HW7_Mengdi

Mengdi Hao

2024-05-10

What are the latent factors of international currency pricing? And how do these factor move against US equities?

We're going to investigate underlying factors in currency exchange rates and regress the S&P 500 onto this information.

FX data is in FXmonthly.csv. SP500 returns are in sp500csv. Currency codes are in currency codes.txt.

Translate the prices to 'returns' via

fx <- read.csv("FXmonthly.csv") fx <- (fx[2:120,]-fx[1:119,])/(fx[1:119,])

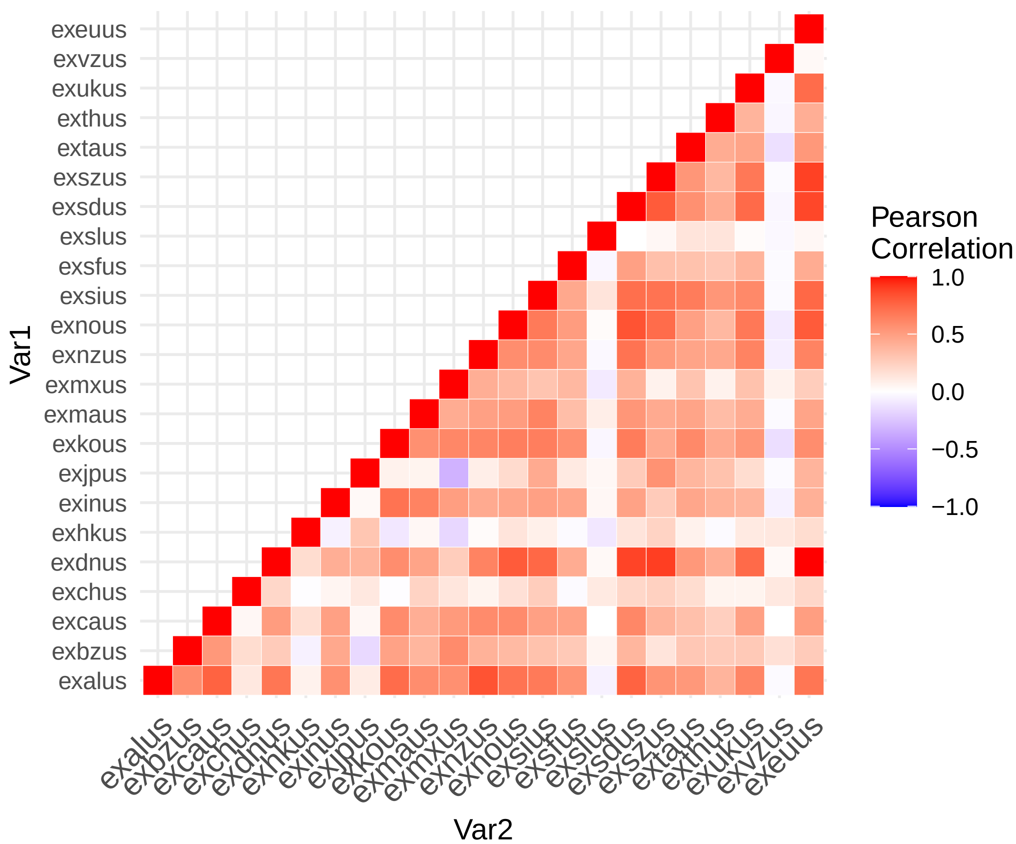
[1] Discuss correlation amongst dimensions of fx. How does this relate to the applicability of factor modelling?

Based on the heatmap below, which visualizes the correlation among different dimensions of foreign exchange (FX) rates, the following insights can be made.

