Question 1

A model that allows major league baseball player salary to differ by position is ln(salary) = β0 + β1years + β2gamesyr + β3bavg + β4hrunsyr + β5rbisyr + β6runsyr + β7fldperc + β8allstar + β9frstbase + β10scndbase + β11thrdbase + β12shrtstop + β13catcher + u where outfield is the base group.

1. State the null hypothesis that, controlling for other factors, catchers and outfielders earn, on average, the same amount. Test this hypothesis using the data in mlb1 and comment on the size of the estimated salary differential.

Soln: H0: β13=0 Ha: β13≠0

ln(salary) = 11.1 + 0.0584years + 0.00977gamesyr + 0.000481bavg + 0.0191hrunsyr + 0.00179rbisyr + 0.0119runsyr + 0.00283fldperc + 0.00634allstar -0.133frstbase – 0.161scndbase + 0.0145thrdbase - 0.0606shrtstop + 0.254catcher + u, n=353 R-squared= 0.6535

The coefficients of catchers are having t-stats=1.93 and a p-value of 0.0543 which is significant at 10% level hence we can reject the H0 at 10% level but not at 5% level.

28.92% catchers have higher salary than outfielders

R-code :

```
model <- lm(log(salary)~years+gamesyr+bavg+hrunsyr+rbisyr+runsyr+fldperc+allstar+
frstbase+scndbase+thrdbase+shrtstop+catcher,data=mlb1)
tidy(model)
summary(model)
(exp(0.254)-1)=0.2892*100=28.92
```

2. State and test the null hypothesis that there is no difference in average salary across positions, once other factors have been controlled for.

Soln: H0: β9 =0,β10 =0,β11=0, β12 =0,β13=0

Ha: β9 ≠0,β10 ≠0,β11≠0, β12 ≠0,β13≠0

The model has F- statistics = 1.77 and p-value=0.1168 >0.1, which is insignificant at 10% level hence we cannot reject the null hypothesis, i.e., there is no difference in average salary across the positions, once other factors have been controlled for.

R-code:

```
modela <- lm(log(salary)~years+gamesyr+bavg+hrunsyr+rbisyr+runsyr+fldperc+allstar,data=mlb1)
summary(modela)
anova(model,modela)
```

3. Are the results from parts 1 and 2 consistent? If not, explain what is happening.

Soln: The result is consistent. In part1 we can see the factor catcher is the only one which is significant at 10% level and there is insignificant difference in salary across the positions. The joint significance test is nothing but testing a significant factor catcher along with other insignificant variables frstbase, scnbase, thrdbase, shortstop at 10%significance level.

Question 2 Use the data in gpa2 for this exercise.

1. Consider the equation

colgpa = β0 + β1hsize + β2hsize2 + β3hsperc + β4sat + β5female + β6athlete + u

where colgpa is cumulative college grade point average, hsize is size of high school graduating class, in hundreds, hsperc is academic percentile in graduating class, sat is combined SAT score, female is a binary gender variable, and athlete is a binary variable, which is one for student-athletes. What are your expectations for the coefficients in this equation? Which ones are you unsure about?
Soln:
colgpa = 1.24 -0.0569hsize + 0.00468hsize2 -0.0132hsperc +0.00165sat + 0.155female + 0.169athlete + u, n=4137 R-squared=0.2925

We are clear on β1(-ve sign), the smaller the class size the better the colgpa
β3 with decrease in highschool percentile the increase in average colgpa
β4(+ve sign) with higher the SAT score the better the students are hence increase in colgpa
β6(+ve sign) we may think that the athletes perform worse than the non-athletes which is not sure along with β5 (+ve sign) not clear on females and males have significant GPA

2. Estimate the equation in part 1 and report the results in the usual form. What is the estimated GPA differential between athletes and nonathletes? Is it statistically significant?
Soln:
colgpa = 1.24 -0.0569hsize + 0.00468hsize2 -0.0132hsperc +0.00165sat + 0.155female + 0.169athlete + u, n=4137 R-squared=0.2925
The coefficient of athlete is 0.169 , which means the estimated GPA of athletes are 0.169 points better than non-athletes and the p-value=0.0000650<0.001 i.,e it is significant even at 0.1% level

