Preliminary examination of the Welch-Goyal data set

Bushra Tariq Kiyani

2023-10-19

```
#Import necessary libraries
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(tidyr)
library(readr)
library(stats)
library(ggplot2)
```

#c) Import the monthly series as data frame in R and make sure to name the variables meaningfully

```
#Import the csv file
stock_data <- read.csv("PredictorData2022.xlsx - Monthly.csv")
```

#d) Generate excess returns series (stock returns minus risk-free rate)

```
#Calculate stock return
stock_data$Index <- as.numeric(stock_data$Index)
```

```
## Warning: NAs introduced by coercion
```

```
stock_data$D12 <- as.numeric(stock_data$D12)
stock_data$stock_return <- stock_data$Index + stock_data$D12

#Calculate excess return
stock_data$excess_return<-stock_data$stock_return - stock_data$Rfree
```

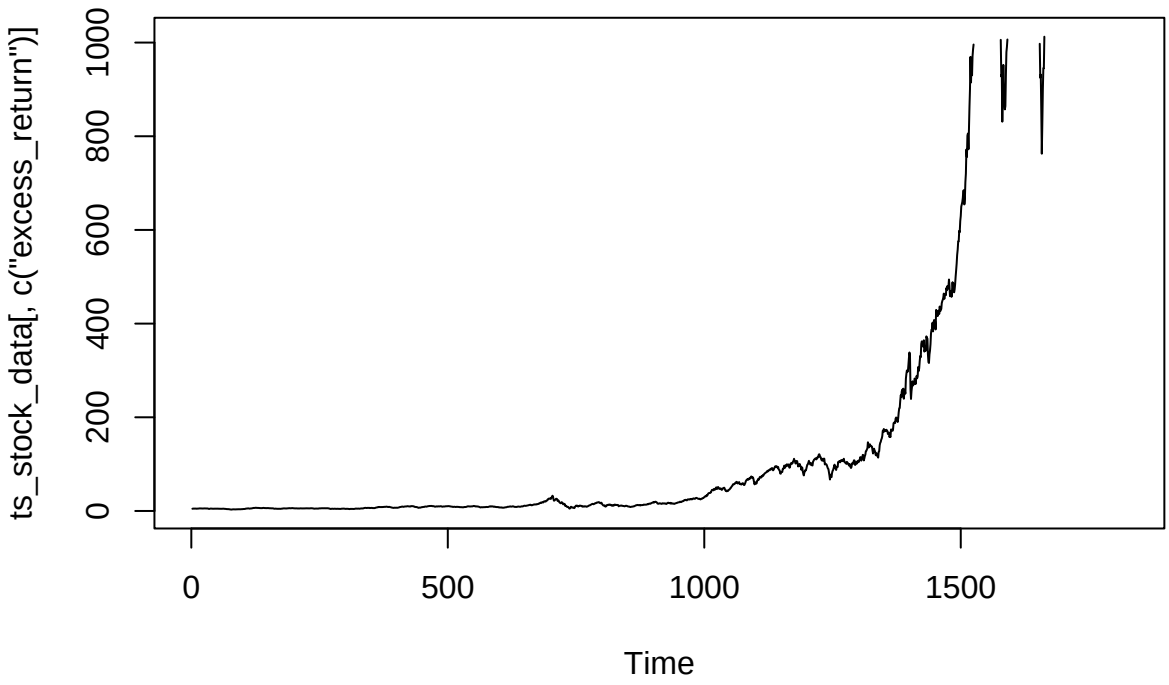
#e) Generate time series plots of excess returns and each predictor from the data frame. Lag the predictor!

```
#Defining the predictors which needs to be calculated first
#.....................................................................
##  Dividend-price ratio (dpr)
##  .....................................................................
##  difference between the log of dividends and the log of prices
dpr<- log(stock_data$D12) - log(stock_data$Index)
stock_data$dividend_price_ratio<-dpr
```

```r
#.....................................................................
##  Earning-price ratio (epr)
##  ...................................................................
##  difference between the log of Earning and the log of prices
epr<- log(stock_data$E12) - log(stock_data$Index)
stock_data$earning_price_ratio<-epr
```

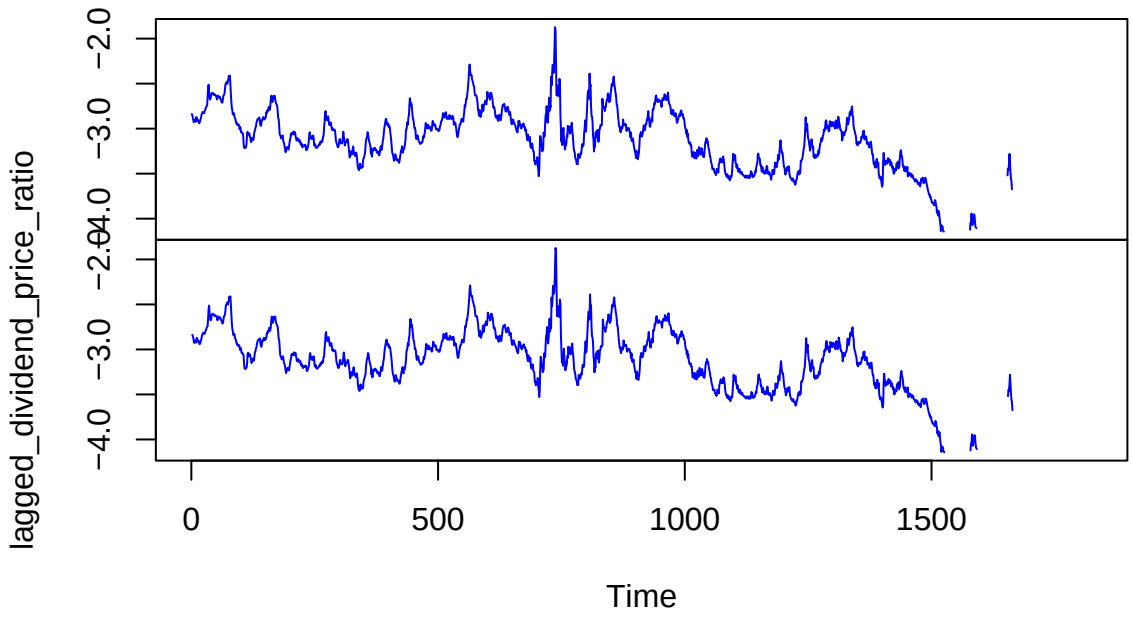
```r
#.....................................................................
##  Dividend-Earning ratio (der)
##  ...................................................................
##  difference between the log of Dividend and the log of Earning
der<- log(stock_data$D12) - log(stock_data$E12)
stock_data$dividend_earning_ratio<-der
```

```r
#.....................................................................
##  Default Return Spread (dfr)
##  ...................................................................
##  difference between long-term corporate bond and long-term government bonds

dfr<- stock_data$corpr - stock_data$ltr
stock_data$default_return_spread<-dfr

#.....................................................................
##  Default yield spread (dfy)
##  ...................................................................
##  difference between BAA and AAA-rated corporate bond yields

dfy<- stock_data$BAA - stock_data$AAA
stock_data$default_yield_spread<-dfy
##  ...................................................................

#.....................................................................
##  Term spread (tms)                                              ####
##  ...................................................................
##  difference between  long term yield on government bonds (lty) and the treasury-bill (tbl)

tms<- stock_data$lty - stock_data$tbl
stock_data$term_spread<-tms

##  ...................................................................

#Create Time Series object

ts_stock_data <- ts(stock_data)#, start=stock_data[1, yyyymm], end=stock_data[nrow(stock_data), yyyymm]

##  ...................................................................
##  Plot ts_stock_data  for "Excess Return"                                  ####
##  ...................................................................

plot(ts_stock_data[, c("excess_return")])
```

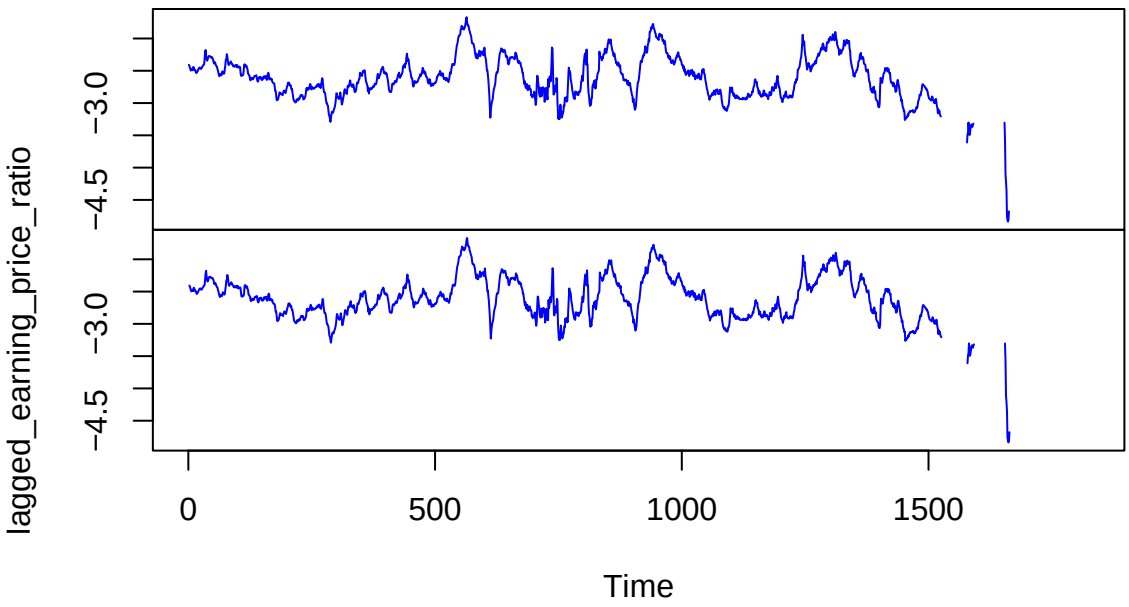



```
#============================================================
#Time series for lagged predictors
#dividend_price_ratio,earning_price_ratio,dividend_earning_ratio,
#default_return_spread, default_yield_spread, term_spread, Index, D12, E12, AAA,
#BAA, Rfree, ltr, lty, corp, tbl
#============================================================
```

