Homework#4

12032924 李熹成

```r
#1. Plotting with ggplot2
library(ggplot2)
library(dplyr)
library(tidyr)
library(forecast)

hydro<-read.csv(file = 'hydrodata.csv',header = T)
hydro_tbl<-as_tibble(hydro) %>%
  mutate(id = factor(id, ordered = TRUE),t=as.Date(t))

glimpse(hydro_tbl)

ggplot(hydro_tbl, aes(x =id, y =q, fill = id)) +
  geom_boxplot() +
  theme_classic()+
  theme_bw() +
  theme(plot.title=element_text(size=15, face="bold"),
        axis.text.x=element_text(size=10),
        axis.text.y=element_text(size=10),
        axis.title.x=element_text(size=10),
        axis.title.y=element_text(size=10)) +
  scale_color_discrete(name="Station") +
  labs(title="Daily flux of Yellow River in 2017-2020",
       x="Station", y="Quantity(m^3 s)",fill='Station name')
```

```r
ggplot(hydro_tbl,aes(x=t,y=q,color=id))+
  geom_line()+
  theme_bw() +
  theme(plot.title=element_text(size=15, face="bold"),
        axis.text.x=element_text(size=10),
        axis.text.y=element_text(size=10),
        axis.title.x=element_text(size=10),
        axis.title.y=element_text(size=10)) +
  scale_color_discrete(name="Station") +
  labs(title="Monthly sum flux of Yellow River in 2019-2020 in Tongguan",
       x="Year", y="Quantity(m^3 s)")+
  facet_wrap( ~ id)

hydro_tbl %>%
  mutate(year=substr(t,1,4)) %>%
  filter(id=='tongguan'&year=='2019') %>%
  ggplot(aes(q)) +
  geom_histogram(bins = 50) +
  theme_bw() +
  theme(plot.title=element_text(size=15, face="bold"),
        axis.text.x=element_text(size=10),
        axis.text.y=element_text(size=10),
        axis.title.x=element_text(size=10),
        axis.title.y=element_text(size=10)) +
  labs(title="Histogram of flux of Yellow River in 2019 in Tongguan",
       x="Quantity(m^3 s)",y='Number of days')
```

```r
hydro_tbl %>%
  mutate(year=substr(t,1,4)) %>%
  filter(year=='2019') %>%
  ggplot(aes(x=t,y=q,color=id)) +
  geom_point(size=0.5) +
  theme_bw() +
  theme(plot.title=element_text(size=15, face="bold"),
        axis.text.x=element_text(size=10),
        axis.text.y=element_text(size=10),
        axis.title.x=element_text(size=10),
        axis.title.y=element_text(size=10)) +
  scale_color_discrete(name="Station")+
  labs(title="Scatter plot of flux of Yellow River in 2019",
       x="time",y='Quantity(m^3 s)')
```

```r
library(fields)
library(maps)
library(RNetCDF)
ex.nc      <- open.nc("IUPB_s5p_201806_global_totalBrOVC.NC")
print.nc(ex.nc)
Lat        <- var.get.nc(ex.nc, "latitude")
Lon        <- var.get.nc(ex.nc, "longitude")
total_BrO_VC      <- var.get.nc(ex.nc, "total_BrO_VC")
close.nc(ex.nc)

par(mar=c(4.5,3,2,1))
image.plot(Lon, Lat, total_BrO_VC,
           horizontal=T, useRaster=T,
           legend.shrink=0.75, axis.args=list(cex.axis = 1.25),
           legend.width=1, legend.mar=2,
           legend.args=list(text="Toal BrO Vertical Column [molec cm^{-2}]",
                            cex=1.25),
           xlab='',ylab='',midpoint=T, axes=F, ann=F
)
title(xlab="",cex.lab=1.25,font.lab=2)
axis(1,at=pretty(Lon),tck=-0.015,lwd=1,cex.axis=1.25,font=1)
title(ylab="",cex.lab=1.25,font.lab=2)
axis(2,at=pretty(Lat),tck=-0.015,lwd=1,cex.axis=1.25,font=1,las=1)
title(main=paste("Toal BrO Vertical Column in Jun. 2018"),
      cex.main=1,font.main=2)
```

```r
# Add map
map('world',add=T,lwd=0.75,col="black")

# Add a box
box(lwd=2)
```

```r
#2. Analysis of the time series of monthly temperature
Baoan<-read.csv(file='2281305.csv',header = T)

Baoan_tbl<-as_tibble(Baoan)

Baoan_temp<-Baoan_tbl %>%
  select(DATE,TMP) %>%
  mutate(ym=substr(DATE,1,7),temp=as.numeric(substr(TMP,1,5)),
         quality=substr(TMP,7,7)) %>%
  filter(ym>='2010-01'&ym<='2020-06'&quality=='1') %>%
  mutate(temp=ifelse(temp=="-9999",NA,temp)) %>%
  group_by(ym) %>%
  summarise(monthly_mean=mean(temp)/10) %>%
  mutate(month=as.Date(paste(ym,'01',sep='-')))


monthly_temp<- ts(Baoan_temp$monthly_mean, start=2010, frequency=12)
plot(monthly_temp,
     type='l',
     xlab='year',
     ylab='temperature(degrees Celsius)',
     main="Monthly average temperature of Bao'an in 2010.1-2020.6 in time series ",
     col = "darkgrey")
box(lwd=2,col="darkgrey")
```

```r
##2.2 Decomposition
monthly_temp_components <- decompose(monthly_temp)
plot(monthly_temp_components)


###Do Box-Ljung test to the result
random<-as.numeric(monthly_temp_components$random)
Box.test(random,type='Ljung',
         lag=log(length(random)))
###Do acf to the result
omit_na_random<-na.omit(random)
rand_acf <- acf(omit_na_random, lag=40,main="white noise")
rand_acf

### Plot hist
hist(monthly_temp_components$random, prob=TRUE,
     main = "Histogram of monthly temperature")
### Add pdf
curve(dnorm(x, mean=mean(monthly_temp_components$random,na.rm=T),
            sd=sd(monthly_temp_components$random,na.rm=T)),
      add=TRUE, col="red")
```

