STAT 6348
Applied Multivariate Analysis
Fall 2022
Project 3

This project is individual work. So do not consult with anybody in or out of class. You can ask me questions.

Sign on this page below and attach with your project. You project will not be graded without it.

This project is entirely my work. I have not discussed about this project with anybody in or out of class. I understand and have complied with the academic integrity policies written in the *Handbook of Operating Procedures* of UT Dallas https://policy.utdallas.edu/utdsp5003.

YOUR NAME ____________________________

DATE ____________________________

YOUR SIGNATURE ____________________________

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

Q1

Q1A) There are 252 observations (252 trading days in a year) of share prices of 11 companies (Amazon, Google, Toyota, Walmart, eBay, Apple, Pepsi, Coca-Cola, HSBC, Chase and Honda) and two stock indices; S&P500, Dow Jones in the stock dataset.

For PC Method:- So I started with finding eigen values of correlation matrix . Then I found factor loadings, communalities or h^2 and specific variances with the help of R which will lead us towards residual matrix of the data through PC method. All the relevant outputs are in output section.

I have created scatterplot with pairs command. It has too many features to make any definitive conclusions just by looking at the scatterplot. A correlation can be seen. However, since there are many variables , we don't have enough insight to draw any meaningful inference.

Factor analysis on the dataset has been applied as the financial market data is likely to be affected by some common factors, even though its not yet defined.

We know that the Σ matrix remains the same for the factors both with and without rotation,, hence the test statistic, p-value. R allows maximum 8 factors for a dataset with 13 variables, and I test that the number of factors, 1:8, are sufficient to describe the data.

H0: $\boldsymbol{\Sigma} = \boldsymbol{L} * \boldsymbol{L}' + \boldsymbol{\Psi}$ vs H1: $\boldsymbol{\Sigma}$ is any other positive definite matrix. Here the level of significance α is 0.05

The test that we use is a Likelihood Ratio test given by, including Bartlett's correction

$$\frac{n-1-(2p+4m+5)}{6}\ln\frac{|\boldsymbol{LL}'+\boldsymbol{\Psi}|}{|\boldsymbol{S}_n|} \sim \chi^2_{2[(p-m)^2-(p+m)]}$$

NUMBER OF FACTORS	DEGREES OF FREEDOM	TEST STATISTICS	P-VALUE	REJECT H_0
1	65	3353.47	0	NO
2	53	2417.91	0	NO
3	42	1198.74	7.59E-224	NO
4	32	847.81	1..62e-157	NO
5	23	403.05	4.18E-71	NO
6	15	255.22	1.05E-45	NO
7	8	148	5..13e-28	NO
8	2	103.16	3..97e-23	NO

As per the notes the off-diagonal elements are small, i.e., the covariance terms of S (correlation terms of R) are also closely explained by the fitted model L˜(L˜)` + Ψ˜ that means the covariance (or correlation) structure of the observed data has been explained well by the m-factor model. In that case we may take the m-factor model to be appropriate. We have following residual matrix on m=2 and m=3 model.

```
> #number of factors=3 #
> f=3
> # Residual matrix for f factor model#
> pred <- market.factor.analysis[[f]]$loadings%*%t(market.factor.analysis[[f]]$loadings) + diag(market.factor.analysis[[f]]$uniquenesses)
> pred
                sp500       amazon      Google      Toyota    Walmart        ebay      apple      pepsi  coca_cola   dow_jones         hsbc      chase       Honda
sp500      1.00087553  0.518050314  0.82982809  0.81913007  0.8701382  0.42817888  0.9388912  0.8470462  0.7097746  0.98312032 -0.474471484  0.9017306  0.04020764
amazon     0.51805031  0.999956490  0.24235472  0.12590904  0.3358920  0.68342562  0.3099982  0.5162160  0.2791928  0.52502045  0.005812444  0.2458124 -0.31352479
Google     0.82982809  0.242354720  0.99997402  0.78947945  0.7126111  0.08028714  0.8618319  0.5995635  0.5286732  0.82631728 -0.414191305  0.8672329  0.31552379
Toyota     0.81913007  0.125909042  0.78947945  1.00000166  0.8845266  0.08046344  0.9109120  0.7258854  0.7783822  0.78210025 -0.700131595  0.8953627  0.13185619
Walmart    0.87013816  0.335892027  0.71261110  0.88452660  1.0000009  0.38181876  0.8828487  0.8858576  0.8958294  0.82294947 -0.725497296  0.8272167 -0.16338879
ebay       0.42817888  0.683425618  0.08028714  0.08046344  0.3818188  1.00001830  0.2087440  0.5990686  0.4226614  0.40990458 -0.124642189  0.1093821 -0.62725586
apple      0.93889118  0.309998217  0.86183190  0.91091197  0.8828487  0.20874398  1.0000006  0.7762596  0.7257952  0.91868278 -0.575299384  0.9474854  0.16651643
pepsi      0.84704624  0.516216047  0.59956350  0.72588537  0.8858576  0.59906861  0.7762596  1.0000007  0.8483512  0.80511215 -0.605124705  0.6982885 -0.33855309
coca_cola  0.70977456  0.279192818  0.52867323  0.77838217  0.8958294  0.42266137  0.7257952  0.8483512  1.0000006  0.64758006 -0.765289076  0.6529119 -0.35755565
dow_jones  0.98312032  0.525020446  0.82631728  0.78210025  0.8229495  0.40990458  0.9186828  0.8051121  0.6475801  0.99999939 -0.412185997  0.8877398  0.08768077
hsbc      -0.47447148  0.005812444 -0.41419130 -0.70013159 -0.7254973 -0.12464219 -0.5752994 -0.6051247 -0.7652891 -0.41218600  0.999999798 -0.5320364  0.20974298
chase      0.90173064  0.245812413  0.86723292  0.89536271  0.8272167  0.10938213  0.9474854  0.6982885  0.6529119  0.88773980 -0.532036402  1.0000005  0.26918825
Honda      0.04020764 -0.313524792  0.31552379  0.13185619 -0.1633888 -0.62725586  0.1665164 -0.3385531 -0.3575556  0.08768077  0.209742976  0.2691883  1.00000014
>
>
> #number of factors=2 #
> f2=2
> # Residual matrix for f factor model#
> pred1 <- market.factor.analysis[[f]]$loadings%*%t(market.factor.analysis[[f]]$loadings) + diag(market.factor.analysis[[f2]]$uniquenesses)
> pred1
                sp500       amazon      Google      Toyota    Walmart        ebay      apple      pepsi  coca_cola   dow_jones         hsbc      chase       Honda
sp500      1.06045783  0.518050314  0.82982809  0.81913007  0.8701382  0.42817888  0.9388912  0.8470462  0.7097746  0.98312032 -0.474471484  0.9017306  0.04020764
amazon     0.51805031  1.348688425  0.24235472  0.12590904  0.3358920  0.68342562  0.3099982  0.5162160  0.2791928  0.52502045  0.005812444  0.2458124 -0.31352479
Google     0.82982809  0.242354720  0.99042233  0.78947945  0.7126111  0.08028714  0.8618319  0.5995635  0.5286732  0.82631728 -0.414191305  0.8672329  0.31552379
Toyota     0.81913007  0.125909042  0.78947945  1.11099807  0.8845266  0.08046344  0.9109120  0.7258854  0.7783822  0.78210025 -0.700131595  0.8953627  0.13185619
Walmart    0.87013816  0.335892027  0.71261110  0.88452660  1.0528378  0.38181876  0.8828487  0.8858576  0.8958294  0.82294947 -0.725497296  0.8272167 -0.16338879
ebay       0.42817888  0.683425618  0.08028714  0.08046344  0.3818188  1.23369544  0.2087440  0.5990686  0.4226614  0.40990458 -0.124642189  0.1093821 -0.62725586
apple      0.93889118  0.309998217  0.86183190  0.91091197  0.8828487  0.20874398  0.9915985  0.7762596  0.7257952  0.91868278 -0.575299384  0.9474854  0.16651643
pepsi      0.84704624  0.516216047  0.59956350  0.72588537  0.8858576  0.59906861  0.7762596  0.9713639  0.8483512  0.80511215 -0.605124705  0.6982885 -0.33855309
coca_cola  0.70977456  0.279192818  0.52867323  0.77838217  0.8958294  0.42266137  0.7257952  0.8483512  1.1427037  0.64758006 -0.765289076  0.6529119 -0.35755565
dow_jones  0.98312032  0.525020446  0.82631728  0.78210025  0.8229495  0.40990458  0.9186828  0.8051121  0.6475801  1.10423699 -0.412185997  0.8877398  0.08768077
hsbc      -0.47447148  0.005812444 -0.41419130 -0.70013159 -0.7254973 -0.12464219 -0.5752994 -0.6051247 -0.7652891 -0.41218600  1.331518337 -0.5320364  0.20974298
chase      0.90173064  0.245812413  0.86723292  0.89536271  0.8272167  0.10938213  0.9474854  0.6982885  0.6529119  0.88773980 -0.532036402  1.0012783  0.26918825
Honda      0.04020764 -0.313524792  0.31552379  0.13185619 -0.1633888 -0.62725586  0.1665164 -0.3385531 -0.3575556  0.08768077  0.209742976  0.2691883  1.01795617
>
```

The factor loadings both without and "varimax" rotation is in the output section.

ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

F1: This is the market factor with SP500, Dow Jones, Apple, Walmart, Chase, Google, Toyota and Pepsi playing major roles. This factor can be called consumer industry factor.
F2: Auto manufacturer Honda and e-commerce company eBay dominate; This factor can be called industry factor

F3: On factor 3, Banking firm HSBC plays a major role; this factor can be called banking factor.

After rotation, the interpretation changes slightly and gets only slightly simple

F1*: This is the market factor with SP500, Dow Jones, Apple, Google, Toyota and Chase playing the major roles.

F2*: Banking firm HSBC and retail industry Walmart plays a major role; This can be called industry group1 factor.

F3*: E-commerce company eBay and Amazon dominate; This can be called ecommerce factor

Note:- Factor with varimax rotation and no rotation has also been put in output section along with aproximate correlation/covariance matrix for m=3 factor model.
The m-factor model here is written as $\boldsymbol{X} - \boldsymbol{\mu} = \boldsymbol{LF} + \boldsymbol{\epsilon}$, with standard notations.

The assumptions in this model are :-

- The relationship between observed variables X and the underlying factors F are linear.
- F and ϵ are independent, that is F1, F2, … , Fm are uncorrelated with $\epsilon_1, \epsilon_2, \ldots, \epsilon_m$
- Mean of F is 0, Cov(F) = I, that is Var(Fj) = 1, Cov(Fi, Fj) = 0. The factors are uncorrelated, this is an orthogonal model.
- Mean of ϵ is 0. Cov(ϵ) = Ψ where $\Psi = \text{diag}(\Psi_1, \Psi_2, \ldots, \Psi_p)$. Therefore Var($\epsilon_i$) = Ψ_i, Cov(ϵ_i, ϵ_j) = 0. The specific factors are uncorrelated.

Q1B. Principal component analysis of the Stocks data with these 13 independent variables has been conducted. The variables differ much in scale their range, using R command summary() and the variances from variance matrix, using R command var (). Because of this reason have PC analysis with correlation matrix has been conducted . Here I have gone by understanding of statistics, using scree plot(in output) , I determine, 3 PCs are sufficient for this analysis, and together they explain 92% of the variation in the data.

```
Loadings:
           Comp.1 Comp.2 Comp.3
sp500       0.336         0.202
amazon      0.145 -0.378  0.525
Google      0.287  0.241  0.109
Toyota      0.316  0.193 -0.145
Walmart     0.333        -0.134
ebay        0.139 -0.530  0.134
apple       0.333  0.153
pepsi       0.314 -0.218
coca_cola   0.296 -0.152 -0.289
dow_jones   0.323         0.250
hsbc       -0.230         0.591
chase       0.315  0.221  0.111
Honda              0.579  0.325
```

PC1: This is the market component with SP500, Dow Jones, Apple, Walmart, Chase, Toyota and Pepsi playing major roles.

PC2: Auto manufacturer Honda and e-commerce company eBay play major roles, but with opposing influence; I call this industry group1 activity.

PC3: Banking firm HSBC and e-commerce company Amazon play major roles; I call this industry group2 activity.

The interpretation of the PCs is closer to that of the factors, without rotation.

Variance Explained	F1 / PC1	F2 / PC2	F2 / PC2 (cumulative)	F3 / PC3	F2 / PC3 (cumulative)
FA method	0.454	0.235	0.689	0.194	0.883
PC method	0.6326	0.178	0.8107	0.1092	0.92

Note:- the above table is based on outputs attached in output section.

We can see that the PC analysis explain more of the variance in the data as that is what is maximized in this analysis. Factor analysis on the other hand maximize the explanation of the covariance/correlation in the data, model is based on maximum likelihood estimation.

