Case study

a. Clustering based on brand loyalty
Variables considered:
[# brands, brand runs, total volume, # transactions, value, Avg. price, share to other brands, max to one brand]

Max to one brand – Calculated highest volume purchased for all the brand categories

As the data consists of percentage of total volume purchased by the households, taking percentage and normalising them will not give actual loyalty of the household to one brand. For example, if we consider 20% of 1000 volume purchased and 20% of 500 volume purchased the scale will differ for 'max to one brand' variable. That's why initially all the percentages are converted into actual volumes before normalising the data.

Step 1:
Finding out the total volumes for each brand category

```
volume <- function(x){
  return(x*brand_loyalty$`Total Volume`)
}
vol<-as.data.frame(lapply(brand_loyalty[9:20],volume))
```

	Pur.Vol.No.Promo....	Pur.Vol.Promo.6..	Pur.Vol.Other.Promo..	Br..Cd..57..144	Br..Cd..55	Br..Cd..272	Br..Cd..286	Br..Cd..24	Br..Cd..481	Br..Cd..352	Br..Cd..5	Others.999
1	8025	0	0	3025	1050	0	0	0	0	0	0	3950
2	12400	1350	225	300	1050	0	0	0	825	0	2025	9775
3	21750	450	900	600	12600	0	700	0	0	0	450	8750
4	1500	0	0	600	900	0	0	0	0	0	0	0
5	5100	1200	2000	400	1200	0	0	0	0	0	0	6700

Step 2:
Finding max to one brand category other than 'others 999'

```
brand_final<-brand_loyalty[,1:8]
brand_final<-cbind(brand_final,vol)
brand_final$max <- apply(brand_final[,12:19], 1, max)
```

Br..Cd..57..144	Br..Cd..55	Br..Cd..272	Br..Cd..286	Br..Cd..24	Br..Cd..481	Br..Cd..352	Br..Cd..5	Others.999	max
3025	1050	0	0	0	0	0	0	3950	3025
300	1050	0	0	0	825	0	2025	9775	2025
600	12600	0	700	0	0	0	450	8750	12600
600	900	0	0	0	0	0	0	0	900

Step 3:
Normalise all the variables considered to get on one scale

```
normalize <- function(x) {
  num <- x - min(x)
  denom <- max(x) - min(x)
```

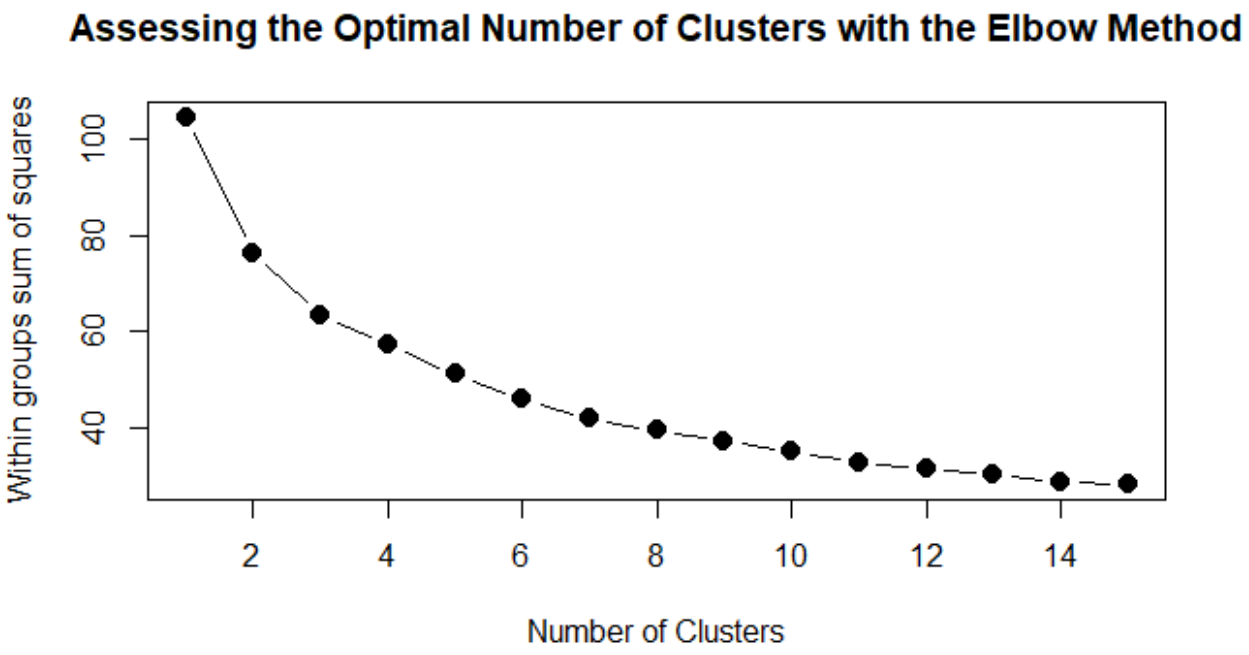
```
 return (num/denom)
}
brand_norm <- as.data.frame(lapply(brand_final[c(1:5,8,20,21)], normalize))
```

	No..of.Brands	Brand.Runs	Total.Volume	No..of..Trans	Value	Avg..Price	Others.999	max
1	0.250	0.21917808	0.155187703	0.16788321	0.125632690	0.1649537584	0.097772277	0.078981723
2	0.500	0.32876712	0.272440635	0.28467153	0.261577336	0.2313416541	0.241955446	0.052872063
3	0.500	0.49315068	0.452261307	0.45255474	0.303848485	0.1017232911	0.216584158	0.328981723
4	0.125	0.04109589	0.026603606	0.02189781	0.014798838	0.0713439244	0.000000000	0.023498695
5	0.250	0.06849315	0.160606956	0.08759124	0.089895070	0.0540338308	0.165841584	0.031331593
6	0.250	0.34246575	0.355207410	0.29197080	0.265355762	0.1357360893	0.385519802	0.036553525

Step 4:

Check for the optimal number of 'K' by plotting Scree plot

```
wss <- (nrow(brand_norm)-1)*sum(apply(brand_norm,2,var))
for (i in 2:15){
set.seed(7)
 wss[i] <- sum(kmeans(brand_norm, centers=i)$withinss)
}
plot(1:15, wss, type="b", xlab="Number of Clusters", ylab="Within groups sum of squares",
   main="Assessing the Optimal Number of Clusters with the Elbow Method", pch=20, cex=2)
```



Analysis:

From screen plot above, we can see within group sum of squares decreasing significantly from k=6 to k=7, whereas for k>7 within group sum of squares does not show significant drop. That's why by Elbow Method K=6 is considered to be optimal value for this model.

