Temperature Selection

Here, we will compare the different metrics available for temperature and select the most efficient one (or most efficient combinaison) for predicting gas consumption.

Load data

We have separated the original excel file into separate .csv files to ease the import. Since temperature and weather data from google trends are monthly values, we decided to set them to the 15th of each month to match with the gas data. We omit here the last 12 lines of the gas.csv which correspond to the values to predict.

```r
library(readr)
gas <- na.omit(read_csv("data/gas.csv", col_types = cols(date = col_date(format = "%Y-%m-%d"))))
temp <- read_csv("data/temp.csv", col_types = cols(date = col_date(format = "%Y-%m-%d")))
google <- read_csv("data/google.csv", col_types = cols(date = col_date(format = "%Y-%m-%d")))
```

The temperature data is taken from the National Center for Environmental Information.

```r
summary(temp)
```

```
##       date             av_temp_val     av_temp_ano      min_temp_val
##  Min.   :1895-01-15   Min.   :21.90   Min.   :-8.590   Min.   :12.52
##  1st Qu.:1925-07-22   1st Qu.:37.96   1st Qu.:-0.990   1st Qu.:26.95
##  Median :1956-01-30   Median :52.73   Median : 0.170   Median :40.11
##  Mean   :1956-01-29   Mean   :52.16   Mean   : 0.171   Mean   :40.19
##  3rd Qu.:1986-08-07   3rd Qu.:66.72   3rd Qu.: 1.450   3rd Qu.:54.01
##  Max.   :2017-02-15   Max.   :76.80   Max.   : 8.910   Max.   :63.55
##   min_temp_ano      max_temp_val    max_temp_ano        prec_val
##  Min.   :-8.6200   Min.   :31.26   Min.   :-9.2300   Min.   :0.540
##  1st Qu.:-0.9400   1st Qu.:48.55   1st Qu.:-1.2400   1st Qu.:2.140
##  Median : 0.1900   Median :65.34   Median : 0.1600   Median :2.510
##  Mean   : 0.1854   Mean   :64.13   Mean   : 0.1542   Mean   :2.501
##  3rd Qu.: 1.3800   3rd Qu.:79.78   3rd Qu.: 1.6800   3rd Qu.:2.880
##  Max.   : 8.4700   Max.   :90.84   Max.   :10.0900   Max.   :4.440
##     prec_ano          cool_days_val   cool_days_ano     heat_days_val
##  Min.   :-1.620000   Min.   :  1.0   Min.   :-72.000   Min.   :   3.0
##  1st Qu.:-0.310000   1st Qu.: 10.0   1st Qu.: -5.000   1st Qu.:  56.0
##  Median :-0.010000   Median : 41.0   Median : -1.000   Median : 312.0
##  Mean   : 0.006603   Mean   :101.0   Mean   :  1.931   Mean   : 385.9
##  3rd Qu.: 0.310000   3rd Qu.:184.8   3rd Qu.:  7.000   3rd Qu.: 692.0
##  Max.   : 2.130000   Max.   :405.0   Max.   : 90.000   Max.   :1184.0
##  heat_days_ano
##  Min.   :-258.000
##  1st Qu.: -26.000
##  Median :  -2.000
##  Mean   :  -3.763
##  3rd Qu.:  15.000
##  Max.   : 259.000
```

```r
summary(google)
```

```
##       date               heatwave     extreme_weather   snow_storm
##  Min.   :2004-01-15   Min.   :2.00   Min.   :1.000   Min.   :  1.000
```

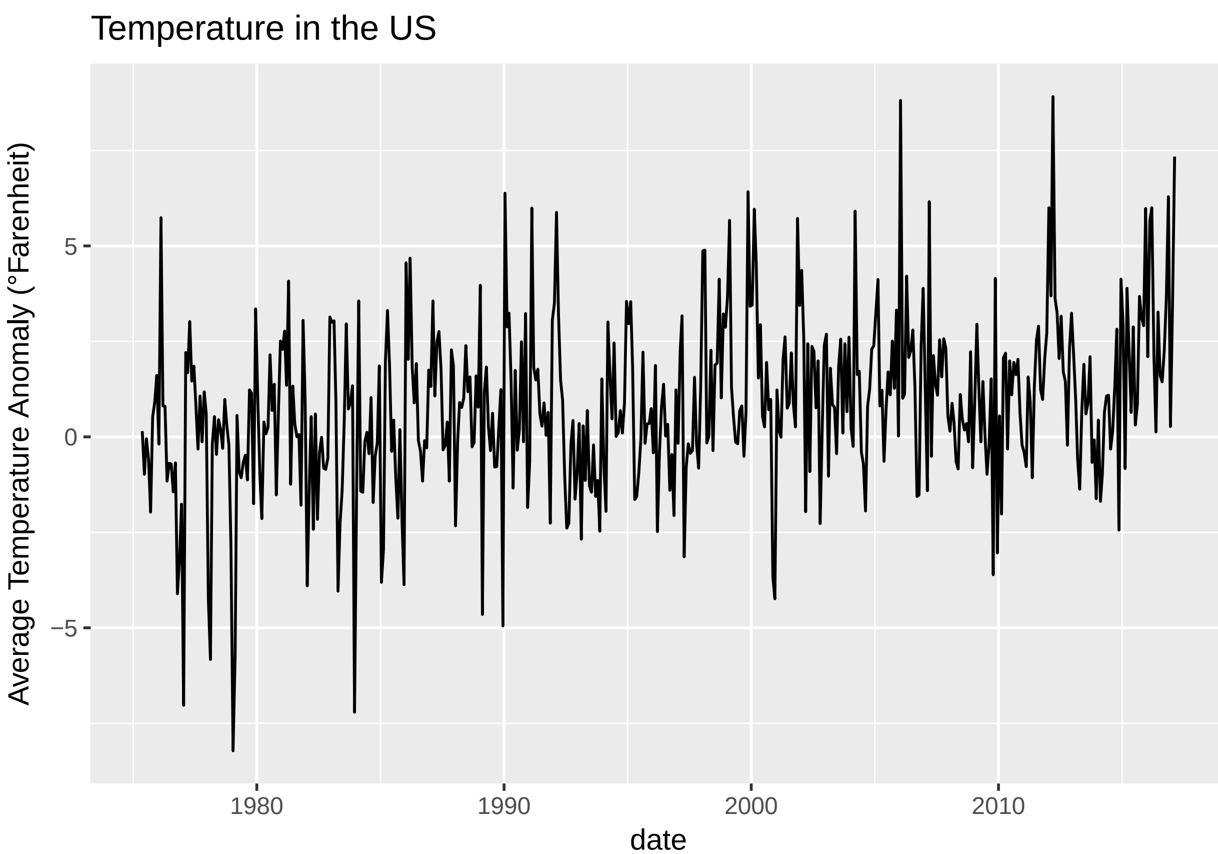
```
##  1st Qu.:2007-04-30   1st Qu.:3.00   1st Qu.:2.000   1st Qu.:  2.000
##  Median :2010-08-15   Median :3.00   Median :2.000   Median :  3.000
##  Mean   :2010-08-15   Mean   :3.27   Mean   :2.252   Mean   :  8.767
##  3rd Qu.:2013-11-30   3rd Qu.:4.00   3rd Qu.:3.000   3rd Qu.:  9.000
##  Max.   :2017-03-15   Max.   :8.00   Max.   :8.000   Max.   :100.000
```

We first notice that these datasets span over different periods of time: while gas consumption, our target value goes from January 1973 to July 2016, temperature data start in January 1895 and end in February 2017. Finally, data from Google Trends go from january 2004 and up to march 2017.

