Solution of HW3

1. Solution:

Part I Using the raw data and the combined 301 students from both schools

(1) 'VARIMAX' rotated result

For factor number from 1 to 8, do factor analysis respectively and get p-values for Bartlett corrected LRT, which are summarized in the table below:

# of factors	p-value	# of factors	p-value
1	7.2760e-116	5	0.0264
2	2.9687e-41	6	0.2967
3	2.9366e-16	7	0.6316
4	1.7106e-04	8	0.8010

If we think alpha=0.05, then for number of factors equals to 1-5, we all reject null and there is no factor; while for number of factors greater or equal to 6, we all don't reject the null and think that the assumed factor model is correct. So we need number of factors greater or equal to 6. In the meanwhile, we want to achieve data reduction as much as possible, so I will choose 6 factors in my model.

Now using 6 factor model to do factor analysis and the result is copied below:

lambdas =

0.2465	0.6198	0.1538	0.2059	-0.0310	0.0134
0.0884	0.4873	0.0117	0.0303	-0.0646	0.1019
0.1558	0.4429	0.0046	0.1204	0.0089	-0.0423
0.0045	0.6314	0.1780	0.1437	0.0358	0.0219
0.8142	0.1383	0.0113	0.1031	0.0796	0.0482
0.7884	0.1769	0.1428	0.0754	0.0753	0.0656
0.8685	0.0983	0.0171	0.1167	-0.0302	0.0831
0.6831	0.1883	0.1028	0.1332	0.0083	0.1651
0.7987	0.2331	0.0938	0.0482	0.0461	0.1263
0.0852	-0.0984	0.1680	0.5152	0.8078	-0.0084
0.2573	0.0906	0.2712	0.5981	0.1014	0.1211
0.0567	0.1747	0.0609	0.5372	0.2643	0.1108
0.1098	0.3438	0.0881	0.6408	0.0261	0.0640
0.1315	0.0305	0.6159	0.0218	-0.0358	0.1612
-0.0420	0.1889	0.6288	-0.0650	0.0741	-0.0480
0.1217	0.3600	0.4549	0.1100	0.0136	0.2173
0.0478	-0.0145	0.5506	0.2509	0.1277	0.0558
0.0891	0.1223	0.4399	0.2038	0.0342	0.1216
0.1837	0.1013	0.2713	0.1488	-0.0671	0.4973
0.2871	0.4541	0.2184	-0.0508	0.0693	0.2443
0.2314	0.3945	0.1419	0.2158	0.2977	0.3241
0.4359	0.3909	0.0983	0.0940	0.0105	0.3208
0.3550	0.5538	0.1539	0.0775	0.1877	0.2768
0.3719	0.1707	0.1927	0.1920	0.2891	0.4548

psi =

0.4879
0.7391
0.7632
0.5472
0.2985
0.3112
0.2143
0.4422
0.2785
0.0368
0.4694
0.5918
0.4466
0.5747
0.5552
0.5892
0.6120
0.7261
0.6085
0.5966
0.5305
0.5356
0.4257
0.4681

By the result above, we can see that the first factor is a 'verbal' factor because it loads heavily on tests 5-9; the second factor is a 'deduction of relations' factor because it loads heavily on tests 1-4, 20 and 23; the third factor is a 'memory' factor because it loads heavily on tests 14-18; the fourth factor is a 'speed' factor because it loads heavily on tests 10-13; the fifth factor is another 'speed' factor because it loads heavily on test 10; the sixed factor is a 'abstract thinking' factor because it loads heavily on tests 19 and 24.

(2) Without rotated result

Now I unrotated the loadings and the result is copied below:

ans =

0.4801	0.2284	0.3750	-0.2357	0.0207	-0.1812
0.3278	0.0271	0.3029	-0.2304	-0.0836	-0.0305
0.2956	0.1365	0.2071	-0.2544	-0.0224	-0.1505
0.2813	0.1778	0.5051	-0.2317	-0.0712	-0.1680
0.6788	0.3714	-0.3143	-0.0150	0.0032	-0.0611

0.6972	0.3791	-0.2117	0.0756	-0.0206	-0.0900
0.7489	0.3165	-0.3418	0.0235	0.0824	-0.0213
0.6556	0.3288	-0.1292	0.0295	0.0393	0.0273
0.7507	0.3412	-0.1919	0.0257	-0.0517	-0.0372
-0.2783	0.9397	-0.0443	-0.0037	-0.0251	-0.0111
0.2155	0.5586	0.1617	0.0336	0.3776	0.0479
0.0188	0.5384	0.1857	-0.1898	0.2015	0.0829
0.2005	0.4459	0.3104	-0.2716	0.3798	0.0038
0.2299	0.1835	0.2975	0.4970	0.0459	-0.0341
0.0866	0.1586	0.4189	0.3973	-0.0955	-0.2642
0.3382	0.2462	0.4520	0.1757	-0.0216	-0.0106
0.0388	0.3800	0.2710	0.3744	0.1503	-0.0773
0.1703	0.2823	0.3009	0.2466	0.1146	-0.0256
0.3410	0.1936	0.2481	0.2041	0.0701	0.3600
0.4847	0.1985	0.2852	-0.0000	-0.2174	0.0223
0.3245	0.4828	0.2710	-0.1120	-0.1452	0.1554
0.5868	0.2544	0.1557	-0.0734	-0.0734	0.1424
0.5315	0.3685	0.2940	-0.1358	-0.2211	0.0481
0.3766	0.5176	0.1098	0.0830	-0.1127	0.3009

By the result above, we can see that the interpretations of factors are similar with those of varimax rotated loadings, but the interpretations become more difficult, as the factor loadings have less variance than the varimax loadings.

