CPIAUCSL Analysis

Vajinder

2024-11-11

```r
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 4.4.2
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.4.1
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(urca)
```

```
## Warning: package 'urca' was built under R version 4.4.2
```

```r
library(dynlm)
```

```
## Warning: package 'dynlm' was built under R version 4.4.2
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.4.1
```

```
##
## Attaching package: 'zoo'
```

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

```
library(car)
```

```
## Warning: package 'car' was built under R version 4.4.1

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.4.1

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.4.1
```

Forecasting models for the rate of inflation - Guidelines

Go to FRED's website (https://fred.stlouisfed.org/) and download the data for:

Consumer Price Index for All Urban Consumers: All Items (CPIAUCSL) - Seasonally adjusted – Monthly Frequency – From 1947:M1 to 2017:M12 In this hands-on exercise you will construct forecasting models for the rate of inflation, based on CPIAUCSL. For this analysis, use the sample period 1970:M01–2012:M12 (where data before 1970 should be used, as necessary, as initial values for lags in regressions).

```
CPI <- read.csv("CPIAUCSL_W.csv")
CPI <- na.omit(CPI)
CPI$DATE <- as.Date(CPI$DATE)
CPI1970_2012 <- subset(CPI, DATE >= "1970-01-01" & DATE <= "2012-12-31")
```

A

(i) Compute the (annualized from month-over-month change) inflation rate for each month.
(ii) Plot the value of Infl from 1970:M01 through 2012:M12. Based on the plot, do you think that Infl has a stochastic trend? Explain.

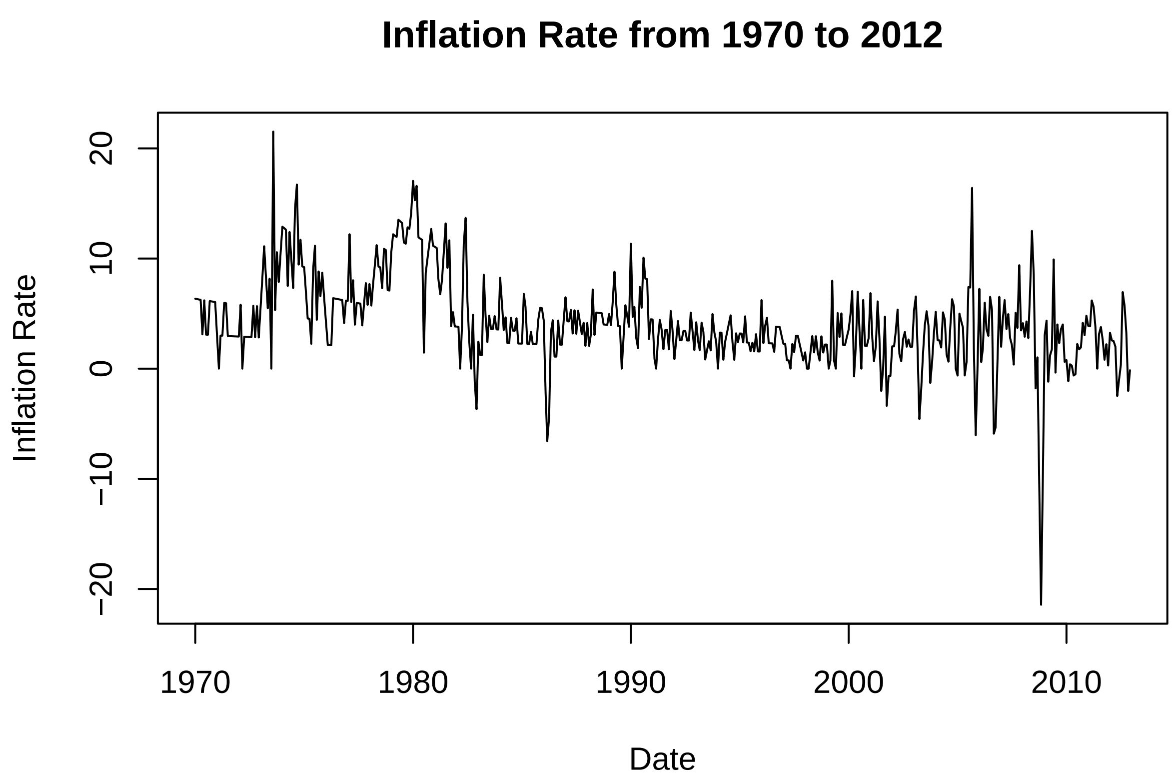
- Based on the plot, we can say that it is not stochastic. Because values are all coming back to inflation rate of approx. 5%. Even if there has been large dip and even high, overall I can visualize a line around 5% where average would lie.

```r
library(dplyr)
inflation_rate <- (log(CPI1970_2012$CPIAUCSL) - log(lag(CPI1970_2012$CPIAUCSL, 1))) * 12 * 100

CPI1970_2012$infl <- inflation_rate

cpi_dec1969 <- CPI$CPIAUCSL[which(CPI$DATE == "1969-12-01")]
cpi_nov1969 <- CPI$CPIAUCSL[which(CPI$DATE == "1969-11-01")]
Infl_dec1969 <- (log(cpi_dec1969) - log(cpi_nov1969)) * 12 * 100
CPI1970_2012$infl[1] <- (log(CPI1970_2012$CPIAUCSL[1]) - log(cpi_dec1969)) * 12 * 100

plot(CPI1970_2012$DATE, CPI1970_2012$infl,
     type = "l", xlab = "Date", ylab = "Inflation Rate", main = "Inflation Rate from 1970 to 2012")
```



B

(i) Compute the first twelve autocorrelations of infl
(ii) Plot the value of ACF from 1970:M01 through 2012:M12. The plot should look "choppy" or "jagged." Explain why this behavior is consistent with the first autocorrelation that you computed in part (i) for

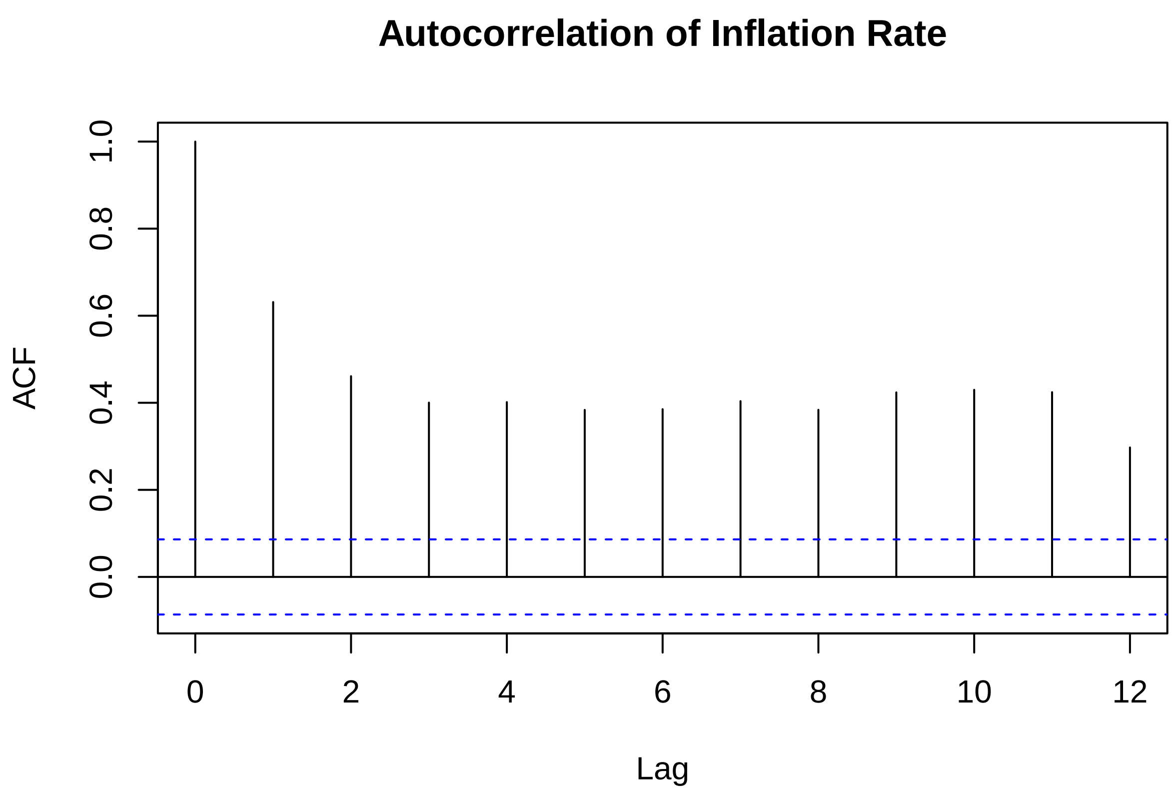
```r
acf(ts(CPI1970_2012$infl), lag.max = 12, plot = F, main = "Autocorrelation of Inflation Rate")
```

```
##
## Autocorrelations of series 'ts(CPI1970_2012$infl)', by lag
##
##     0     1     2     3     4     5     6     7     8     9    10    11    12
## 1.000 0.631 0.461 0.400 0.402 0.384 0.385 0.404 0.384 0.424 0.430 0.424 0.297
```