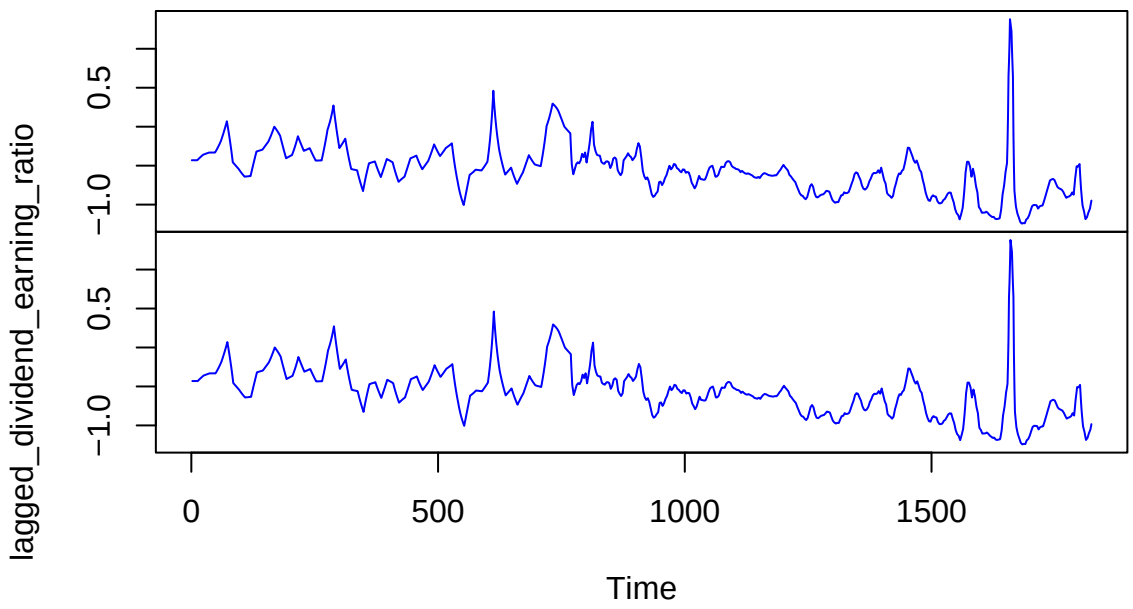
```
#===============
#For divided price ratio
#===============
# Create a lag of dividend_price_ratio
lagged_dividend_price_ratio <- lag(stock_data$dividend_price_ratio)

# Create a new time series object with the lag
#ts_lagged <- ts(cbind(lagged_dividend_price_ratio))
ts_lagged <- ts(cbind(stock_data$dividend_price_ratio, lagged_dividend_price_ratio))

# Plot the original and lagged dividend_price_ratio
plot(ts_lagged, main = "Original and Lagged Dividend-Price Ratio", col = c("blue", "red"), lty = c(1, 2)
```



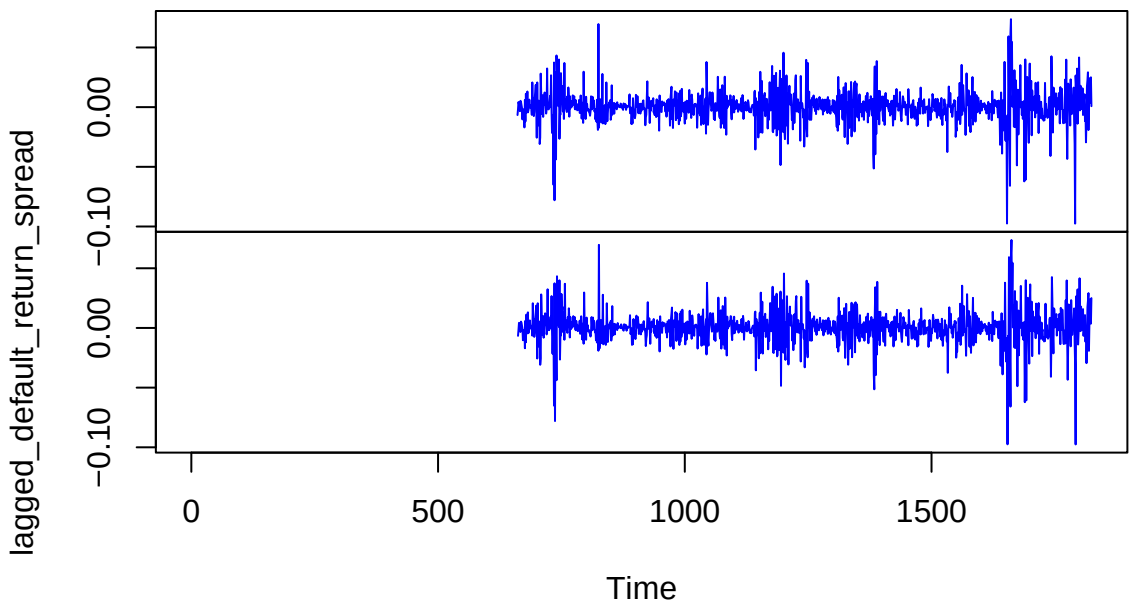

```
#===============
#For earning_price_ratio
#===============
# Create a lag of dividend_price_ratio
lagged_earning_price_ratio <- lag(stock_data$earning_price_ratio)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$earning_price_ratio,lagged_earning_price_ratio))
#ts_lagged <- ts(cbind(lagged_earning_price_ratio))

# Plot the original and lagged earning_price_ratio
plot(ts_lagged, main = "Original and Lagged earning_price_ratio", col = c("blue", "red"), lty = c(1, 2))
```



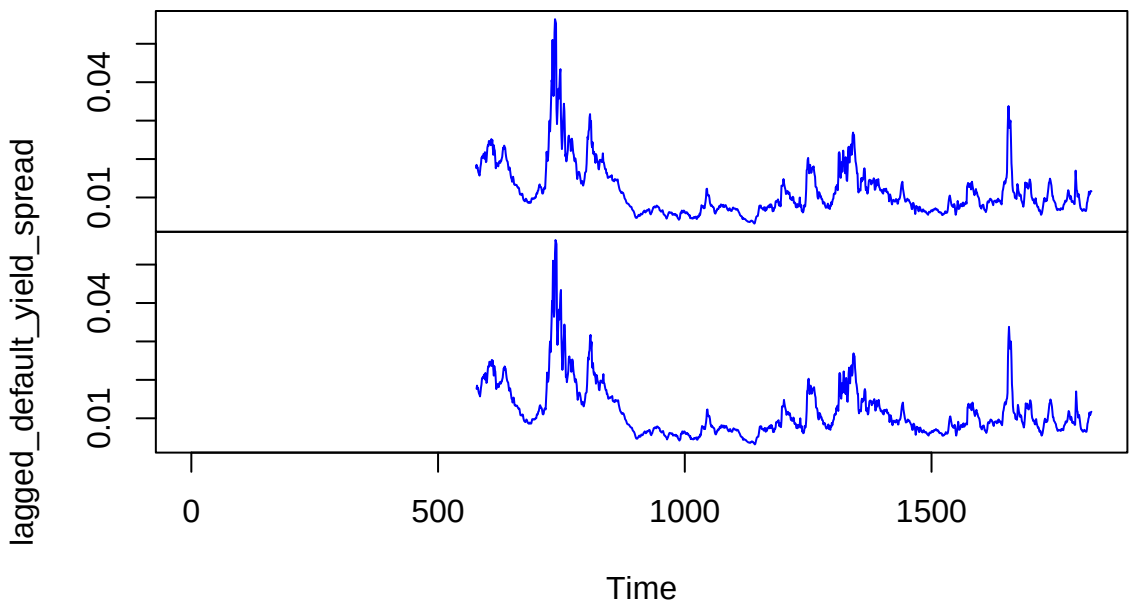

```
#================
#For dividend_earning_ratio
#================
# Create a lag of dividend_price_ratio
lagged_dividend_earning_ratio <- lag(stock_data$dividend_earning_ratio)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$dividend_earning_ratio,lagged_dividend_earning_ratio))
#ts_lagged <- ts(cbind(lagged_dividend_earning_ratio))

# Plot the Original and lagged dividend_earning_ratio
plot(ts_lagged, main = "Original and Lagged dividend_earning_ratio", col = c("blue", "red"), lty = c(1,
```



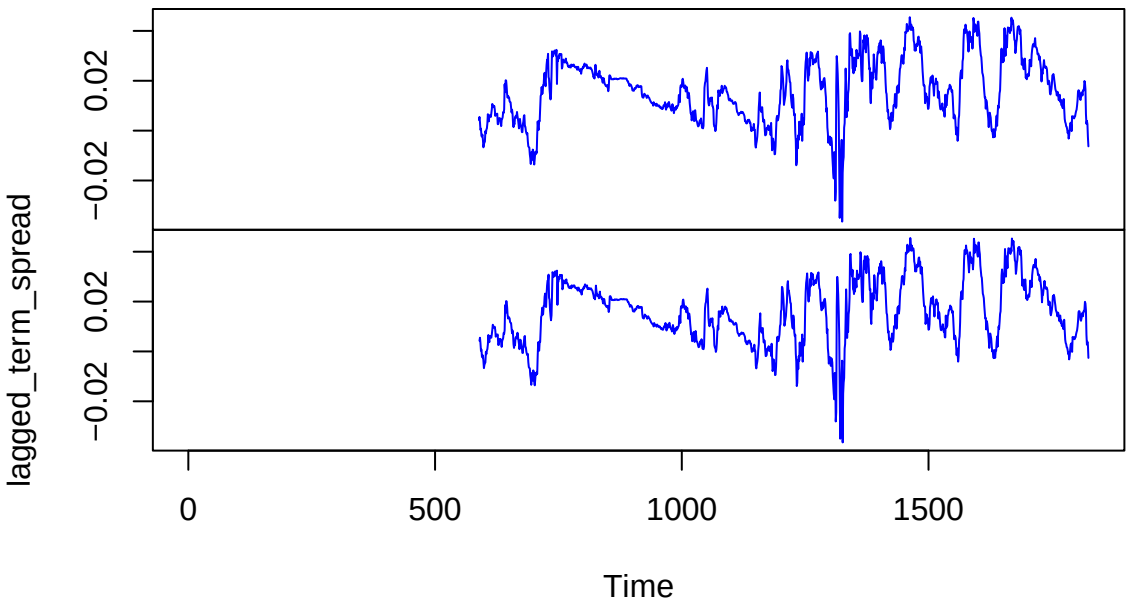

```
#================
#For default_return_spread
#================
# Create a lag of default_return_spread
lagged_default_return_spread <- lag(stock_data$default_return_spread)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$default_return_spread,lagged_default_return_spread))
#ts_lagged <- ts(cbind(lagged_default_return_spread))

# Plot the original and lagged default_return_spread
plot(ts_lagged, main = "original and Lagged default_return_spread", col = c("blue", "red"), lty = c(1, 1
```