R-code:
colgpa = β0 + β1hsize + β2hsize^2 + β3hsperc + β4sat + β5female + β6athlete + u
model1 <- lm(colgpa~hsize+I(hsize^2)+hsperc+sat+female+athlete,data=gpa2)
tidy(model1)
summary(model1)

3. Drop sat from the model and reestimate the equation. Now, what is the estimated effect of being an athlete? Discuss why the estimate is different than that obtained in part 2.
Soln:
3.05 -0.0534hsize + 0.00532hsize2 -0.0171hsperc + 0.00581female + 0.00545athlete + u, n=4137 R-squared=0.1885
On dropping SAT the coefficient on athlete is 0.00545 with t-stats=0.122 and p-value=0.9032 >0.1 which is insignificant at 10% level. In this model without SAT the combined SAT score is not controlled, hence athlete becomes statistically no different from zero and athlete score less than non-athlete on an average.

R-code:

```
model2 <- lm(colgpa~hsize+I(hsize^2)+hsperc+female+athlete,data=gpa2)
tidy(model2)
summary(model2)
```

4. In the model from part 1, allow the effect of being an athlete to differ by gender and test the null hypothesis that there is no ceteris paribus difference between women athletes and women nonathletes.

Soln: H0: there is no difference between women athletes and women non athletes

To find the effect of athlete to differ by gender we will consider women-nonathlete as base so the equation will be

colgpa = β0 + β1hsize + β2hsize^2 + β3hsperc + β4sat + β5female*athlete + β6male*non-athlete + β7male*athlete+ u

colgpa = 1.4-0.0568hsize+0.00467(hsize^2)-0.0132hsperc+0.00165sat+0.175 female*athlete-0.155 male*non-athlete+0.0128 male*athlete +u

the coefficients of female athlete =0.175 with p-value=0.0372>0.01 and <0.05 which is significant at 5%level but not at 1% level.

This shows that female athletes are 0.175 points better than female non-athletes at 5% significance level

Hence, we can reject the null hypothesis at 5% level but not at 1% level.

R-code:

```
model3 <-lm(colgpa~hsize+I(hsize^2)+hsperc+sat+I(female*athlete)+I((1-female)*(1-athlete))+I((1-female)*athlete),data=gpa2)
tidy(model3)
summary(model3)
```

5. Does the effect of sat on colgpa differ by gender? Justify your answer.

Soln: If we add female*sat for the equation i.e.,

colgpa = 1.26 -0.0569hsize + 0.00469hsize2 -0.0132hsperc +0.00163sat + 0.0000512(sat*female)+0.102female + 0.168athlete + u, n=4137 R-squared=0.2925

The coefficients of (female*sat) is 0.0000512 with t-statistics= 0.397<1.96 and p-value=0.692 >0.05 which is insignificant at 5%level hence there is no effect of sat on colgpa differ by gender

R-code:

```
model4 <- lm(colgpa~hsize+I(hsize^2)+hsperc+sat+I(sat*female)+female+athlete,data=gpa2)
tidy(model4)
summary(model4)
```

Question 3

Use the data in loanapp for this exercise. The binary variable to be explained is approve, which is equal to one if a mortgage loan to an individual was approved. The key explanatory variable is white, a dummy variable equal to one if the applicant was white. The other applicants in the data set are

black and Hispanic. To test for discrimination in the mortgage loan market, a linear probability model can be used: approve = β0 + β1white + other factors . . . 1

1. If there is discrimination against minorities, and the appropriate factors have been controlled for, what is the sign of β1?

Soln : Sign of β1 will be positive as they are majority with 1681 out of 1989

R-code:

```
sum(as.numeric(loanapp$white==1))#white 1681
sum(as.numeric(loanapp$black==1))#black 191
sum(as.numeric(loanapp$hispan==1))#111
```

2. Regress approve on white and report the results in the usual form. Interpret the coefficient on white. Is it statistically significant? Is it practically large?

Soln: approve = β0 + β1white +u

Approve = 0.70799 + 0.201white+ u, n=1989, R-squared=0.0489

If the applicant is white then there is probability of getting loan approved by 20.1% .Yes, it is statistically significant with p-value = 2e-16 which is significant at 0.1% level. Its large.

R-code:

```
model <- lm(approve~white,data=loanapp)
summary(model)
```

3. As controls, add the variables hrat, obrat, loanprc, unem, male, married, dep, sch, cosign, chist, pubrec, mortlat1, mortlat2, and vr. What happens to the coefficient on white? Is there still evidence of discrimination against nonwhites?