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

Q2

Substituting $\bar{\boldsymbol{x}}_1$, $\bar{\boldsymbol{x}}_2$, and $\boldsymbol{S}_{pooled}$ for $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$ and $\boldsymbol{\Sigma}$, where $\boldsymbol{S}_{pooled} = \frac{(n_1-1)\boldsymbol{S}_1+(n_2-1)\boldsymbol{S}_2}{n_1+n_2-2}$, and taking log we get the *estimated* minimum ECM rule for two normal populations as:

$$R_1 : (\bar{\boldsymbol{x}}_1 - \bar{\boldsymbol{x}}_2)'\boldsymbol{S}_{pooled}^{-1}\boldsymbol{x} - \frac{1}{2}(\bar{\boldsymbol{x}}_1 - \bar{\boldsymbol{x}}_2)'\boldsymbol{S}_{pooled}^{-1}(\bar{\boldsymbol{x}}_1 + \bar{\boldsymbol{x}}_2) \geq \ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right]$$

Allocate to π_2 otherwise.
If $\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right) = 1$, then basically we will comparing $(\bar{\boldsymbol{x}}_1 - \bar{\boldsymbol{x}}_2)'\boldsymbol{S}_{pooled}^{-1}\boldsymbol{x} = \hat{\boldsymbol{a}}'\boldsymbol{x} = \hat{y}$

Some background:-

We know that we need to perform linear discriminant analysis to classify if patients are diabetic or not based on other variables. We would use the formula shown in the adjoining screenshot to the left to create a code that will classify patient as having diabetes or not.

Also the formula used to find AER and APER is as follows:-

$APER = (n_{1M} + n_{2M})/(n_1 + n_2)$

Q2A) Q1.The discriminant analysis on the diabetes dataset has been performed.

The coefficients in the discriminant are as follows:-

	MANUALLY CALCULATED A'	MASS:LDA OUTPUT FOR A'
PREGNANCIES	-0.148436901	0.093864
GLUCOSE	-0.046585207	0.026986
BLOOD PRESSURE	0.020449175	-0.01063
SKIN THICKNESS	-0.006120851	0.000704
INSULIN	0.003017222	-0.00082
BMI	-0.078964735	0.06037
DIABETES PEDIGREE FUNCTION	-1.171166479	0.671152
AGE	0.001480299	0.011949

Note:- the above table is based on outputs attached in output section.

The coefficients computed by the two methods are multiple of each other. The classification rule for new patients is as follows:-

$\hat{\boldsymbol{a}}'\boldsymbol{x}_{\text{new}} \geq -$ -8.016908 allocates the patient to group 1, that is Outcome=0 meaning there is presence of diabetes

if not then allocate to group 2, that is Outcome=1, meaning there is no Diabetes.

We are given a new patient comes with pregnancies = 5, glucose = 150, blood pressure = 90, skin thickness = 20, insulin = 100, BMI = 35, diabetes pedigree function = 0.5, and age = 35 and we are to predict the diabetes status for this patient. the number calculated with above conditions is −9.01, so we can conclude that there is no diabetes.

2B) The confusion matrix to get the "plug-in" estimate of misclassification rate is computed with following results:-

APER = 21.61% and the "leave-one-out" estimate of misclassification rate with cross-validation, AER = 22.52%.

The calculations based on the output posted in output section is as follows:-

APER= (54+112)/(446+54+112+156) = 0.2161458

AER= (58+115)/(442+58+115+153) = 0.2252604

So, AER > APER, and which is in-line with our expectation.

(2C) As per the question logistic regression model is developed In that model Outcome is designated as the dependent variable and all others as independent, numeric variable .The probability computed for the new patient with above mentioned conditions namely pregnancies = 5, glucose = 150, blood pressure = 90, skin thickness = 20, insulin = 100, BMI = 35, diabetes pedigree function = 0.5, and age = 35 is 0.57. It is less than the prior probability which was 0.65 of diabetic patients in the given dataset. So, we can conclude that the new patient is diabetic. Hence the logistic regression model contradicts the results of discriminant analysis, under the assumption of prior probability being the same as that in the existing data.

The confusion matrix is computed to get the "plug-in" estimate of misclassification rate with following results:-

APER = 22.665% and the "leave-one-out" estimate of misclassification rate with cross-validation, AER = 23.18%.

The calculations based on the output posted in output section is as follows:-

APER= (31+147)/(472+28+146+122) = 0.22665625

AER= (31+147)/(469+31+147+121) = 0.2317708

We can see higher rates of misclassification in case of logistic regression outcome, using prior probabilities same as that in the existing data. Hence, I have reran the logistic regression model with prior probability = 0.5 for each of the outcomes, as there can be one of the two outcomes, meaning either patient has Diabetes or No Diabetes. The confusion matrix is computed to get the “plug-in” estimate of misclassification rate with following results:-

APER = 21.74% and the “leave-one-out” estimate of misclassification rate with cross-validation, AER = 22.26%.

The calculations based on the output posted in output section is as follows:-

APER= (55+112)/(445+55+112+156) = 0.2174479

AER= (57+114)/(443+57+114+154) = 0.226560032

In my opinion, using prior probabilities the one which is same as that in the existing data, is more accurate.

Q3

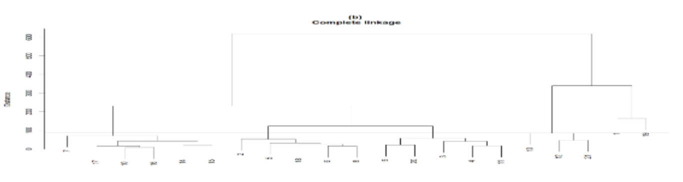
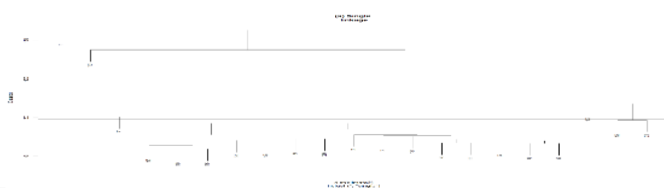
Q3A) The scatterplot matrix reveals three major findings,

The most of the variables are highly correlated with each other.

- Some of the two variables form a positive linear relationship for example:- MAT vs. MCMT and AHM vs. SHM
- Some of the two variables form a negative linear relationship for example:-MWMT vs. MAP and MSP vs. SHM).

Beside this we can summarize that the scatterplot matrix shows clusters of points in the climactic variables namely MAT, MWMT, MCMT, TD, MAP, MSP for the different BIOME and Cluster analysis is likely to reveal further insight than scatterplot matrix.

Q 3 B) As per the question I have done hierarchical clustering and we have following results:-

Note:- Zoomed version of dendrograms are in the output

TYPE OF LINKAGE	SINGLE LINKAGE	COMPLETE LINKAGE
NO. OF CLUSTERS	6	6
CUT POINT	h=48	h=90

On the table below ,BIOME variable refers to 4 different ecosystems, namely Montane represented by 5, Boreal represented by 9, Parkland represented by 4 and Grassland represented by 3;

Obs	ECOSYS	BIOME	Single Linkage	Complete Linkage	K-means
1	A	Montane	C1	C1	K1
2	AP	Boreal	C2	C2	K3
3	BSA	Boreal	C2	C3	K4
4	CM	Boreal	C2	C3	K4
5	CP	Parkland	C2	C2	K4
6	DM	Boreal	C2	C2	K4
7	DMG	Grassland	C3	C4	K3
8	FF	Grassland	C2	C2	K4
9	FP	Parkland	C2	C3	K4
10	KU	Boreal	C2	C4	K3
11	LBH	Boreal	C2	C3	K4
12	LF	Montane	C4	C5	K2
13	M	Montane	C5	C5	K2
14	MG	Grassland	C2	C4	K3
15	NF	Grassland	C2	C4	K3
16	NM	Boreal	C2	C4	K3
17	Peac	Boreal	C2	C4	K3
18	PRP	Parkland	C2	C2	K4
19	SA	Montane	C6	C6	K1
20	UBH	Boreal	C2	C3	K4
21	UF	Montane	C4	C5	K2

Notes:- Means of the clusters are as follows:-

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

```
>
> # Mean vectors for Complete Linkage cluster#
> colMeans(climate2[cluster.complete == 1,])
 MAT  MWMT  MCMT    TD   MAP   MSP
-2.5   8.6 -12.6  21.3 927.2 387.2
> colMeans(climate2[cluster.complete == 2,])
  MAT   MWMT   MCMT     TD    MAP    MSP
 1.46  16.28 -16.04  32.34 450.10 290.42
> colMeans(climate2[cluster.complete == 3,])
  MAT   MWMT   MCMT     TD    MAP    MSP
-0.60  14.72 -18.38  33.12 506.60 316.94
> colMeans(climate2[cluster.complete == 4,])
      MAT      MWMT       MCMT        TD       MAP       MSP
  0.75000  17.15000 -18.23333  35.35000 376.05000 237.65000
> colMeans(climate2[cluster.complete == 5,])
       MAT       MWMT       MCMT         TD        MAP        MSP
  1.766667  13.966667 -11.566667  25.533333 607.533333 384.533333
> colMeans(climate2[cluster.complete == 6,])
 MAT  MWMT  MCMT    TD   MAP   MSP
-0.2  11.3 -11.8  23.1 764.0 371.6
```

```
> # Mean vectors for each cluster#
>
> # Mean vectors for Single Linkage cluster#
> colMeans(climate2[cluster.single == 1,])
 MAT  MWMT  MCMT    TD   MAP   MSP
-2.5   8.6 -12.6  21.3 927.2 387.2
> colMeans(climate2[cluster.single == 2,])
        MAT        MWMT         MCMT          TD          MAP          MSP
  0.3066667  15.9600000 -17.9533333  33.9133333 447.0800000 283.1866667
> colMeans(climate2[cluster.single == 3,])
 MAT  MWMT  MCMT    TD   MAP   MSP
 4.2  18.5 -12.2  30.7 333.6 214.9
> colMeans(climate2[cluster.single == 4,])
  MAT   MWMT   MCMT     TD    MAP    MSP
 1.55  14.05 -12.30  26.30 610.50 408.75
> colMeans(climate2[cluster.single == 5,])
 MAT  MWMT  MCMT    TD   MAP   MSP
 2.2  13.8 -10.1  24.0 601.6 336.1
> colMeans(climate2[cluster.single == 6,])
 MAT  MWMT  MCMT    TD   MAP   MSP
-0.2  11.3 -11.8  23.1 764.0 371.6
```

Note:- Zoomed version is in the output section

Q3C)K-means clustering with set.seed(6348) has been performed and based on the plot of wss, I have chosen 4 clusters as that yields a sharp drop in within group sum of squares output for Cluster means is as follows:-

```
K-means clustering with 4 clusters of sizes 2, 3, 7, 9

Cluster means:
        MAT     MWMT      MCMT       TD      MAP      MSP
1 -1.3500000  9.95000 -12.20000 22.20000 845.6000 379.4000
2  1.7666667 13.96667 -11.56667 25.53333 607.5333 384.5333
3  0.4714286 17.11429 -18.88571 35.97143 383.5714 241.7429
4  0.6111111 15.34444 -16.58889 31.95556 483.8667 307.8333

Clustering vector:
 [1] 1 3 4 4 4 4 3 4 4 3 4 2 2 3 3 3 3 4 1 4 2

Within cluster sum of squares by cluster:
[1] 13447.030  4673.760  7560.377 11038.722
 (between_SS / total_SS =  92.1 %)
```

The comparison between hierarchical and k-means clustering is given in the above table.

One thing we can see is that the number of clusters in K-means is same as that of the categories in BIOME variable.

Also WSS plot in the output shows 4 is optimum number of groups /cluster for this data

Q3D) The classification rule for new observation is as follows:-

$\hat{\boldsymbol{a}}'\boldsymbol{x}_{\text{new}} \geq -5566.198$ allocates the observation to group 1, that is Outcome=combined biome which comprised of grassland and parkland.

The confusion matrix to get the "plug-in" estimate of misclassification rate is computed with following results:-

APER = 21.61% and the "leave-one-out" estimate of misclassification rate with cross-validation, AER = 22.52%.