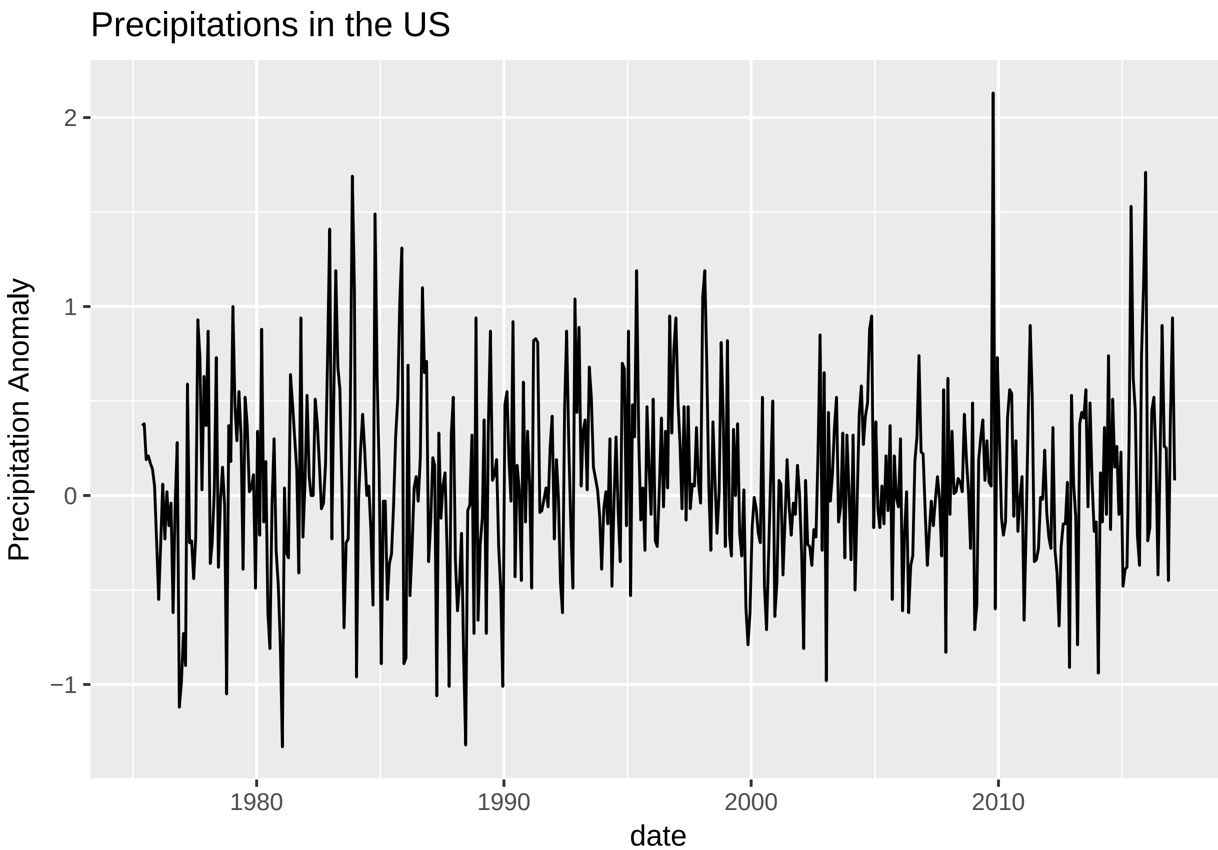
Plots

Temperature

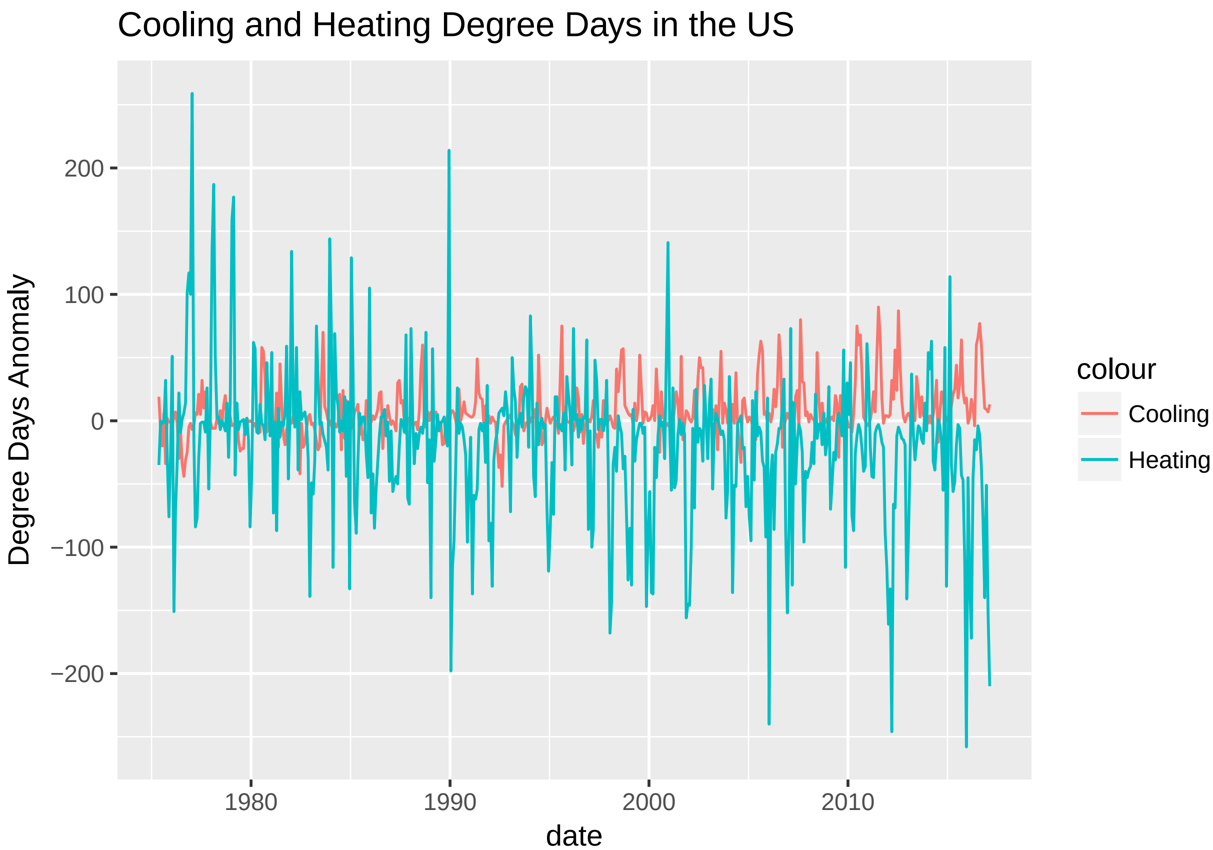
```
library(ggplot2)
temp_trunc = temp[temp$date >= 1973-01-15,]
ggplot(temp_trunc)+geom_line(aes(x=date, y=av_temp_ano))+
    labs(title = "Temperature in the US",  y = "Average Temperature Anomaly (°Farenheit)")
```




```
ggplot(temp_trunc)+geom_line(aes(x=date, y=prec_ano))+
    labs(title = "Precipitations in the US",  y = "Precipitation Anomaly")
```