Soln: approve= 0.937+0.129white+0.00183hrat-0.00543 obrat-0.147 loanprc-0.00730 unem,-0.00414male+0.0458 married - 0.00683dep + 0.00175 sch+ 0.00977cosign+ 0.133 chist -0.242pubrec -0.0573mortlat1 -0.114 mortlat2 -0.0314vr +u, n=1971, R-squared=0.1656

Coefficient on white is reduced by 0.072 units but it's still we can see there is discrimination against nonwhites as the coefficient is still significant and positive.

R-code:

```
model <- lm(approve~white + hrat + obrat + loanprc + unem + male + married + dep + sch + cosign + chist + pubrec + mortlat1+mortlat2+vr,data=loanapp)
summary(model)
tidy(model)
```

4. Now, allow the effect of race to interact with the variable measuring other obligations as a percentage of income (obrat). Is the interaction term significant?

Soln: we can add the factor obrat*white to find the effect of race to interact with other obligations. The interaction term (obrat*white) has coefficient= 0.008088 with p-value 0.000423<0.01 hence significant at 1% level.

R-code:

model <- lm(approve ~ white + hrat + I(obrat*white) + obrat +loanprc + unem + male + married + dep + sch + cosign + chist + pubrec + mortlat1+mortlat2+vr,data=loanapp)
summary(model)

5. Using the model from part 4, what is the effect of being white on the probability of approval when obrat = 32, which is roughly the mean value in the sample? Obtain a 95% confidence interval for this effect.

Soln: approve= -0.145975 white + 0.008088 obrat *white
d(approve)/d(white) = -0.145975 + 0.008088 *32
d(approve)/d(white) = 0.1128377

The standard error of this marginal effect when obrat = 32 is
$sqrt$(2^2 32^2 [$se(\beta(obrat\hat{}))$] 2) = 64 $*$ 0.0022155 = 0.141792.

Therefore, the 95% confidence interval of this marginal effect of being white is 0.1128371 ± 1.96 $*$ 0.141792 = [−0.165, 0.39] The 95% confidence interval for the effect of being white encompasses zero and, therefore, for the case where obrat = 32, there appears to be no significant discrimination against non- whites.

Question 4

1. **Use the data in hprice1 to obtain the heteroskedasticity-robust standard errors for equation: price = β0 + β1lotsize + β2sqrft + β3bdrms + u**
   Discuss any important differences with the usual standard errors.

Soln: price = -21.8+0.00207lotsize+0.123sqrft+13.9bdrms+u , n=88, R-squared = 0.6724

Standard error using OLS

term	estimate	std.error	statistic	p.value
(Intercept)	-21.8	29.5	-0.739	0.462
lotsize	0.00207	0.00064	3.22	0.00182
sqrft	0.123	0.0132	9.28	1.66E-14
bdrms	13.9	9.01	1.54	0.128

heteroskedasticity-robust standard errors

term	estimate	std.error	statistic	p.value
(Intercept)	-21.8	41	-0.531	0.596
lotsize	0.00207	0.00715	0.289	0.772
sqrft	0.123	0.0407	3.01	0.00258
bdrms	13.9	11.6	1.2	0.231

We can see that the coefficient of lotsize has the p-value in OLS is 0.00182 which is significant at 1% level whereas the p-value in model with heteroskedasticity-robust standard errors is 0.772 which is not significant at 1% level

The OLS standard error on coefficient of bdrms is 9.01 which is lower than the heteroskedasticity-robust standard errors which is 11.6

2)Repeat part 1 for equation ln(price) = β0 + β1 ln(lotsize) + β2 ln(sqrft) + β3bdrms + u

Soln: ln(price) = -1.3+0.168ln(lotsize)+0.7 ln(sqrft)+0.037 bdrms+u , n= 88, R-squared =0.6274

OLS Standard errors

term	estimate	std.error	p.value
(Intercept)	-1.3	0.651	0.0497E
log(lotsize)	0.168	0.0383	0.0000331
log(sqrft)	0.7	0.0929	5.01E-11
bdrms	0.037	0.0275	0.183

heteroskedasticity-robust standard errors

term	estimate	std.error	p.value
(Intercept)	-1.3	0.85	0.127
log(lotsize)	0.168	0.0533	0.00162
log(sqrft)	0.7	0.121	8E-09
bdrms	0.037	0.0356	0.299

3. What does this example suggest about heteroskedasticity and the log transformation?

Soln:Using the logarithmic transformation of the dependent variable often mitigates, if not eliminates, heteroskedasticity. The standard errors of heteroskedasticity is slightly higher than the usual standard error. The log(lotsize) and log(sqrft) is still having significance level and bdrms are still insignificant.

