Macroeconomics Project

Samuel and Matthias

07/06/2021

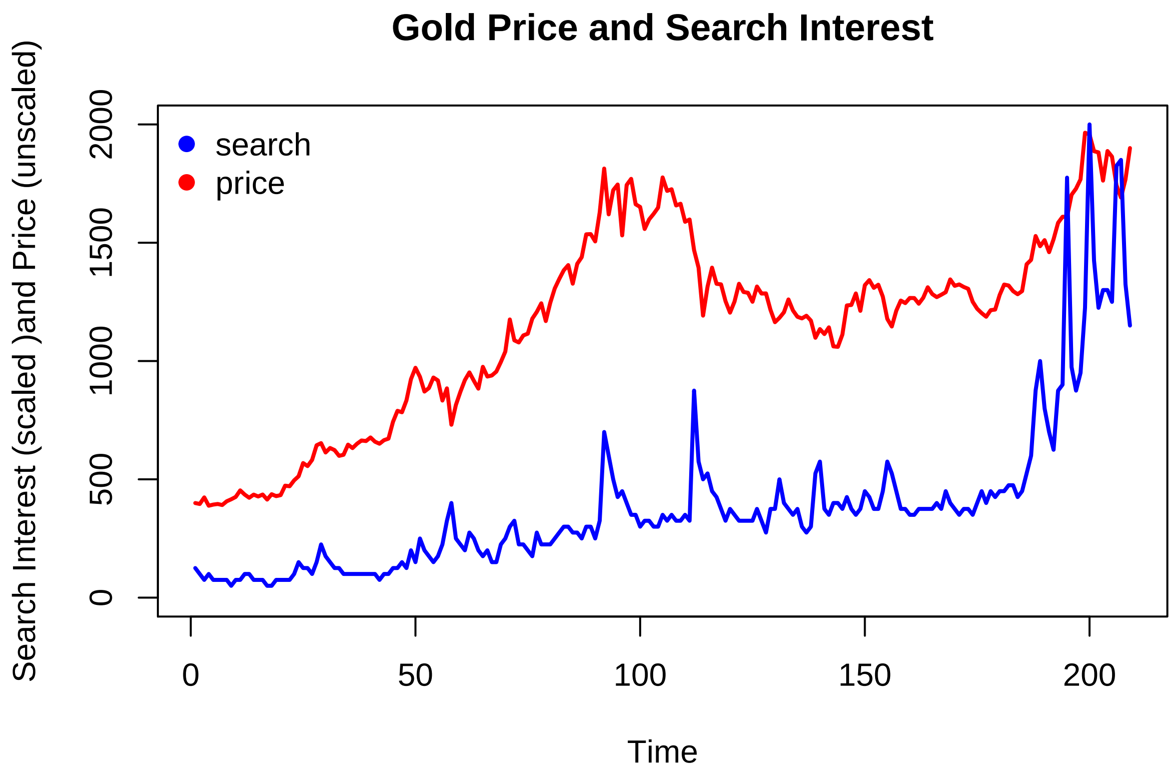
Idea

We want to look at the relationship between certain prices and the respective search interest on google for these prices. Can we find granger causality for this relationship? What are possible issues? For example: modern trading algorithms scrape data from the internet and then buy or sell based on the sentiment. Large spikes in search interest may trigger such algorithms. As media spreads the news of price increases more people will look up prices of goods and commodities, again triggering the algorithms. This is basically a feedback loop.

