7144CEM
PRINCIPLES OF DATA SCIENCE

Group Assignment (Task 1)

Group 19

Sujan Tumbaraguddi.....14194733
Vineet Jain.....14234121
Ramana kulanthaivelu.....14231913
Vishnu Priya.....14368415

Table of Contents

Contributions

S.no.	Title	Code	Interpretation & Evaluation	Validation
1.	Principal Component Analysis (Q No. 01)	Vineet Jain	Vineet Jain, Ramana kulanthaivelu, Vishnu Priya	Sujan Tumbaraguddi, Ramana kulanthaivelu, Vishnu Priya
2.	Factor Analysis (Q No. 02)	Sujan Tumbaraguddi	Sujan Tumbaraguddi, Vineet Jain	Vineet Jain, Ramana kulanthaivelu, Vishnu Priya
3.	Cluster Analysis (Q No. 03)	Ramana kulanthaivelu, Vishnu Priya	Ramana kulanthaivelu, Vishnu Priya, Sujan Tumbaraguddi	Sujan Tumbaraguddi, Vineet Jain
4.	Summary and Insights (Q No. 04)		Sujan Tumbaraguddi, Ramana kulanthaivelu, Vishnu Priya, Vineet Jain	Sujan Tumbaraguddi, Ramana kulanthaivelu, Vishnu Priya, Vineet Jain

Multivariate Statistical Analysis (MLO2)

```
# R Studio

# 7144CEM Principles of Data Science

# The original dataset has 768 rows (500 'no diabetes' and 268 'diabetes').
# We have taken a subset of 200 'no diabetes' cases and 200 'diabetes' cases.
# 400 observations and 8 variables.

#########################################################################
# GROUP TASK -- Multivariate Statistical Analysis (MLO2)
#########################################################################


# Install the tidyverse package if not already installed
#install.packages("tidyverse")
library(tidyverse)
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.3     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ ggplot2   3.4.4     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.0
## ✓ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts
to become errors
#############################################################
# Uncomment below code when extracting the data sample
#############################################################

# Setting the seed to '733'
#set.seed(733)

# Read Pima Indians Diabetes Dataset as csv file
#data = read_csv('Pima_Indians_Diabetes_Dataset1.csv')
#View(data)

# To take stratified random sampling based on the 'Target' variable
#strata_sample = data %>%
#  group_by(Target) %>%
#  slice_sample(n=200) %>%
#  ungroup()
#View(strata_sample)
```

```
# Save the obtained stratified sample
#write_csv(strata_sample, 'PI_Diabetes_Datasample.csv')

# Read PI_Diabetes_Datasample Dataset as csv file and convert to Tibble
strata_sample <- as_tibble(read_csv('PI_Diabetes_Datasample.csv'))
## Rows: 400 Columns: 9
## ── Column specification ────────────────────────────────────────────────────
## Delimiter: ","
## dbl (9): Number of times pregnant, Plasma glucose concentration, Diastolic b...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

2.1 Principal Component Analysis (PCA)

```
##########Principal Component Analysis###############

# Q.No. 01

# Assign labels "No Diabetes" and "Diabetes" to a new 'class' variable based on the
'Target' variable
pca_sample <- strata_sample
pca_sample$class <- ifelse(pca_sample$Target == 0, "No Diabetes", "Diabetes")
strata_sub = pca_sample[,1:8]
```

In my “pca_sample” data frame, I have added a new column named “class” using this code. Based on the values found in the “Target” column, this column's values are calculated. “class” is set to “Not Diabetes” if “Target” is zero, and “Diabetes” if not. Also, Using the first eight columns of the “pca_sample” dataset, the second line creates a new dataset called “strata_sub”. So that our Principal Component Analysis will make use of these columns for further analysis.

```
# 1. Principal Component Analysis on first 8 input variables
pca_results = prcomp(strata_sub, center=TRUE, scale.=TRUE)
summary(pca_results)

## Importance of components:
##                           PC1    PC2    PC3    PC4     PC5     PC6     PC7
## Standard deviation     1.4328 1.3092 1.0151 0.9320 0.85458 0.83106 0.68551
## Proportion of Variance 0.2566 0.2142 0.1288 0.1086 0.09129 0.08633 0.05874
## Cumulative Proportion  0.2566 0.4708 0.5997 0.7082 0.79951 0.88584 0.94459
##                           PC8
## Standard deviation     0.66582
## Proportion of Variance 0.05541
## Cumulative Proportion  1.00000
```

Here in the above code, I have performed PCA on the first 8 input variables using "strata_sub". From summary() function, I got PCs that are arranged in order of significance for capturing the variability in my data. They are just linear combinations of my original 8 variables.

1. The standard deviation of PC1 is the highest at 1.4328, then PC2 at 1.3092, and so forth because the principal components are arranged according to how much variation they explain. Higher value of standard deviation means that more of the total variation of the data is captured by the PC.
2. The Proportion of Variance indicates the extent to which each PC accounts for the overall variance in our data. A total of 25.66% of the variance is explained by PC1, 21.42% by PC2, and so forth.
3. The cumulative contribution of each PC to the overall variance is displayed as Cumulative Proportion. So, when PC1 and PC2 are combined, they explain a total of 47.08% of the variance, while PC1 alone only explains 25.66%.

```
# Variance explained by each principal component and cumulative percentage of variance explained
var_explained = 100*((pca_results$sd)^2)/(sum((pca_results$sd)^2))
var_explained

## [1] 25.660062 21.424046 12.881392 10.856849  9.128926  8.633209  5.874046
## [8]  5.541471

cumsum(var_explained)

## [1]  25.66006  47.08411  59.96550  70.82235  79.95127  88.58448  94.45853
## [8] 100.00000
```

In this step, I have comprehended how much of the overall variance of the data can be explained by each major component separately and how the cumulative variance explained rises with the number of components considered using the var_explained that we have discussed above.