Question 5

Use the data set gpa1 for this exercise.

1) Use OLS to estimate a model relating colGPA to hsGPA, ACT, skipped, and PC. Obtain the OLS residuals and fitted values.

Soln: colGPA = 1.36+ 0.413 hsGPA +0.0133 ACT – 0.0710 skipped +0.124 PC + u ,

n=141, R-squared=0.2593

R-code:

```
model <- lm(colGPA~hsGPA+ACT+skipped+PC,data=gpa1)
summary(model)
tidy(model)
tidyw(model)
res <- residuals(model)
fit <- predict(model)
```

2. In the regression of uˆ 2 i on colGP A , colGP A ^ 2 , obtain the fitted values, say hˆ i .

Soln: The fitted values for the regression of uˆ 2 i on colGP A , colGP A ^ 2 is in variable h

R-code :

```
modeld <- lm(I(res^2)~fit+I(fit^2),data=gpa1)
h<-predict(modeld)tidy(modeld)
```

3. Verify that the fitted values from part 2 are all strictly positive. Then, obtain the weighted least squares estimates using weights 1/hˆ i . Compare the weighted least squares estimates for the effect of skipping lectures and the effect of PC ownership with the corresponding OLS estimates. What about their statistical significance?

Soln: Since the fitted values from part 2 is stored in variable h the minimum value is 0.02738136 hence verified that all are positive.

colGPA = 1.4 + 0.403 hsGPA + 0.0132 ACT - 0.0764 skipped +0.126PC + u ,

n=141, R-squared=0.03062

We can see that coefficients of skipping lecture is having minimal change i.e., from 0.0710 to 0.0764 with still having inverse effect on colGPA , corresponding OLS estimate with the p-value 0.00768 <0.01 which is significant at 1% level in weighted least square estimates p-value of 0.000762 <0.01 which is at 1% level.

The coefficients of PC ownership have changed from 0.124 to 0.126 which is minimal change. p-value in OLS estimate is 0.0316 < 0.05 which is significant at 5% level whereas in weighted least square estimates the p-value for PC is 0.0269 <0.05 which is significant at 5% level

R-code:

```
min(h)
modele <- lm(colGPA~hsGPA+ACT+skipped+PC,weights=(1/h),data=gpa1)
summary(modele)
tidy(modele)
```

4. In the WLS estimation from part 3, obtain heteroskedasticity-robust standard errors. In other words, allow for the fact that the variance function estimated in part 2 might be misspecified. Do the standard errors change much from part 3?

Soln: The standard errors do have a change which is slight increase and all the independent variables that were statistically significant before are still significant now.

Standard error in OLS

term	estimate	std.error	statistic	p.value
Intercept)	1.4	0.298	4.7	6.39E-06
hsGPA	0.403	0.0834	4.83	3.65E-06
ACT	0.0132	0.00983	1.34	0.183
skipped	-0.0764	0.0222	-3.44	0.000762
PC	0.126	0.0563	2.24	0.0269

heteroskedasticity-robust standard errors for part 3

term	estimate	std.error	statistic	p.value
Intercept)	1.4	0.324	4.33	1.51E-05
hsGPA	0.403	0.0894	4.5	6.68E-06
ACT	0.0132	0.0109	1.21	0.228
skipped	-0.0764	0.0224	-3.41	0.000641
PC	0.126	0.0606	2.08	0.0376

Question 6

1. Go online and search for daily bitcoin prices. Find and download a historical series of bitcoin prices going back to at least 2014.

2. Go to St Louis FRED. Find and download the S&P500 (SP500), the London bullion market price for gold in US dollars (GOLDAMGBD228NLBM), the US/Euro exchange rate (DEXUSEU), and the West Texas Intermediate spot price of oil (DCOILWTICO). These should all be available daily as well.