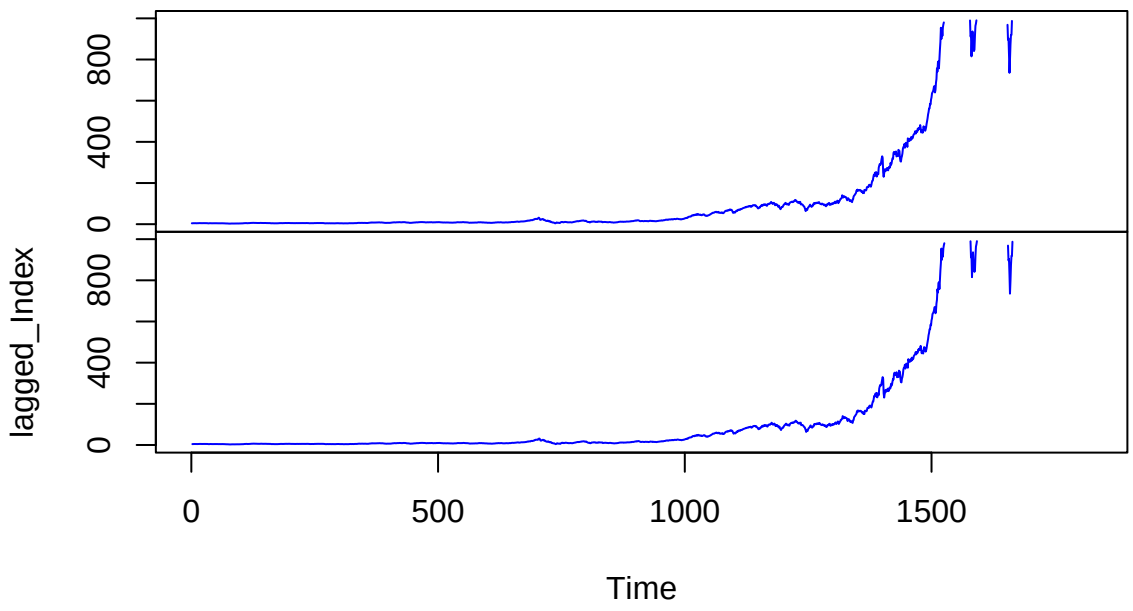

```
#================
#For default_yield_spread
#================
# Create a lag of default_yield_spread
lagged_default_yield_spread <- lag(stock_data$default_yield_spread)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$default_yield_spread,lagged_default_yield_spread))
#ts_lagged <- ts(cbind(lagged_default_yield_spread))

# Plot the original and lagged default_yield_spread
plot(ts_lagged, main = "Original and Lagged default_yield_spread", col = c("blue", "red"), lty = c(1, 2)
```



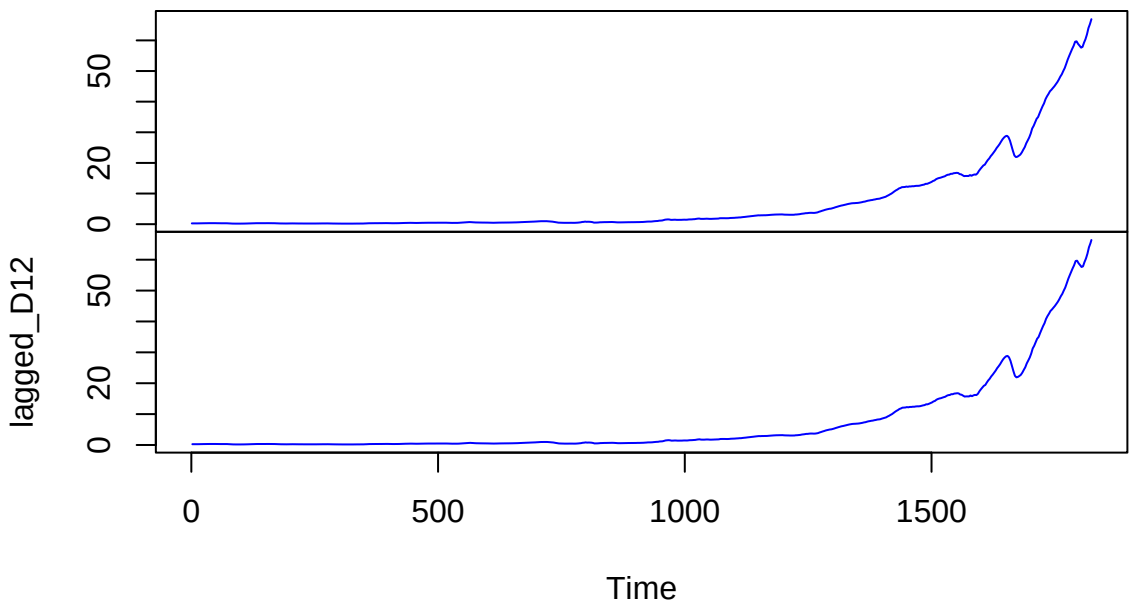

```
#================
#For term_spread
#================
# Create a lag of term_spread
lagged_term_spread <- lag(stock_data$term_spread)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$term_spread,lagged_term_spread))
#ts_lagged <- ts(cbind(lagged_term_spread))

# Plot the original and lagged term_spread
plot(ts_lagged, main = "Original and Lagged term_spread", col = c("blue", "red"), lty = c(1, 2))
```



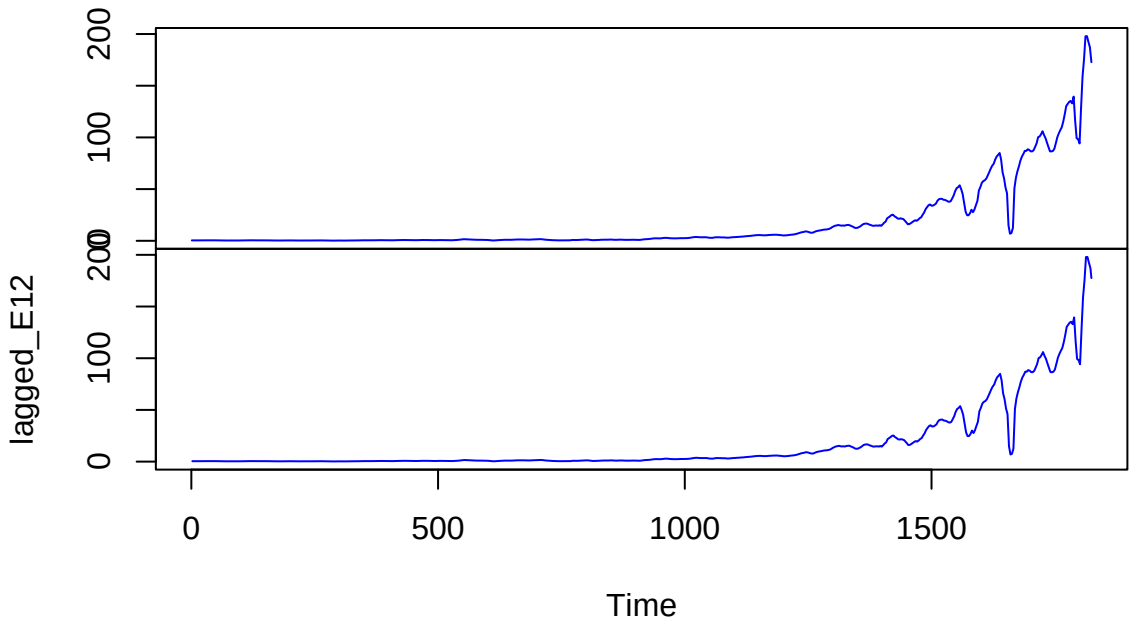

```
#===============
#For lagged_Index
#===============
# Create a lag of Index
lagged_Index <- lag(stock_data$Index)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$Index,lagged_Index))
#ts_lagged <- ts(cbind(lagged_Index))

# Plot the original and lagged Index
plot(ts_lagged, main = "original and Lagged Index", col = c("blue", "red"), lty = c(1, 2))
```



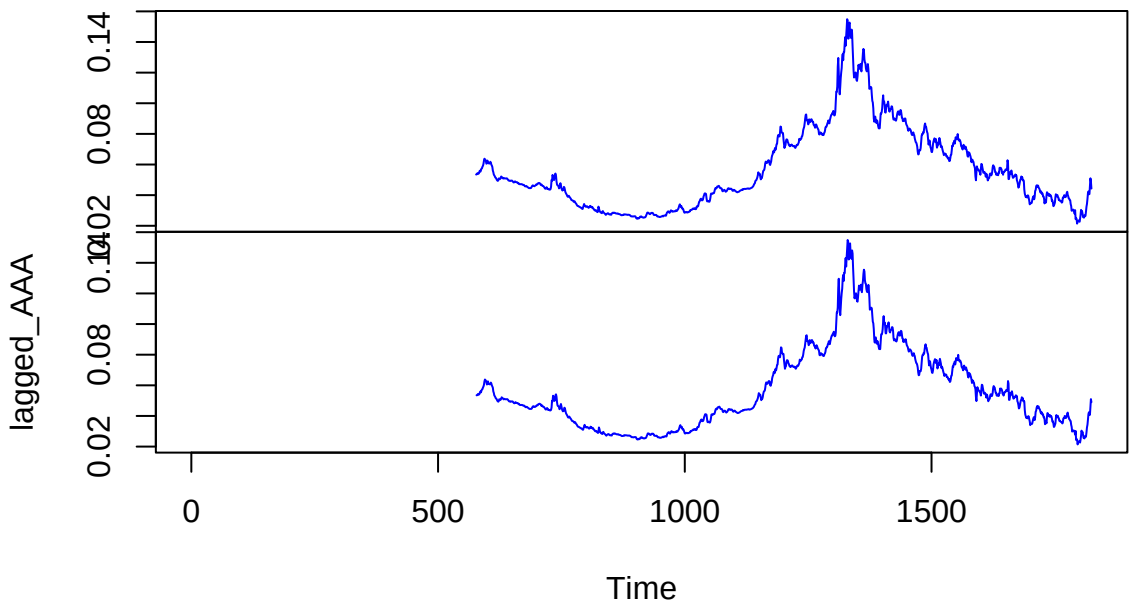

```
#================
#For D12
#================
# Create a lag of Index
lagged_D12 <- lag(stock_data$D12)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$D12,lagged_D12))
#ts_lagged <- ts(cbind(lagged_D12))

# Plot the original and lagged D12
plot(ts_lagged, main = "original and Lagged D12", col = c("blue", "red"), lty = c(1, 2))
```



