Bio-Data Science Task 2 Report

Summary: Now that I know the basic usage of R, here, I will develop some R codes for solving intermediate to complex scientific problems under a microbiology theme.

I will reproduce the following tutorial: https://bioinfo4all.wordpress.com/2021/01/31/tutorial-6-how-to-do-principal-component-analysis-pca-in-r/. I intend to improve my R and Data Analytics skills by this task.

During microbial growth, there are 3 key phases; lag phase, exponential phase and stationary phase. For the 10 samples I want to determine the time when the bacteria enters its stationary phase.

Explanations of codes and visualizations are provided below:

1. I will import my already downloaded .csv file in R

```
microbial_stationary_phase.csv <- read.csv(file.choose())
microbial_stationary_phase.csv
```

2. Let me read the table

```
data <- read.csv("microbial_stationary_phase.csv", row.names = 1)
```

3. Now I will need to install and load two R package which will allow me to do PCA in R. To install:

```
install.packages(c("factoextra", "FactoMineR"))
```

4. To load:

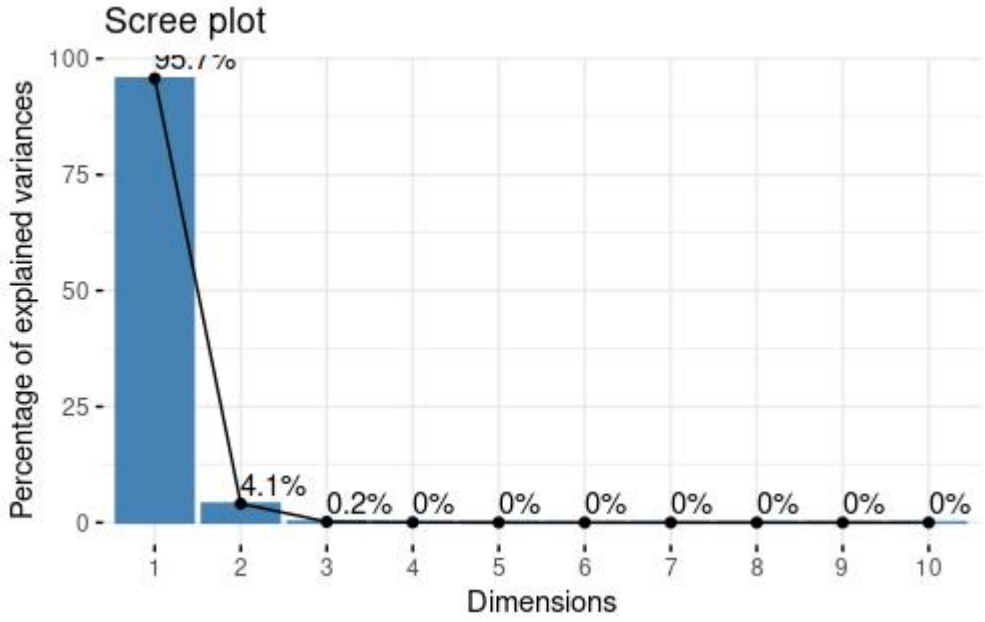
```
library("factoextra")
library("FactoMineR")
```

5. Let me create the Principal Component Table. Note that: scale.unit = TRUE is an argument to standardize the values.

```
pca.data <- PCA(data, scale.unit = TRUE, graph = FALSE)
```

6. To make sure that most of the data will be presented in the PCA plot, I need to use the fviz_eig() function. I will be using the table I created with PCA() function; pca.data

```
fviz_eig(pca.data, addlabels = TRUE)
```