The calculations based on the output posted in output section is as follows:-

APER= (7)/(9+7+5) = 0.33
AER= (6+1)/(8+1+1+1+6+4) = 0.33

ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

R OUTPUT for Q 1 :-

For PC method of factor analysis:-

Eigen decomposition of correlation matrix

```
eigen() decomposition
$values
 [1] 8.224600825 2.314613628 1.419227132 0.388759550 0.244030569 0.147201866 0.094009903 0.068531644 0.040661614 0.025425324 0.018820303 0.012063644 0.002053998

$vectors
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]	[,13]
[1,]	-0.33552210	-0.007322427	-0.20232747	0.10209208	0.02402816	-0.14033675	-0.05589108	0.27974651	-0.18029251	0.10230643	-0.04547493	-0.10064098	0.824129755
[2,]	-0.14530947	0.377952991	-0.52481536	-0.31915879	0.19882443	0.58781188	-0.19998584	0.09862401	-0.01540064	-0.05012398	-0.02043937	0.11891987	-0.081602182
[3,]	-0.28673686	-0.240755356	-0.10939299	0.48974539	0.42481252	0.29382539	0.33858726	-0.37565953	0.15945192	0.08323348	-0.22881852	-0.02532172	-0.009443356
[4,]	-0.31616110	-0.192913544	0.14543238	-0.14299116	-0.31511539	0.33391720	0.01681475	0.03763887	0.40689573	-0.12546694	0.32944267	-0.56383729	0.031314390
[5,]	-0.33348720	0.038344683	0.13441358	-0.19622615	-0.23428394	-0.10932420	-0.24161793	-0.21685887	0.51893068	0.32437548	-0.25811026	0.45889215	0.082371746
[6,]	-0.13941147	0.530439703	-0.13403573	0.50537670	-0.48366519	-0.03252559	0.15504285	-0.14561869	0.01622178	-0.29682978	0.17705274	0.16059033	-0.009271627
[7,]	-0.33287676	-0.153127231	-0.04269931	0.06051623	0.22980759	-0.14218399	-0.03050080	0.09379396	-0.08807435	0.21742673	0.76208427	0.32852603	-0.183768860
[8,]	-0.31359138	0.218321305	0.01743095	-0.19322841	0.21234195	-0.34468148	-0.31209962	-0.58944391	-0.23620478	-0.17209650	0.01211709	-0.34496556	-0.055249477
[9,]	-0.29558303	0.151860854	0.28918972	-0.37727991	-0.16363357	0.14149217	0.61727667	-0.05181539	-0.42400872	0.19941563	-0.08843594	0.07073380	-0.026695472
[10,]	-0.32334580	-0.030303607	-0.24958190	0.20651879	-0.20327503	-0.19819961	-0.11899417	0.35626448	-0.13475187	0.34336341	-0.31054698	-0.29150211	-0.500430643
[11,]	0.22965659	0.016229092	-0.59060795	-0.25650703	-0.05826386	-0.35031144	0.44212270	-0.18955792	0.28973704	0.21775721	0.13252618	-0.15561378	0.041458349
[12,]	-0.31506335	-0.221341539	-0.11059291	-0.19916285	0.11040505	-0.27479308	0.21173052	0.26591039	0.14049128	-0.69291811	-0.18923400	0.21199084	-0.126430943
[13,]	0.02023744	-0.578562805	-0.32463058	-0.04534612	-0.45795836	0.16548405	-0.16034775	-0.32977971	-0.37621746	-0.09441144	-0.01014882	0.18639049	0.023234496

Specific variances , factor loadings and commuanlities

```
> correlation.pa
```

	PA1	PA2	PA3	h2	u2
1	-0.96	-0.01	-0.24	0.98	0.0159
2	-0.42	0.58	-0.63	0.90	0.1048
3	-0.82	-0.37	-0.13	0.83	0.1726
4	-0.91	-0.29	0.17	0.94	0.0617
5	-0.96	0.06	0.16	0.94	0.0563
6	-0.40	0.81	-0.16	0.84	0.1634
7	-0.95	-0.23	-0.05	0.97	0.0318
8	-0.90	0.33	0.02	0.92	0.0804
9	-0.85	0.23	0.34	0.89	0.1094
10	-0.93	-0.05	-0.30	0.95	0.0496

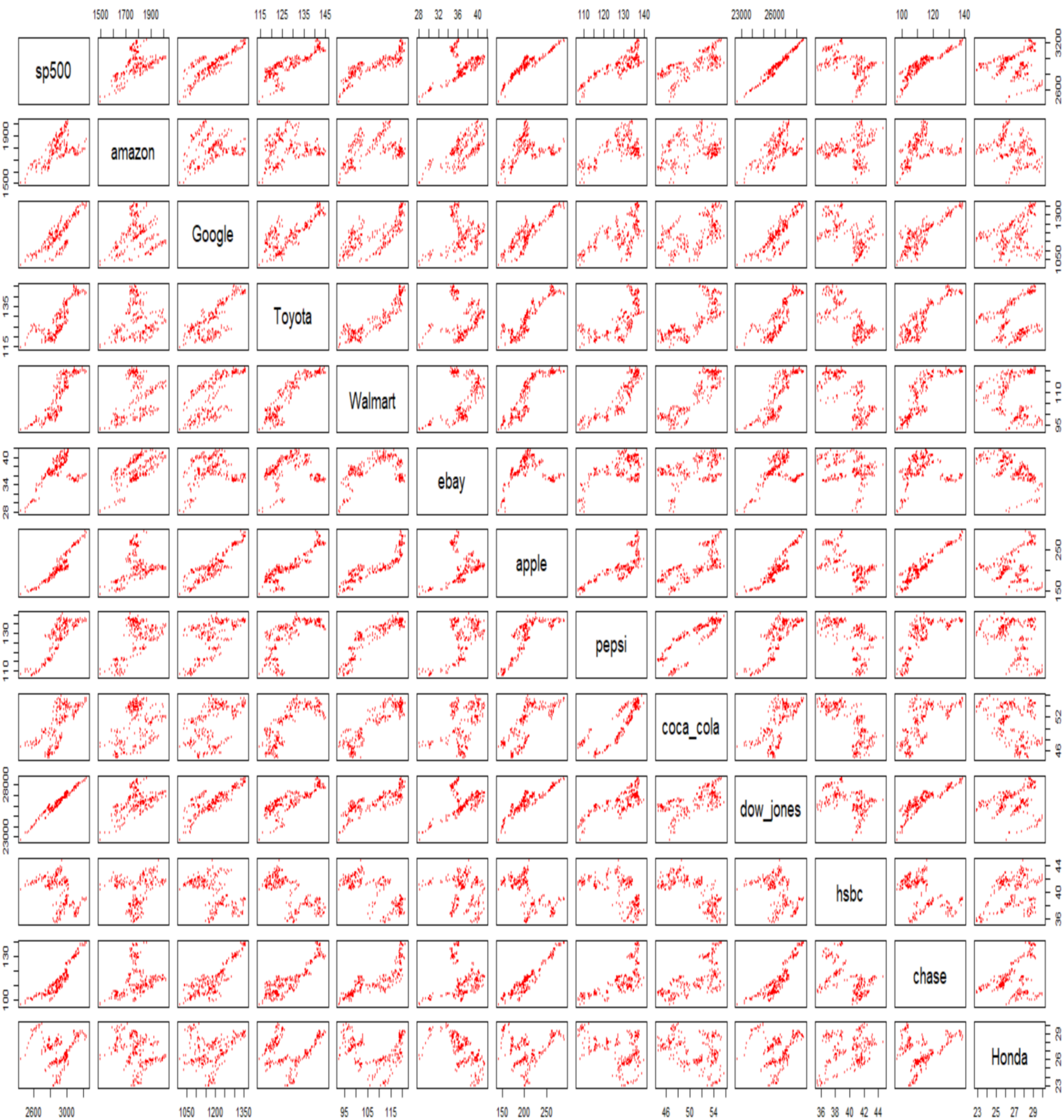
Residual matrix through PC method:-

```
> residual.PC
```

	sp500	amazon	Google	Toyota	Walmart	ebay	apple	pepsi	coca_cola	dow_jones	hsbc	chase	Honda
sp500	0.06240000	0.1268956492	0.0419889414	-0.05917476	-0.05292555	0.0513431833	0.027635582	-0.011404401	-0.103163812	0.09137286	0.15871708	0.03679947	0.0849278656
amazon	0.12689565	0.3824000000	0.0599581648	-0.08299040	-0.09941387	0.0003294783	0.022604048	-0.010401331	-0.180381854	0.13566180	0.43359038	0.08977720	0.2462082228
Google	0.04198894	0.0599581648	0.0181000000	-0.06773268	-0.08423768	0.0777154758	0.035202814	-0.027010856	-0.108011003	0.03508438	0.04039161	-0.01854180	0.0004634821
Toyota	-0.05917476	-0.0829903959	-0.0677326838	0.02610000	0.04703145	-0.0225156309	-0.033562704	-0.027240060	0.086354563	-0.06614699	-0.11221254	-0.03032769	-0.0243735220
Walmart	-0.05292555	-0.0994138688	-0.0842376827	0.04703145	0.01850000	-0.0425260495	-0.025069653	0.014635202	0.065692295	-0.05696416	-0.08100587	-0.01632334	-0.0304940674
ebay	0.05134318	0.0003294783	0.0777154758	-0.02251563	-0.04252605	0.0205000000	-0.006357484	-0.062530370	-0.103861673	0.10547718	0.08193108	-0.01937215	0.1111072594
apple	0.02763558	0.0226040482	0.0352028141	-0.03356270	-0.02506965	-0.0063574842	0.012800000	0.011412071	-0.038182256	0.01331306	0.02986813	0.01672527	-0.0065357001
pepsi	-0.01140440	-0.0104013309	-0.0270108562	-0.02724006	0.01463520	-0.0625303698	0.011412071	0.000700000	0.004904106	-0.03392571	0.01431459	0.01865576	-0.0156395844
coca_cola	-0.10316381	-0.1803818538	-0.1080110029	0.08635456	0.06569230	-0.1038616730	-0.038182256	0.004904106	0.115200000	-0.13612792	-0.18386833	-0.01906677	-0.1059664343
dow_jones	0.09137286	0.1356618047	0.0350843809	-0.06614699	-0.05696416	0.1054771778	0.013313057	-0.033925705	-0.136127919	0.08300000	0.19510272	0.02292066	0.1221579429
hsbc	0.15871708	0.4335903750	0.0403916148	-0.11221254	-0.08100587	0.0819310770	0.029868126	0.014314591	-0.183868333	0.19510272	0.49340000	0.12482778	0.2613904996
chase	0.03679947	0.0897771966	-0.0185418010	-0.03032769	-0.01632334	-0.0193721494	0.016725266	0.018655763	-0.019066765	0.02292066	0.12482778	0.02160000	0.0250198661
Honda	0.08492787	0.2462082228	0.0004634821	-0.02437352	-0.03049407	0.1111072594	-0.006535700	-0.015639584	-0.105966434	0.12215794	0.26139050	0.02501987	0.1497000000

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

Scatterplot:_



ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

Residual matrix for m=2 and m= 3 models

