Solutions to Selected Computer Lab Problems and Exercises in Chapter 18 of *Statistics and Data Analysis for Financial Engineering, 2nd ed.* by David Ruppert and David S. Matteson

Problem 4 (a) (i) `sdev` is a vector contain the square roots of the eigenvalues of the covariance matrix. This fact can be verified by using `eigen` to find the eigenvalues. See the `R` output below. Apparently, `princomp()` uses n (the sample size) rather than $n-1$ as the divisor in the sample covariance matrix. To compensate, I used `cov(delta_yield)*((n-1)/n)` in the second line of the `R` code. With the change, the square roots of the eigenvalues are equal to the `stdev` from `princomp()`.

```
> options(digits=3,width=60)
> eig = eigen(cov(delta_yield)*((n-1)/n))
> sqrt(eig$values)
 [1] 0.17688 0.05333 0.03200 0.01444 0.01103 0.00885 0.00809
 [8] 0.00636 0.00428 0.00374 0.00197
> pca_del$sdev
 Comp.1  Comp.2  Comp.3  Comp.4  Comp.5  Comp.6  Comp.7
0.17688 0.05333 0.03200 0.01444 0.01103 0.00885 0.00809
 Comp.8  Comp.9 Comp.10 Comp.11
0.00636 0.00428 0.00374 0.00197
```

(ii) `loadings` are the eigenvectors of the covariance matrix. Below is a check that the first eigenvector is equal to the first column of `loadings`. The other eigenvector can be checked in the same way. Eigenvectors are determined only up to sign changes, so you might find that some eigenvectors from `eigen()` and the corresponding ones from `princomp()` have different signs.

```
> pca_del$loadings[,1]
  X1mon   X2mon   X3mon   X4mon   X5mon X5.5mon X6.5mon
-0.0646 -0.2152 -0.2972 -0.3220 -0.3350 -0.3341 -0.3322
X7.5mon X8.5mon X9.5mon     NA.
-0.3338 -0.3299 -0.3206 -0.3167
> eig$vector[,1]
 [1] -0.0646 -0.2152 -0.2972 -0.3220 -0.3350 -0.3341 -0.3322
 [8] -0.3338 -0.3299 -0.3206 -0.3167
```

(iii) `center` is the mean vector. This is verified below.

```
> pca_del$center
    X1mon     X2mon     X3mon     X4mon     X5mon   X5.5mon
 0.000317 -0.000574 -0.000852 -0.000985 -0.001051 -0.001079
  X6.5mon   X7.5mon   X8.5mon   X9.5mon       NA.
-0.001077 -0.001087 -0.001160 -0.001232 -0.001237
> colMeans(delta_yield)
    X1mon     X2mon     X3mon     X4mon     X5mon   X5.5mon
 0.000317 -0.000574 -0.000852 -0.000985 -0.001051 -0.001079
  X6.5mon   X7.5mon   X8.5mon   X9.5mon       NA.
-0.001077 -0.001087 -0.001160 -0.001232 -0.001237
```

(iv) `scores` contains the projections of the yields minus their means onto the eigenvectors. See the `R` output which only checks the projections onto the first eigenvectors.

```
> mu = colMeans(delta_yield)
> pca_del$scores[1,]
   Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6
 0.190005  0.035301  0.043083 -0.017083  0.001045  0.000545
   Comp.7    Comp.8    Comp.9   Comp.10   Comp.11
 0.010066 -0.009695 -0.004194  0.000811 -0.000014
> S = as.matrix(delta_yield-mu) %*% eig$vector
> S[1,]
 [1]  0.190005  0.035301  0.043083 -0.017083  0.001045
 [6]  0.000545 -0.010066 -0.009695 -0.004194 -0.000811
[11]  0.000014
```

