

# report 2

Yixiao xhao

05/08/2019

```
##read data into R ##install.packages("TSA") ##install.packages("forecast")
```

```
data<-read.csv("~/Desktop/report1/XTEITT01CNM156S.csv")
data.ts<-ts(data$XTEITT01CNM156S,start=c(1992,1),frequency = 12)
library(TSA)
```

```
##
## Attaching package: 'TSA'
```

```
## The following objects are masked from 'package:stats':
##
##      acf, arima
```

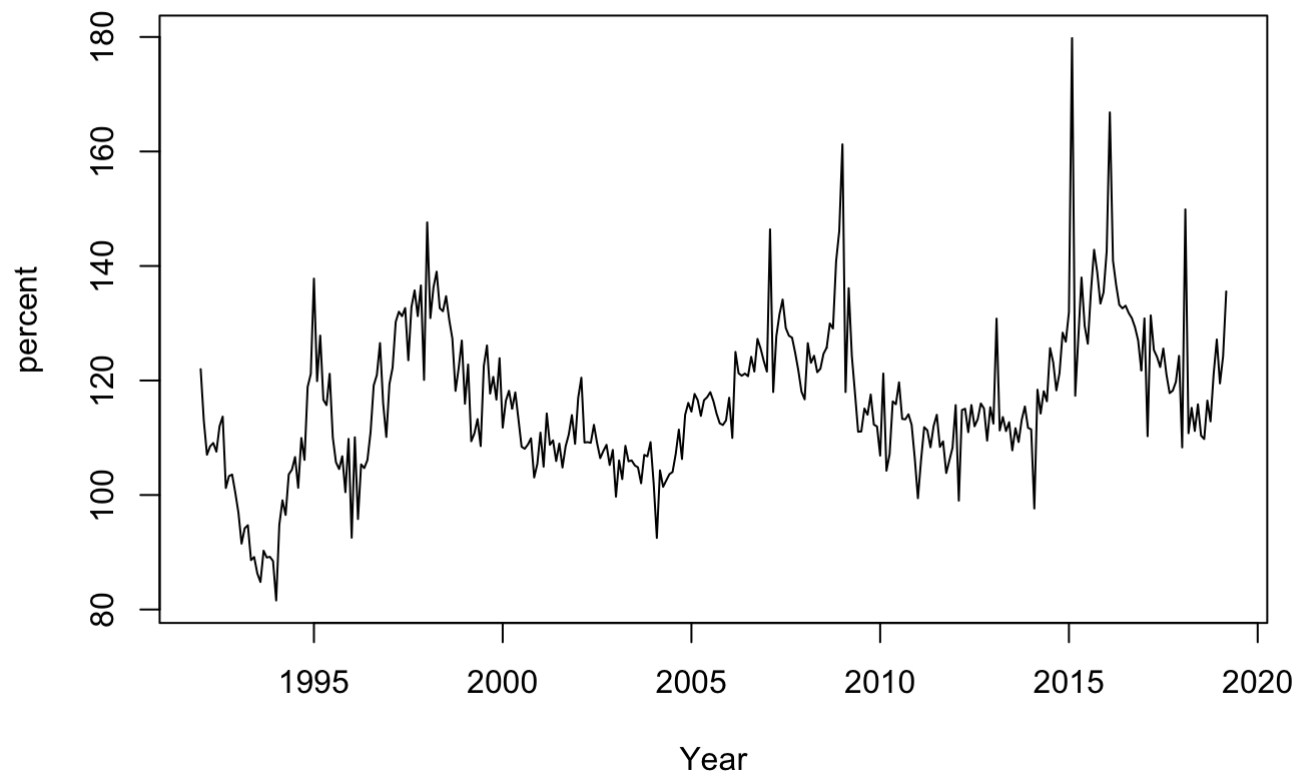
```
## The following object is masked from 'package:utils':
##
##      tar
```

```
library(forecast)
```

```
##identification
```