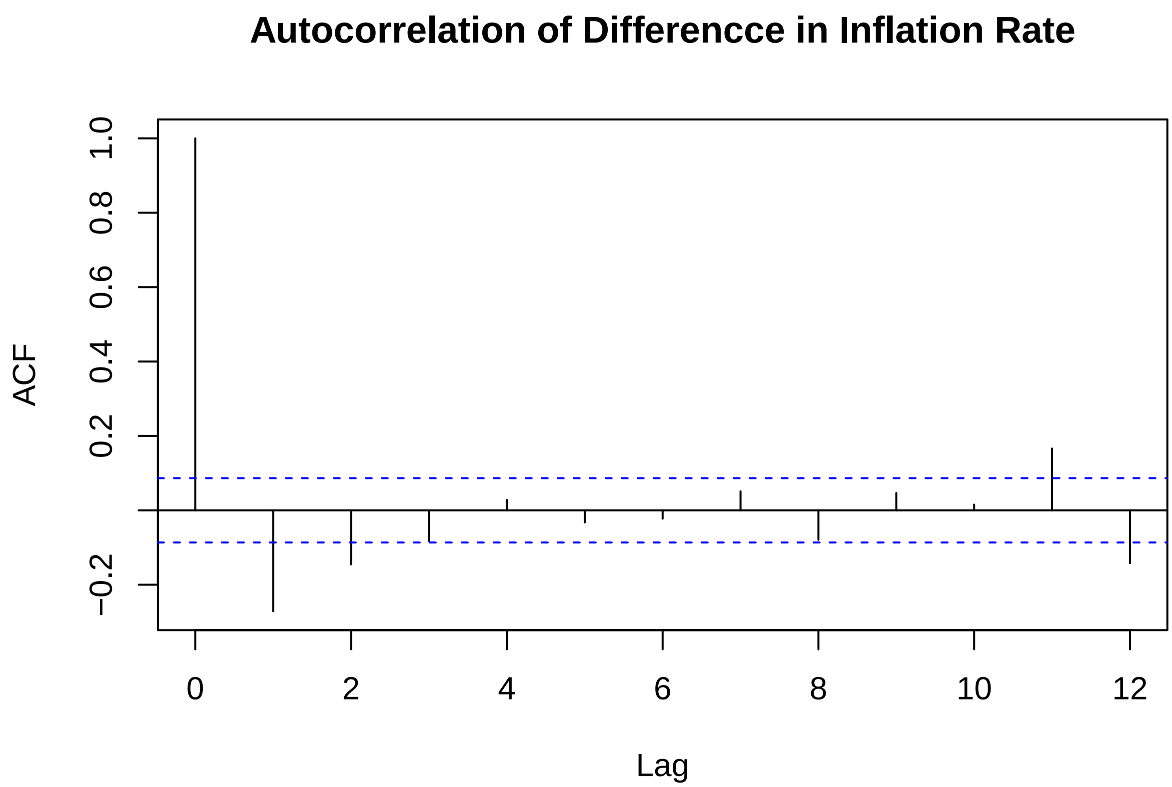
```r
acf(ts(diff(CPI1970_2012$infl)), lag.max = 12, plot = F, main = "Autocorrelation of Differencce in Infla
```

```
##
## Autocorrelations of series 'ts(diff(CPI1970_2012$infl))', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000 -0.271 -0.146 -0.082  0.028 -0.033 -0.023  0.051 -0.080  0.047  0.015
##     11     12
##  0.166 -0.142
```

```r
acf(ts(CPI1970_2012$infl), lag.max = 12, plot = T, main = "Autocorrelation of Inflation Rate")
```




```r
acf(ts(diff(CPI1970_2012$infl)), lag.max = 12, plot = T, main = "Autocorrelation of Differencce in Infla
```



C

(i) Compute Run an OLS regression of on - Does knowing the inflation this month help predict the inflation next month? Explain.
(ii) Estimate an AR(2) model for Infl. Is the AR(2) model better than an AR(1) model? Explain.
(iii) Estimate an AR(p) model for . What lag length is chosen by BIC? What lag length is chosen by AIC?
(iv) Use the AR(2) model to predict "the level of the inflation rate" in 2013:M01—that is

```
Model1 <- lm(infl ~ lag(infl, 1), data = CPI1970_2012)
summary(Model1)
```

```
##
## Call:
## lm(formula = infl ~ lag(infl, 1), data = CPI1970_2012)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.4167  -1.4655  -0.2554   1.6150  19.9865
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.53876    0.19916   7.726 5.87e-14 ***
## lag(infl, 1)  0.63293    0.03423  18.489  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 3.106 on 513 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.3999, Adjusted R-squared:  0.3987
## F-statistic: 341.9 on 1 and 513 DF,  p-value: < 2.2e-16
```

```r
Model2 <- lm(infl ~ lag(infl, 1) + lag(infl, 2), data = CPI1970_2012)

summary(Model2)
```

```
##
## Call:
## lm(formula = infl ~ lag(infl, 1) + lag(infl, 2), data = CPI1970_2012)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.0555  -1.4247  -0.2298   1.5675  19.2838
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.36972    0.21027   6.514 1.75e-10 ***
## lag(infl, 1)  0.56554    0.04398  12.858  < 2e-16 ***
## lag(infl, 2)  0.10667    0.04408   2.420   0.0159 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.094 on 511 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.4064, Adjusted R-squared:  0.4041
## F-statistic:   175 on 2 and 511 DF,  p-value: < 2.2e-16
```

```r
IC <- function(model) {
  SSR <- sum(model$residuals^2)
  T <- length(model$residuals)
  p <- length(model$coef)

  return(
    round(c("Lags" = p - 1,
            "AIC" = AIC(model),
            "BIC" = BIC(model),
            "R.Square" = summary(model)$r.squared), 4)
  )
}
max_lag <- 1:6

table <- sapply(max_lag, function(max_lag) {
  formula <- as.formula(paste("infl ~", paste("lag(infl,", 1:max_lag, ")", collapse = " + ")))

  model <- lm(formula, data = CPI1970_2012)
  IC(model)
})
IC_table <- as.data.frame(t(table))
print(IC_table)
```

```
##   Lags      AIC      BIC R.Square
## 1    1 2632.824 2645.557   0.3999
## 2    2 2624.805 2641.774   0.4064
## 3    3 2614.746 2635.947   0.4153
## 4    4 2601.462 2626.892   0.4277
## 5    5 2595.665 2625.320   0.4317
## 6    6 2587.093 2620.968   0.4387
```

```
best_by_AIC <- IC_table[which.min(IC_table$AIC),]
best_by_BIC <- IC_table[which.min(IC_table$BIC),]
best_by_R2 <- IC_table[which.max(IC_table$R.Square),]

inflation_nov_dec_2012 <- filter(CPI1970_2012, format(CPI1970_2012$DATE, "%Y-%m") %in% c("2012-11", "20

infl_2012_12 <- inflation_nov_dec_2012 |>
  filter(format(DATE, "%m") == "12") |>
  pull(infl)

infl_2012_11 <- inflation_nov_dec_2012 |>
  filter(format(DATE, "%m") == "11") |>
  pull(infl)

infl_2012_12
```

```
## [1] -0.1453067
```

```
infl_2012_11
```

```
## [1] -2.016907
```

```
beta0 <- coef(Model2)[1]
beta1 <- coef(Model2)[2]
beta2 <- coef(Model2)[3]

pred_infl_2013_01 <-beta0 + beta1 * infl_2012_12 + beta2* infl_2012_11
pred_infl_2013_01
```

```
## (Intercept)
##    1.072409
```