- **High Positive Correlation**: The heatmap shows a high degree of positive correlation among many of the FX rate pairs. This suggests that movements in one currency pair are often mirrored by similar movements in another, possibly due to shared economic or geopolitical factors affecting those currencies.
- **Implication for Factor Modelling**: Given the high correlation among many of the pairs, it is likely that these movements can be largely explained by a few common factors using factor modelling. These factors could be global economic indicators, geopolitical events, or market sentiment. Factor modeling can thus significantly reduce the dimensionality of the problem, focusing on these underlying factors instead of the larger number of individual FX rates.

```
fx <- read.csv("FXmonthly.csv")
fx <- (fx[2:120,]-fx[1:119,])/(fx[1:119,])

library(ggplot2)
library(reshape2)

cormat <- round(cor(fx),2)
melted_cormat <- melt(cormat)
# ggplot(data = melted_cormat, aes(x=Var1, y=Var2, fill=value)) +
#   geom_tile()

# Get lower triangle of the correlation matrix
  get_lower_tri<-function(cormat){
    cormat[upper.tri(cormat)] <- NA
    return(cormat)
  }
  # Get upper triangle of the correlation matrix
```

```r
  get_upper_tri <- function(cormat){
    cormat[lower.tri(cormat)]<- NA
    return(cormat)
  }

upper_tri <- get_upper_tri(cormat)

# Melt the correlation matrix
library(reshape2)
melted_cormat <- melt(upper_tri, na.rm = TRUE)
# Heatmap
library(ggplot2)
ggplot(data = melted_cormat, aes(Var2, Var1, fill = value))+
 geom_tile(color = "white")+
 scale_fill_gradient2(low = "blue", high = "red", mid = "white",
   midpoint = 0, limit = c(-1,1), space = "Lab",
   name="Pearson\nCorrelation") +
  theme_minimal()+
 theme(axis.text.x = element_text(angle = 45, vjust = 1,
    size = 12, hjust = 1))+
 coord_fixed()
```



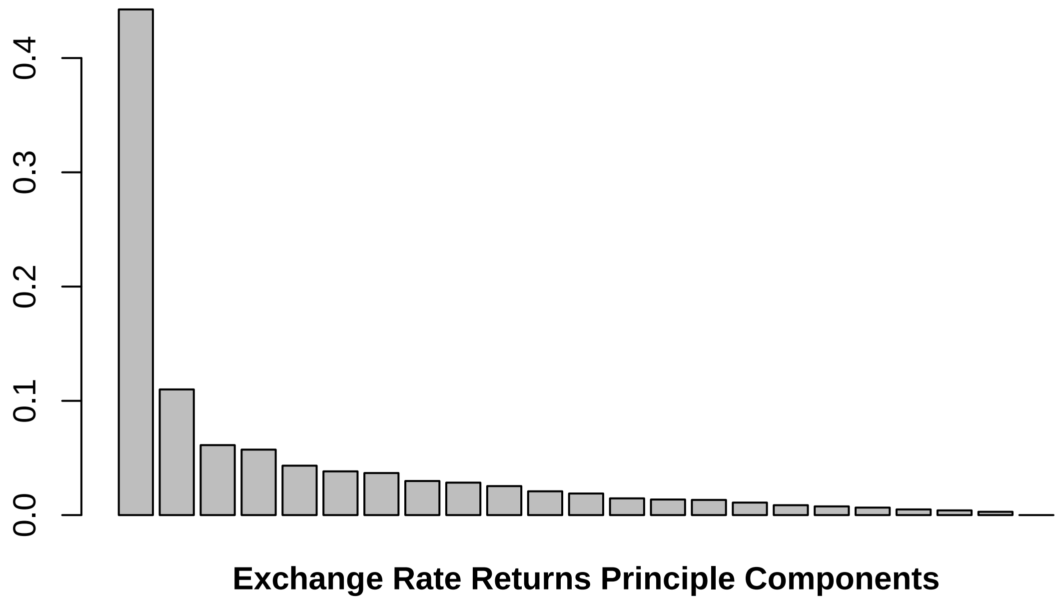
[2] Fit, plot, and interpret principal components.

The PCA results for the exchange rate returns reveal that a significant portion of the variance is captured by the first few principal components, emphasizing the potential for substantial dimensionality reduction.