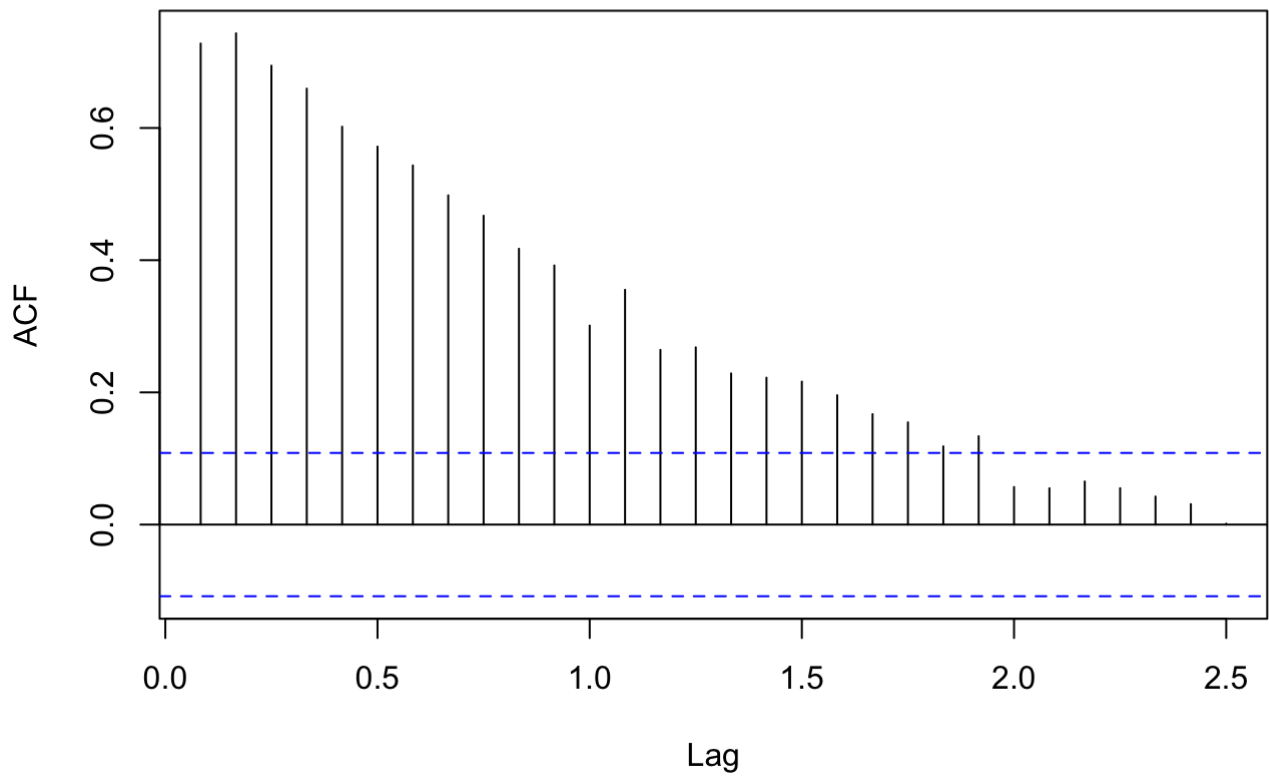
```
plot(data.ts,ylab="percent",xlab="Year",main="Ration of Exports to I,port for China")
```

