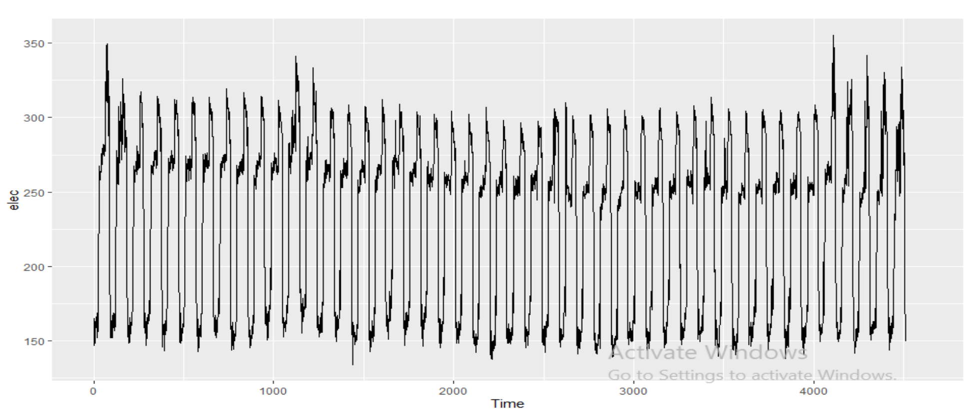
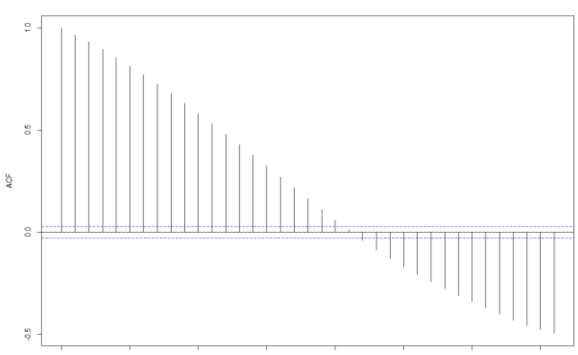
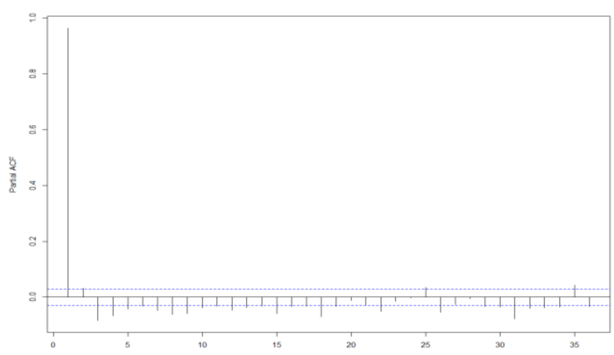
Assignment - Time Series Forecasting

```
#Loading necessary packages
library(forecast)
library(ggplot2)

#loading the data
data=read.csv('C:\\Users\\gt\\Desktop\\Elec-train.csv')

#Creating the power time series
elec=ts(data$Power..kW.)
autoplot(elec)
```



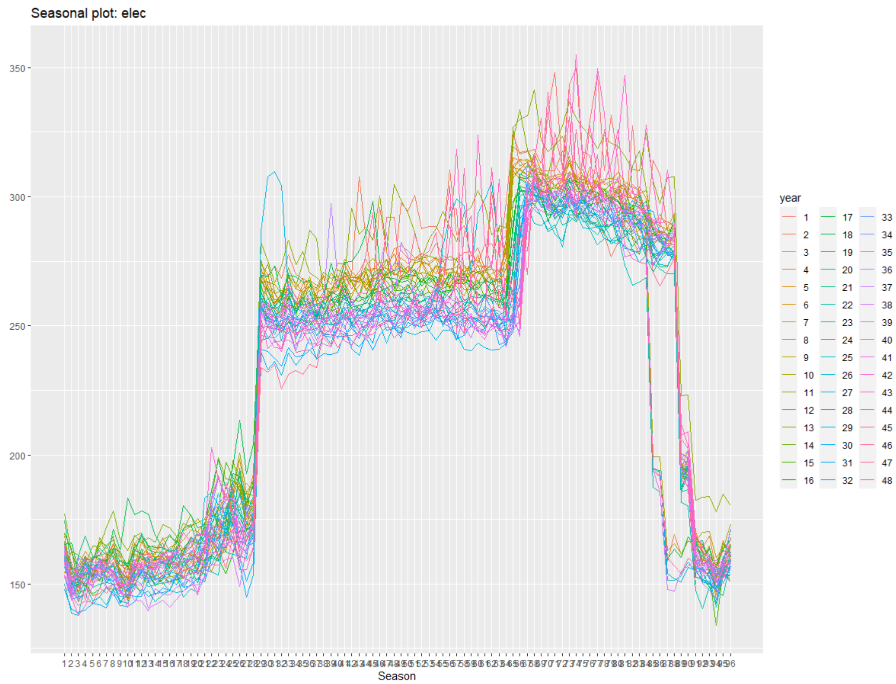

```
#We see a seasonal pattern we can check the ACF and PACF
tmp=acf(elec , na.action = na.pass,plot=FALSE)
autoplot(tmp)
```





#the ACF shows a pattern and a trend and checking the excel file there seems to be a peek around 19:00 for each day of the 48 days.

```
#Adding the frequency
elec=ts(data$Power..kW.,frequency = 96)

#Confirming the seasonality
ggseasonplot(elec)
```



Forecasting without covariates

HoltWinters exponential smoothing

```
#splitting the series into train and test to find the best model
train=head(elec,n=4411)
t=head(elec,n=4507)
test=tail(t,n=96)

#We see a seasonal pattern probably additive
fith=hw(train,seasonal = 'additive')
pred=forecast(fith,h=96)
autoplot(test)+autolayer(pred$mean,series="HW without covariate")
#finding the RMSE
print(sqrt(mean(pred$mean-test)^2))
```

unfortunately I get an Error in ets(x, "AAA", alpha = alpha, beta = beta, gamma = gamma, phi = phi, :