```
> #number of factors=3 #
> f=3
> # Residual matrix for f factor model#
> pred <- market.factor.analysis[[f]]$loadings%*%t(market.factor.analysis[[f]]$loadings) + diag(market.factor.analysis[[f]]$uniquenesses)
> pred
                sp500      amazon      Google      Toyota    Walmart        ebay      apple      pepsi  coca_cola   dow_jones        hsbc      chase       Honda
sp500      1.00087553 0.518050314  0.82982809  0.81913007  0.8701382  0.42817888  0.9388912  0.8470462  0.7097746  0.98312032 -0.474471484  0.9017306  0.04020764
amazon     0.51805031 0.999956490  0.24235472  0.12590904  0.3358920  0.68342562  0.3099982  0.5162160  0.2791928  0.52502045  0.005812444  0.2458124 -0.31352479
Google     0.82982809 0.242354720  0.99997402  0.78947945  0.7126111  0.08028714  0.8618319  0.5995635  0.5286732  0.82631728 -0.414191305  0.8672329  0.31552379
Toyota     0.81913007 0.125909042  0.78947945  1.00000166  0.8845266  0.08046344  0.9109120  0.7258854  0.7783822  0.78210025 -0.700131595  0.8953627  0.13185619
Walmart    0.87013816 0.335892027  0.71261110  0.88452660  1.0000009  0.38181876  0.8828487  0.8858576  0.8958294  0.82294947 -0.725497296  0.8272167 -0.16338879
ebay       0.42817888 0.683425618  0.08028714  0.08046344  0.3818188  1.00001830  0.2087440  0.5990686  0.4226614  0.40990458 -0.124642189  0.1093821 -0.62725586
apple      0.93889118 0.309998217  0.86183190  0.91091197  0.8828487  0.20874398  1.0000006  0.7762596  0.7257952  0.91868278 -0.575299384  0.9474854  0.16651643
pepsi      0.84704624 0.516216047  0.59956350  0.72588537  0.8858576  0.59906861  0.7762596  1.0000007  0.8483512  0.80511215 -0.605124705  0.6982885 -0.33855309
coca_cola  0.70977456 0.279192818  0.52867323  0.77838217  0.8958294  0.42266137  0.7257952  0.8483512  1.0000006  0.64758006 -0.765289076  0.6529119 -0.35755565
dow_jones  0.98312032 0.525020446  0.82631728  0.78210025  0.8229495  0.40990458  0.9186828  0.8051121  0.6475801  0.99999939 -0.412185997  0.8877398  0.08768077
hsbc      -0.47447148 0.005812444 -0.41419130 -0.70013159 -0.7254973 -0.12464219 -0.5752994 -0.6051247 -0.7652891 -0.41218600  0.999999798 -0.5320364  0.20974298
chase      0.90173064 0.245812413  0.86723292  0.89536271  0.8272167  0.10938213  0.9474854  0.6982885  0.6529119  0.88773980 -0.532036402  1.0000005  0.26918825
Honda      0.04020764 -0.313524792 0.31552379  0.13185619 -0.1633888 -0.62725586  0.1665164 -0.3385531 -0.3575556  0.08768077  0.209742976  0.2691883  1.00000014
>
>
> #number of factors=2 #
> f2=2
> # Residual matrix for f factor model#
> predl <- market.factor.analysis[[f]]$loadings%*%t(market.factor.analysis[[f]]$loadings) + diag(market.factor.analysis[[f2]]$uniquenesses)
> predl
                sp500      amazon      Google      Toyota    Walmart        ebay      apple      pepsi  coca_cola   dow_jones        hsbc      chase       Honda
sp500      1.06045783 0.518050314  0.82982809  0.81913007  0.8701382  0.42817888  0.9388912  0.8470462  0.7097746  0.98312032 -0.474471484  0.9017306  0.04020764
amazon     0.51805031 1.348688425  0.24235472  0.12590904  0.3358920  0.68342562  0.3099982  0.5162160  0.2791928  0.52502045  0.005812444  0.2458124 -0.31352479
Google     0.82982809 0.242354720  0.99042233  0.78947945  0.7126111  0.08028714  0.8618319  0.5995635  0.5286732  0.82631728 -0.414191305  0.8672329  0.31552379
Toyota     0.81913007 0.125909042  0.78947945  1.11099807  0.8845266  0.08046344  0.9109120  0.7258854  0.7783822  0.78210025 -0.700131595  0.8953627  0.13185619
Walmart    0.87013816 0.335892027  0.71261110  0.88452660  1.0528378  0.38181876  0.8828487  0.8858576  0.8958294  0.82294947 -0.725497296  0.8272167 -0.16338879
ebay       0.42817888 0.683425618  0.08028714  0.08046344  0.3818188  1.23369544  0.2087440  0.5990686  0.4226614  0.40990458 -0.124642189  0.1093821 -0.62725586
apple      0.93889118 0.309998217  0.86183190  0.91091197  0.8828487  0.20874398  0.9915985  0.7762596  0.7257952  0.91868278 -0.575299384  0.9474854  0.16651643
pepsi      0.84704624 0.516216047  0.59956350  0.72588537  0.8858576  0.59906861  0.7762596  0.9713639  0.8483512  0.80511215 -0.605124705  0.6982885 -0.33855309
coca_cola  0.70977456 0.279192818  0.52867323  0.77838217  0.8958294  0.42266137  0.7257952  0.8483512  1.1427037  0.64758006 -0.765289076  0.6529119 -0.35755565
dow_jones  0.98312032 0.525020446  0.82631728  0.78210025  0.8229495  0.40990458  0.9186828  0.8051121  0.6475801  1.10423699 -0.412185997  0.8877398  0.08768077
hsbc      -0.47447148 0.005812444 -0.41419130 -0.70013159 -0.7254973 -0.12464219 -0.5752994 -0.6051247 -0.7652891 -0.41218600  1.331518337 -0.5320364  0.20974298
chase      0.90173064 0.245812413  0.86723292  0.89536271  0.8272167  0.10938213  0.9474854  0.6982885  0.6529119  0.88773980 -0.532036402  1.0012783  0.26918825
Honda      0.04020764 -0.313524792 0.31552379  0.13185619 -0.1633888 -0.62725586  0.1665164 -0.3385531 -0.3575556  0.08768077  0.209742976  0.2691883  1.01795617
>
```

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

P values when factor is 1, 2, 3, 4, 5, 6, 7 , 8 respectively.

Factor = 1 and 2

```
Loadings:
          Factor1
sp500      0.998
amazon     0.504
Google     0.838
Toyota     0.830
Walmart    0.875
ebay       0.414
apple      0.946
pepsi      0.847
coca_cola  0.716
dow_jones  0.983
hsbc      -0.489
chase      0.909
Honda

               Factor1
SS loadings      7.740
Proportion Var   0.595

Test of the hypothesis that 1 factor is sufficient.
The chi square statistic is 3353.47 on 65 degrees of freedom.
The p-value is 0
```

```
Loadings:
          Factor1 Factor2
sp500      0.929   0.271
amazon     0.265   0.503
Google     0.902
Toyota     0.905   0.102
Walmart    0.841   0.441
ebay       0.117   0.768
apple      0.979   0.124
pepsi      0.724   0.644
coca_cola  0.661   0.579
dow_jones  0.910   0.211
hsbc      -0.532  -0.352
chase      0.971
Honda      0.269  -0.858

               Factor1 Factor2
SS loadings      7.330   2.796
Proportion Var   0.564   0.215
Cumulative Var   0.564   0.779

Test of the hypothesis that 2 factors are sufficient.
The chi square statistic is 2417.91 on 53 degrees of freedom.
The p-value is 0
```

Factor = 3 and 4

```
    0.005     0.328     0.191     0.060     0.046     0.163     0.035

Loadings:
          Factor1 Factor2 Factor3
sp500      0.887   0.281   0.361
amazon     0.309           0.753
Google     0.877   0.201
Toyota     0.764   0.593
Walmart    0.663   0.668   0.262
ebay               0.151   0.900
apple      0.893   0.397
pepsi      0.562   0.561   0.529
coca_cola  0.423   0.803   0.303
dow_jones  0.902   0.198   0.354
hsbc      -0.287  -0.809
chase      0.913   0.334
Honda      0.451  -0.435  -0.659

               Factor1 Factor2 Factor3
SS loadings      5.903   3.060   2.520
Proportion Var   0.454   0.235   0.194
Cumulative Var   0.454   0.689   0.883

Test of the hypothesis that 3 factors are sufficient.
The chi square statistic is 1198.74 on 42 degrees of freedom.
The p-value is 7.59e-224
```

```
Loadings:
          Factor1 Factor2 Factor3 Factor4
sp500      0.929   0.344   0.113
amazon     0.298   0.762  -0.281   0.209
Google     0.895           0.138  -0.133
Toyota     0.851           0.411   0.126
Walmart    0.773   0.272   0.461   0.226
ebay       0.109   0.919   0.155  -0.134
apple      0.949           0.238
pepsi      0.659   0.527   0.372   0.297
coca_cola  0.561   0.318   0.600   0.348
dow_jones  0.930   0.340
hsbc      -0.416          -0.903
chase      0.961           0.100   0.213
Honda      0.357  -0.658  -0.511  -0.155

               Factor1 Factor2 Factor3 Factor4
SS loadings      6.825   2.557   2.160   0.441
Proportion Var   0.525   0.197   0.166   0.034
Cumulative Var   0.525   0.722   0.888   0.922

Test of the hypothesis that 4 factors are sufficient.
The chi square statistic is 847.81 on 32 degrees of freedom.
The p-value is 1.62e-157

[[5]]
```

Factor = 5 and 6

```
Loadings:
          Factor1 Factor2 Factor3 Factor4 Factor5
sp500      0.920   0.340   0.146   0.102
amazon     0.314   0.728  -0.304   0.244
Google     0.909           0.182  -0.123
Toyota     0.795           0.480   0.267   0.191
Walmart    0.722   0.280   0.504   0.331
ebay       0.102   0.922   0.144           0.129
apple      0.944           0.278   0.106
pepsi      0.643   0.529   0.370   0.299  -0.160
coca_cola  0.504   0.331   0.625   0.407
dow_jones  0.917   0.338                   0.175
hsbc      -0.374          -0.906
chase      0.938           0.150   0.252
Honda      0.355  -0.697  -0.445           0.352

               Factor1 Factor2 Factor3 Factor4 Factor5
SS loadings      6.482   2.574   2.306   0.606   0.254
Proportion Var   0.499   0.198   0.177   0.047   0.020
Cumulative Var   0.499   0.697   0.874   0.921   0.940

Test of the hypothesis that 5 factors are sufficient.
The chi square statistic is 403.05 on 23 degrees of freedom.
The p-value is 4.18e-71
```

```
Loadings:
          Factor1 Factor2 Factor3 Factor4 Factor5 Factor6
sp500      0.891   0.314   0.176   0.254
amazon     0.271           0.375   0.883
Google     0.892   0.229                          -0.221
Toyota     0.685   0.695  -0.134           0.150
Walmart    0.644   0.695   0.200   0.120           0.142
ebay       0.174           0.842   0.314   0.249
apple      0.896   0.420
pepsi      0.603   0.527   0.427   0.280  -0.153   0.180
coca_cola  0.420   0.792   0.295   0.121
dow_jones  0.901   0.242   0.165   0.223   0.204
hsbc      -0.346  -0.749  -0.236   0.372           0.208
chase      0.860   0.403  -0.175   0.128           0.133
Honda      0.296  -0.257  -0.826           0.278

               Factor1 Factor2 Factor3 Factor4 Factor5 Factor6
SS loadings      5.681   3.054   2.012   1.266   0.247   0.182
Proportion Var   0.437   0.235   0.155   0.097   0.019   0.014
Cumulative Var   0.437   0.672   0.827   0.924   0.943   0.957

Test of the hypothesis that 6 factors are sufficient.
The chi square statistic is 255.22 on 15 degrees of freedom.
The p-value is 1.05e-45
```

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

Factor = 7 and 8

```
Loadings:
          Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7
sp500       0.878   0.319           0.299   0.152
amazon      0.249           0.240   0.933
Google      0.888   0.233  -0.105                  -0.173
Toyota      0.676   0.713  -0.154
Walmart     0.634   0.700   0.156   0.155   0.120           0.177
ebay        0.148           0.634   0.445   0.517
apple       0.901   0.409
pepsi       0.613   0.494   0.406   0.336                   0.241
coca_cola   0.442   0.762   0.334   0.143            0.106
dow_jones   0.872   0.268           0.278   0.285
hsbc       -0.370  -0.723  -0.312   0.342            0.195
chase       0.869   0.394  -0.158   0.111            0.195
Honda       0.248  -0.187  -0.921  -0.144

               Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7
SS loadings      5.598   2.940   1.773   1.548   0.405   0.132   0.114
Proportion Var   0.431   0.226   0.136   0.119   0.031   0.010   0.009
Cumulative Var   0.431   0.657   0.793   0.912   0.943   0.954   0.962

Test of the hypothesis that 7 factors are sufficient.
The chi square statistic is 148 on 8 degrees of freedom.
The p-value is 5.13e-28
```

```
Loadings:
          Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7 Factor8
sp500       0.865   0.337   0.281           0.212
amazon      0.241           0.914  -0.144   0.283
Google      0.900   0.228           0.101                  -0.152  -0.178
Toyota      0.671   0.714           0.173
Walmart     0.617   0.713   0.149           0.169   0.132
ebay        0.117   0.126   0.351  -0.355   0.846
apple       0.891   0.423
pepsi       0.581   0.529   0.338  -0.311   0.229   0.336
coca_cola   0.421   0.779   0.150  -0.268   0.149           0.121
dow_jones   0.858   0.287   0.244           0.287                   0.164
hsbc       -0.356  -0.746   0.322   0.291                   0.211
chase       0.869   0.392   0.119   0.127                   0.204
Honda       0.268  -0.217  -0.156   0.866  -0.316

               Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7 Factor8
SS loadings       5.46   3.108   1.406   1.221   1.139   0.154   0.129   0.077
Proportion Var    0.42   0.239   0.108   0.094   0.088   0.012   0.010   0.006
Cumulative Var    0.42   0.659   0.767   0.861   0.949   0.961   0.971   0.976

Test of the hypothesis that 8 factors are sufficient.
The chi square statistic is 103.16 on 2 degrees of freedom.
The p-value is 3.97e-23
```

Factor with Varimax Rotation:-

```
Call:
factanal(x = stocks[, -1], factors = nf, rotation = "varimax")

Uniquenesses:
    sp500    amazon    Google    Toyota   Walmart      ebay     apple     pepsi coca_cola dow_jones      hsbc     chase     Honda
    0.005     0.328     0.191     0.060     0.046     0.163     0.035     0.090     0.085     0.023     0.262     0.055     0.174

Loadings:
          Factor1 Factor2 Factor3
sp500       0.887   0.281   0.361
amazon      0.309           0.753
Google      0.877   0.201
Toyota      0.764   0.593
Walmart     0.663   0.668   0.262
ebay                0.151   0.900
apple       0.893   0.397
pepsi       0.562   0.561   0.529
coca_cola   0.423   0.803   0.303
dow_jones   0.902   0.198   0.354
hsbc       -0.287  -0.809
chase       0.913   0.334
Honda       0.451  -0.435  -0.659

               Factor1 Factor2 Factor3
SS loadings      5.903   3.060   2.520
Proportion Var   0.454   0.235   0.194
Cumulative Var   0.454   0.689   0.883

Test of the hypothesis that 3 factors are sufficient.
The chi square statistic is 1198.74 on 42 degrees of freedom.
The p-value is 7.59e-224
```

Factor with no Rotation:-

```
Call:
factanal(x = stocks[, -1], factors = nf, rotation = "none")

Uniquenesses:
    sp500    amazon    Google    Toyota   Walmart      ebay     apple     pepsi coca_cola dow_jones      hsbc     chase     Honda
    0.005     0.328     0.191     0.060     0.046     0.163     0.035     0.090     0.085     0.023     0.262     0.055     0.174

