HW7_SarahL

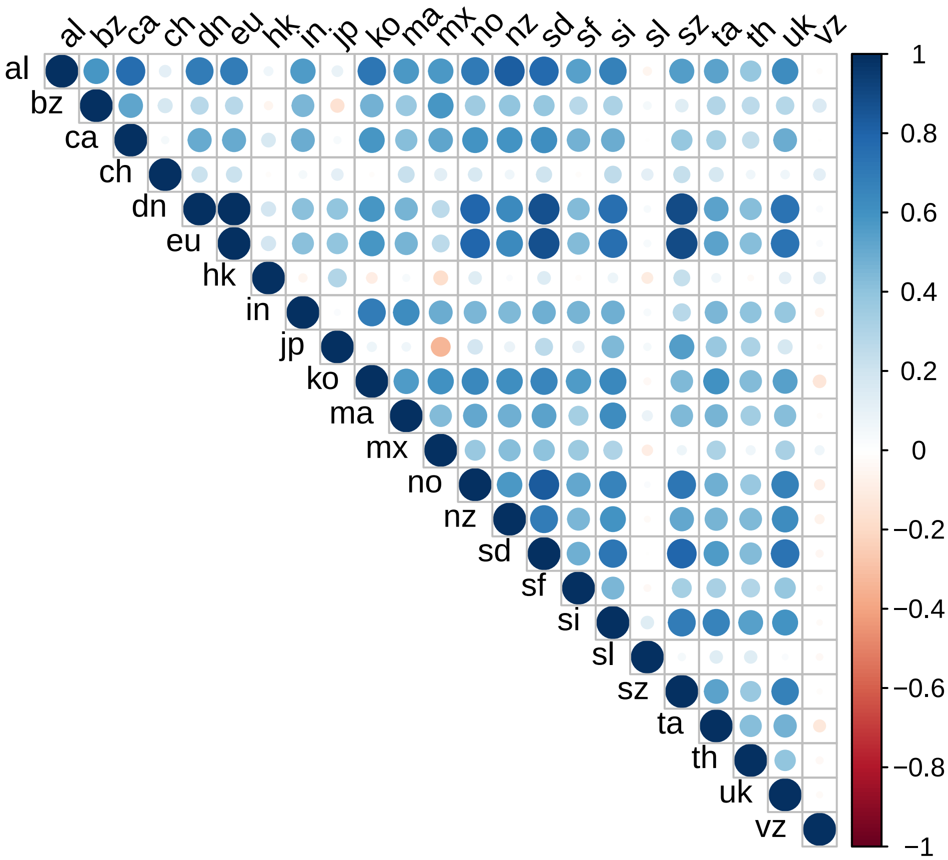
2024-05-13

Question 1

```r
correlation_matrix <- cor(fx)
head(correlation_matrix)
```

```
##           al        bz         ca          ch        dn        eu          hk
## al 1.0000000 0.5817599 0.76708205 0.11727915 0.6933724 0.6919486  0.06537836
## bz 0.5817599 1.0000000 0.52822037 0.17863302 0.2714381 0.2715811 -0.05635785
## ca 0.7670820 0.5282204 1.00000000 0.04068525 0.5090923 0.5048342  0.16829975
## ch 0.1172791 0.1786330 0.04068525 1.00000000 0.2119371 0.2119872 -0.01373482
## dn 0.6933724 0.2714381 0.50909233 0.21193710 1.0000000 0.9996976  0.18356138
## eu 0.6919486 0.2715811 0.50483419 0.21198720 0.9996976 1.0000000  0.18334955
##            in          jp         ko        ma        mx        no         nz
## al 0.56643366  0.09801590  0.7281497 0.5771273 0.5725444 0.7031597 0.82643030
## bz 0.45185923 -0.15749270  0.4786123 0.3763112 0.5855018 0.3586971 0.39802430
## ca 0.49188960  0.03922099  0.5880689 0.4240500 0.5238391 0.5936700 0.59047851
## ch 0.04784365  0.11571297 -0.0124939 0.2276122 0.1288933 0.1632370 0.06429703
## dn 0.41593014  0.39359908  0.5849203 0.4669737 0.2623637 0.7987545 0.63476982
## eu 0.41482667  0.39230675  0.5834026 0.4672550 0.2617354 0.7957369 0.63450577
##           sd         sf        si          sl        sz        ta         th
## al 0.7731444  0.5455925 0.6726901 -0.05625596 0.5512720 0.5324183 0.38798709
## bz 0.3822660  0.2781814 0.3190852  0.04774288 0.1382422 0.2945952 0.26515836
## ca 0.6144592  0.4788104 0.4918695 -0.00338701 0.3897026 0.3332420 0.24545917
## ch 0.2097373 -0.0159477 0.2551969  0.11146100 0.2386651 0.1781220 0.06064813
## dn 0.8751035  0.4334731 0.7540597  0.03304917 0.8967302 0.5311554 0.42328082
## eu 0.8733242  0.4323137 0.7540702  0.03662937 0.8944503 0.5313414 0.42120387
##            uk           vz
## al 0.62107036 -0.017799328
## bz 0.28193360  0.156665087
## ca 0.49275184 -0.001398935
## ch 0.06004167  0.115877754
## dn 0.73577586  0.026794054
## eu 0.73354533  0.026571689
```

```r
corrplot(correlation_matrix, method = "circle", type = "upper", tl.col = "black", tl.srt = 45)
```