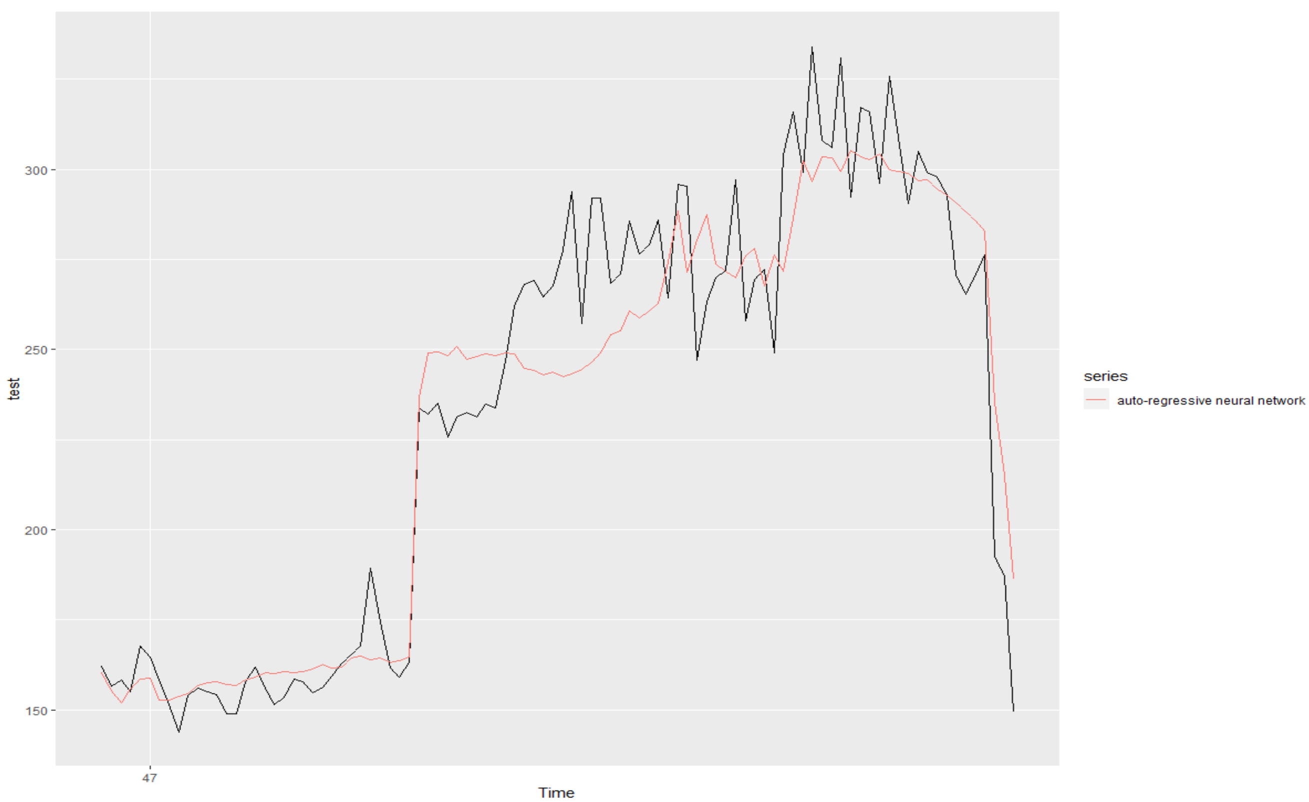
Frequency too high which indicates that HoltWinters is the wrong model for this series.

```
#HoltWinters multiplicative exponential smoothing
fitm=hw(train,seasonal = 'multiplicative')
predm=forecast(fitm,h=96)
autoplot(test)+autolayer(predm$mean,series="HW without covariate")
#finding the RMSE
print(sqrt(mean(predm$mean-test)^2))
```

unfortunately I get an Error in ets(x, "AAA", alpha = alpha, beta = beta, gamma = gamma, phi = phi, :
Frequency too high which indicates that HoltWinters is the wrong model for this series.

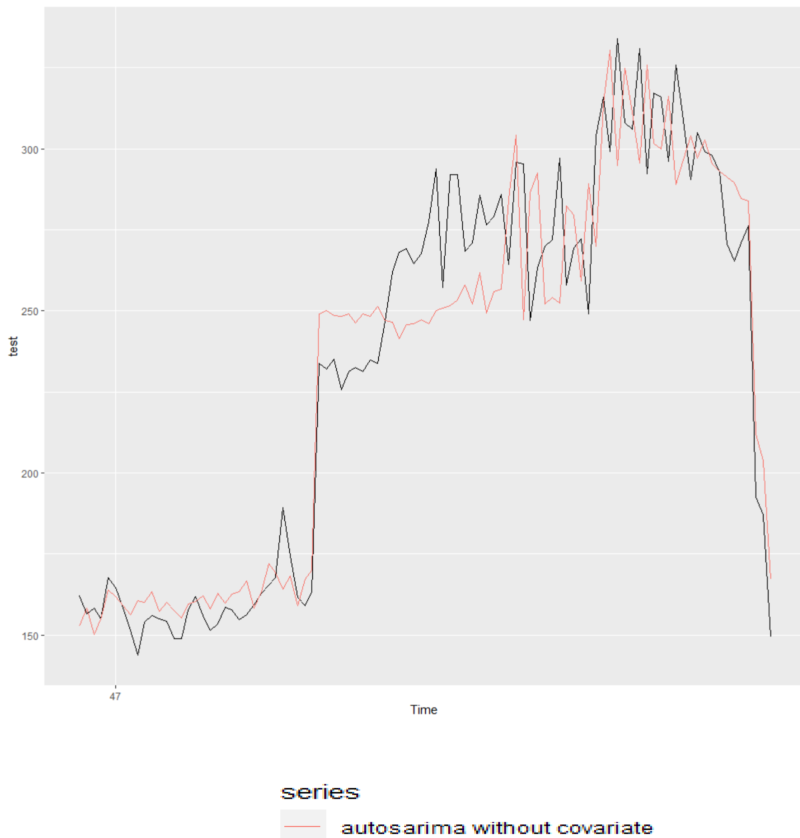
#Forecasting with auto-regressive neural network

```
fitn=nnetar(train,lambda='auto')
fitn    #Model:  NNAR(11,1,6)[96]
prevn=forecast(fitn,h=96)
#finding the RMSE
print(sqrt(mean(prevn$mean-test)^2)) #2.163596
autoplot(test)+autolayer(prevn$mean,series='auto-regressive neural network ')
```



SARIMA MODEL

```
sarima=auto.arima(train)
sarima  #ARIMA(5,0,0)(0,1,0)[96]  #AIC=32639.92
prevs=forecast(sarima,h=96)
#finding the RMSE
print(sqrt(mean(prevs$mean-test)^2)) #1.067773
autoplot(test)+autolayer(prevs$mean,series="autosarima without covariate")
```



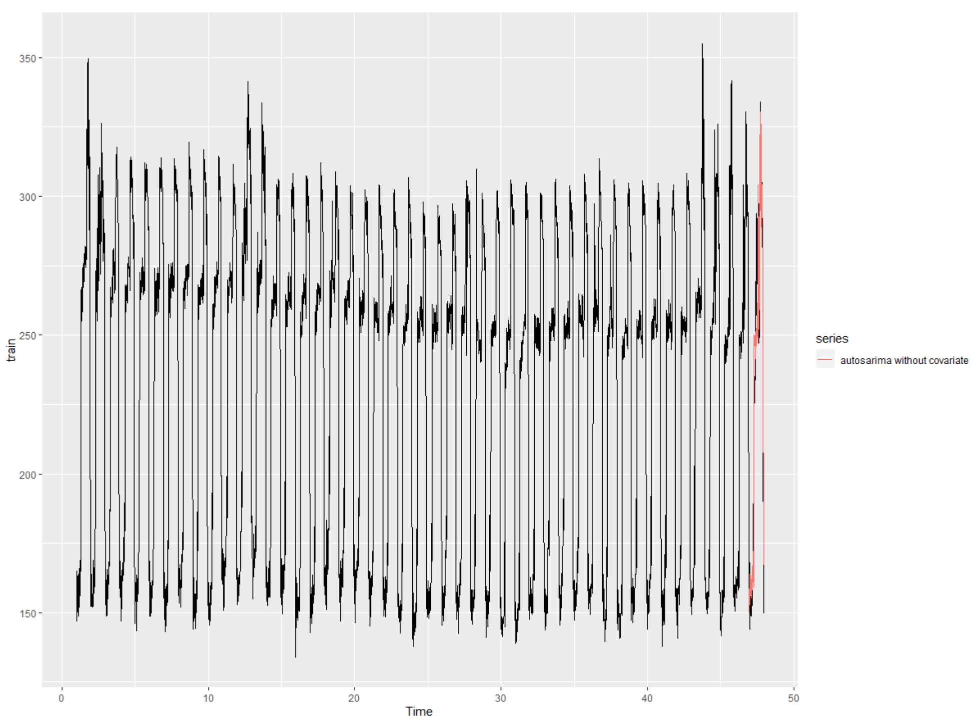
#According to the RMSE Auto SARIMA Model is a better model than auto-regressive neural network

```
#we will focus on finding the best SARIMA Model using the 2/17/2010 as test data and the AIC
#splitting the series into train and test
train=head(elec,n=4507)
test=tail(elec,n=96) #NA values
```