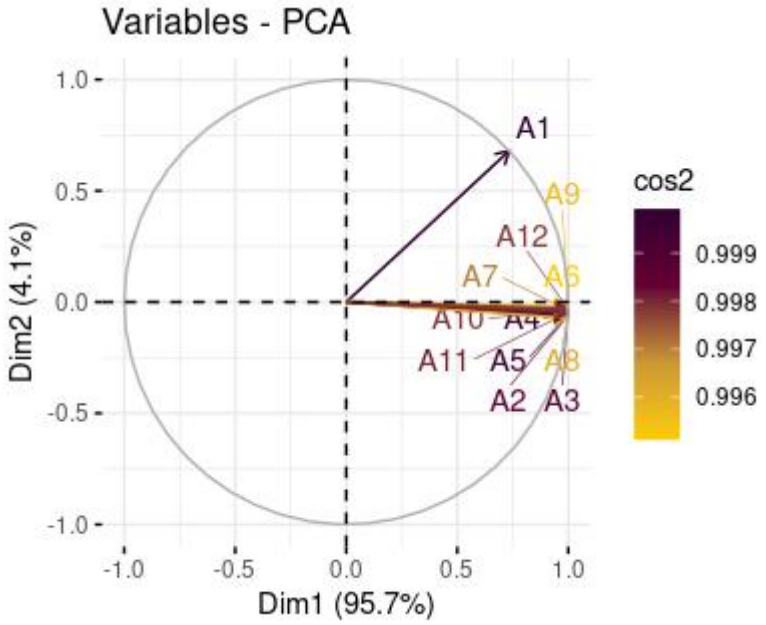


7. To avoid unlabeled data points (too many overlaps), I will increase max.overlaps

```
options(ggrepel.max.overlaps = Inf)
```

8. To understand the correlation between the samples and how they are well represented by my model I can use fviz_pca_var() function to draw a variable correlation plot by using the command below:

```
fviz_pca_var(pca.data, col.var = "cos2", gradient.cols = c("#FFCC00", "#CC9933", "#660033", "#330033"), repel = TRUE)
```



9. We can see below the Bacteria cell type are next to each other, this means they are correlated to each other. Here we do not have any negative correlation between the variables but if there was, the arrow will be on the opposite sides. One last thing, since the arrow is close to the circle (long), it means the variable is well represented.

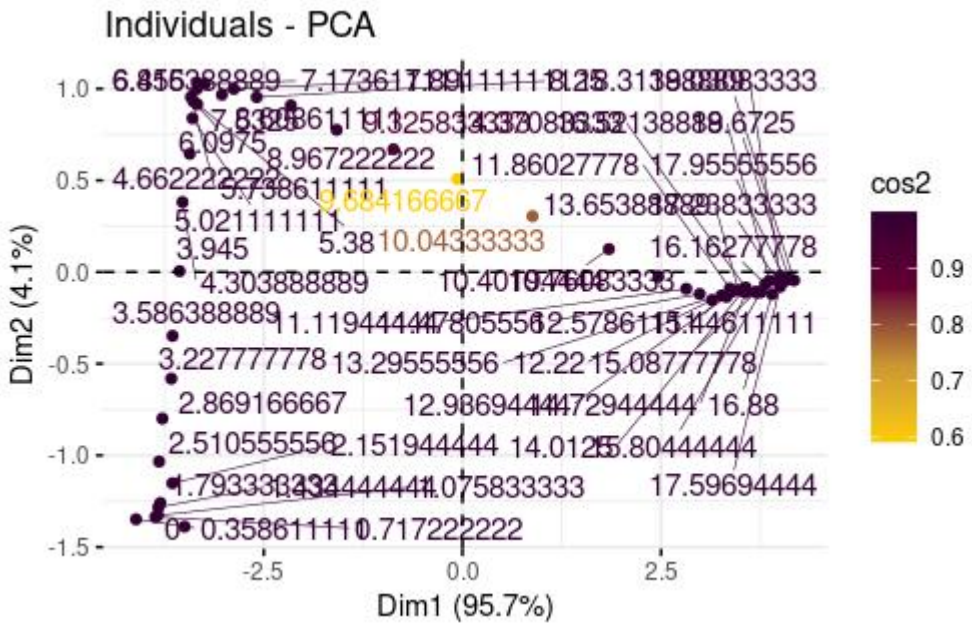
When doing a PCA plot we have the option to plot the Bacteria cell types or the Time. It gives us the opportunity to look at the data from different angles which could enable us to find a pattern or a marker.