Optimal value of K = 6

Step 5:

Clustering the data with optimal K value

set.seed(7)

km_brand = kmeans(brand_norm, 6, nstart=100)

 # Examine the result of the clustering algorithm

km_brand

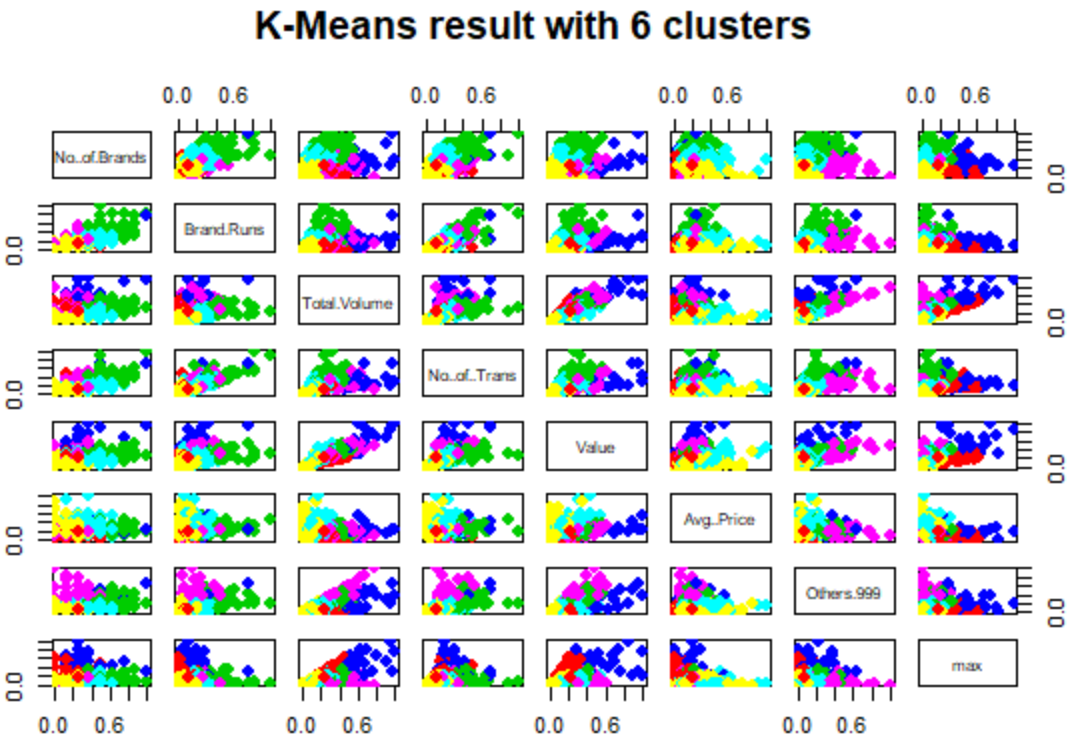
Results:

```
> km_brand$centers
  No..of.Brands Brand.Runs Total.Volume No..of..Trans      Value Avg..Price Others.999        max
1     0.2021605 0.09318451   0.28762427    0.19383617 0.18749918 0.09882029 0.06966905 0.28947072
2     0.6194444 0.40928463   0.30025071    0.39148418 0.28807312 0.24186379 0.21566557 0.09831738
3     0.3920455 0.20672478   0.66283288    0.28832117 0.60427528 0.20658090 0.24108911 0.52584263
4     0.4006410 0.23400070   0.18459032    0.22844844 0.19005294 0.27516033 0.12718926 0.07815157
5     0.2351695 0.21917808   0.39929625    0.25955710 0.32522488 0.17218159 0.41966983 0.06805107
6     0.1633987 0.08989166   0.09574207    0.09751443 0.09009505 0.23785971 0.07046205 0.04992833
```

```
> km_brand$size
[1]  81  90  22 195  59 153
> km_brand$withinss
[1]  4.516747  9.181149  5.384680 11.638164  5.649940  9.032387
```

col =(km_brand$cluster +1)

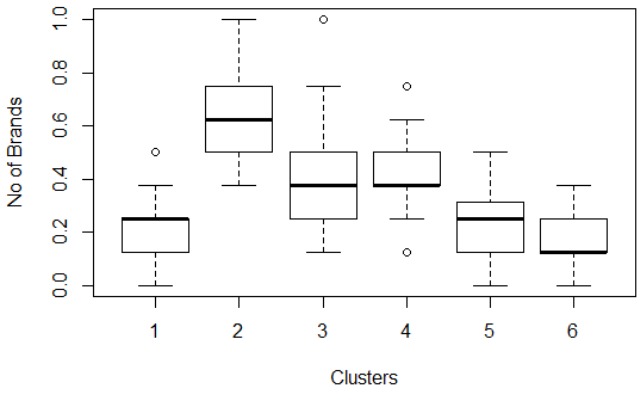
plot(brand_norm, col = col , main="K-Means result with 6 clusters", pch=20, cex=2)



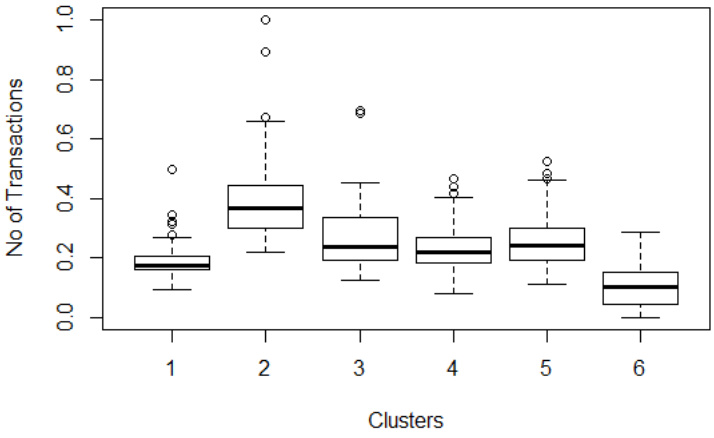
Step 6:

To analyse the distribution and importance of variables considered for brand loyalty boxplots are plotted for each variable across the clusters

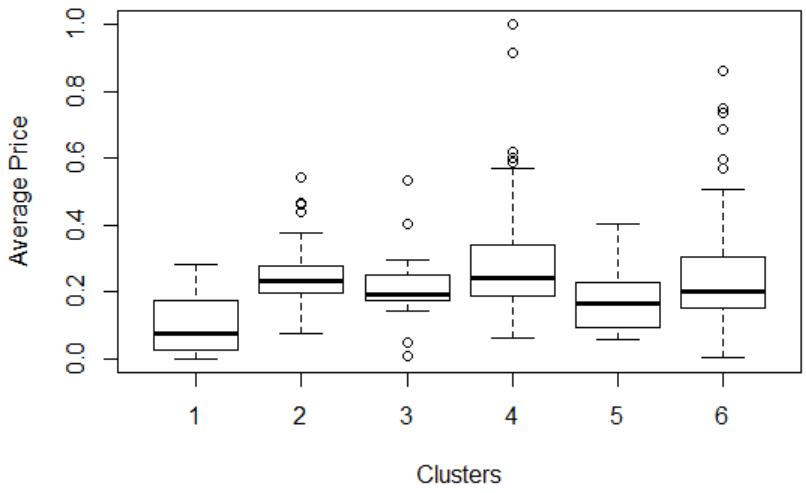
boxplot(brand_norm$No..of.Brands~km_brand$cluster, ylab="No of Brands", xlab="Clusters")
boxplot(brand_norm$Brand.Runs~km_brand$cluster, ylab="Brand Runs", xlab="Clusters")
boxplot(brand_norm$Total.Volume~km_brand$cluster, ylab="Total Volume", xlab="Clusters")
boxplot(brand_norm$No..of..Trans~km_brand$cluster, ylab="No of Transactions", xlab="Clusters")
boxplot(brand_norm$Value~km_brand$cluster, ylab="Value", xlab="Clusters")
boxplot(brand_norm$Avg..Price~km_brand$cluster, ylab="Average Price", xlab="Clusters")
boxplot(brand_norm$Others.999~km_brand$cluster, ylab="Others99", xlab="Clusters")
boxplot(brand_norm$max~km_brand$cluster, ylab="Max to one brand", xlab="Clusters")


No of Brands
0.0
0.2
0.4
0.6
0.8
1.0
1
2
3
4
5
6
Clusters

img (i)


No of Transactions
0.0
0.2
0.4
0.6
0.8
1.0
1
2
3
4
5
6
Clusters

img (ii)



img (iii)

