Regression with ARMA errors - daily

Zoe Yu

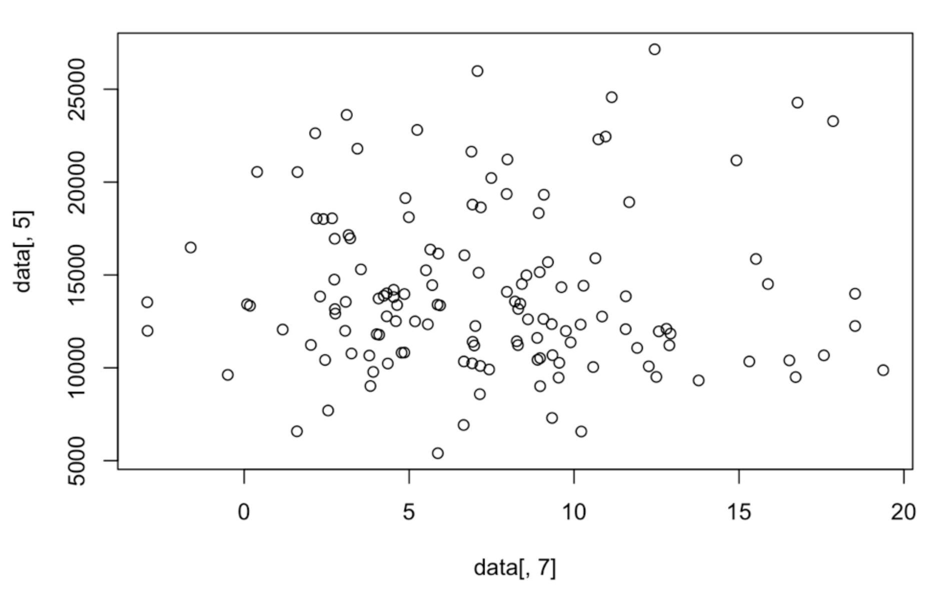
11/21/2020

load data & set up training/testing dataset

```r
data = read.csv('daily.csv')
temperature = ts(data[,7], frequency=7)
energy = ts(data[,5], frequency=7)
weekend = ts(data[,4],frequency=7)
temperature_train = ts(data[1:120,7], frequency=7)
temperature_test = ts(data[121:136,7], frequency=7)
energy_train = ts(data[1:120,5], frequency=7)
energy_test = ts(data[121:136,5], frequency=7)
weekend_train = ts(data[1:120,4], frequency=7)
weekend_test = ts(data[121:136,4], frequency=7)
```

plot for energy y against temperature x

```r
plot(data[,7],data[,5])
```

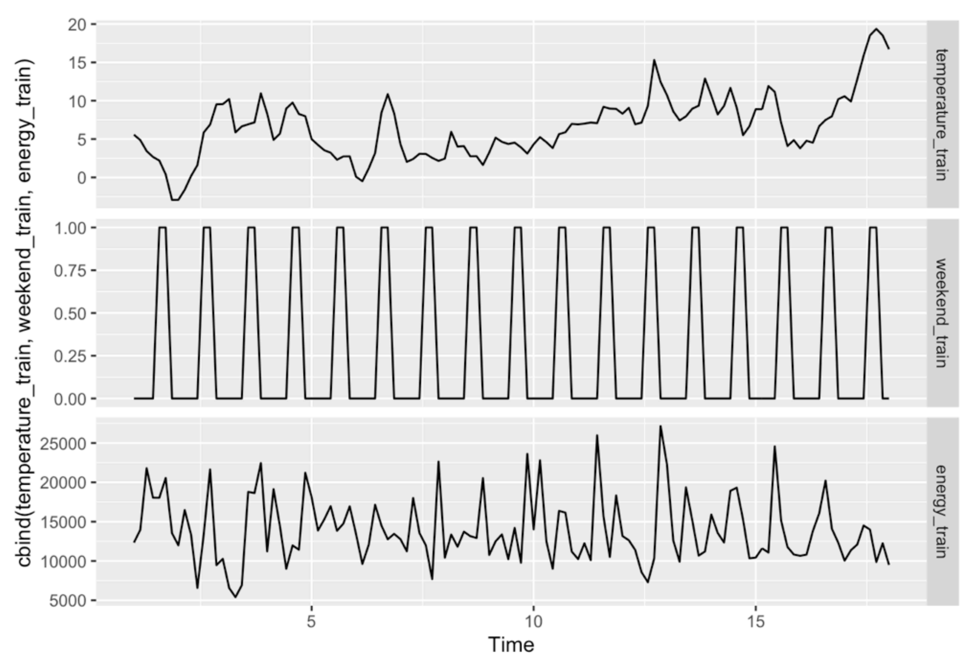



```r
cor(data[,7],data[,5])
```

```
## [1] -0.009796044
```

