:T));
thetahat = fminsearch(f, [.5;.5; 0;.5;.3]);
sig2 = thetahat(5);
H = speye(T−2) + spdiags(ones(T−2−1,1),[−1],T−2,T−2)*thetahat(4);
X = [[y(2:T−1)] [y(1:T−2)] ones(T−2,1)];
uhat = H\(y(3:T) − X*[thetahat(1) thetahat(2) thetahat(3)]');
yhatARMA21_T = X*thetahat(1:3) + (H−speye(T−2))*uhat;
MSE_ARMA21 = mean((y(3:T)−yhatARMA21_T).^2);
k = 4;
AIC_ARMA21 = MSE_ARMA21*(T−2) + k*2;
BIC_ARMA21 = MSE_ARMA21*(T−2) + k*log(T−2);
yhatARMA21_T=zeros(12,1);
yhatARMA21_T(1) = thetahat(3) + thetahat(1)*y(T)...
 + thetahat(2)*y(T−1) + thetahat(4)*uhat(end);
yhatARMA21_T(2) = thetahat(3) + thetahat(1)*yhatARMA21_T(1) + thetahat(2)*y(T);
for i=3:12
yhatARMA21_T(i) = thetahat(3) + thetahat(1)*yhatARMA21_T(i−1)...
 + thetahat(2)*yhatARMA21_T(i−2);
end
%% ARMA(3,1)

m = 4;
y = INDPRO(:,2);
hold on
plot(y);
y0 = y(1:m); y = y(m+1:end); T = length(y);
```

```
T0 = 120-m; h = 1;
yhatARMA31 = zeros(T-h-T0+1,1);
% theta = [phi1 phi2 mu psi];
f = @(theta) loglike_ARMA31(theta,y0,y(1:T0));
thetahat = fminsearch(f, [.5;.5;.5;0;.5]);

for t = T0:T-h
yt = y(1:t);
% MLE
f = @(theta) loglike_ARMA31(theta, y0, yt);
thetahat = fminsearch(f, thetahat);
% make uhat
H = speye(t) + spdiags(ones(t-1,1),[-1],t,t)*thetahat(5);
X = [[y0(m); y(1:t-1)] [y0(m-1:end); y(1:t-2)] [y0(m-2:end); y(1:t-3)] ones(t,1)];
uhat = H\(yt - X*[thetahat(1) thetahat(2) thetahat(3) thetahat(4)]');
% store
yhatARMA31(t-T0+1,:) = thetahat(4) + thetahat(1)*y(t)...
 + thetahat(2)*y(t-1) + thetahat(3)*y(t-2) + thetahat(5)*uhat(end);
end
plot((T0+h:T), yhatARMA31);
ytph = y(T0+h:end);
MSFA_ARMA31 = mean((ytph-yhatARMA31).^2);

y = INDPRO(:,2); T = length(y);
f = @(theta) loglike_ARMA31_sig(theta,y(1:3),y(4:T));
thetahat = fminsearch(f, [.5;.5;.5; 0;.5;.3]);
sig2 = thetahat(6);
H = speye(T-3) + spdiags(ones(T-3-1,1),[-1],T-3,T-3)*thetahat(5);
X = [[y(3:T-1)] [y(2:T-2)] [y(1:T-3)] ones(T-3,1)];
uhat = H\(y(4:T) - X*[thetahat(1) thetahat(2) thetahat(3) thetahat(4)]');
yhatARMA31_T = X*thetahat(1:4) + (H-speye(T-3))*uhat;
MSE_ARMA31 = mean((y(4:T)-yhatARMA31_T).^2);
k = 5;
AIC_ARMA31 = MSE_ARMA31*(T-3) + k*2;
BIC_ARMA31 = MSE_ARMA31*(T-3) + k*log(T-3);
yhatARMA31_T = zeros(12,1);
yhatARMA31_T(1) = thetahat(4) + thetahat(1)*y(T)...
 + thetahat(2)*y(T-1) + thetahat(3)*y(T-2)+ thetahat(5)*uhat(end);
yhatARMA31_T(2) = thetahat(4) + thetahat(1)*yhatARMA31_T(1)...
 + thetahat(2)*y(T) + thetahat(3)*y(T-1);
yhatARMA31_T(3) = thetahat(4) + thetahat(1)*yhatARMA31_T(2)...
 + thetahat(2)*yhatARMA31_T(1) + thetahat(3)*y(T);
for i = 4:12
yhatARMA31_T(i) = thetahat(4) + thetahat(1)*yhatARMA31_T(i-1) ...
+ thetahat(2)*yhatARMA31_T(i-2) + thetahat(3)*yhatARMA31_T(i-3);
end
%% ARMA(4,1)