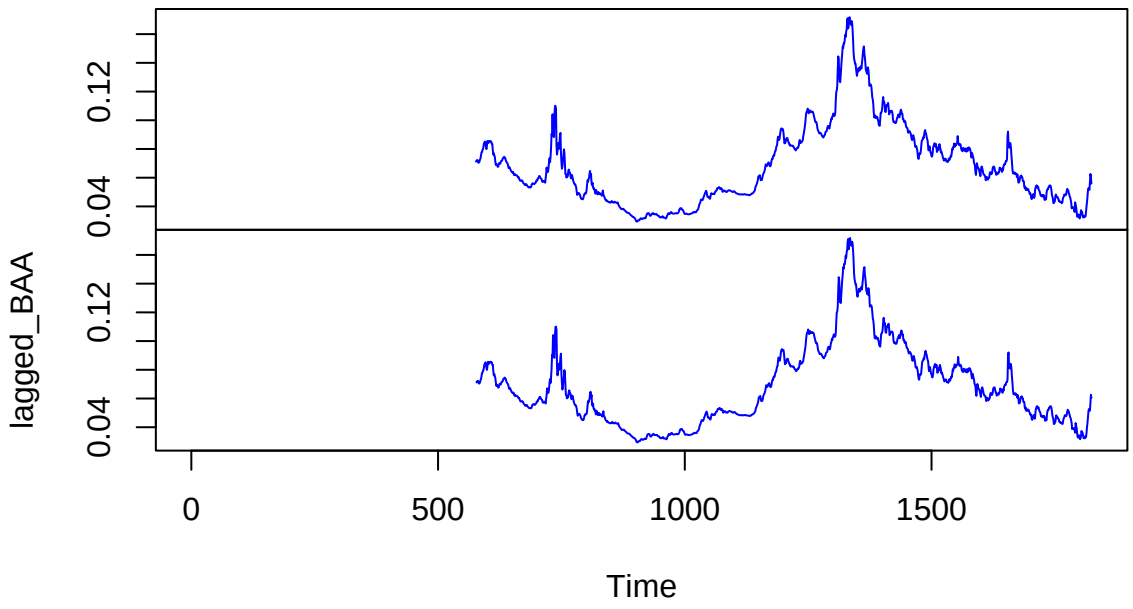

```r
#===============
#For E12
#===============
# Create a lag of E12
lagged_E12 <- lag(stock_data$E12)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$E12,lagged_E12))
#ts_lagged <- ts(cbind(lagged_E12))

# Plot the original and lagged E12
plot(ts_lagged, main = "original and Lagged E12", col = c("blue", "red"), lty = c(1, 2))
```



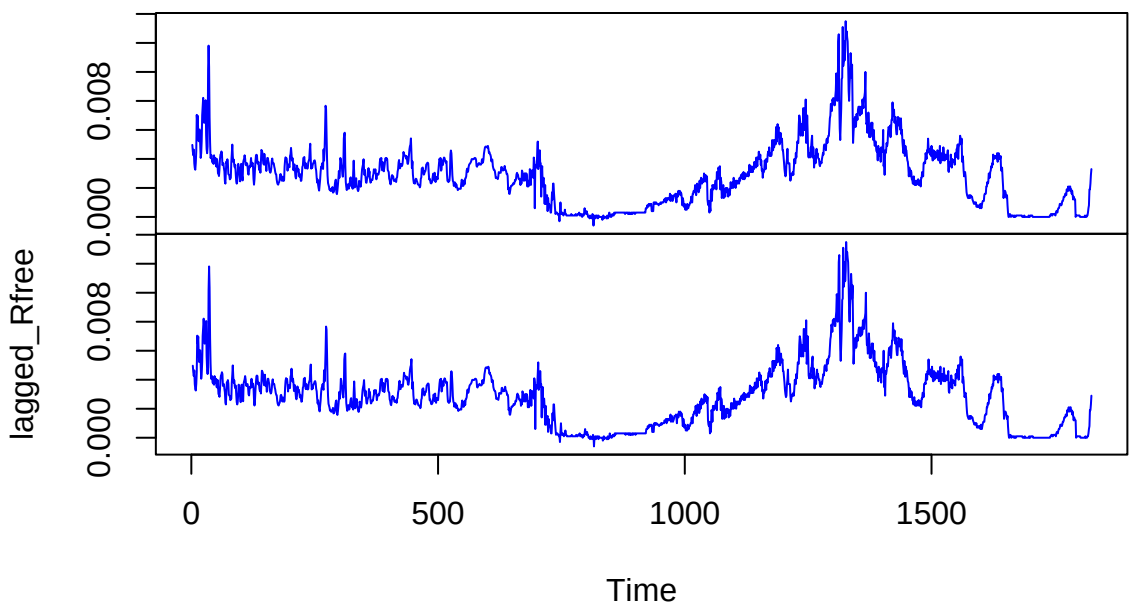

```
#===============
#For AAA
#===============
# Create a lag of AAA
lagged_AAA <- lag(stock_data$AAA)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$AAA,lagged_AAA))
#ts_lagged <- ts(cbind(lagged_AAA))

# Plot the original and lagged AAA
plot(ts_lagged, main = "original and Lagged AAA", col = c("blue", "red"), lty = c(1, 2))
```



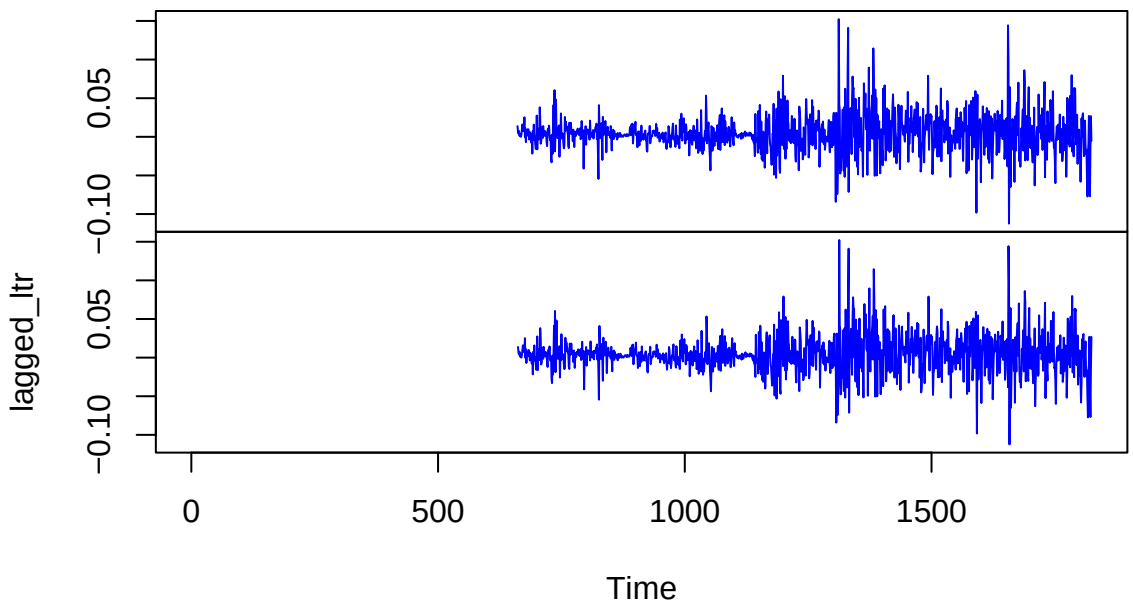

```
#===============
#For BAA
#===============
# Create a lag of BAA
lagged_BAA <- lag(stock_data$BAA)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$BAA,lagged_BAA))
#ts_lagged <- ts(cbind(lagged_BAA))

# Plot the original and lagged BAA
plot(ts_lagged, main = "Original and Lagged BAA", col = c("blue", "red"), lty = c(1, 2))
```



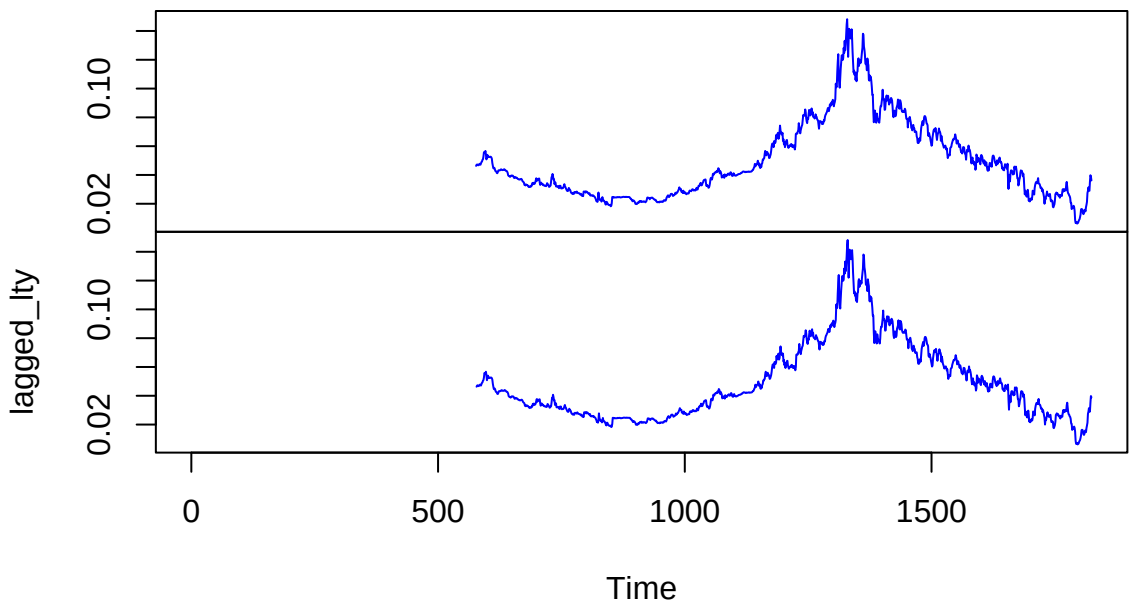

```r
#===============
#For Rfree
#===============
# Create a lag of Rfree
lagged_Rfree <- lag(stock_data$Rfree)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$Rfree,lagged_Rfree))
#ts_lagged <- ts(cbind(lagged_Rfree))

# Plot the original and lagged lagged_Rfree
plot(ts_lagged, main = "Origoinal and Lagged Rfree", col = c("blue", "red"), lty = c(1, 2))
```



