Mini Research

Contents

Libraries

```
library(readr)
library(dplyr)
```

```
##
## 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
```

```
library(stargazer)
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```
library(forecast)
```

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

```
data <- read_csv("data/for_regressing/interbank_nii.csv")
```

```
## Rows: 65 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (2): IR, median
## date (1): Date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
data_clean <- na.omit(data)
```

```
model <- lm(median ~ IR, data = data_clean)

stargazer(model, type = "text", align = TRUE, single.row = TRUE)
```

```
##
```

```
## ================================================
##                          Dependent variable:
##                      ---------------------------
##                                 median
## ------------------------------------------------
## IR                          0.0002*** (0.00003)
## Constant                     0.012*** (0.0005)
## ------------------------------------------------
## Observations                        62
## R2                                 0.290
## Adjusted R2                        0.278
## Residual Std. Error          0.002 (df = 60)
## F Statistic             24.537*** (df = 1; 60)
## ================================================
## Note:                *p<0.1; **p<0.05; ***p<0.01
```

```
cat(paste("AIC: ", AIC(model)))
```

```
## AIC:  -598.28833390611
```

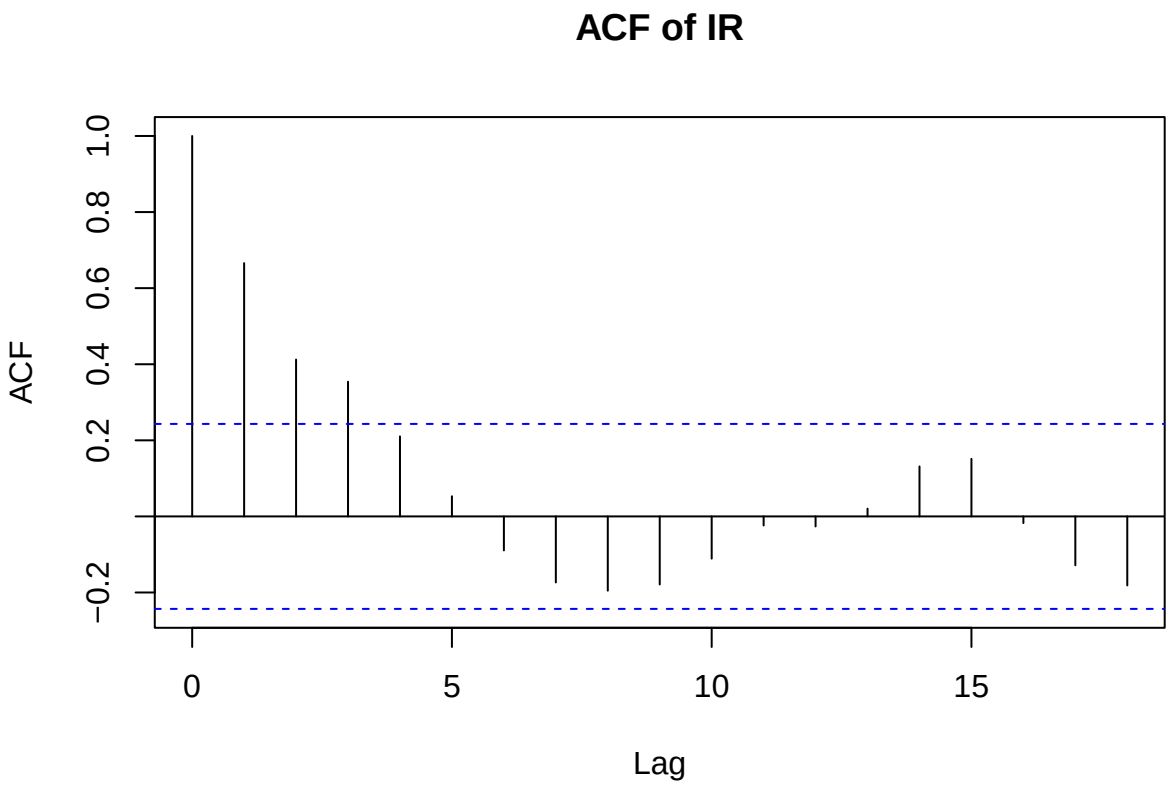
```
data_clean <- data_clean %>%
  mutate(IR_lag1 = lag(IR, 1),
         IR_lag2 = lag(IR, 2),
         IR_lag3 = lag(IR, 3))

model <- lm(median ~ IR + IR_lag1 + IR_lag2 + IR_lag3, data = data_clean)

stargazer(model, type = "text", align = TRUE, single.row = TRUE)
```

```
##
## ================================================
##                          Dependent variable:
##                      ---------------------------
##                                 median
## ------------------------------------------------
## IR                           0.0001 (0.0001)
## IR_lag1                     0.00005 (0.0001)
## IR_lag2                      0.0001 (0.0001)
## IR_lag3                    0.0001** (0.00004)
## Constant                     0.010*** (0.0005)
## ------------------------------------------------
## Observations                        59
## R2                                 0.546
## Adjusted R2                        0.513
## Residual Std. Error          0.002 (df = 54)
## F Statistic             16.266*** (df = 4; 54)
## ================================================
## Note:                *p<0.1; **p<0.05; ***p<0.01
```