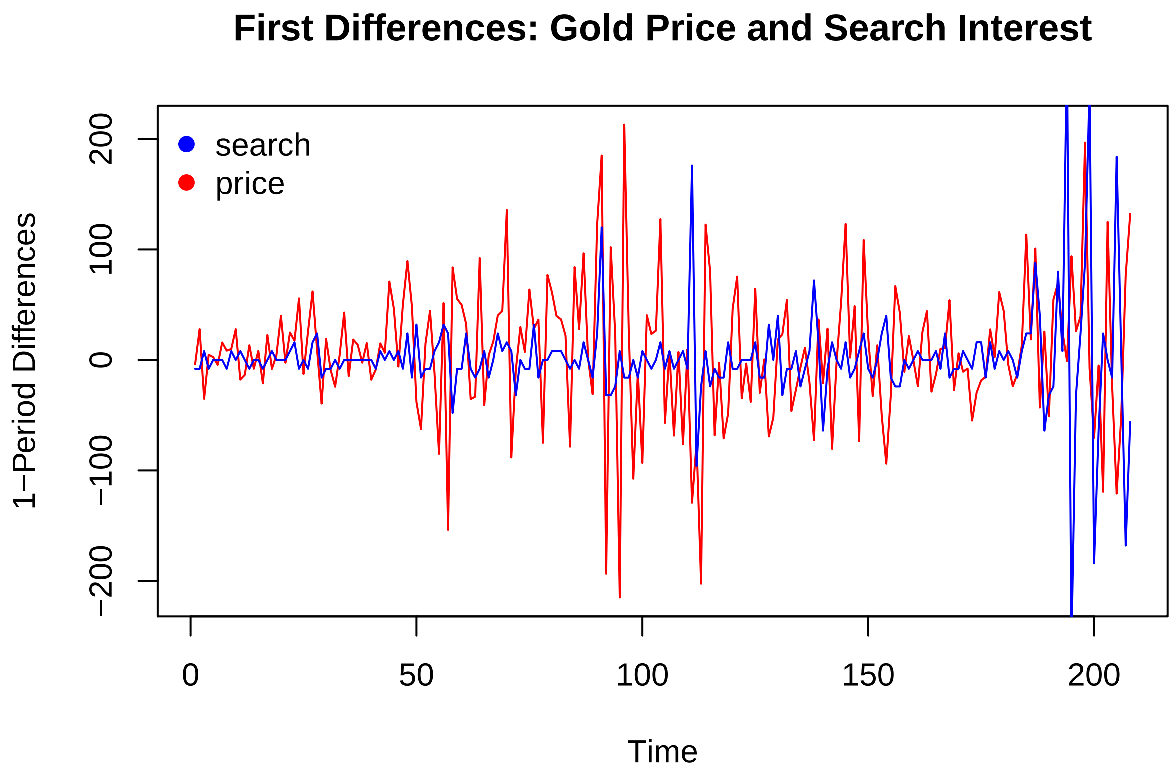
Project Code

```
# clear workspace
rm(list=ls())

# load needed libraries
library(readr)
library(vars)

# set working directory
setwd("/Users/samue/Downloads/Studium/Economics (Master - Vienna)/2. Semester/Macroeconometrics/Project/

# import search trends
data <- read.csv("btc-vs-gold-2004.csv")
# import prices data:
gold_pr <- read.csv("gold-2004.csv")
```

```
# plot gold price
plot(gold_pr$GOLDPMGBD228NLBM,type = 'l', lwd = 2, col = 'red',
     ylim = c(0,2000), main = 'Gold Price and Search Interest',
     xlab = 'Time', ylab = 'Search Interest (scaled )and Price (unscaled)')
# add gold search interest scaled up
lines(25*data$GOLD, lwd = 2, col = 'blue')
legend('topleft', legend = c('search','price'),
       col = c('blue','red'), bty = "n", pch = c(19,19))
```