Ration of Exports to I,port for China



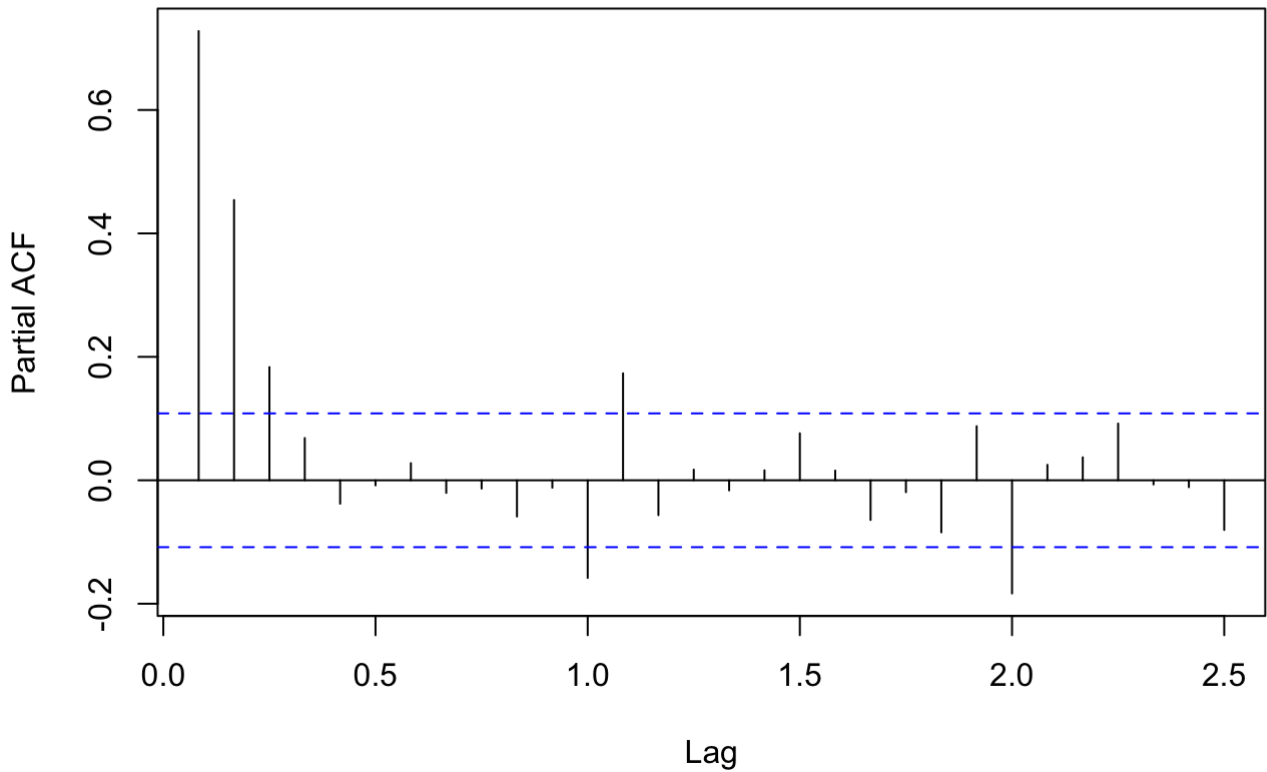
```
acf(data.ts,lag=30)
```

Series data.ts



```
pacf(data.ts,lag=30)
```

Series data.ts



```
eacf(data.ts)
```

##	AR/MA													
##	0	1	2	3	4	5	6	7	8	9	10	11	12	13
##	0	x	x	x	x	x	x	x	x	x	x	x	x	x
##	1	x	x	o	o	o	o	o	o	o	x	x	x	x
##	2	x	o	o	o	o	o	o	o	o	o	x	x	o
##	3	x	o	x	o	o	o	o	o	o	o	x	o	o
##	4	x	x	o	o	o	o	o	o	o	o	o	o	o
##	5	x	x	o	o	o	o	o	o	o	o	o	o	o
##	6	x	o	x	o	o	o	o	o	o	o	x	o	o
##	7	x	o	x	o	o	o	o	o	o	o	x	o	o

```
auto.arima(data.ts)
```

```
## Series: data.ts
## ARIMA(1,1,1)(1,0,2)[12]
##
## Coefficients:
##          ar1          ma1          sar1          sma1          sma2
##      -0.1949  -0.4143  -0.6282   0.4888  -0.3472
## s.e.   0.0946   0.0876   0.0890   0.0893   0.0515
##
## sigma^2 estimated as 52.36:  log likelihood=-1107.88
## AIC=2227.76   AICc=2228.02   BIC=2250.48
```