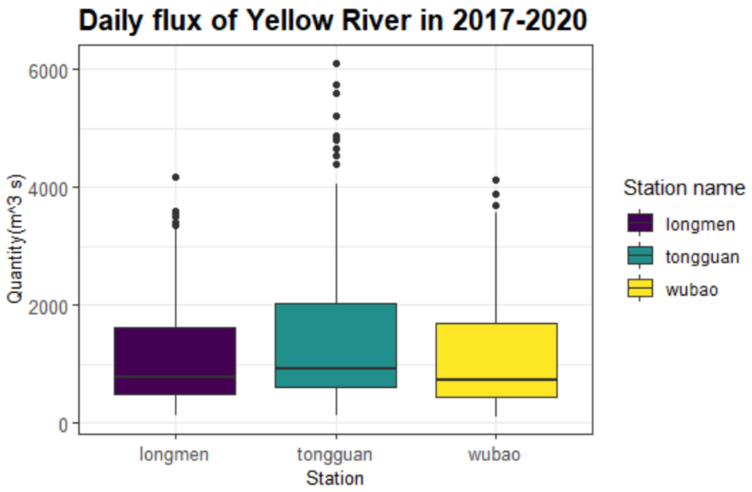
```r
##2.3 Fit an ARIMA(p,d,q) model
# hist(monthly_temp,
#      main = "Histogram of monthly mean temperature",
#      xlab = "Temperature(Degrees Celsius)")

monthly_temp_log<-log(monthly_temp)

# hist(monthly_temp_log,
#      main = "Histogram of log monthly mean temperature",
#      xlab = "Temperature(Degrees Celsius)")

monthly_temp_log_d1 <- diff(monthly_temp_log)
# hist(monthly_temp_log_d1,
#      main = "Histogram of difference of log monthly mean temperature",
#      xlab = "Temperature(Degrees Celsius)")


# Automated forecasting using an ARIMA model
model1 <- auto.arima(monthly_temp,trace=T)
model2 <- auto.arima(monthly_temp_log,trace=T)
model3 <- auto.arima(monthly_temp_log_d1,trace=T)

# Check acf and pacf
acf(monthly_temp_log)
pacf(monthly_temp_log)
```

```r
# 模型评价###############################
##source:https://blog.csdn.net/mr_muli/article/details/82779250
qqnorm(model2$residuals)
qqline(model2$residuals)
Box.test(model2$residuals,type="Ljung-Box")

## 2.5 Make predictions
month_forecast  <- 5
month_in_plot   <- 10
forecast <- forecast(model2, month_forecast)

# Plot predictions along with real values
plot(forecast, include = month_in_plot, xlab="Time",
     ylab="log(Monthly mean)",type="o",lwd=2)
```