#Auto SARIMA MODEL

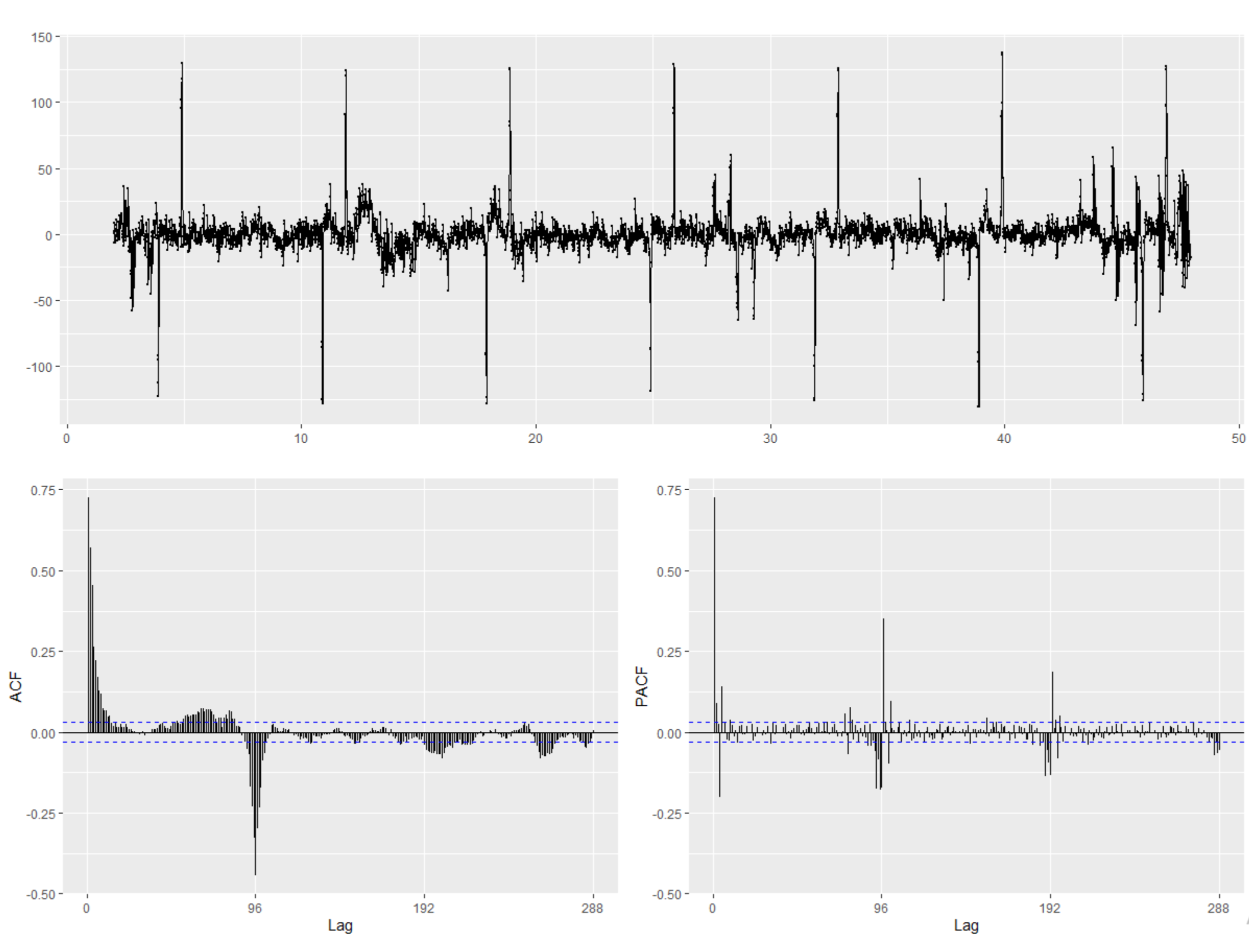
```
sarima=auto.arima(train) #ARIMA(5,0,0)(0,1,0)[96]
sarima #AIC=33583.29
prevs=forecast(sarima,h=96)
autoplot(train)+autolayer(prevs$mean,series="autosarima without covariate")
checkresiduals(sarima)
```

```
Ljung-Box test

data:  Residuals from ARIMA(5,0,0)(0,1,0)[96]
Q* = 1421.1, df = 187, p-value < 2.2e-16
```

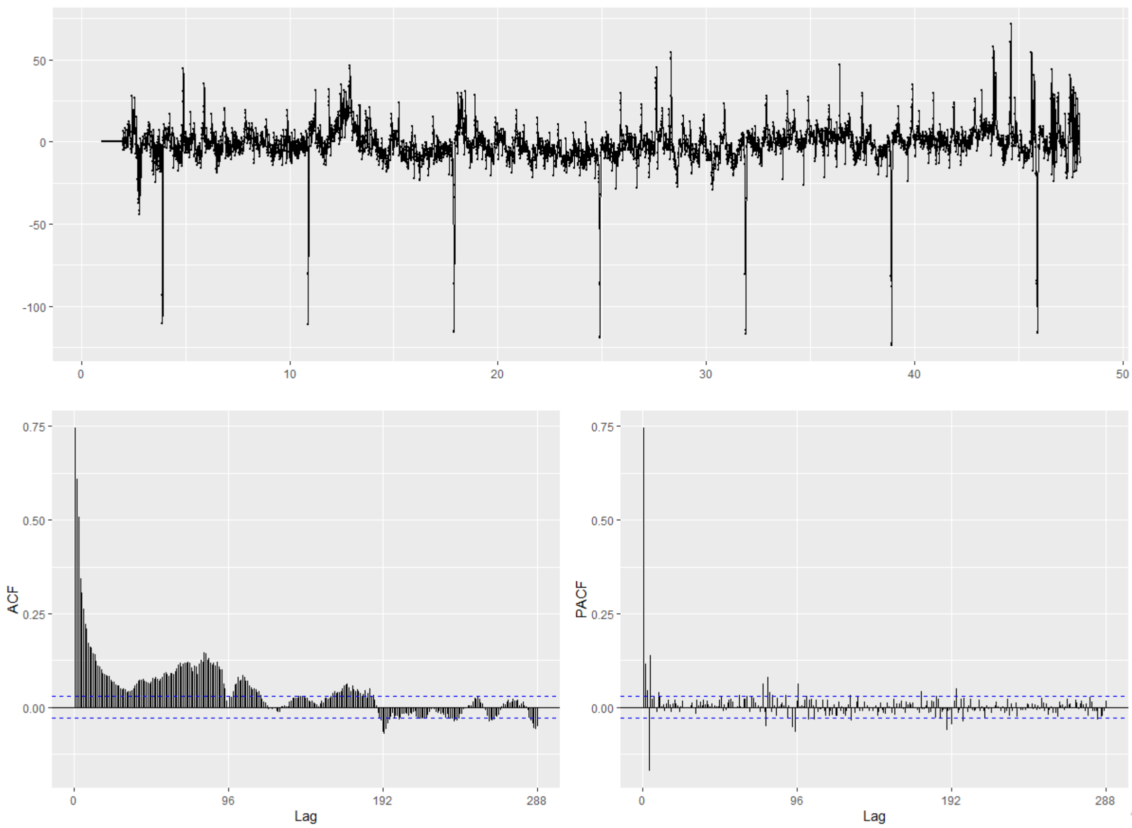


#Manually choosing the order of SARIMA

```
#we start by differeciating to remove the seasonal pattern
train %>% diff(lag=96) %>% ggtsdisplay() #significant lag at 96 in the ACF and exponential decay of the seasonal lags in the PACF suggests a seasonal MA 1
```