Loadings:
          Factor1 Factor2 Factor3
sp500       0.994
amazon      0.474   0.354   0.567
Google      0.840  -0.312
Toyota      0.861  -0.102  -0.434
Walmart     0.903   0.222  -0.301
ebay        0.397   0.699   0.437
apple       0.958  -0.142  -0.166
pepsi       0.860   0.409
coca_cola   0.751   0.432  -0.405
dow_jones   0.976           0.148
hsbc       -0.532  -0.264   0.620
chase       0.921  -0.256  -0.176
Honda              -0.908

               Factor1 Factor2 Factor3
SS loadings      7.938   2.108   1.438
Proportion Var   0.611   0.162   0.111
Cumulative Var   0.611   0.773   0.883

Test of the hypothesis that 3 factors are sufficient.
The chi square statistic is 1198.74 on 42 degrees of freedom.
The p-value is 7.59e-224
```

Approximate correlation/covariance matrix for m=3 factor model:-

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

```
> stocks.cor <- cor(stocks[,-1])
> stocks.cor
```

	sp500	amazon	Google	Toyota	Walmart	ebay	apple	pepsi	coca_cola	dow_jones	hsbc	chase	Honda
sp500	1.00000000	0.5242956	0.8328889	0.8173252	0.8680744	0.42724318	0.9419356	0.8492956	0.7105362	0.9846729	-0.4750829	0.90419947	0.03612787
amazon	0.52429565	1.0000000	0.1897582	0.1310096	0.3385861	0.63812948	0.2882040	0.5589987	0.3100181	0.4972618	0.1679904	0.27057720	-0.28939178
Google	0.83288894	0.1897582	1.0000000	0.7857673	0.6807623	0.10601548	0.8993028	0.5888891	0.5038890	0.8161844	-0.5082084	0.84525820	0.27686348
Toyota	0.81732524	0.1310096	0.7857673	1.0000000	0.9032314	0.10658437	0.8976373	0.6960599	0.7931546	0.7946530	-0.7186125	0.88727231	0.17622648
Walmart	0.86807445	0.3385861	0.6807623	0.9032314	1.0000000	0.39007395	0.8731303	0.8984352	0.8954923	0.8328358	-0.7134059	0.82727666	-0.14089407
ebay	0.42724318	0.6381295	0.1060155	0.1065844	0.3900740	1.00000000	0.1873425	0.5647696	0.4224383	0.4369772	-0.1658689	0.06522785	-0.62569274
apple	0.94193558	0.2882040	0.8993028	0.8976373	0.8731303	0.18734252	1.0000000	0.7905121	0.7164177	0.9083131	-0.6017319	0.94992527	0.13886430
pepsi	0.84929560	0.5589987	0.5888891	0.6960599	0.8984352	0.56476963	0.7905121	1.0000000	0.8458041	0.7865743	-0.5730854	0.71645576	-0.36003958
coca_cola	0.71053619	0.3100181	0.5038890	0.7931546	0.8954923	0.42243833	0.7164177	0.8458041	1.0000000	0.6428721	-0.7402683	0.66773323	-0.35936643
dow_jones	0.98467286	0.4972618	0.8161844	0.7946530	0.8328358	0.43697718	0.9083131	0.7865743	0.6428721	1.0000000	-0.4196973	0.87692066	0.11035794
hsbc	-0.47508292	0.1679904	-0.5082084	-0.7186125	-0.7134059	-0.16586892	-0.6017319	-0.5730854	-0.7402683	-0.4196973	1.0000000	-0.47597222	0.28339050
chase	0.90419947	0.2705772	0.8452582	0.8872723	0.8272767	0.06522785	0.9499253	0.7164558	0.6677332	0.8769207	-0.4759722	1.00000000	0.27021987
Honda	0.03612787	-0.2893918	0.2768635	0.1762265	-0.1408941	-0.62569274	0.1388643	-0.3600396	-0.3593664	0.1103579	0.2833905	0.27021987	1.00000000

```
>
```

	Factor1	Factor2	Factor3
SS loadings	5.903	3.060	2.520
Proportion Var	0.454	0.235	0.194
Cumulative Var	0.454	0.689	0.883

```
Test of the hypothesis that 3 factors are sufficient.
The chi square statistic is 1198.74 on 42 degrees of freedom.
The p-value is 7.59e-224
```

Importance of components:-

Importance of components:

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9	Comp.10	Comp.11	Comp.12	Comp.13
Standard deviation	2.8678565	1.5213854	1.1913132	0.62350585	0.49399450	0.38366895	0.306610345	0.261785492	0.201647252	0.159453202	0.137187111	0.1098346205	0.0453210556
Proportion of Variance	0.6326616	0.1780472	0.1091713	0.02990458	0.01877158	0.01132322	0.007231531	0.005271665	0.003127816	0.001955794	0.001447716	0.0009279726	0.0001579999
Cumulative Proportion	0.6326616	0.8107088	0.9198801	0.94978470	0.96855628	0.97987951	0.987111036	0.992382701	0.995510518	0.997466312	0.998914028	0.9998420001	1.0000000000

Principal Components Output:-

Loadings:

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9	Comp.10	Comp.11	Comp.12	Comp.13
sp500	0.336		0.202	0.102		0.140		0.280	0.180	0.102		0.101	0.824
amazon	0.145	-0.378	0.525	-0.319	0.199	-0.588	0.200					-0.119	
Google	0.287	0.241	0.109	0.490	0.425	-0.294	-0.339	-0.376	-0.159		0.229		
Toyota	0.316	0.193	-0.145	-0.143	-0.315	-0.334			-0.407	-0.125	-0.329	0.564	
Walmart	0.333		-0.134	-0.196	-0.234	0.109	0.242	-0.217	-0.519	0.324	0.258	-0.459	
ebay	0.139	-0.530	0.134	0.505	-0.484		-0.155	-0.146		-0.297	-0.177	-0.161	
apple	0.333	0.153			0.230	0.142				0.217	-0.762	-0.329	-0.184
pepsi	0.314	-0.218		-0.193	0.212	0.345	0.312	-0.589	0.236	-0.172		0.345	
coca_cola	0.296	-0.152	-0.289	-0.377	-0.164	-0.141	-0.617		0.424	0.199			
dow_jones	0.323		0.250	0.207	-0.203	0.198	0.119	0.356	0.135	0.343	0.311	0.292	-0.500
hsbc	-0.230		0.591	-0.257		0.350	-0.442	-0.190	-0.290	0.218	-0.133	0.156	
chase	0.315	0.221	0.111	-0.199	0.110	0.275	-0.212	0.266	-0.140	-0.693	0.189	-0.212	-0.126
Honda		0.579	0.325		-0.458	-0.165	0.160	-0.330	0.376			-0.186	

```
>
```

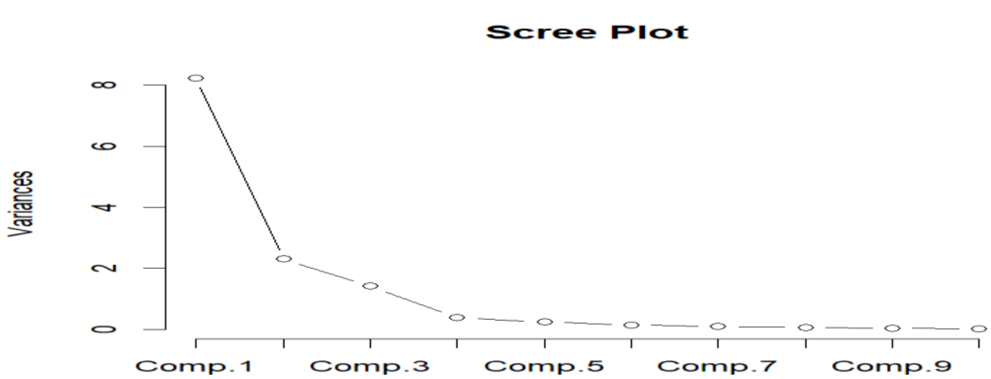
ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

PC Loadings:-

```
Loadings:
           Comp.1 Comp.2 Comp.3
sp500       0.336         0.202
amazon      0.145 -0.378  0.525
Google      0.287  0.241  0.109
Toyota      0.316  0.193 -0.145
Walmart     0.333        -0.134
ebay        0.139 -0.530  0.134
apple       0.333  0.153
pepsi       0.314 -0.218
coca_cola   0.296 -0.152 -0.289
dow_jones   0.323         0.250
hsbc       -0.230         0.591
chase       0.315  0.221  0.111
Honda              0.579  0.325
```

	Minimum	Maximum	Variance
Amazon	1500	2021	10577.52
Google	1016	1361	6689.13
Toyota	114.7	145.1	67.82
Walmart	92.86	121.28	79.91
eBay	28.07	41.57	7.17
Apple	142.2	291.5	1173.76
Pepsi	107.3	140.3	82.52
Coca-Cola	44.69	55.77	10.66
HSBC	35.53	44.7	4.79
Chase	97.11	139.14	106.69
Honda	22.9	30.07	2.73
S&P500	2448	3240	22949.58
Dow Jones	22686	28645	1183499.76

```
     sp500           amazon          Google          Toyota          Walmart            ebay            apple            pepsi
 Min.   :2448   Min.   :1500   Min.   :1016   Min.   :114.7   Min.   : 92.86   Min.   :28.07   Min.   :142.2   Min.   :107.3
 1st Qu.:2821   1st Qu.:1735   1st Qu.:1121   1st Qu.:121.8   1st Qu.: 99.50   1st Qu.:35.83   1st Qu.:185.6   1st Qu.:121.8
 Median :2918   Median :1786   Median :1185   Median :126.4   Median :109.51   Median :37.17   Median :202.9   Median :130.9
 Mean   :2910   Mean   :1788   Mean   :1187   Mean   :128.4   Mean   :108.30   Mean   :37.15   Mean   :207.7   Mean   :127.8
 3rd Qu.:3000   3rd Qu.:1855   3rd Qu.:1239   3rd Qu.:135.8   3rd Qu.:117.57   3rd Qu.:39.27   3rd Qu.:223.2   3rd Qu.:135.4
 Max.   :3240   Max.   :2021   Max.   :1361   Max.   :145.1   Max.   :121.28   Max.   :41.57   Max.   :291.5   Max.   :140.3
   coca_cola       dow_jones          hsbc           chase             Honda
 Min.   :44.69   Min.   :22686   Min.   :35.53   Min.   : 97.11   Min.   :22.90
 1st Qu.:47.60   1st Qu.:25759   1st Qu.:38.13   1st Qu.:105.44   1st Qu.:25.80
 Median :51.65   Median :26394   Median :40.85   Median :111.26   Median :27.02
 Mean   :50.80   Mean   :26359   Mean   :40.03   Mean   :113.63   Mean   :26.92
 3rd Qu.:53.85   3rd Qu.:27073   3rd Qu.:41.68   3rd Qu.:117.70   3rd Qu.:28.19
 Max.   :55.77   Max.   :28645   Max.   :44.70   Max.   :139.14   Max.   :30.07
```



ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

R OUTPUT for Q 2:-

```
> # doing manual calculation #
> t(a) %*%t (newdata)
           [,1]
[1,] -9.007773
>
> # using MASS:lda decision #
> predict(disc, newdata = newdata)$class
[1] 1
Levels: 0 1
```

LDA Calculations:-

Mass LDA CALCULATIONS

```
Coefficients of linear discriminants:
                                   LD1
Pregnancies               0.0938638298
Glucose                   0.0269863520
BloodPressure            -0.0106293929
SkinThickness             0.0007043468
Insulin                  -0.0008229296
BMI                       0.0603702056
DiabetesPedigreeFunction  0.6711517147
Age                       0.0119490869
```

Manual calculations

```
                                 [,1]
Pregnancies              -0.148436901
Glucose                  -0.046585207
BloodPressure             0.020449175
SkinThickness            -0.006120851
Insulin                   0.003017222
BMI                      -0.078964735
DiabetesPedigreeFunction -1.171166479
Age                       0.001480299
```

```
> #Predicting for a new data #
> predict(fit.diabetes, newdata = newdata, type = "response")
        1
0.5780861
```

```
> predict(fit.diabetes, newdata = newdata, type = "response")
        1
0.5780861
>
> 10/(10+11)
[1] 0.6510417
```

FOR 2B CALCULATIONS:-

FOR APER CALCULATIONS:-

```
> table(Outcome, pred.group)
       pred.group
Outcome   0   1
      0 446  54
      1 112 156
```

FOR AER CALCULATIONS:-