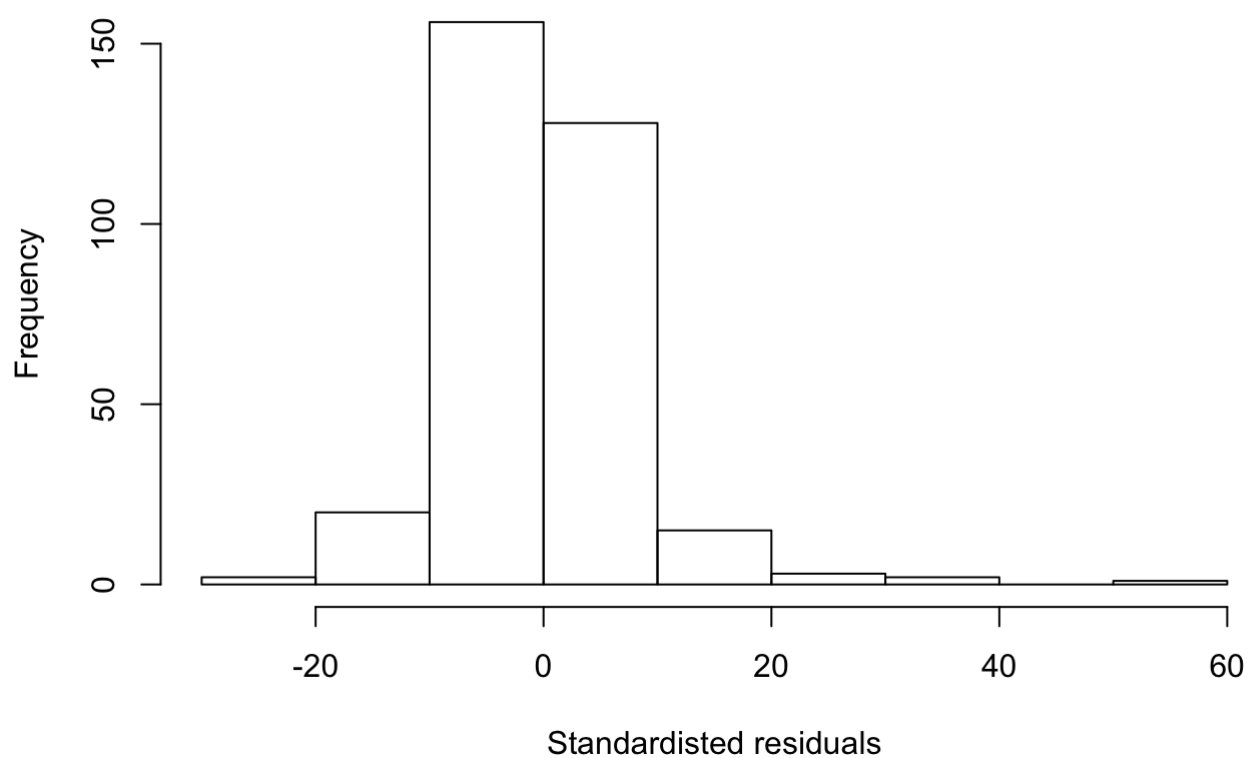
##estimation

```
data.ts.fit<-Arima(data.ts,order=c(1,0,2),include.constant = TRUE,method="ML")
summary(data.ts.fit)
```

```
## Series: data.ts
## ARIMA(1,0,2) with non-zero mean
##
## Coefficients:
##          ar1          ma1          ma2          mean
##      0.9484  -0.6223   0.0968  116.8217
## s.e.   0.0198   0.0552   0.0576   3.7093
##
## sigma^2 estimated as 59.52:  log likelihood=-1130.76
## AIC=2271.52   AICc=2271.7   BIC=2290.47
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.00232698  7.667895  5.25731 -0.3986427  4.447924  0.4756581
##              ACF1
## Training set -0.01866109
```