<u>Part II Using data of each school separately</u>

(1) Pasteur School

For factor number from 1 to 8, do factor analysis respectively and get p-values for Bartlett corrected LRT, which are summarized in the table below:

# of factors	p-value	# of factors	p-value
1	3.6725e-49	5	0.3905
2	1.4590e-16	6	0.7121
3	1.8481e-06	7	NA
4	0.0464	8	0.9529

If we think alpha=0.05, then for number of factors equals to 1-4, we all reject null and there is no factor; while for number of factors greater or equal to 5, we all don't reject the null and think that the assumed factor model is correct. So we need number of factors greater or equal to 5. In the meanwhile, we want to achieve data reduction as much as possible, so I will choose 5 factors in my model.

Now using 5 factor model to do factor analysis and the result is copied below:

lambdas =

0.3137	0.5779	0.1376	0.0960	0.0686
0.0368	0.5167	-0.0111	0.0591	-0.1452
0.0976	0.4439	-0.1772	0.0026	0.0988
-0.0162	0.6711	0.1895	0.1705	0.0109
0.8059	0.0431	-0.0668	0.0991	0.1433
0.7815	0.1578	0.0975	0.0558	0.2023
0.9043	0.0790	0.0051	0.1094	-0.0613
0.6838	0.1690	0.1388	0.1516	0.0144
0.7750	0.2492	0.0375	0.1010	0.1564
0.1408	-0.2070	0.1161	0.4967	0.6436
0.3489	0.0681	0.2312	0.6704	0.0679
0.0781	0.0968	-0.0064	0.5249	0.2190
0.0693	0.2717	0.0863	0.5436	0.0537
0.0418	0.0492	0.6900	-0.0262	0.0402
-0.1326	0.1255	0.6128	-0.1100	0.0964
0.0833	0.3866	0.4753	0.1753	0.1610
0.0676	-0.0543	0.5226	0.2887	0.0889
0.1004	-0.0053	0.4649	0.0892	-0.0073
0.0667	0.2441	0.3567	0.2405	0.0346
0.1230	0.5143	0.1888	0.0128	0.0954
0.2840	0.3879	0.1411	0.1934	0.4324
0.4685	0.4808	0.0289	0.1516	0.0511
0.3570	0.5877	0.1442	0.0906	0.2981
0.2179	0.2954	0.2265	0.2375	0.5301

psi =

0.5348
0.7069
0.7523
0.4842
0.3138
0.3108
0.1603
0.4614
0.3012
0.2629
0.3661
0.6610
0.6156
0.5175
0.5697
0.5610
0.6281
0.7658
0.7496

0.6755
0.5246
0.5230
0.4093
0.4766

By the result above, we can see that the first factor is a 'verbal' factor because it loads heavily on tests 5-9; the second factor is a 'deduction of relations' factor because it loads heavily on tests 1-4, 20, 22 and 23; the third factor is a 'memory' factor because it loads heavily on tests 14-18; the fourth factor is a 'speed' factor because it loads heavily on tests 10-13; the fifth factor is a 'mathematical abstract thinking' factor because it loads heavily on test 10, 21 and 24.

Now I unrotated the loadings and the result is copied below:
ans =

0.5306	0.2016	0.3673	-0.0887	0.0135
0.1905	0.1384	0.4345	-0.1531	0.1592
0.2278	0.0630	0.2958	-0.3184	-0.0545
0.3006	0.4377	0.4488	-0.1323	0.1223
0.7474	-0.3302	-0.1163	-0.0164	-0.0686
0.7949	-0.1787	-0.0243	0.0668	-0.1428
0.8117	-0.4029	-0.0125	0.1094	0.0801
0.7080	-0.1389	0.0233	0.1165	0.0621
0.8132	-0.1736	0.0437	-0.0127	-0.0734
0.3588	0.3597	-0.6598	-0.1548	-0.1400
0.5685	0.2653	-0.2974	0.0164	0.3894
0.3006	0.2851	-0.2615	-0.2409	0.2024
0.3366	0.3492	-0.0624	-0.1788	0.3366
0.1712	0.3789	0.0521	0.5491	-0.0738
0.0250	0.4428	0.1351	0.4360	-0.1590
0.3715	0.4957	0.1653	0.1672	-0.0065
0.2317	0.3776	-0.2012	0.3508	0.1099
0.1866	0.2327	-0.0344	0.3769	0.0442
0.2786	0.3613	0.0724	0.1408	0.1309
0.3380	0.2790	0.3569	-0.0406	-0.0581
0.5520	0.3265	0.0263	-0.1690	-0.1866
0.6221	0.0425	0.2611	-0.1288	0.0586
0.6225	0.2748	0.2816	-0.1532	-0.1580
0.5140	0.4264	-0.1005	-0.1207	-0.2297

By the result above, we can see that the interpretations of factors are similar with those of varimax rotated loadings, but the interpretations become more difficult, as the factor loadings have less variance than the varimax loadings.

(2) Grant-White School

By textbook, when analyzing Grant-White school, we prefer to use variables 25 and 26 instead of variables 3 and 4.

For factor number from 1 to 8, do factor analysis respectively and get p-values for Bartlett corrected LRT, which are summarized in the table below:

# of factors	p-value	# of factors	p-value
1	1.1180e-32	5	0.1559
2	2.5965e-13	6	0.2642
3	7.1471e-05	7	NA
4	0.0374	8	NA

If we think alpha=0.05, then for number of factors equals to 1-4, we all reject null and there is no factor; while for number of factors greater or equal to 5, we all don't reject the null and think that the assumed factor model is correct. So we need number of factors greater or equal to 5. In the meanwhile, we want to achieve data reduction as much as possible, so I will choose 5 factors in my model.