#D Use the ADF test for the regression in Equation (15.32) with two lags of to test for a stochastic trend in . Does the Inflation rate has a units root?

```
CPI1970_2012 <- CPI1970_2012 |>
  mutate(Delta_Infl = c(NA, diff(infl)))
CPI1970_2012$Delta_Infl[1] <- CPI1970_2012$infl[1] - Infl_dec1969
CPI1970_2012 <- CPI1970_2012 %>%
  mutate(
    Delta_Infl_LAG1 = lag(Delta_Infl, 1),
    Delta_Infl_LAG2 = lag(Delta_Infl, 2)
  )
```

```r
adf_test_without_t <- ur.df(CPI1970_2012 $Delta_Infl, type = "drift", lags = 2, selectlags = "Fixed")

summary(adf_test_without_t)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression drift
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + z.diff.lag)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.9238  -1.5024  -0.0506   1.6759  18.5859
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.02104    0.13880  -0.152     0.88
## z.lag.1     -1.92702    0.09558 -20.161  < 2e-16 ***
## z.diff.lag1  0.53839    0.07162   7.517 2.54e-13 ***
## z.diff.lag2  0.22522    0.04335   5.195 2.96e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.144 on 509 degrees of freedom
## Multiple R-squared:  0.6733, Adjusted R-squared:  0.6714
## F-statistic: 349.7 on 3 and 509 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -20.1611 203.2361
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau2 -3.43 -2.86 -2.57
## phi1  6.43  4.59  3.78
```

```r
t <- 1:nrow(CPI1970_2012 )

eq02 <- lm(Delta_Infl ~ t + Delta_Infl_LAG1 + Delta_Infl_LAG2, data = CPI1970_2012 )

residuals_eq02 <- residuals(eq02)

adf_test_with_t <- ur.df(residuals_eq02, type = "drift", lags = 2, selectlags = "Fixed")
summary(adf_test_with_t)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
```

```
## ###############################################
##
## Test regression drift
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + z.diff.lag)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.1208  -1.4149   0.0137   1.5769  18.0719
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.01200    0.13715   0.087     0.93
## z.lag.1     -1.44741    0.07928 -18.256  < 2e-16 ***
## z.diff.lag1  0.36661    0.06255   5.861 8.28e-09 ***
## z.diff.lag2  0.26272    0.04300   6.110 1.99e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.1 on 507 degrees of freedom
## Multiple R-squared:  0.5628, Adjusted R-squared:  0.5602
## F-statistic: 217.6 on 3 and 507 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -18.2561 166.6427
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau2 -3.43 -2.86 -2.57
## phi1  6.43  4.59  3.78
```

#E Use the QLR test with 15% trimming to test the stability of the coefficients in the AR(2) model for "the inflation" Infl. (You cannot use the strucchange package. You must demonstrate that you understand how the QLR test is structured) Is the AR(2) model stable? Explain.

```r
library(timeSeries)
```

```
## Warning: package 'timeSeries' was built under R version 4.4.2
```

```
## Loading required package: timeDate
```

```
## Warning: package 'timeDate' was built under R version 4.4.1
```

```
##
## Attaching package: 'timeSeries'
```

```
## The following object is masked from 'package:zoo':
##
##     time<-
```

```
## The following object is masked from 'package:dplyr':
##
##     lag

## The following objects are masked from 'package:graphics':
##
##     lines, points
```

```r
library(dynlm)
library(car)

# Ensure infl is a time series
CPI1970_2012$infl <- ts(CPI1970_2012$infl, start = c(1970, 1), frequency = 12)

# Create a time vector from the time series
Annual_Infl_ts <- CPI1970_2012$infl
TV <- time(Annual_Infl_ts)

# Parameters for the time window
w <- 0.15
NOB <- length(TV)
tau0 <- TV[round(NOB * w, 0)]
tau1 <- TV[round(NOB * (1 - w), 0)]
tau <- seq(tau0, tau1, 0.083)  # Time points for rolling windows

# Initialize vector for F-statistics
Fstats <- numeric(length(tau))

# Loop to compute F-statistics
for (i in 1 : length(tau) ) {
  # Create dummy variable D
  D <- ifelse(TV >= tau[i], 1, 0)
  # estimate ADL(2,2) model with intercations
  eq <- dynlm(infl ~ L(infl)+L(infl,2) +
          D*L(infl) , CPI1970_2012)
  coefs <- names(coef(eq)) # get the names of the coefs

  # compute and save the F-statistic
  Fstats[i] <- linearHypothesis(eq,c("D=0","L(infl):D"))$F[2]

}

QLR <- max(Fstats)
QLR
```

```
## [1] 25.72615
```

```r
as.yearqtr(tau[which.max(Fstats)])
```

```
## [1] "1982 Q2"
```

#F

(i)Using the AR(2) model for Infl with a sample period that begins in 1970:M01, compute pseudo out-of-sample forecasts for the inflation beginning in 2005:M12 and going through 2012:M12.

(ii)Are the pseudo out-of-sample forecasts biased? That is, do the forecast errors have a nonzero mean? Yes, Forecast Errors have a non zero mean. Though very neglible but still it is non zero.

(iii)How large is the RMSFE of the pseudo out-of-sample forecasts? Is this consistent with the AR(2) model for Infl estimated over the 1970:M01–2005:M12 sample period?

(iv)There is a large outlier in 2008:Q4. Why did inflation fall so much in 2008:Q4? (Hint: Collect some data on oil prices. What happened to oil prices during 2008?)

The large outlier in inflation during 2008:Q4 was caused by the collapse in oil prices, driven by the 2008 Global Financial Crisis. This significant economic downturn sharply reduced global demand for energy and other commodities, resulting in lower inflation. This event highlights the sensitivity of inflation to oil price fluctuations and broader economic conditions.

June 2008:

Price Peak: WTI crude oil prices reached an all-time high of $147 per barrel (in July, but started dropping in late June). Reasons for High Prices: Strong global demand, particularly from emerging markets like China and India. Supply concerns due to geopolitical tensions in oil-producing regions (e.g., Middle East). Speculative trading in commodities markets. December 2008:

Price Low: By December 2008, prices plummeted to around $30-$40 per barrel. Reasons for the Decline: Global Financial Crisis: The 2008 global financial meltdown caused a sharp contraction in economic activity, reducing oil demand significantly. Collapse in Consumer and Industrial Demand: With industries slowing and consumer spending declining, oil consumption fell globally. Strengthening of the U.S. Dollar: Oil prices are denominated in USD, and during the crisis, the USD strengthened, putting downward pressure on oil prices. Speculative Unwinding: Traders and hedge funds unwound speculative positions in oil, accelerating the price collapse.