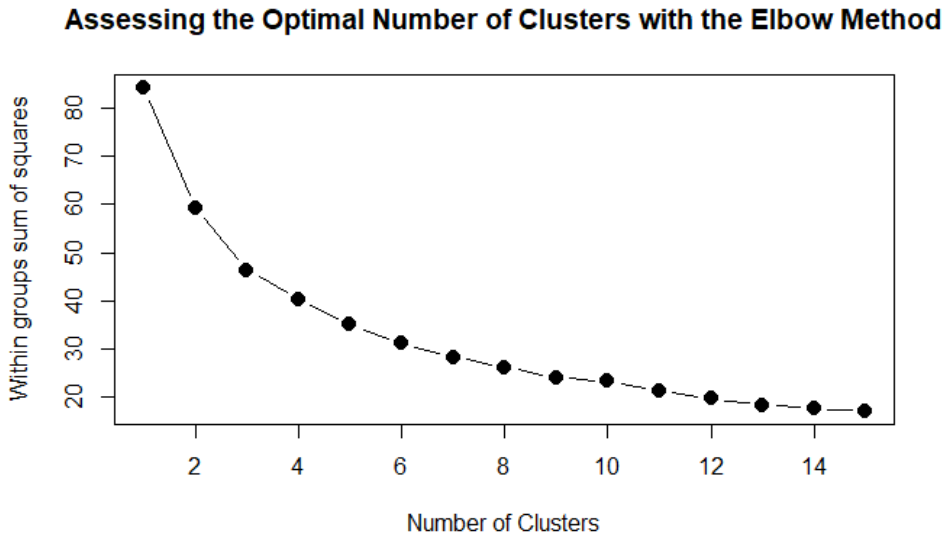
Analysis:

From img (i) we can see, No. of brands across six clusters are distributed varyingly which adds higher information for cluster segmentation. Whereas, from img(ii) and img(iii) we can say 'No of transactions' and 'Average price value' have similar distribution across six clusters. Therefore, these two variables will not add additive information for segment clustering. As per this analysis, we will drop 'No of transactions' and 'Average price value' from the brand loyalty dataset and build the K-means clustering model again.

Step 7:

Results after dropping 'not so important' variables

```
brand_norm_imp<-brand_norm[,c(1,2,3,5,7,8)]

wss_1 <- (nrow(brand_norm_imp)-1)*sum(apply(brand_norm_imp,2,var))

for (i in 2:15){

  set.seed(7)

  wss_1[i] <- sum(kmeans(brand_norm_imp, centers=i)$withinss)

}

plot(1:15, wss_1, type="b", xlab="Number of Clusters", ylab="Within groups sum of squares",

    main="Assessing the Optimal Number of Clusters with the Elbow Method", pch=20, cex=2)
```



By Elbow method, optimum **K = 6**

set.seed(7)

km_brand_imp = kmeans(brand_norm_imp, 6, nstart =100)

```
> km_brand_imp
K-means clustering with 6 clusters of sizes 77, 22, 85, 77, 187, 152

Cluster means:
  No..of.Brands Brand.Runs Total.Volume      Value Others.999        max
1     0.1883117 0.08610568    0.2909083 0.21293377 0.06600714 0.30026957
2     0.4090909 0.22166874    0.6649603 0.60387274 0.25059631 0.50792191
3     0.6455882 0.40773570    0.2675894 0.26470056 0.18905067 0.08647673
4     0.2743506 0.23750222    0.3877575 0.33324528 0.39762601 0.07091146
5     0.3990642 0.23104534    0.1854952 0.17747965 0.11822245 0.08681811
6     0.1554276 0.08940159    0.0972730 0.08823667 0.07441050 0.04879672

Within cluster sum of squares by cluster:
[1] 3.551351 4.533183 5.745186 6.418782 5.850307 4.704661
 (between_SS / total_SS =  63.4 %)
```

Results:

	k	betweens	totss	tot.withinss	within/total	within/between	between/total
with all var	6	59.3185	104.7216	45.40307	0.433559743	0.765411634	0.56643997
imp var	6	53.29082	84.09429	30.80347	0.366296808	0.578025821	0.633703192

From the above table, it is analysed that when variables contributing less to cluster segmentation are removed from the data, within group squared distance/total squared distance value in decreased and between group squared distance/total squared distance value is increased.

b. Clustering based on basis-of-purchase

Variables considered: [Pur-vol-no-promo, Pur-vol-promo-6, Pur-vol-other, all price categories, selling propositions]

To decide upon the important variables, we calculated how many percentage of households invested in **PropCat 5 – PropCat15 categories less than 10%**. And got the below results, which shows only ProCat5 category has highest percentage of households investing more than 10% of total purchase volume. That is why, we dropped all the other selling propositions except PropCat5 and performed the K-means clustering on remaining variables.

	PropCat 5	PropCat 6	PropCat 7	PropCat 8	PropCat 9	PropCat 10	PropCat 11	PropCat 12	PropCat 13	PropCat 14	PropCat 15
Counts < 10%	122	448	464	464	534	568	553	593	566	442	562
%instances < 10%	20.33333	74.66667	77.33333	77.33333	89	94.66667	92.16667	98.83333	94.33333	73.66667	93.66667

#Subsetting basis of purchase varaibles

sp_data<-BathSoap_Data[,c(20,21,22,32:46)]

#Calculating actual volume for percenatges

vol_sp<-as.data.frame(lapply(sp_data[1:18],volume))

#Normalising the data

sp_norm <- as.data.frame(lapply(vol_sp[c(1:18)], normalize))

#Taking important variables

sp_norm_final<- sp_norm[,c(1:8)]

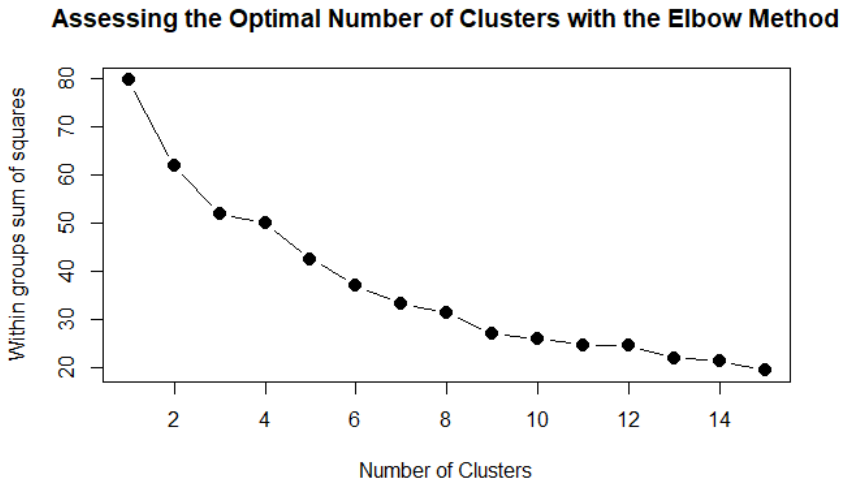
	Pur.Vol.No.Promo....	Pur.Vol.Promo.6..	Pur.Vol.Other.Promo..	Pr.Cat.1	Pr.Cat.2	Pr.Cat.3	Pr.Cat.4	PropCat.5
1	0.169393140	0.000000000	0.000000	0.068119891	0.108173077	0.033333333	0.017857143	0.087547580
2	0.261741425	0.166666667	0.028125	0.148955495	0.183894231	0.042063492	0.026785714	0.138662316
3	0.459102902	0.055555556	0.112500	0.100817439	0.176682692	0.411904762	0.000000000	0.122892877
4	0.031662269	0.000000000	0.000000	0.000000000	0.014423077	0.028571429	0.000000000	0.013050571

#Calculating optimum K value with Scree plot

*wss_2 <- (nrow(sp_norm_final)-1)*sum(apply(sp_norm_final,2,var))*

for (i in 2:15){

```
  set.seed(7)
  wss_2[i] <- sum(kmeans(sp_norm_final, centers=i)$withinss)
}
plot(1:15, wss_2, type="b", xlab="Number of Clusters", ylab="Within groups sum of squares",
     main="Assessing the Optimal Number of Clusters with the Elbow Method", pch=20, cex=2)
```



Optimum K value :

k	betweens	totss	tot.withinss	within/total	within/between	between/total
5	41.13351	79.57699	38.44348	0.483097941	0.934602469	0.516902059
6	45.53346	79.57699	34.04353	0.427806204	0.747659633	0.572193796
7	49.09445	79.57699	30.48254	0.383057213	0.620895845	0.616942787

As we see from above results, within/total ratio is decreasing significantly from k =5 to k= 6 than k>6. So optimal value of **K = 6** is considered.