Now using 5 factor model to do factor analysis and the result is copied below:

lambdas =

0.1653	0.6547	0.1241	0.1809	0.2081
0.1078	0.4417	0.0870	0.0954	0.0035
0.1341	0.5593	-0.0478	0.1115	0.0941
0.2304	0.5334	0.0894	0.0810	0.0136
0.7383	0.1894	0.1914	0.1486	0.0558
0.7724	0.1866	0.0313	0.2477	0.1248
0.7982	0.2141	0.1425	0.0883	0.0511
0.5709	0.3430	0.2390	0.1275	0.0437
0.8078	0.2026	0.0333	0.2187	-0.0067
0.1806	-0.1080	0.8450	0.1803	0.0295
0.1951	0.0656	0.4217	0.4365	0.4194
0.0295	0.2321	0.6938	0.1022	0.1316
0.1862	0.4322	0.4771	0.0774	0.5408
0.1845	0.0614	0.0440	0.5523	0.0800
0.1043	0.1224	0.0586	0.5089	-0.0024
0.0697	0.4061	0.0556	0.5087	0.0548
0.1543	0.0718	0.2105	0.5947	-0.0260
0.0322	0.3000	0.3218	0.4575	0.0060
0.1562	0.2210	0.1438	0.3784	0.0460
0.3728	0.4618	0.1272	0.2930	-0.1926
0.1716	0.3981	0.4311	0.2381	0.0018
0.3637	0.4234	0.1141	0.3203	-0.0678
0.3614	0.5424	0.2486	0.2306	-0.1129
0.3680	0.1789	0.4955	0.3207	-0.0661

psi =

```
0.4526
0.7766
0.6456
0.6477
0.3573
0.2907
0.2863
0.4812
0.2573
0.2082
0.4134
0.4361
0.2525
0.6489
0.7118
0.5654
0.5724
0.5960
0.7608
0.5086
0.5695
0.5683
0.4474
0.4799
```

By the result above, we can see that the interpretations of factors are similar with those of varimax rotated loadings of the combined data.

Now I unrotated the loadings and the result is copied below:
ans =

0.5549	-0.0032	0.4659	-0.1495	0.0015
0.3444	-0.0287	0.2917	-0.0563	0.1250
0.3734	-0.1422	0.4267	-0.1045	0.0418
0.4634	-0.1045	0.3032	-0.1128	0.1482
0.7226	-0.2536	-0.2249	-0.0756	-0.0044
0.7208	-0.3742	-0.1685	-0.0139	-0.1453
0.7278	-0.3355	-0.2323	-0.1317	0.0131
0.6917	-0.1442	-0.0421	-0.1066	0.0801
0.7232	-0.4245	-0.1967	0.0169	-0.0214
0.5182	0.6035	-0.3795	0.0411	0.1158
0.5701	0.3495	-0.0240	0.0649	-0.3670
0.4872	0.5444	0.0052	-0.1180	0.1277
0.6305	0.3467	0.2011	-0.3833	-0.2058

0.3929	-0.0013	0.0649	0.3689	-0.2378
0.3456	0.0268	0.1282	0.3678	-0.1281
0.4559	0.0247	0.3781	0.2755	-0.0855
0.4530	0.1283	0.0333	0.4382	-0.1130
0.4749	0.2521	0.2182	0.2588	0.0177
0.4179	0.0511	0.1376	0.1964	-0.0669
0.5961	-0.1672	0.1806	0.1546	0.2271
0.5741	0.2267	0.1539	0.0252	0.1590
0.5946	-0.1395	0.1802	0.1287	0.0982
0.6650	-0.0636	0.2131	0.0332	0.2445
0.6571	0.1864	-0.1262	0.1451	0.1292

By the result above, we can see that the interpretations of factors are similar with those of varimax rotated loadings, but the interpretations become more difficult, as the factor loadings have less variance than the varimax loadings, and all the loadings of some factor are all very small.

Part III Comparison of the results above

By comparing the results above, we found the following phenomenon:

(1) When we use the combined data, we need to use six factors, while for each school we only need to use five factors.

(2) When we use varimax rotation for the combined data and for each school, the loadings and interpretation of factors are very similar.

(3) When we use factor analysis without rotation for the combined data and for each school, the variance of loadings for the factors become smaller and the factors are much more difficult to interpret.

Matlab code:

```
[lambdas,psi,T,stats]=factoran(X,1)
[lambdas,psi,T,stats]=factoran(X,2)
[lambdas,psi,T,stats]=factoran(X,3)
[lambdas,psi,T,stats]=factoran(X,4)
[lambdas,psi,T,stats]=factoran(X,5)
[lambdas,psi,T,stats]=factoran(X,6)
[lambdas,psi,T,stats]=factoran(X,7)
[lambdas,psi,T,stats]=factoran(X,8)
[lambdas,psi,T,stats]=factoran(X,6)
```