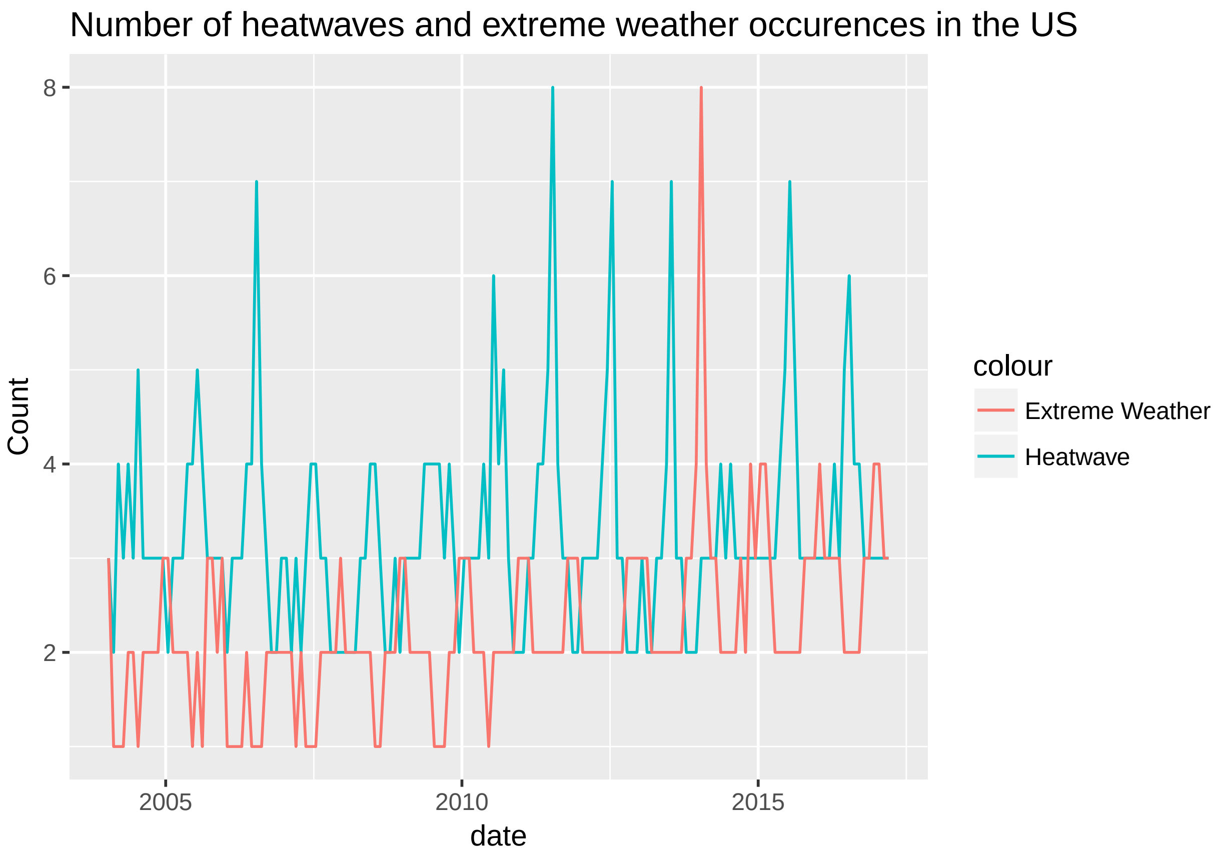



```r
ggplot(temp_trunc)+geom_line(aes(x=date, y=cool_days_ano, colour = "Cooling"))+
    geom_line(aes(x=date, y=heat_days_ano, colour ="Heating"))+
    labs(title = "Cooling and Heating Degree Days in the US",  y = "Degree Days Anomaly")
```

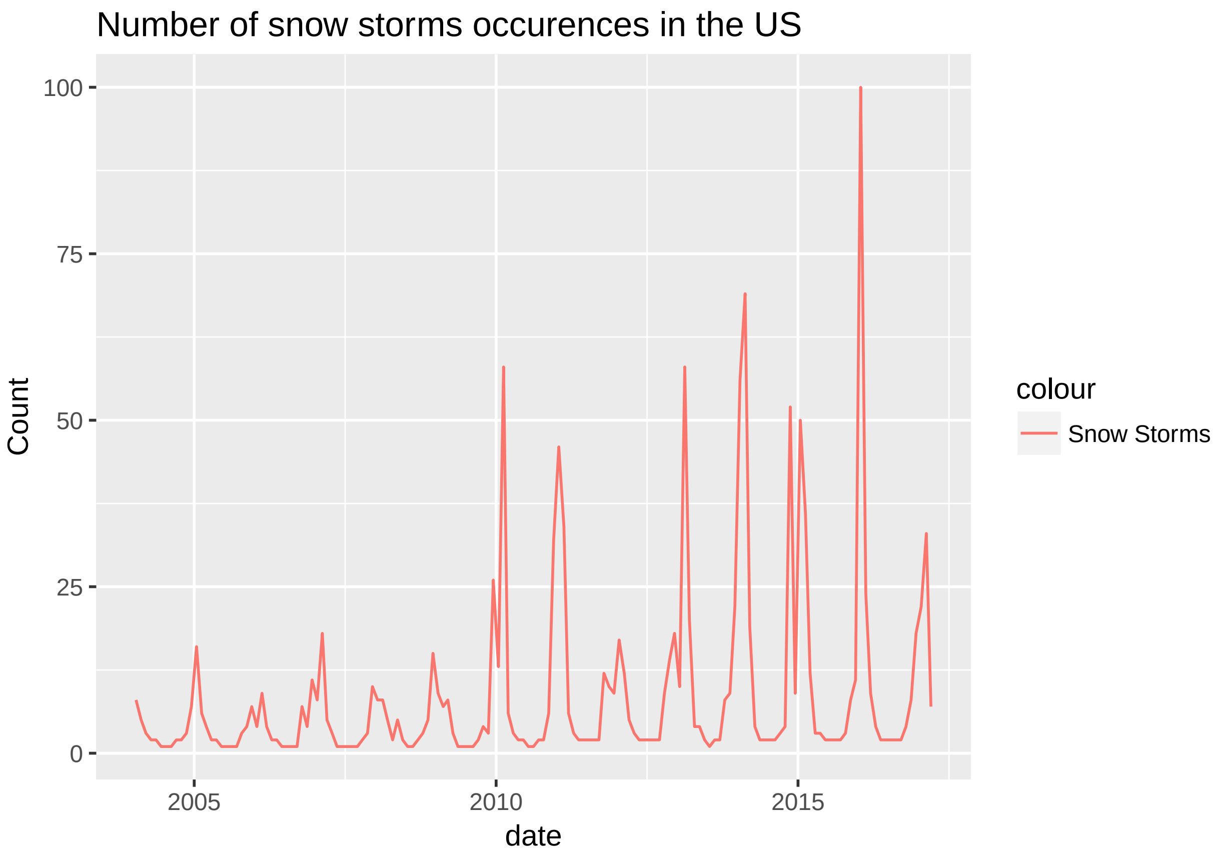


Google trends

```
ggplot(google)+geom_line(aes(x=date, y=heatwave, colour="Heatwave"))+
    geom_line(aes(x=date, y=extreme_weather, colour = "Extreme Weather"))+
    labs(title = "Number of heatwaves and extreme weather occurences in the US",  y = "Count")
```




```
ggplot(google)+geom_line(aes(x=date, y=snow_storm, colour="Snow Storms"))+
    labs(title = "Number of snow storms occurences in the US",  y = "Count")
```



Covariance analysis

Let us merge the dataframes

```r
gas_temp = merge(gas, temp, by = "date", all.x = TRUE)
gas_goo = merge(gas, google, by = "date", all = FALSE)
all = merge(gas_temp, google, by= "date",  all = FALSE)
```

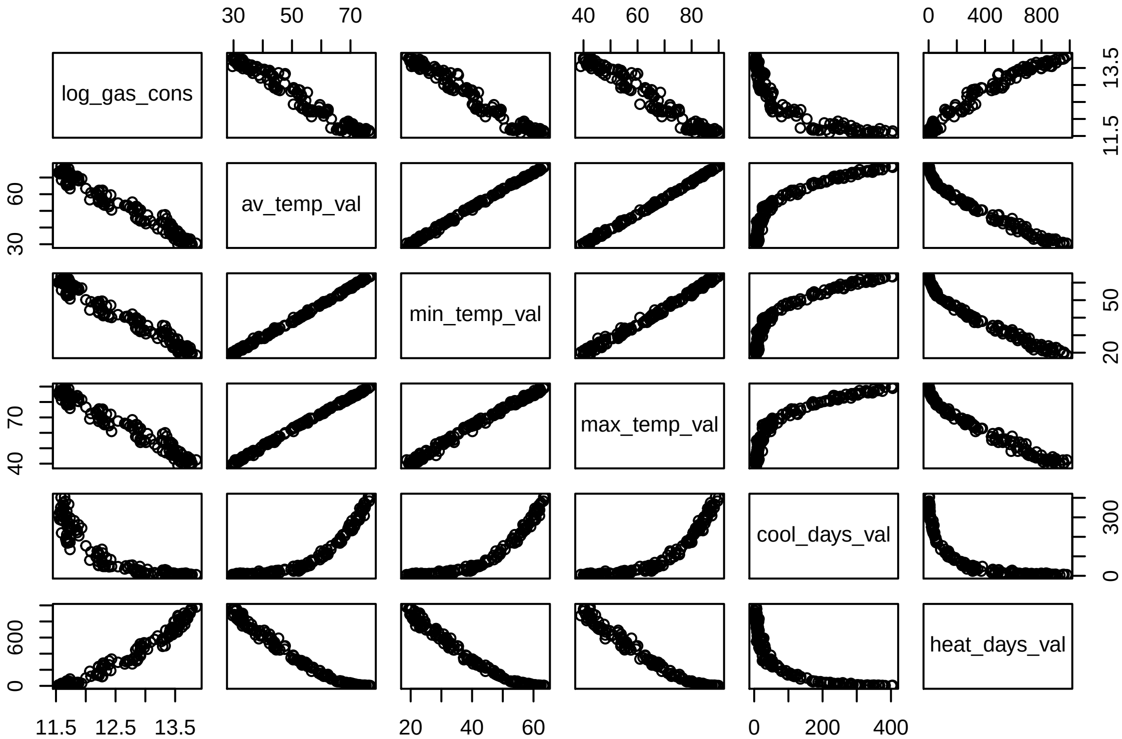
The covariances show us that: - the av_temp_val, the min_temp_val, the max_temp_val, cool_days_val are highly negatively correlated with the log gas consumption - the heat_days_val is highly positively correlated with the log gas consumption This is logical since it is when the temperature drops and that heating is necessary that the gas consumption is expected to rise. We also note that min_temp_val (resp. heat_days_val) is the most negatively (resp. positively) correlated variable with log gas consumption.

```r
cor(all[,c(3)],all[,-c(1,2,3)])
```

```
##      av_temp_val av_temp_ano min_temp_val min_temp_ano max_temp_val
## [1,]  -0.9796413  0.06763444   -0.9802413  0.001284384   -0.9772051
##      max_temp_ano   prec_val   prec_ano cool_days_val cool_days_ano
## [1,]     0.117867 -0.4530804 -0.1699716    -0.8721013    -0.5303987
##      heat_days_val heat_days_ano   heatwave extreme_weather snow_storm
## [1,]     0.9775618    -0.2019406 -0.5541514       0.4600452  0.5538882
```

