Multivariate Statistics – Homework 3

Note: Solve the tasks below using **R** and upload your solutions until Friday, the 14th of May.

This exercise deals with *principal component analysis*. For the first part, we take the data set on mammal's milk from the **R** package **flexclust** again . . .

1. (2 points) Load **flexclust** and the data again

```
require(flexclust)
data(milk)
```

   to get access to the `milk` data set on the mammal's milk of 25 animals. Have a look at the help page and the summary of the data again. Are there missing values? What is the unit of the variables? What is their type?

2. (2 points) Are the variables directly comparabe or should the data be scaled? Can the data matrix directly be used in pca? Compute means and standard deviations for each variable.

3. (2 points) Apply `prcomp` to the data setting argument `scale=FALSE` and `center=TRUE`. Give the loadings matrix and interpret it.

4. (2 points) Give the score matrix and interpret it. What is the meaning of these values?

5. (2 points) Apply the summary method to the pca result. How much variance is explained by the first principal component? How many PCs would you choose?

6. (3 points) Provide the score plot and a loadings plot of the data using the first two PCs.

7. (2 points) Now provide a biplot of the pca and interpret the results.

8. (3 points) Finally, cluster the data into 4 clusters using kmeans. Visualize your cluster solution in a biplot by setting the `biplot` argument `xlabs` to the vector of cluster membership of your cluster solution.

9. (2 points) Now select one of your own jpegs and read the photo into R using package **jpeg**. What is the resolution of your photo? What is the class of `photo`?

```
require(jpeg)

photo <- readJPEG("my_photo.jpg")
nrow(photo)
ncol(photo)
```

   Create separate matrices for every RGB color scale and perform pca. *Notice:* As this example is focused on image compression and not description or interpretation of the variables, the data does not require centering (subtracting the variable means from the respective observation vectors), and the center argument is set to FALSE. If the argument is not set to FALSE, the returned image will not have the right RGB values due to having their respective means subtracted from each pixel color vector (try it out).

```
r <- photo[,,1]
g <- photo[,,2]
b <- photo[,,3]
r.pca <- prcomp(r, center = F)
g.pca <- prcomp(g, center = F)
b.pca <- prcomp(b, center = F)
rgb.pca <- list(r.pca, g.pca, b.pca)
```

10. (3 points) In order to get the compressed RGB matrices you have to multiply the score matrix `x` with the loadings matrix `rotation`. Try it for different numbers of principal components. How many PCs do you need to get satisfying picture quality?

```
for (i in seq.int(3, round(nrow(photo)/2), length.out = 5)) {
  pca.img <- sapply(rgb.pca, function(j) {
    compressed.img <- j$x[,1:i] %*% t(j$rotation[,1:i])
  }, simplify = 'array')
  writeJPEG(pca.img, paste('PIC_', round(i,0), '_components.jpg', sep = ''),
            quality=1)
}
```

11. (2 points) Finally, include two versions of your picture with too few and just right numbers of components in this document using the Rmarkdown command `![some caption](PIC_3_components.jpg)`. *Please notice:* The maximum file size for upload to BOKUlearn is 250MB.