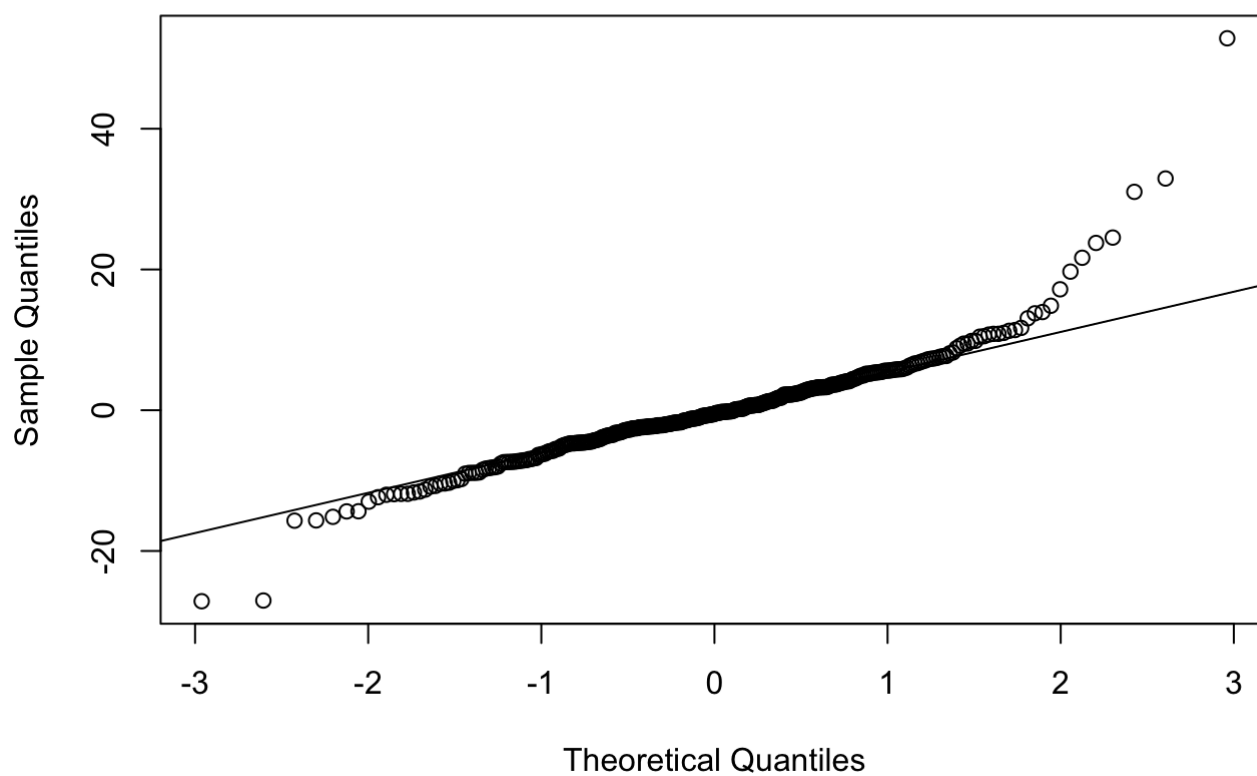
##Diagnostics ##Check for normality:

```
hist(residuals(data.ts.fit),xlab="Standardisted residuals",main="")
```



```
qqnorm(residuals(data.ts.fit))  
qqline(residuals(data.ts.fit))
```