```
# 1(a). Screeplot using ggplot
ggplot(NULL,aes(x=1:8,y=var_explained),ncp=5) +
  geom_col() +
  geom_line() +
  geom_hline(yintercept = 0, color = "gray", linetype = 2) +
  geom_vline(xintercept = 0, color = "gray", linetype = 2) +
  geom_bar(stat = "identity" ,fill = "skyblue" ) +
  geom_text(aes(label = round(var_explained, 2), vjust = -0.5)) +
  ggtitle('Scree plot - Pima Indians Diabetes (8)') +
  xlab('Principal Component (PC)') +
  ylab('Percentage Variance Explained')
```

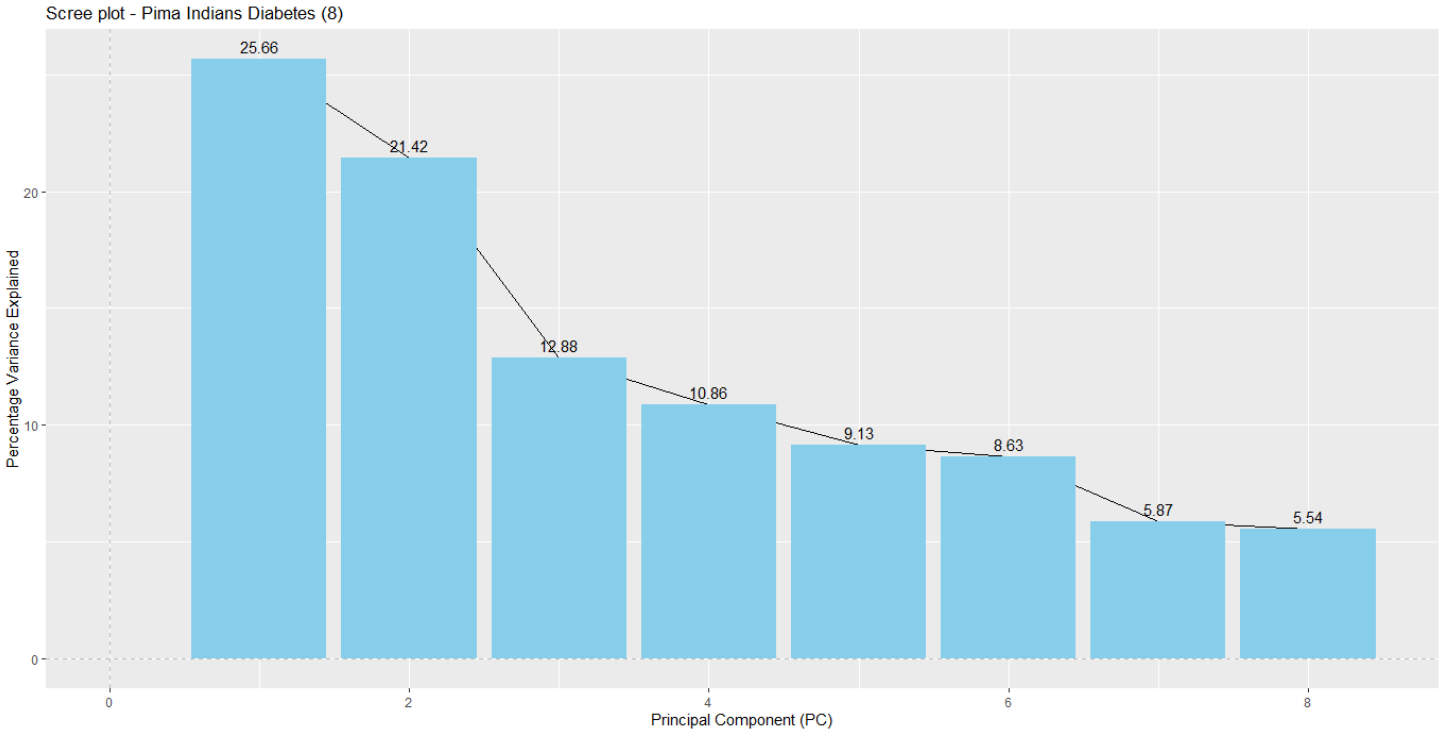

Image 01

Here each principal component's contribution to the variance in the data is shown graphically in this Scree plot (Image 01). It shows how many components to keep for your research and helps you grasp the relative relevance of each component. Within this figure, you can observe the trend of the proportion of variation explained decreasing from PC1 to PC8, with the majority of the variance described by the first five components only (almost 80%).

```
# Loadings plot for the first three principal components
loadings = as.data.frame(pca_results$rotation[,1:3])
loadings

##                                 PC1        PC2         PC3
## Number of times pregnant     0.1141424  0.5958024  0.01814440
## Plasma glucose concentration 0.3968360  0.1522477 -0.56980988
## Diastolic blood pressure     0.3470061  0.2472860  0.52068333
## Triceps skinfold thickness   0.4400407 -0.3124659  0.28999826
## 2-Hour serum insulin         0.4202667 -0.2388777 -0.38045494
## Body mass index              0.4383827 -0.1032942  0.35836853
## Diabetes pedigree function   0.3340355 -0.1375285 -0.20358049
## Age                          0.1872443  0.6135214 -0.07182189

loadings$Symbol = row.names(loadings)
loadings = gather(loadings, key='Component', value='Weight', -Symbol)
ggplot(loadings, aes(x=Symbol,y=Weight)) +
  geom_bar(stat='identity',fill = "skyblue") +
```

```
facet_grid(Component~.) +
ggtitle('Loadings for PC1-PC3')
```

Image 02

1. In Image 02, “Triceps skinfold thickness”, “2-Hour serum insulin” and “Body mass index” has a positive loading of 0.4400407, 0.4202667 and 0.4383827 respectively on PC1 which is close to 1, indicating that an increase in these variables is associated with a higher value of PC1 which indicates into a person's metabolic health. Also “Plasma glucose concentration” and “Diastolic blood pressure” have a relatively stronger positive loading of 0.3968360 and 0.3470061 on PC1.
2. In above Figure “Age” and “Number of times pregnant” has positive loadings of 0.6135214 and 0.5958024 respectively on PC2 which is close to 1, indicating that an increase in these variables is associated with a higher value of PC2 which indicates into a person’s physical health. Also “Triceps skinfold thickness” with negative loading of -0.3124659 on PC2 which is close to -1, indicating that an increase in these variables is associated with a lower value of PC2.
3. Also “Diastolic blood pressure” and “Body mass index” has a positive loading of 0.52068333 and 0.35836853 respectively on PC3 which is close to 1, indicating that an increase in these variables is associated with a higher value of PC3 which indicates into person’s cardiovascular well-being and “Plasma glucose concentration” with negative loading of -0.56980988on PC3 which is close to -1, indicating that an increase in these variables is associated with a lower value of PC3.

Biplot is a type of scatterplot used in PCA. In this special plot, the original data is represented by principal components that explain most of the data variance using the loading vectors and PC scores (Schork, (n.d.)).

```r
# Biplot with labels and loadings for PC1 and PC2 as axes
#install.packages("ggfortify")
#install.packages("ggplot2")
library(ggfortify)
library(ggplot2)
autoplot(pca_results,
         label=TRUE, label.size=3, shape=FALSE,
         loadings=TRUE, loadings.label=TRUE,
         data=pca_sample, colour='class') +
  ggtitle("Biplot of PCA Results - PC1/PC2")
```

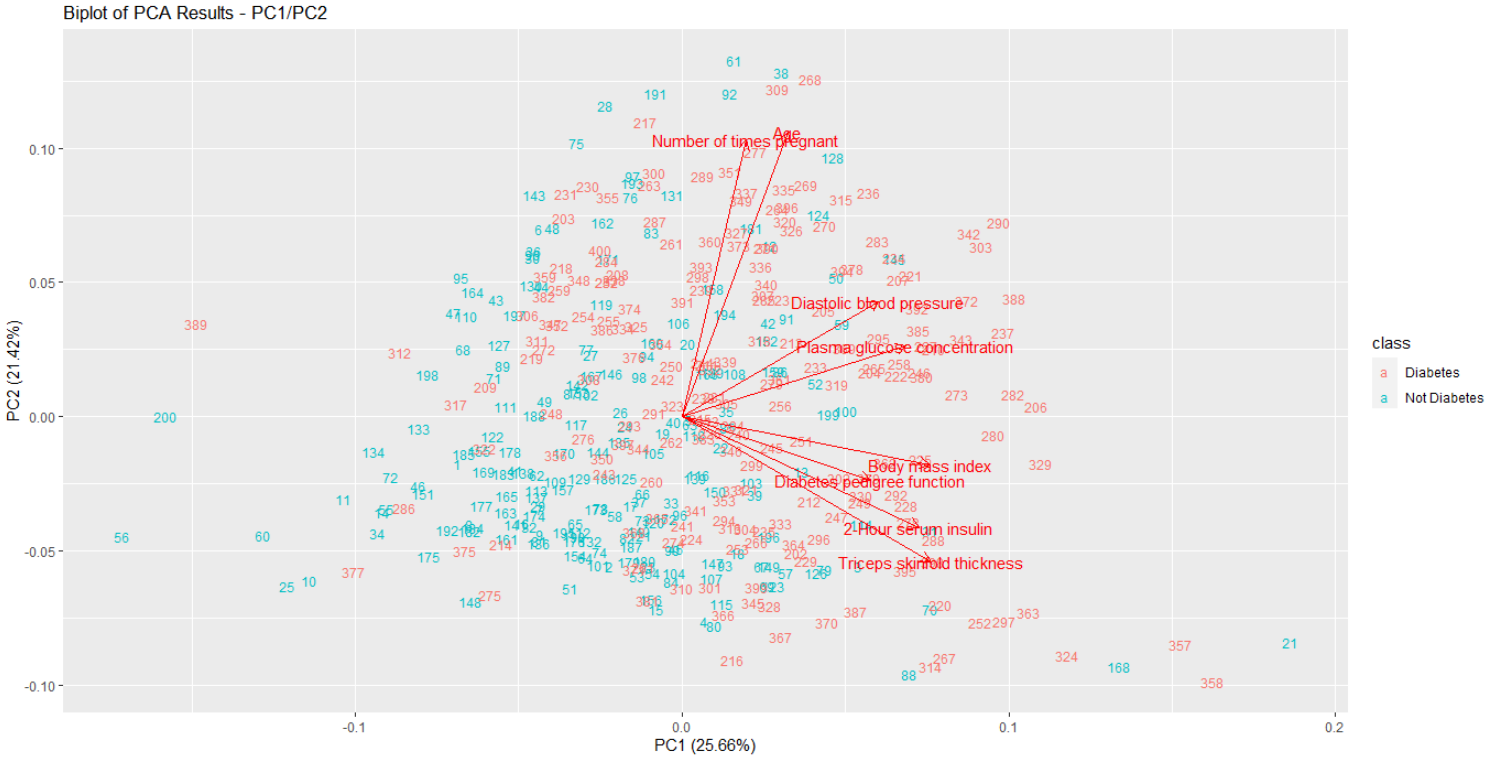

Image 03

The projection of each data point, also called variables, in our dataset onto these two primary components is indicated by its location in the PC1-PC2 space. In terms of their PC1 and PC2 scores, data points that are closer together in the plot typically exhibit comparable patterns.

The loading arrows indicate the direction and degree of correlation between the original variables and the principal components; they are not visible in this plot because of “shape = FALSE” in our code. Positive correlations exist between variables and PC1 or PC2 arrows, whilst negative correlations occur between variables and those pointing in the opposite way.

We can also determine whether there are any groupings or clusters in the data that are connected to the PCA results by color-coding the data points according to their class or category.

In other words, the left and bottom axes are of the PCA plot - I used them to read PCA scores of the samples (dots). The top and right axes belong to the loading plot - I used them to read how strongly each characteristic (vector) influences the principal components (BioTuring Team, 2018).