3. Merge all the data sets together (you can use either R or Excel or whatever).

Soln: Merged data obtained from part2 in excel. Merged the part1 and part2 data in R

R-code:

```
bitcoin <- fread("C:/Ranjitha/R_learning/JasonParker/Assignment/bitcoin.csv")
fred1 <- fread ("C:/Ranjitha/R_learning/JasonParker/Assignment/fred.csv")
fred <- merge(bitcoin,fred1,all=T,by='DATE')
#sorting the date in dataset
fred <- fred[order(as.Date(fred$DATE,format="%m/%d/%Y")),]
fred$DATE <- as.Date(fred$DATE, "%m/%d/%Y")
fred$time <- substr(fred$DATE,0,4)
#converting the values as numeric
fred$Bitcoin <- (as.numeric(fred$Bitcoin))
fred$DCOILWTICO <- (as.numeric(fred$DCOILWTICO))
fred$DEXUSEU <- (as.numeric(fred$DEXUSEU))
fred$GOLDAMGBD228NLBM <- (as.numeric(fred$GOLDAMGBD228NLBM))
fred$SP500 <- (as.numeric(fred$SP500))
fred$time <- (as.numeric(fred$time))
#ommiting NA values
fred <- na.omit(fred)
```

4. Plot the series in R.

Soln:

R-code:

```
ts.plot(fred$DCOILWTICO)
ts.plot(fred$DEXUSEU)
ts.plot(fred$GOLDAMGBD228NLBM)
ts.plot(fred$SP500)
```

```
ts.plot(fred$Bitcoin)
```

5. Use a naïve regression to find spurious correlations to the bitcoin price in the data set (e.g., regress the bitcoin price on the other series without any differencing to see if you find any interesting but total bullshit relationships).

Soln: I could see that all the models are having very high significance with very low p-values.
R-square for bitcoin against oil= 0.02039, gold = 0.43,euro exchange = 0.07565, SP =0.758 respectively

R-code:

```
DCOmodel <- lm(Bitcoin~DCOILWTICO,data=fred)
DEXmodel <- lm(Bitcoin~DEXUSEU,data=fred)
GOLDmodel <- lm(Bitcoin~GOLDAMGBD228NLBM,data=fred)
SPmodel <- lm(Bitcoin~SP500,data=fred)

tidy(DCOmodel)
tidy(GOLDmodel)
tidy(DEXmodel)
tidy(SPmodel)
summary(DCOmodel)
summary(GOLDmodel)
summary(DEXmodel)
summary(SPmodel)
```

6. Use the KPSS test to find how many differences each series takes to become stationary.

Soln: Bitcoin is level stationary after 1st differencing, oil is level stationary after 1st differencing, gold is level stationary after 1st differencing, euro exchange is level stationary after first differencing and SP500 is level stationary after 1st differencing

R-code:

```
kpss.test(fred$Bitcoin)
kpss.test(fred$Bitcoin,null="Trend")
kpss.test(diff(fred$Bitcoin)) #bitcoin is level stationary after 1st differencing

kpss.test(fred$DEXUSEU)
kpss.test(fred$DEXUSEU,null="Trend")
kpss.test(diff(fred$DEXUSEU)) )# euro exchange is level stationary after 1st differencing

rep.kpss(fred$DCOILWTICO) #oil is level stationary after 1st differencing
rep.kpss(fred$DEXUSEU)# euro exchange  is level stationary after 1st differencing
rep.kpss(fred$GOLDAMGBD228NLBM)#gold is level stationary after 1st differencing
rep.kpss(fred$SP500)#SP500 is level stationary after 1st differencing
```

7. After taking differences, regress the bitcoin price on the other series. What relationships do you find now?

Soln: I could see that oil now is significant at 10% level with p-value 0.0884, euro exchange is insignificant, gold is significant at 1% level and sp model is still significant at 0.1% level.
R-squared for oil is 0.00197, euro exchange = 0.001575, gold= 0.006458 and sp500 = 0.02253

R-code:

```
DCOmodel <- lm(diff(Bitcoin)~diff(DCOILWTICO),data=fred)
DEXmodel <- lm(diff(Bitcoin)~diff(DEXUSEU),data=fred)
GOLDmodel <- lm(diff(Bitcoin)~diff(GOLDAMGBD228NLBM),data=fred)
SPmodel <- lm(diff(Bitcoin)~diff(SP500),data=fred)
tidy(DCOmodel)
tidy(DEXmodel)
tidy(GOLDmodel)
tidy(SPmodel)

summary(DCOmodel)
summary(DEXmodel)
summary(GOLDmodel)
summary(SPmodel)
```

8. Remove all the data before 2017 where the bitcoin price starts to spike. Plot the new data. This is the data you are to use for the rest of the question.