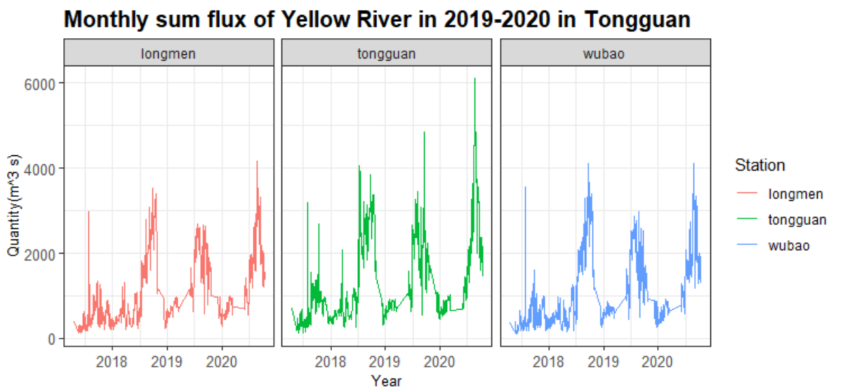
```r
# Get predicted values

# 2020-07
exp(forecast$mean[1])
exp(forecast$lower[1,1])
exp(forecast$upper[1,1])

# 2020-08
exp(forecast$mean[2])
exp(forecast$lower[2,1])
exp(forecast$upper[2,1])

# Verify the predictions
Baoan_temp2<-Baoan_tbl %>%
  select(DATE,TMP) %>%
  mutate(ym=substr(DATE,1,7),temp=as.numeric(substr(TMP,1,5)),
         quality=substr(TMP,7,7)) %>%
  filter(ym>='2010-01'&ym<='2020-08'&quality=='1') %>%
  mutate(temp=ifelse(temp=="-9999",NA,temp)) %>%
  group_by(ym) %>%
  summarise(monthly_mean=mean(temp)/10) %>%
  mutate(month=as.Date(paste(ym,'01',sep='-')))

tail(Baoan_temp2)
```