#Clustering K = 6

```
set.seed(7)
km_sp_2 = kmeans(sp_norm_final, 6, nstart =100)
```

#Output:

```
> km_sp_1
K-means clustering with 5 clusters of sizes 46, 39, 21, 320, 174

Cluster means:
  Pur.Vol.No.Promo.... Pur.Vol.Promo.6.. Pur.Vol.Other.Promo..   Pr.Cat.1
1            0.2324974        0.41861245            0.06005435 0.14270821
2            0.3804208        0.01250396            0.10416667 0.02978178
3            0.6929966        0.08406820            0.13303571 0.14757147
4            0.1309334        0.04027392            0.03291992 0.08838613
5            0.3265529        0.05156450            0.04349856 0.12267427
   Pr.Cat.2   Pr.Cat.3    Pr.Cat.4  PropCat.5
1 0.1177258 0.04987923 0.133605072 0.16696220
2 0.0537198 0.49540090 0.008852259 0.03077202
3 0.6161802 0.03885110 0.108843537 0.60190580
4 0.0737162 0.02288690 0.017029390 0.06193617
5 0.2489572 0.03804050 0.038827313 0.17832718
```

```
Within cluster sum of squares by cluster:
[1]  6.435506  2.784375  6.113112 10.193839 12.916650
 (between_SS / total_SS =  51.7 %)
```

c. Clustering analysis on both brand loyalty and basis of purchase

Step 1:

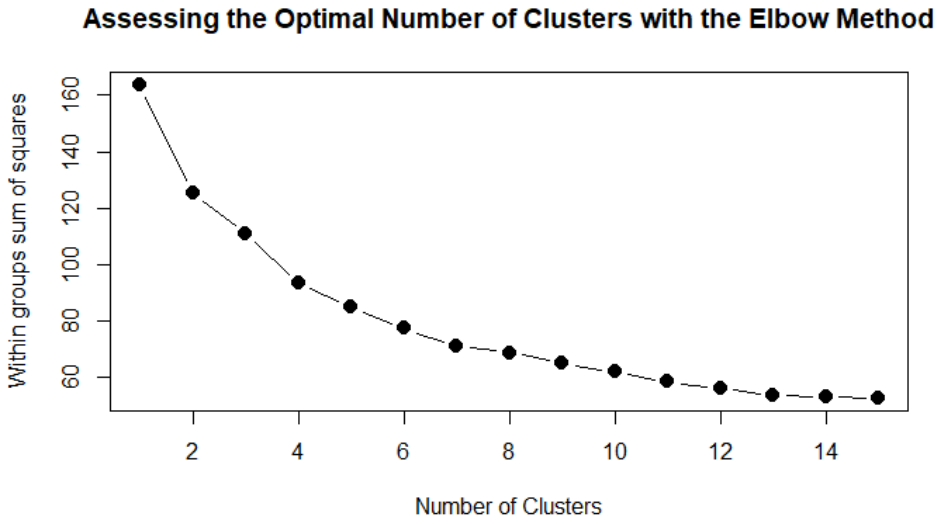
Combine brand loyalty and basis of purchase variables

brsp_data<-cbind(brand_norm_imp,sp_norm_final)

Step 2:

Optimum value of K by scree plot

```
wss_3 <- (nrow(brsp_data)-1)*sum(apply(brsp_data,2,var))

for (i in 2:15){

 set.seed(7)

  wss_3[i] <- sum(kmeans(brsp_data, centers=i)$withinss)

}

plot(1:15, wss_3, type="b", xlab="Number of Clusters", ylab="Within groups sum of squares",

    main="Assessing the Optimal Number of Clusters with the Elbow Method", pch=20, cex=2)
```



k	betweens	totss	tot.withinss	within/total	within/between	between/total
5	79.11092	163.6713	84.56036	0.516647451	1.068883537	0.483352426
6	86.53491	163.6713	77.13638	0.471288369	0.891390307	0.52871157
7	92.60364	163.6713	71.06764	0.434209541	0.767438947	0.565790337

Analysis:

From the above table, we can see there is a drop in within/total ratio value from K = 5 to K=6 and its not showing significant decrease for K>6 so optimum K value considered as K = 5.

#Clustering K = 5

set.seed(7)

km_brsp=kmeans(brsp_data, 5, nstart =100)

```
> km_brsp
K-means clustering with 5 clusters of sizes 35, 15, 254, 198, 98

Cluster means:
  No..of.Brands Brand.Runs Total.Volume     Value Others.999        max Pur.Vol.No.Promo....
1     0.2178571 0.08688845    0.3673662 0.1928842 0.05832390 0.41559120            0.3723603
2     0.3750000 0.20365297    0.7038198 0.6411974 0.24344884 0.57380331            0.7148707
3     0.1894685 0.11325639    0.1369294 0.1247650 0.08856709 0.08091168            0.1385341
4     0.4936869 0.29922513    0.2007818 0.2046352 0.13781466 0.07611691            0.1979843
5     0.3941327 0.27690802    0.4199449 0.3659614 0.35942615 0.12226914            0.4148867
  Pur.Vol.Promo.6.. Pur.Vol.Other.Promo..   Pr.Cat.1   Pr.Cat.2   Pr.Cat.3   Pr.Cat.4  PropCat.5
1        0.02768959            0.11589286 0.02695731 0.04565591 0.49784580 0.01394558 0.02245009
2        0.07633745            0.17250000 0.14050257 0.72118590 0.04275132 0.01934524 0.59762190
3        0.03794838            0.02850640 0.06241355 0.09115309 0.02570616 0.02317972 0.07821094
4        0.08315563            0.03570076 0.12010385 0.13672446 0.02436267 0.01719201 0.10206414
5        0.15460443            0.06906888 0.18131939 0.25222356 0.07096696 0.11134900 0.23004628
```

```
within cluster sum of squares by cluster:
[1]  5.075316  6.562796 22.413342 18.765495 31.743413
 (between_SS / total_SS =  48.3 %)
```

Q 2. a

While deciding on the better segmentation, within/total and between/total ratio is considered for all the three approaches above.

Variable considered	Within/total	Between/total
Brand loyalty	36.63%	63.67%
Selling Proposition	42.78%	57.22%
Brand loyalty and Selling proposition	51.54%	48.33%

Analysis:

Based on above table, we can see for the first clustering within/total ratio is lowest and between/total ratio is highest. This explains that within group variance is decreasing when the segmentation is done on brand loyalty criteria, giving more well-defined clusters. Therefore, segmentation one is best.