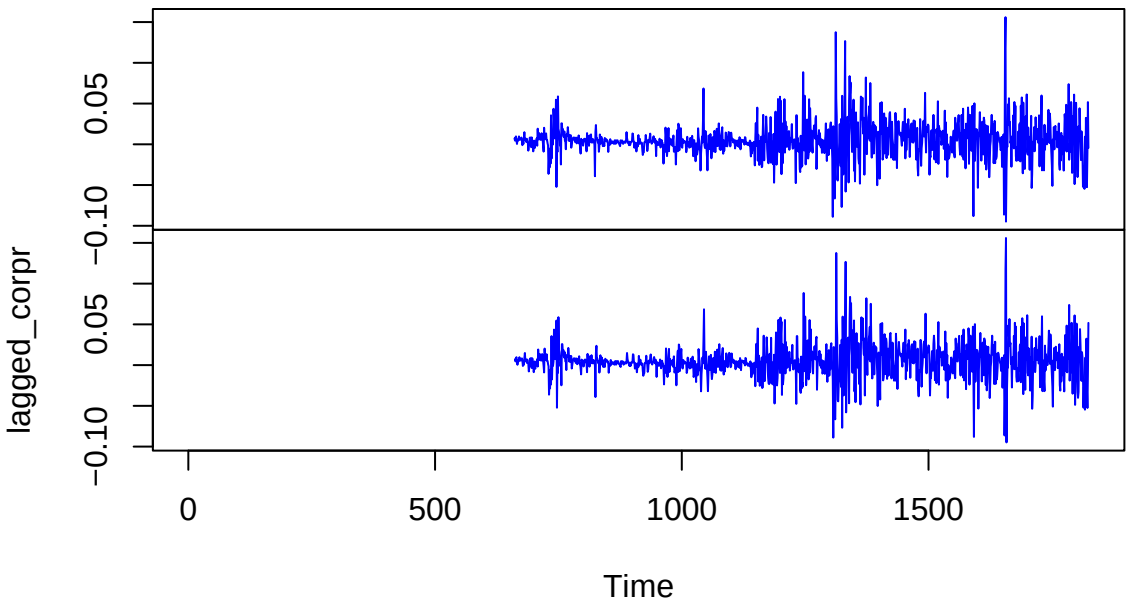

```
#===============
#For ltr
#===============
# Create a lag of ltr
lagged_ltr <- lag(stock_data$ltr)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$ltr,lagged_ltr))
#ts_lagged <- ts(cbind(lagged_ltr))

# Plot the original and lagged ltr
plot(ts_lagged, main = "Original and Lagged ltr", col = c("blue", "red"), lty = c(1, 2))
```



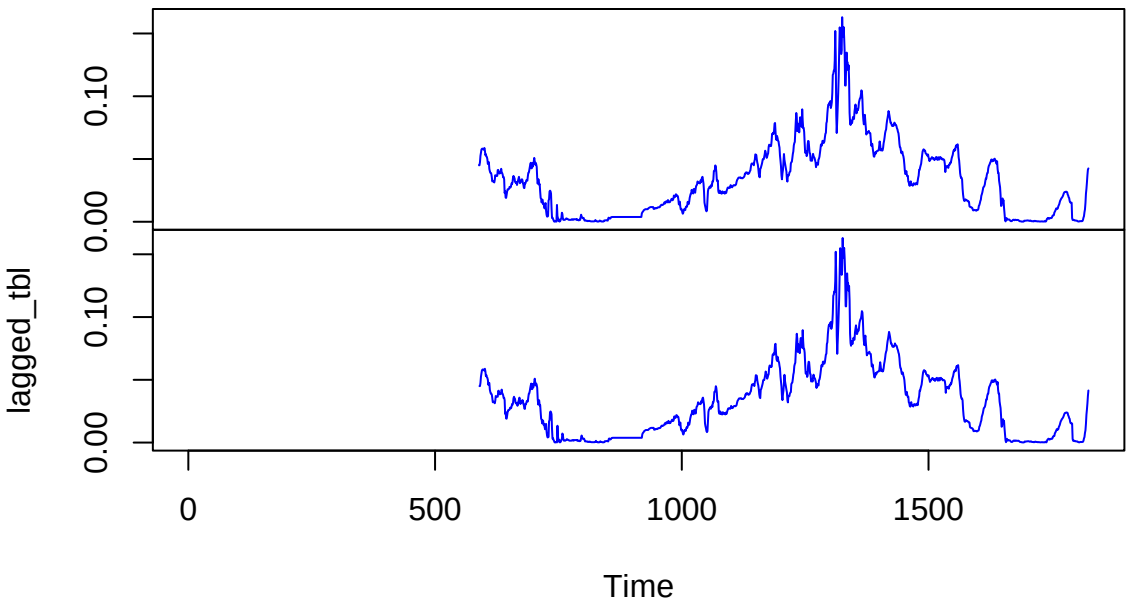

```
#===============
#For lty
#===============
# Create a lag of lty
lagged_lty <- lag(stock_data$lty)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$lty,lagged_lty))
#ts_lagged <- ts(cbind(lagged_lty))

# Plot the original and lagged lty
plot(ts_lagged, main = "Original and Lagged lty", col = c("blue", "red"), lty = c(1, 2))
```




```
#===============
#For corpr
#===============
# Create a lag of corpr
lagged_corpr <- lag(stock_data$corpr)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$corpr,lagged_corpr))

# Plot the Original and lagged corpr
plot(ts_lagged, main = "Original and Lagged corpr", col = c("blue", "red"), lty = c(1, 2))
```




```
#===============
#For tbl
#===============
# Create a lag of tbl
lagged_tbl <- lag(stock_data$tbl)

# Create a new time series object with the lag
ts_lagged <- ts(cbind(stock_data$tbl,lagged_tbl))

# Plot the Original and lagged tbl
plot(ts_lagged, main = "Original and Lagged tbl", col = c("blue", "red"), lty = c(1, 2))
```



#f) Generate scatterplots (again excess returns and each single lagged predictor) #g) Add the respective regression line to the scatterplots

```
################################################
# Excess Returns vs. lagged_dividend_price_ratio
################################################
scatterplot1 <- ggplot(stock_data, aes(x = excess_return, y = lagged_dividend_price_ratio)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_dividend_price_ratio", x = "Excess returns", y

print(scatterplot1)
```

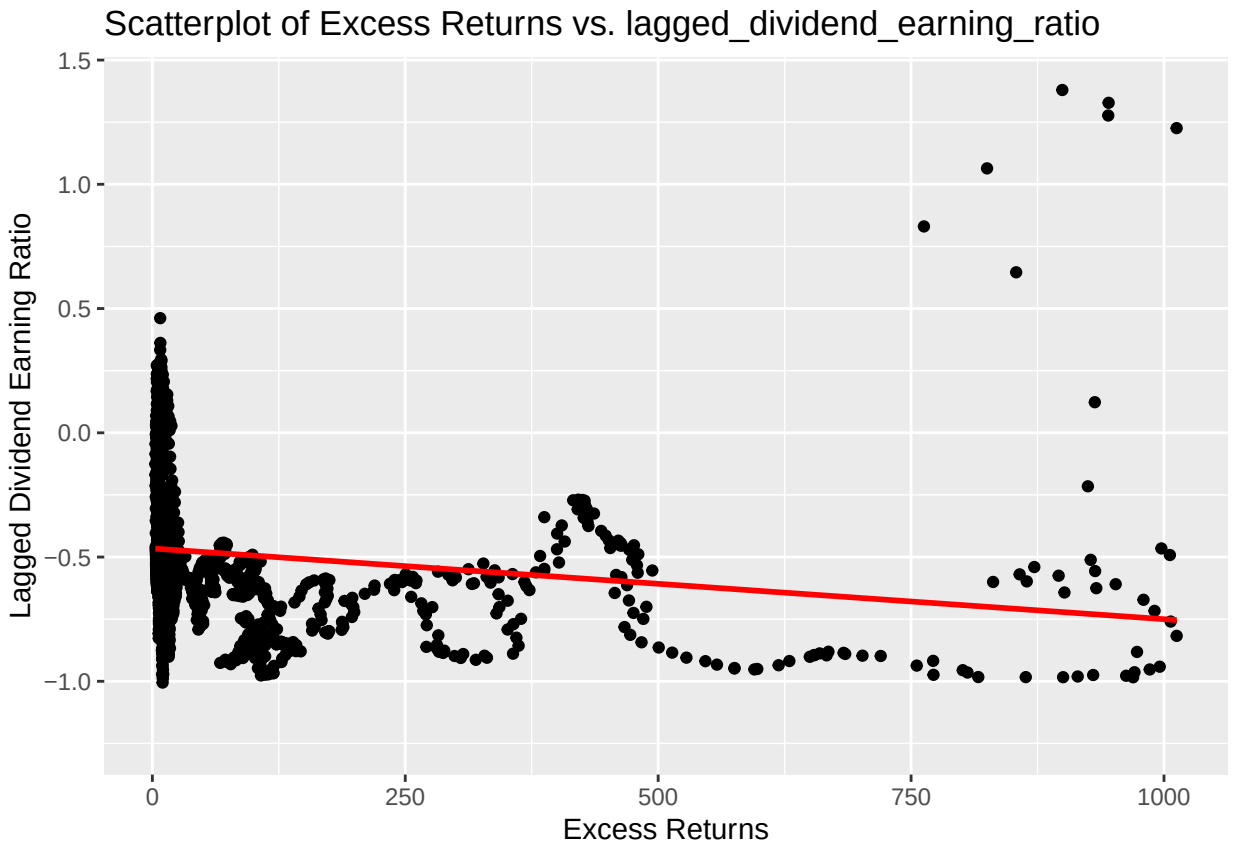
```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 278 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 278 rows containing missing values (`geom_point()`).
```


```
################################################
# Excess Returns vs. lagged_earning_price_ratio
################################################
scatterplot2 <- ggplot(stock_data, aes(x = excess_return, y = lagged_earning_price_ratio)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_earning_price_ratio", x = "Excess Returns", y =

print(scatterplot2)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 278 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 278 rows containing missing values (`geom_point()`).
```