PC1 explains 44.25% of the variance, highlighting its role in capturing the most dominant market trends. PC2 accounts for 11%, indicating it captures important but less dominant market dynamics. PC3 and subsequent components progressively explain less variance, with diminishing contributions to the total data variability.

The cumulative variance explained by the first three components is approximately 61.37%, suggesting that these components catch the major movements in exchange rates effectively. The steep decline in variance explained by additional components beyond the third suggests that further components contribute less significantly. This indicates that we should focus on the first few components for a simplified yet effective analysis and modeling of exchange rate movements.

```r
# Standardize fx
fx <- scale(fx)

# Apply PCA on standardized fx
pcfx <- prcomp(fx)

# Predict the PCA value
zfx <- predict(pcfx)


# Extract variance explained ratio
PEVs <- pcfx$sdev^2

# Total variance
total <- sum(PEVs)

# Barplot
barplot(PEVs/total)
mtext(side=1, "Exchange Rate Returns Principle Components", line=1, font=2)
```




```r
# Print summary of PCA
summary(pcfx)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     3.1904 1.5905 1.18680 1.14792 0.99740 0.93815 0.92009
## Proportion of Variance 0.4425 0.1100 0.06124 0.05729 0.04325 0.03827 0.03681
## Cumulative Proportion  0.4425 0.5525 0.61377 0.67107 0.71432 0.75258 0.78939
##                            PC8     PC9    PC10    PC11    PC12    PC13    PC14
## Standard deviation     0.82835 0.80841 0.76390 0.69185 0.65917 0.58024 0.56012
## Proportion of Variance 0.02983 0.02841 0.02537 0.02081 0.01889 0.01464 0.01364
## Cumulative Proportion  0.81923 0.84764 0.87301 0.89382 0.91271 0.92735 0.94099
##                           PC15    PC16    PC17    PC18    PC19    PC20    PC21
## Standard deviation     0.55254 0.50190 0.44624 0.41834 0.38808 0.33724 0.30771
## Proportion of Variance 0.01327 0.01095 0.00866 0.00761 0.00655 0.00494 0.00412
## Cumulative Proportion  0.95427 0.96522 0.97388 0.98149 0.98803 0.99298 0.99709
##                          PC22    PC23
## Standard deviation     0.2580 0.01557
## Proportion of Variance 0.0029 0.00001
## Cumulative Proportion  1.0000 1.00000
```

[3] Regress SP500 returns onto currency movement factors, using both 'glm on first K' and lasso techniques. Use the results to add to your factor interpretation.

From the GLM regression results, it is evident that only PC1, PC2, and PC3 show significant effects at the 5% significance level. Specifically, PC1, with a positive coefficient of 0.00597, suggests a significant positive relationship with the S&P 500 returns, indicating that a one unit increase in PC1 leads to approximately a 0.597% increase in the index. Conversely, PC2 and PC3, which exhibit negative coefficients of -0.111 and -0.0082 respectively, imply that positive changes in these principal components result in a decrease in the S&P 500 returns. These findings underscore the influence of currency movement factors on stock market performance, validating the use of these principal components in understanding and predicting market trends. The significant principal components capture substantial aspects of the variability in currency movements, accounting for 44.25%, 11%, and 6.12% of the variance respectively, thereby providing a meaningful reduction in dimensionality and highlighting key drivers of market behavior.

The initial LASSO regression with a single lambda value showed no non-zero coefficients, indicating an overly strict penalty that may have suppressed meaningful predictors.