As seen in the following pairs plot, we kind of what to use the log of the cool_days_val. Let's try !

```r
corr <- all[,c(3,4,6,8,12,14)]
pairs(corr)
```



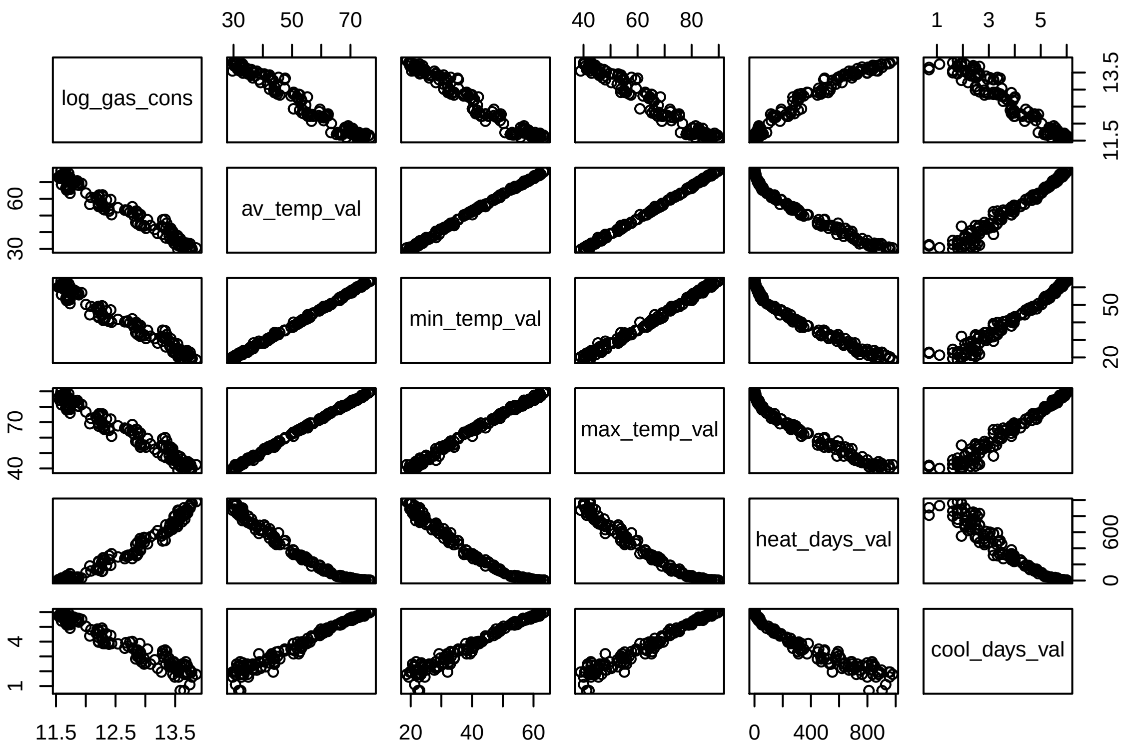
An interesting result here is that there appears to be four "steps" in the log_gas_cons and av_temp_val and min_temp_val scatter plots. There are perhaps four "steps" in the gas consumption behaviour according to the value of the temperatue ? This could be interesting to study in a further study.

```
cor(all[,c(3)],log(all$cool_days_val))
```

```
## [1] -0.9668738
```

Which is better ! The best variables to use are thus : - the av_temp_val, - the min_temp_val, - the max_temp_val, - the log(cool_days_val), - the heat_days_val

```
top_corr <- cbind(all[,c(3,4,6,8,14)], log(all$cool_days_val))
colnames(top_corr) <- c(colnames(all[,c(3,4,6,8,14)]), "cool_days_val")
pairs(top_corr)
```



Granger causality tests

Whilst the correlations give us an indication of the variables that best explain the log gas consumption, we have to make sure that there is a causality link between the temperature and the log gas consumption.

Here, we will test the most important variables in terms of correlation positively and negatively to avoid redundance : heat_days_val and av_temp_val As seen with the unit root tests bellow, we need to do a first seasonal difference.

```
library(forecast)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: timeDate

## This is forecast 7.3

##
## Attaching package: 'forecast'

## The following object is masked _by_ '.GlobalEnv':
##
##     gas
```

```
# test for unit root and number of differences required
ndiffs(all$min_temp_val, alpha=0.05, test=c("kpss"))
```

```
## [1] 0
```

```r
ndiffs(all$heat_days_val, alpha=0.05, test=c("kpss"))
```

```
## [1] 0
```

```r
ndiffs(all$log_gas_cons, alpha=0.05, test=c("kpss"))
```

```
## [1] 0
```

```r
# test for unit roots in seasonality
nsdiffs(all$min_temp_val, m=12, test=c("ocsb"))
```

```
## [1] 1
```

```r
nsdiffs(all$heat_days_val, m=12, test=c("ocsb"))
```

```
## [1] 1
```

```r
nsdiffs(all$log_gas_cons, m=12, test=c("ocsb"))
```

```
## [1] 1
```

```r
# test for unit roots in the first seasonal difference
nsdiffs(diff(all$min_temp_val, lag = 12), m=12, test=c("ocsb"))
```

```
## [1] 0
```

```r
nsdiffs(diff(all$heat_days_val, lag = 12), m=12, test=c("ocsb"))
```

```
## [1] 0
```

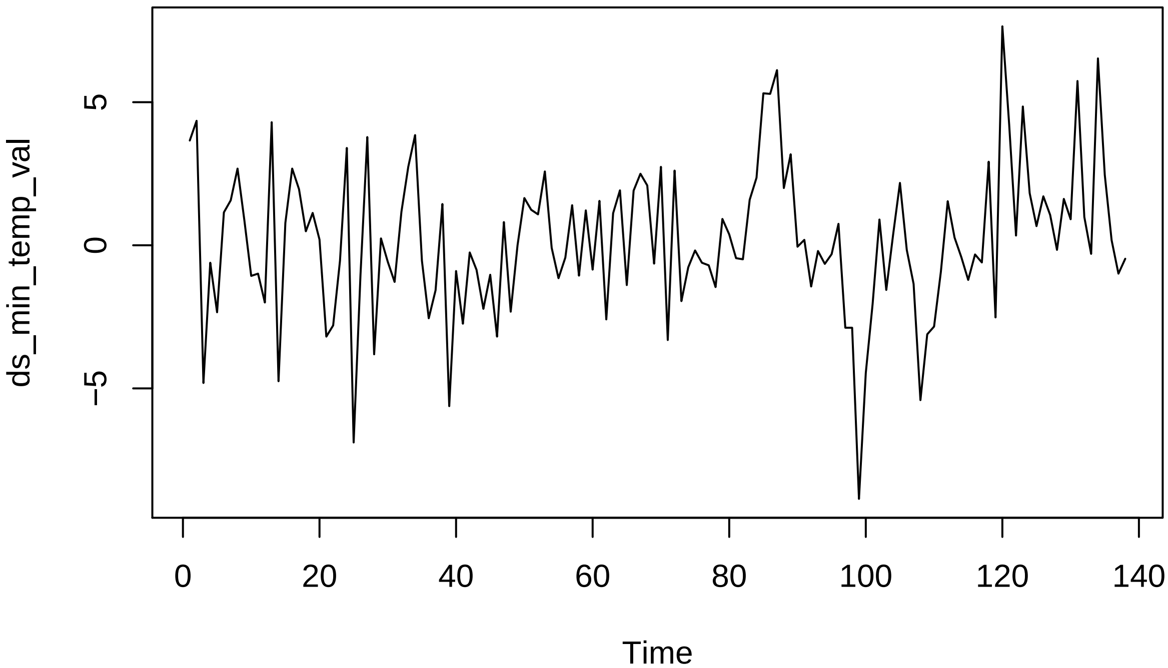
```r
nsdiffs(diff(all$log_gas_cons, lag = 12), m=12, test=c("ocsb"))
```

```
## [1] 0
```

As we can see, the first seasonal difference removes the unit root : it is thus the time series we will consider.