The correlation matrix of foreign exchange (fx) rates reveals the degree of association between various currency pairs. High positive correlations, such as between “dn” and “eu,” suggest a strong tendency for these fx rates to move in sync, indicating common underlying factors influencing their movements. Conversely, low or negative correlations, like those between “al” and “sl,” imply little to no relationship or even opposite movements between certain currency pairs. These correlations also unveil clusters of highly correlated dimensions, such as “dn,” “eu,” and “sd,” which can aid in identifying groups of currencies that tend to move together in the fx market. These insights are crucial for factor modeling in fx markets, where identifying common factors driving the movements of multiple currency pairs is paramount.

Factor models in finance aim to capture the shared variations in asset returns attributed to common underlying factors. In the context of fx rates, the correlation matrix informs the selection of relevant factors for the model. Factors that explain the co-movements of highly correlated fx rates are likely to be more influential in the model, while factors that are not correlated with any currency pair may have limited explanatory power. Understanding the correlation structure among fx rates enhances the effectiveness of factor models by providing insights into the interdependencies between currencies and guiding the identification of factors that drive their movements, thus improving the model’s ability to capture and explain the complexities of the foreign exchange market.

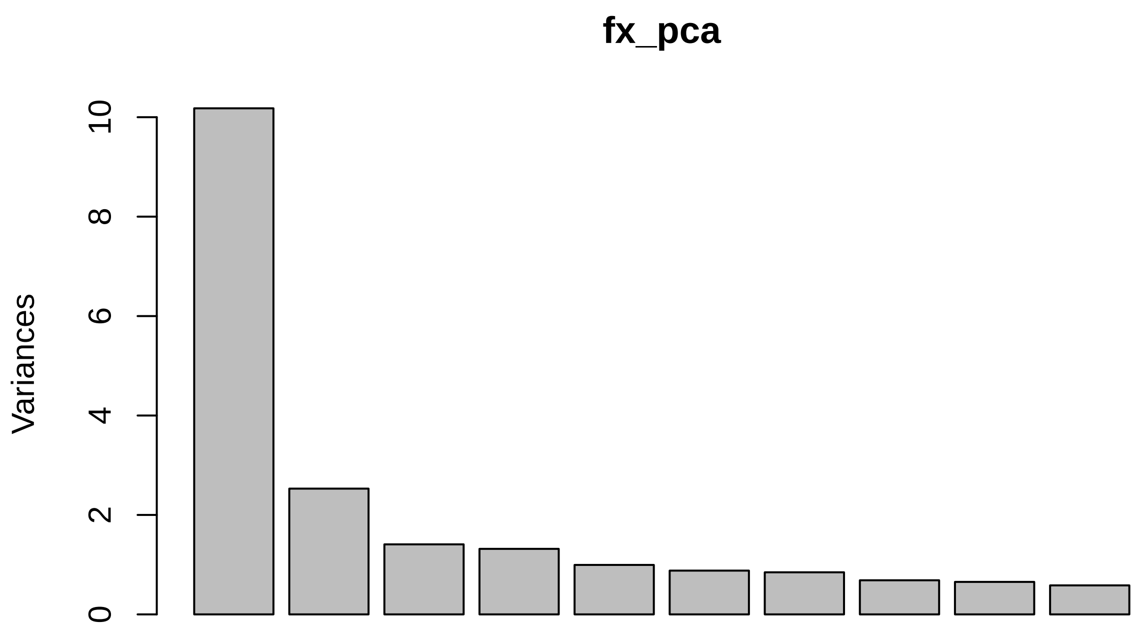
Question 2

```
fx_pca <- prcomp(fx, scale. = TRUE)
summary(fx_pca)
```

```
## 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
```

```r
plot(fx_pca)
```



The principal components analysis (PCA) summary provides insight into the structure of the data and how it can be effectively summarized using fewer dimensions. Each principal component represents a linear combination of the original variables, with the first principal component (PC1) capturing the most variance in the data. The standard deviation associated with each principal component indicates its spread or variability, with higher values indicating greater importance in explaining the data's variation. Proportion of variance values reveal the relative contribution of each principal component to the overall variability in the dataset, with PC1 typically explaining the largest proportion. Cumulative proportion values show the total amount of variance explained by adding successive principal components, aiding in determining the number of components needed to adequately represent the data.