Key Events Leading to the Fall Financial Crisis:

The collapse of Lehman Brothers in September 2008 intensified the global financial crisis. Banks reduced lending, and companies scaled back operations, reducing energy consumption. Reduced Global Demand:

Global GDP growth slowed significantly, reducing oil demand. Transportation and manufacturing, which are heavily oil-dependent, saw sharp declines in activity. Oil Market Oversupply:

While demand fell, production remained relatively stable in the short term, leading to an oversupply in the market. OPEC Response:

In late 2008, OPEC announced production cuts to stabilize prices, but the market continued to slide due to weak demand and high inventories. Economic Impact of the Price Drop Short-Term Relief: Lower oil prices eased inflationary pressures and reduced costs for consumers and businesses. Long-Term Consequences: Oil-exporting countries faced significant economic challenges due to reduced revenues. Data Sources to Analyze the Trend Use EIA or FRED to obtain historical monthly WTI crude oil prices. On Trading Economics or Investing.com, you can view the price chart for this period. The steep decline between June and December 2008 highlights the significant impact of macroeconomic events on oil markets.

```r
dates <- CPI1970_2012$DATE
forecast_start_date <- as.Date("2005-12-01")
forecast_end_date <- as.Date("2012-12-01")
forecast_period <- which(dates >= forecast_start_date & dates <= forecast_end_date)
estimation_period <- which(dates < forecast_start_date)
pseudo_forecasts <- numeric(length(forecast_period))
forecast_errors <- numeric(length(forecast_period))
for (i in 1:length(forecast_period)) {
  estimation_data <- CPI1970_2012[estimation_period, ]
```

```r
  ar2_model <- dynlm(infl ~ L(infl, 1) + L(infl, 2), data = estimation_data)
  forecast_value <- predict(ar2_model, newdata = CPI1970_2012[forecast_period[i], , drop = FALSE])
  actual_value <- CPI1970_2012$infl[forecast_period[i]]
  pseudo_forecasts[i] <- forecast_value
  forecast_errors[i] <- actual_value - forecast_value
}
forecast_results <- data.frame(
  Date = dates[forecast_period],
  Actual = CPI1970_2012$infl[forecast_period],
  Forecast = pseudo_forecasts,
  Error = forecast_errors
)
print(forecast_results)
```

```
##          Date      Actual      Forecast         Error
## 1  2005-12-01   0.0000000  4.107077e-15 -4.107077e-15
## 2  2006-01-01   7.2471282  7.247128e+00  1.776357e-15
## 3  2006-02-01   0.6019564  6.019564e-01 -3.552714e-15
## 4  2006-03-01   1.8040595  1.804059e+00 -2.442491e-15
## 5  2006-04-01   5.9940185  5.994018e+00  8.881784e-16
## 6  2006-05-01   3.5820922  3.582092e+00 -8.881784e-16
## 7  2006-06-01   2.9769303  2.976930e+00 -1.332268e-15
## 8  2006-07-01   6.5233667  6.523367e+00  1.776357e-15
## 9  2006-08-01   5.3110487  5.311049e+00  0.000000e+00
## 10 2006-09-01  -5.9026189 -5.902619e+00 -9.769963e-15
## 11 2006-10-01  -5.3372957 -5.337296e+00 -8.881784e-15
## 12 2006-11-01   0.5942065  5.942065e-01 -3.552714e-15
## 13 2006-12-01   6.5169254  6.516925e+00  1.776357e-15
## 14 2007-01-01   1.9894873  1.989487e+00 -2.220446e-15
## 15 2007-02-01   4.6450190  4.645019e+00  0.000000e+00
## 16 2007-03-01   6.2239770  6.223977e+00  8.881784e-16
## 17 2007-04-01   3.5954034  3.595403e+00 -8.881784e-16
## 18 2007-05-01   4.9493720  4.949372e+00  0.000000e+00
## 19 2007-06-01   2.7768866  2.776887e+00 -1.332268e-15
## 20 2007-07-01   2.1348150  2.134815e+00 -2.220446e-15
## 21 2007-08-01   0.3698798  3.698798e-01 -3.774758e-15
## 22 2007-09-01   5.0743203  5.074320e+00  0.000000e+00
## 23 2007-10-01   3.6941933  3.694193e+00 -8.881784e-16
## 24 2007-11-01   9.3937969  9.393797e+00  5.329071e-15
## 25 2007-12-01   3.4725881  3.472588e+00 -8.881784e-16
## 26 2008-01-01   4.1301304  4.130130e+00 -8.881784e-16
## 27 2008-02-01   2.8978904  2.897890e+00 -1.332268e-15
## 28 2008-03-01   4.2859703  4.285970e+00 -8.881784e-16
## 29 2008-04-01   2.7740484  2.774048e+00 -1.332268e-15
## 30 2008-05-01   7.0800625  7.080062e+00  1.776357e-15
## 31 2008-06-01  12.5084626  1.250846e+01  7.105427e-15
## 32 2008-07-01   8.5392785  8.539278e+00  3.552714e-15
## 33 2008-08-01  -1.7875015 -1.787502e+00 -5.551115e-15
## 34 2008-09-01   1.0256716  1.025672e+00 -3.108624e-15
## 35 2008-10-01 -10.3627397 -1.036274e+01 -1.243450e-14
## 36 2008-11-01 -21.4369129 -2.143691e+01 -2.131628e-14
## 37 2008-12-01  -9.9211261 -9.921126e+00 -1.243450e-14
## 38 2009-01-01   3.0330892  3.033089e+00 -1.332268e-15
```

```
## 39 2009-02-01   4.3632507  4.363251e+00 -8.881784e-16
## 40 2009-03-01  -1.1853246 -1.185325e+00 -5.107026e-15
## 41 2009-04-01   1.2078909  1.207891e+00 -2.886580e-15
## 42 2009-05-01   1.7644948  1.764495e+00 -2.442491e-15
## 43 2009-06-01   9.9184318  9.918432e+00  5.329071e-15
## 44 2009-07-01  -0.3576118 -3.576118e-01 -4.440892e-15
## 45 2009-08-01   4.0114317  4.011432e+00 -8.881784e-16
## 46 2009-09-01   2.3148306  2.314831e+00 -1.776357e-15
## 47 2009-10-01   3.5969220  3.596922e+00 -8.881784e-16
## 48 2009-11-01   4.0115959  4.011596e+00 -8.881784e-16
## 49 2009-12-01   0.6240494  6.240494e-01 -3.552714e-15
## 50 2010-01-01   0.7782263  7.782263e-01 -3.441691e-15
## 51 2010-02-01  -1.1426759 -1.142676e+00 -5.107026e-15
## 52 2010-03-01   0.3975759  3.975759e-01 -3.774758e-15
## 53 2010-04-01   0.2760169  2.760169e-01 -3.885781e-15
## 54 2010-05-01  -0.6238886 -6.238886e-01 -4.662937e-15
## 55 2010-06-01  -0.5026595 -5.026595e-01 -4.551914e-15
## 56 2010-07-01   2.2410104  2.241010e+00 -2.220446e-15
## 57 2010-08-01   1.7523561  1.752356e+00 -2.442491e-15
## 58 2010-09-01   1.9367356  1.936736e+00 -2.220446e-15
## 59 2010-10-01   4.1709584  4.170958e+00 -8.881784e-16
## 60 2010-11-01   3.0367642  3.036764e+00 -1.332268e-15
## 61 2010-12-01   4.8102377  4.810238e+00  0.000000e+00
## 62 2011-01-01   3.8853539  3.885354e+00 -4.440892e-16
## 63 2011-02-01   3.8511831  3.851183e+00 -4.440892e-16
## 64 2011-03-01   6.1922537  6.192254e+00  8.881784e-16
## 65 2011-04-01   5.6197393  5.619739e+00  8.881784e-16
## 66 2011-05-01   3.8119965  3.811997e+00 -4.440892e-16
## 67 2011-06-01   0.0000000  4.107077e-15 -4.107077e-15
## 68 2011-07-01   3.1399326  3.139933e+00 -1.332268e-15
## 69 2011-08-01   3.7793967  3.779397e+00 -4.440892e-16
## 70 2011-09-01   2.6030321  2.603032e+00 -1.776357e-15
## 71 2011-10-01   0.8099756  8.099756e-01 -3.441691e-15
## 72 2011-11-01   2.2153739  2.215374e+00 -2.220446e-15
## 73 2011-12-01   0.2852163  2.852163e-01 -3.830269e-15
## 74 2012-01-01   3.2645906  3.264591e+00 -8.881784e-16
## 75 2012-02-01   2.5621980  2.562198e+00 -1.776357e-15
## 76 2012-03-01   2.5095386  2.509539e+00 -1.776357e-15
## 77 2012-04-01   1.9912929  1.991293e+00 -2.220446e-15
## 78 2012-05-01  -2.4843861 -2.484386e+00 -6.217249e-15
## 79 2012-06-01  -0.9920458 -9.920458e-01 -4.996004e-15
## 80 2012-07-01   0.3465219  3.465219e-01 -3.774758e-15
## 81 2012-08-01   6.9512613  6.951261e+00  1.776357e-15
## 82 2012-09-01   5.7119038  5.711904e+00  8.881784e-16
## 83 2012-10-01   3.2317977  3.231798e+00 -1.332268e-15
## 84 2012-11-01  -2.0169074 -2.016907e+00 -5.773160e-15
## 85 2012-12-01  -0.1453067 -1.453067e-01 -4.246603e-15
```

```r
mean_ferror <- mean(forecast_errors)
t_test_result <- t.test(forecast_errors)
print(t_test_result)
```

```
##
##  One Sample t-test
```