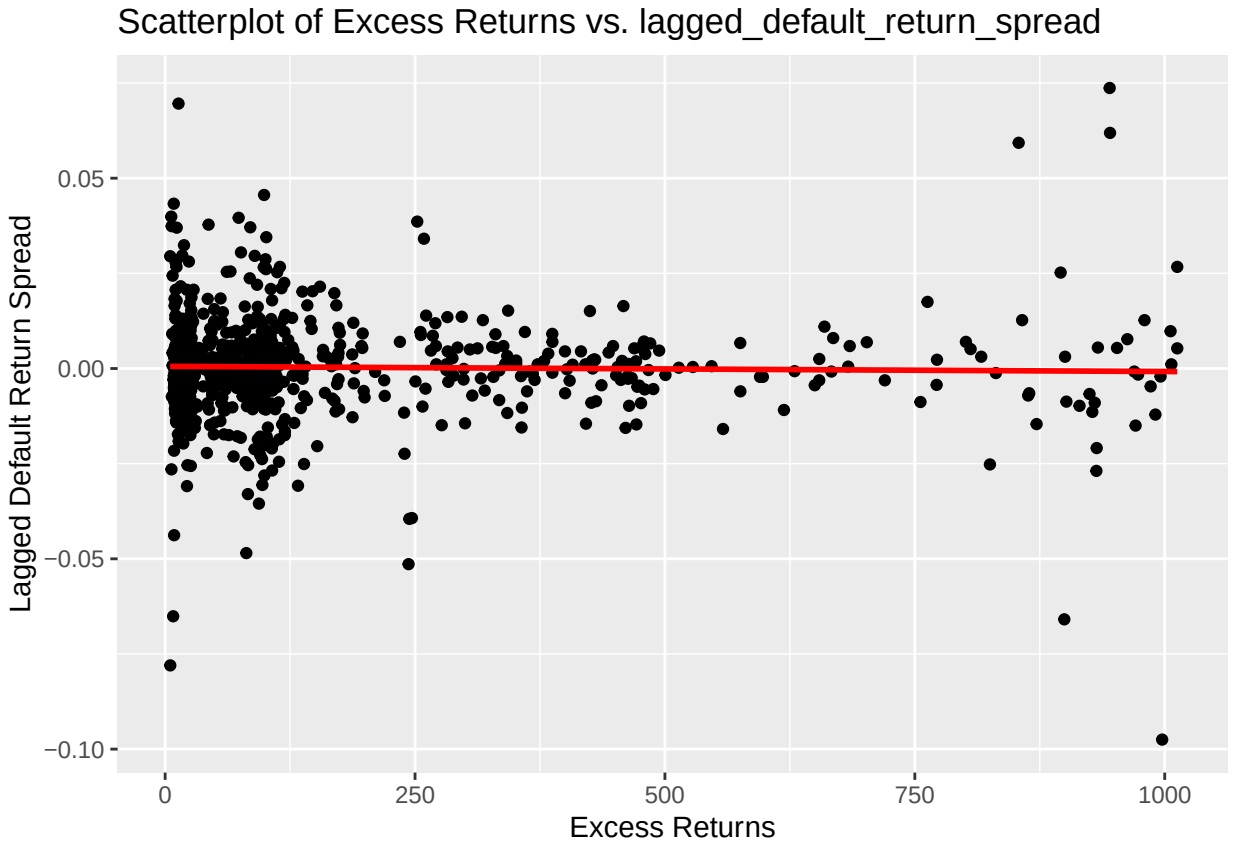

```
################################################
# Excess Returns vs. lagged_dividend_earning_ratio
################################################
scatterplot3 <- ggplot(stock_data, aes(x = excess_return, y = lagged_dividend_earning_ratio)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_dividend_earning_ratio", x = "Excess Returns",

print(scatterplot3)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 274 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 274 rows containing missing values (`geom_point()`).
```



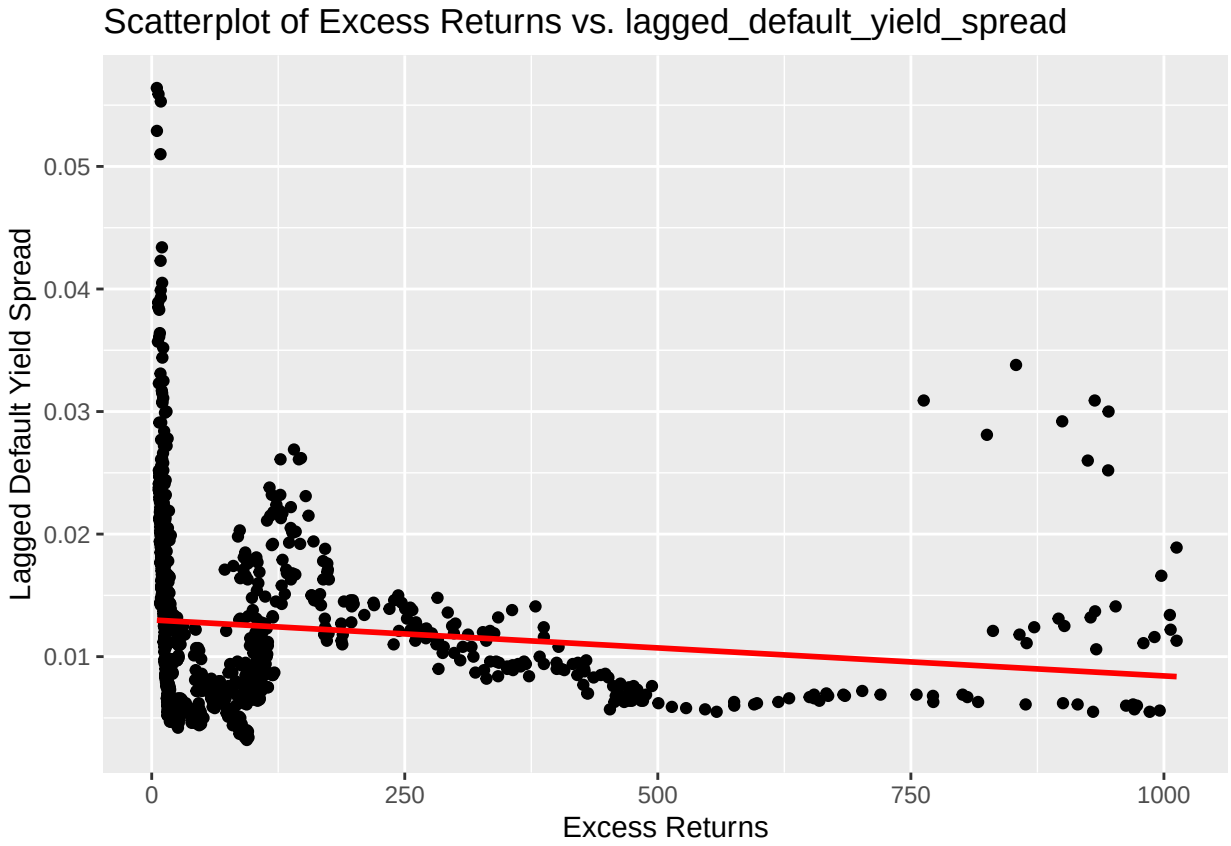

```
################################################
# Excess Returns vs. lagged_default_return_spread
################################################
scatterplot4 <- ggplot(stock_data, aes(x = excess_return, y = lagged_default_return_spread)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_default_return_spread", x = "Excess Returns", y

print(scatterplot4)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 934 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 934 rows containing missing values (`geom_point()`).
```




```
################################################
# Excess Returns vs. lagged_default_yield_spread
################################################
scatterplot5 <- ggplot(stock_data, aes(x = excess_return, y = lagged_default_yield_spread)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_default_yield_spread", x = "Excess Returns", y

print(scatterplot5)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 850 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 850 rows containing missing values (`geom_point()`).
```

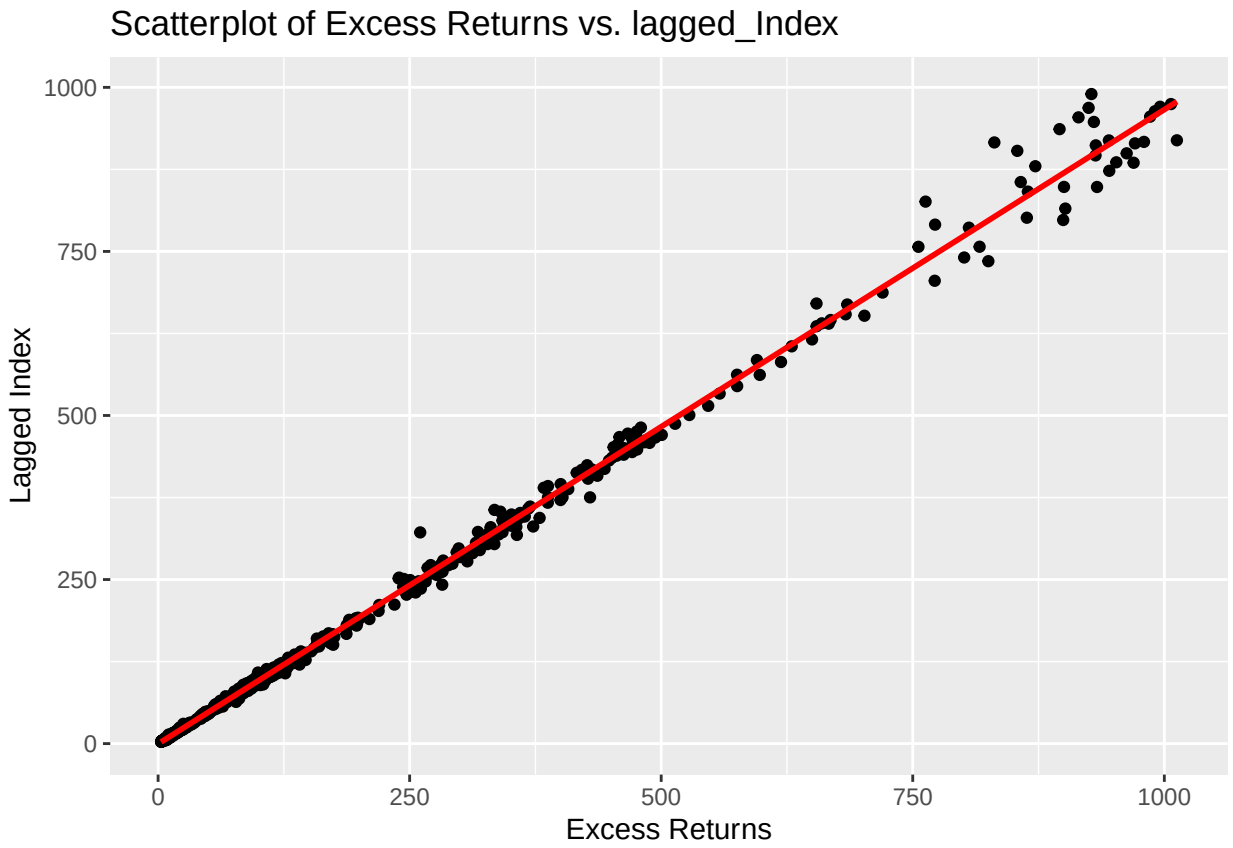



```
################################################
# Excess Returns vs. lagged_term_spread
################################################
scatterplot6 <- ggplot(stock_data, aes(x = excess_return, y = lagged_term_spread)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_term_spread", x = "Excess Returns", y = "Lagge

print(scatterplot6)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 862 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 862 rows containing missing values (`geom_point()`).
```