m = 5;
y = INDPRO(:,2);
hold on
plot(y);
y0 = y(1:m); y = y(m+1:end); T = length(y);
T0 = 120-m; h = 1;
yhatARMA41 = zeros(T-h-T0+1,1);
% theta = [phi1 phi2 mu psi];
f = @(theta) loglike_ARMA41(theta,y0,y(1:T0));
thetahat = fminsearch(f, [.5;.5;.5;.5;0;.5]);

for t = T0:T-h
yt = y(1:t);
% MLE
f = @(theta) loglike_ARMA41(theta, y0, yt);
```

```
thetahat = fminsearch(f, thetahat);
% make uhat
H = speye(t) + spdiags(ones(t-1,1),[-1],t,t)*thetahat(6);
X = [[y0(m); y(1:t-1)] [y0(m-1:end); y(1:t-2)] [y0(m-2:end); y(1:t-3)] ...
[y0(m-3:end); y(1:t-4)] ones(t,1)];
uhat = H\(yt - X*[thetahat(1) thetahat(2) thetahat(3) thetahat(4) thetahat(5)]');
% store
yhatARMA41(t-T0+1,:) = thetahat(4) + thetahat(1)*y(t) + thetahat(2)*y(t-1)...
+ thetahat(3)*y(t-2)+ thetahat(4)*y(t-3) + thetahat(5)*uhat(end);
end
plot((T0+h:T), yhatARMA41);
ytph = y(T0+h:end);
MSFA_ARMA41 = mean((ytph-yhatARMA41).^2);

y = INDPRO(:,2);
dely = y(2:end) - y(1:end-1); T = length(dely);
Y = dely(2:end); X=[ones(T-1,1) dely(1:end-1)];
betahat_I = (X'*X)\X'*Y;
delyhat = X*betahat_I;
yhat_I = y(2:end-1)+ delyhat;
MSE_I = mean((y(3:end)-yhat_I).^2);
k = 3;
AIC_I = MSE_I*450 + k*2;
BIC_I = MSE_I*450 + k*log(450);


%% ARMA(2,2)

m = 4;
y = INDPRO(:,2);
hold on
plot(y);
y0 = y(1:m); y = y(m+1:end); T = length(y);
T0 = 120-m; h = 1;
yhatARMA22 = zeros(T-h-T0+1,1);
% theta = [phi1 phi2 mu psi];
f = @(theta) loglike_ARMA22(theta,y0,y(1:T0));
thetahat = fminsearch(f, [.5;.5;0;.5;.5]);

for t = T0:T-h
yt = y(1:t);
% MLE
f = @(theta) loglike_ARMA22(theta, y0, yt);
thetahat = fminsearch(f, thetahat);
% make uhat
H = speye(t) + spdiags(ones(t-1,1),[-1],t,t)*thetahat(4)...
+spdiags(ones(t-2,1),[-2],t,t)*thetahat(5);
X = [[y0(m); y(1:t-1)] [y0(m-1:end); y(1:t-2)] ones(t,1)];
uhat = H\(yt - X*[thetahat(1) thetahat(2) thetahat(3)]');
% store
yhatARMA22(t-T0+1,:) = thetahat(3) + thetahat(1)*y(t) + ...
thetahat(2)*y(t-1) + thetahat(4)*uhat(end)+ thetahat(5)*uhat(end-1);
end
plot((T0+h:T), yhatARMA22);
ytph = y(T0+h:end);
MSFA_ARMA22 = mean((ytph-yhatARMA22).^2);

y = INDPRO(:,2); T = length(y);
f = @(theta) loglike_ARMA22_sig(theta,y(1:2),y(3:T));
thetahat = fminsearch(f, [.5;.5;0;.5;.5;.5]);
sig2 = thetahat(6);
H = speye(T-2) + spdiags(ones(T-2-1,1),[-1],T-2,T-2)*thetahat(4)...
+spdiags(ones(T-2-2,1),[-2],T-2,T-2)*thetahat(5);
```