plot the ts training data

```r
autoplot(cbind(temperature_train,weekend_train,energy_train),facets=TRUE)
```




```r
kpss.test(weekend)
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  weekend
## KPSS Level = 0.018045, Truncation lag parameter = 4, p-value = 0.1
```

```r
kpss.test(temperature)
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  temperature
## KPSS Level = 1.5637, Truncation lag parameter = 4, p-value = 0.01
```

```r
kpss.test(energy)
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  energy
## KPSS Level = 0.056142, Truncation lag parameter = 4, p-value = 0.1
```

Energy and Weekend ts is stationary; Temperature is not stationary.

```r
xreg = cbind(temperature_train,weekend_train)
(fit = auto.arima(energy_train,xreg = xreg))
```

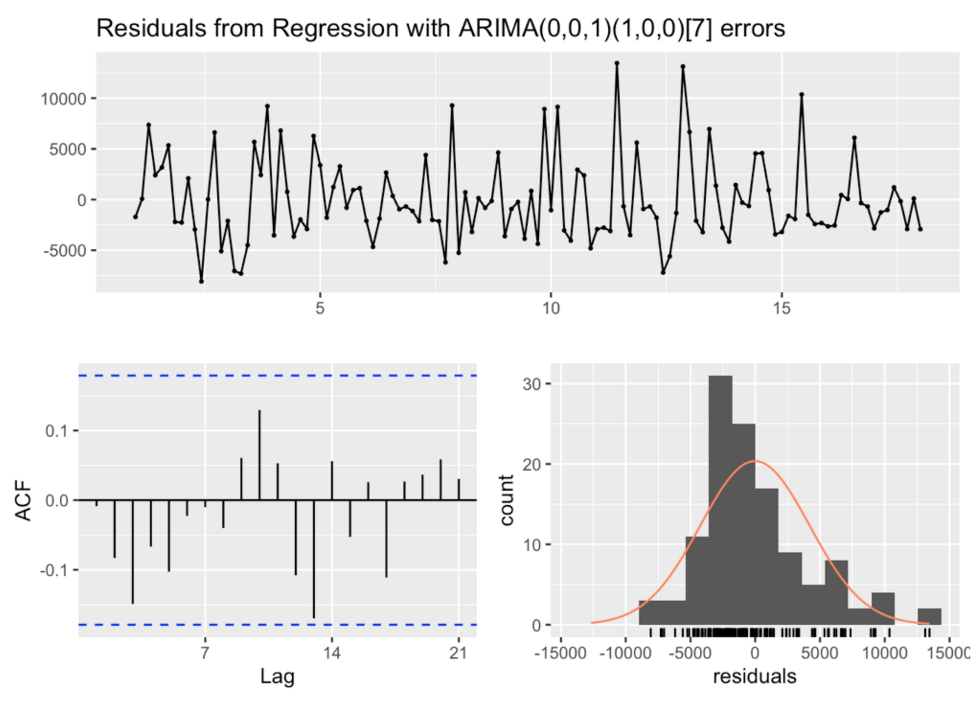
```
## Series: energy_train
## Regression with ARIMA(0,0,1)(1,0,0)[7] errors
##
## Coefficients:
##          ma1    sar1   intercept  temperature_train  weekend_train
##       0.1640  0.1761  14637.8437           -96.1714       -10.2921
## s.e.  0.0958  0.0917    946.0709           111.2716      1098.6568
##
## sigma^2 estimated as 18416073:  log likelihood=-1171.57
## AIC=2355.13   AICc=2355.88   BIC=2371.86
```