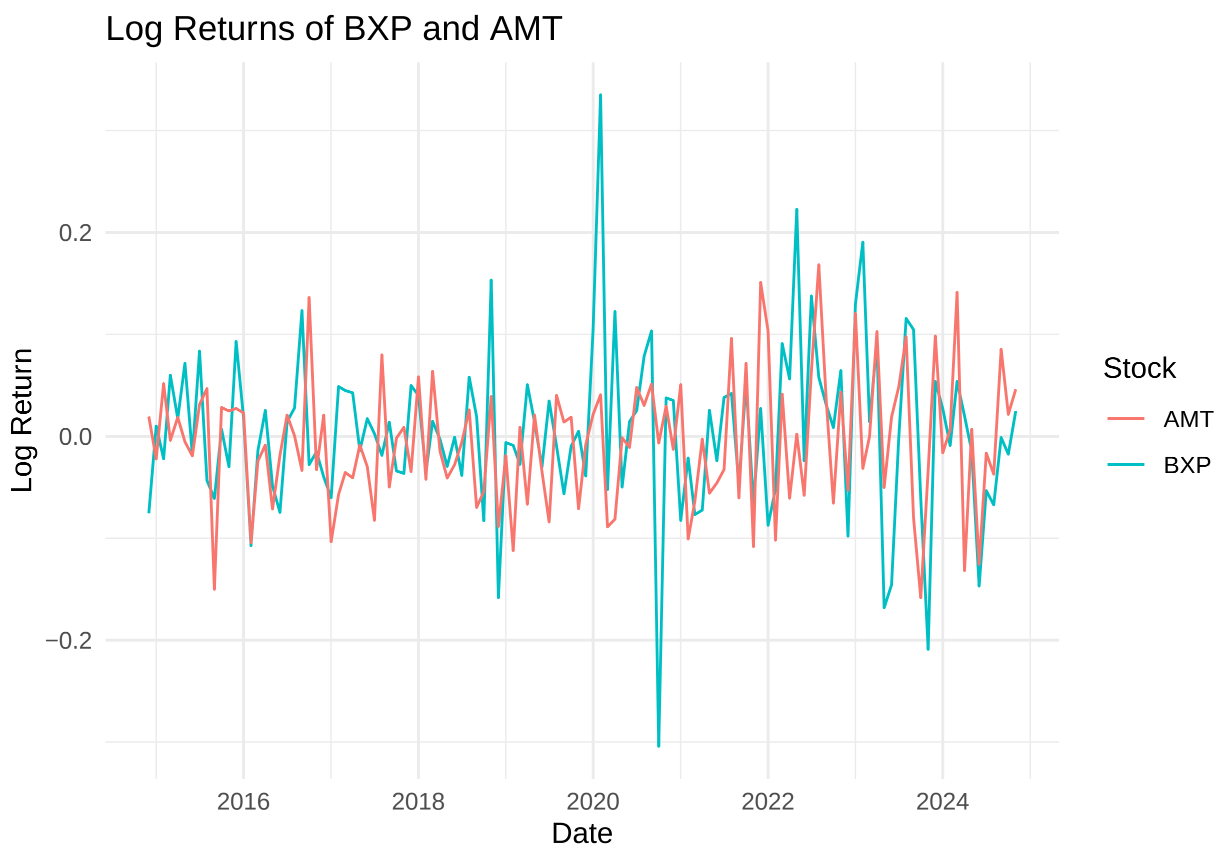
```
##
## data:  forecast_errors
## t = -5.2018, df = 84, p-value = 1.371e-06
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -2.967721e-15 -1.326195e-15
## sample estimates:
##     mean of x
## -2.146958e-15
```

```
RMSFE <- sqrt(mean(forecast_errors^2))
cat("RMSFE:", RMSFE, "\n")
```

```
## RMSFE: 4.349556e-15
```

```
BXP <- read.csv("BXP.csv")
AMT <- read.csv("AMT.csv")
BXP$Date <- as.Date(BXP$Date, format = "%d-%m-%Y")
AMT$Date <- as.Date(AMT$Date, format = "%d-%m-%Y")
BXP$log_price <- log(BXP$Price)
AMT$log_price <- log(AMT$Price)
BXP$log_return <- c(NA, diff(log(BXP$Price)))
AMT$log_return <- c(NA, diff(log(AMT$Price)))
BXP <- na.omit(BXP)
AMT <- na.omit(AMT)
```

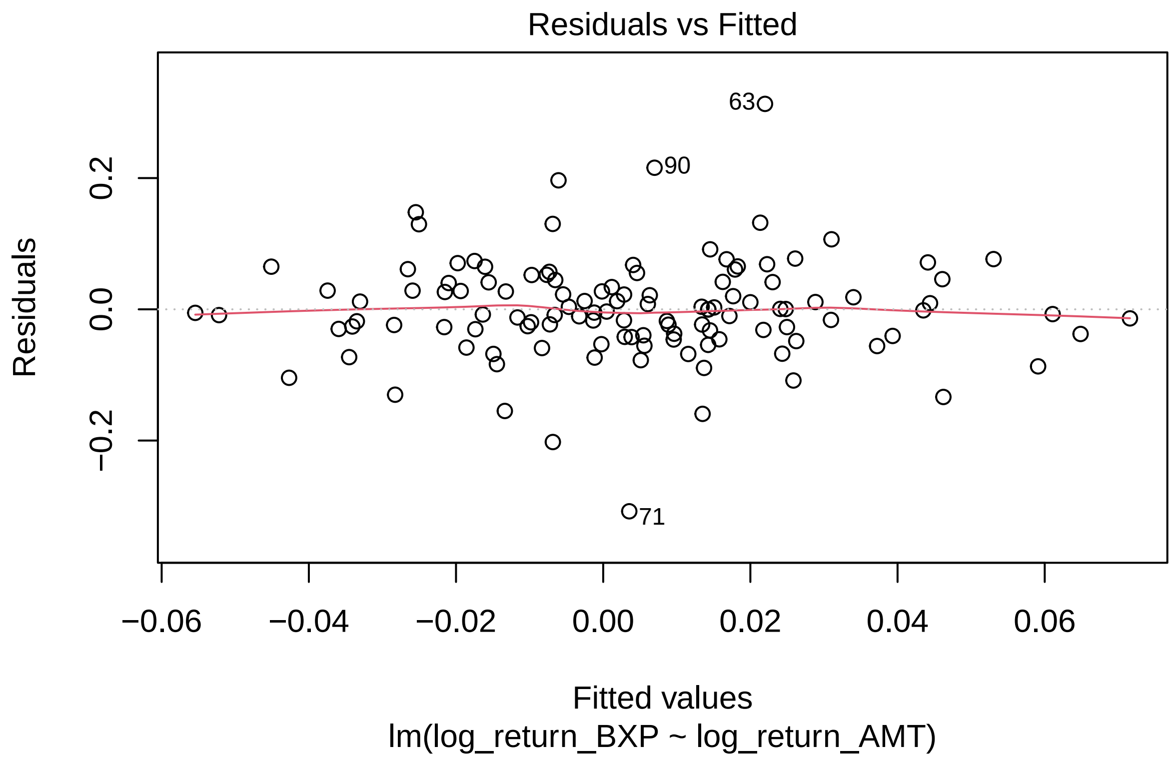
```
merged_data1 <- merge(BXP, AMT, by = "Date", suffixes = c("_BXP", "_AMT"))
ggplot(merged_data1, aes(x = Date)) +
  geom_line(aes(y = log_return_BXP, color = "BXP")) +
  geom_line(aes(y = log_return_AMT, color = "AMT")) +
  labs(title = "Log Returns of BXP and AMT", y = "Log Return", color = "Stock") +
  theme_minimal()
```