```
X = [[y(2:T-1)] [y(1:T-2)] ones(T-2,1)];
uhat = H\(y(3:T) - X*[thetahat(1) thetahat(2) thetahat(3)]');
yhatARMA22_T = X*thetahat(1:3) + (H-speye(T-2))*uhat;
MSE_ARMA22 = mean((y(3:T)-yhatARMA22_T).^2);
k = 5;
AIC_ARMA22 = MSE_ARMA22*(T-2) + k*2;
BIC_ARMA22 = MSE_ARMA22*(T-2) + k*log(T-2);

yhatARMA22_T = zeros(12,1);
yhatARMA22_T(1) = thetahat(3) + thetahat(1)*y(T) + ...
thetahat(2)*y(T-1) + thetahat(4)*uhat(end)+ thetahat(5)*uhat(end-1);
yhatARMA22_T(2) = thetahat(3) + thetahat(1)*yhatARMA22_T(1)...
 + thetahat(2)*y(T) + thetahat(5)*uhat(end);
for i=3:12
yhatARMA22_T(i) = thetahat(3) + thetahat(1)*yhatARMA22_T(i-1)...
 + thetahat(2)*yhatARMA22_T(i-2);
end
%% ARMA(1,2)

m = 4;
y = INDPRO(:,2);
hold on
plot(y);
y0 = y(1:m); y = y(m+1:end); T = length(y);
T0 = 120-m; h = 1;
yhatARMA12 = zeros(T-h-T0+1,1);
% theta = [phi1 phi2 mu psi];
f = @(theta) loglike_ARMA12(theta,y0,y(1:T0));
thetahat = fminsearch(f, [.5;0;.5;to let the table be the bottom of page
\centering.5]);

for t = T0:T-h
yt = y(1:t);
% MLE
f = @(theta) loglike_ARMA12(theta, y0, yt);
thetahat = fminsearch(f, thetahat);
% make uhat
H = speye(t) + spdiags(ones(t-1,1),[-1],t,t)*thetahat(3)...
+spdiags(ones(t-2,1),[-2],t,t)*thetahat(4);
X = [[y0(m); y(1:t-1)] ones(t,1)];
uhat = H\(yt - X*[thetahat(1) thetahat(2)]');
% store
yhatARMA12(t-T0+1,:) = thetahat(2) + thetahat(1)*y(t)...
 +  thetahat(3)*uhat(end)+ thetahat(4)*uhat(end-1);
end
plot((T0+h:T), yhatARMA12);
ytph = y(T0+h:end);
MSFA_ARMA12 = mean((ytph-yhatARMA12).^2);
```

4.3 Log Likelihood

```
% negative of the log likeliood for ARMA(1,1)
% input: x = [phi1 mu psi]
% input: y = data; y0 = lags;
function ell = loglike_ARMA11(x,y0,y)
phi = zeros(2,1);
phi(1) = x(1);
phi(2) = x(2); psi = x(3);
N = length(y); m = length(y0);
A = speye(N);
B = sparse(2:N, 1:N-1, ones(1,N-1), N,N);
gam = A + B*psi;
```

```
gam2 = gam*gam';
X = [[y0(m);y(1:N-1)] ones(N,1)];
ell = -(y-X*phi)'*(gam2\(y-X*phi));
ell = -ell;

% negative of the log likeliood for ARMA(2,1)
% input: x = [phi1 phi2 mu psi]
% input: y = data; y0 = lags;
function ell = loglike_ARMA21(x,y0,y)
phi = zeros(3,1);
phi(1) = x(1); phi(2) = x(2);
phi(3) = x(3); psi = x(4);
N = length(y); m = length(y0);
A = speye(N);
B = sparse(2:N, 1:N-1, ones(1,N-1), N,N);
gam = A + B*psi;
gam2 = gam*gam';
X = [[y0(m);y(1:N-1)] [y0(m-1:end);y(1:N-2)] ones(N,1)];
ell = -(y-X*phi)'*(gam2\(y-X*phi));
ell = -ell;

