

Stats 415 - Final Project

```
# Removing the index column
final_project <- final_project %>% select(Asset_1:Asset_3)
Asset_1 <- final_project %>% select(Asset_1)

# Defining the function
h_min_br <- function(df, h){
  lag_df <- lag(df, n=3, default=head(df, 1))
  return((df - lag_df) / lag_df)
}

# For h = 3
# h3 <- h_min_br(final_project, 3)
# For h = 10
# h10 <- h_min_br(final_project, 10)
# For h = 30
# h30 <- h_min_br(final_project, 30)
# Combining them into one df
df <- cbind(h_min_br(final_project, 3), h_min_br(final_project, 10),
            h_min_br(final_project, 30))

# Renaming the columns
colnames(df) <- c("Asset_1_BRet_3", "Asset_2_BRet_3",
                  "Asset_3_BRet_3", "Asset_1_BRet_10",
                  "Asset_2_BRet_10", "Asset_3_BRet_10",
                  "Asset_1_BRet_30", "Asset_2_BRet_30",
                  "Asset_3_BRet_30")

# Writing the new df to a csv file
# write_csv(Asset_i_BRet_h, "output/bret_NN.csv")
```

2.3 Linear Regression

```
Asset_1_lead <- lead(Asset_1, n=10, default=tail(Asset_1, 1))
Asset_1_HRet_10 <- (Asset_1_lead - Asset_1) / Asset_1
colnames(Asset_1_HRet_10) <- c("Asset_1_HRet_10")
df <- cbind(df, Asset_1_HRet_10)

train_size <- floor(nrow(df) * 0.7)
test_size <- nrow(df) - train_size

train_set <- head(df, train_size)
test_set <- tail(df, test_size)

lr_modl <- lm(Asset_1_HRet_10 ~ Asset_1_BRet_3 + Asset_2_BRet_3 + Asset_3_BRet_3
              , data=train_set)
summary(lr_modl)
```

```

## Call:
## lm(formula = Asset_1_HRet_10 ~ Asset_1_BRet_3 + Asset_2_BRet_3 +
##      Asset_3_BRet_3, data = train_set)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.147288 -0.000917  0.000011  0.000927  0.085485
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.115e-05 4.760e-06 -2.341  0.0192 *
## Asset_1_BRet_3 6.714e-02 3.383e-03 19.843 <2e-16 ***
## Asset_2_BRet_3 2.983e-02 1.878e-03 15.888 <2e-16 ***
## Asset_3_BRet_3 2.134e-02 1.885e-03 11.323 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.002883 on 366908 degrees of freedom
## Multiple R-squared:  0.003444, Adjusted R-squared:  0.003436
## F-statistic: 422.7 on 3 and 366908 DF, p-value: < 2.2e-16

train_pred <- predict.lm(lr_modl, train_set)
test_pred <- predict.lm(lr_modl, test_set)

# In-sample correlation
cor(as.matrix(cbind(train_pred, train_set$Asset_1_HRet_10)))

##          train_pred
## train_pred 1.00000000 0.05868781
##                  0.05868781 1.00000000

# Out-sample correlation
cor(as.matrix(cbind(test_pred, test_set$Asset_1_HRet_10)))

##          test_pred
## test_pred 1.00000000 0.04335569
##                  0.04335569 1.00000000

train_pred <- data.frame(train_pred)
test_pred <- data.frame(test_pred)
colnames(train_pred) <- c("Asset_1_HRet_10_pred")
colnames(test_pred) <- c("Asset_1_HRet_10_pred")
Asset_1_HRet_10_lead <- rbind(train_pred, test_pred)
df <- cbind(df, Asset_1_HRet_10_lead)

# 3 Weeks Rolling correlation
for (i in 1:nrow(df)) {
  start = max(i - 30240, 1)
  df$Rho[i] = cor(df$Asset_1_HRet_10[start:i], df$Asset_1_HRet_10_pred[start:i])
}
plot(df$Rho)

```