```
# 1(b). Biplot with labels and loadings for PC2 and PC3 as axes
autoplot(pca_results,x=2,y=3,
         label=TRUE, label.size=3, shape=FALSE,
         loadings=TRUE, loadings.label=TRUE,
         data=pca_sample, colour='class') +
  ggtitle("Biplot of PCA Results - PC2/PC3")
```

Image 04

In Image 04 the projection of each data point, also called variables in our dataset onto these two primary components is indicated by its location in the PC2-PC3 space. In terms of their PC2 and PC3 scores, data points that are closer together in the plot typically exhibit comparable patterns.

2.2 Factor Analysis (FA)

```
###############Factor Analysis###############
# Q.No. 02

# Remove the 'Target' column for further analysis
strata_sample <- select(strata_sample, -Target)
summary(strata_sample)
##  Number of times pregnant Plasma glucose concentration Diastolic blood pressure
##  Min.   : 0.00            Min.   :  0.0                Min.   :  0.00
##  1st Qu.: 1.00            1st Qu.:100.0                1st Qu.: 64.00
##  Median : 3.00            Median :122.0                Median : 72.00
##  Mean   : 4.05            Mean   :124.9                Mean   : 70.42
##  3rd Qu.: 7.00            3rd Qu.:147.0                3rd Qu.: 80.00
##  Max.   :15.00            Max.   :197.0                Max.   :114.00
##  Triceps skinfold thickness 2-Hour serum insulin Body mass index
##  Min.   : 0.00              Min.   :  0.00       Min.   : 0.00
##  1st Qu.: 0.00              1st Qu.:  0.00       1st Qu.:28.38
##  Median :25.00              Median :  7.00       Median :32.90
##  Mean   :21.52              Mean   : 85.67       Mean   :32.98
##  3rd Qu.:34.25              3rd Qu.:140.00       3rd Qu.:37.12
##  Max.   :99.00              Max.   :846.00       Max.   :59.40
##  Diabetes pedigree function      Age
##  Min.   :0.0780             Min.   :21.00
##  1st Qu.:0.2425             1st Qu.:25.00
##  Median :0.3770             Median :31.00
##  Mean   :0.4820             Mean   :34.38
##  3rd Qu.:0.6412             3rd Qu.:42.00
##  Max.   :2.4200             Max.   :81.00
```

We always assume the variables are continuous and homogeneous because factor analysis is based on the covariance or correlation matrix of the variables, which indicates that the variables should be measured on a continuous scale.

```
# Computing correlation matrix
cor <- cor(strata_sample)
View(cor)

# Computing eigenvalues
ev <- eigen(cor)
print(ev$values)
## [1] 2.0528050 1.7139237 1.0305114 0.8685479 0.7303141 0.6906567 0.4699236
## [8] 0.4433177
```

By looking at the eigenvalues, we can determine that only three of them are greater than 1.0, which means we can keep up to three components and carry on with the study. When compared to a single variable, eigenvalues greater than 1.0 will have more variance and significant information.

```r
# Performing scree plot analysis
#install.packages("nFactors")
library(nFactors)
## Loading required package: lattice
##
## Attaching package: 'nFactors'
##
## The following object is masked from 'package:lattice':
##
##     parallel
NScree = nScree(x = ev$values)
plotnScree(NScree)
```

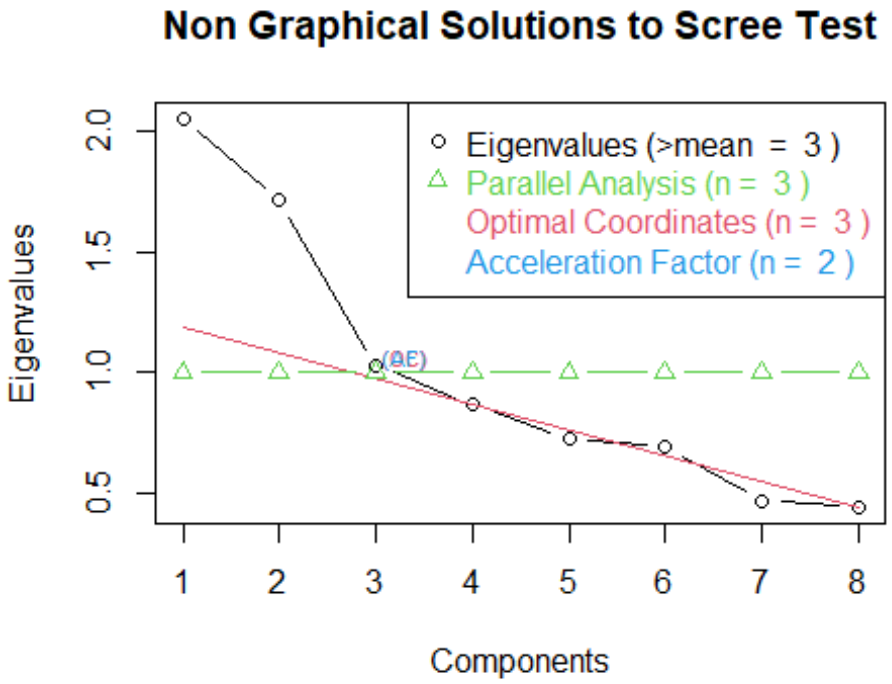

Image 05 - A scree plot with more information about a non-graphical scree analysis may be seen in the figure above. We may infer that three components should be taken into consideration for additional analysis based on the plot and legend above, which show that there are three eigenvalues above the mean of 1.0.

- **Loadings** - The loadings show the relationship between the extracted factors and the observed variables.
- **Uniqueness** - The specific variances, also known as uniquenesses, are the variations in the observed variables that cannot be accounted for by the factors.

- **Communalities** - The communalities indicate the extent to which the contributing components may account for the volatility of each variable.

