SMF Solution 3. 8.11.2013

Q1.
(i) First, load the data into R: in order to do so, first choose the directory where your file is saved by doing "File" and "Change Dir". The following code loads the data of the file into *rates*.

```
rates<-read.csv(file="treasury_02-01-03_to_11-04-13.csv",header=TRUE)
```

(ii) We now need to deal with the missing data: here we replace the missing values by the average between the day before and the day after.

```
#dealing with missing data:take the average between the previous and next value
for (i in 1:length(rates[,1])){
if (is.na(rates[i,1])){
for (j in 1:length(rates)){
rates[i,j] = (rates[i-1,j]+ rates[i+1,j])/2
}
}
}
```

(iii) Now, we want to take the centred returns

```
#take the returns:
#creates a matrix of the right size:
returns<-matrix(ncol=length(rates),nrow=(length(rates[,1])-1))
#and fill with corresponding data
for (i in 1:length(rates)){
for (j in 2:length(rates[,1])){
returns[j-1,i]<-rates[j,i]-rates[j-1,i]
}
}
#take the centred returns
centred_returns <- returns-colMeans(returns)
```

The preliminary treatment having been done, we can now perform the proper PCA. One can indeed see that the data is quite correlated (Fig. 1).

```
pairs(centred_returns, main="Centred returns data 02/02/2003-11/04/2013")
```

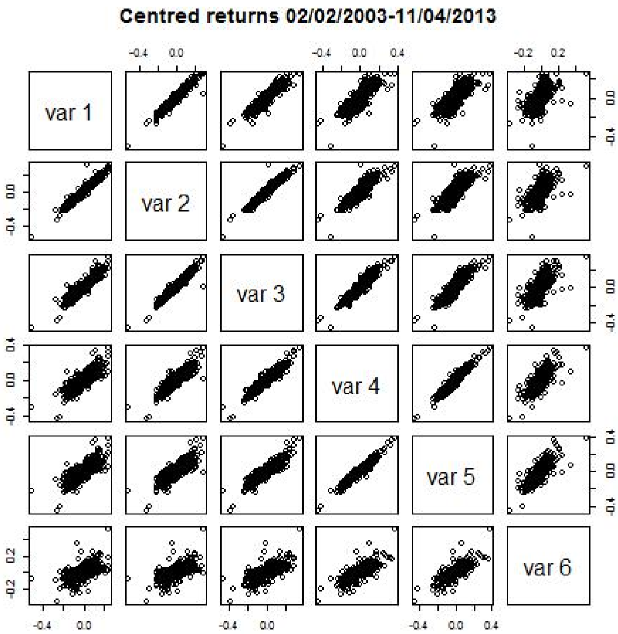

Figure 1: Pairwise plotting of the returns for maturities in decreasing order 10, 7, 5, 3, 2 and 1 year

(iv) We then use the *princomp* command to run a PCA (the following syntax runs the PCA with the correlation matrix, see at the end for a note on the use of the covariance matrix instead)

```
components<-princomp(centred_returns, cor=T)
summary(components)
```

The results are

```
 Importance of components:
                          Comp.1    Comp.2     Comp.3      Comp.4      Comp.5
Standard deviation     2.2625286 0.7875452 0.42479154 0.196996416 0.161922917
Proportion of Variance 0.8531726 0.1033713 0.03007464 0.006467931 0.004369838
Cumulative Proportion  0.8531726 0.9565439 0.98661854 0.993086469 0.997456307
                            Comp.6
Standard deviation     0.123540105
Proportion of Variance 0.002543693
Cumulative Proportion  1.000000000
```