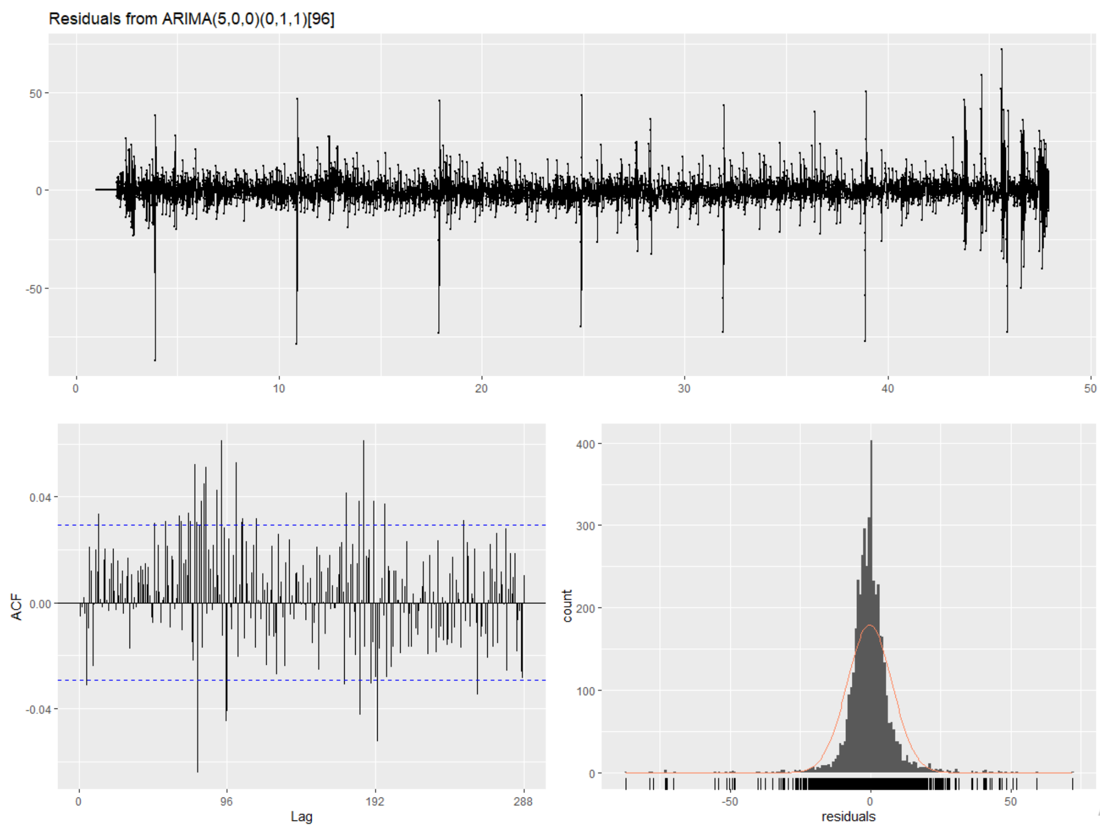

```
fitm=Arima(train,order=c(0,0,0),seasonal=c(0,1,1))
fitm #AIC=35019.64 worse than the auto.arima
fitm %>% residuals()%>% ggtsdisplay()
```



#There is still significant ACF lags. We can add some additional non-seasonal terms in the Arima model

```
fitm1=Arima(train,order = c(5,0,0),seasonal = c(0,1,1))
fitm1 #AIC=31111.22 is better than auto.arima and fitm
checkresiduals(fitm1)
```

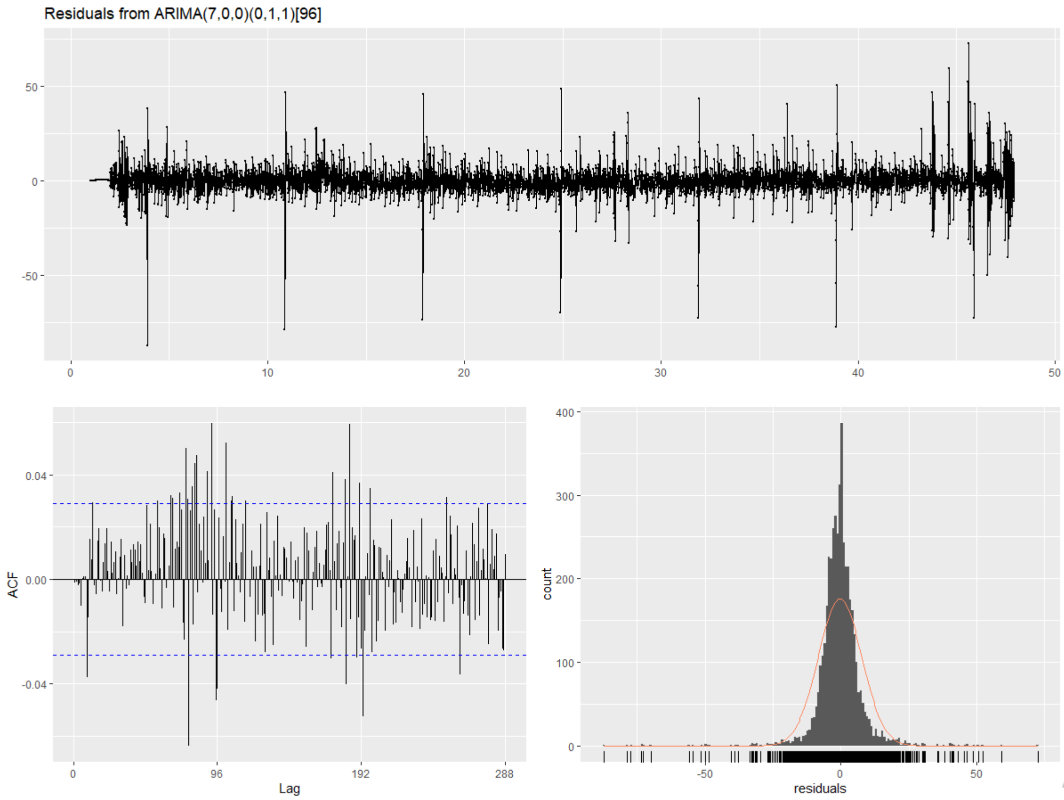
```
Ljung-Box test

data:  Residuals from ARIMA(5,0,0)(0,1,1)[96]
Q* = 356.71, df = 186, p-value = 7.503e-13
```




```
fitm2=Arima(train,order = c(7,0,0),seasonal = c(0,1,1))
fitm2  #AIC=31108.1  better than auto.arima, fitm, fitm1
checkresiduals(fitm2)
```

```
Ljung-Box test

data:  Residuals from ARIMA(7,0,0)(0,1,1)[96]
Q* = 337.9, df = 184, p-value = 3.558e-11
```