```
# factor analysis without rotation
mlfa1 = factanal(strata_sample, factors = 3, rotation = 'none')
print(mlfa1, sort = T)
##
## Call:
## factanal(x = strata_sample, factors = 3, rotation = "none")
##
## Uniquenesses:
##     Number of times pregnant Plasma glucose concentration
##                        0.597                        0.133
##     Diastolic blood pressure   Triceps skinfold thickness
##                        0.810                        0.219
##         2-Hour serum insulin              Body mass index
##                        0.698                        0.794
##   Diabetes pedigree function                          Age
##                        0.906                        0.370
##
## Loadings:
##                              Factor1 Factor2 Factor3
## Plasma glucose concentration  0.911  -0.129  -0.142
## Triceps skinfold thickness    0.221   0.839   0.167
## Number of times pregnant      0.177  -0.273   0.545
## Age                           0.318  -0.341   0.642
## Diastolic blood pressure      0.219   0.122   0.358
## 2-Hour serum insulin          0.393   0.379
## Body mass index               0.275   0.341   0.118
## Diabetes pedigree function    0.238   0.193
##
##                Factor1 Factor2 Factor3
## SS loadings      1.345   1.223   0.905
## Proportion Var   0.168   0.153   0.113
## Cumulative Var   0.168   0.321   0.434
##
## Test of the hypothesis that 3 factors are sufficient.
## The chi square statistic is 14.75 on 7 degrees of freedom.
## The p-value is 0.0394
lm1 <- loadings(mlfa1)
sl1 <- lm1^2
communalities1 <- rowSums(sl1)
print(communalities1)
##     Number of times pregnant Plasma glucose concentration
##                   0.40290685                   0.86727903
##     Diastolic blood pressure   Triceps skinfold thickness
##                   0.19049539                   0.78068082
##         2-Hour serum insulin              Body mass index
##                   0.30245525                   0.20562654
```

```
##   Diabetes pedigree function                           Age
##                   0.09390612                   0.62997375
```

After considering **three factors** (from the eigenvalues and the scree plot) and assuming **no rotation** in the above maximum-likelihood factor analysis, we can calculate the communalities and obtain uniquenesses (specific variance), loadings, SS loadings, proportion variance, and cumulative variance.

From obtained **Uniquenesses**, the variables "Diabetes pedigree function", "Diastolic blood pressure", and "Body mass index" have high uniqueness values, indicating that the factors do not explain a large fraction of the variance in these variables.

From obtained **Loadings**,

- Variables “Plasma glucose concentration” and “2-Hour serum insulin” show higher loading, indicating a strong correlation with factor 1, indicating that they relate to **“glucose metabolism”**.
- Variables “Triceps skinfold thickness” and “Body mass index” have a greater loading, showing a strong link with factor 2, which in turn suggests they are associated with **“body composition”.**
- Variables “Age” and “Number of times pregnant” show higher loading, indicating a strong link with factor 3, which in turn implies they are associated with **“age & pregnancy”**.

From obtained **Communalities**, the variables “Plasma glucose concentration” and “Triceps skinfold thickness” exhibit higher communalities, indicating that their variation is tightly related to the retrieved three components.

Overall Evaluation:

The three factors capture distinct aspects of the dataset, with Factor 1 reflecting glucose metabolism, Factor 2 reflecting body composition, and Factor 3 reflecting age and pregnancy-related variables. The variables with lower communalities and higher specific variances might not be adequately explained by the extracted factors, indicating that other latent factors or variables not included in the analysis may be influencing these variables.

```
# factor analysis with “promax” rotation
mlfa2 = factanal(strata_sample, factors = 3, rotation = 'promax')
print(mlfa2, sort = T)
##
## Call:
## factanal(x = strata_sample, factors = 3, rotation = "promax")
##
## Uniquenesses:
##     Number of times pregnant Plasma glucose concentration
##                        0.597                        0.133
##     Diastolic blood pressure   Triceps skinfold thickness
##                        0.810                        0.219
##         2-Hour serum insulin              Body mass index
##                        0.698                        0.794
##   Diabetes pedigree function                          Age
##                        0.906                        0.370
##
## Loadings:
##                              Factor1 Factor2 Factor3
## Triceps skinfold thickness    0.949          -0.183
```

```
## Number of times pregnant              0.638
## Age                                   0.777
## Plasma glucose concentration -0.186           0.986
## Diastolic blood pressure      0.283   0.342
## 2-Hour serum insulin          0.369  -0.117   0.273
## Body mass index               0.409
## Diabetes pedigree function    0.197           0.163
##
##                Factor1 Factor2 Factor3
## SS loadings      1.365   1.153   1.121
## Proportion Var   0.171   0.144   0.140
## Cumulative Var   0.171   0.315   0.455
##
## Factor Correlations:
##         Factor1 Factor2 Factor3
## Factor1   1.000  -0.457  -0.191
## Factor2  -0.457   1.000   0.021
## Factor3  -0.191   0.021   1.000
##
## Test of the hypothesis that 3 factors are sufficient.
## The chi square statistic is 14.75 on 7 degrees of freedom.
## The p-value is 0.0394
lm2 <- loadings(mlfa2)
sl2 <- lm2^2
communalities2 <- rowSums(sl2)
print(communalities2)
##     Number of times pregnant Plasma glucose concentration
##                   0.41004991                   1.01160657
##     Diastolic blood pressure   Triceps skinfold thickness
##                   0.19817811                   0.93793715
##         2-Hour serum insulin              Body mass index
##                   0.22435726                   0.17543556
##   Diabetes pedigree function                          Age
##                   0.06674016                   0.61505956
```

We gain Uniquenesses (Specific Variance), Loadings, SS Loadings, Proportion Variance, and Cumulative Variance by considering **three factors** (w.r.t Scree plot and eigenvalues) and **'promax' rotation** in the above maximum-likelihood factor analysis.

From obtained **Uniquenesses**, the variables "Diabetes pedigree function", "Diastolic blood pressure", and "Body mass index" have high uniqueness values, indicating that the factors do not explain a large fraction of the variance in these variables.

From obtained **Loadings**,

- Variables "Triceps skinfold thickness" and "Body mass index" show higher loading, demonstrating a strong connection with factor 1, and consequently with **"body composition"**.
- Variables "Number of times pregnant" and "Age" have a greater loading, showing a strong link with factor 2, which in turn suggests they are associated with **"age and pregnancy"**.

- Variables “Plasma glucose concentration” and “2-Hour serum insulin” have a greater loading, suggesting a strong correlation with factor 3, implying they relate to **“glucose metabolism”**.

From obtained **Communalities**, the variables “Plasma glucose concentration” and “Triceps skinfold thickness” exhibit higher communalities, indicating that their variation is tightly related to the retrieved three factors.

Overall Evaluation:

Different features of the dataset appear to be captured by the factors that resulted from the "promax" rotation. Factor 1 is linked to certain health indicators, and body composition; Factor 2 is linked to age and pregnancy; and Factor 3 is linked to insulin response and glucose metabolism. The variables with smaller communalities and higher specific variances, which may not be adequately explained by the extracted factors, indicate that other latent factors or unmeasured variables are influencing these observations.