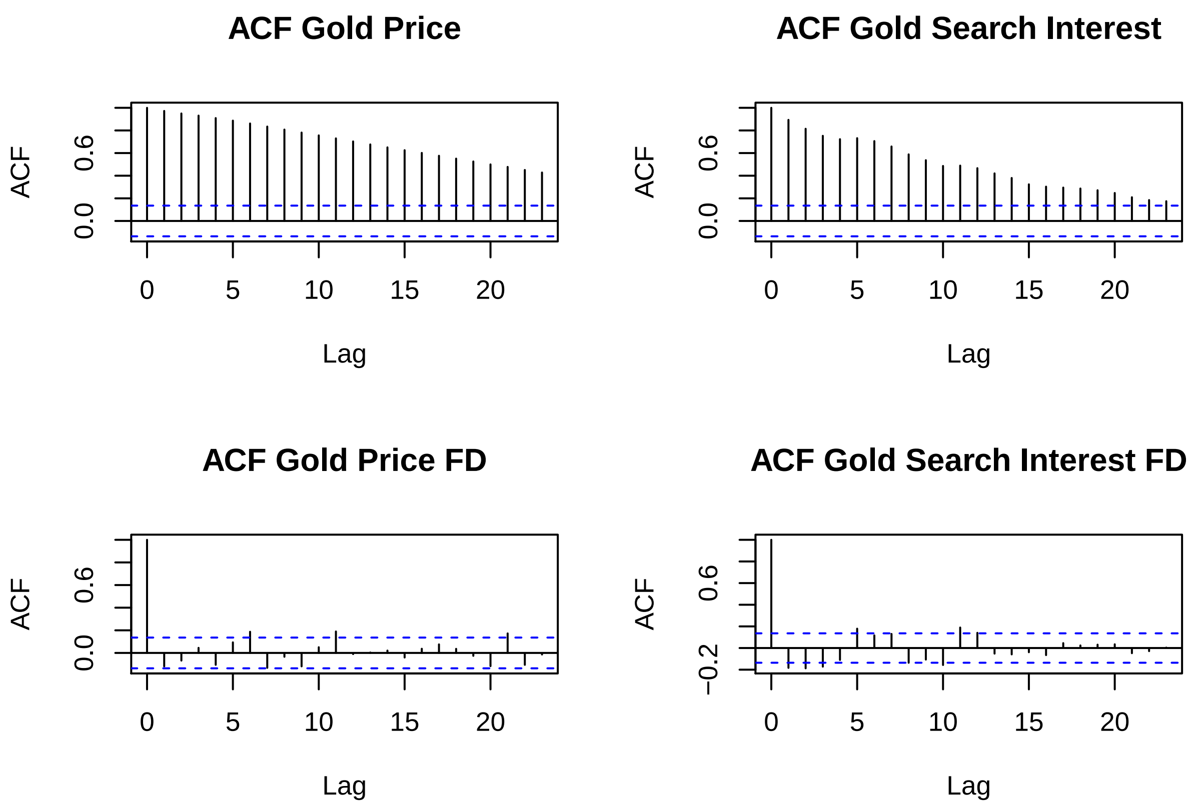



```
# create first differenced prices and search interest
t <- length(gold_pr$DATE)
gold_price_FD <- rep(0,t-1)
for(i in 2:209){gold_price_FD[i-1] <- gold_pr$GOLDPMGBD228NLBM[i]-gold_pr$GOLDPMGBD228NLBM[i-1]}
gold_search_FD <- rep(0,t-1)
for(i in 2:209){gold_search_FD[i-1] <- data$GOLD[i]-data$GOLD[i-1]}

# plot first differenced variables
plot(gold_price_FD, type = 'l', lwd = 1, col = 'red',
     xlab = 'Time', ylab = '1-Period Differences',
     main = 'First Differences: Gold Price and Search Interest')
lines(gold_search_FD*8, lwd = 1, col = 'blue')
legend('topleft', legend = c('search','price'),
       col = c('blue','red'), bty = "n", pch = c(19,19))
```



It appears that the more volatile regions match. Issue seems to be the scaling of the variables.

```
# plot ACF for unmodified variables:
par(mfrow=c(2,2))      # changes the plot layout to more easily compare them
acf(gold_pr$GOLDPMGBD228NLBM, main = 'ACF Gold Price')
acf(data$GOLD, main = 'ACF Gold Search Interest')

# plot ACF for differenced variables
acf(gold_price_FD,main = 'ACF Gold Price FD')
acf(gold_search_FD, main = 'ACF Gold Search Interest FD')
```