% negative of the log likeliood for ARMA(2,1)
% input: x = [phi1 phi2 mu psi sig2]
% input: y = data; y0 = lags;
function ell = loglike_ARMA21_sig(x,y0,y)
phi = zeros(3,1);
phi(1) = x(1); phi(2) = x(2);
phi(3) = x(3); psi = x(4); sig2 = x(5);
N = length(y); m = length(y0);
A = speye(N);
B = sparse(2:N, 1:N-1, ones(1,N-1), N,N);
gam = A + B*psi;
gam2 = gam*gam';
X = [[y0(m);y(1:N-1)] [y0(m-1:end);y(1:N-2)] ones(N,1)];
ell = -log(det(sig2*gam2))-(y-X*phi)'*(gam2\(y-X*phi))/sig2;
ell = -ell;

% negative of the log likeliood for ARMA(3,1)
% input: x = [phi1 phi2 phi3 mu psi]
% input: y = data; y0 = lags;
function ell = loglike_ARMA31(x,y0,y)
phi = zeros(4,1);
phi(1) = x(1); phi(2) = x(2); phi(3) = x(3);
phi(4) = x(4); psi = x(5); % phi(4)=mu
N = length(y); m = length(y0);
A = speye(N);
B = sparse(2:N, 1:N-1, ones(1,N-1), N,N);
gam = A + B*psi;
gam2 = gam*gam';
X = [[y0(m);y(1:N-1)] [y0(m-1:end);y(1:N-2)] [y0(m-2:end);y(1:N-3)] ones(N,1)];
ell = -(y-X*phi)'*(gam2\(y-X*phi));
ell = -ell;

% negative of the log likeliood for ARMA(3,1)
% input: x = [phi1 phi2 phi3 mu psi sig2]
% input: y = data; y0 = lags;
function ell = loglike_ARMA31_sig(x,y0,y)
phi = zeros(4,1);
phi(1) = x(1); phi(2) = x(2);
phi(3) = x(3);phi(4) = x(4);
psi = x(5); sig2 = x(6);
N = length(y); m = length(y0);
A = speye(N);
```

```
B = sparse(2:N, 1:N-1, ones(1,N-1), N,N);
gam = A + B*psi;
gam2 = gam*gam';
X = [[y0(m);y(1:N-1)] [y0(m-1:end);y(1:N-2)] [y0(m-2:end);y(1:N-3)] ones(N,1)];
ell = -log(det(sig2*gam2))-(y-X*phi)'*(gam2\(y-X*phi))/sig2;
ell = -ell;

% negative of the log likeliood for ARMA(2,2)
% input: x = [phi1 phi2 mu psi1 psi2]
% input: y = data; y0 = lags;
function ell = loglike_ARMA22(x,y0,y)
phi = zeros(3,1);
phi(1) = x(1); phi(2) = x(2);
phi(3) = x(3);  % phi(3) = mu
psi1 = x(4); psi2 = x(5);
N = length(y); m = length(y0);
A = speye(N);
B = sparse(2:N, 1:N-1, ones(1,N-1), N,N);
C = sparse(3:N, 1:N-2, ones(1,N-2), N,N);
gam = A + B*psi1 + C*psi2;
gam2 = gam*gam';
X = [[y0(m);y(1:N-1)] [y0(m-1:end);y(1:N-2)] ones(N,1)];
ell = -(y-X*phi)'*(gam2\(y-X*phi));
ell = -ell;