The optimized LASSO regression, determined via cross-validation (CV), identified four significant PCs (PC1, PC2, and PC3) with non-zero coefficients. This suggests that when the penalty is appropriately adjusted, these components still hold predictive power over S&P 500 returns.

```r
sp <- read.csv("sp500.csv")

# Combine the top 12 PCs with SP500 into one dataframe
data <- data.frame(sp500 = sp$sp500, zfx[, 1:12])

reg1 <- glm(sp500 ~ ., data = data)
summary(reg1)
```

```
##
## Call:
## glm(formula = sp500 ~ ., data = data)
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0004431  0.0036339   0.122  0.90318
## PC1          0.0059741  0.0011438   5.223 8.86e-07 ***
## PC2         -0.0111795  0.0022945  -4.872 3.88e-06 ***
## PC3         -0.0082100  0.0030749  -2.670  0.00878 **
## PC4          0.0024263  0.0031790   0.763  0.44704
## PC5         -0.0042391  0.0036588  -1.159  0.24922
## PC6         -0.0002007  0.0038899  -0.052  0.95896
## PC7          0.0012710  0.0039662   0.320  0.74925
## PC8          0.0022731  0.0044055   0.516  0.60694
## PC9         -0.0014776  0.0045142  -0.327  0.74406
## PC10        -0.0056542  0.0047772  -1.184  0.23923
## PC11        -0.0020874  0.0052747  -0.396  0.69310
## PC12         0.0029706  0.0055362   0.537  0.59269
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.001571456)
##
##     Null deviance: 0.26463  on 118  degrees of freedom
```

```
## Residual deviance: 0.16657  on 106  degrees of freedom
## AIC: -416.29
##
## Number of Fisher Scoring iterations: 2
```

```r
library(gamlr)
```

```
## Warning: package 'gamlr' was built under R version 4.3.3
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 4.3.3
```

```r
# Transform zfx to sparse matrix
X <- as.matrix(data.frame(zfx[, 1:12]))

# sp500 as y
y <- sp$sp500

set.seed(123)

# Use cv.gamlr to choose optimal lambda
reg2 <- cv.gamlr(x=X, y=y, nfold=20)

# Print optimallambda
print(reg2$lambda.min)
```

```
## [1] 0.003093208
```

```r
# Print coefs
coef(reg2)
```

```
## 13 x 1 sparse Matrix of class "dgCMatrix"
##                   seg18
## intercept  0.0004430924
## PC1        0.0032649286
## PC2       -0.0057448941
## PC3       -0.0009270600
## PC4        .
## PC5        .
## PC6        .
## PC7        .
## PC8        .
## PC9        .
## PC10       .
## PC11       .
## PC12       .
```

[4] Fit lasso to the original covariates and describe how it differs from PCR here.

From the plots below, comparing the LASSO applied directly to the original covariates (fx) versus the LASSO after principal component regression (PCR) reveals interesting differences in model behavior:

Number of Variables: - In Original LASSO, the plot shows that the number of nonzero coefficients decreases as log lambda increases. Starting with all variables being included at the lowest lambda, the number quickly reduces to only a few variables as the penalization becomes stronger. It tends to retain more covariates initially, providing a broader but potentially noisier or more redundant view of the predictors' impact on the response variable. This approach might be beneficial when no clear understanding of the underlying data structure exists, or when all original variables are considered potentially influential. - In LASSO PCR, The number of variables starts at a lower point (12, the number of principal components used), and similarly, it decreases as lambda increases. This model inherently begins with fewer variables due to the dimensionality reduction performed by PCR before applying LASSO. This can enhance model interpretability and efficiency but at the cost of losing some data granularity. It might overlook some finer details that could be important if minor components have significant explanatory power regarding the response.

Error Characteristics: - Both models show a U-shaped error curve with respect to log lambda, which is typical for LASSO models. This shape reflects the trade-off between model complexity and the bias-variance trade-off. - The original LASSO shows lower mean squared errors (MSE) overall compared to LASSO PCR. This could be due to the fact that using the original covariates, the original LASSO retains more subtle information that reduces MSE of the model. - The best performing lambda (where the MSE is minimized) is reached earlier in the lambda sequence for the original LASSO compared to LASSO PCR, indicating a higher sensitivity to regularization.

```r
set.seed(123)
reg3 <- cv.gamlr(x=as.matrix(fx), y, nfold=20)

par(mfrow=c(1,2))
plot(reg3, main="Original LASSO")
plot(reg2, main="LASSO PCR")
```