Problem 4 (d) 2 since the first two principal components have 95.56% of the variance as seen in the cumulative proportions.

```
> summary(pca_del)
Importance of components:
                       Comp.1 Comp.2 Comp.3  Comp.4  Comp.5
Standard deviation      0.177 0.0534 0.0320 0.01446 0.01108
Proportion of Variance  0.876 0.0797 0.0287 0.00585 0.00344
Cumulative Proportion   0.876 0.9556 0.9844 0.99021 0.99365
                        Comp.6  Comp.7  Comp.8   Comp.9
Standard deviation     0.00892 0.00818 0.00646 0.004346
Proportion of Variance 0.00223 0.00187 0.00117 0.000529
Cumulative Proportion  0.99588 0.99775 0.99892 0.999447
                        Comp.10  Comp.11
Standard deviation     0.003910 0.002111
Proportion of Variance 0.000428 0.000125
Cumulative Proportion  0.999875 1.000000
```

Problem 5. The output below is used for this problem. To save space, some of the output that was not needed has been deleted. The intercept's p-value is below 0.025 for General motors (GM) and Ford, but not for United Technologies Incorporatted (UTX) and Merck. Of course, a p-value only shows statistical significance, not the size of an effect. However, estimated intercepts for GM and Ford are -0.23 and -0.18 and these are reasonably large in magnitude. Since they are negative, this suggests than these two stocks were overpriced; see Section 17.6.3 for discussion.

```
> fit1 = lm(as.matrix(stocks_diff)~FF_data$Mkt.RF)
> summary(fit1)
Response GM :

Call:
lm(formula = GM ~ FF_data$Mkt.RF)

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     -0.2290     0.0865   -2.65   0.0083 **
FF_data$Mkt.RF   1.2500     0.1273    9.82   <2e-16 ***
---
Residual standard error: 1.94 on 501 degrees of freedom
```

```
Multiple R-squared:  0.161,Adjusted R-squared:  0.16
F-statistic: 96.4 on 1 and 501 DF,  p-value: <2e-16


Response Ford :

Call:
lm(formula = Ford ~ FF_data$Mkt.RF)

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     -0.1835     0.0676   -2.72   0.0069 **
FF_data$Mkt.RF   1.3195     0.0995   13.26   <2e-16 ***
---
Residual standard error: 1.51 on 501 degrees of freedom
Multiple R-squared:  0.26,Adjusted R-squared:  0.258
F-statistic:  176 on 1 and 501 DF,  p-value: <2e-16


Response UTX :

Call:
lm(formula = UTX ~ FF_data$Mkt.RF)

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     0.00219    0.03883    0.06     0.95
FF_data$Mkt.RF  0.91932    0.05718   16.08   <2e-16 ***
---
Residual standard error: 0.87 on 501 degrees of freedom
Multiple R-squared:  0.34,Adjusted R-squared:  0.339
F-statistic:  258 on 1 and 501 DF,  p-value: <2e-16


Response Merck :

Call:
lm(formula = Merck ~ FF_data$Mkt.RF)

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     -0.0888     0.0920   -0.97     0.33
FF_data$Mkt.RF   0.6255     0.1355    4.62    5e-06 ***
---
Residual standard error: 2.06 on 501 degrees of freedom
Multiple R-squared:  0.0408,Adjusted R-squared:  0.0389
F-statistic: 21.3 on 1 and 501 DF,  p-value: 4.99e-06
```

Problem 6. The correlation matrix is below. All correlations are reasonably close to 0 (less than 0.1 in magnitude) except the correlation between GM and Ford residuals. That correlation is 0.52 and has a very small p-value.