```
Outcome   0   1
      0 442  58
      1 115 153
```

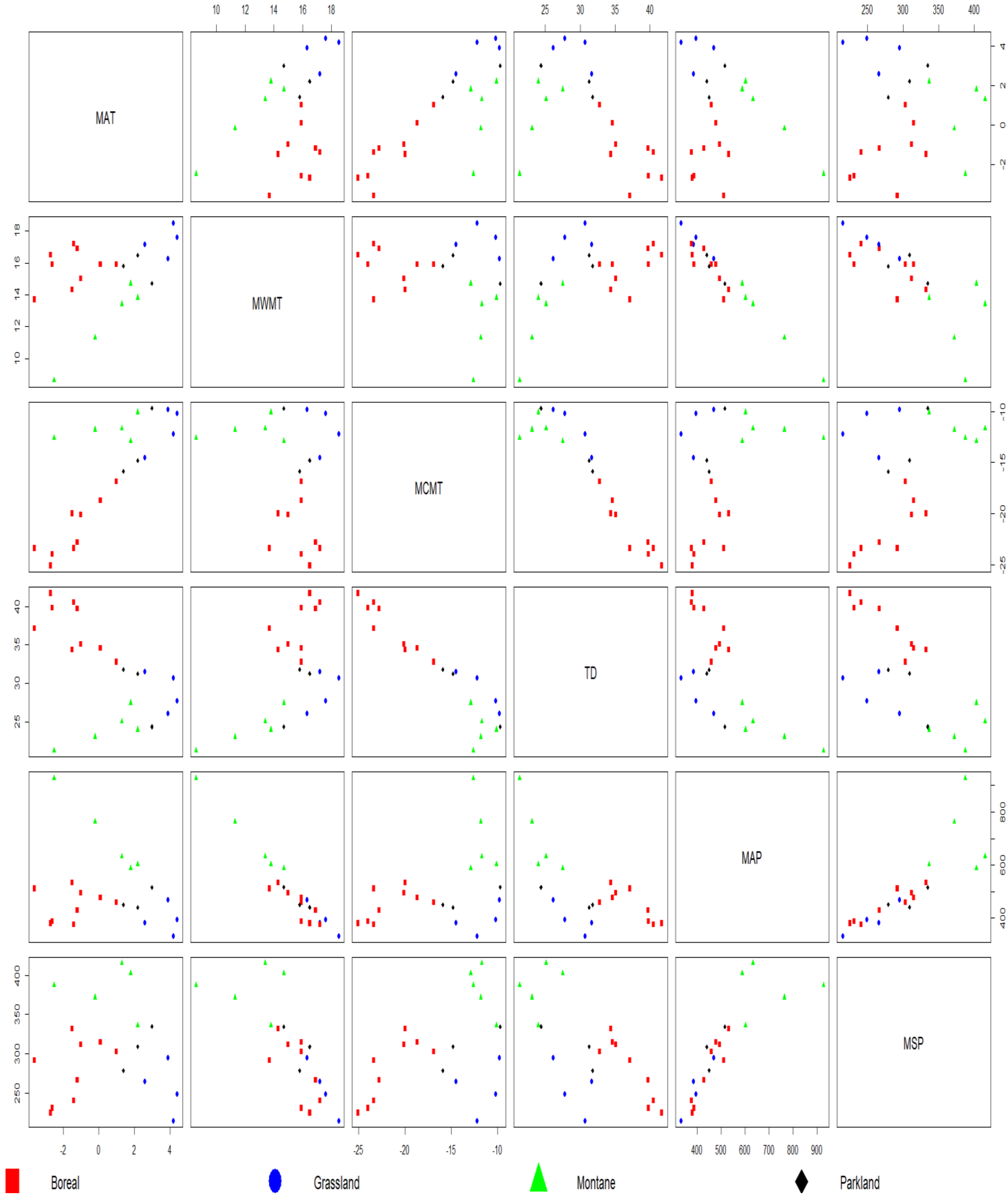
ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

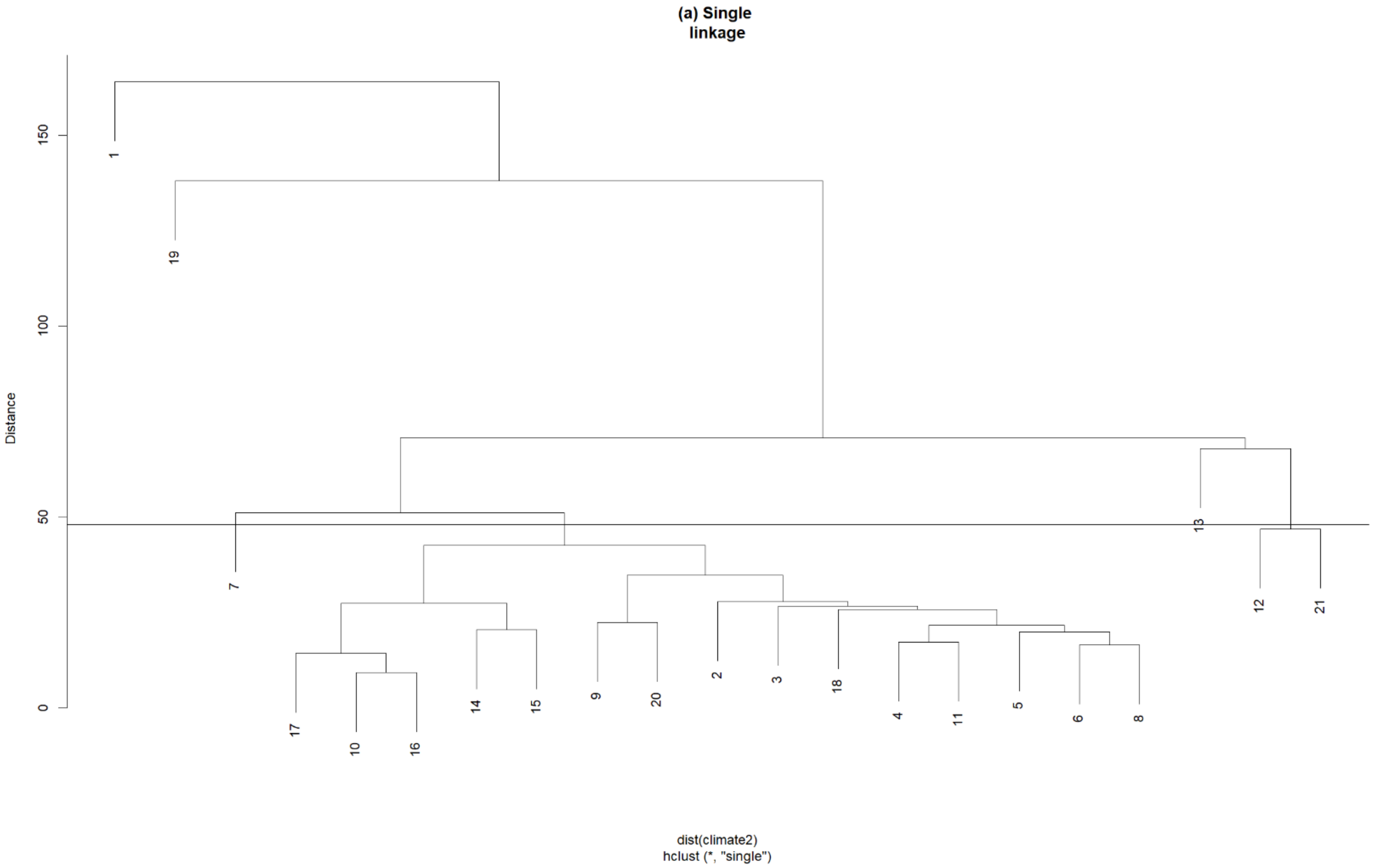
FOR 2C AER AND APER CALCULATIONS: -

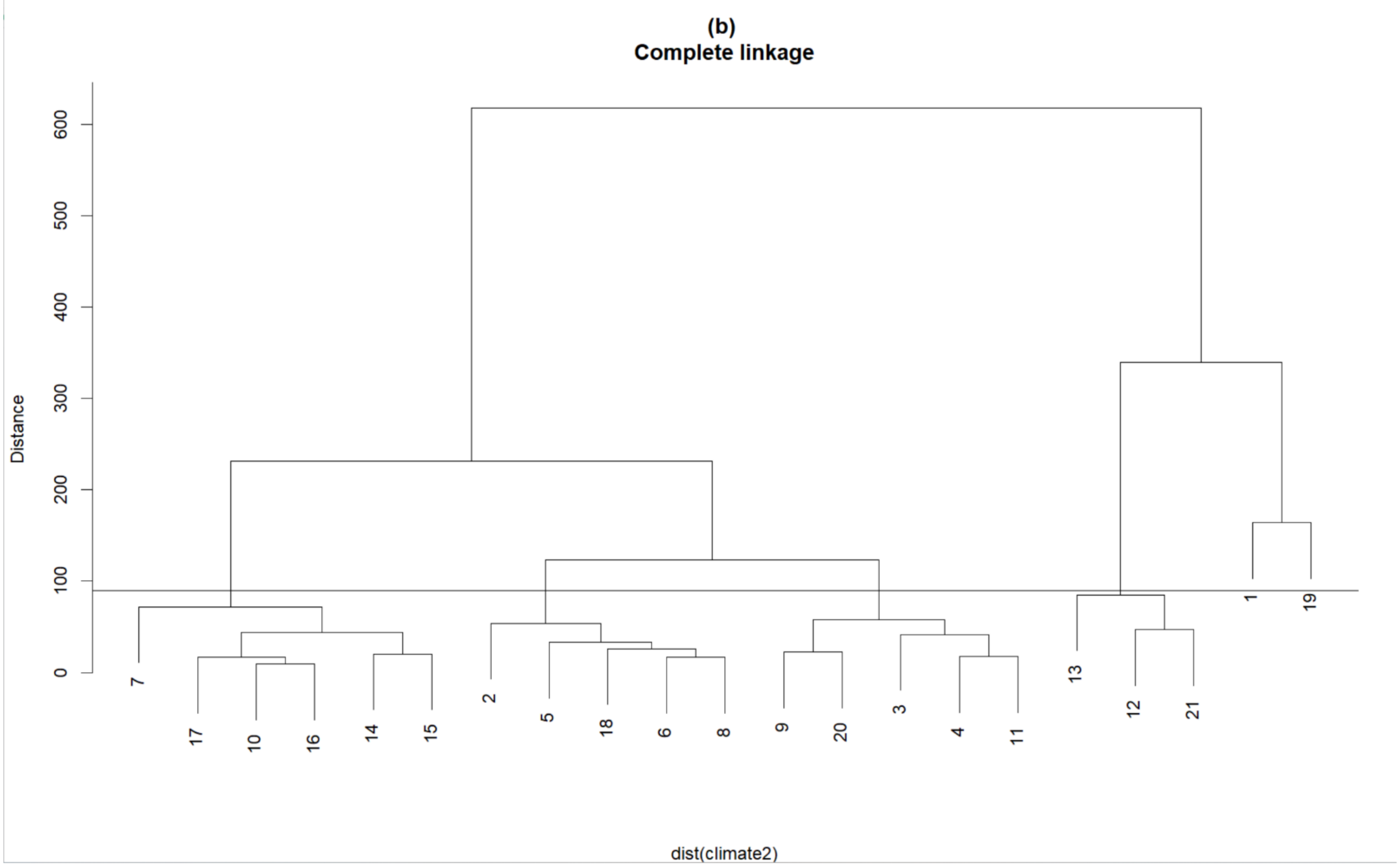
```
> table(Outcome, (predict(fit.diabetes, type = "response") > 10/(10+11)))

Outcome FALSE TRUE
      0   472   28
      1   146  122
>
>
> # doing Cross-Validation with Leave-one-out method #
> newpred = numeric(length(Outcome)) # array of size(#obs)
> for (i in 1:length(Outcome)) {
+   newdat = diabetes[-i,]
+   newfit = glm(Outcome ~
+                  Pregnancies+Glucose+BloodPressure+SkinThickness+Insulin+BMI+DiabetesPedigreeFunction+Age,
+                family=binomial, data = newdat)
+   newpred[i] = predict(newfit, newdat = data.frame(diabetes[i,-9]), type="response")
+ }
>
> table(Outcome, (newpred > 10/(10+11)))

Outcome FALSE TRUE
      0   469   31
      1   147  121
> # ABOVE ---> Prior probability same as prob in existing data.#
> # Below ---> Prior probability = 0.5 for each case#
> # Plug-in estimate #
> table(Outcome, (predict(fit.diabetes, type = "response") > 0.5))

Outcome FALSE TRUE
      0   445   55
      1   112  156
>
> # Cross-Validation with Leave-one-out method #
> newpred = numeric(length(Outcome)) # array of size(#obs)
> for (i in 1:length(Outcome)) {
+   newdat = diabetes[-i,]
+   newfit = glm(Outcome ~
+                  Pregnancies+Glucose+BloodPressure+SkinThickness+Insulin+BMI+DiabetesPedigreeFunction+Age,
+                family=binomial, data = newdat)
+   newpred[i] = predict(newfit, newdat = data.frame(diabetes[i,-9]), type="response")
+ }
> table(Outcome, (newpred > 0.5))