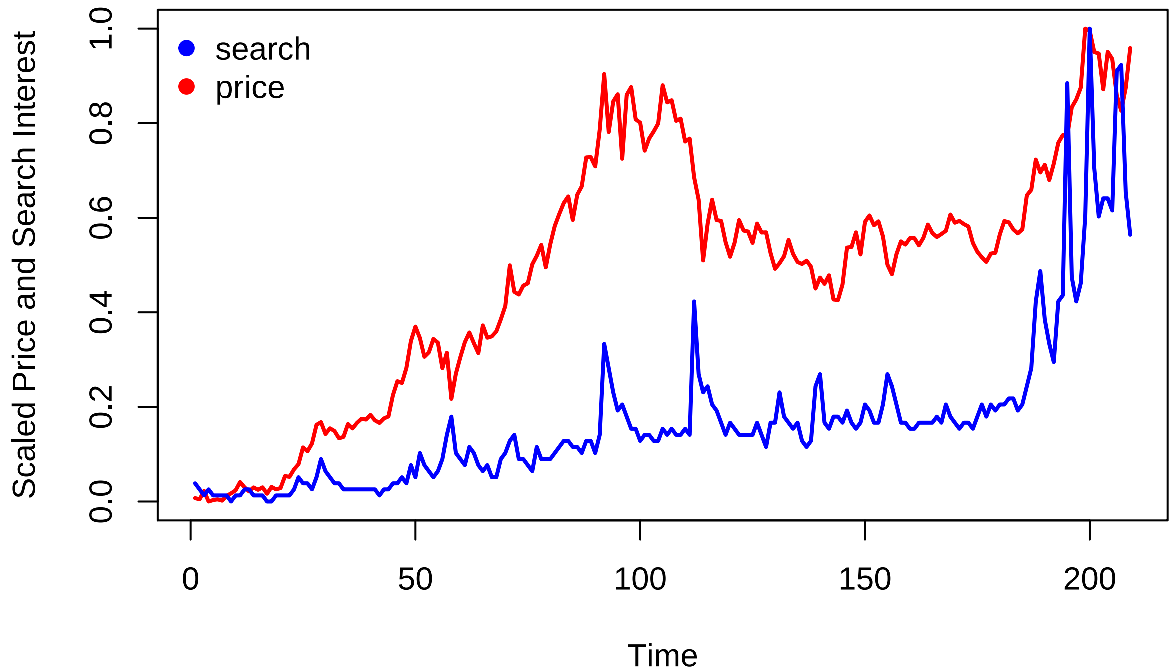



```
par(mfrow = c(1,1))    # revert layout changes
```

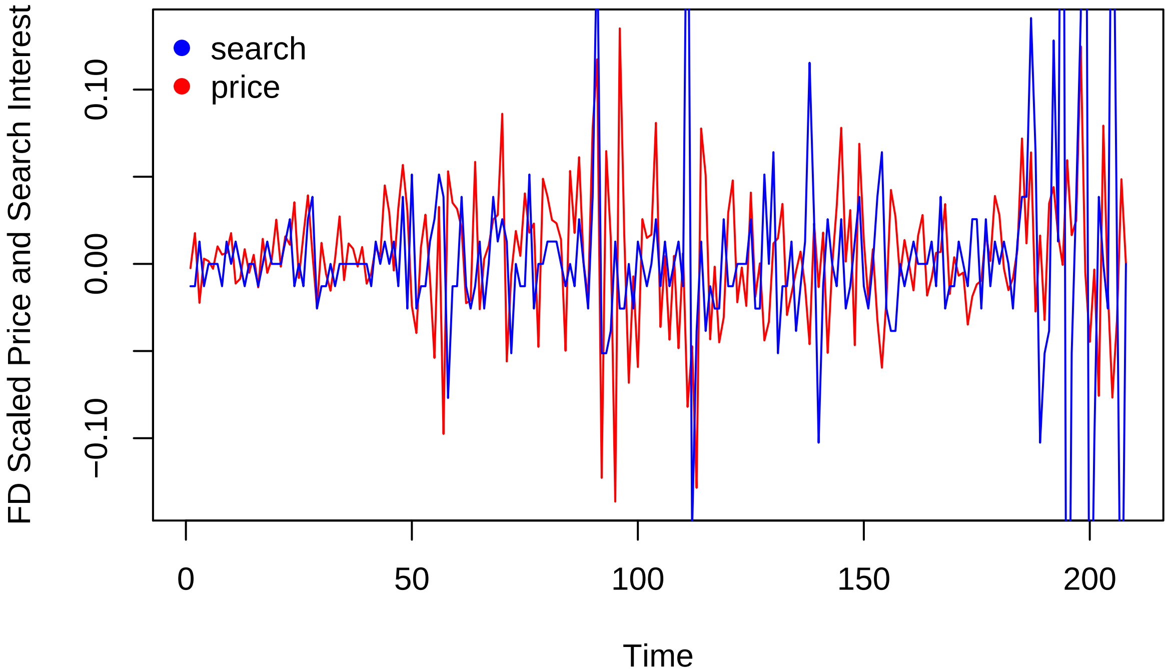
Autocorrelation for the differenced variables seems like no month-on-month relationship between the changes. Kind of like a random walk?