Best fit model: Regression with ARIMA(0,0,1)(1,0,0)[7] errors

```r
summary(fit)
```

```
## Series: energy_train
## Regression with ARIMA(0,0,1)(1,0,0)[7] errors
##
## Coefficients:
##          ma1    sar1   intercept  temperature_train  weekend_train
##       0.1640  0.1761  14637.8437           -96.1714       -10.2921
## s.e.  0.0958  0.0917    946.0709           111.2716      1098.6568
##
## sigma^2 estimated as 18416073:  log likelihood=-1171.57
## AIC=2355.13   AICc=2355.88   BIC=2371.86
##
## Training set error measures:
##                     ME    RMSE      MAE       MPE     MAPE      MASE
## Training set -22.31588 4201.04 3216.003 -9.086014 24.60502 0.7700799
##                     ACF1
## Training set -0.00870926
```

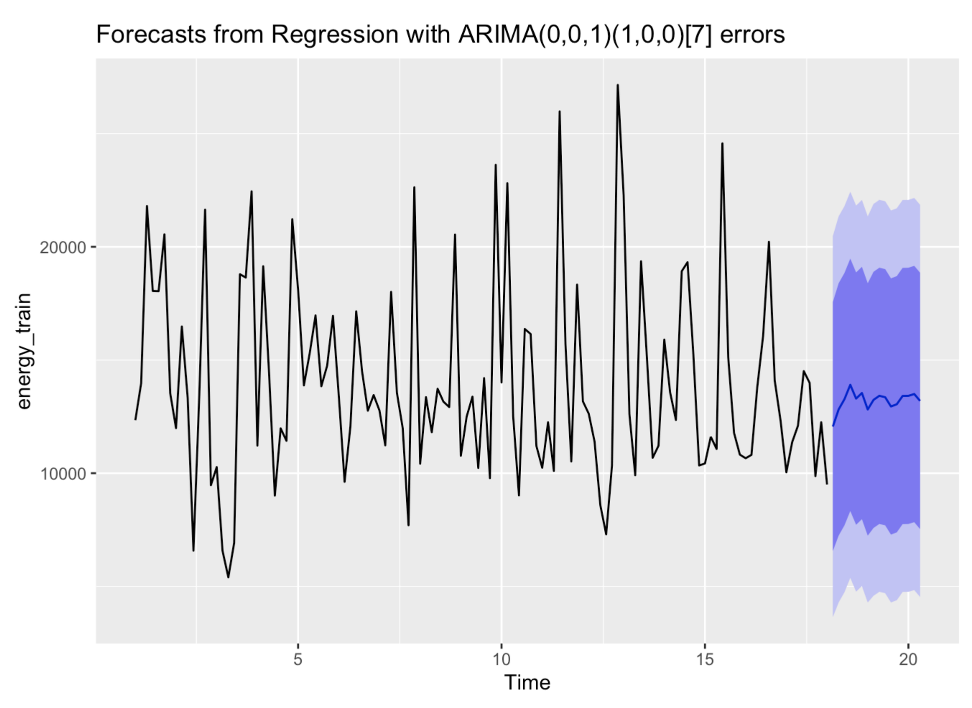
```r
checkresiduals(fit)
```



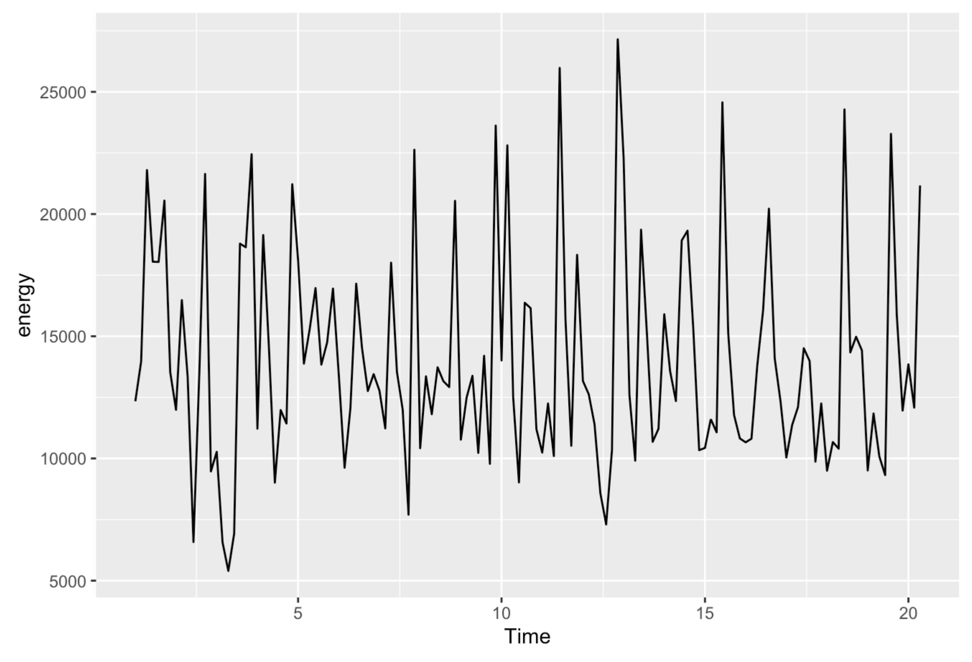

```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,0,1)(1,0,0)[7] errors
## Q* = 14.901, df = 9, p-value = 0.0937
##
## Model df: 5.   Total lags used: 14
```