Outcome FALSE TRUE
      0   443   57
      1   114  154
>
```

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

FOR 3 R OUTPUTS:-




(a) Single
linkage
Distance
0
50
100
150
1
19
7
13
12
21
17
10
16
14
15
9
20
2
3
18
4
11
5
6
8
dist(climate2)
hclust (*, "single")


(b)
Complete linkage
Distance
0
100
200
300
400
500
600
7
17
10
16
14
15
2
5
18
6
8
9
20
3
4
11
13
12
21
1
19
dist(climate2)

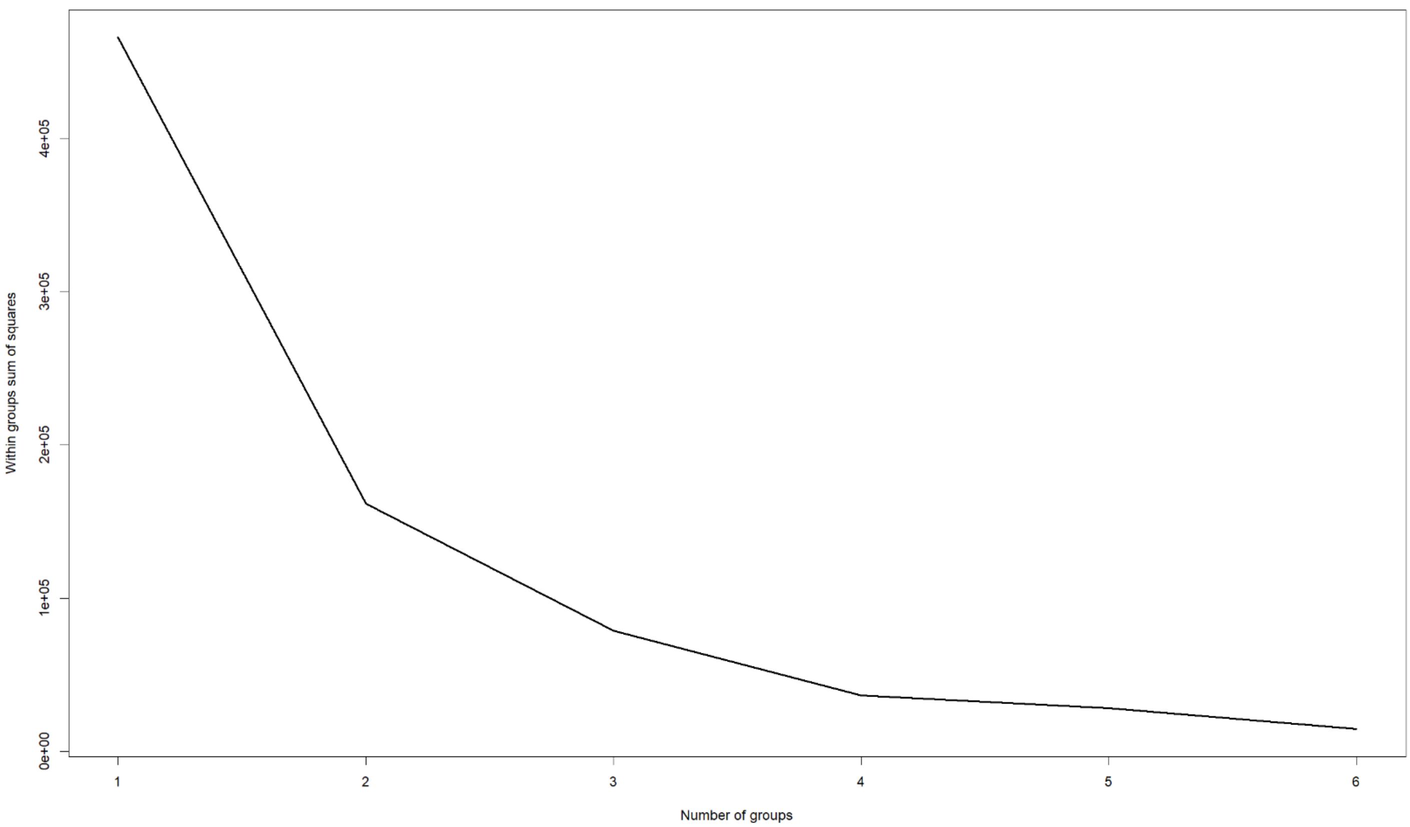
ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

MEAN VECTOR OF CLUSTERS:-

```
> # Mean vectors for each cluster#
>
> # Mean vectors for Single Linkage cluster#
> colMeans(climate2[cluster.single == 1,])
  MAT  MWMT  MCMT    TD   MAP   MSP
 -2.5   8.6 -12.6  21.3 927.2 387.2
> colMeans(climate2[cluster.single == 2,])
        MAT        MWMT        MCMT          TD         MAP         MSP
  0.3066667  15.9600000 -17.9533333  33.9133333 447.0800000 283.1866667
> colMeans(climate2[cluster.single == 3,])
  MAT  MWMT  MCMT    TD   MAP   MSP
  4.2  18.5 -12.2  30.7 333.6 214.9
> colMeans(climate2[cluster.single == 4,])
   MAT   MWMT   MCMT     TD    MAP    MSP
  1.55  14.05 -12.30  26.30 610.50 408.75
> colMeans(climate2[cluster.single == 5,])
  MAT  MWMT  MCMT    TD   MAP   MSP
  2.2  13.8 -10.1  24.0 601.6 336.1
> colMeans(climate2[cluster.single == 6,])
  MAT  MWMT  MCMT    TD   MAP   MSP
 -0.2  11.3 -11.8  23.1 764.0 371.6
>
> # Mean vectors for Complete Linkage cluster#
> colMeans(climate2[cluster.complete == 1,])
  MAT  MWMT  MCMT    TD   MAP   MSP
 -2.5   8.6 -12.6  21.3 927.2 387.2
> colMeans(climate2[cluster.complete == 2,])
   MAT   MWMT   MCMT     TD    MAP    MSP
  1.46  16.28 -16.04  32.34 450.10 290.42
> colMeans(climate2[cluster.complete == 3,])
   MAT   MWMT   MCMT     TD    MAP    MSP
 -0.60  14.72 -18.38  33.12 506.60 316.94
> colMeans(climate2[cluster.complete == 4,])
      MAT      MWMT      MCMT        TD       MAP       MSP
  0.75000  17.15000 -18.23333  35.35000 376.05000 237.65000
> colMeans(climate2[cluster.complete == 5,])
       MAT       MWMT       MCMT         TD        MAP        MSP
  1.766667  13.966667 -11.566667  25.533333 607.533333 384.533333
> colMeans(climate2[cluster.complete == 6,])
  MAT  MWMT  MCMT    TD   MAP   MSP
 -0.2  11.3 -11.8  23.1 764.0 371.6
```

K-MEANS OUTPUT

```
K-means clustering with 4 clusters of sizes 2, 3, 7, 9

Cluster means:
         MAT     MWMT      MCMT       TD      MAP      MSP
1 -1.3500000  9.95000 -12.20000 22.20000 845.6000 379.4000
2  1.7666667 13.96667 -11.56667 25.53333 607.5333 384.5333
3  0.4714286 17.11429 -18.88571 35.97143 383.5714 241.7429
4  0.6111111 15.34444 -16.58889 31.95556 483.8667 307.8333

Clustering vector:
 [1] 1 3 4 4 4 4 3 4 4 3 4 2 2 3 3 3 3 4 1 4 2

Within cluster sum of squares by cluster:
[1] 13447.030  4673.760  7560.377 11038.722
 (between_SS / total_SS =  92.1 %)
```

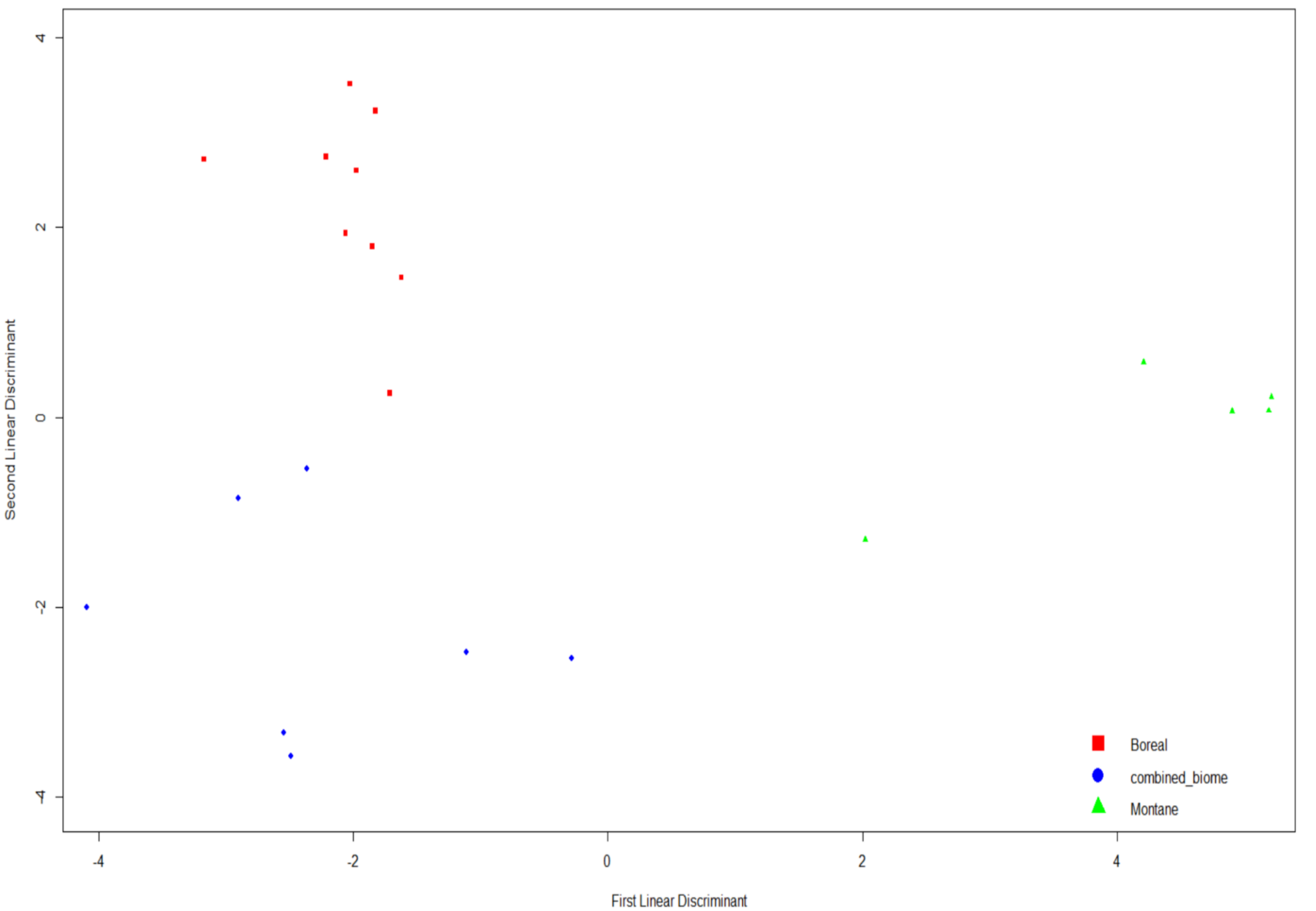


We can see that 4 is optimum number of groups from this WSS plot.

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

3D)

LINEAR DISCRIMINANT SCATTERPLOT:-




```
> m.decision
          [,1]
[1,] -5566.198
```

```
> #APER#
> table(df$BIOME,pred.group)
                pred.group
                 Boreal combined_biome Montane
  Boreal              9              0       0
  combined_biome      0              7       0
  Montane             0              0       5
>
> #AER#
> table(df$BIOME, disc.crossvalidation$class)

                 Boreal combined_biome Montane
  Boreal              8              1       0
  combined_biome      1              6       0
  Montane             1              0       4
```

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

R CODES:-

```
############################################## Q 1 ###############################################################


stocks <- read.csv("C:/Users/alexk/OneDrive/Desktop/studies/6348 2022/6348 project 3/stocks.csv",

            header = TRUE)

my_cols <- c("red")

pairs(stocks[,-1], col = my_cols,pch = 21,  cex = 0.5)


############################################## Q 1  A ###############################################################

# PC ANALYSIS #

stocks_cor <- cor(stocks[,-1])


stocks_eigen <- eigen(stocks_cor)

stocks_eigen

# factoer = 3 #

lambda <- as.matrix(stocks_eigen$vectors[,1:3]) %*% diag(sqrt(stocks_eigen$values[1:3]))

lambda

# finding communalities #

h2 <- rowSums(lambda^2)

u2 <- 1 - h2

# We collect the results into a data frame.

correlation.pa <- data.frame(cbind(round(lambda, 2), round(h2, 2), round(u2, 4)))

colnames(correlation.pa) <- c('PA1', 'PA2', 'PA3', 'h2', 'u2')

correlation.pa

# PC RESIDUALs #

R <- data.matrix(stocks_cor, rownames.force = NA)

L_data <- correlation.pa[,-(3:5)]

L <- data.matrix(L_data, rownames.force = NA)

Psi <- diag(correlation.pa$u2)

residual.PC <- R - (L %*% t(L)) - Psi

residual.PC

# ML method #

market.factor.analysis<- lapply(1:8,function(nf) factanal(stocks[,-1],factors=nf,rotation="none"))

market.factor.analysis
```

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