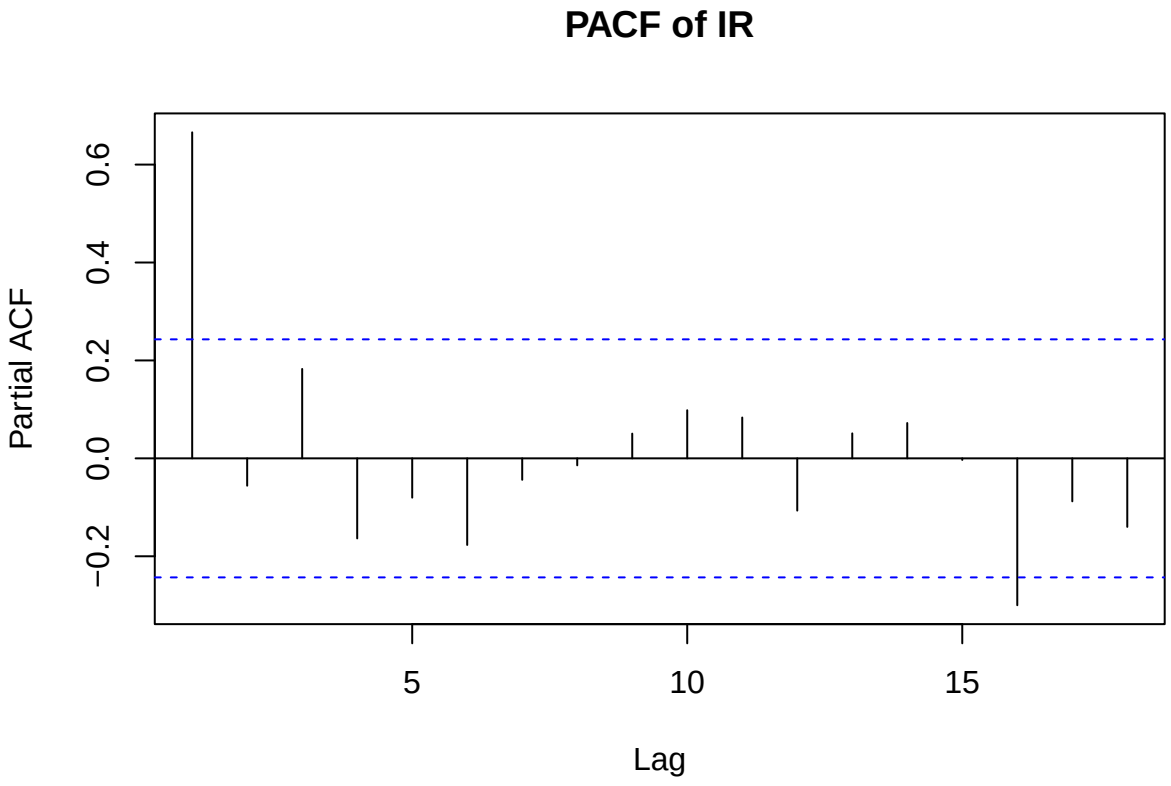
```
cat(paste("AIC: ", AIC(model)))
```

```
## AIC:  -587.029235492865
```

```r
acf(data$IR, main="ACF of IR")
```




```r
pacf(data$IR, main="PACF of IR")
```



```r
# data$Date <- ymd(data$Date)
#
# basic_model <- lm(median ~ IR, data = data)
# summary(basic_model)
#
# create_lags <- function(data, var, max_lag) {
#   for (i in 1:max_lag) {
#     data[paste0(var, "_lag", i)] <- lag(data[[var]], i)
#   }
#   return(data)
# }
#
# data_with_lags <- create_lags(data, "IR", 24)
#
# test_lag_combinations <- function(data, max_lag) {
#   best_model <- NULL
#   best_aic <- Inf
#
#   for (i in 1:max_lag) {
#     for (combination in combn(1:max_lag, i, simplify = FALSE)) {
#       formula <- as.formula(paste("median ~", paste(paste0("IR_lag", combination), collapse = " + ")))
#       model <- lm(formula, data = data)
#       current_aic <- AIC(model)
#
#       if (current_aic < best_aic) {
#         best_model <- model
#         best_aic <- current_aic
#       }
#     }
#   }
#
#   return(best_model)
# }
#
# best_model <- test_lag_combinations(data_with_lags, 24)
#
# summary(best_model)
# print(paste("Best AIC:", AIC(best_model)))
```