Problem#1

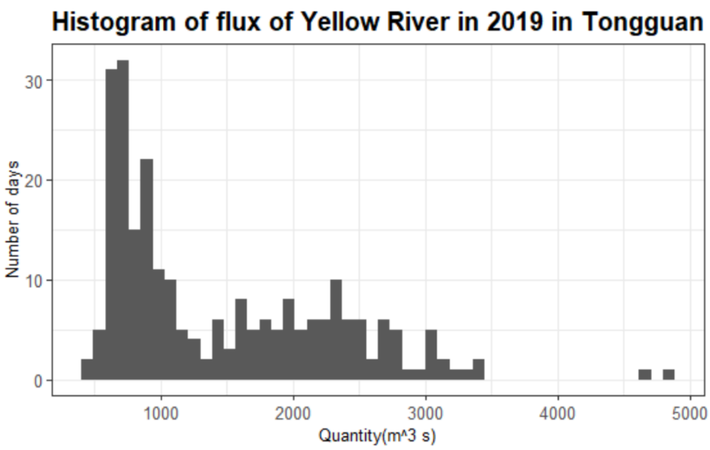
Boxplot



Time series



Histogram



Scatter plot

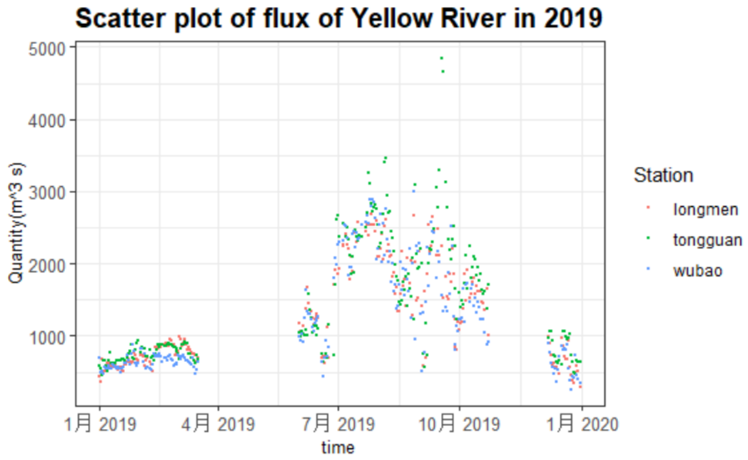
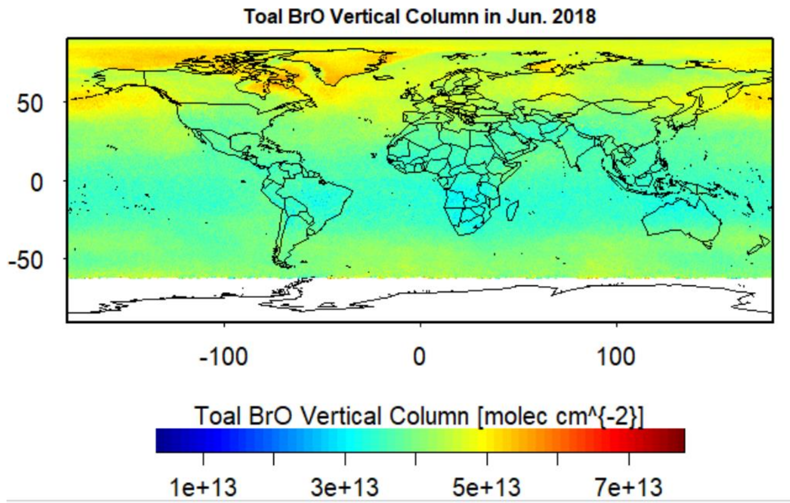


Image plot:

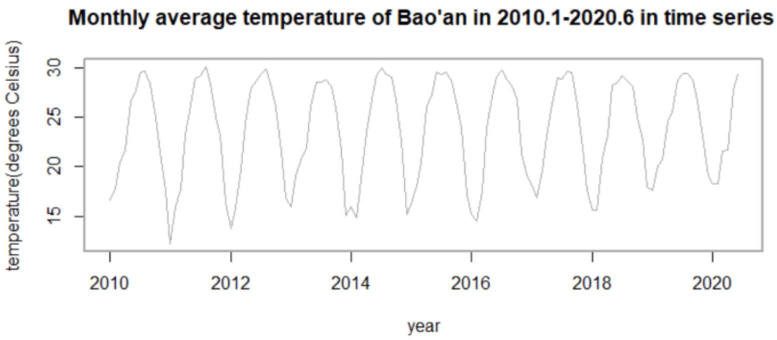


Problem#2

There will be a need of prediction for Sept. 2020 and Nov. 2020 but the dataset has only data till 2020-09-11. Thus, I choose July. 2020 and Aug. 2020 as the test month.

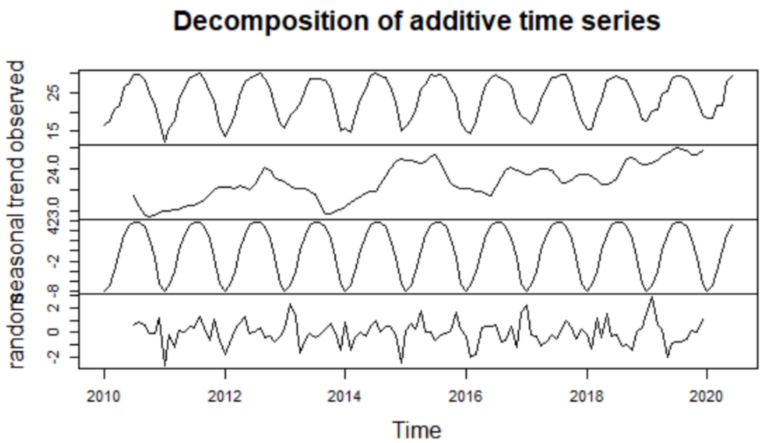
2.1

Construct a time series of monthly-averaged temperature from 2010 Jan. to 2020 Jun.