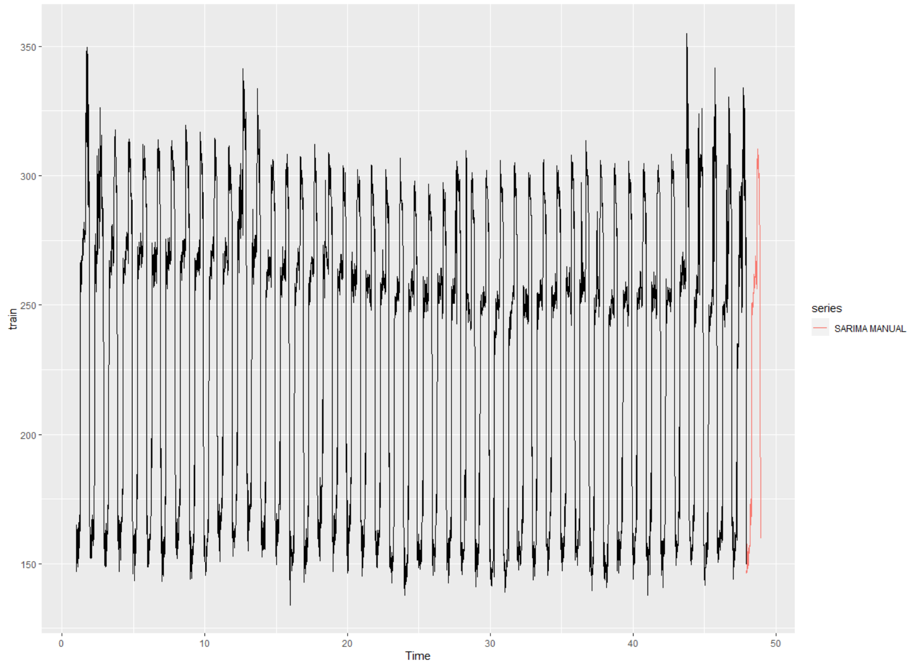
```
#The p-value should be bigger Than 5% for the residuals to be white noise for
our model to have captured All correlations but we have too much data .
```



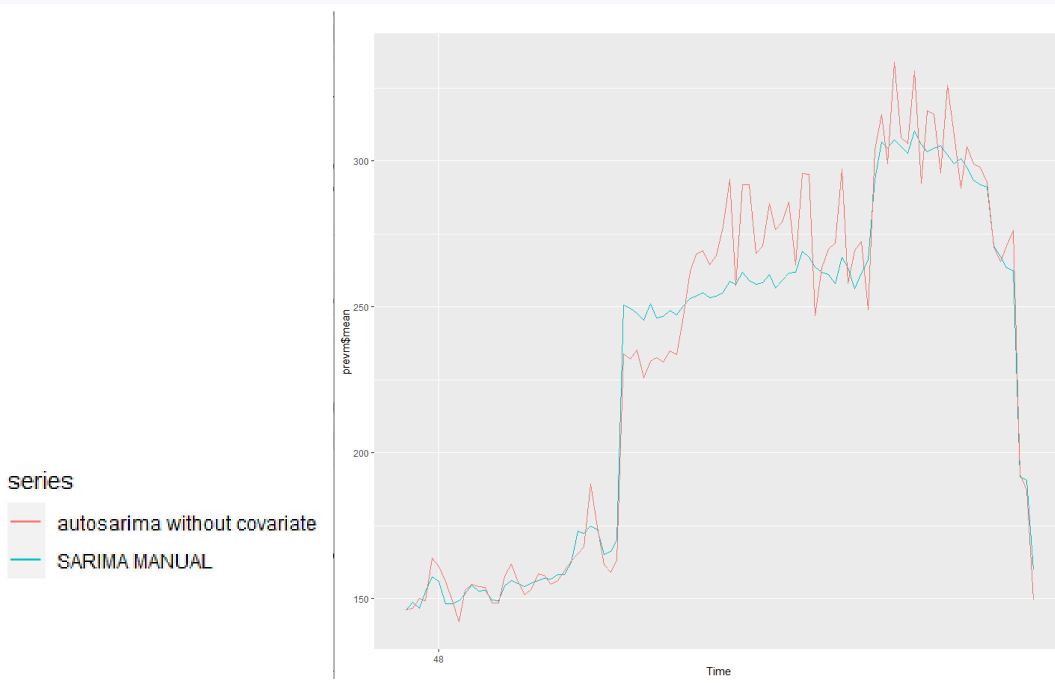

```
fitm3=Arima(train,order = c(8,0,0),seasonal = c(0,1,1))
fitm3 #AIC=31110.09  #worse than fitm2 so I stop because adding more in the seasonal and nonseasonal freezes the computer .
```

Forecasting the 2/17/2010 power consumption without covariates

```
#Forecasting with  ARIMA(7,0,0)(0,1,1)[96]
prevm=forecast(fitm2,h=96)
df=as.data.frame(prevm$mean, row.names = NULL)
View(df)
autoplot(train)+autolayer(prevm$mean,series='SARIMA MANUAL')
```



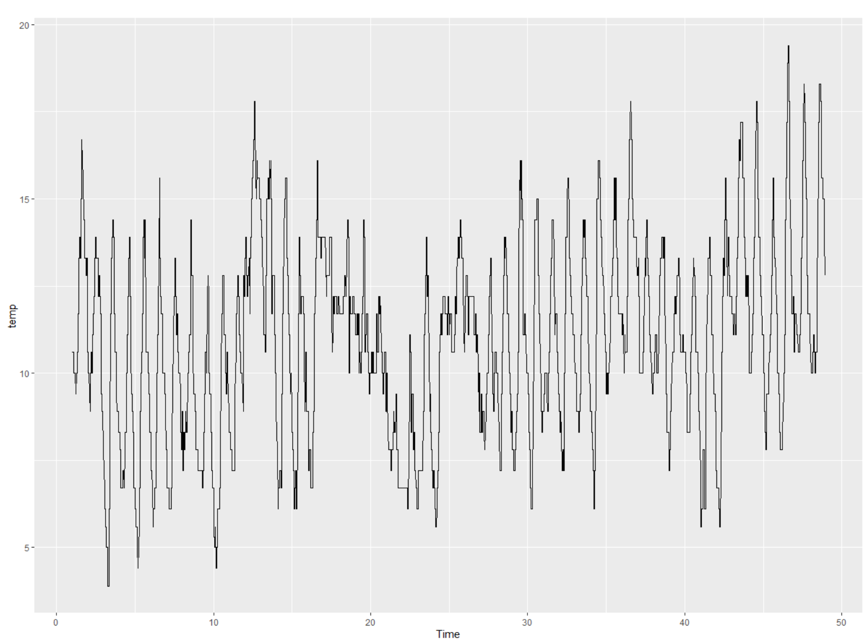

```
#comparing the models
autoplot(prevm$mean,series='SARIMA MANUAL')+autolayer(prevs$mean,series="autosarima without covariate")
```