Q 2. b

#Getting all the variables for instances falling into different clusters

Cluster1Instances = BathSoap_Data[km_brand_imp$cluster == 1,]
Cluster2Instances = BathSoap_Data[km_brand_imp$cluster == 2,]
Cluster3Instances = BathSoap_Data[km_brand_imp$cluster == 3,]
Cluster4Instances = BathSoap_Data[km_brand_imp$cluster == 4,]
Cluster5Instances = BathSoap_Data[km_brand_imp$cluster == 5,]
Cluster6Instances = BathSoap_Data[km_brand_imp$cluster == 6,]

Comment on characteristics – Brand Loyalty

```
> km_brand_imp$centers
  No..of.Brands Brand.Runs Total.Volume      Value Others.999        max
1     0.1883117 0.08610568    0.2909083 0.21293377 0.06600714 0.30026957
2     0.4090909 0.22166874    0.6649603 0.60387274 0.25059631 0.50792191
3     0.6455882 0.40773570    0.2675894 0.26470056 0.18905067 0.08647673
4     0.2743506 0.23750222    0.3877575 0.33324528 0.39762601 0.07091146
5     0.3990642 0.23104534    0.1854952 0.17747965 0.11822245 0.08681811
6     0.1554276 0.08940159    0.0972730 0.08823667 0.07441050 0.04879672
```

Clusters	No of brands	Brand runs	Total Volume	Value	Others 999	Max to one brand
1	Below average	Below average	Average	Below Average	Below average	Above Average
2	Above Average	Average	Highest	Highest	Above Average	Above Average
3	Highest	Above average	Average	Average	Average	Below Average
4	Below average	Average	Average	Above Average	Above Average	Below Average

5	Above average	Average	Below average	Below average	Average	Below Average
6	Below average	Below average	Below average	Below average	Below Average	Below Average

Comment on characteristics – Demographics (For Cluster 1)

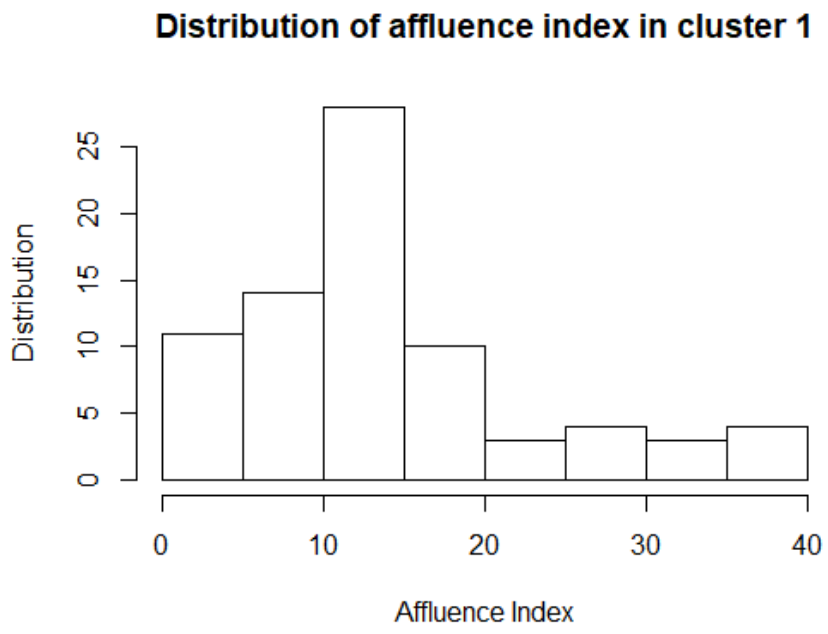
```
> table(Cluster1Instances$SEC)

 1  2  3  4
11 18 15 33
```

Similarly, it is calculated for all demographic variables.

From the below table we can see which level of categorical variables has highest frequency values in Cluster1.

	SEC	FEH	MT	SEX	AGE	EDU	HS	CHILD	CS
Mode category	Low socio economy class	Non-vegetarian	Marathi	Female	Homemaker age 45+	5-9 years of school	5	None	Cable or broadcast TV available

```
hist(Cluster1Instances$`Affluence Index`, main= "Distribution of affluence in
dex in cluster 1", xlab = 'Affluence Index', ylab='Distribution')
```



Affluence Index value is right skewed with median value of 13.

Comment on characteristics – Demographics all the clusters

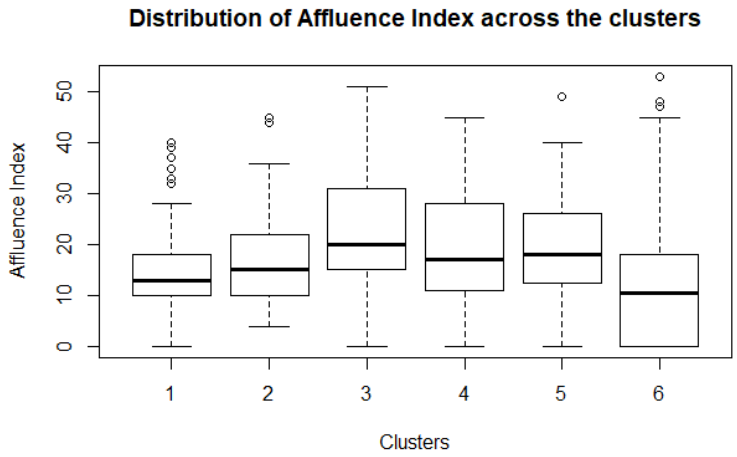
Combined clusters values and actual dataset

```
BathSoap_Data<-cbind(BathSoap_Data,km_brand_imp$cluster)
```

```
BathSoap_Data$`km_brand_imp$cluster`<-as.factor(BathSoap_Data$`km_brand_imp$cluster`)
```

```
table(BathSoap_Data$SEC,BathSoap_Data$`km_brand_imp$cluster`)
> table(BathSoap_Data$SEC,BathSoap_Data$`km_brand_imp$cluster`)

   1  2  3  4  5  6
1 11  4 23 14 50 48
2 18  4 30 15 53 30
3 15  7 18 25 48 37
4 33  7 14 23 36 37
> table(BathSoap_Data$FEH,BathSoap_Data$`km_brand_imp$cluster`)

   1   2   3   4   5   6
0  1   0   1   2  10  55
1 22   6  25  10  64  38
2  9   1   6   4  10   4
3 45  15  53  61 103  55
> table(BathSoap_Data$MT,BathSoap_Data$`km_brand_imp$cluster`)

    1   2   3   4   5   6
0   1   0   1   2  10  55
3   0   0   0   0   3   2
4  16   3  13   5  27  19
5   3   1   5   6   6   6
6   2   1   0   2   3   3
8   0   0   1   3   3   1
9   0   1   2   2   3   1
10 46  13  55  40 115  57
12  1   0   3   0   3   0
13  1   0   1   0   5   1
14  0   0   0   1   1   1
15  1   0   1   2   2   2
16  3   0   2   2   2   1
17  3   3   1  12   3   3
19  0   0   0   0   1   0
> table(BathSoap_Data$SEX,BathSoap_Data$`km_brand_imp$cluster`)

   1   2   3   4   5   6
0  1   0   1   2   9  55
1  3   0   4   2   9   3
2 73  22  80  73 169  94
> table(BathSoap_Data$AGE,BathSoap_Data$`km_brand_imp$cluster`)

   1  2  3  4  5  6
1  2  1  1  1  5  5
2 14  2 15  7 41 50
3 20  4 22 26 57 40
4 41 15 47 43 84 57
>
```