2.2

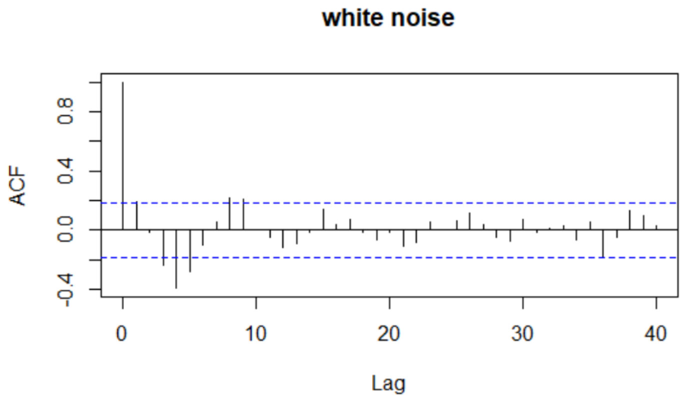
The decomposition are as below and



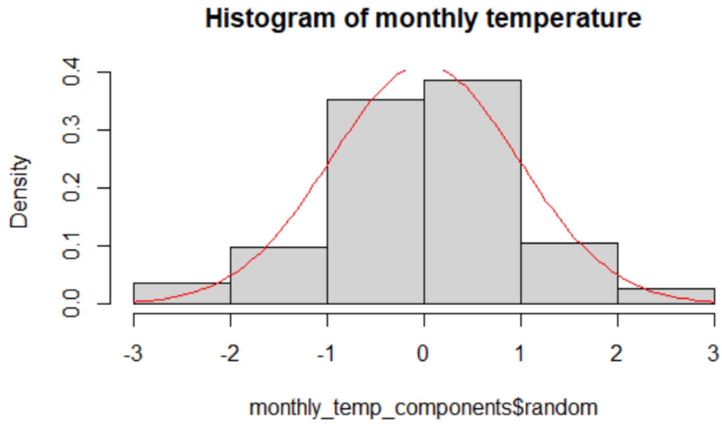
From the graph above the plot of random term seems random. But it reject the Box-Ljung test with the p-value = 1.587e-05:

```
              Box-Ljung test

data:  random
X-squared = 29.463, df = 4.8363, p-value = 1.587e-05
```

And the *acf* shows a little autocorrelation within each other:



The random obey the normal distribution:



Thus, it can be concluded that the error part follows a white noise distribution.

2.3

The ARIMA model for monthly temperature is as below:

The sigma^2 is 1.279

```
Series: monthly_temp
ARIMA(1,0,0)(1,1,1)[12] with drift

Coefficients:
         ar1      sar1     sma1   drift
      0.2066   -0.1072  -0.8159  0.0087
s.e.  0.0935    0.1554   0.1969  0.0035

sigma^2 estimated as 1.279:  log likelihood=-181.32
AIC=372.65   AICc=373.2   BIC=386.33
```

The ARIMA model for log monthly temperature is as below:

The sigma^2 is 0.003773

```
Series: monthly_temp_log
ARIMA(0,0,2)(0,1,1)[12] with drift

Coefficients:
         ma1     ma2     sma1  drift
      0.2487  0.1956  -0.8042  5e-04
s.e.  0.0903  0.1054   0.1181  3e-04

sigma^2 estimated as 0.003773:  log likelihood=152.1
AIC=-294.2   AICc=-293.64   BIC=-280.52
```

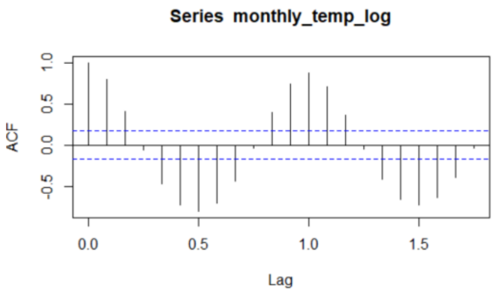
The ARIMA model for difference of log monthly temperature is as below:

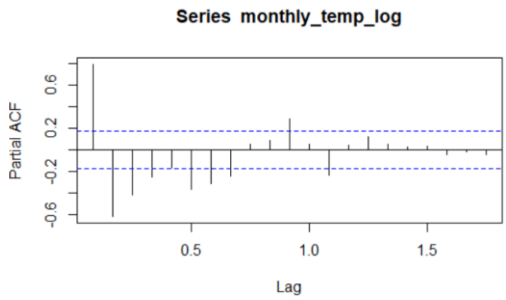
The sigma^2 is 0.005097

```
Series: monthly_temp_log_d1
ARIMA(1,0,0)(0,1,1)[12]

Coefficients:
          ar1     sma1
      -0.4100  -0.7332
s.e.   0.0856   0.1128

sigma^2 estimated as 0.005097:  log likelihood=134.23
AIC=-262.46   AICc=-262.24   BIC=-254.27
```

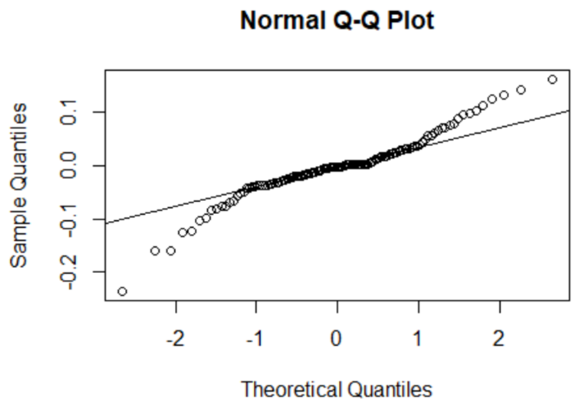
So, we choose model 2 which is log monthly mean temperature.

The *acf* and *pacf* is as below:





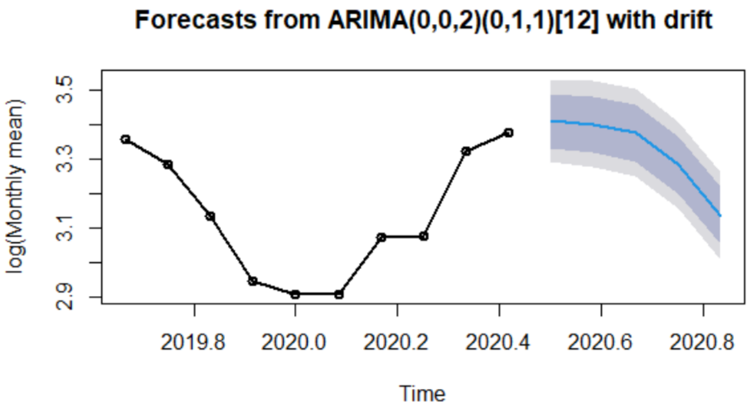
The model residual test is as below:




```
        Box-Ljung test

data:  model2$residuals
X-squared = 0.00056829, df = 1, p-value = 0.981
```

The model passes the test.

2.4



Forecast for 2020.07:

```
> exp(forecast$mean[1])
[1] 30.34925
> exp(forecast$lower[1,1])
     80%
28.04663
> exp(forecast$upper[1,1])
     80%
32.84092
```

Forecast for 2020.08

```
> exp(forecast$mean[2])
[1] 30.03242
> exp(forecast$lower[2,1])
     80%
27.68724
> exp(forecast$upper[2,1])
     80%
32.57625
```

Verify with the real value:

	ym	monthly_mean	month
	<chr>	<dbl>	<date>
1	2020-03	21.6	2020-03-01
2	2020-04	21.8	2020-04-01
3	2020-05	27.8	2020-05-01
4	2020-06	29.3	2020-06-01
5	2020-07	30.3	2020-07-01
6	2020-08	29.3	2020-08-01

The Relative bias for July is: $\frac{30.34925-30.3}{30.34925} = 0.0016 = 0.16\%$

The Relative bias for August is: $\frac{30.03242-29.3}{30.03242} = 0.0244 = 2.44\%$