Might help with the interpretation: scale all variables $\mathbf{X}$ such that $X_t \in [0,1] \forall t \in T$.

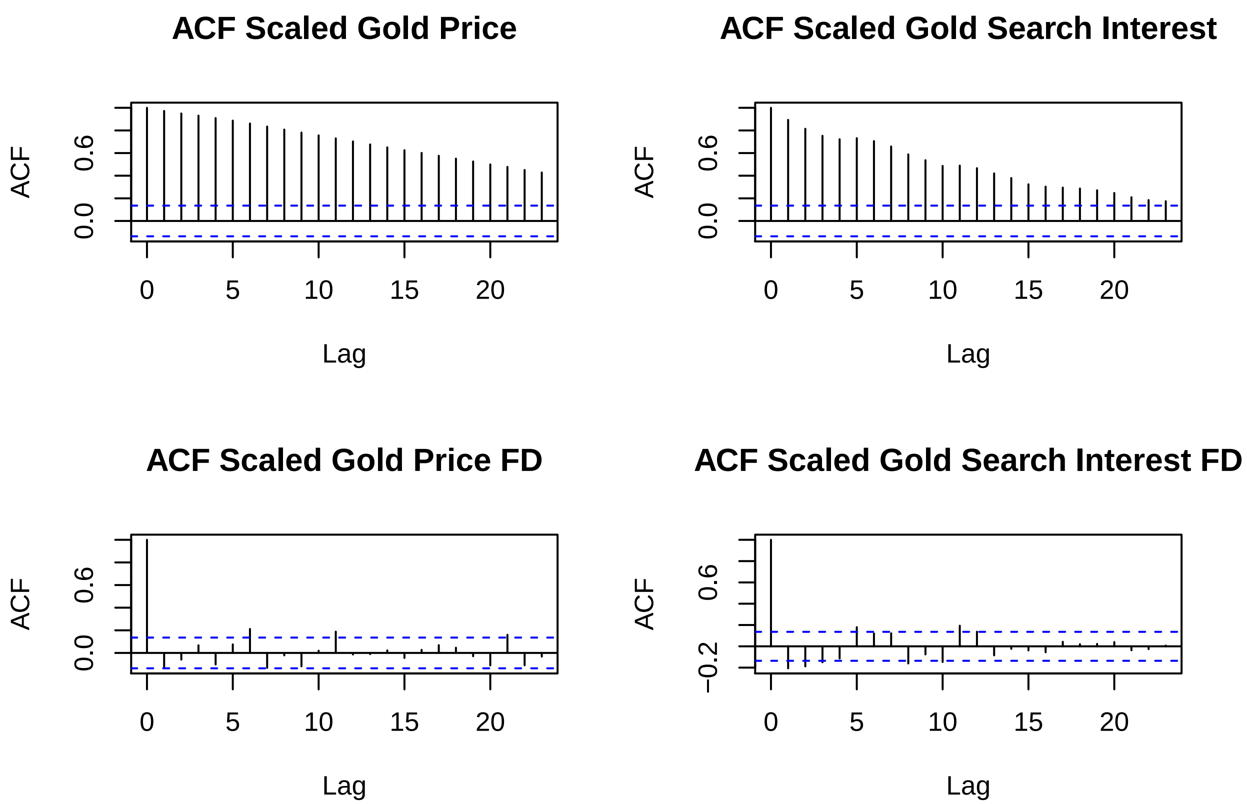
```
range01 <- function(x){(x-min(x))/(max(x)-min(x))}
plot(range01(gold_pr$GOLDPMGBD228NLBM), lwd = 2, type = 'l',
     ylab = 'Scaled Price and Search Interest',
     xlab = 'Time', col = 'red')
lines(range01(data$GOLD), lwd = 2, col = 'blue')
legend('topleft', legend = c('search','price'),
       col = c('blue','red'), bty = "n", pch = c(19,19))
```




```
# save scaled variables
gold_price_scaled <- range01(gold_pr$GOLDPMGBD228NLBM)
gold_search_scaled <- range01(data$GOLD)

# create first difference on scaled variables:
gold_search_scaled_FD <- rep(0,t-1)
gold_price_scaled_FD <- rep(0,t-1)

for(i in 2:t-1){
  gold_price_scaled_FD[i-1] <- gold_price_scaled[i]-gold_price_scaled[i-1]
}
for(i in 2:t-1){
  gold_search_scaled_FD[i-1] <- gold_search_scaled[i]-gold_search_scaled[i-1]
}

# plot first differenced:
plot(gold_price_scaled_FD, lwd = 1, type = 'l',
     ylab = 'FD Scaled Price and Search Interest',
     xlab = 'Time', col = 'red')
lines(gold_search_scaled_FD, lwd = 1, col = 'blue')
legend('topleft', legend = c('search','price'),
       col = c('blue','red'), bty = "n", pch = c(19,19))
```




```r
# plot ACFs
par(mfrow=c(2,2))      # changes the plot layout to more easily compare them
acf(gold_price_scaled, main = 'ACF Scaled Gold Price')
acf(gold_search_scaled, main = 'ACF Scaled Gold Search Interest')
acf(gold_price_scaled_FD,main = 'ACF Scaled Gold Price FD')
acf(gold_search_scaled_FD, main = 'ACF Scaled Gold Search Interest FD')
```




```r
par(mfrow = c(1,1))    # revert layout changes
```