The correlation between GM and Merck residuals is -0.0878 and is statistically significant at 0.05 but might be too small to be of practical significance.

```
> cor(residuals(fit1))
           GM     Ford     UTX    Merck
GM     1.0000  0.52016 -0.0100 -0.08776
Ford   0.5202  1.00000 -0.0238 -0.00958
UTX   -0.0100 -0.02376  1.0000 -0.00550
Merck -0.0878 -0.00958 -0.0055  1.0000
```

```
> res = residuals(fit1)
> cor.test(res[,"GM"],res[,"Ford"])

Pearson's product-moment correlation

data:  res[, "GM"] and res[, "Ford"]
t = 10, df = 500, p-value <2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.453 0.581
sample estimates:
 cor
0.52

> cor.test(res[,"GM"],res[,"UTX"])

Pearson's product-moment correlation

data:  res[, "GM"] and res[, "UTX"]
t = -0.2, df = 500, p-value = 0.8
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.0974  0.0775
sample estimates:
  cor
-0.01

> cor.test(res[,"GM"],res[,"Merck"])

Pearson's product-moment correlation

data:  res[, "GM"] and res[, "Merck"]
t = -2, df = 500, p-value = 0.05
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.17385 -0.00033
sample estimates:
    cor
-0.0878

> cor.test(res[,"Ford"],res[,"UTX"])

Pearson's product-moment correlation

data:  res[, "Ford"] and res[, "UTX"]
t = -0.5, df = 500, p-value = 0.6
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.1110  0.0638
sample estimates:
```

```
    cor
-0.0238

> cor.test(res[,"Ford"],res[,"Merck"])

Pearson's product-moment correlation

data:  res[, "Ford"] and res[, "Merck"]
t = -0.2, df = 500, p-value = 0.8
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.0969  0.0779
sample estimates:
     cor
-0.00958

> cor.test(res[,"UTX"],res[,"Merck"])

Pearson's product-moment correlation

data:  res[, "UTX"] and res[, "Merck"]
t = -0.1, df = 500, p-value = 0.9
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.0929  0.0820
sample estimates:
    cor
-0.0055
```

Problem 7. We see below that the estimated covariance matrix using the CAPM is similar to the sample covariance matrix, with the exception of the covariance between GM and Ford. Since these two stocks have a high residual correlation and the CAPM assumes that the residual correlation is 0, it is not surprising that the CAPM estimated covariance matrix severely underestimates the correlation between GM and Ford.

```
> attach(FF_data)
The following objects are masked from FF_data (pos = 3):

    date, HML, Mkt.RF, RF, SMB

The following objects are masked from FF_data (pos = 4):

    date, HML, Mkt.RF, RF, SMB

> sigF = var(Mkt.RF)
> bbeta = as.matrix(fit1$coef)
```

```
> bbeta = bbeta[-1,]  #  delete intercepts so bbeta has the four slopes
> n=dim(stocks_diff)[1]
> sigeps =  as.matrix((var(as.matrix(res))))
> sigeps = diag(as.matrix(sigeps))
> sigeps = diag(sigeps,nrow=4)
> cov_equities = sigF* bbeta %*% t(bbeta) + sigeps
> cov_equities
        GM  Ford   UTX Merck
[1,] 4.464 0.761 0.530 0.360
[2,] 0.761 3.090 0.559 0.381
[3,] 0.530 0.559 1.145 0.265
[4,] 0.360 0.381 0.265 4.423
> cov(stocks_diff)
          GM  Ford   UTX  Merck
GM    4.4641 2.282 0.513 0.0108
Ford  2.2825 3.090 0.528 0.3507
UTX   0.5130 0.528 1.145 0.2553
Merck 0.0108 0.351 0.255 4.4228
```

Problem 13. The factor loadings and uniqueness are in the output below. We see that the uniquenesses for Ford and GM are 0.423 and 0.399, respectively.

```
Uniquenesses:
  GM_AC    F_AC  UTX_AC  CAT_AC  MRK_AC  PFE_AC  IBM_AC MSFT_AC
  0.399   0.423   0.718   0.714   0.519   0.410   0.760   0.749

Loadings:
        Factor1 Factor2
GM_AC    0.693  -0.348
F_AC     0.692  -0.313
UTX_AC   0.531
CAT_AC   0.529
MRK_AC   0.551   0.421
PFE_AC   0.574   0.511
IBM_AC   0.490
MSFT_AC  0.499
```