```
# factor analysis with “varimax” rotation
mlfa3 = factanal(strata_sample, factors = 3, rotation = 'varimax')
print(mlfa3, sort = T)
##
## Call:
## factanal(x = strata_sample, factors = 3, rotation = "varimax")
##
## Uniquenesses:
##     Number of times pregnant Plasma glucose concentration
##                        0.597                        0.133
##     Diastolic blood pressure   Triceps skinfold thickness
##                        0.810                        0.219
##         2-Hour serum insulin              Body mass index
##                        0.698                        0.794
##   Diabetes pedigree function                          Age
##                        0.906                        0.370
##
## Loadings:
##                              Factor1 Factor2 Factor3
## Triceps skinfold thickness    0.877
## Number of times pregnant              0.631
## Age                                   0.781   0.125
## Plasma glucose concentration  0.187   0.181   0.894
## Diastolic blood pressure      0.265   0.346
## 2-Hour serum insulin          0.473           0.272
## Body mass index               0.434           0.106
## Diabetes pedigree function    0.259           0.164
##
##                Factor1 Factor2 Factor3
## SS loadings      1.361   1.175   0.937
## Proportion Var   0.170   0.147   0.117
## Cumulative Var   0.170   0.317   0.434
##
## Test of the hypothesis that 3 factors are sufficient.
```

```
## The chi square statistic is 14.75 on 7 degrees of freedom.
## The p-value is 0.0394
lm3 <- loadings(mlfa3)
sl3 <- lm3^2
communalities3 <- rowSums(sl3)
print(communalities3)
##     Number of times pregnant Plasma glucose concentration
##                   0.40290685                   0.86727903
##     Diastolic blood pressure   Triceps skinfold thickness
##                   0.19049539                   0.78068082
##         2-Hour serum insulin              Body mass index
##                   0.30245525                   0.20562654
##   Diabetes pedigree function                          Age
##                   0.09390612                   0.62997375
```

We obtain Uniquenesses (Specific Variance), Loadings, SS Loadings, Proportion Variance, and Cumulative Variance by taking into account **three factors** (with respect to the Scree plot and eigenvalues) and **"varimax" rotation** in the above maximum-likelihood factor analysis.

From obtained **Uniquenesses**, High uniqueness values for the variables "Diabetes pedigree function", "Diastolic blood pressure" and "body mass index" suggest that a sizable amount of their variance cannot be attributed to the factors.

From obtained **Loadings**,

- The variables "2-hour serum insulin" and "triceps skinfold thickness" show higher loading, indicating a strong correlation with factor 1, which suggests a relationship between **"body composition" and "insulin levels"**.
- The variables "Age" and "Number of times pregnant" show higher loading, suggesting a strong correlation with factor 2, which suggests a relationship with **"age & pregnancy"**.
- Variables "Plasma glucose concentration" & "Diabetes pedigree function" have higher loading, indicating it has a strong association with factor 3 which inturn indicates they are associated with **'insulin levels and diabetes within the different age categories'**.

From obtained **Communalities**, the variables "Triceps skinfold thickness" and "plasma glucose concentration" exhibit greater communalities, suggesting a strong correlation between their variance and the three factors that were retrieved.

Overall Evaluation:

Different elements of the dataset appear to be captured by the three factors that were obtained from the "varimax" rotation. Factor 1 is linked to insulin levels and body composition, Factor 2 to age & pregnancy, and Factor 3 to insulin levels and diabetes within the different age categories. The likelihood that other latent factors or unmeasured variables are impacting these observations is suggested by the variables with smaller communalities and higher specific variances, which may not be sufficiently explained by the extracted factors.

```
# Install the psych package if not already installed
#install.packages("psych")
library(psych)
```

```
##
## Attaching package: 'psych'
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
# Assess sample using the Kaiser-Meyer-Olkin (KMO) measure
KMO(strata_sample)
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = strata_sample)
## Overall MSA =  0.61
## MSA for each item =
##     Number of times pregnant Plasma glucose concentration
##                         0.59                         0.58
##     Diastolic blood pressure   Triceps skinfold thickness
##                         0.69                         0.58
##         2-Hour serum insulin              Body mass index
##                         0.57                         0.68
##   Diabetes pedigree function                          Age
##                         0.77                         0.57
```

Data suitability for factor analysis can be ascertained statistically using the Kaiser-Meyer-Olkin measure (KMO) factor analysis.

The Overall Measure of Sampling Adequacy (MSA), derived from the obtained KMO measure, is 0.61, which is less than 0.7 (0.7 is recognised as the standard for adequacy). We can draw the conclusion that this data sample is unsuitable for factor analysis and that the variables in it have a lower common variance.

```
# Install below packages if not already installed
#install.packages("GPArotation")
#install.packages("paran")
#install.packages("semPlot")
library(semPlot)
library(GPArotation)
##
## Attaching package: 'GPArotation'
## The following objects are masked from 'package:psych':
##
##     equamax, varimin
library(paran)
## Loading required package: MASS
##
## Attaching package: 'MASS'
## The following object is masked from 'package:dplyr':
##
##     select
fa1 = fa(strata_sample, nfactors = 3, rotate = 'none')
fa.diagram(fa1)
```

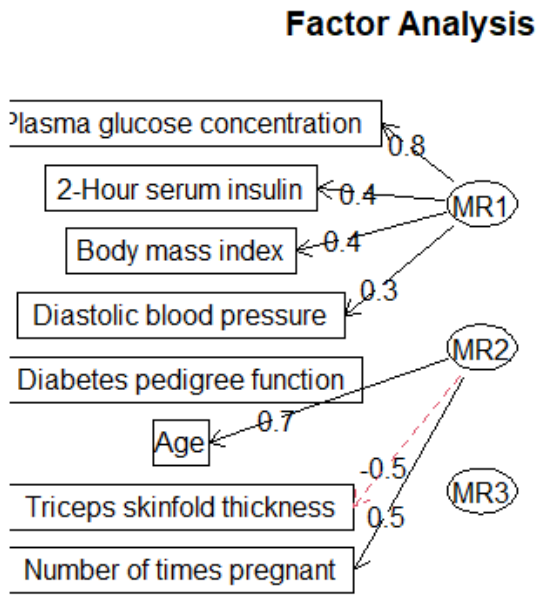

Image 06 - Factor analysis diagram with different factors without rotation and their loading with the variables.

```
fa2 = fa(strata_sample, nfactors = 3, rotate = 'promax')
fa.diagram(fa2)
```

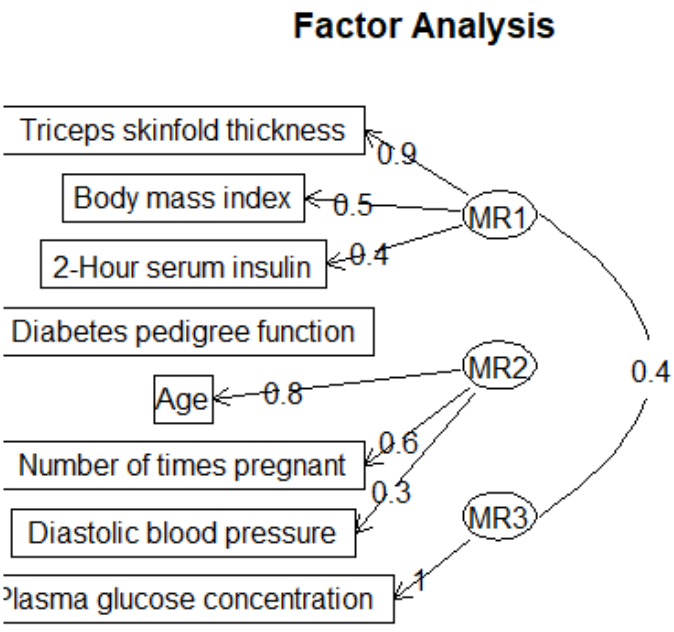

Image 07 - Factor analysis diagram with different factors with promax rotation and their loading with the variables

```
fa3 = fa(strata_sample, nfactors = 3, rotate = 'varimax')
fa.diagram(fa3)
```

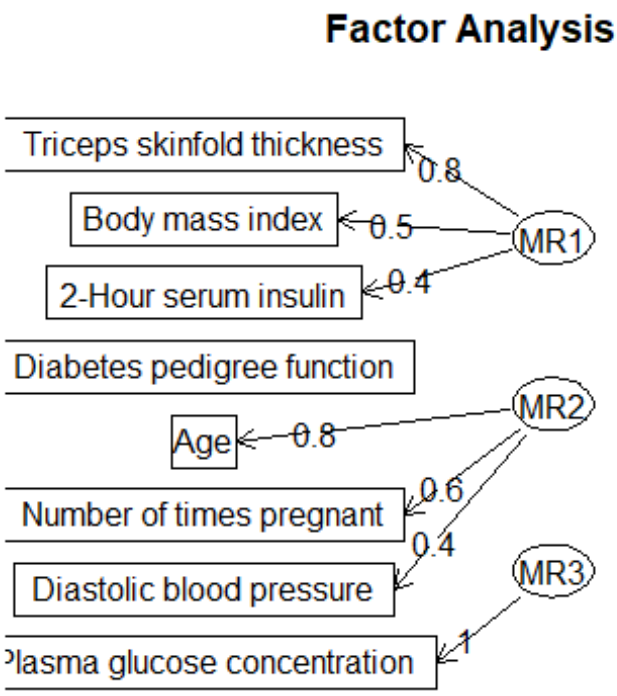

Image 08 - Factor analysis diagram with different factors with varimax rotation and their loading with the variables

2.3 Cluster Analysis