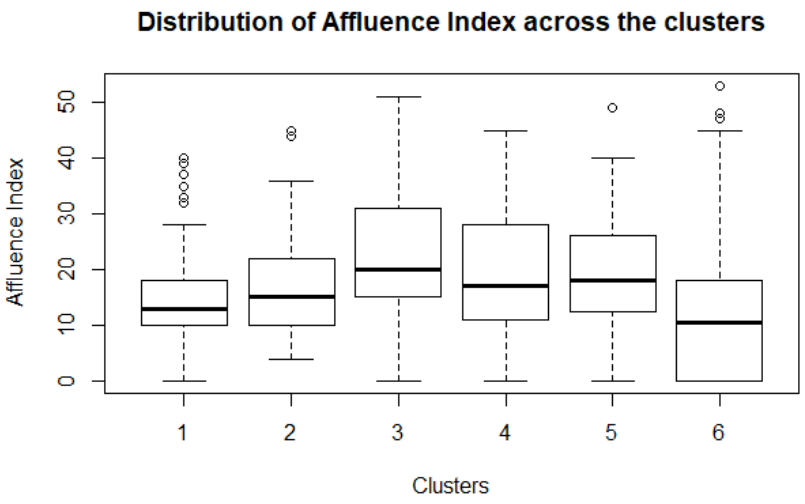
```
> table(BathSoap_Data$EDU,BathSoap_Data$`km_brand_imp$cluster`)

    1  2  3  4  5  6
0   3  1  1  3 10 55
1  15  2  7  3  9 13
2   1  0  1  3  3  1
3   9  3  2  6  9  4
4  22  9 15 30 37 23
5  16  5 36 22 74 36
6   2  0  5  3  9  4
7   9  1 15  7 29 12
8   0  0  3  0  6  4
9   0  1  0  0  1  0
'
> table(BathSoap_Data$HS,BathSoap_Data$`km_brand_imp$cluster`)

     1  2  3  4  5  6
0    1  0  1  2  9 55
1    0  0  0  0  0  2
2    5  0  4  0 16 16
3    8  1  7  6 33 18
4   13  3 25 13 63 30
5   25  7 26 23 42 19
6   10  3  9 17 19  7
7    6  2  6  4  3  1
8    3  2  2  5  2  4
9    4  1  2  6  0  0
10   2  0  2  0  0  0
12   0  1  1  0  0  0
15   0  2  0  1  0  0
'
> table(BathSoap_Data$CHILD,BathSoap_Data$`km_brand_imp$cluster`)

    1  2  3  4  5  6
1   3  3 12 11 20 10
2  18  7 24 20 47 29
3   9  0 12  9 18 13
4  46 12 36 35 93 45
5   1  0  1  2  9 55
'  -  -  -  -  -  --
> table(BathSoap_Data$CS,BathSoap_Data$`km_brand_imp$cluster`)

     1   2   3   4   5   6
0    7   1   5   6  16  64
1   57  19  76  63 152  76
2   13   2   4   8  19  12
'

boxplot(BathSoap_Data$`Affluence Index`~BathSoap_Data$`km_brand_imp$cluster`,
main="Distribution of Affluence Index across the clusters", xlab="Clusters",
ylab="Affluence Index")
```



This table gives the level of demographic variables, the instances in one cluster belongs to:

Clusters	SEC	FEH	MT	SEX	Age of homemaker	EDU	CHILD	CS
1	Low SEC class	Non-vegetarian	Marathi	Female	45+	Illiterate/5-9 years/10-12 years	None	Cable or Broadcast TV available
2	No specific	Non-vegetarian	Marathi	Female	45+	5-9 years of school	None	Cable or Broadcast TV available
3	Class B	Non-vegetarian & veg but serve egg	Marathi	Female	Above 24 even distribution	5-9 years/10-12 years/college graduate	Children age 7-14	Cable or Broadcast TV available
4	More in class 3 & 4	Non-vegetarian	Marathi	Female	45+	5-9 years/10-12 years	Children age 7-14	Cable or Broadcast TV available
5	Even distribution	Non-vegetarian	Marathi	Female	Above 24 even distribution	5-9 years/10-12 years/college graduate	Children age 7-14/ None	Cable or Broadcast TV available
6	Even distribution	Non-vegetarian & not specified	Marathi & not specified	Female	Above 24 even distribution	10-12 years/college graduate	Not specified	Cable or Broadcast TV available/Not specified

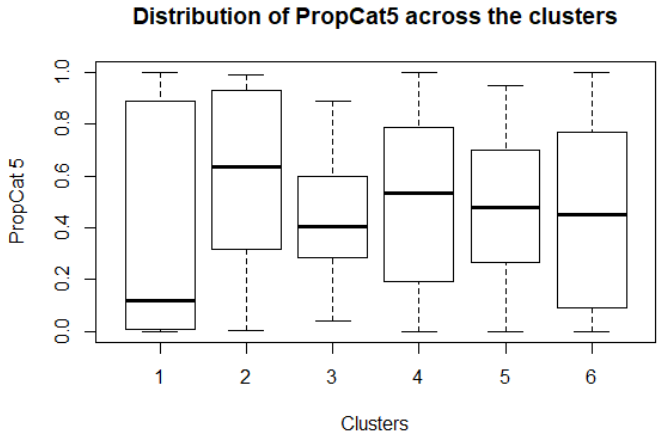
```
boxplot(BathSoap_Data$`Affluence Index`~BathSoap_Data$`km_brand_imp$cluster`,
main="Distribution of Affluence Index across the clusters", xlab="Clusters",
ylab="Affluence Index")
```



Affluence Index distribution is not even across the cluster. Therefore, it can be considered as important variable to decide upon development of advertising and promotional campaigns.

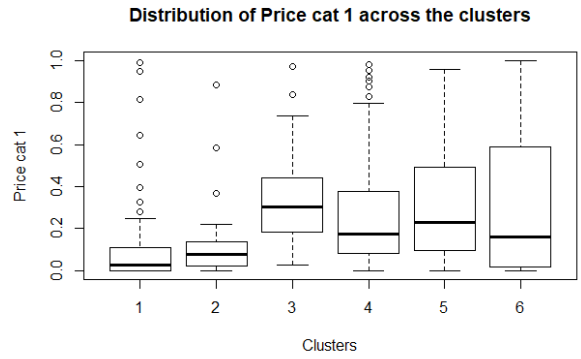
Comment on characteristics – Basis of purchase

```
boxplot(BathSoap_Data$`PropCat 5`~km_brand_imp$cluster, main = "Distribution
of PropCat5 across the clusters", xlab="Clusters", ylab="PropCat 5")
```

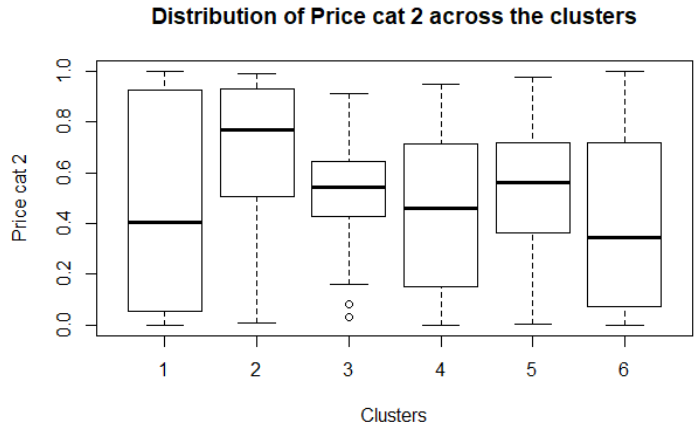


PropCat 5 distribution is not even across the cluster. Like for cluster1 is right skewed for Propcat5. For all the other clusters its normally distributed within cluster data.