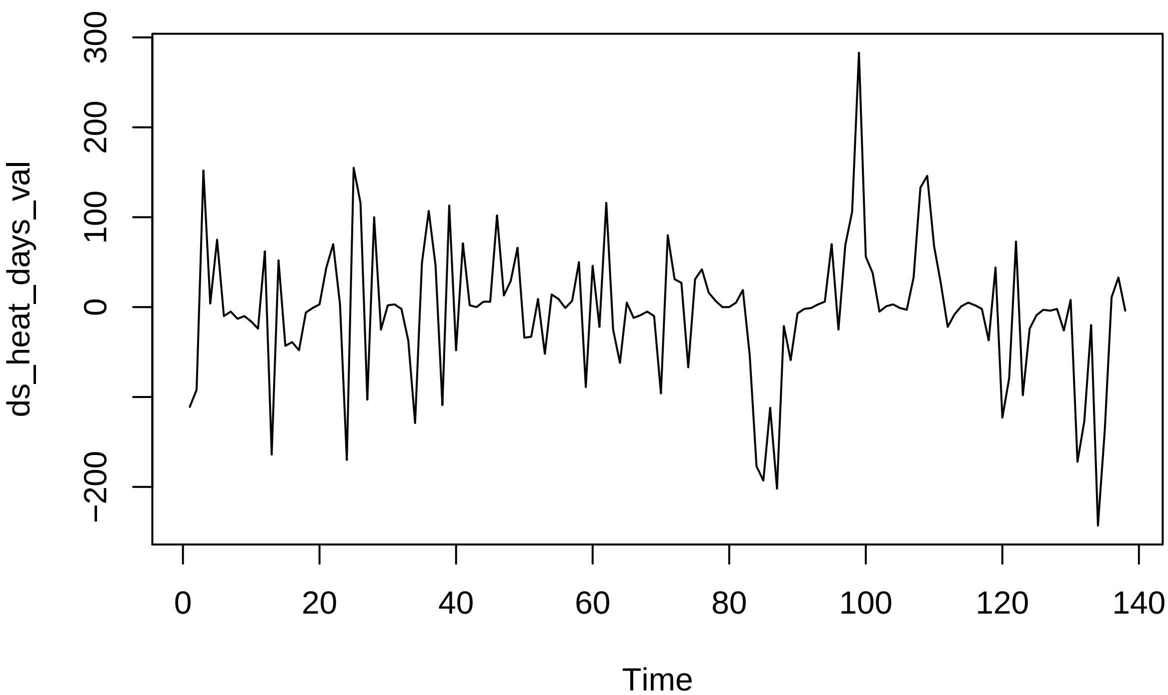
```r
# differenced time series
ds_min_temp_val <- diff(all$min_temp_val, lag = 12)
ds_heat_days_val <- diff(all$heat_days_val, lag = 12)
ds_log_gas_cons <- diff(all$log_gas_cons, lag = 12)

plot.ts(ds_min_temp_val, main = "Diff seasonal min_temp_val")
```

Diff seasonal min_temp_val

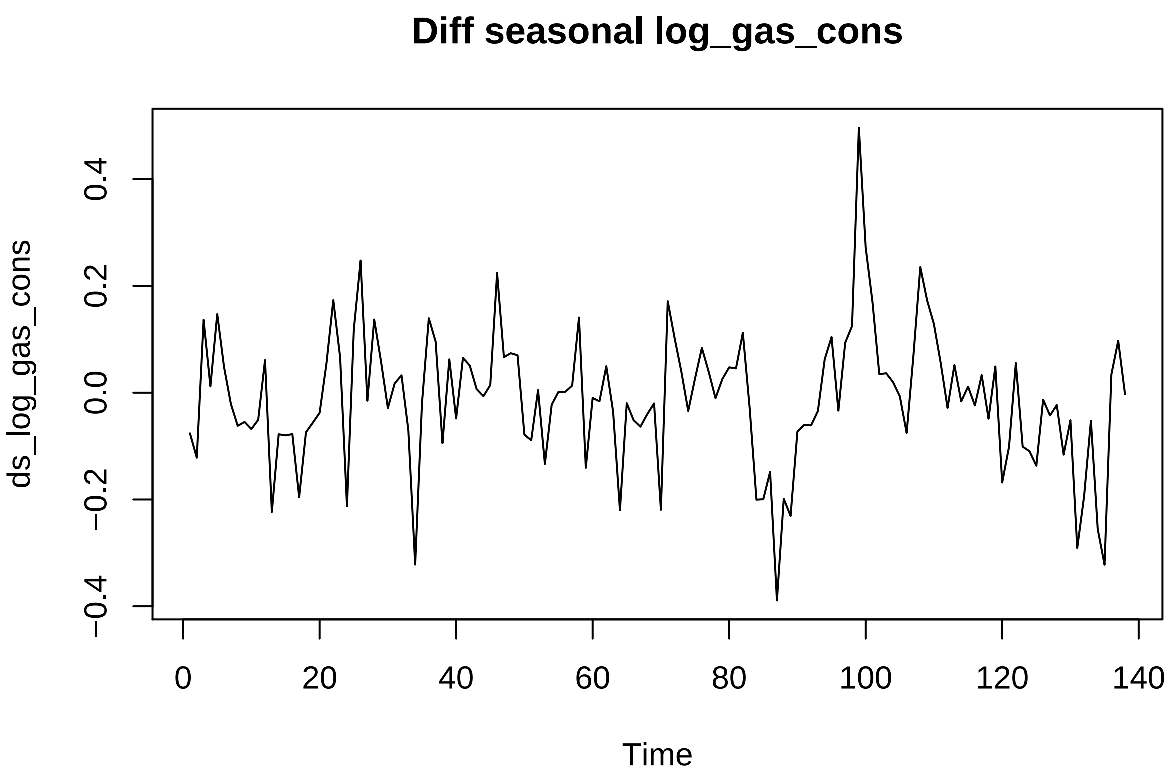



```r
plot.ts(ds_heat_days_val, main = "Diff seasonal heat_days_val")
```

Diff seasonal heat_days_val




```r
plot.ts(ds_log_gas_cons, main = "Diff seasonal log_gas_cons")
```