### Normal Q-Q Plot



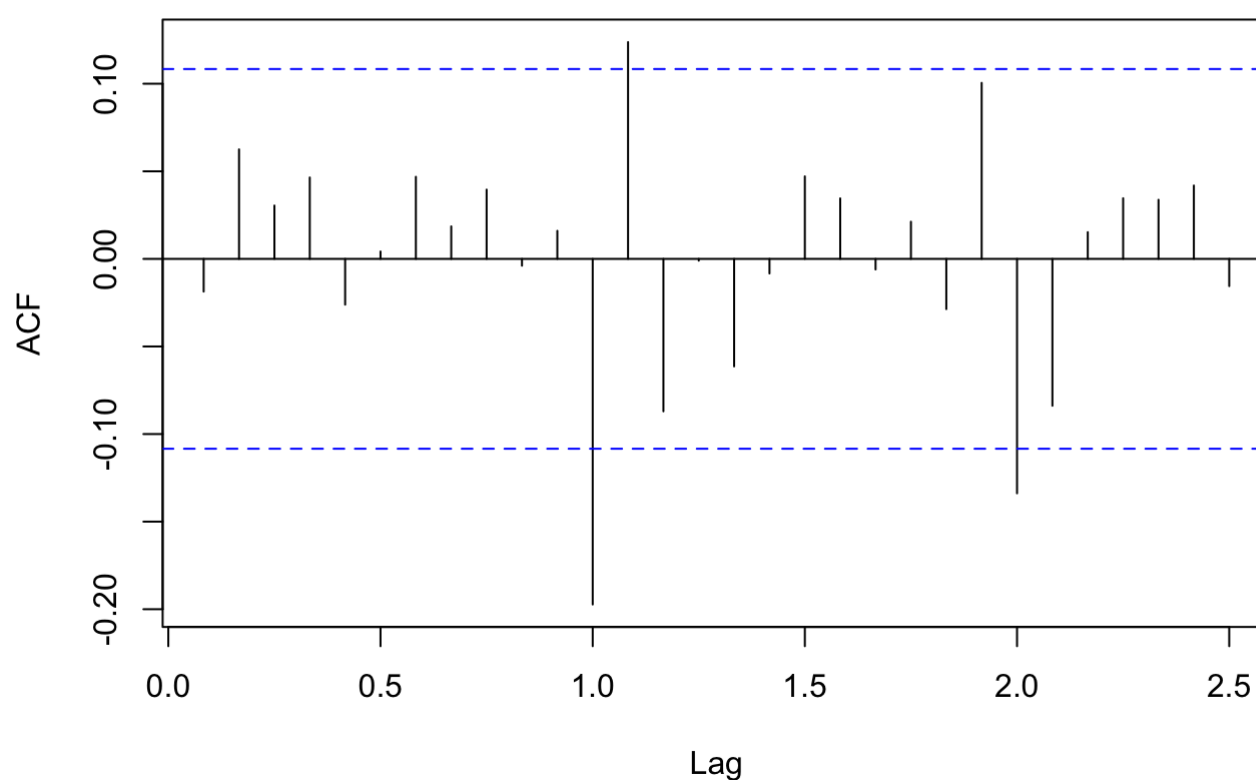
```
shapiro.test(residuals(data.ts.fit))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(data.ts.fit)  
## W = 0.89981, p-value = 7.075e-14
```

##Check for serial correlation

```
acf(residuals(data.ts.fit),main="Sample ACF for standardized residuals",lag=30)
```

### Sample ACF for standardized residuals



```
Box.test(residuals(data.ts.fit),lag=10,type ="Ljung-Box",fitdf=3)
```

```
##  
##  Box-Ljung test  
##  
## data:  residuals(data.ts.fit)  
## X-squared = 4.0588, df = 7, p-value = 0.773
```

##Forecast ##Number of period for forecasting is 10, the confidence level as 95% and 90%

```
forecast1<- forecast(data.ts.fit,h=10,level=95)  
forecast2<-forecast(data.ts.fit,h=10,level=90)  
forecast1
```

```
##          Point Forecast      Lo 95      Hi 95
## Apr 2019      126.3227 111.2012 141.4443
## May 2019      127.1648 111.2595 143.0700
## Jun 2019      126.6308 109.5815 143.6800
## Jul 2019      126.1243 108.1081 144.1406
## Aug 2019      125.6441 106.8005 144.4877
## Sep 2019      125.1886 105.6307 144.7465
## Oct 2019      124.7566 104.5779 144.9353
## Nov 2019      124.3470 103.6258 145.0682
## Dec 2019      123.9585 102.7612 145.1558
## Jan 2020      123.5900 101.9735 145.2065
```

```
forecast2
```