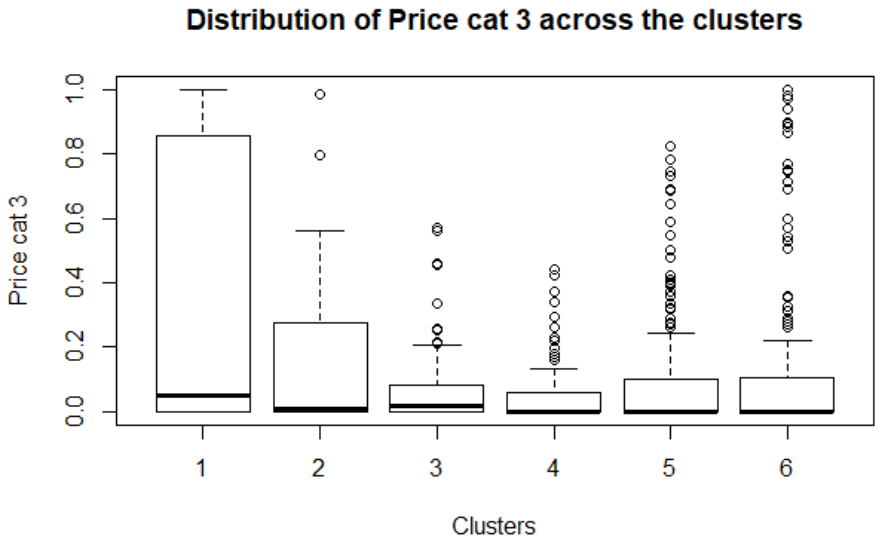
```
> boxplot(BathSoap_Data$`Pr Cat 1`~km_brand_imp$cluster, main = "Distribution
of Price cat 1 across the clusters", xlab="Clusters", ylab="Price cat 1")
```



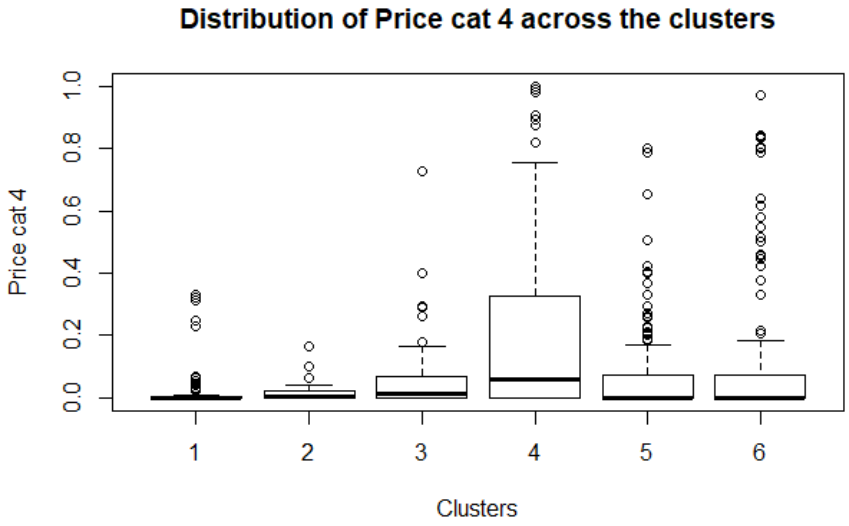

```
> boxplot(BathSoap_Data$`Pr Cat 2`~km_brand_imp$cluster, main = "Distribution
of Price cat 2 across the clusters", xlab="Clusters", ylab="Price cat 2")
```




```
> boxplot(BathSoap_Data$`Pr Cat 3`~km_brand_imp$cluster, main = "Distribution
of Price cat 3 across the clusters", xlab="Clusters", ylab="Price cat 3")
```




```
> boxplot(BathSoap_Data$`Pr Cat 4`~km_brand_imp$cluster, main = "Distribution
of Price cat 4 across the clusters", xlab="Clusters", ylab="Price cat 4")
```



From above boxplots for basis of purchase variables the distribution can be concluded across each cluster. Here price categories percentage distribution is not even all the clusters which tells that households purchase preferences change as per the clusters segmented on Brand loyalty basis.

Assignment 6:

Answer 1:

Similarity Matrix:

	P1	P2	P3	P4	P5	P6
P1	**1**	**0.7**	**0.65**	**0.4**	**0.2**	**0.05**
P2	**0.7**	**1**	**0.95**	**0.7**	**0.5**	**0.35**
P3	**0.65**	**0.95**	**1**	**0.75**	**0.55**	**0.4**
P4	**0.4**	**0.7**	**0.75**	**1**	**0.8**	**0.65**
P5	**0.2**	**0.5**	**0.55**	**0.8**	**1**	**0.85**
P6	**0.05**	**0.35**	**0.4**	**0.65**	**0.85**	**1**

We will now convert the above matrix to dissimilarity matrix using the formula:

Dissimilarity score = 1 – Similarity score

Dissimilarity matrix looks like below:

	P1	P2	P3	P4	P5	P6
P1	**0**	**0.3**	**0.35**	**0.6**	**0.8**	**0.95**

P2	**0.3**	**0**	**0.05**	**0.3**	**0.5**	**0.65**
P3	**0.35**	**0.05**	**0**	**0.25**	**0.45**	**0.6**
P4	**0.6**	**0.3**	**0.25**	**0**	**0.2**	**0.35**
P5	**0.8**	**0.5**	**0.45**	**0.2**	**0**	**0.15**
P6	**0.95**	**0.65**	**0.6**	**0.35**	**0.15**	**0**

(a) Single-link clustering

Single-link distance between clusters C1 and C2 is the minimum distance between any object in C1 and any object in C2

Iteration 1: The clusters P2 and P3 are closest with shortest distance of 0.05

Using the distance matrix distance between cluster (P2, P3) and P1 is computed as:

min (0.3, 0.35) = 0.3

Likewise, we calculate all the distances and the updated distance matrix looks like below:

	P1	**P2&P3**	**P4**	**P5**	**P6**
P1	**0**	**0.3**	**0.6**	**0.8**	**0.95**
P2&P3	**0.3**	**0**	**0**	**0.25**	**0.45**
P4	**0.6**	**0.25**	**0**	**0.2**	**0.35**
P5	**0.8**	**0.45**	**0.2**	**0**	**0.15**
P6	**0.95**	**0.6**	**0.35**	**0.15**	**0**

Iteration 2: The clusters P5 and P6 are closest with shortest distance of 0.15

Updated matrix:

	P1	**P2&P3**	**P4**	**P5&P6**
P1	**0**	**0.3**	**0.6**	**0.8**
P2&P3	**0.3**	**0**	**0.25**	**0.45**
P4	**0.6**	**0.25**	**0**	**0.2**
P5&P6	**0.8**	**0.45**	**0.2**	**0**

Iteration 3: The clusters P4 and P5&P6 are closest with shortest distance of 0.2

Updated matrix:

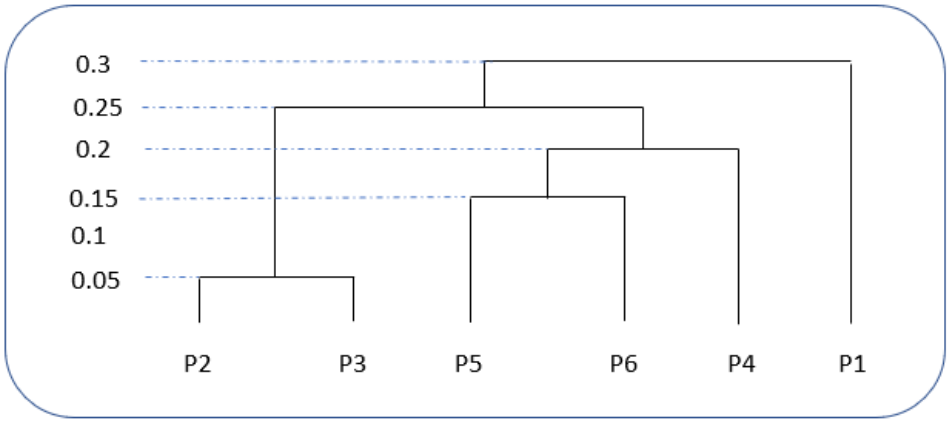
	P1	P2&P3	P4&P5&P6
P1	**0**	**0.3**	**0.6**
P2&P3	**0.3**	**0**	**0.25**
P4&P5&P6	**0.6**	**0.25**	**0**