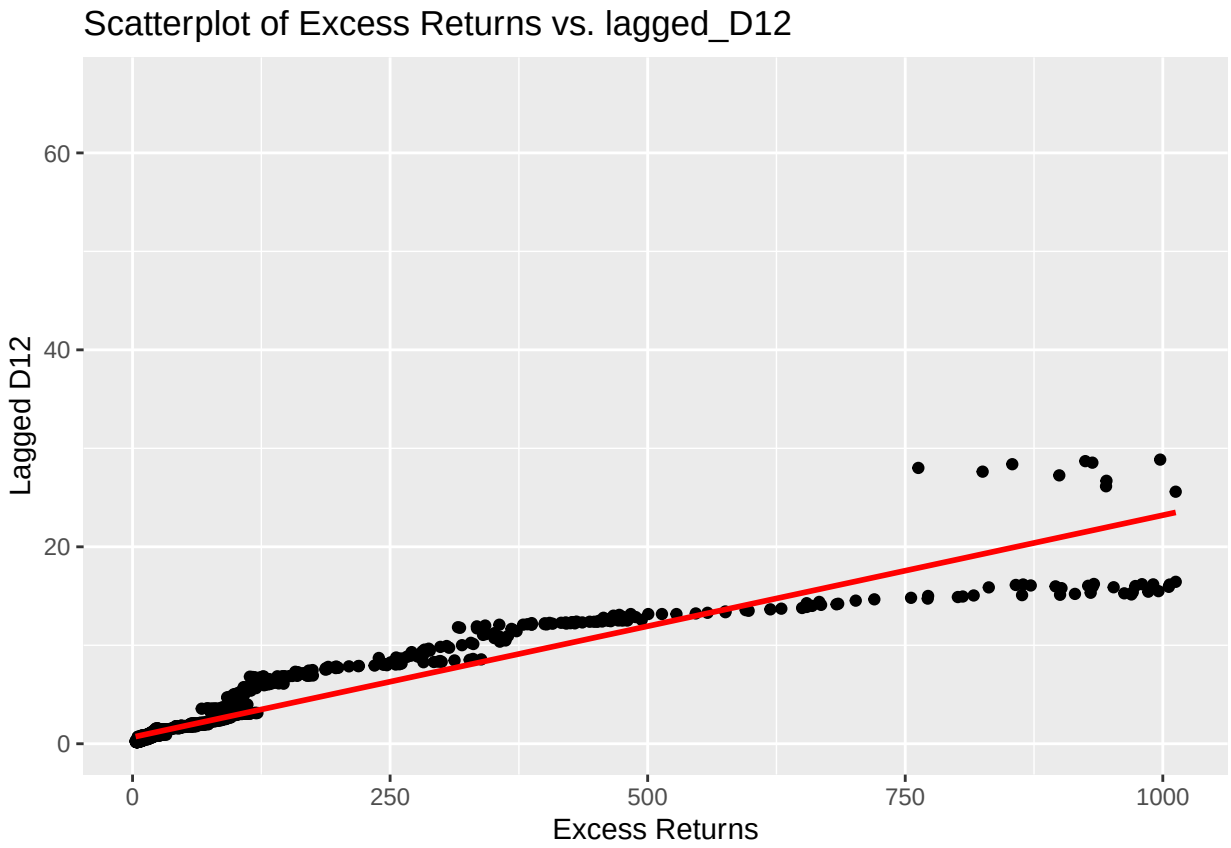

```
################################################
# Excess Returns vs. lagged_Index
################################################
scatterplot7 <- ggplot(stock_data, aes(x = excess_return, y = lagged_Index)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_Index", x = "Excess Returns", y = "Lagged Inde

print(scatterplot7)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 278 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 278 rows containing missing values (`geom_point()`).
```



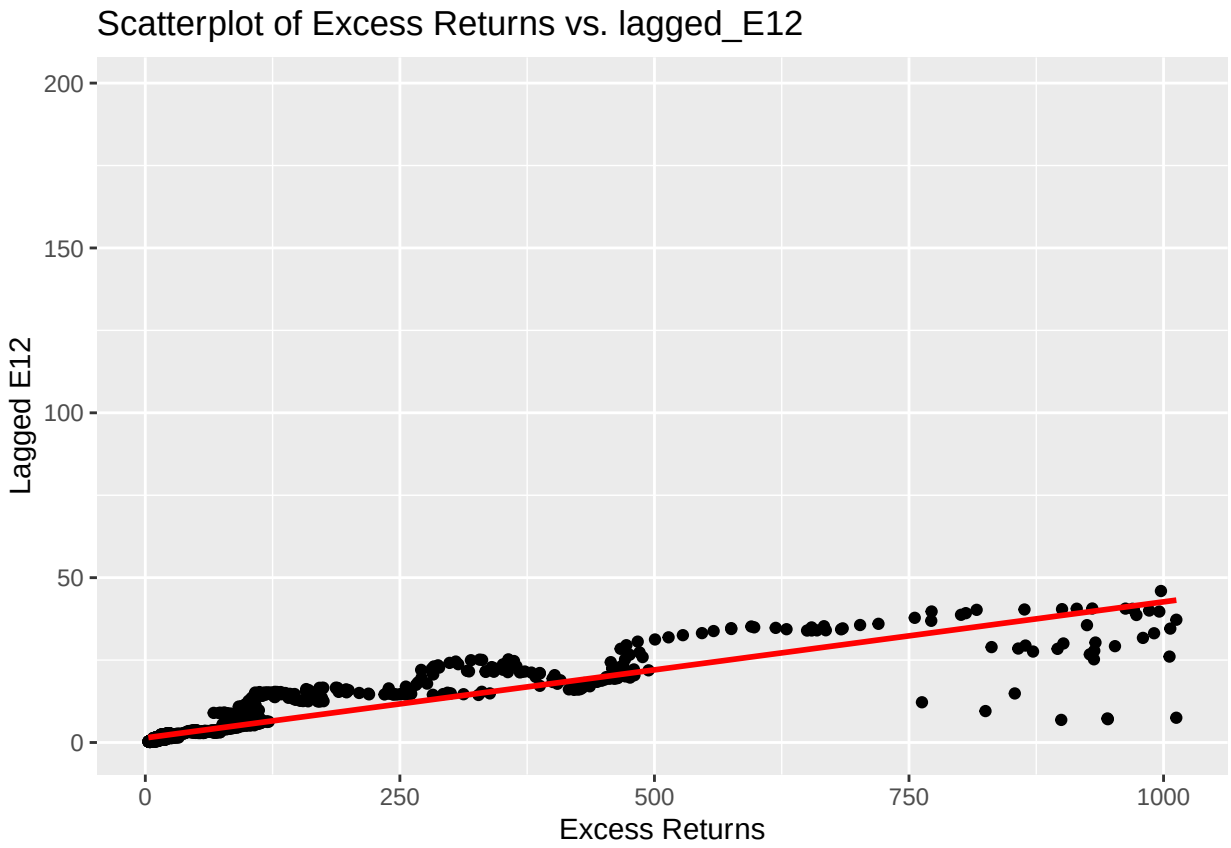

```
################################################
# Excess Returns vs. lagged_D12
################################################
scatterplot8 <- ggplot(stock_data, aes(x = excess_return, y = lagged_D12)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_D12", x = "Excess Returns", y = "Lagged D12")

print(scatterplot8)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 274 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 274 rows containing missing values (`geom_point()`).
```



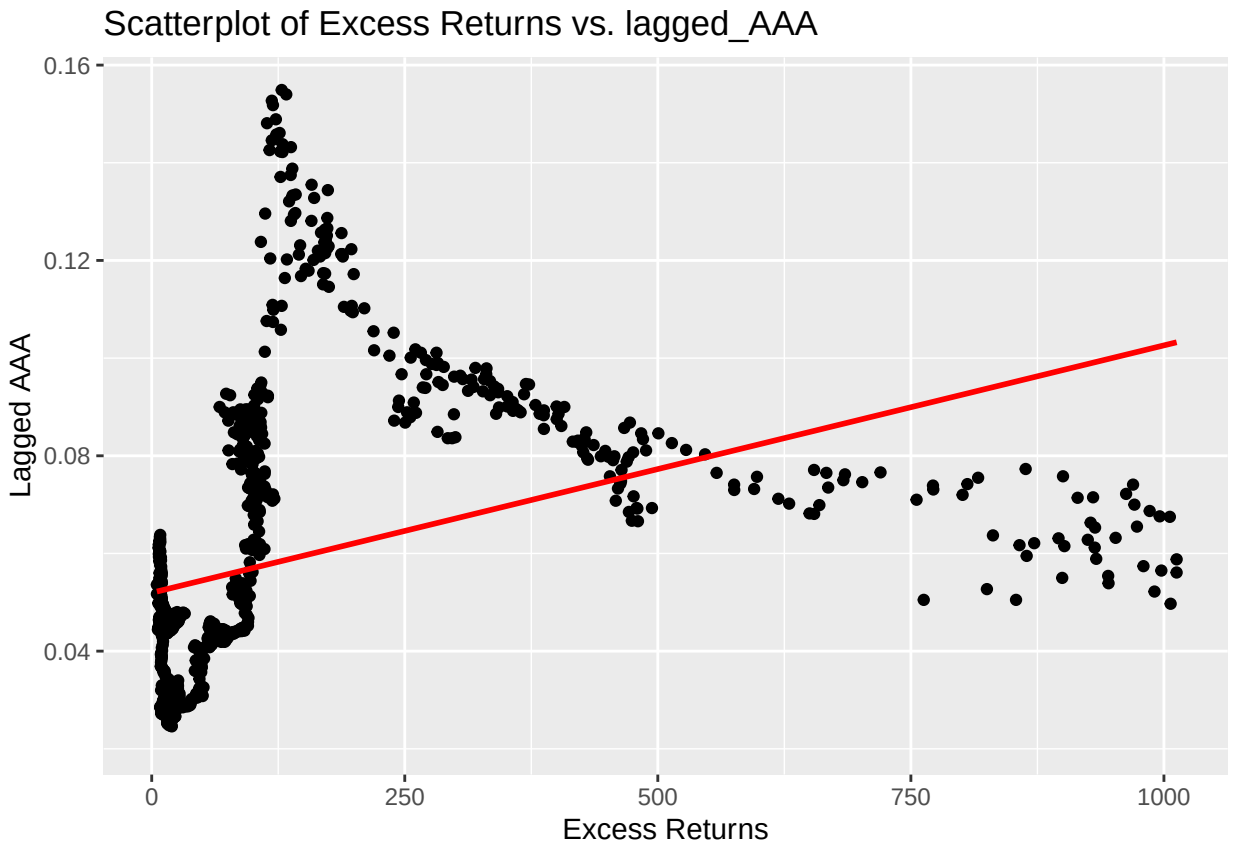

```
################################################
# Excess Returns vs. lagged_E12
################################################
scatterplot9 <- ggplot(stock_data, aes(x = excess_return, y = lagged_E12)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_E12", x = "Excess Returns", y = "Lagged E12")

print(scatterplot9)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 274 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 274 rows containing missing values (`geom_point()`).
```