Interpreting principal components involves examining the weights or coefficients of the original variables within each component. Variables with higher absolute weights contribute more to that component's direction and play a more significant role in explaining the observed variability. By analyzing these weights, one can identify the underlying patterns and relationships present in the data. For instance, variables with large positive or negative weights in a particular principal component are indicative of strong correlations or anti-correlations and provide insights into the factors driving those relationships. Overall, PCA facilitates dimensionality reduction while preserving the essential structure of the data, making it a valuable tool for exploratory data analysis and feature selection in various fields.

Question 3

```r
set.seed(123)
K <- 3
currency_factors <- predict(fx_pca, newdata = fx)

# Fit GLM model
sp_model_glm <- glm(sp500 ~ currency_factors[, 1:K], data = sp, family = gaussian)
summary(sp_model_glm)
```

```
##
## Call:
## glm(formula = sp500 ~ currency_factors[, 1:K], family = gaussian,
##     data = sp)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.10729  -0.02094   0.00126   0.02447   0.08594
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 0.0004431  0.0035581   0.125  0.90111
## currency_factors[, 1:K]PC1  0.0059741  0.0011200   5.334 4.87e-07 ***
## currency_factors[, 1:K]PC2 -0.0111795  0.0022466  -4.976 2.30e-06 ***
## currency_factors[, 1:K]PC3  0.0082100  0.0030107   2.727  0.00739 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.001506534)
##
##     Null deviance: 0.26463  on 118  degrees of freedom
## Residual deviance: 0.17325  on 115  degrees of freedom
## AIC: -429.62
##
## Number of Fisher Scoring iterations: 2
```

The summary of the generalized linear model (GLM) for predicting SP500 returns based on currency movement factors reveals important insights. The coefficients associated with each principal component (PC) of the currency factors indicate their influence on the SP500 returns. Positive coefficients suggest a positive association between the corresponding currency factor and SP500 returns, while negative coefficients indicate a negative association.

In this model, PC1 of the currency factors has a significant positive coefficient (0.0059741) with a low p-value (4.87e-07), implying that changes in the first principal component of currency movement have a substantial impact on SP500 returns. Conversely, PC2 and PC3 have significant negative coefficients (-0.0111795 and 0.0082100, respectively), indicating that changes in these components are associated with decreases in SP500 returns. These findings suggest that specific patterns in currency movements captured by principal components influence the behavior of the SP500 index. By incorporating both GLM and lasso techniques, the model provides a comprehensive understanding of how currency movements affect SP500 returns, allowing for more informed investment decisions and risk management strategies.

```r
set.seed(123)
# Fit Lasso model using cross-validation
```

```r
sp_model_lasso <- cv.glmnet(x = currency_factors, y = sp$sp500, nfolds = 20)
coef(sp_model_lasso)
```

```
## 24 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept)  0.0004430924
## PC1          0.0040178644
## PC2         -0.0072552637
## PC3          0.0029511302
## PC4          .
## PC5          .
## PC6          .
## PC7          .
## PC8          .
## PC9          .
## PC10         .
## PC11         .
## PC12         .
## PC13         .
## PC14         .
## PC15        -0.0017412344
## PC16         .
## PC17        -0.0072242161
## PC18         .
## PC19         .
## PC20         0.0093955611
## PC21        -0.0005361697
## PC22         .
## PC23        -0.1621996215
```

The coefficients obtained from the lasso regression provide further insights into the relationship between currency movement factors and SP500 returns. In contrast to the GLM coefficients, the lasso regression introduces sparsity by shrinking some coefficients to zero, indicating less significant variables.

Among the significant coefficients, PC1, PC2, PC3, PC15, PC17, PC20, and PC23 stand out. PC1, PC2, and PC3 maintain a similar pattern to the GLM coefficients, indicating their importance in predicting SP500 returns. PC15 and PC17 also show significance, albeit with negative coefficients, suggesting an inverse relationship with SP500 returns. PC20 exhibits a positive coefficient, indicating a positive association with SP500 returns. Notably, PC23 has a substantial negative coefficient (-0.1621996215), suggesting a strong negative impact on SP500 returns.