```
################Cluster Analysis###################
# Q.No. 03


#importing the  library ('cluster') for clustering
library('cluster')


##Task 1.3 (a)

data_clustering <- strata_sample[, 1:8]
methods <- c("single", "complete", "average", "ward")
dist_matrices <- c("manhattan", "euclidean")
results <- list()

for (method in methods) {
  for (dist_matrix in dist_matrices) {
    print(paste('Method =', method, ', ', Dist_matrix =', dist_matrix))
    agnes_result <- agnes(data_clustering, method = method, metric = dist_matrix)
    print(paste("Agglomerative coefficient:"))
    print(agnes_result$ac)
    results[[paste(method, dist_matrix)]] <- agnes_result
  }
}

## [1] "Method = single ,  Dist_matrix = manhattan"
## [1] "Agglomerative coefficient:"
## [1] 0.8072024
## [1] "Method = single ,  Dist_matrix = euclidean"
## [1] "Agglomerative coefficient:"
## [1] 0.8407644
## [1] "Method = complete ,  Dist_matrix = manhattan"
## [1] "Agglomerative coefficient:"
## [1] 0.9649909
## [1] "Method = complete ,  Dist_matrix = euclidean"
## [1] "Agglomerative coefficient:"
## [1] 0.9772495
## [1] "Method = average ,  Dist_matrix = manhattan"
## [1] "Agglomerative coefficient:"
## [1] 0.9527873
## [1] "Method = average ,  Dist_matrix = euclidean"
## [1] "Agglomerative coefficient:"
## [1] 0.9714479
## [1] "Method = ward ,  Dist_matrix = manhattan"
## [1] "Agglomerative coefficient:"
## [1] 0.9886947
## [1] "Method = ward ,  Dist_matrix = euclidean"
```

```
## [1] "Agglomerative coefficient:"
## [1] 0.9928664
```

	Single	Complete	Ward	Average
Manhattan	0. 8072024	0. 9649909	0. 9886947	0. 9527873
Euclidean	0. 8407644	0. 9772495	0.9928664	0. 9714479

Table 02 - Agglomerative coefficients for different Methods and Distance Matrices

The 'strata_sample' dataset is subjected to agglomerative hierarchical clustering using different methods and distance matrices, with the R code computing the agglomerative coefficient for every combination. The findings imply that distinct and well-separated clusters are produced by the Ward method with the Euclidean distance matrix, which produces the highest coefficient (0.9928664). Strong clustering tendencies are indicated by other approaches comparatively high coefficients (0.8072024 to 0.9886947). These results imply that there are probably well-defined clusters in the dataset that can be successfully located by a variety of clustering methods.

Criteria for selection of relevant Method and Distance Matrix:

- Agglomerative Coefficient must be near to 1, ie; higher value.

Considering the above criterion and "Table 02", we can conclude that the most relevant cluster analysis we could get are from:

1. Method = ward, Distance matrix = Euclidean
2. Method = ward, Distance matrix = Manhattan

```
##1st Relevant selection
# Print and interpret the dendrograms for a selected method and distance metric
selected_method1 <- "ward"
selected_dist_matrix1 <- "euclidean"
selected_agnes1 <- results[[paste(selected_method1, selected_dist_matrix1)]]
# Set graphical parameters for a clear and clean plot
par(mfrow=c(1,1), mar=c(6,6,2,2))
# Modify global text size and then plot the dendrogram
par(cex = 0.8)  # Adjust text size here
# Plot the dendrogram with clear labels
plot(selected_agnes1, main = paste("Dendrogram_1: Method -", selected_method1, ", ",
"Distance Metric -", selected_dist_matrix1), hang = -1)

## Warning in plot.window(xlim, ylim, log = log, ...): "hang" is not a graphical
## parameter

## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "hang"
## is not a graphical parameter
```

```
## Warning in axis(1, at = at.vals, labels = lab.vals, ...): "hang" is not a
## graphical parameter
```

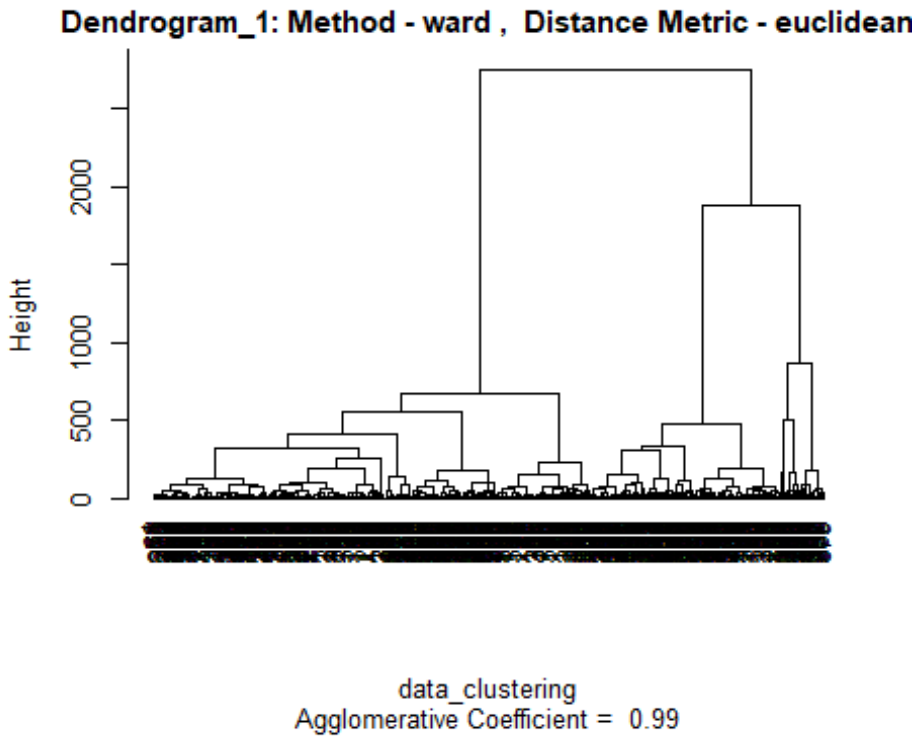

Image 09

Based on the dendrogram, we can infer that if we cut the dendrogram, we might receive three distinct clusters, possibly in the 1000–1500 "Height" range. These three clusters show that we can divide Pima Indians into three groups **(Patterns)** according to similar medical characteristics. Overall, the source code and results show how various clustering approaches are used and aid in assessing the quality of the generated clusters. We can conclude that the hierarchical clustering with Method 'ward' and Distance matrix 'Euclidean' gives the higher relationship with the dataset inside each cluster and suggests that the dataset is ideal for performing cluster analysis based on the agglomerative coefficient (0.9928664) and the three distinct clusters in the dendrogram.