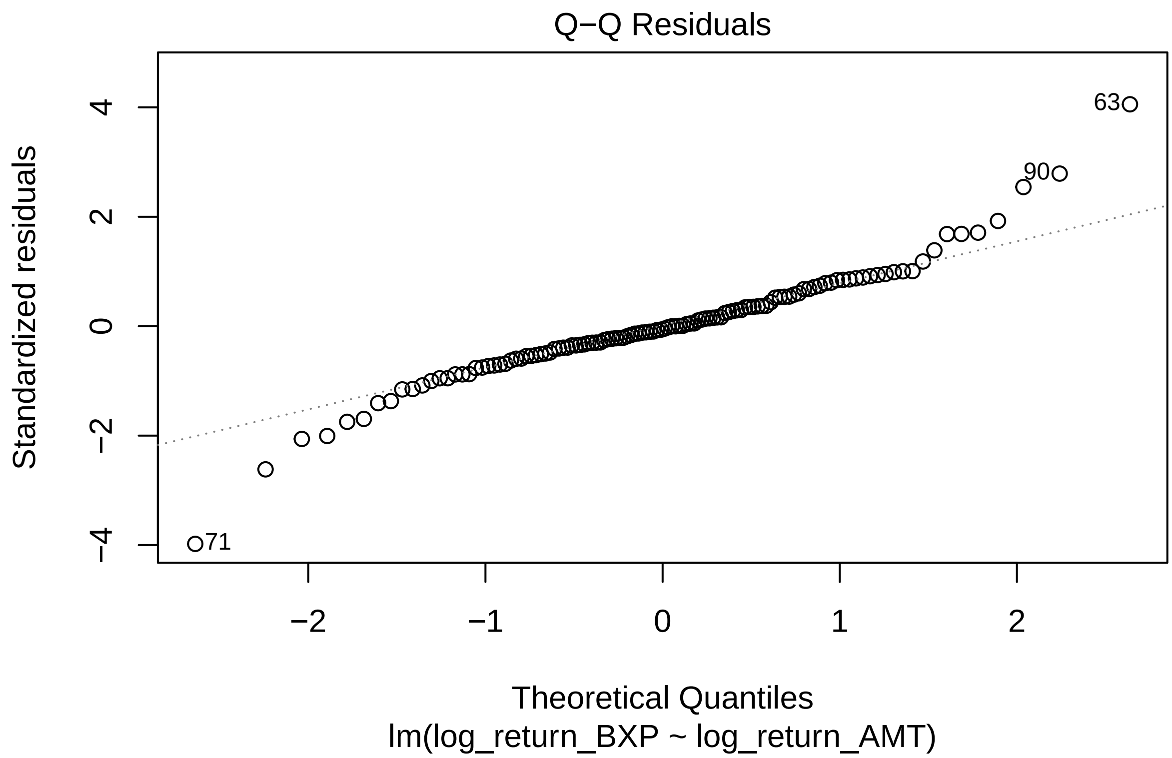


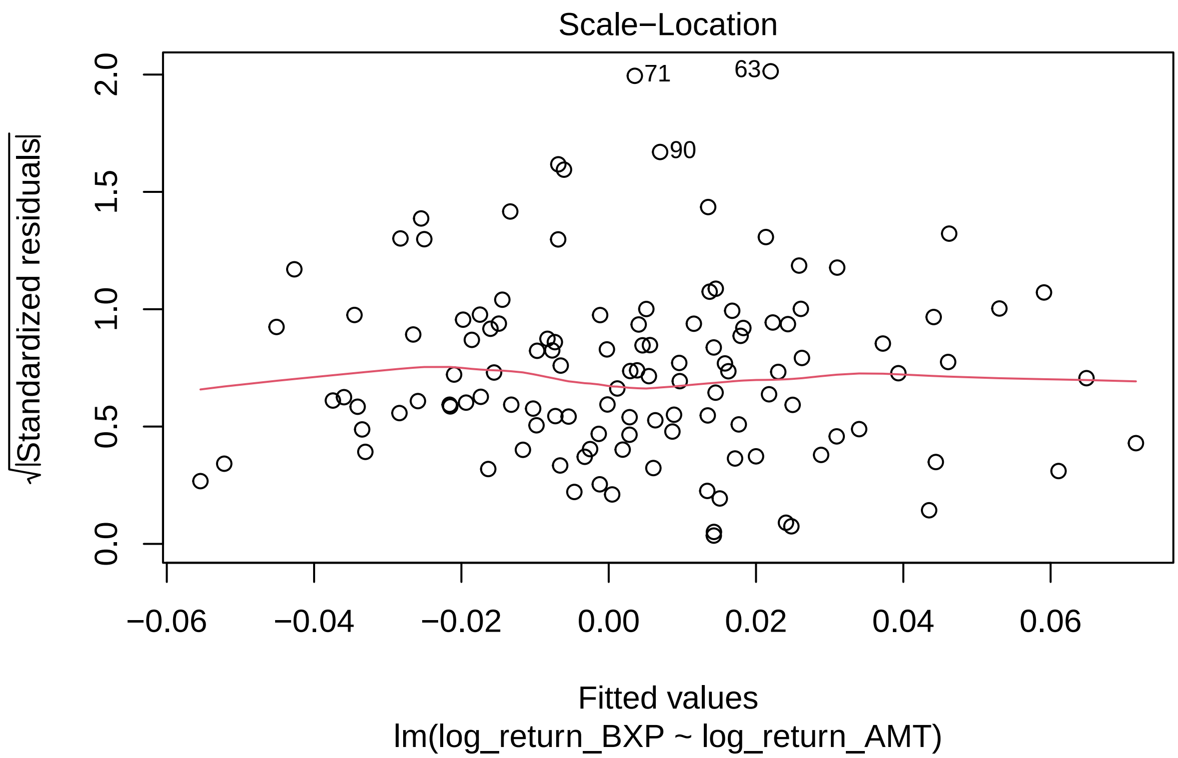

```r
regression_model <- lm(log_return_BXP ~ log_return_AMT, data = merged_data1)
summary(regression_model)
```

```
##
## Call:
## lm(formula = log_return_BXP ~ log_return_AMT, data = merged_data1)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.307798 -0.038072 -0.004206  0.041226  0.313073
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.006158   0.007122   0.865 0.388973
## log_return_AMT 0.388766   0.111517   3.486 0.000689 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07769 on 118 degrees of freedom
## Multiple R-squared:  0.09338,    Adjusted R-squared:  0.08569
## F-statistic: 12.15 on 1 and 118 DF,  p-value: 0.0006889
```