Iteration 4: The clusters P2&P3 and P4&P5&P6 are closest with shortest distance of 0.25

Updated matrix:

	P1	P2&P3&P4&P5&P6
P1	**0**	**0.3**
P2&P3P2&P3&P4&P5&P6	**0.3**	**0**

Dendrogram:



(b) Complete-link clustering

Complete-link distance between clusters C1 and C2 is the maximum distance between any object in C1 and any object in C2

Iteration 1: The clusters P2 and P3 are closest with shortest distance of 0.05

Using the distance matrix distance between cluster (P2, P3) and P1 is computed as:

max (0.3, 0.35) = 0.35

Likewise, we calculate all the distances and the updated distance matrix looks like below:

	P1	P2&P3	P4	P5	P6
P1	0	0.35	0.6	0.8	0.95
P2&P3	0.35	0.05	0.3	0.5	0.65
P4	0.6	0.3	0	0.2	0.35
P5	0.8	0.5	0.2	0	0.15
P6	0.95	0.65	0.35	0.15	0

Iteration 2: The clusters P5 and P6 are closest with shortest distance of 0.15

Updated matrix:

	P1	P2&P3	P4	P5&P6
P1	0	0.35	0.6	0.95
P2&P3	0.35	0.05	0.3	0.65
P4	0.6	0.3	0	0.35
P5&P6	0.95	0.65	0.35	0.15

Iteration 3: The clusters P4 and P2&P3 are closest with shortest distance of 0.3

Updated matrix:

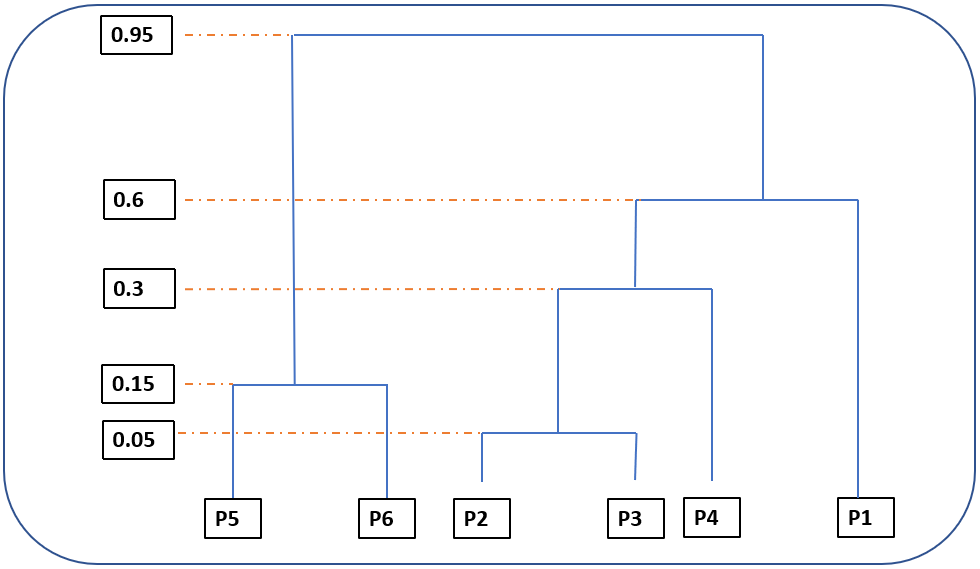
	P1	P2&P3&P4	P5&P6
P1	0	0.6	0.95
P2&P3&P4	0.6	0.3	0.65
P5&P6	0.95	0.65	0.15

Iteration 4: The clusters P2&P3&P4 and P1 are closest with shortest distance of 0.6

Updated matrix:

	P1&P2&P3&P4	P5&P6
P1&P2&P3&P4	0.6	0.95
P5&P6	0.95	0.15

Dendrogram:



Answer 2:

According to us, the given clustering cannot be the final result of the k-means algorithm.

The final iteration in k means in a cluster makes sure that clusters are homogeneous i.e within cluster variance is minimum. This means, each data point is nearest to its centroid than previous iterations. This isn't the case with the figure.

The blue points near the red dot will get assigned to cluster 2 because they are closer to the centroid of cluster 2 and not closer to the centroid of cluster 1. Hence, this cannot be the final result of k-means algorithm.

Also, this cannot be the final result of the correct agglomerative algorithm.

In agglomerative, we start by defining each point to be a cluster and then combine existing clusters at each step.

There are 24 data points given. Initially there will be 24 clusters.

The agglomerative algorithm selects 2 clusters whose distance is the smallest. Here there is a tie amongst 12 pairs and we end up getting 12 clusters to begin with. We continue merging the clusters to get the two clusters in the end, and the resulting two clusters will have equal number of data points each. Hence, it is not possible to end up with the clustering given, in which we have only one data point in 1 cluster and 23 data points in other.

Answer 3:

For an individual point i, silhouette value is calculated as:

s = (b-a) / max(a,b)

where a: average distance of i to the points in the same cluster

b: min (average distance of i to points in another cluster)

Let A = (0,0), B = (0,1), C = (2,3) be the points in cluster 1 and D = (3,3), E = (3,4) be the points in cluster 2.

Manhattan distances are calculated and tabulated below:

Formula used: | x1 – x2 | + | y1 – y2 |

Points	Manhattan distance between them
A-B	1
A-C	5
A-D	6
A-E	7
B-C	4
B-D	5
B-E	6
C-D	1
C-E	2
D-E	1

The silhouette coefficient s for all the points is tabulated below:

Point	a	b	s
A	3	6.5	0.538
B	2.5	5.5	0.545
C	4.5	1.5	-0.667
D	1	4	0.75
E	1	5	0.8

The silhouette coefficient lies between -1 and 1.

The closer the silhouette coefficient is to 1, the better is the clustering. The value of a is a measure of how dissimilar i is to its own cluster, a small value means it is well matched. Also, a large value of b means that i is badly matched to its neighboring cluster. Thus, s close to 1 means that the data is appropriately clustered.

From these coefficients as calculated above, we learn below points:

1. The points D and E are appropriately clustered in Cluster 2 as they have s value close to 1.
2. Point C is very dissimilar to Cluster 1 points A and B as it's a value is high (4.5). The s value of point C is in negative. Hence, we can say that is in inappropriately clustered in Cluster 1.

3. Points A and B are not very dissimilar to the points in Cluster 1 (A, B and C) but they are also not very similar. They are somewhere in between. We believe that this could be because of the inappropriate clustering of Point C in Cluster 1.

The average silhouette over all data of a cluster is a measure of how tightly grouped all the data in the cluster are. Thus, the average silhouette over all data of the entire dataset is a measure of how appropriately the data have been clustered.

Average silhouette coefficient for Cluster 1: (0.538 + 0.545 - 0.667)/3 = **0.14**

Average silhouette coefficient for Cluster 2: (0.75 + 0.8)/2 = **0.775**

Interpretation:

1. Cluster 1 is very loosely grouped and the data in it is not appropriately clustered.
2. Cluster 2 is tightly grouped and the data in it is appropriately clustered as compared to Cluster 1.

Average silhouette coefficient for overall data: (0.538+0.545-0.667+0.75+0.8)/5 = **0.393**

As the average silhouette for overall data is only 0.393, we can say that the data is loosely grouped and also that the data is not appropriately clustered.