```
##2nd Relevant selection
# Print and interpret the dendrograms for a selected method and distance metric
selected_method2 <- "ward"
selected_dist_matrix2 <- "manhattan"
selected_agnes2 <- results[[paste(selected_method2, selected_dist_matrix2)]]
# Plot the dendrogram with clear labels
plot(selected_agnes2, main = paste("Dendrogram_2: Method -", selected_method2, "Distance
Metric -", selected_dist_matrix2), hang = -1)

## Warning in plot.window(xlim, ylim, log = log, ...): "hang" is not a graphical
## parameter
```

```
## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "hang"
## is not a graphical parameter

## Warning in axis(1, at = at.vals, labels = lab.vals, ...): "hang" is not a
## graphical parameter
```

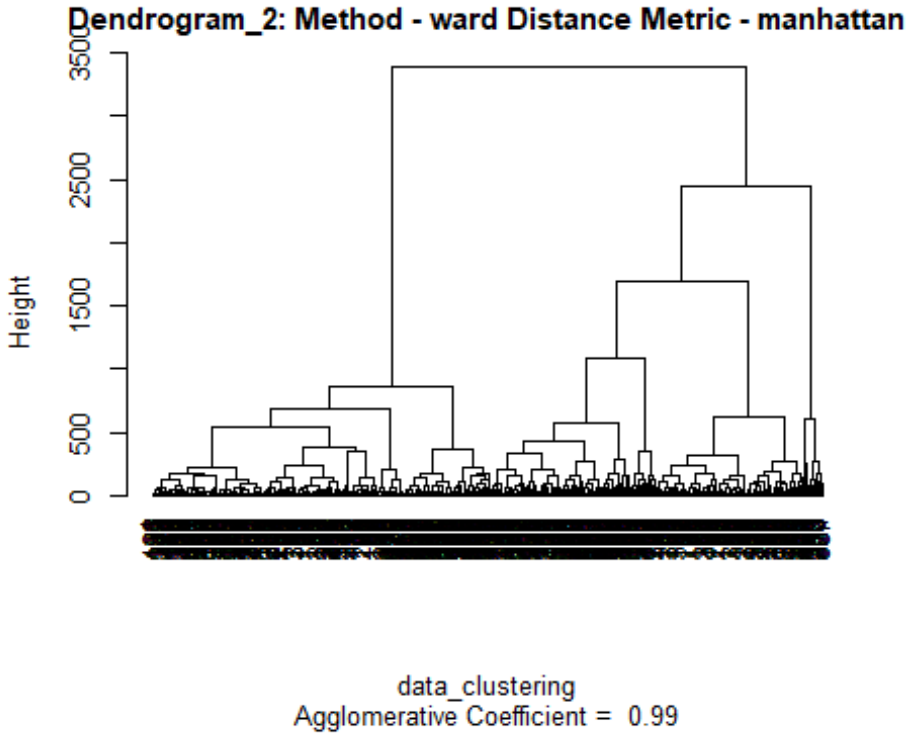

Image 10

Based on the dendrogram, we can infer that if we cut the dendrogram, we might receive **three distinct clusters**, at Height of 2000 (OR) we might also receive **four distinct clusters** at Height of 1500. These clusters show that we can divide Pima Indians into three groups **(Patterns)** according to similar medical characteristics. Overall, the source code and results show how various clustering approaches are used and aid in assessing the quality of the generated clusters. We can conclude that the hierarchical clustering with Method 'ward' and Distance matrix 'Manhattan' gives the higher relationship with the dataset inside each cluster and suggests that the dataset is ideal for performing cluster analysis based on the agglomerative coefficient (0.9886947) and the distinct clusters in the dendrogram.

```
##Task 1.3 (b)
# Transpose the data to cluster the measurements (columns)
data_transposed <- t(data_clustering)
```

The data undergoes transformation to enable column clustering, and the specified distance metric and algorithm are used to carry out hierarchical clustering. To illustrate the connections between diabetes measures, a dendrogram is created.

```r
for (method in methods) {
  for (dist_matrix in dist_matrices) {
    print(paste('Method =', method,'Dist_matrix =', dist_matrix))
    agnes_result <- agnes(data_transposed, method = method, metric = dist_matrix)
    print(paste("Agglomerative coefficient:"))
    print(agnes_result$ac)
    results[[paste(method, dist_matrix)]] <- agnes_result
  }
}
```

```
## [1] "Method = single Dist_matrix = manhattan"
## [1] "Agglomerative coefficient:"
## [1] 0.6507378
## [1] "Method = single Dist_matrix = euclidean"
## [1] "Agglomerative coefficient:"
## [1] 0.7136693
## [1] "Method = complete Dist_matrix = manhattan"
## [1] "Agglomerative coefficient:"
## [1] 0.7508032
## [1] "Method = complete Dist_matrix = euclidean"
## [1] "Agglomerative coefficient:"
## [1] 0.7364847
## [1] "Method = average Dist_matrix = manhattan"
## [1] "Agglomerative coefficient:"
## [1] 0.6420413
## [1] "Method = average Dist_matrix = euclidean"
## [1] "Agglomerative coefficient:"
## [1] 0.6772807
## [1] "Method = ward Dist_matrix = manhattan"
## [1] "Agglomerative coefficient:"
## [1] 0.7539886
## [1] "Method = ward Dist_matrix = euclidean"
## [1] "Agglomerative coefficient:"
## [1] 0.7364501
```

The ‘strata_sample’ dataset is transposed and subjected to agglomerative hierarchical clustering using different methods and distance matrices, with the R code computing the agglomerative coefficient for every combination. The findings imply that distinct and well-separated clusters are produced by the Ward method with the Manhattan distance matrix, which produces the highest coefficient (0. 7539886), indicating a modest degree of similarity throughout the clusters. Strong clustering tendencies are indicated by other approaches comparatively high coefficients (0.7508032 to 0.7364847). These results imply that there are probably well-defined clusters in the dataset that can be successfully located by a variety of clustering methods.

Criteria for selection of relevant Method and Distance Matrix:

- Agglomerative Coefficient must be near to 1, ie; higher value.

Considering the above criterion, we can conclude that the “best” result for cluster analysis we could get are from:

- Method = ward, Distance matrix = Manhattan

```
# Define the best distance metric and hierarchical clustering method
best_dist_matrix <- "manhattan"
best_method <- "ward"
# Perform hierarchical clustering on the transposed data
best_cluster_result <- agnes(data_transposed, method = best_method, metric =
best_dist_matrix)
# Plot the dendrogram for feature clustering
par(mfrow=c(1, 1))
plot(best_cluster_result, main = paste("Dendrogram: Method -", best_method, ", ",
"Distance Metric -", best_dist_matrix), hang = -1)

## Warning in plot.window(xlim, ylim, log = log, ...): "hang" is not a graphical
## parameter

## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "hang"
## is not a graphical parameter