The following Granger causality tests show that : - the ds_min_temp_val Granger-cause ds_log_gas_cons at order = 1, 2, 3, 4, 7, 8, 9, 10, 11 : thus all the time exception in summer ! - the ds_heat_days_val Granger-cause ds_log_gas_cons at order = 7 and 11

```r
library(lmtest)
# performing the granger causality test
grangertest(ds_log_gas_cons ~ ds_min_temp_val, order = 1)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:1) + Lags(ds_min_temp_val, 1:1)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:1)
##   Res.Df Df     F  Pr(>F)
## 1    134
## 2    135 -1 6.362 0.01283 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_min_temp_val, order = 2)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:2) + Lags(ds_min_temp_val, 1:2)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:2)
##   Res.Df Df      F   Pr(>F)
## 1    131
## 2    133 -2 4.8138 0.009608 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_min_temp_val, order = 3)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:3) + Lags(ds_min_temp_val, 1:3)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:3)
##   Res.Df Df      F  Pr(>F)
## 1    128
## 2    131 -3 3.8295 0.01149 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_min_temp_val, order = 4)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:4) + Lags(ds_min_temp_val, 1:4)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:4)
##   Res.Df Df      F  Pr(>F)
## 1    125
## 2    129 -4 2.8495 0.02663 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_min_temp_val, order = 7)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:7) + Lags(ds_min_temp_val, 1:7)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:7)
##   Res.Df Df      F    Pr(>F)
## 1    116
## 2    123 -7 3.8898 0.0007672 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_min_temp_val, order = 8)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:8) + Lags(ds_min_temp_val, 1:8)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:8)
##   Res.Df Df      F    Pr(>F)
## 1    113
## 2    121 -8 3.7551 0.0006207 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_min_temp_val, order = 9)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:9) + Lags(ds_min_temp_val, 1:9)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:9)
##   Res.Df Df      F   Pr(>F)
## 1    110
## 2    119 -9 3.1763 0.001896 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_min_temp_val, order = 10)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:10) + Lags(ds_min_temp_val, 1:10)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:10)
##   Res.Df  Df      F    Pr(>F)
## 1    107
## 2    117 -10 3.3606 0.0007699 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_min_temp_val, order = 11)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:11) + Lags(ds_min_temp_val, 1:11)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:11)
##   Res.Df  Df      F    Pr(>F)
## 1    104
## 2    115 -11 4.0897 5.429e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_heat_days_val, order = 7)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:7) + Lags(ds_heat_days_val, 1:7)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:7)
##   Res.Df Df     F  Pr(>F)
## 1    116
## 2    123 -7 1.771 0.09963 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_log_gas_cons ~ ds_heat_days_val, order = 11)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:11) + Lags(ds_heat_days_val, 1:11)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:11)
##   Res.Df  Df     F   Pr(>F)
## 1    104
## 2    115 -11 2.851 0.002657 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Strangely, we also note that the test shows that : - the ds_log_gas_cons Granger-cause ds_cool_days_val at order = 2, 3, 4, 5, 7, 8, 9, 10 and 11 - the ds_log_gas_cons Granger-cause ds_heat_days_val at order = 11

```r
grangertest(ds_min_temp_val ~ ds_log_gas_cons, order = 2)
```

```
## Granger causality test
##
```

```
## Model 1: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:2) + Lags(ds_log_gas_cons, 1:2)
## Model 2: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:2)
##   Res.Df Df      F  Pr(>F)
## 1    131
## 2    133 -2 4.5546 0.01224 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_min_temp_val ~ ds_log_gas_cons, order = 3)
```

```
## Granger causality test
##
## Model 1: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:3) + Lags(ds_log_gas_cons, 1:3)
## Model 2: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:3)
##   Res.Df Df     F  Pr(>F)
## 1    128
## 2    131 -3 3.209 0.02535 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_min_temp_val ~ ds_log_gas_cons, order = 4)
```

```
## Granger causality test
##
## Model 1: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:4) + Lags(ds_log_gas_cons, 1:4)
## Model 2: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:4)
##   Res.Df Df      F  Pr(>F)
## 1    125
## 2    129 -4 2.3277 0.05983 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_min_temp_val ~ ds_log_gas_cons, order = 5)
```

```
## Granger causality test
##
## Model 1: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:5) + Lags(ds_log_gas_cons, 1:5)
## Model 2: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:5)
##   Res.Df Df      F  Pr(>F)
## 1    122
## 2    127 -5 2.2456 0.05399 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_min_temp_val ~ ds_log_gas_cons, order = 7)
```

```
## Granger causality test
##
## Model 1: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:7) + Lags(ds_log_gas_cons, 1:7)
## Model 2: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:7)
##   Res.Df Df      F  Pr(>F)
## 1    116
## 2    123 -7 2.4953 0.02005 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_min_temp_val ~ ds_log_gas_cons, order = 8)
```

```
## Granger causality test
##
## Model 1: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:8) + Lags(ds_log_gas_cons, 1:8)
## Model 2: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:8)
##   Res.Df Df      F  Pr(>F)
## 1    113
## 2    121 -8 2.4078 0.01943 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_min_temp_val ~ ds_log_gas_cons, order = 9)
```

```
## Granger causality test
##
## Model 1: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:9) + Lags(ds_log_gas_cons, 1:9)
## Model 2: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:9)
##   Res.Df Df      F  Pr(>F)
## 1    110
## 2    119 -9 2.3885 0.01646 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_min_temp_val ~ ds_log_gas_cons, order = 10)
```

```
## Granger causality test
##
## Model 1: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:10) + Lags(ds_log_gas_cons, 1:10)
## Model 2: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:10)
##   Res.Df  Df      F   Pr(>F)
## 1    107
## 2    117 -10 2.5232 0.009108 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_min_temp_val ~ ds_log_gas_cons, order = 11)
```

```
## Granger causality test
##
## Model 1: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:11) + Lags(ds_log_gas_cons, 1:11)
## Model 2: ds_min_temp_val ~ Lags(ds_min_temp_val, 1:11)
##   Res.Df  Df      F   Pr(>F)
## 1    104
## 2    115 -11 3.3601 0.000537 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(ds_heat_days_val ~ ds_log_gas_cons, order = 11)
```

```
## Granger causality test
##
## Model 1: ds_heat_days_val ~ Lags(ds_heat_days_val, 1:11) + Lags(ds_log_gas_cons, 1:11)
## Model 2: ds_heat_days_val ~ Lags(ds_heat_days_val, 1:11)
##   Res.Df  Df      F   Pr(>F)
## 1    104
## 2    115 -11 2.6397 0.005131 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We will thus only consider ds_min_temp_val and ds_heat_days_val. However, it could be interesting to include in our models dummy variables that model extreme weather cases… We could also try to compute a form of "felt air temperature" with the precipitations, but we do not have time to do so in this study.

Creating dummy variables for extreme cases

The extreme weather cases are represented by the following variables: - min_temp_val - min_temp_ano - max_temp_val - max_temp_ano - heatwave - extreme_weather - snow_storm Here, we will only consider : heatwave, extreme_weather, snow_storm, because the other variables should either be used to compute the trend or are already used (such as min_temp_val).

To create a unique "extreme weather" variable, we will do two dummy variables : extreme_heat and extreme_cold.

```
extreme <- all[,c(16,17,18)]

# extreme_heat will be 1 when heatwave > mean(heatwave)
extreme_heat <- as.numeric(extreme$heatwave > mean(extreme$heatwave))
# extreme_cold will be 1 when snow_storm > mean(snow_storm) and extreme_weather > mean(extreme_weather)
extreme_cold <- as.numeric(extreme$snow_storm > mean(extreme$snow_storm) & extreme$extreme_weather > mea
```

How about doing a PCA and Granger causality tests ?