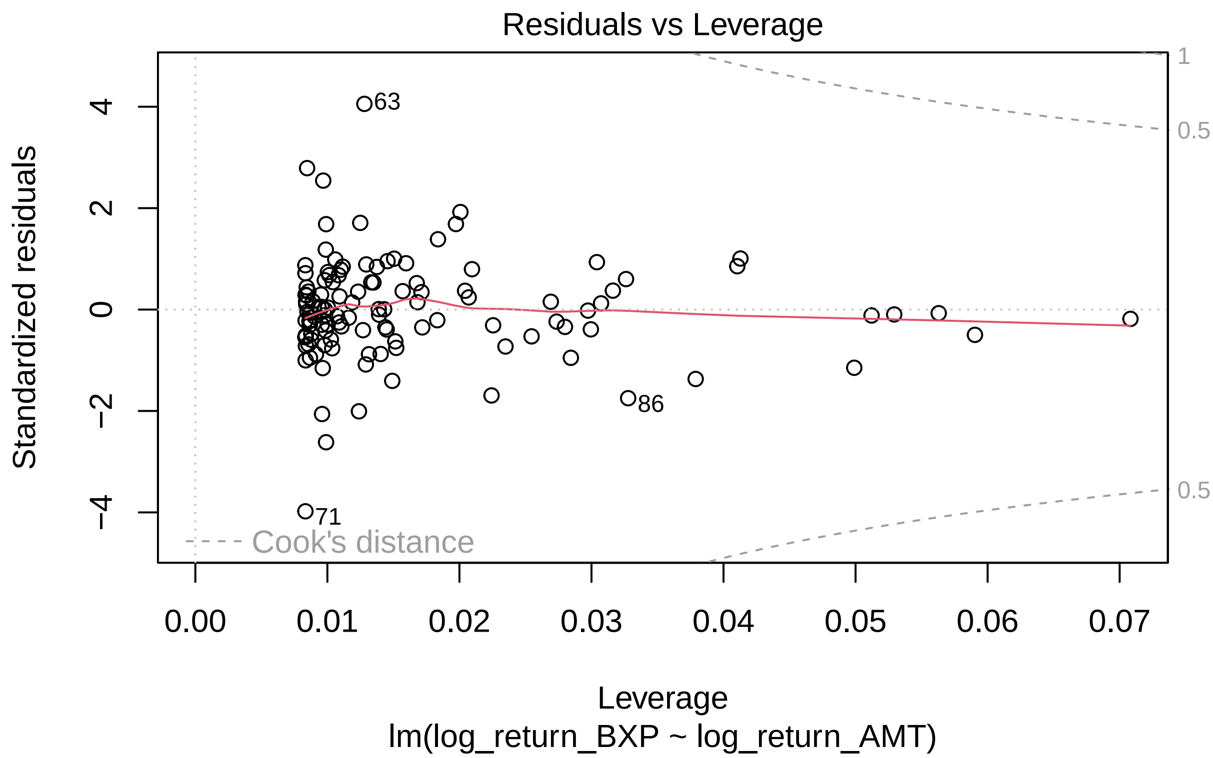
```r
merged_data1$residuals <- regression_model$residuals
plot(regression_model)
```


Residuals vs Fitted
Residuals
0.2
0.0
−0.2
63
90
71
−0.06
−0.04
−0.02
0.00
0.02
0.04
0.06
Fitted values
lm(log_return_BXP ~ log_return_AMT)


Q−Q Residuals
Standardized residuals
4
2
0
−2
−4
63
90
71
−2
−1
0
1
2
Theoretical Quantiles
lm(log_return_BXP ~ log_return_AMT)



Scale−Location
√|Standardized residuals|
2.0
1.5
1.0
0.5
0.0
71
63
90
−0.06
−0.04
−0.02
0.00
0.02
0.04
0.06
Fitted values
lm(log_return_BXP ~ log_return_AMT)




```r
adf_test_residuals <- ur.df(merged_data1$residuals, type = "none", selectlags = "AIC")
summary(adf_test_residuals)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression none
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.307038 -0.037459 -0.004018  0.041331  0.313129
##
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## z.lag.1    -1.01141    0.13133  -7.701 4.95e-12 ***
## z.diff.lag  0.00739    0.09238   0.080    0.936
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.07791 on 116 degrees of freedom
## Multiple R-squared:  0.5021, Adjusted R-squared:  0.4935
## F-statistic: 58.48 on 2 and 116 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -7.7013
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau1 -2.58 -1.95 -1.62
```

```
ecm_model <- lm(log_return_BXP ~
                lag(log_return_BXP, 1) +   # Positive lag for spot return
                lag(log_return_AMT, 1) +   # Positive lag for futures return
                (log_return_BXP - log_return_AMT) +  # Cointegration term
                residuals,
                data = merged_data1)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): the response appeared on
## the right-hand side and was dropped

## Warning in model.matrix.default(mt, mf, contrasts): problem with term 3 in
## model.matrix: no columns are assigned
```

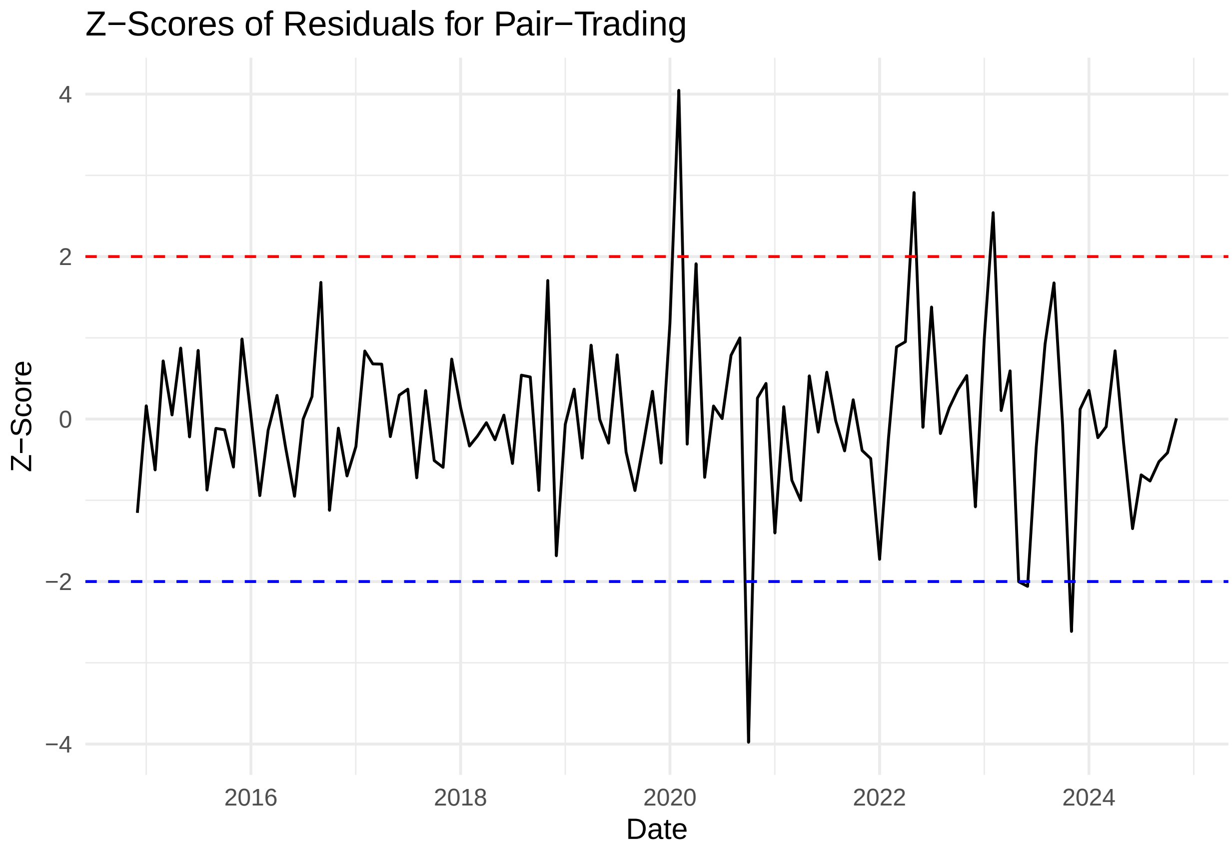
```
summary(ecm_model)
```

```
## Warning in summary.lm(ecm_model): essentially perfect fit: summary may be
## unreliable

##
## Call:
## lm(formula = log_return_BXP ~ lag(log_return_BXP, 1) + lag(log_return_AMT,
##     1) + (log_return_BXP - log_return_AMT) + residuals, data = merged_data1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -4.941e-16  1.940e-18  4.220e-18  6.210e-18  4.758e-17
##
## Coefficients: (1 not defined because of singularities)
##                         Estimate Std. Error   t value Pr(>|t|)
## (Intercept)            1.900e-18  4.270e-18 4.450e-01    0.657
## lag(log_return_BXP, 1) 1.000e+00  5.501e-17 1.818e+16   <2e-16 ***
## lag(log_return_AMT, 1) 2.302e-17  6.999e-17 3.290e-01    0.743
## residuals                     NA         NA        NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.643e-17 on 117 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.822e+32 on 2 and 117 DF,  p-value: < 2.2e-16
```

```
merged_data1$z_score <- (merged_data1$residuals - mean(merged_data1$residuals)) / sd(merged_data1$residu
```