## Warning in axis(1, at = at.vals, labels = lab.vals, ...): "hang" is not a
## graphical parameter

## Warning in axis(ax$side, at = 0:(length(x$order) - 1), las = 1, labels =
## labels, : "hang" is not a graphical parameter
```

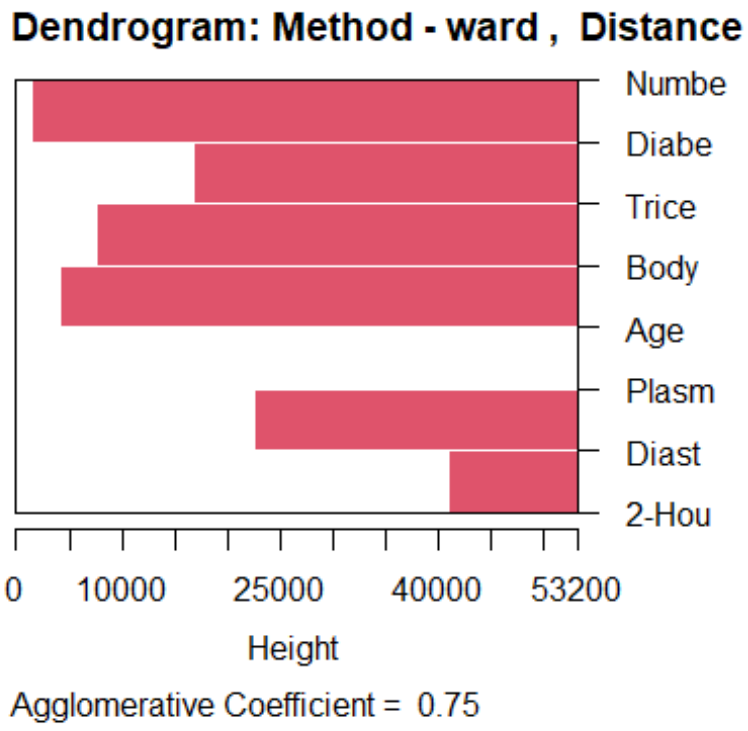

Image 11

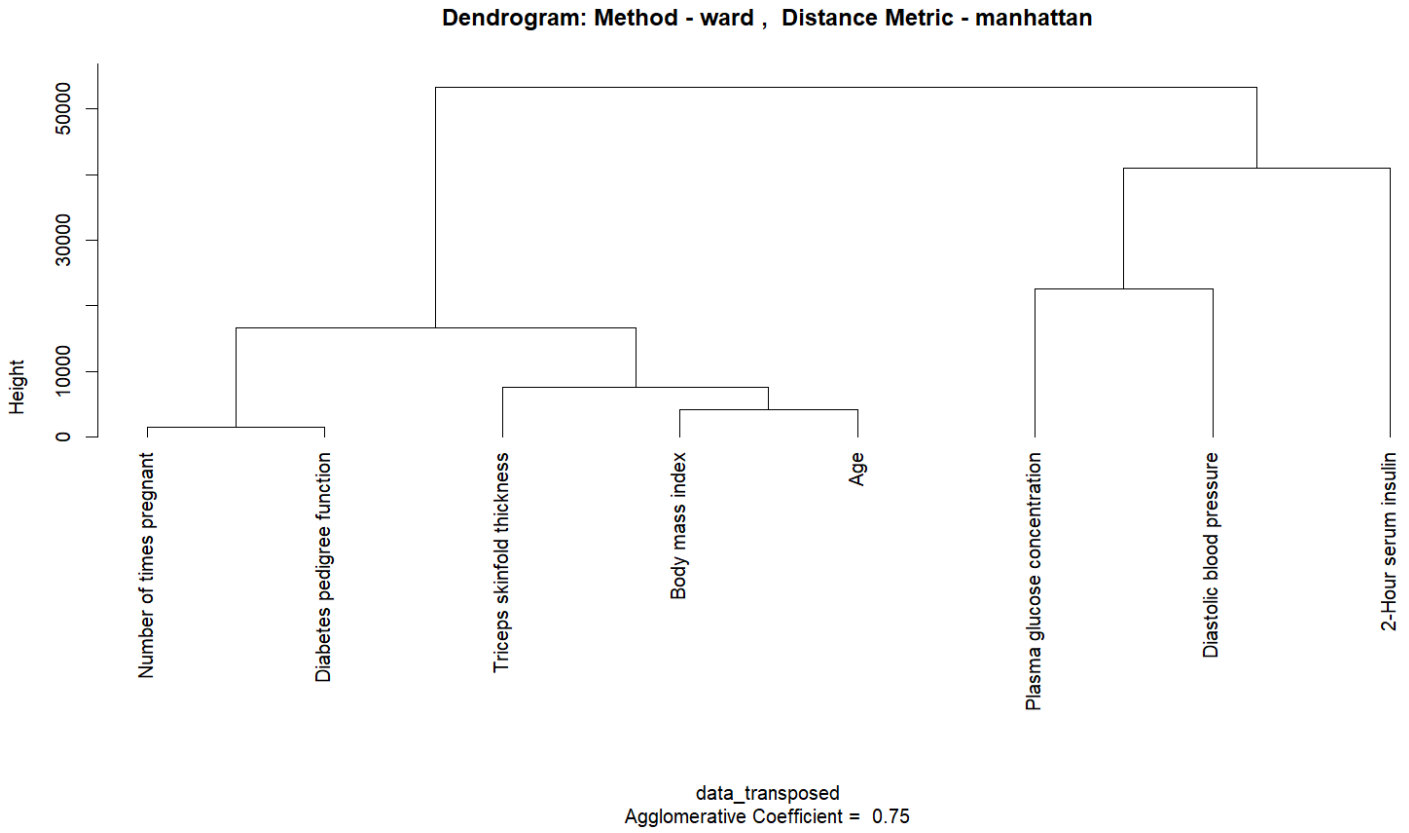

Image 12

Based on the dendrogram, we can infer that if we cut the dendrogram, we might receive **three distinct clusters**, at Height of 3000. These clusters show that we can divide 8 Variables into three groups **(Patterns)** indicating the high Covariance and variance among variables in each cluster.

Cluster 1:

- Number of times pregnant
- Diabetes pedigree function
- Triceps skinfold thickness
- Body mass index
- Age

Cluster 2:

- Plasma glucose concentration
- Diastolic blood pressure

Cluster 3:

- 2 – Hour serum insulin

(OR)

we might receive **two distinct clusters**, at Height of 5000.

Cluster 1:

- Number of times pregnant
- Diabetes pedigree function
- Triceps skinfold thickness
- Body mass index
- Age

Cluster 2:

- Plasma glucose concentration
- Diastolic blood pressure
- 2 – Hour serum insulin

Overall, the source code and results show how various clustering approaches are used and aid in assessing the quality of the generated clusters. We can conclude that the hierarchical clustering with Method ‘ward’ and Distance matrix ‘Manhattan’ gives the higher relationship between the variables inside each cluster and suggests that the dataset is ideal for performing cluster analysis based on the agglomerative coefficient (0. 7539886) and the distinct clusters in the dendrogram. This implies that the selected combination combines relevant diabetes measures efficiently.

Insights

1. **Common Findings**: Some variables, such “Plasma glucose concentration”, “Triceps skinfold thickness”, and “Age” are the most significant predictors for the onset of diabetes, according to PCA and FA. By confirming the existence of three or four separate groups within the dataset, cluster analysis may be able to distinguish between people with higher body mass indices and glucose levels and those with lower values. Furthermore, the cluster in Image 12 that includes the variables “Age”, “Body mass index”, and “Triceps skinfold thickness” indicates that age and body composition are the most important indicators of the beginning of diabetes.

2. **Consistency in Patterns**: Higher values for “Plasma glucose concentration”, “Triceps skinfold thickness” and “Age” may be linked to a higher risk of diabetes, according to the PCA and FA results. Also, the cluster in Image 12 consisting of “Triceps skinfold thickness”, “Body mass index”, & “Age” tells that Body composition and Age are the most significant predictors for the onset of diabetes.

3. **Conflicting Insights**: The cluster analysis reveals that some people, like “Plasma glucose concentration”, with seemingly low-risk profiles, nevertheless develop diabetes, despite PCA and FA suggesting a strong correlation between “Plasma glucose concentration”, “Triceps skinfold thickness” and “Age” and the onset of diabetes. The intricate interactions between several risk variables that aren’t fully represented by the primary components found using PCA and FA may be the cause of this dispute.

4. **Possible Explanations**: The contradictory observations may result from the complex nature of diabetes and its multiple aetiology. There may be additional genetic or environmental factors that affect diabetes development that aren’t fully accounted for by the variables used in the analysis. Additionally, we may want to examine them more closely to see whether there is anything unique about the subject or if there may just be data input errors.

5. **Implications and Recommendations**: A more thorough model for diabetes risk assessment can be developed by having a better understanding of the similarities and differences between the analyses. It's important to consider additional risk variables that could contribute to the development of diabetes in addition to concentrating on the major predictors found through PCA and FA analysis. Additional investigation, potentially integrating genetic or lifestyle data, may offer a more comprehensive comprehension of the onset of diabetes in the Pima Indian community.

References

Schork, J. (n.d.). *Biplot for PCA Explained.* Statisticsglobe. Retrieved November 02, 2023, from https://statisticsglobe.com/biplot-pca-explained

Team, B. (2018, June 19). *How to read PCA biplots and scree plots.* Medium. https://bioturing.medium.com/how-to-read-pca-biplots-and-scree-plots-186246aae063