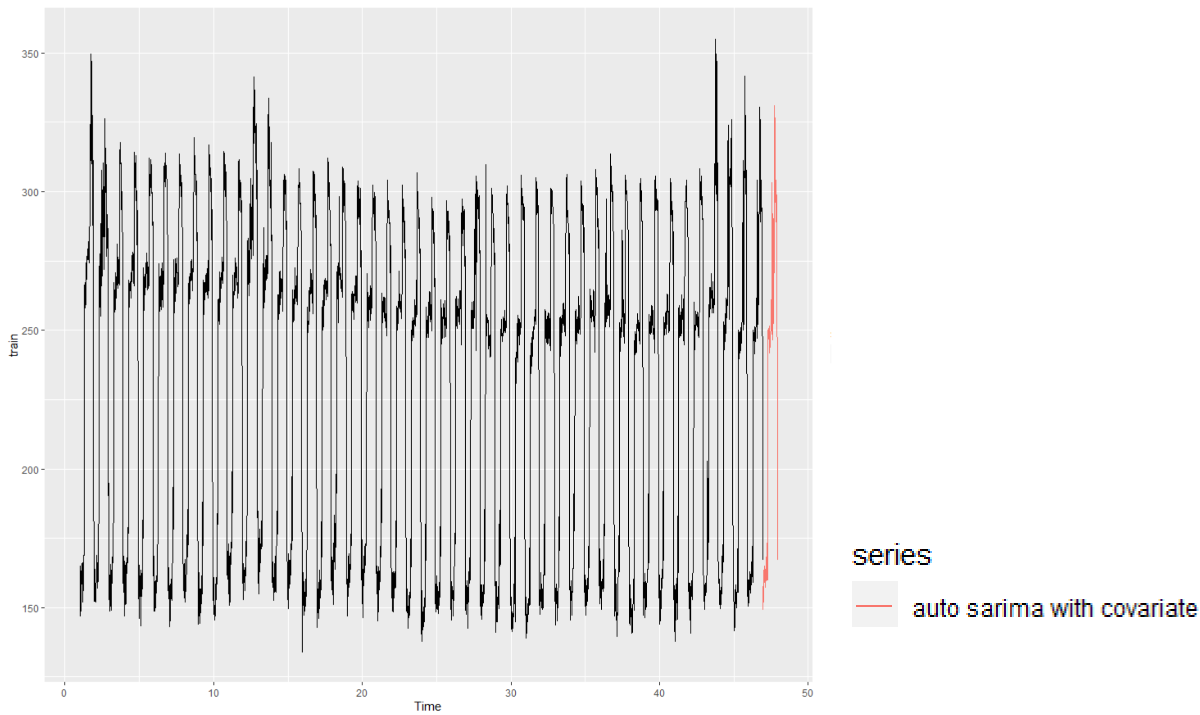
Forecasting with covariates

#Dynamic regression model

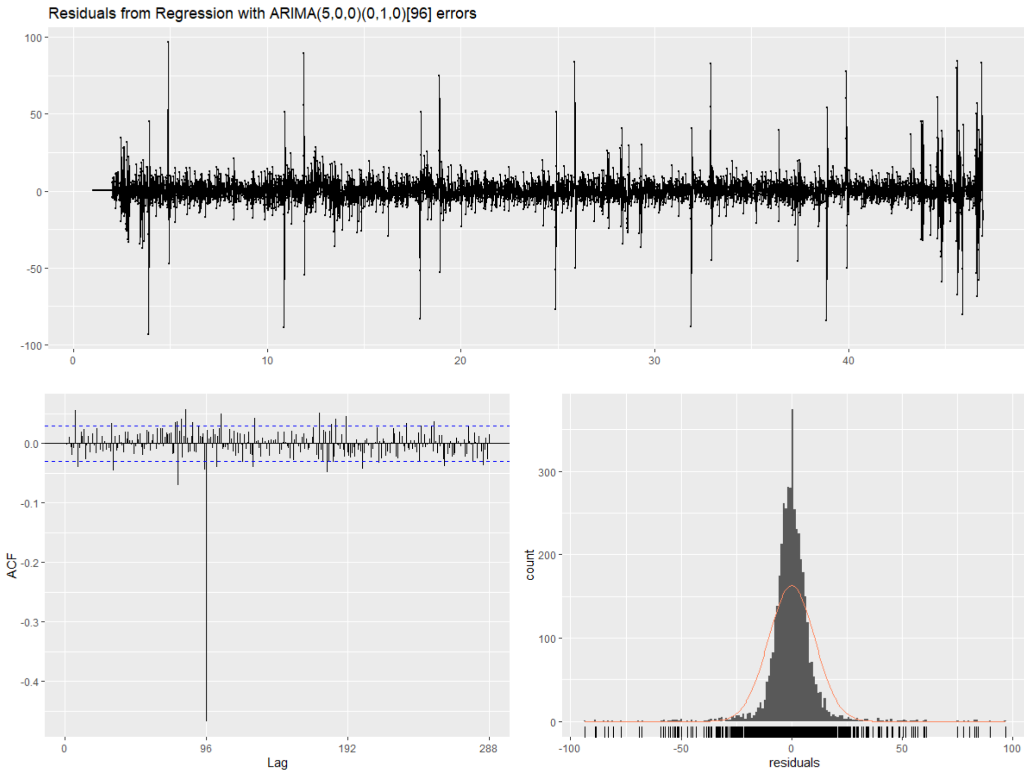
```
#REsplitting the power series into train and test to be able to compute RMSE
train=head(elec,n=4411)
t=head(elec,n=4507)
test=tail(t,n=96)

#creating the temperature time serie
temp=ts(data$Temp..C..,frequency = 96)
autoplot(temp)


#splitting the temperature serie into train and test
train_temp=head(temp,n=4411)
ttemp=head(temp,n=4507)
test_temp=tail(ttemp,n=96)
```




```
#use a dynamic regression model for forecasting power consumption (kW), using Outdoor temperature as external covariate.
sarima_co=auto.arima(train,xreg=train_temp) #ARIMA(5,0,0)(0,1,0)[96]
sarima_co    #AIC=32636.82
prevsco=forecast(sarima_co,h=96,xreg=test_temp)
autoplot(train)+autolayer(prevsco$mean,series="auto sarima with covariate")
```




```
#Finding the RMSE
print(sqrt(mean(prevsco$mean-test)^2)) # 0.6935747 a huge improvement than without covariates
summary(sarima_co)
checkresiduals(sarima_co) #there seems to be a big lag at 96 (seasonal period) in the ACF so we can add this to our SARIMA #ARIMA(5,0,0)(0,1,1)[96]
```