```
market.factor.analysis.rotation <- lapply(1:8,function(nf) factanal(stocks[,-1],factors=nf,rotation="varimax"))

market.factor.analysis.rotation

stocks.cor <- cor(stocks[,-1])

stocks.cor

#number of factors=3 #

f=3

# Residual matrix for f factor model#

pred <- market.factor.analysis[[f]]$loadings%*%t(market.factor.analysis[[f]]$loadings) + diag(market.factor.analysis[[f]]$uniquenesses)

pred


#number of factors=2 #

f2=2

# Residual matrix for f factor model#

pred1 <- market.factor.analysis[[f]]$loadings%*%t(market.factor.analysis[[f]]$loadings) + diag(market.factor.analysis[[f2]]$uniquenesses)

pred1


round(stocks.cor-pred, digits=3)

market.factor.analysis[[3]]

market.factor.analysis.rotation[[3]]

names(market.factor.analysis.rotation[[3]])

pred <- market.factor.analysis.rotation[[3]]$loadings%*%t(market.factor.analysis.rotation[[3]]$loadings) +
diag(market.factor.analysis.rotation[[3]]$uniquenesses)

print(pred)



# 13*12/2-13=65 pairs of corr data, hence not much inference. #

cor(stocks[,-1])

# checking  the range #

summary(stocks[,-1])


# checking the variance#

var(stocks[,-1])


############################################## Q 1 B ##############################################################
```

ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

```
vec <- eigen(cor(stocks[,-1])); vec$values


# checks if any eigen values are too close to zero, IT hints at linear dependence#

# using CORRELATION matrix#

stocks_pc = princomp(stocks[,-1], cor=TRUE)


summary(stocks_pc,loadings=TRUE)


screeplot(stocks_pc, type="lines", main = "Scree Plot")


##################################################Q2###########################################################
#


diabetes <- read.csv("C:/Users/alexk/OneDrive/Desktop/New folder/diabetes.csv")


View(diabetes)

attach(diabetes)


##################################################Q2 A #########################################


#Finding the discriminant function coefficient a#


m0 = apply(diabetes[Outcome == 0, -9], 2, mean)

m0

m1 = apply(diabetes[Outcome == 1, -9], 2, mean)

l0 = length(Outcome[Outcome == 0])

l1 = length(Outcome[Outcome == 1])

no.diabetes = diabetes[Outcome == 0, -9]

presence.of.diabetes = diabetes[Outcome == 0, -9]


S.pooled = ((l0 - 1)*var(no.diabetes)+(l1 - 1)*var(presence.of.diabetes))/(l0 + l1 - 2)

a = solve(S.pooled) %*% (m0 - m1)


#Finding the threshold for classification #

m.decision = t(a) %*% (m0 + m1)/2
```

ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

```
m.decision

# using MASS::lda function#
library(MASS)
disc = lda(Outcome ~
     Pregnancies+Glucose+BloodPressure+SkinThickness+Insulin+BMI+DiabetesPedigreeFunction+Age, data =
     diabetes, prior=c(l0/(l0+l1), l1/(l0+l1)))

disc # disc$scaling #coefficients are saved here #
# new data put as per the condition given to us by the question itself#
newdata = matrix(c(5,150,90,20,100,35,0.5,35),nrow = 1)
newdata

# creating new data frame with outcome removed #
colnames(newdata) = colnames(diabetes[-9])
newdata = data.frame(newdata)

# doing manual calculation #
t(a) %*%t (newdata)

####################################################Q2 B ##############################################

# using MASS:lda decision #
predict(disc, newdata = newdata)$class

# Confusion Matrix - to get "plug-in" estimate of misclassification rate (APER) #
pred.group = predict(disc, method = "plug-in")$class
cbind(Outcome, pred.group)
table(Outcome, pred.group)

# Leave-one-out estimate of misclassification rate: use CV = TRUE option #
```

```
disc.crossvalidation = lda(Outcome ~
        Pregnancies+Glucose+BloodPressure+SkinThickness+Insulin+BMI+DiabetesPedigreeFunction+Age, data =
        diabetes, prior=c(l0/(l0+l1), l1/(l0+l1)), CV=TRUE)
names(disc.crossvalidation)
disc.crossvalidation$class
cbind(Outcome, disc.crossvalidation$class)
table(Outcome, disc.crossvalidation$class)

#######################################################Q2C#######################################################

# finding Logistic Regression model for diabetes data set #
fit.diabetes = glm(Outcome ~
          Pregnancies+Glucose+BloodPressure+SkinThickness+Insulin+BMI+DiabetesPedigreeFunction+Age,
         family=binomial, data = diabetes)
summary(fit.diabetes)

#Predicting for a new data #
predict(fit.diabetes, newdata = newdata, type = "response")

l0/(l0+l1)

# finding Plug-in estimate #
table(Outcome, (predict(fit.diabetes, type = "response") > l0/(l0+l1)))



# doing Cross-Validation with Leave-one-out method #
newpred = numeric(length(Outcome)) # array of size(#obs)
for (i in 1:length(Outcome)) {
 newdat = diabetes[-i,]
 newfit = glm(Outcome ~
        Pregnancies+Glucose+BloodPressure+SkinThickness+Insulin+BMI+DiabetesPedigreeFunction+Age,
       family=binomial, data = newdat)
 newpred[i] = predict(newfit, newdat = data.frame(diabetes[i,-9]), type="response")
}
```

```
table(Outcome, (newpred > l0/(l0+l1)))

# ABOVE ---> Prior probability same as prob in existing data.#

# Below ---> Prior probability = 0.5 for each case#

# Plug-in estimate #

table(Outcome, (predict(fit.diabetes, type = "response") > 0.5))


# Cross-Validation with Leave-one-out method #

newpred = numeric(length(Outcome)) # array of size(#obs)

for (i in 1:length(Outcome)) {

  newdat = diabetes[-i,]

  newfit = glm(Outcome ~

        Pregnancies+Glucose+BloodPressure+SkinThickness+Insulin+BMI+DiabetesPedigreeFunction+Age,

       family=binomial, data = newdat)

  newpred[i] = predict(newfit, newdat = data.frame(diabetes[i,-9]), type="response")

}

table(Outcome, (newpred > 0.5))






###################Q 3 ###############################################


library(plotly)

climate <- read.csv("C:/Users/alexk/OneDrive/Desktop/New folder/AB_Climate_Means.csv", header = TRUE) # RStudioCloud environment

View(climate)


#################Q 3 A ##############################

###########BIOME as category ###################

dcol1<-factor(climate$BIOME)

dcol1

mycols1<-c("red","blue","green","black")

pairs(climate[,3:8], col = mycols1[as.numeric(dcol1)],pch = c(15:18)[as.numeric(dcol1)])
```

ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

```
legend("bottom", col = mycols1, legend = levels(dcol1), pch = c(15:18), xpd = NA,ncol = 4,  bty = "n", inset = c(-.5,-.1), pt.cex = 3)

climate2 <- climate[,3:8]

####################Q 3 B ##############################

#finding the number of cluster #

cluster.single <- cutree(hclust(dist(climate2), method="single"), h=48)

max(cluster.single)

cluster.complete <- cutree(hclust(dist(climate2), method="complete"), h=90)

max(cluster.complete)

#dendograms from hierarchical clustering#

#dendograms from single clustering#

plot(hclust(dist(climate2),method="single"),labels=row.names(climate2), ylab="Distance", main="(3A) Single Linkage"); abline(h=48)

#dendograms from complete clustering#

plot(hclust(dist(climate2),method="complete"),labels=row.names(climate2), ylab="Distance", main="(3B) Complete Linkage"); abline(h=90)

# initializing Single Linkage method #

nc = 1

row.names(climate2)[cluster.single == nc]

climate.single.cluster <- lapply(1:6, function(nc) {row.names(climate2)[cluster.single == nc]})

climate.single.cluster

# initializing Complete Linkage method #

nc = 1

row.names(climate2)[cluster.c == nc]

climate.complete.cluster <- lapply(1:6, function(nc) {row.names(climate2)[cluster.c == nc]})

climate.complete.cluster
```

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

```
# Mean vectors for each cluster#


# Mean vectors for Single Linkage cluster#
colMeans(climate2[cluster.single == 1,])
colMeans(climate2[cluster.single == 2,])
colMeans(climate2[cluster.single == 3,])
colMeans(climate2[cluster.single == 4,])
colMeans(climate2[cluster.single == 5,])
colMeans(climate2[cluster.single == 6,])


# Mean vectors for Complete Linkage cluster#
colMeans(climate2[cluster.complete == 1,])
colMeans(climate2[cluster.complete == 2,])
colMeans(climate2[cluster.complete == 3,])
colMeans(climate2[cluster.complete == 4,])
colMeans(climate2[cluster.complete == 5,])
colMeans(climate2[cluster.complete == 6,])


#####################Q 3 C #########################
# perfroming k-means clustering#
set.seed(6354)
wss <- numeric(6)
for(i in 1:6) {
 W <- sum(kmeans(climate2, i)$withinss)
 wss[i] <- W
}
#Plotting the wss vs number of clusters
plot(1:6, wss, type="l", xlab="Number of groups", ylab="Within groups sum of squares", lwd=2)
climate2.kmean <- kmeans(climate2, 4)
climate2.kmean


###################################Q 3 D#########################################
cl <- read.csv("C:/Users/alexk/OneDrive/Desktop/New folder/AB_Climate_Means.csv", header = TRUE) # RStudioCloud environment
```

ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

```
df<-data.frame(cl)

df

df$BIOME[df$BIOME=="Grassland"]<-"combined_biome"

df


df1<-data.frame(df)

df1

df1$BIOME[df1$BIOME=="Parkland"]<-"combined_biome"

df1


df<-df1[2:10]


m0 = apply(df[df$BIOME == "combined_biome", -1],2,mean)

m1 = apply(df[df$BIOME == "Boreal", -1],2,mean)

m2 = apply(df[df$BIOME == "Montane", -1],2,mean)


l0 = length(df[df$BIOME == "combined_biome",-1])

l1 = length(df[df$BIOME == "Boreal",-1])

l2 = length(df[df$BIOME == "Montane",-1])


combined_biome = df[df$BIOME == "combined_biome", -1]

boreal = df[df$BIOME == "Boreal", -1]

montane = df[df$BIOME == "Montane", -1]



S.pooled = ((l0 - 1)*var(combined_biome)+(l1 - 1)*var(boreal)+(l2 - 1)*var(montane))/(l0 + l1+l2 - 3)

a = solve(S.pooled) %*% (m0 - m1-m2)

#Finding the threshold for classification #

m.decision = t(a) %*% (m0 + m1+m2)/3

m.decision


library(MASS)
```

ASHISH MANI ACHARYA   AXA190076   STAT 6348   PROJECT 3   11/30/2022

```
disc = lda(BIOME ~
      MAT+MWMT+MCMT+TD+MAP+MSP+AHM+SHM, data = df1, prior=c(l0/(l0+l1+l2), l1/(l0+l1+l2),l2/(l0+l1+l2)))

disc
table(df$BIOME, pred.group)
table(df$BIOME,pred.group)

# Leave-one-out estimate of misclassification rate: use CV = TRUE option #
disc.crossvalidation =lda(BIOME ~
            MAT+MWMT+MCMT+TD+MAP+MSP+AHM+SHM, data = df1, prior=c(l0/(l0+l1+l2), l1/(l0+l1+l2),l2/(l0+l1+l2)),CV=TRUE)
names(disc.crossvalidation)
disc.crossvalidation$class
cbind(df$BIOME, disc.crossvalidation$class)
table(df$BIOME, disc.crossvalidation$class)

#APER#
table(df$BIOME,pred.group)

#AER#
table(df$BIOME, disc.crossvalidation$class)

#################for linear discriminant plot ############
df<-data.frame(climate)
df
df$BIOME[df$BIOME=="Grassland"]<-"combined_biome"
df

df1<-data.frame(df)
df1
df1$BIOME[df1$BIOME=="Parkland"]<-"combined_biome"
df1
```

ASHISH MANI ACHARYA AXA190076 STAT 6348 PROJECT 3 11/30/2022

```
df1[,2]

climate_df<-df1[,3:10]

climate_df


climate.lda<-lda(climate_df,df1[,2],prior=c(1/3,1/3,1/3))

climate.lda


climate.ld<-predict(climate.lda)$x

climate.ld


climate.cv<-lda(climate_df,df1[,2],prior=c(1/3,1/3,1/3),CV=TRUE)

library(MASS)

##Alternatively use predict function

dcol1<-factor(df1$BIOME)

dcol1

mycols1<-c("red","blue","green")

eqscplot(climate.ld[1:21,1],climate.ld[1:21,2],xlab="First Linear Discriminant",ylab="Second Linear
Discriminant",xlim=range(climate.ld[,1]),ylim=range(climate.ld[,2]),cex=0.8,col = mycols1[as.numeric(dcol1)],pch =
c(15:18)[as.numeric(dcol1)])

legend("bottomright", col = mycols1, legend = levels(dcol1), pch = c(15:18), xpd = NA,  bty = "n", inset = c(-0.1,-.01), pt.cex = 2)
```