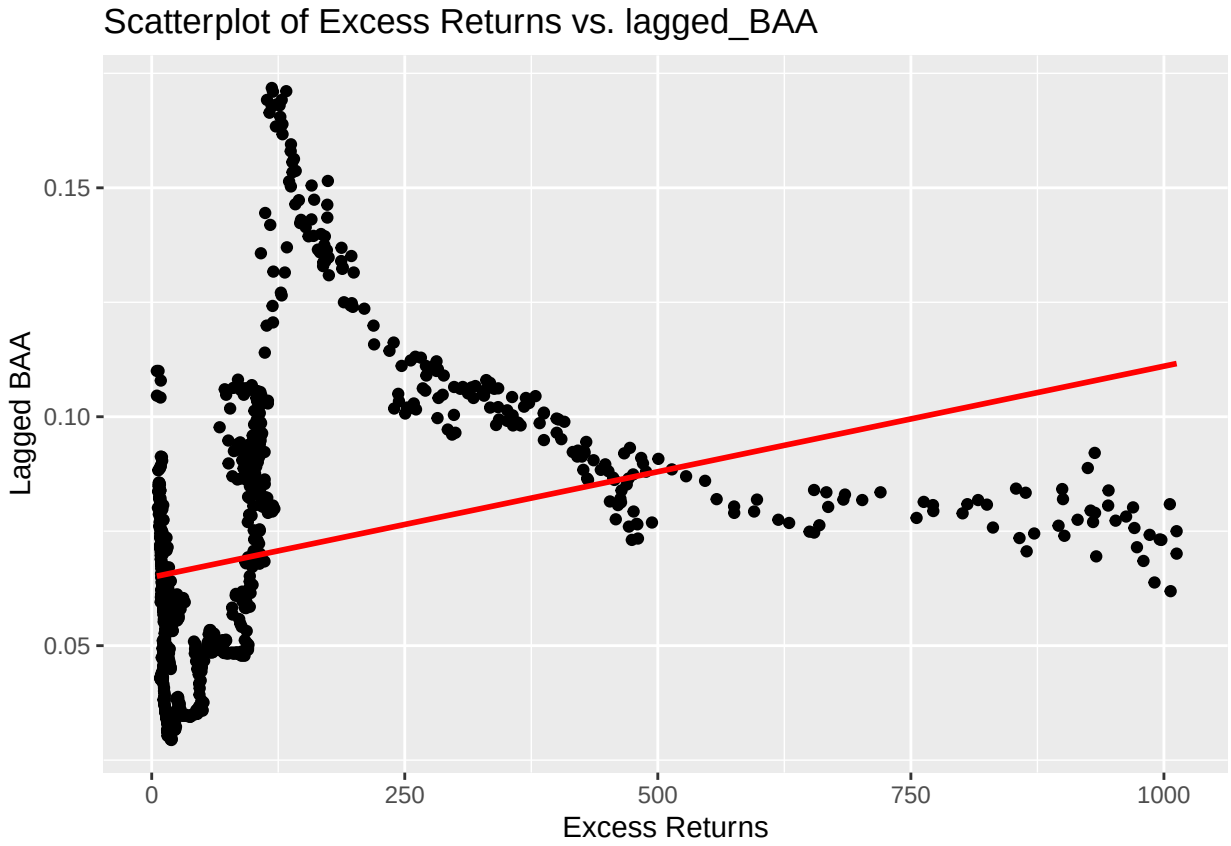

```
################################################
# Excess Returns vs. lagged_AAA
################################################
scatterplot10 <- ggplot(stock_data, aes(x = excess_return, y = lagged_AAA)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_AAA", x = "Excess Returns", y = "Lagged AAA")

print(scatterplot10)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 850 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 850 rows containing missing values (`geom_point()`).
```




```
################################################
# Excess Returns vs. lagged_BAA
################################################
scatterplot11 <- ggplot(stock_data, aes(x = excess_return, y = lagged_BAA)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_BAA", x = "Excess Returns", y = "Lagged BAA")

print(scatterplot11)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 850 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 850 rows containing missing values (`geom_point()`).
```




```
################################################
# Excess Returns vs. lagged_Rfree
################################################
scatterplot12 <- ggplot(stock_data, aes(x = excess_return, y = lagged_Rfree)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_Rfree", x = "Excess Returns", y = "Lagged Rfree

print(scatterplot12)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 275 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 275 rows containing missing values (`geom_point()`).
```


```
################################################
# Excess Returns vs. lagged_ltr
################################################
scatterplot13 <- ggplot(stock_data, aes(x = excess_return, y = lagged_ltr)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_ltr", x = "Excess Returns", y = "Lagged ltr")

print(scatterplot13)
```

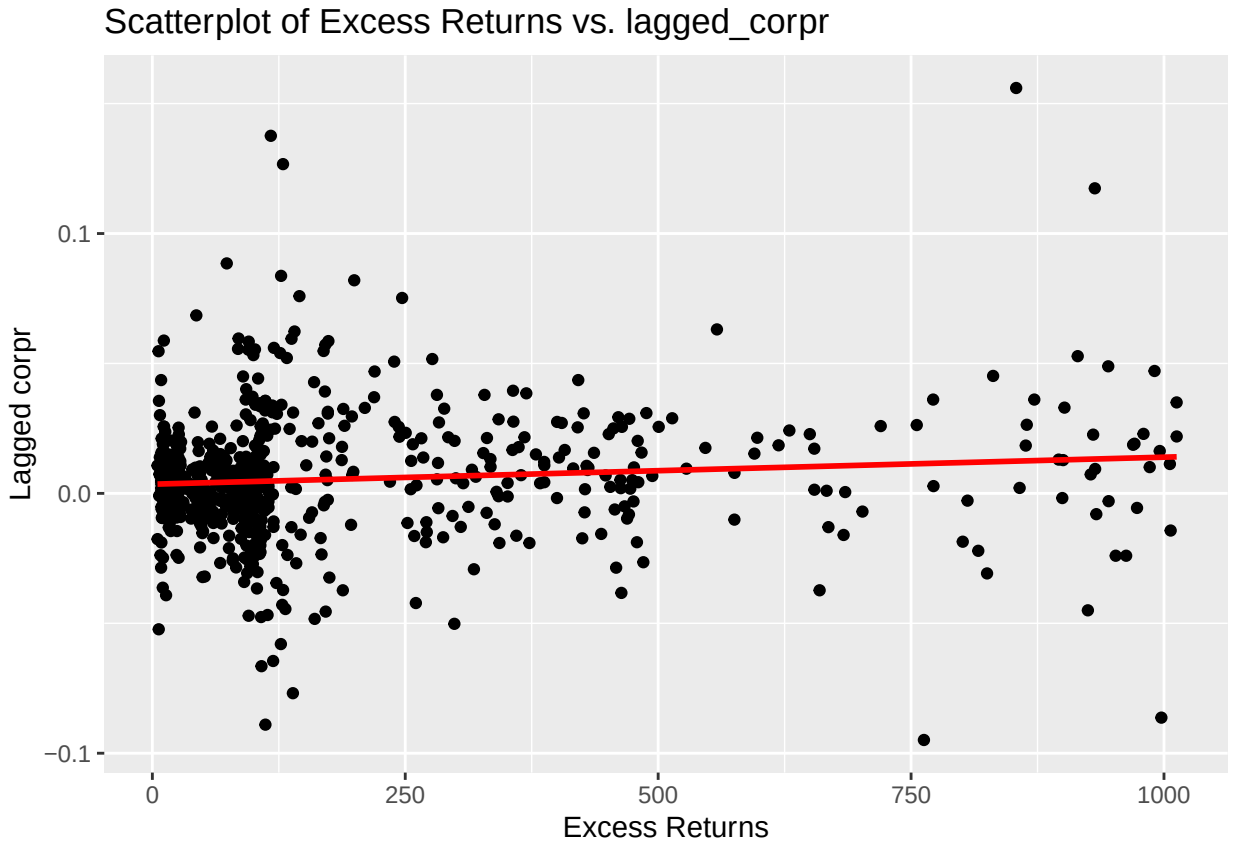
```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 934 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 934 rows containing missing values (`geom_point()`).
```


```
################################################
# Excess Returns vs. lagged_lty
################################################
scatterplot14 <- ggplot(stock_data, aes(x = excess_return, y = lagged_lty)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_lty", x = "Excess Returns", y = "Lagged lty")

print(scatterplot14)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 850 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 850 rows containing missing values (`geom_point()`).
```


```
################################################
# Excess Returns vs. lagged_corpr
################################################
scatterplot15 <- ggplot(stock_data, aes(x = excess_return, y = lagged_corpr)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_corpr", x = "Excess Returns", y = "Lagged corp

print(scatterplot15)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 934 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 934 rows containing missing values (`geom_point()`).
```




```
################################################
# Excess Returns vs. lagged_tbl
################################################
scatterplot16 <- ggplot(stock_data, aes(x = excess_return, y = lagged_tbl)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Scatterplot of Excess Returns vs. lagged_tbl", x = "Excess Returns", y = "Lagged tbl")

print(scatterplot16)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 862 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 862 rows containing missing values (`geom_point()`).
```


Scatterplot of Excess Returns vs. lagged_tbl
Lagged tbl
0.15
0.10
0.05
0.00
0
250
500
750
1000
Excess Returns