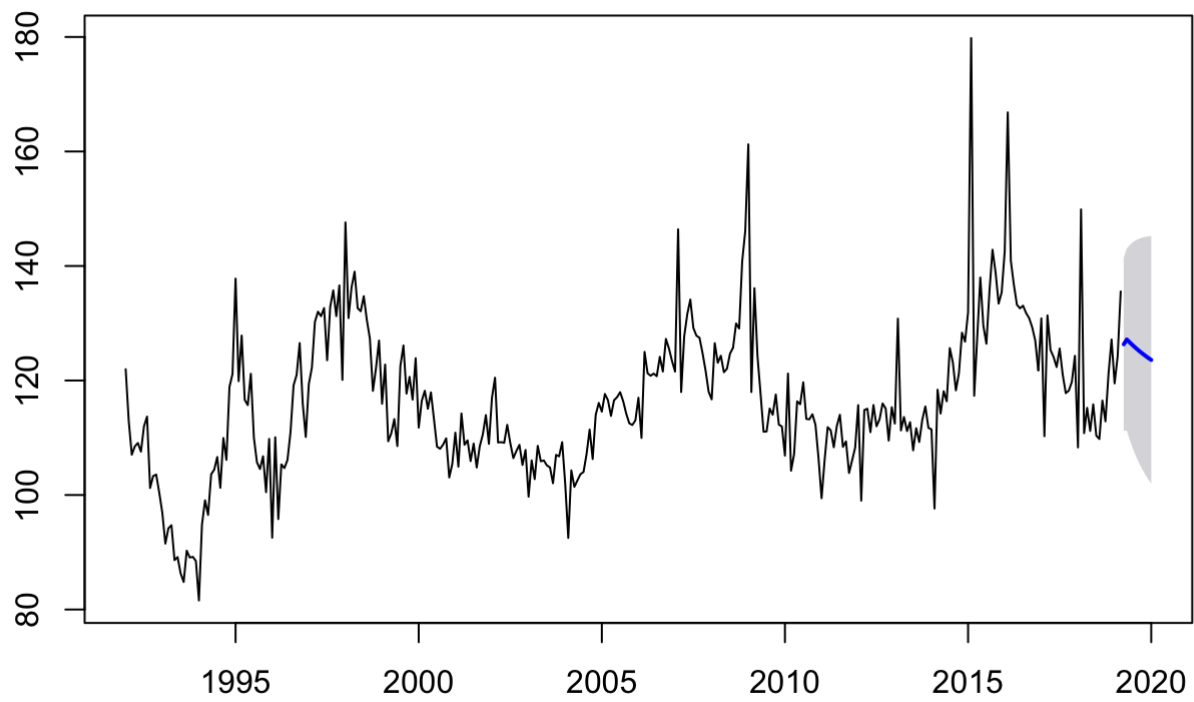
```
##          Point Forecast      Lo 90      Hi 90
## Apr 2019      126.3227 113.6323 139.0132
## May 2019      127.1648 113.8166 140.5129
## Jun 2019      126.6308 112.3226 140.9390
## Jul 2019      126.1243 111.0046 141.2441
## Aug 2019      125.6441 109.8300 141.4581
## Sep 2019      125.1886 108.7751 141.6021
## Oct 2019      124.7566 107.8221 141.6911
## Nov 2019      124.3470 106.9572 141.7368
## Dec 2019      123.9585 106.1691 141.7478
## Jan 2020      123.5900 105.4488 141.7312
```

```
par(mforw=c(1,2))
```

```
## Warning in par(mforw = c(1, 2)): "mforw" is not a graphical parameter
```