Question 4

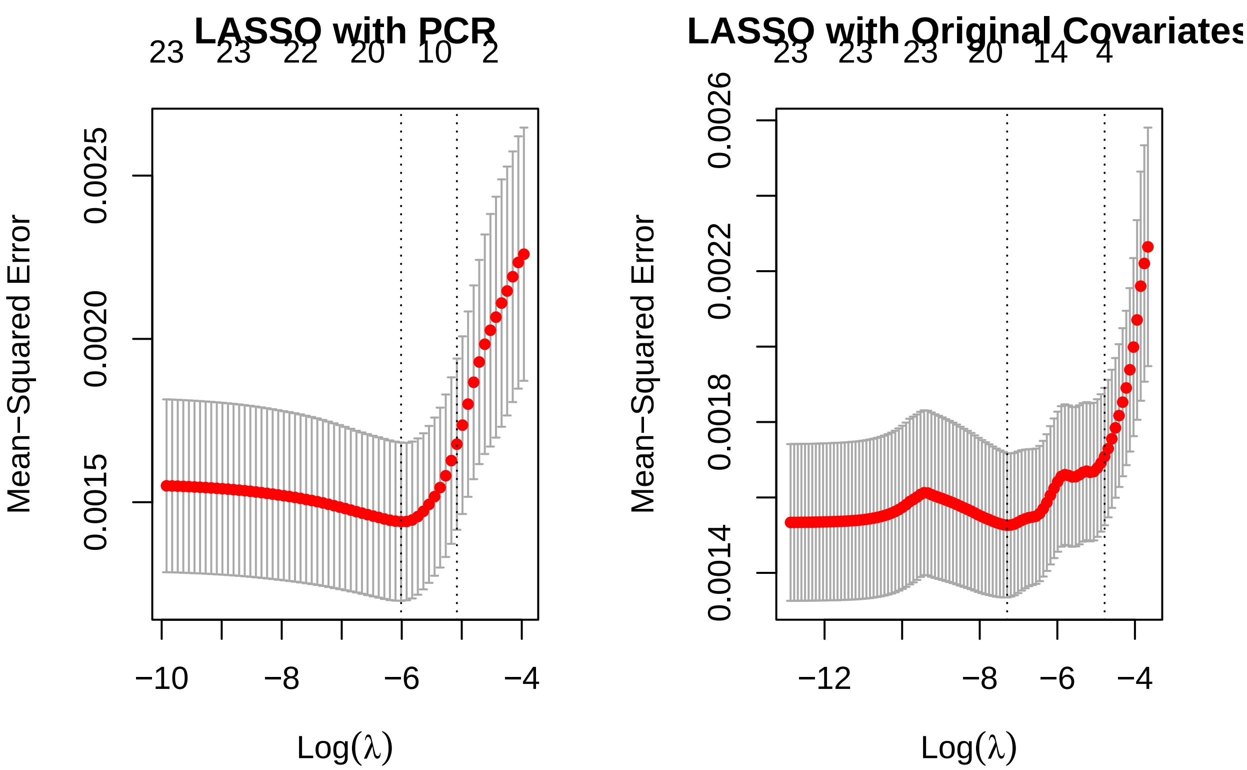
```r
fx_matrix <- as.matrix(fx)

lasso_model <- cv.glmnet(x = fx_matrix, y = sp$sp500, nfolds = 20)
coef(lasso_model)
```

```
## 24 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept)  0.0009592809
## al          -0.0902500219
## bz          -0.0577613057
```

```
## ca           .
## ch           .
## dn           .
## eu           .
## hk           .
## in           .
## jp           .
## ko           .
## ma           .
## mx          -0.4962779381
## no           .
## nz           .
## sd          -0.0988220792
## sf           .
## si           .
## sl           .
## sz           .
## ta           .
## th           .
## uk           .
## vz           .
```

```r
par(mfrow = c(1,2))
plot(sp_model_lasso, main = "LASSO with PCR")
plot(lasso_model, main = "LASSO with Original Covariates")
```



Lasso: The coefficients obtained from the lasso regression model differ significantly from those obtained from principal component regression (PCR). While PCR considers all principal components in the model, lasso regression applies a regularization technique that penalizes the absolute size of the coefficients, effectively shrinking some coefficients towards zero and inducing sparsity in the model.

PCR: In contrast to PCR, where coefficients are assigned to each principal component regardless of their magnitude, lasso regression automatically selects a subset of features that are most relevant for predicting the target variable. This results in a sparse coefficient matrix, as seen in the output, where many coefficients are effectively set to zero.

As a result, while both lasso and PCR are techniques for regression modeling, they differ in their approach to feature selection and model complexity. Lasso explicitly selects a subset of features by shrinking coefficients, leading to sparser models, while PCR reduces dimensionality by creating new features based on linear combinations of the original features, potentially resulting in more complex models with all principal components retained.