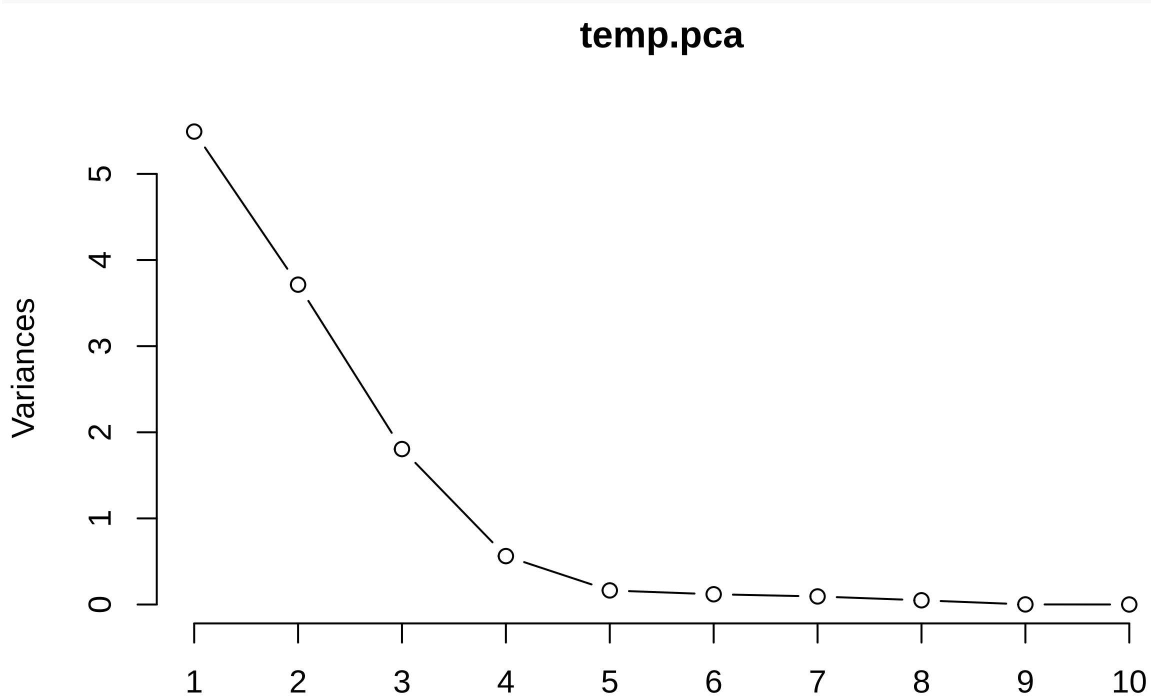
We will now try doing a PCA on all our temperature datasets, except the variables used for the dummies.

```
temp <- all[,-c(1,2,3, 16, 17, 18)]
# Apply PCA with scaling = TRUE is highly
temp.pca <- prcomp(temp,
                   center = TRUE,
                   scale. = TRUE)
# print method
print(temp.pca)
```

```
## Standard deviations:
##  [1] 2.3433245269 1.9272025708 1.3435041828 0.7499129144 0.4044813589
##  [6] 0.3451714559 0.3062483488 0.2218956604 0.0345113092 0.0193167433
## [11] 0.0030193055 0.0001485734
##
## Rotation:
##                     PC1          PC2         PC3         PC4         PC5
## av_temp_val    0.42005782 -0.014488232 -0.07753831  0.18159244  0.02596611
## av_temp_ano   -0.01925074 -0.515571522 -0.01887798  0.02647139 -0.11927308
## min_temp_val   0.42155876 -0.012929312 -0.06119864  0.16482401 -0.05343869
## min_temp_ano   0.01436657 -0.494385138  0.10656575 -0.03090579 -0.48089002
## max_temp_val   0.41791495 -0.015810412 -0.09194861  0.19587103  0.09620052
## max_temp_ano  -0.04571603 -0.496394438 -0.12139680  0.06993346  0.18822721
## prec_val       0.23660392  0.043858399  0.59548224 -0.08270736  0.01099520
## prec_ano       0.10055724  0.049375918  0.70568658 -0.13475306 -0.03736933
## cool_days_val  0.39407259  0.001294117 -0.14460533 -0.14445881 -0.60261993
## cool_days_ano  0.26745456 -0.199521233 -0.14260115 -0.84269110  0.35857938
## heat_days_val -0.41036825  0.018610963  0.01554362 -0.27591283 -0.36797829
## heat_days_ano  0.08484915  0.445215175 -0.24750378 -0.25064268 -0.27865647
```

```
##                       PC6         PC7         PC8         PC9        PC10
## av_temp_val    0.073948451 -0.01059985 -0.01477616  0.33042689  0.06259798
## av_temp_ano    0.210760210  0.13759422 -0.07277600 -0.05259287  0.01022916
## min_temp_val   0.054996306 -0.06076733  0.03681540  0.15236747  0.78015320
## min_temp_ano   0.335198778 -0.38862807  0.30727949 -0.01535986 -0.10598499
## max_temp_val   0.090398430  0.03411715 -0.06124199  0.48571546 -0.57701323
## max_temp_ano   0.093420246  0.56409435 -0.38497246 -0.08046024  0.06837001
## prec_val       0.008827238  0.54481839  0.52707661 -0.06938841 -0.03170154
## prec_ano       0.120239986 -0.21697600 -0.63666753  0.05004616  0.02269790
## cool_days_val -0.535279088  0.14566906 -0.20841765 -0.27111313 -0.12282054
## cool_days_ano -0.040277155 -0.13896581  0.07097357  0.05653813  0.02119351
## heat_days_val -0.144672102  0.24123991 -0.01388207  0.72052606  0.14488166
## heat_days_ano  0.703006532  0.25343891 -0.12219537 -0.13795239 -0.02100856
##                        PC11          PC12
## av_temp_val   -0.0907356186  8.102758e-01
## av_temp_ano   -0.7992345395 -9.075989e-02
## min_temp_val   0.0569819036 -3.810232e-01
## min_temp_ano   0.3751177278  4.263284e-02
## max_temp_val   0.0392901580 -4.307304e-01
## max_temp_ano   0.4554917620  5.187126e-02
## prec_val      -0.0001431338 -4.445399e-05
## prec_ano      -0.0005087485  3.844764e-05
## cool_days_val -0.0026271071  2.886470e-04
## cool_days_ano -0.0002386775 -6.742599e-05
## heat_days_val  0.0024253203 -4.544846e-04
## heat_days_ano -0.0011702009  6.779625e-05
```

```r
plot(temp.pca, type = "l")
```




```r
# summary method
summary(temp.pca)
```

```
## Importance of components:
```

```
##                          PC1    PC2    PC3     PC4     PC5     PC6
## Standard deviation     2.3433 1.9272 1.3435 0.74991 0.40448 0.34517
## Proportion of Variance 0.4576 0.3095 0.1504 0.04686 0.01363 0.00993
## Cumulative Proportion  0.4576 0.7671 0.9175 0.96439 0.97802 0.98795
##                            PC7    PC8     PC9    PC10     PC11      PC12
## Standard deviation     0.30625 0.2219 0.03451 0.01932 0.003019 0.0001486
## Proportion of Variance 0.00782 0.0041 0.00010 0.00003 0.000000 0.0000000
## Cumulative Proportion  0.99577 0.9999 0.99997 1.00000 1.000000 1.0000000
```

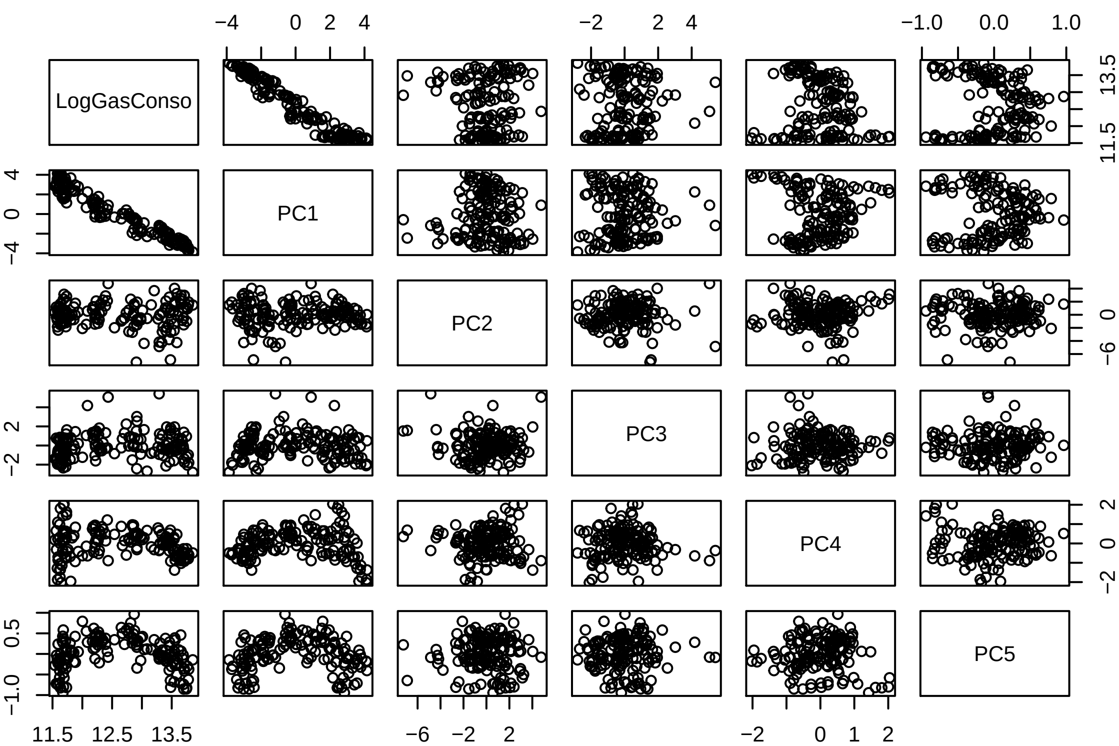
As we can see, we have a drop in the variance explained starting PCA 5 (all equal to 0 after PCA 10, that's why they are not in the graph). Let's look at the correlations with the log gas consumption :

```r
cor(all[,c(3)],temp.pca$x)
```

```
##              PC1        PC2        PC3        PC4         PC5         PC6
## [1,] -0.9678181 -0.0243758 0.06514828 -0.1422299 -0.05015072 -0.02503642
##            PC7        PC8        PC9        PC10       PC11       PC12
## [1,] 0.1085916 0.05840695 0.02430098 -0.06925745 -0.0240102 -0.0213973
```