Or, in graphic format

```
plot(components)
```

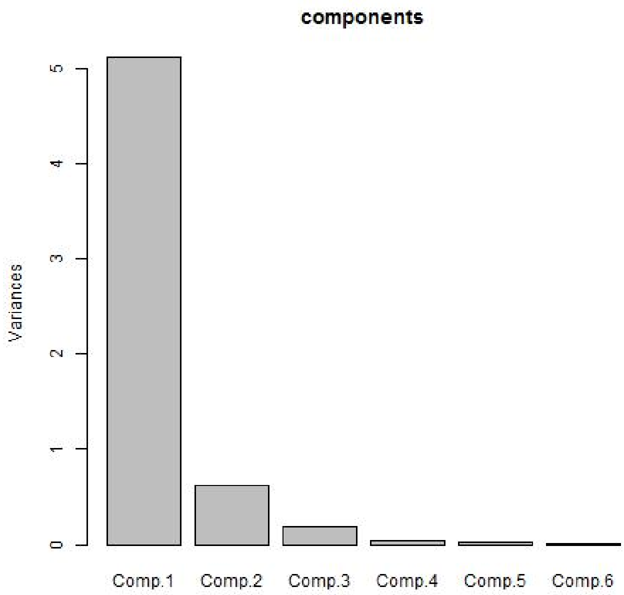

Figure 2: Variance explained by the components

One can see that the biggest component itself accounts for about 85 percent of the variance, the second one for about 10 percent and the third one for about 3 percent. Looking at the cumulative proportion, 98.66 percent of the variance are explained by the first 3 components. By typing

```
print(loadings(components),cutoff=0)
```

one is able to obtain the expression of the components (the eigenvectors):

```
Loadings:
     Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6
[1,] -0.404 -0.441  0.402 -0.400 -0.511  0.246
[2,] -0.422 -0.338  0.199  0.062  0.389 -0.716
[3,] -0.433 -0.189 -0.052  0.334  0.509  0.635
[4,] -0.430  0.077 -0.408  0.557 -0.556 -0.151
[5,] -0.415  0.271 -0.569 -0.642  0.138 -0.022
[6,] -0.339  0.759  0.553  0.051  0.011  0.006
```

For information, if one does not use the *cutoff* parameter, one will see blank space when they want to display the data. These blank spaces indicate values close to zero that are non-zero. Graphical display of the first three principal components is given by:

```
barplot(t(components$loadings[,1:3]),beside=TRUE)
```

The **barplot** command allows you to print bar plots. If *beside* is set as true, the columns are plotted next to one another. Finally, notice the use of the transpose command to have the bars grouped by the coordinate number (i.e. first coordinate of principal components 1,2 and 3 together, etc.) Without the transpose, one gets the coordinates grouped by components rather than the coordinate number. But this is purely a matter of taste. The display of (with the transpose) corresponds to the following figure.

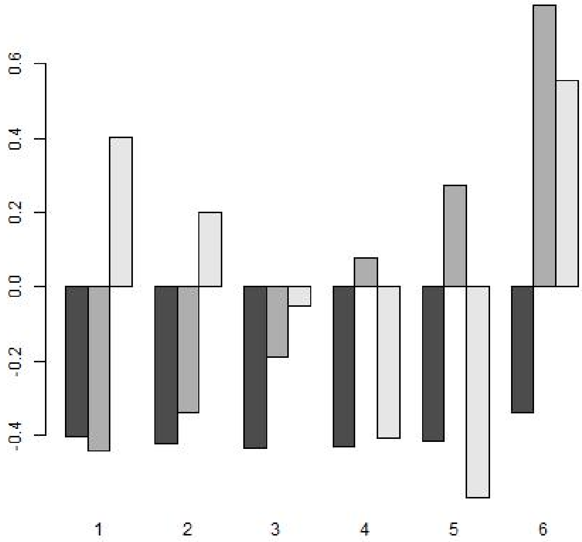

Figure 3: Values of the first three components across the maturities–the longest ones coming first. The darkest is the first component and the lightest is the third one

(v) One can notice the following effects (cf Lai-Xing):

1. The first component can be interpreted as a *parallel shift component*. The factor loadings are roughly constant among maturities, meaning the change in the rate for a maturity is roughly the same for other maturities. Consequently, the first factor accounts for the “average rate”
2. The second component corresponds to a *tilt*. The factor loadings of the second component have a monotonic change with maturities:changes in long-maturity and short maturity have opposite signs. The second factor consequently accounts for the “slope” across maturities.
3. The third component is the *curvature*. The factor loadings of the third component are different for the mid-term rates and the average of long- and short-term rates, revealing a curvature resembling the convex shape of the relationship between the rates and their maturities.

One last note: the PCA can be carried out with the covariance matrix instead by simply using

```
princomp(components)
```

instead of

```
princomp(components, cor=T)
```

The results are expected to be a priori different.

NHB/PMBF