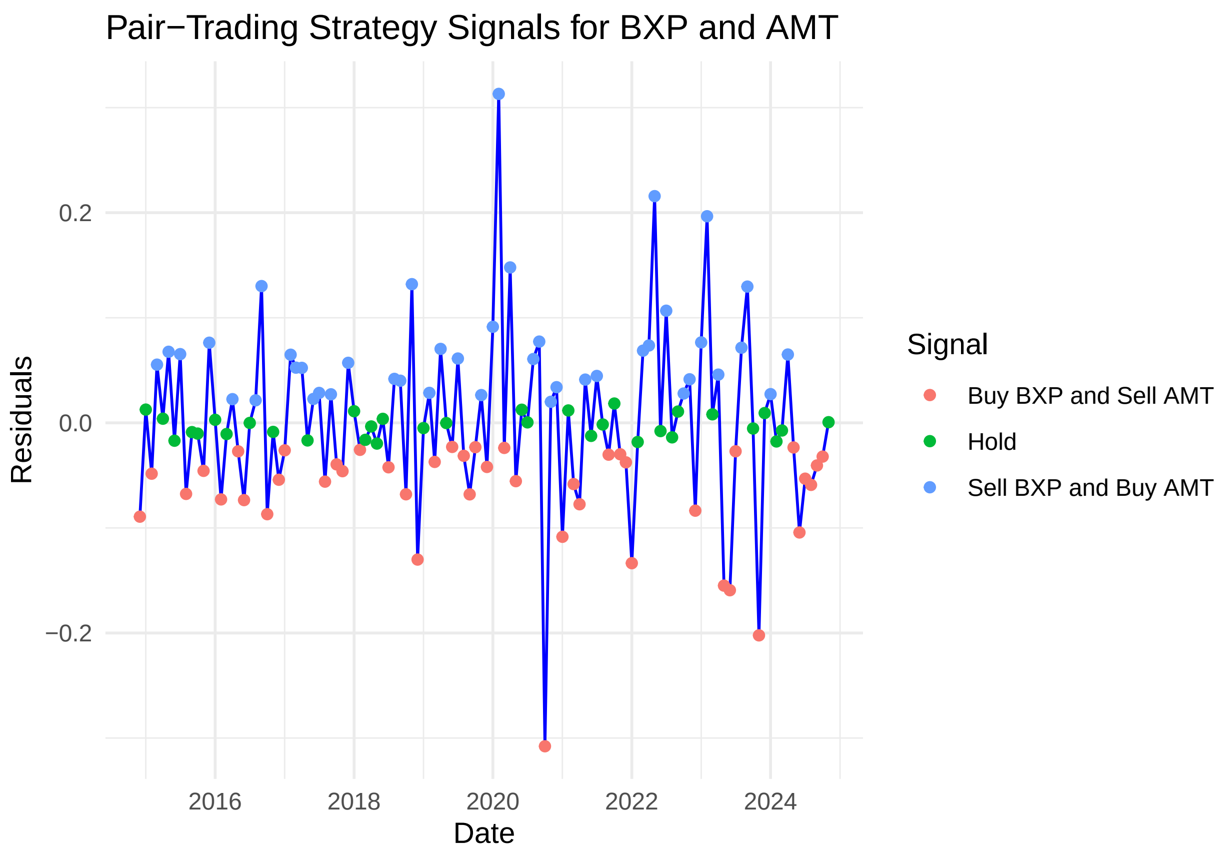
```
ggplot(merged_data1, aes(x = Date, y = z_score)) +
  geom_line() +
  geom_hline(yintercept = 2, linetype = "dashed", color = "red") +
  geom_hline(yintercept = -2, linetype = "dashed", color = "blue") +
  labs(title = "Z-Scores of Residuals for Pair-Trading", y = "Z-Score") +
  theme_minimal()
```




```
threshold <- 0.02

merged_data1 <- merged_data1 %>%
  mutate(
    signal = case_when(
      residuals > threshold ~ "Sell BXP and Buy AMT",
      residuals < -threshold ~ "Buy BXP and Sell AMT",
      TRUE ~ "Hold"
    )
  )
ggplot(merged_data1, aes(x = Date)) +
  geom_line(aes(y = residuals), color = "blue") +
  geom_point(aes(y = residuals, color = signal)) +
  labs(title = "Pair-Trading Strategy Signals for BXP and AMT",
       y = "Residuals",
```

```
      color = "Signal") +
  theme_minimal()
```




```
merged_data1 <- merged_data1 %>%
  mutate(
    d_log_return_BXP = c(NA, diff(log_return_BXP)),
    d_log_return_AMT = c(NA, diff(log_return_AMT)),
    cointegration_term = log_return_BXP - log_return_AMT
  ) %>%
  na.omit()
vecm_model_1 <- lm(
  d_log_return_BXP ~
    lag(d_log_return_BXP, 1) +
    lag(d_log_return_AMT, 1) +
    cointegration_term,
  data = merged_data1
)
vecm_model_2 <- lm(
  d_log_return_AMT ~
    lag(d_log_return_BXP, 1) +
    lag(d_log_return_AMT, 1) +
    cointegration_term,
  data = merged_data1
)
summary(vecm_model_1)
```

```
## Warning in summary.lm(vecm_model_1): essentially perfect fit: summary may be
## unreliable
```

```
##
## Call:
## lm(formula = d_log_return_BXP ~ lag(d_log_return_BXP, 1) + lag(d_log_return_AMT,
##     1) + cointegration_term, data = merged_data1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -6.043e-16 -1.760e-18  5.130e-18  1.092e-17  6.189e-17
##
## Coefficients:
##                            Estimate Std. Error    t value Pr(>|t|)
## (Intercept)              -6.361e-19  5.369e-18 -1.180e-01    0.906
## lag(d_log_return_BXP, 1)  1.000e+00  6.415e-17  1.559e+16   <2e-16 ***
## lag(d_log_return_AMT, 1) -4.930e-17  7.428e-17 -6.640e-01    0.508
## cointegration_term        2.867e-17  8.802e-17  3.260e-01    0.745
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.772e-17 on 115 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.61e+32 on 3 and 115 DF,  p-value: < 2.2e-16
```

```r
summary(vecm_model_2)
```

```
## Warning in summary.lm(vecm_model_2): essentially perfect fit: summary may be
## unreliable
```

```
##
## Call:
## lm(formula = d_log_return_AMT ~ lag(d_log_return_BXP, 1) + lag(d_log_return_AMT,
##     1) + cointegration_term, data = merged_data1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -6.492e-16  7.000e-19  5.800e-18  9.600e-18  7.479e-17
##
## Coefficients:
##                            Estimate Std. Error    t value Pr(>|t|)
## (Intercept)              -2.500e-19  5.735e-18 -4.400e-02   0.9653
## lag(d_log_return_BXP, 1) -1.750e-16  6.852e-17 -2.554e+00   0.0119 *
## lag(d_log_return_AMT, 1)  1.000e+00  7.934e-17  1.260e+16   <2e-16 ***
## cointegration_term        2.182e-17  9.402e-17  2.320e-01   0.8169
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.166e-17 on 115 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 9.864e+31 on 3 and 115 DF,  p-value: < 2.2e-16
```