The correlation of the log gas consumption and the PC1 is almost as good as the one between min_temp_val and the log gas consumption. It could be interesting to test the different results on the predictions (with min_temp_val and with PC1).

```r
test <- cbind(all[,c(3)],temp.pca$x[,c(1,2,3,4,5)])
colnames(test) <- c('LogGasConso', colnames(temp.pca$x[,c(1,2,3,4,5)]))
pairs(test)
```



Granger-causality test between PC1 and log gas consumption is thus :

```r
# test for unit root and number of differences required
ndiffs(temp.pca$x[,1], alpha=0.05, test=c("kpss"))
```

```
## [1] 0
```

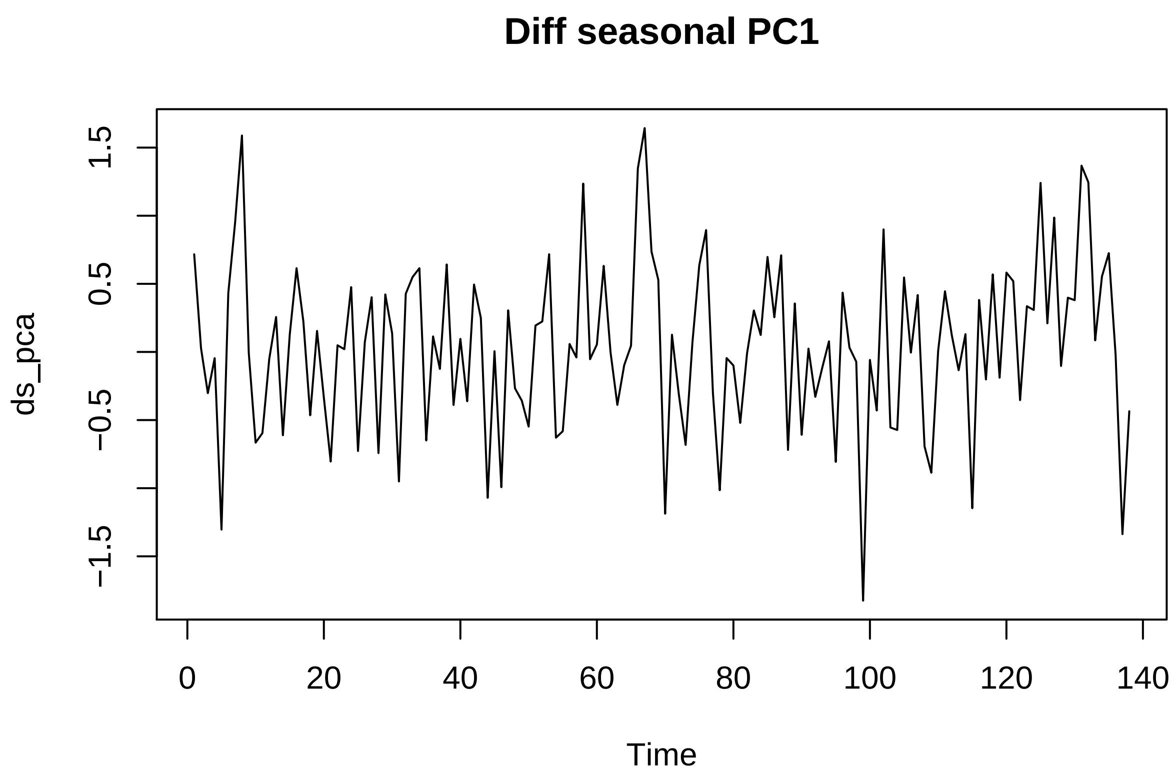
```r
# test for unit roots in seasonality
nsdiffs(temp.pca$x[,1], m=12, test=c("ocsb"))
```

```
## [1] 1
```

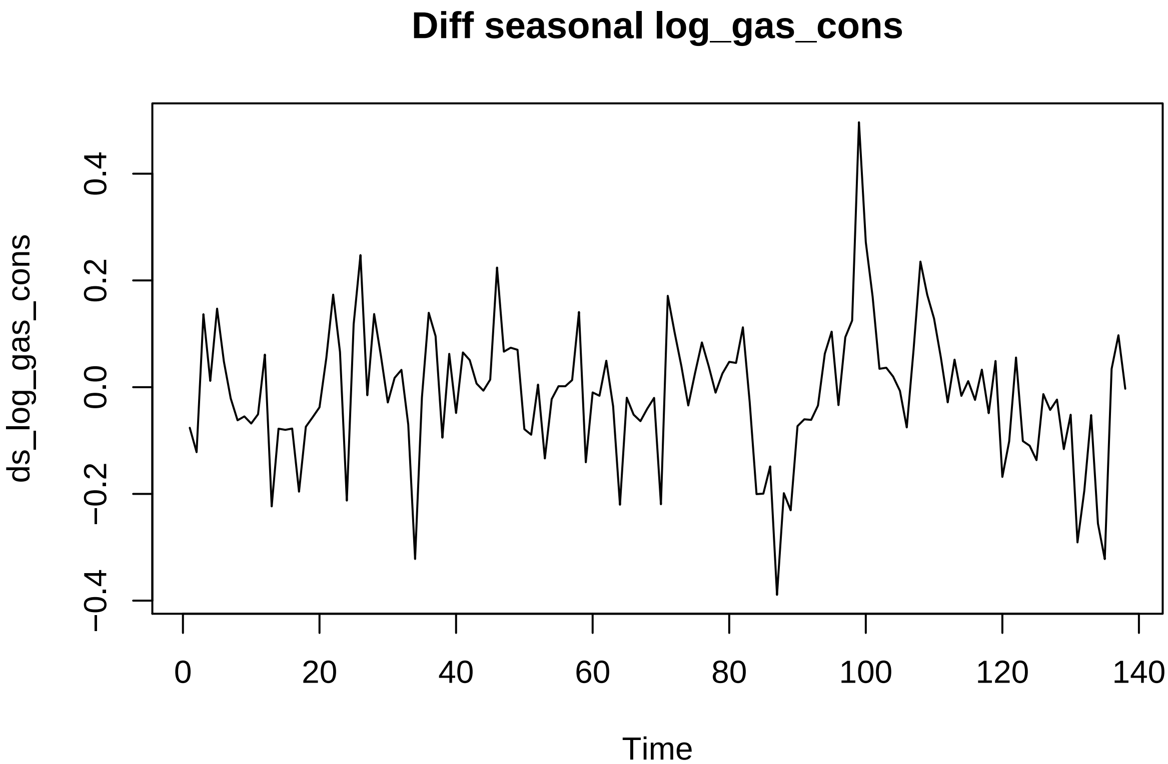
```r
# test for unit roots in the first seasonal difference
nsdiffs(diff(temp.pca$x[,1], lag = 12), m=12, test=c("ocsb"))
```

```
## [1] 0
```

```r
# Doing the difference
ds_pca <- diff(temp.pca$x[,1], lag = 12)

# Plots
plot.ts(ds_pca, main = "Diff seasonal PC1")
```




```r
plot.ts(ds_log_gas_cons, main = "Diff seasonal log_gas_cons")
```



The problem is that the PC1 never Granger-causes the log gas consumption ! So, unfortunately, we will not be using it in our forecasts.

```
grangertest(ds_log_gas_cons ~ ds_pca, order = 11)
```

```
## Granger causality test
##
## Model 1: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:11) + Lags(ds_pca, 1:11)
## Model 2: ds_log_gas_cons ~ Lags(ds_log_gas_cons, 1:11)
##   Res.Df  Df     F Pr(>F)
## 1    104
## 2    115 -11 0.916  0.528
```