Unsurprisingly the rescaling does not matter for the autocorrelation as it is a scaled measure of linear relationships anyway.

```r
# save variable vectors as time series format:
gold_price_scaled <- ts(gold_price_scaled, frequency = 12,
                        start = c(2004, 1), end = c(2021, 5))
gold_search_scaled <- ts(gold_search_scaled, frequency = 12,
                         start = c(2004,1), end = c(2021,5))

# set up data for estimation using `VAR()`
VAR_data <- window(ts.union(gold_price_scaled, gold_search_scaled),
                   start = c(2004, 1), end = c(2021, 5))

# estimate model coefficients using `VAR()`
VAR_est <- VAR(y = VAR_data, p = 2)
summary(VAR_est)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: gold_price_scaled, gold_search_scaled
## Deterministic variables: const
## Sample size: 207
## Log Likelihood: 636.839
## Roots of the characteristic polynomial:
```

```
## 0.9924 0.8559 0.1625 0.1625
## Call:
## VAR(y = VAR_data, p = 2)
##
##
## Estimation results for equation gold_price_scaled:
## ==================================================
## gold_price_scaled = gold_price_scaled.l1 + gold_search_scaled.l1 + gold_price_scaled.l2 + gold_searc
##
##                        Estimate Std. Error t value Pr(>|t|)
## gold_price_scaled.l1   0.869163   0.070442  12.339   <2e-16 ***
## gold_search_scaled.l1 -0.020645   0.039085  -0.528   0.5979
## gold_price_scaled.l2   0.110635   0.070855   1.561   0.1200
## gold_search_scaled.l2  0.042045   0.039316   1.069   0.2862
## const                  0.011315   0.005945   1.903   0.0584 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.0395 on 202 degrees of freedom
## Multiple R-Squared: 0.9771,  Adjusted R-squared: 0.9766
## F-statistic:  2151 on 4 and 202 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation gold_search_scaled:
## ===================================================
## gold_search_scaled = gold_price_scaled.l1 + gold_search_scaled.l1 + gold_price_scaled.l2 + gold_sear
##
##                       Estimate Std. Error t value Pr(>|t|)
## gold_price_scaled.l1   0.34683    0.12492   2.776  0.00601 **
## gold_search_scaled.l1  0.74430    0.06931  10.738  < 2e-16 ***
## gold_price_scaled.l2  -0.26598    0.12565  -2.117  0.03551 *
## gold_search_scaled.l2  0.10172    0.06972   1.459  0.14615
## const                 -0.01107    0.01054  -1.050  0.29498
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.07006 on 202 degrees of freedom
## Multiple R-Squared: 0.8372,  Adjusted R-squared: 0.834
## F-statistic: 259.8 on 4 and 202 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##                    gold_price_scaled gold_search_scaled
## gold_price_scaled          1.561e-03         -4.454e-05
## gold_search_scaled        -4.454e-05          4.908e-03
##
## Correlation matrix of residuals:
##                    gold_price_scaled gold_search_scaled
## gold_price_scaled             1.0000            -0.0161
## gold_search_scaled           -0.0161             1.0000
```