% negative of the log likeliood for ARMA(2,2)
% input: x = [phi1 phi2 mu psi1 psi2 sig^2]
% input: y = data; y0 = lags;
function ell = loglike_ARMA22_sig(x,y0,y)
phi = zeros(3,1);
phi(1) = x(1); phi(2) = x(2);
phi(3) = x(3);  % phi(3) = mu
psi1 = x(4); psi2 = x(5); sig2 = x(6);
N = length(y); m = length(y0);
A = speye(N);
B = sparse(2:N, 1:N-1, ones(1,N-1), N,N);
C = sparse(3:N, 1:N-2, ones(1,N-2), N,N);
gam = A + B*psi1 + C*psi2;
gam2 = gam*gam';
X = [[y0(m);y(1:N-1)] [y0(m-1:end);y(1:N-2)] ones(N,1)];
ell = -log(det(sig2*gam2))-(y-X*phi)'*(gam2\(y-X*phi))/sig2;
ell = -ell;
```

4.3.1 VAR

```
%% VAR(1) model with 2 series
load 'INDPRO.csv'; % industrial production index  2012=100
load 'TCU.csv';% x1 tocal capacity utilization
load 'UNRATE.csv' % x2 unemployment rate
load 'PPIACO.csv';% x3 producer price index  1982=100
load 'CPIAUCSL.csv'; % x4 CPI for urban consumer
T=length(INDPRO(:,2)); x = zeros(T,2);
x(:,1) = INDPRO(:,2); x(:,2) = TCU(:,2);
T = length(x(:,1));
n=2;
X = zeros(n*T,6); y = zeros(n*T,1);
T0 = 120;
yhatVAR1 = zeros(T - T0,n);

X(1,:) = [1 0 0 0 0 0]; y(1) = x(1,1);
X(2,:) = [0 1 0 0 0 0]; y(2) = x(1,2);
for i = 3:2*T
```

```
if mod(i,2) == 1 %t is odd
X(i,:) = [1 0 x((i - 1)/2,1) x((i - 1)/2,2) 0 0];
y(i) = x((i+1)/2,1);
else %t is even
X(i,:) = [0 1 0 0 x(i/2 - 1,1) x(i/2 - 1,2)];
y(i) = x(i/2,2);
end
end

for t = T0:T - 1
yt = y(1:2*t);
Xt = X(1:2*t,:);
beta = (Xt'*Xt)\Xt'*yt;
b = [beta(1); beta(2)];
B = [beta(3) beta(4);beta(5) beta(6)];
%store forecasts
yhatVAR1(t - T0+1,:) = (b + B*[yt(2*t - 1);yt(2*t)])';
end
ytph = x(T0+1:end,:); % observed y {t+h}
MSFE_VAR21 = mean((ytph - yhatVAR1).^2)
L = length(ytph);
plot(1:L,ytph(:,1),1:L,yhatVAR1(:,1))

%% VAR(1) model with 4 series

load 'INDPRO.csv'; % industrial production index  2012=100
load 'TCU.csv';% x1 tocal capacity utilization
load 'UNRATE.csv' % x2 unemployment rate
load 'PPIACO.csv';% x3 producer price index  1982=100
load 'CPIAUCSL.csv'; % x4 CPI for urban consumer

x(:,1) = INDPRO(:,2); x(:,2) = TCU(:,2);
 x(:,3) = UNRATE(:,2); x(:,4) = CPIAUCSL(:,2);
T = length(x(:,1));
n=4;
X = zeros(n*T,20); y = zeros(n*T,1);
T0 = 120;
yhatVAR1 = zeros(T - T0,n);

% set up Big X and y
X(1,:) = [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]; y(1) = x(1,1);
X(2,:) = [0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]; y(2) = x(1,2);
X(3,:) = [0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]; y(3) = x(1,3);
X(4,:) = [0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]; y(4) = x(1,4);

for i = 5:n*T
if mod(i,4) == 1
X(i,:) = [1 0 0 0 x((i - 1)/4,1) x((i - 1)/4,2) x((i - 1)/4,3)...
 x((i - 1)/4,4) 0 0 0 0 0 0 0 0 0 0 0 0];
y(i) = x((i+3)/4,1);
elseif mod(i,4) == 2
X(i,:) = [0 1 0 0 0 0 0 0 x((i - 2)/4,1) x((i - 2)/4,2) ...
x((i - 2)/4,3) x((i - 2)/4,4) 0 0 0 0 0 0 0 0 ];
y(i) = x((i+2)/4,2);
elseif mod(i,4) == 3
X(i,:) = [0 0 1 0 0 0 0 0 0 0 0 0 x((i - 3)/4,1) x((i - 3)/4,2)...
 x((i - 3)/4,3) x((i - 3)/4,4) 0 0 0 0 ];
y(i) = x((i+1)/4,3);
else %t is even
X(i,:) = [0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 x((i - 4)/4,1)...
 x((i - 4)/4,2) x((i - 4)/4,3) x((i - 4)/4,4)];
y(i) = x(i/4,4);
end
```