All lags are inside the significant intervals

```r
xreg = cbind(temperature_test,weekend_test)
fcast = forecast(fit,xreg = xreg,h=16)
autoplot(fcast)
```

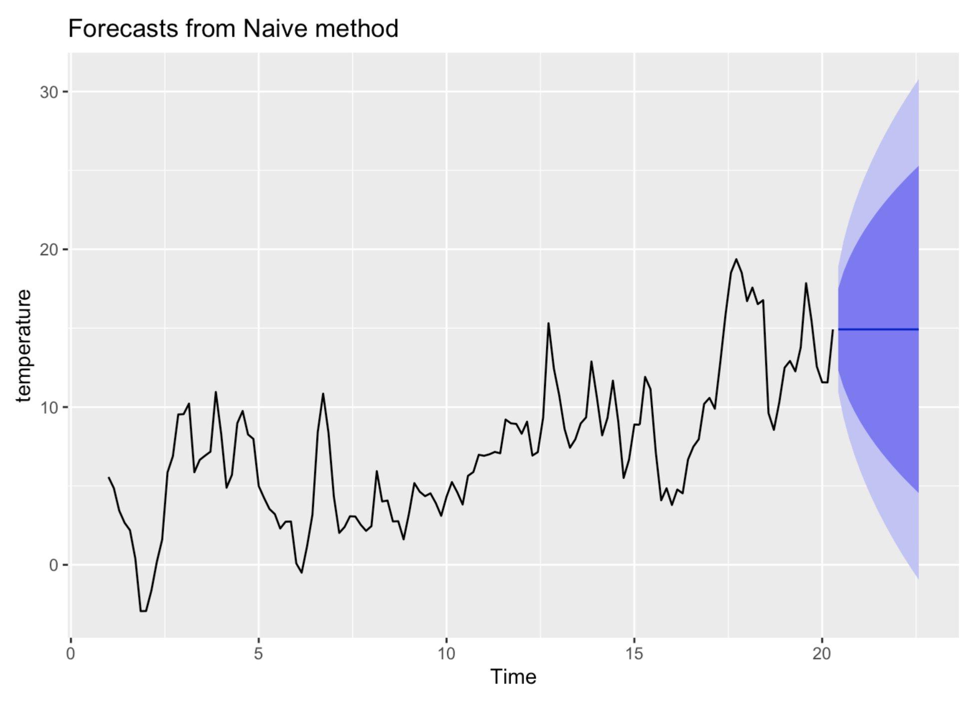



```r
autoplot(energy)
```



forecasting

```r
temperature_naive <- naive(temperature, h=16)
autoplot(temperature_naive)
```




```r
weekend_naive = c(0,1,1,0,0,0,0,0,1,1,0,0,0,0,0,1)
temperature_naive = rep(14.92361,16)
xreg = cbind(temperature_naive,weekend_naive)
xreg
```

```
##       temperature_naive weekend_naive
##  [1,]          14.92361             0
##  [2,]          14.92361             1
##  [3,]          14.92361             1
##  [4,]          14.92361             0
##  [5,]          14.92361             0
##  [6,]          14.92361             0
##  [7,]          14.92361             0
##  [8,]          14.92361             0
##  [9,]          14.92361             1
## [10,]          14.92361             1
## [11,]          14.92361             0
## [12,]          14.92361             0
## [13,]          14.92361             0
## [14,]          14.92361             0
## [15,]          14.92361             0
## [16,]          14.92361             1
```

```r
fcast = forecast(fit,xreg = xreg,h=16)
autoplot(fcast)
```