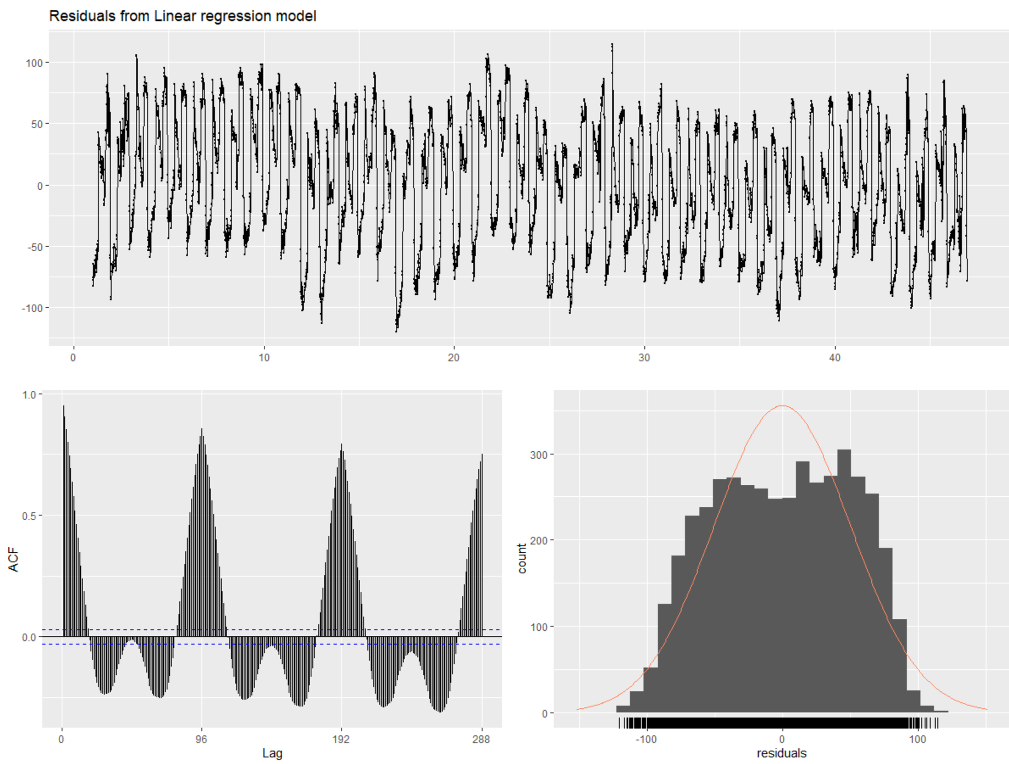

```
plot(pacf(sarima_co$residuals))
```

adding the temperature covariate allows us to improve the forecasting.
But if we check the residual, there is still some autocorrelations.

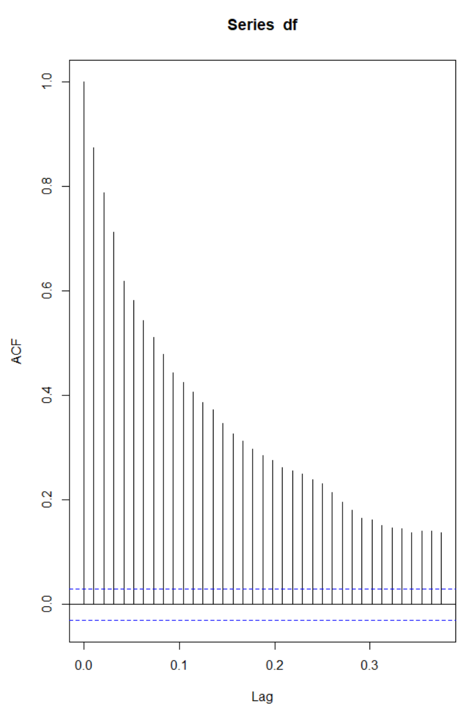
```
#We can try to find a better model manually
plot(data$Power..kW.,data$Temp..C..,col='red',type = 'p') #there doesn't seem to be any relationship
fitco2=tslm(train~train_temp)
summary(fitco2) # the feature temperature seems significant as small p-value
```

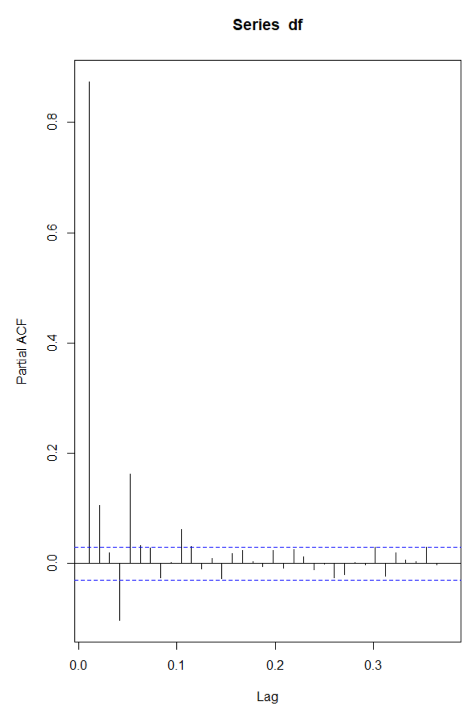
```
Call:
tslm(formula = train ~ train_temp)

Residuals:
    Min      1Q  Median      3Q     Max 
-120.23  -41.79    1.72   42.67  114.67 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 122.4807     3.1240   39.21   <2e-16 ***
train_temp   10.1044     0.2807   36.00   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 50.54 on 4409 degrees of freedom
Multiple R-squared:  0.2271,	Adjusted R-squared:  0.227 
F-statistic:  1296 on 1 and 4409 DF,  p-value: < 2.2e-16
```

```
checkresiduals(fitco2)
```

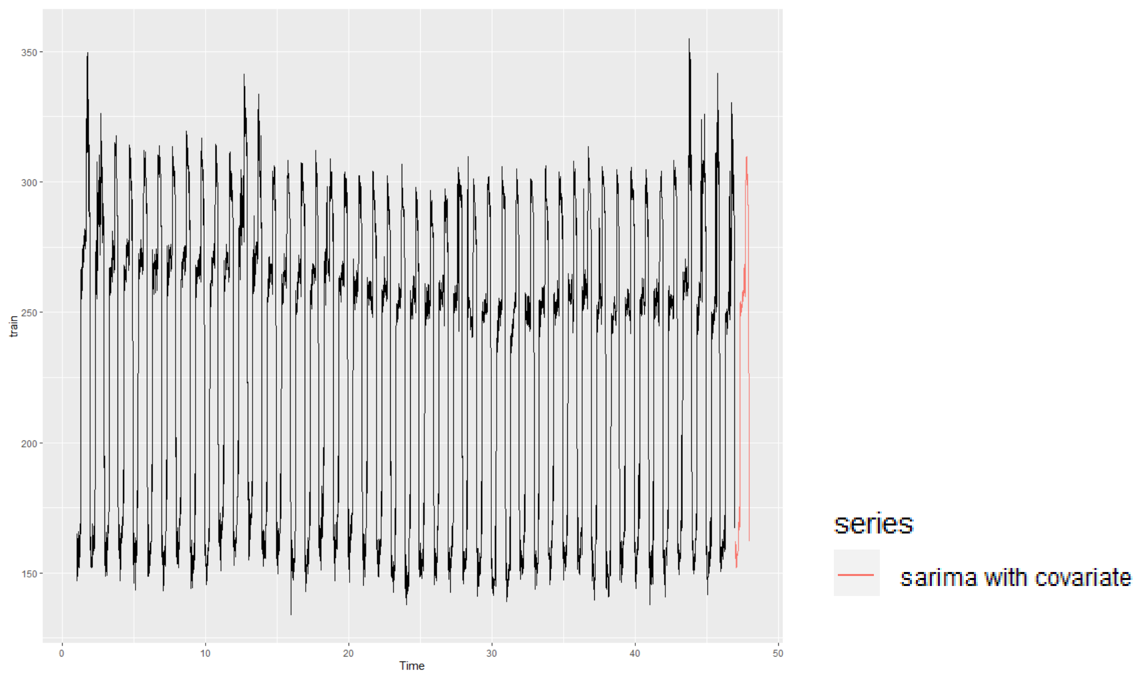



```
plot(pacf(fitco2$residuals))
#Here the residuals are highly correlated and the ACF suggests a seasonal pattern
df=diff(fitco2$residuals,lag=96)
acf(df) #exponential decay
pacf(df)#a lag at 5
#Exponential decay in the ACF and a huge lag at 5 in the PACF this suggests non seasonal AR 5
```