```
end

for t = T0:T - 1
yt = y(1:n*t);
Xt = X(1:n*t,:);
beta = (Xt'*Xt)\Xt'*yt;
b = [beta(1); beta(2); beta(3); beta(4)];
B = [beta(5) beta(6) beta(7) beta(8);...
beta(9) beta(10) beta(11) beta(12);...
beta(13) beta(14) beta(15) beta(16);...
beta(17) beta(18) beta(19) beta(20);];
%store forecasts
yhatVAR1(t - T0+1,:) = (b + B*[yt(n*t-3); yt(n*t-2); yt(n*t-1); yt(n*t)])';
end
ytph = x(T0+1:end,:); % observed y {t+h}
MSFE_VAR41 = mean((ytph - yhatVAR1).^2)
L = length(ytph);
plot(1:L,ytph(:,1),1:L,yhatVAR1(:,1))

%% VAR(1) model with 3 series

load 'INDPRO.csv'; % industrial production index  2012=100
load 'TCU.csv';% x1 tocal capacity utilization
load 'UNRATE.csv' % x2 unemployment rate
load 'PPIACO.csv';% x3 producer price index  1982=100
load 'CPIAUCSL.csv'; % x4 CPI for urban consumer

T = length(INDPRO(:,1)); x = zeros(T,3);
x(:,1) = INDPRO(:,2); x(:,2) = TCU(:,2);
x(:,3) = UNRATE(:,2); %x(:,4) = CPIAUCSL(:,2);
T = length(x(:,1));
n=3;
X = zeros(n*T,12); y = zeros(n*T,1);
T0 = 120;
yhatVAR1 = zeros(T - T0,n);

% set up Big X and y
X(1,:) = [1 0 0 0 0 0 0 0 0 0 0 0 ]; y(1) = x(1,1);
X(2,:) = [0 1 0 0 0 0 0 0 0 0 0 0 ]; y(2) = x(1,2);
X(3,:) = [0 0 1 0 0 0 0 0 0 0 0 0 ]; y(3) = x(1,3);

for i = 4:n*T
if mod(i,3) == 1
X(i,:) = [1 0 0  x((i - 1)/3,1) x((i - 1)/3,2) x((i - 1)/3,3)  0 0 0 0 0 0] ;
y(i) = x((i+2)/3,1);
elseif mod(i,3) == 2
X(i,:) = [0 1 0  0 0 0  x((i - 2)/3,1) x((i - 2)/3,2) x((i - 2)/3,3) 0 0 0 ];
y(i) = x((i+1)/3,2);
else %t is even
X(i,:) = [ 0 0 1 0 0 0 0 0 0 x((i - 3)/3,1) x((i - 3)/3,2) x((i - 3)/3,3)];
y(i) = x(i/3,3);
end
end

for t = T0:T - 1
yt = y(1:n*t);
Xt = X(1:n*t,:);
beta = (Xt'*Xt)\Xt'*yt;
b = [beta(1); beta(2); beta(3)];
B = [beta(4) beta(5) beta(6);...
beta(7) beta(8) beta(9);...
beta(10) beta(11) beta(12)];
%store forecasts
```

```
yhatVAR1(t - T0+1,:) = (b + B*[yt(n*t-2); yt(n*t-1); yt(n*t)])';
end
ytph = x(T0+1:end,:); % observed y {t+h}
MSFE_VAR31 = mean((ytph - yhatVAR1).^2)
L = length(ytph);
plot(1:L,ytph(:,1),1:L,yhatVAR1(:,1))
```