Soln:
R-code:

```
fred2017 <- fred %>% filter(DATE > '2016-12-31')
ts.plot(fred2017$DCOILWTICO)
ts.plot(fred2017$DEXUSEU)
ts.plot(fred2017$GOLDAMGBD228NLBM)
ts.plot(fred2017$SP500)
ts.plot(fred2017$Bitcoin)
```

9. Plot the ACF and PACF of the bitcoin price.

Soln:
R-code:

```
ggtsdisplay(fred2017$Bitcoin)
```

10. Fit various arima models to the bitcoin price. Which model fits best using the AIC?

Soln: model with p=2, d=1 and q=2 with AIC= 13616.39 and BIC=13640.46

R-code:

```
modeld <- arima(fred2017$Bitcoin,c(20,1,20))
summary(modeld)
outp <- matrix(0,4^2,5)
count <- 1
for(i in 0:3){
 for(j in 0:3){
  modeld <- arima(fred2017$Bitcoin,c(i,1,j))
  outp[count,] <- c(i,1,j,AIC(modeld),BIC(modeld))
  count <- count + 1
 }
}
outp <- data.table(outp)
names(outp) <- c('p','d','q','aic','bic')
outp
outp[aic==0,]$aic <- 9999
outp[bic==0,]$bic <- 9999
outp[which.min(outp$aic),,]
#   p d q      aic          bic
#  2 1 2 13616.39 13640.46
```

11. Forecast the next 30 days of the bitcoin price and plot the forecast.

Soln: Modelf has lower AIC hence forcasting using modelf

R-code:

```
modele <- arima(fred2017$Bitcoin,c(2,1,2),seasonal=list(order=c(1,1,1),period=30))
 modelf <- arima(fred2017$Bitcoin,c(2,1,2),seasonal=list(order=c(0,1,1),period=30))
AIC(modele)#13282.68
AIC(modelf)#13276.59
plot(forecast(modelf,h=30))
```

12. Plot the periodogram of the data. Do you see any seasonality in the data?

Soln: I could see spike raise in every 15 days and every 4 months and 11 months not much of seasonality

R-code:

```
dbit <- diff(fred2017$Bitcoin)
TSA::periodogram(dbit)#has 15 days spike
1/0.08
dco <- diff(fred2017$DCOILWTICO)
dex <- diff(fred2017$DEXUSEU)
gold <- diff(fred2017$GOLDAMGBD228NLBM)
```

```
sp <- diff(fred2017$SP500)
```

13. Fit a model where you regress the stationarity-transformed price on dummy variables for the different days of the week. Obtain the residuals from the model. Plot the periodogram of these residuals. Has the periodogram changed greatly? Do you think this transformation helps us to capture any seasonality in the data?

Soln: Yes er could see the spikes for every 13 days this is capturing seasonality

R-code:

```
fred2017$weekday <- as.factor(weekdays(fred2017$DATE))
nrows <- nrow(fred2017)
modelw <- lm(diff(Bitcoin)~weekday[2:nrows],data=fred2017)
summary(modelw)
x<- residuals(modelw)
TSA::periodogram(x,log='no',plot=TRUE,ylab="Periodogram",xlab="Frequency",lwd=2)
```

14. Using the AIC, select a VAR model which best captures the relationships between our 5 variables. What Granger causality relationships do you see between our prices?

Soln: AIC value = 27288.76 with lag as 3

R-code:

```
xdata <- cbind(dbit,dco,dex,gold,sp)
xdata <- dbit
AIC(vars::VAR(xdata,1,type="both"))
AIC(vars::VAR(xdata,2,type="both"))
AIC(vars::VAR(xdata,3,type="both"))# is having lower AIC value hence choosing this one
```

15. Forecast the next 30 days of the prices using the VAR model. Compare your forecasts to one from the ARIMA model.

Soln:

R-code:

```
test <- vars::VAR(xdata,3,type="both")
prd <- predict(test, n.ahead = 30, ci = 0.95, dumvar = NULL)
plot(prd)
```