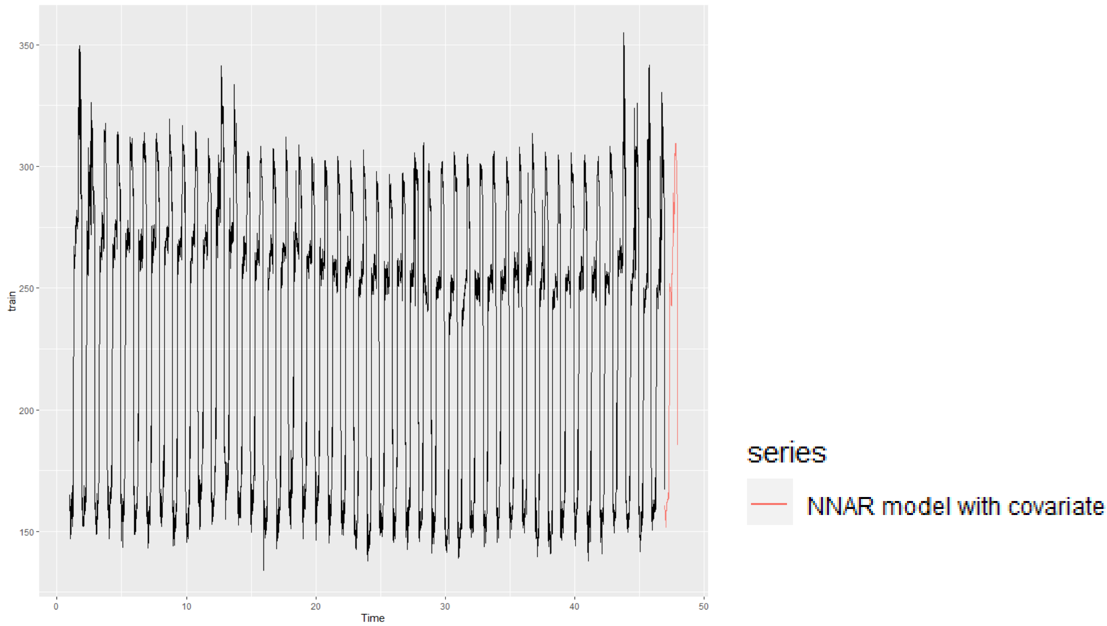




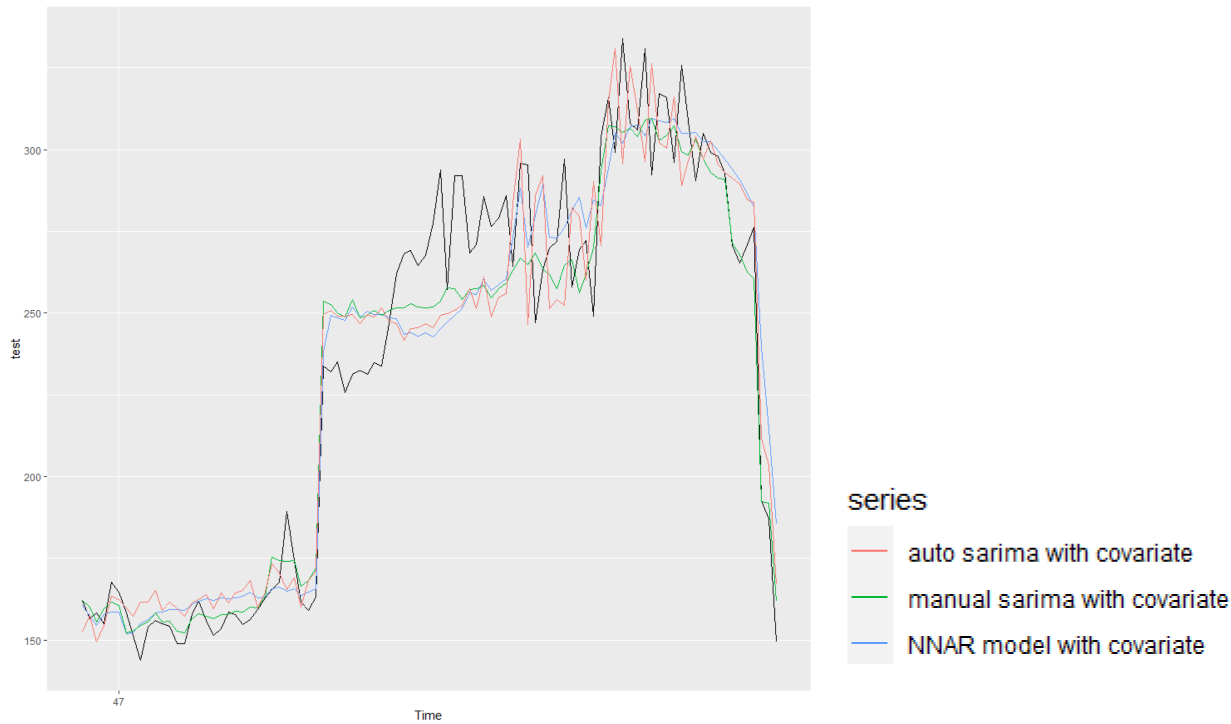
```
sarima_co2=Arima(train,xreg=train_temp,order=c(5,0,0),seasonal =c(0,1,1))
sarima_co2 #AIC=30298.51
checkresiduals(sarima_co2)
plot(pacf(sarima_co2$residuals))
prevsco2=forecast(sarima_co2,h=96,xreg=test_temp)
#finding the RMSE
print(sqrt(mean(prevsco2$mean-test)^2)) #3.239365 the auto.arima is better
```




```
# with a NNAR model with covariates
nfit_co=nnetar(train,xreg = train_temp)
prevn_co=forecast(nfit_co,h=96,xreg = test_temp)
print(sqrt(mean(prevn_co$mean-test)^2)) #0.2121313 best result yet
autoplot(train)+autolayer(prevn_co$mean,series="NNAR model with covariate")
```




```
#comparing the three models
autoplot(test)+autolayer(prevn_co$mean,series="NNAR model with covariate")+autolayer(prevsco2$mean,series="manual sarima with covariate")+autolayer(prevsco$mean,series="auto sarima with covariate")
```



#Using the best model (NNAR model with covariates) according to RMSE to forecast 2/17/2010 power consumption with temperature

```
#re-splitting the data
trainelec=head(elec,n=4507)
traintemp=head(temp,n=4507)
testtemp=tail(temp,n=96)

nfit_be=nnetar(trainelec,xreg = traintemp)
prevn_be=forecast(nfit_be,h=96,xreg = testtemp)
df2=as.data.frame(prevn_be$mean)
View(df2)
```