Problem 14. The results of the likelihood ratio tests below strongly suggest that there are more than two factors. It seems that four factors are sufficient, but not 3.

```
Test of the hypothesis that 2 factors are sufficient.
The chi square statistic is 564.66 on 13 degrees of freedom.
The p-value is 2.6e-112

Test of the hypothesis that 3 factors are sufficient.
The chi square statistic is 162.29 on 7 degrees of freedom.
The p-value is 1.06e-31
```

```
Test of the hypothesis that 4 factors are sufficient.
The chi square statistic is 0.3 on 2 degrees of freedom.
The p-value is 0.86
```

Problem 15. The output below contains the estimated correlation matrix using the factor model after line 4. For comparison, the sample correlation matrix is also printed.

```
> loadings = matrix(as.numeric(loadings(fact)),ncol=2)
> unique = as.numeric(fact$unique)
> options(digits=2)
> loadings %*% t(loadings) + diag(unique)
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
[1,] 1.00 0.59 0.38 0.39 0.24 0.22 0.34 0.33
[2,] 0.59 1.00 0.38 0.39 0.25 0.24 0.34 0.33
[3,] 0.38 0.38 1.00 0.28 0.28 0.29 0.26 0.26
[4,] 0.39 0.39 0.28 1.00 0.26 0.26 0.26 0.26
[5,] 0.24 0.25 0.28 0.26 1.00 0.53 0.27 0.29
[6,] 0.22 0.24 0.29 0.26 0.53 1.00 0.28 0.31
[7,] 0.34 0.34 0.26 0.26 0.27 0.28 1.00 0.24
[8,] 0.33 0.33 0.26 0.26 0.29 0.31 0.24 1.00

> cor(stocks_returns)
        GM_AC F_AC UTX_AC CAT_AC MRK_AC PFE_AC IBM_AC MSFT_AC
GM_AC    1.00 0.62   0.35   0.36   0.25   0.23   0.32    0.31
F_AC     0.62 1.00   0.35   0.37   0.26   0.25   0.30    0.30
UTX_AC   0.35 0.35   1.00   0.40   0.26   0.28   0.29    0.28
CAT_AC   0.36 0.37   0.40   1.00   0.24   0.25   0.30    0.29
MRK_AC   0.25 0.26   0.26   0.24   1.00   0.55   0.24    0.27
PFE_AC   0.23 0.25   0.28   0.25   0.55   1.00   0.26    0.29
IBM_AC   0.32 0.30   0.29   0.30   0.24   0.26   1.00    0.41
MSFT_AC  0.31 0.30   0.28   0.29   0.27   0.29   0.41    1.00
```

Exercise 4. The eigenvectors are below. The first eigenvector has all positive weights so is like a market index. The second eigenvector has negative weights for mining and gold and the other weights are positive so it contrast the gold and mining funds with the other funds.

```
> equityFunds = read.csv("equityFunds.csv")
> pcaEq = prcomp(equityFunds[,2:9])

> pcaEq$rotation
        PC1    PC2     PC3    PC4     PC5     PC6    PC7    PC8
EASTEU 0.36  0.374 -0.4594 -0.721  0.0013  0.0042 -0.020  0.031
LATAM  0.41  0.259 -0.2600  0.514  0.5040  0.3288  0.070  0.261
CHINA  0.17  0.171  0.0749  0.116  0.1074 -0.8609  0.365  0.197
INDIA  0.25  0.541  0.7693 -0.082 -0.1055  0.1770 -0.045 -0.012
```

ENERGY	0.24	0.123	-0.2517	0.342	-0.8439	0.0717	0.153	0.076
MINING	0.48	-0.162	-0.0088	0.126	0.0769	-0.1190	-0.032	-0.839
GOLD	0.56	-0.653	0.2402	-0.199	-0.0421	0.0680	0.023	0.400
WATER	0.11	0.063	-0.0532	0.154	-0.0580	-0.3091	-0.914	0.151