Let me start by plotting the Bacteria cell types first. To do that I need to used the PCA() function again and use t() function to flip our table, so that I can put the cell types as rows.

```
pca.data <- PCA(t(data, scale.unit = TRUE, graph = FALSE)
```

10. Then I will use the viz_pca_ind() function for the visualization as shown below:

```
fviz_pca_ind(pca.data, col.ind = "cos2", gradient.cols = c("#FFCC00", "#CC9933", "#660033", "#330033"), repel = TRUE)
```



11. To add labels to the PCA plot I can use ggpubr package. First I need to install and load the package

```
install.packages('devtools')
library(devtools)
install_github("kassambara/ggpubr")
library(ggpubr)
```

12. Then I need to assign the previous command to a

```
a <- fviz_pca_ind(pca.data, col.ind = "cos2", gradient.cols = c("#FFCC00", "#CC9933", "#660033", "#330033"), repel = TRUE)
```

13. Now I can use ggpar() function to add labels

```
ggpar(a, title = "Principal Component Analysis", xlab = "PC1", ylab = "PC2", legend.title = "Cos2", legend.position = "top", ggtheme = theme_minimal())
```

14. Now let me plot the Time instead of the Bacteria cell types. I will use the PCA() function

```
pca.data <- PCA(data, scale.unit = TRUE,ncp = 2, graph = FALSE)
```

15. To color the time that enters the stationary phase in the PCA plot I will be using the first column (A1): This was the exponential phase because immediately after which the bacteria cells entered the stationary phase. First I need to convert the column to a factor by the following command

```
data$A1 <- as.factor(data$A1)
```

16. Next, for the coloring palette I will use the commands below.

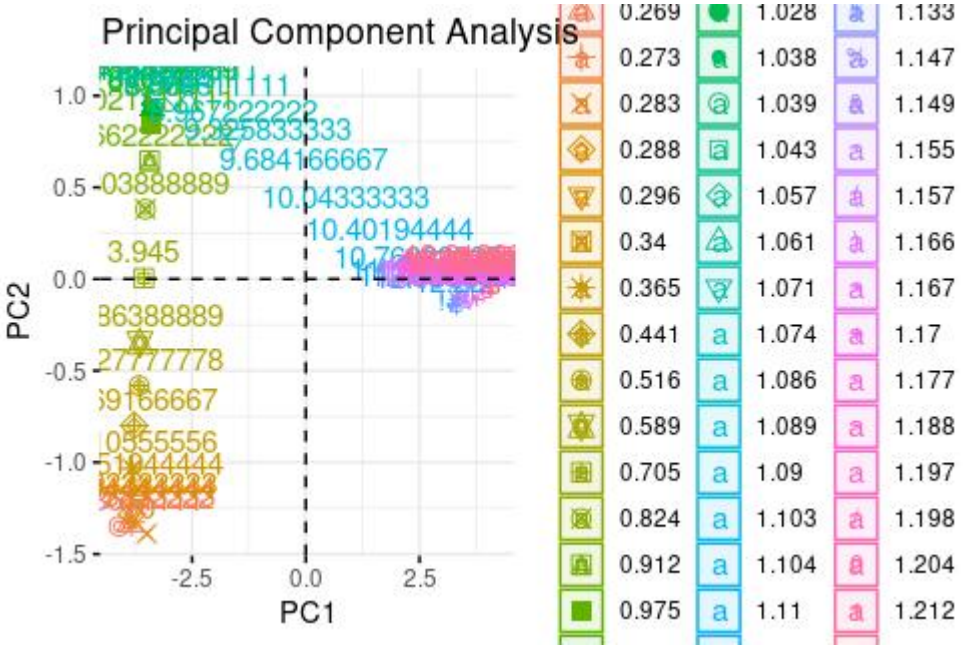
```
install.packages("RColorBrewer")
library(RColorBrewer)
```

17. I will use fviz_pca_ind() function to create the PCA plot and assign it to “a” as I did previously.

```
a <- fviz_pca_ind(pca.data, col.ind = data$A1, addEllipses = TRUE)
```

18. Then I will use the ggpar() function to add labels:

```
ggpar(a, title = "Principal Component Analysis", xlab = "PC1", ylab = "PC2", legend.title = "Bacteria Stationary Phase Time", legend.position = "top", ggtheme = theme_minimal())
```


Principal Component Analysis
PC2
PC1
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.5
0.0
2.5
3.945
9.684166667
10.04333333
10.40194444
0.269
1.028
1.133
0.273
1.038
1.147
0.283
1.039
1.149
0.288
1.043
1.155
0.296
1.057
1.157
0.34
1.061
1.166
0.365
1.071
1.167
0.441
1.074
1.17
0.516
1.086
1.177
0.589
1.089
1.188
0.705
1.09
1.197
0.824
1.103
1.198
0.912
1.104
1.204
0.975
1.11
1.212