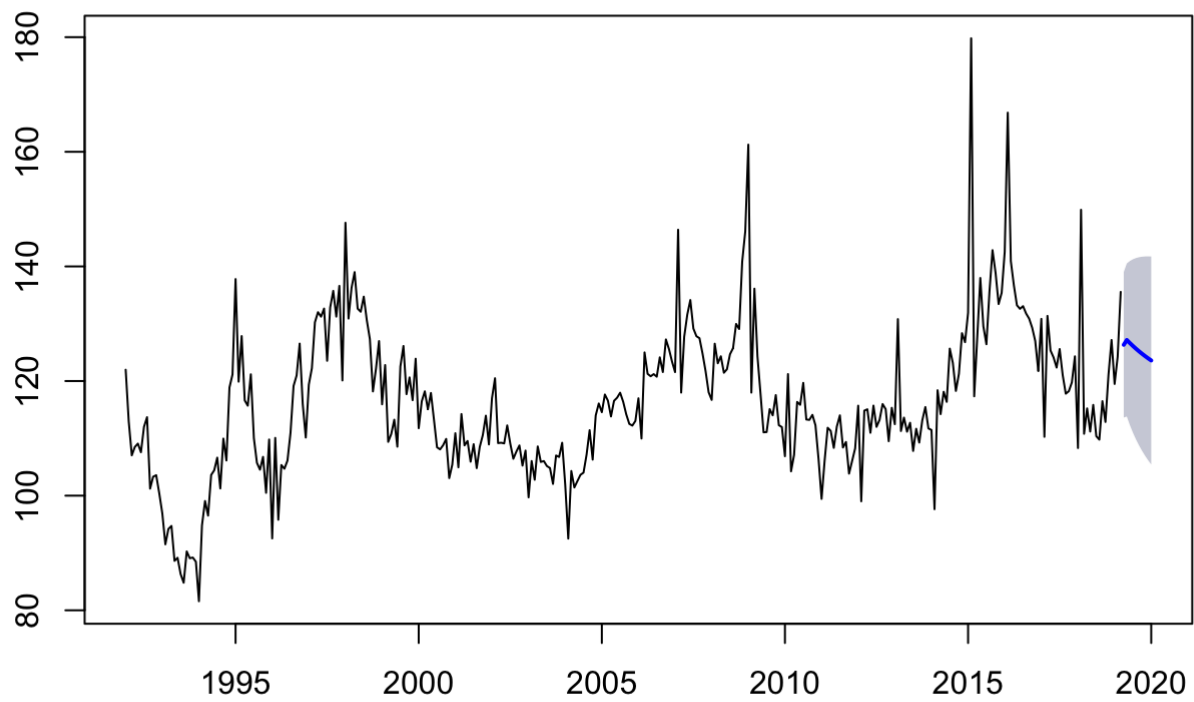
```
plot(forecast1,main="CI=95%")
```

CI=95%



```
plot(forecast2,main="CI=90%")
```

CI=90%





forecast1\$mean

```
##           Jan Feb Mar      Apr      May      Jun      Jul      Aug
## 2019                126.3227 127.1648 126.6308 126.1243 125.6441
## 2020 123.5900
##           Sep      Oct      Nov      Dec
## 2019 125.1886 124.7566 124.3470 123.9585
## 2020
```

forecast1\$upper

```
##           Jan Feb Mar      Apr      May      Jun      Jul      Aug
## 2019                141.4443 143.0700 143.6800 144.1406 144.4877
## 2020 145.2065
##           Sep      Oct      Nov      Dec
## 2019 144.7465 144.9353 145.0682 145.1558
## 2020
```

forecast1\$lower

```
##           Jan Feb Mar      Apr      May      Jun      Jul      Aug
## 2019                111.2012 111.2595 109.5815 108.1081 106.8005
## 2020 101.9735
##           Sep      Oct      Nov      Dec
## 2019 105.6307 104.5779 103.6258 102.7612
## 2020
```

forecast2\$mean

```
##           Jan Feb Mar      Apr      May      Jun      Jul      Aug
## 2019                126.3227 127.1648 126.6308 126.1243 125.6441
## 2020 123.5900
##           Sep      Oct      Nov      Dec
## 2019 125.1886 124.7566 124.3470 123.9585
## 2020
```

forecast2\$upper

```
##           Jan Feb Mar      Apr      May      Jun      Jul      Aug
## 2019                139.0132 140.5129 140.9390 141.2441 141.4581
## 2020 141.7312
##           Sep      Oct      Nov      Dec
## 2019 141.6021 141.6911 141.7368 141.7478
## 2020
```

forecast2\$lower

##	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug
## 2019				113.6323	113.8166	112.3226	111.0046	109.8300
## 2020	105.4488							
##	Sep	Oct	Nov	Dec				
## 2019	108.7751	107.8221	106.9572	106.1691				
## 2020								