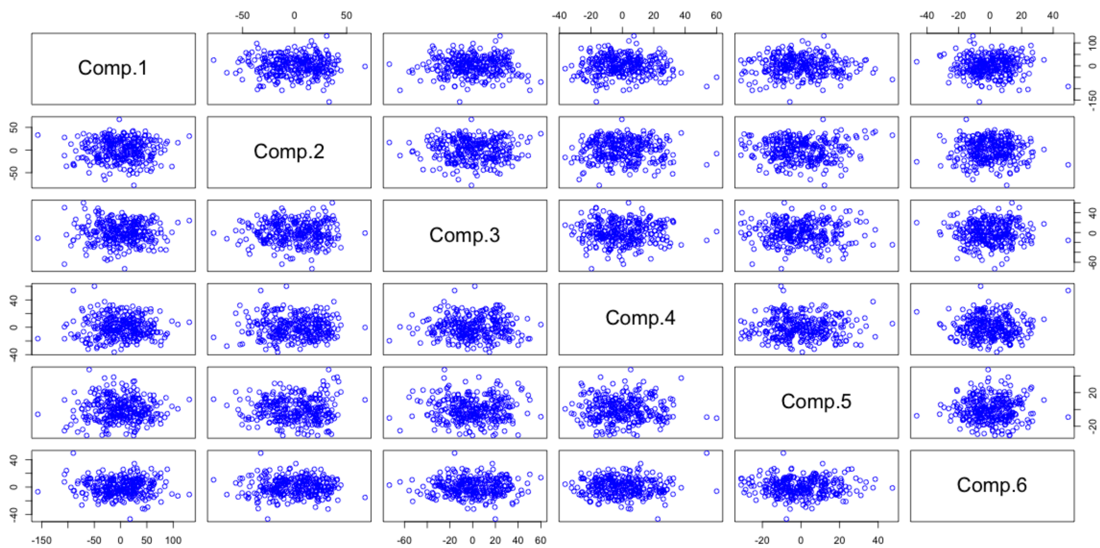
```
lambdas*inv(T)

[lambdas,psi,T,stats]=factoran(X1,1)
[lambdas,psi,T,stats]=factoran(X1,2)
[lambdas,psi,T,stats]=factoran(X1,3)
[lambdas,psi,T,stats]=factoran(X1,4)
[lambdas,psi,T,stats]=factoran(X1,5)
[lambdas,psi,T,stats]=factoran(X1,6)
[lambdas,psi,T,stats]=factoran(X1,7)
[lambdas,psi,T,stats]=factoran(X1,8)
[lambdas,psi,T,stats]=factoran(X1,5)
lambdas*inv(T)

X2=psychtests(:,[1:2 25:26 5:24])
[lambdas,psi,T,stats]=factoran(X2,1)
[lambdas,psi,T,stats]=factoran(X2,2)
[lambdas,psi,T,stats]=factoran(X2,3)
[lambdas,psi,T,stats]=factoran(X2,4)
[lambdas,psi,T,stats]=factoran(X2,5)
[lambdas,psi,T,stats]=factoran(X2,6)
[lambdas,psi,T,stats]=factoran(X2,7)
[lambdas,psi,T,stats]=factoran(X2,8)
[lambdas,psi,T,stats]=factoran(X2,5)
lambdas*inv(T)
```

2. Solution:

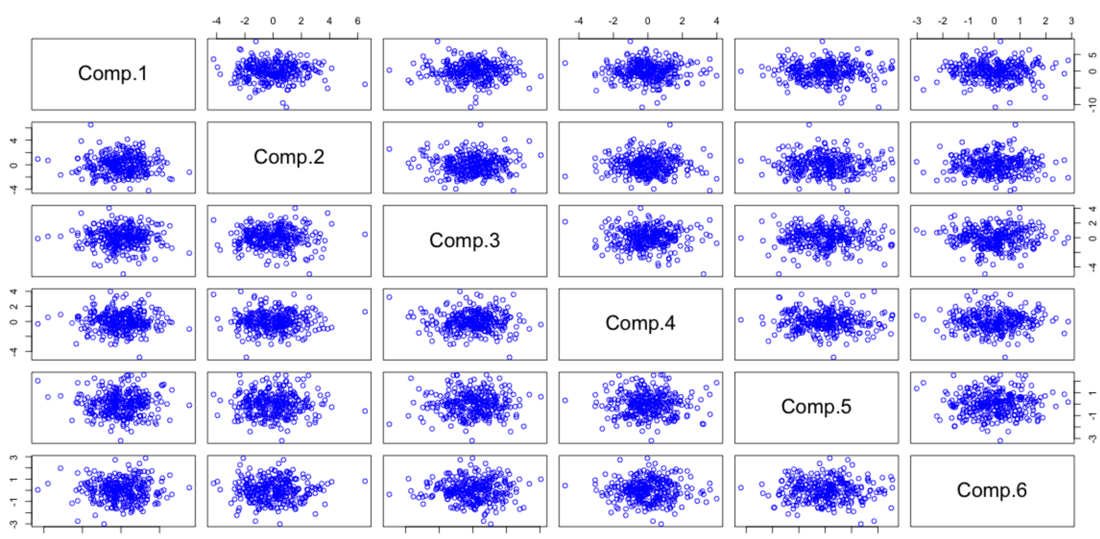
(1) Decide whether to standardize or sphere the data

As the goal of our problem is to divide the 301 individuals into two clusters, so we hope our data have a somewhat apparent two-cluster shape. So we can use this to decide whether or not we should do some transformation for our data. In order to visualized the data, we do PCA for raw data, standardized data and sphered data and use first six principal components to get the following scatter plot matrix respectively:

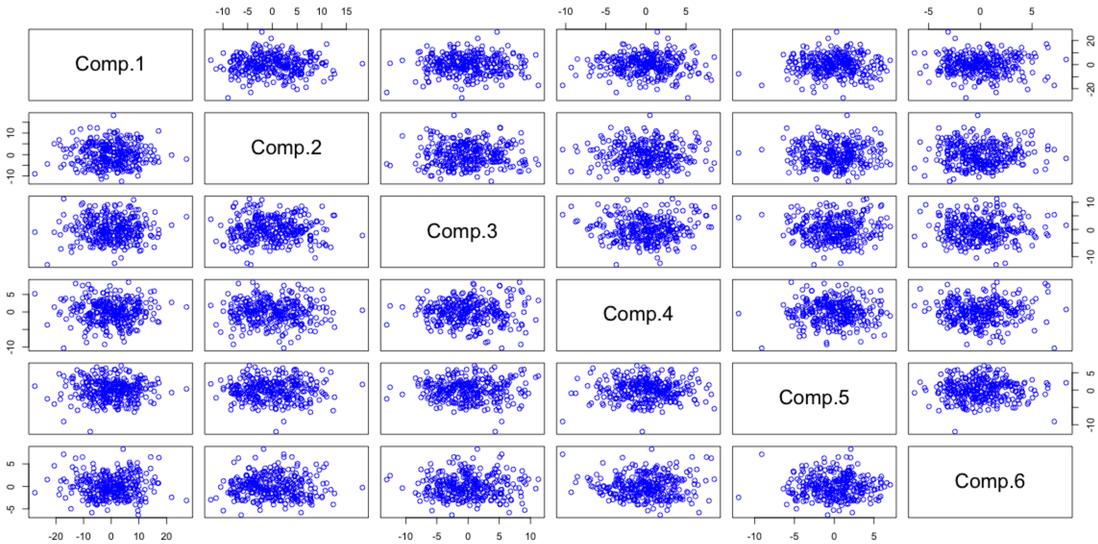
For raw data:



For standardized data:



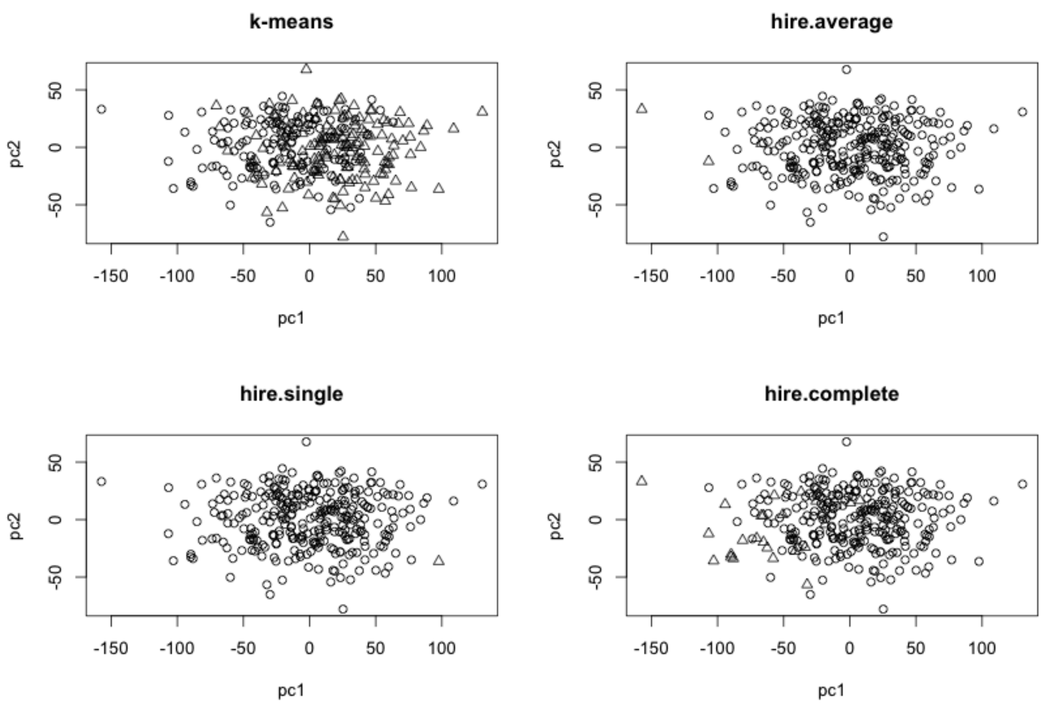
For sphered data:



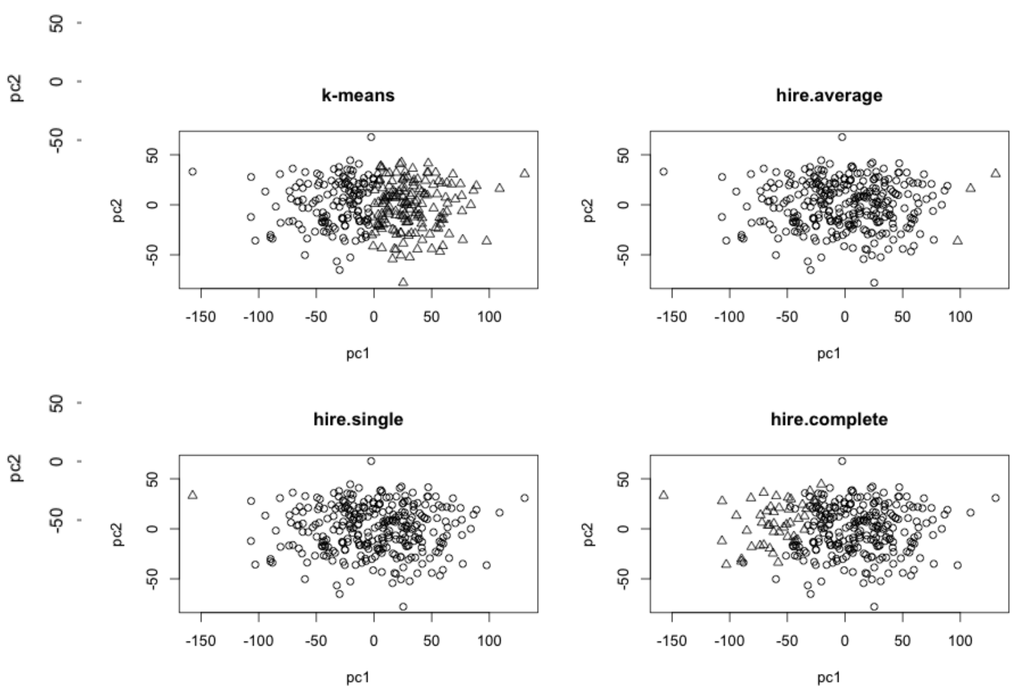
By examining the scatter plot matrix above, it seems that all of the three scatter plots do not have a clear two-group shape. However, they do have a little differences: the plots for standardized data looks like that they have only one center and all the data scatter around this center, so I do not think standardized data is a good choice; for the sphered data, the scatter plots are relatively too dispersive, so it also not a good choice; the raw data is better than these two from these aspects.
In conclusion, I prefer to use the raw data to do clustering.

(2) Choice of dissimilarity measure
Firstly, using squared 2-norm:

Secondly, using standardized distance:



Finally, using Mahalanobis distance:

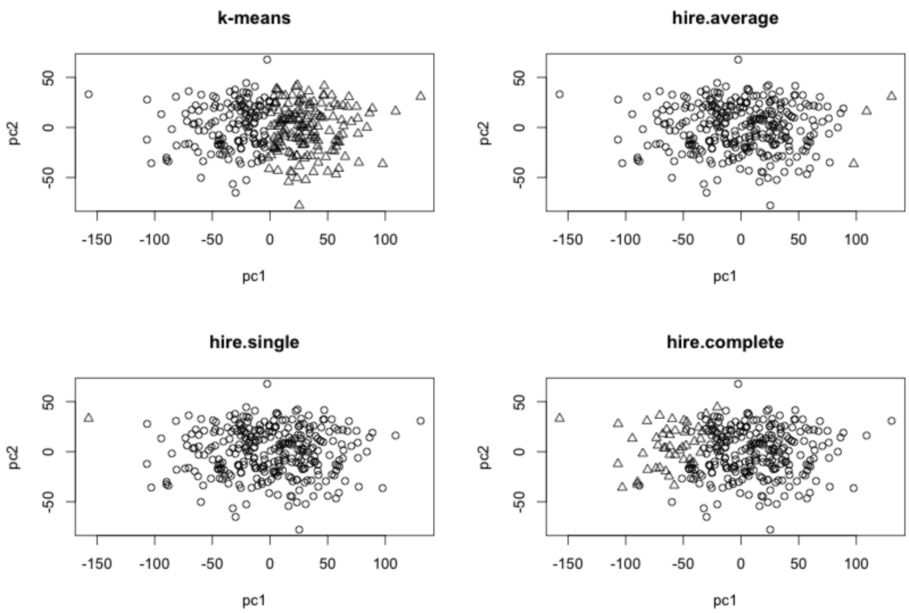


By examining the clustering result of the three dissimilarity measures, I think the squared 2-norm dissimilarity is preferable, as it provided a clear boundary.

(3) Choice of algorithm

As from part two, we know that squared 2-norm is a preferable dissimilarity measure, I will use this dissimilarity to choose an appropriate algorithm.

As the data is numerical and there are not severe outliers, so I think K-means will do a better job than K-medroid. So I will use K-means, hierarchical algorithm with single, average and complete linkage respectively. The cluster results from K-means,



hierarchical algorithm with single, average and complete linkage is copied below:

Comparing the results by eye, I think K-means algorithm did a better job, as it provided a clear boundary between two clusters and the number of individuals in each clusters are similar. While the hierarchical methods tend to cluster most of the individuals into one group.

To be more precise, I want to compare the within-cluster scatter for the four methods above, and the result is summarized in the following table:

	W(c)
K-means	843911.4
Hierarchical-average	1143020
Hierarchical-single	1156451

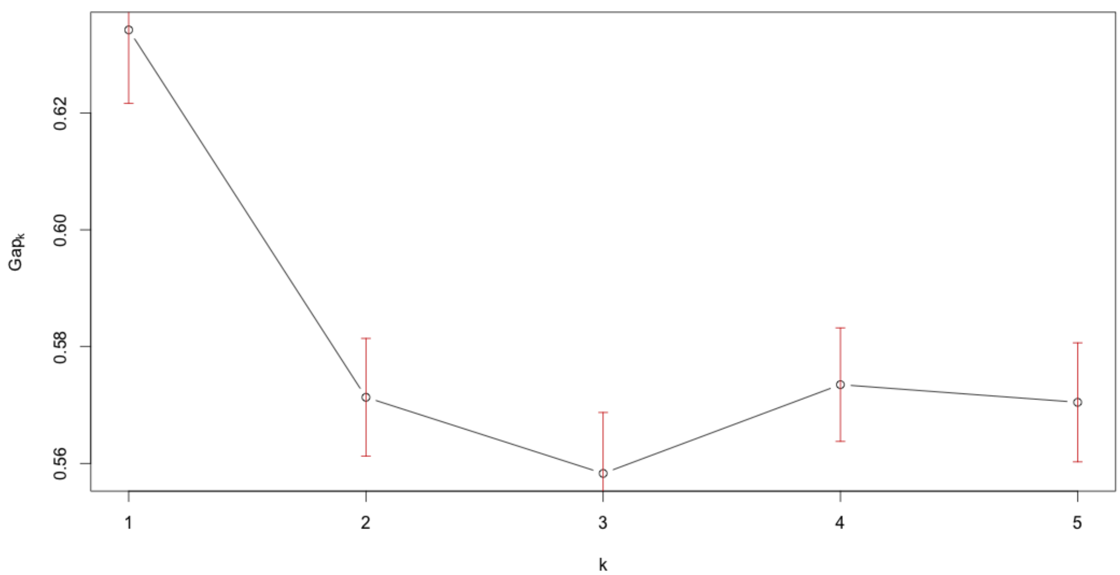
Hierarchical-complete	960582.3

We can see that K-means has the smallest within-cluster scatter, so K-means is the algorithm that I want to use and the clustering result is given above.

(4) Are there really two clusters

In order to see if there are really two clusters, I got the gap statistics and plot the result to give a clear visualization:

	logW	E.logW	gap	SE.sim
[1,]	8.751667	9.385887	0.6342198	0.01256870
[2,]	8.597053	9.168377	0.5713237	0.01008075
[3,]	8.549839	9.108129	0.5582900	0.01043546
[4,]	8.492614	9.066103	0.5734895	0.00971120
[5,]	8.459411	9.029875	0.5704640	0.01018293



As $Gap(1) > Gap(2) - s(2)$, so by the criterion of Gap statistics, the estimate of K is 1, which means that there is only one cluster.

R Code:

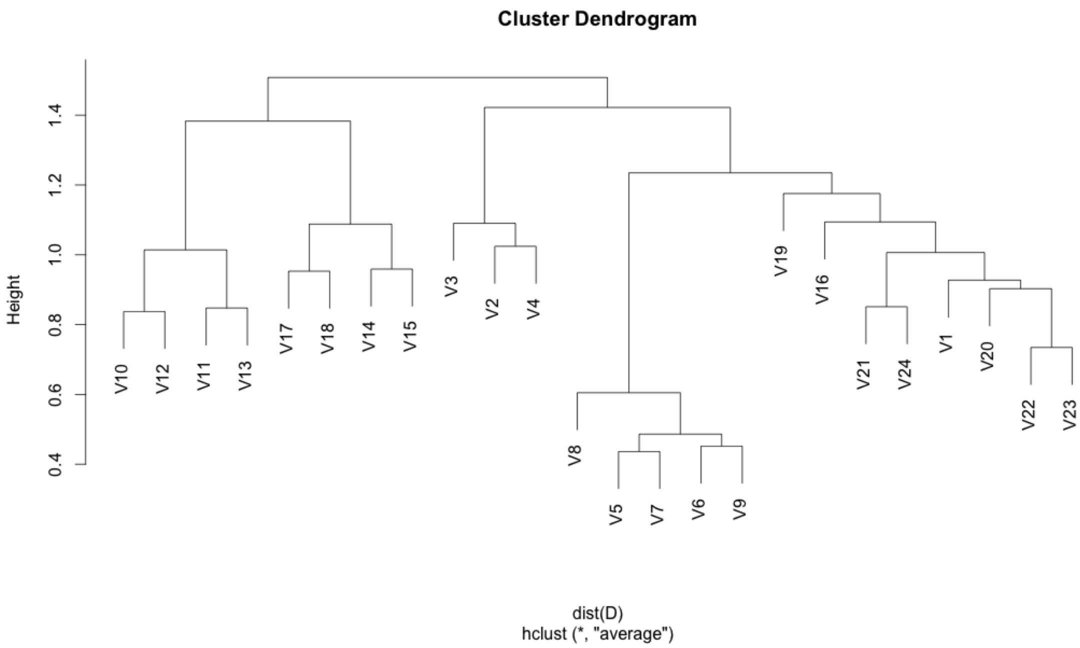
```
psych<-read.csv("24psychtests.csv",header = TRUE)
psychtest<-psych[, 5:28]
scale.psychtest<-scale(psychtest)

#sphere data
cov<-cov(psychtest)
mean<-colMeans(psychtest, na.rm = FALSE, dims = 1)
inverse<-solve(cov)
library(mgcv)
Z<-mroot(inverse,method='svd')
sphere.psychtest<-t(Z%*%(t(psychtest)-mean))

#PCA of raw data
spr <-princomp(psychtest)
U<-spr$loadings
L<-(spr$sdev)^2
Z <-spr$scores
pairs(Z[ ,c(1,2,3,4,5,6)],pch=c(rep(1,301)),col=c(rep("blue",301)))

#PCA of standardized data
spr2 <-princomp(scale.psychtest)
U2<-spr2$loadings
L2<-(spr2$sdev)^2
Z2 <-spr2$scores
pairs(Z2[ ,c(1,2,3,4,5,6)],pch=c(rep(1,301)),col=c(rep("blue",301)))

#PCA of sphered data
spr3 <-princomp(sphere.psychtest)
U3<-spr3$loadings
L3<-(spr3$sdev)^2
Z3 <-spr3$scores
pairs(Z3[ ,c(1,2,3,4,5,6)],pch=c(rep(1,301)),col=c(rep("blue",301)))


#kmeans for raw data
k<-2
kmeansobj<-kmeans(psychtest, k)
kmeansobj

#kmeans for standerdized data
k<-2
scale.kmeansobj<-kmeans(scale.psychtest, k)
scale.kmeansobj

#kmeans for sphered data
```

```
k<-2
sphere.kmeansobj<-kmeans(sphere.psychtest, k)
sphere.kmeansobj

#Hiearchical for raw data
tree.avg<-hclust(dist(psychtest),method='average')
plot(tree.avg)
membership<-cutree(tree.avg, k=2)
membership
tree.avg2<-hclust(dist(psychtest),method='single')
plot(tree.avg2)
membership2<-cutree(tree.avg2, k=2)
membership2
tree.avg3<-hclust(dist(psychtest),method='complete')
plot(tree.avg3)
membership3<-cutree(tree.avg3, k=2)
membership3

#Hiearchical for standardized  data
scale.tree.avg<-hclust(dist(scale.psychtest),method='average')
scale.membership<-cutree(scale.tree.avg, k=2)
scale.membership
scale.tree.avg2<-hclust(dist(scale.psychtest),method='single')
scale.membership2<-cutree(scale.tree.avg2, k=2)
scale.membership2
scale.tree.avg3<-hclust(dist(scale.psychtest),method='complete')
scale.membership3<-cutree(scale.tree.avg3, k=2)
scale.membership3

#Hiearchical for sphere  data
sphere.tree.avg<-hclust(dist(sphere.psychtest),method='average')
sphere.membership<-cutree(sphere.tree.avg, k=2)
sphere.membership
sphere.tree.avg2<-hclust(dist(sphere.psychtest),method='single')
sphere.membership2<-cutree(sphere.tree.avg2, k=2)
sphere.membership2
sphere.tree.avg3<-hclust(dist(sphere.psychtest),method='complete')
sphere.membership3<-cutree(sphere.tree.avg3, k=2)
sphere.membership3

library(graphics)
par(mfrow = c(2, 2))
plot(Z[,1],Z[,2],pch=kmeansobj$cluster,xlab="pc1",ylab="pc2",main="k-means")
plot(Z[,1],Z[,2],pch=membership,xlab="pc1",ylab="pc2",main="hire.average")
plot(Z[,1],Z[,2],pch=membership2,xlab="pc1",ylab="pc2",main="hire.single")
plot(Z[,1],Z[,2],pch=membership3,xlab="pc1",ylab="pc2",main="hire.complete")
```

```
par(mfrow = c(2, 2))
plot(Z[,1],Z[,2],pch=scale.kmeansobj$cluster,xlab="pc1",ylab="pc2",main="k-
means")
plot(Z[,1],Z[,2],pch=scale.membership,xlab="pc1",ylab="pc2",main="hire.average")
plot(Z[,1],Z[,2],pch=scale.membership2,xlab="pc1",ylab="pc2",main="hire.single")
plot(Z[,1],Z[,2],pch=scale.membership3,xlab="pc1",ylab="pc2",main="hire.complete
")

par(mfrow = c(2, 2))
plot(Z[,1],Z[,2],pch=sphere.kmeansobj$cluster,xlab="pc1",ylab="pc2",main="k-
means")
plot(Z[,1],Z[,2],pch=sphere.membership,xlab="pc1",ylab="pc2",main="hire.average"
)
plot(Z[,1],Z[,2],pch=sphere.membership2,xlab="pc1",ylab="pc2",main="hire.single"
)
plot(Z[,1],Z[,2],pch=sphere.membership3,xlab="pc1",ylab="pc2",main="hire.single"
)

#within-cluster scatter
x<-cbind(psychtest,membership)
c1<-x[x$membership==1, ]
c2<-x[x$membership==2, ]
cc1<-c1[, 1:24]
cc2<-c2[, 1:24]
mean1<-colMeans(cc1, na.rm = FALSE, dims = 1)
mean2<-colMeans(cc2, na.rm = FALSE, dims = 1)
w1<-t(t(cc1)-mean1)%*%(t(cc1)-mean1)
w2<-t(t(cc2)-mean2)%*%(t(cc2)-mean2)
w<-sum(diag(w1))+sum(diag(w2))
w

library(cluster)
gap<-clusGap(psychtest,FUN=kmeans,K.max=5)
gap
plot(gap)
```

3. Solution:

(a) The dendrogram of the sequence of clustering assignment is:



(b) Now if we want to cluster the variables into four clusters, the result of membership is:

```
> membership
 V1  V2  V3  V4  V5  V6  V7  V8  V9 V10 V11 V12
  1   2   2   2   1   1   1   1   1   3   3   3
V13 V14 V15 V16 V17 V18 V19 V20 V21 V22 V23 V24
  3   4   4   1   4   4   1   1   1   1   1   1
```

So the first cluster has V1, V5, V6, V7, V8, V9, V16, V19, V20, V21, V22, V23 and V24; the second group has V2, V3 and V4; the third group has V10, V11, V12, V13; the fourth group has V14, V15, V17 and V18.

This result has some connection to the varimax rotated factor loadings obtained in problem 2 (the combined data). We can see that variables within the same cluster in this problem tend to have large loadings in the same factor variable, for example, the third cluster variables V10, V11, V12, V13 all have relatively large loadings for the fourth factor variable. Similarly, the fourth group variables V14, V15, V17 and V18 all have large loadings for the third factor variables. The loadings are copied below.

0.2465	0.6198	0.1538	0.2059	-0.0310	0.0134
0.0884	0.4873	0.0117	0.0303	-0.0646	0.1019
0.1558	0.4429	0.0046	0.1204	0.0089	-0.0423
0.0045	0.6314	0.1780	0.1437	0.0358	0.0219
0.8142	0.1383	0.0113	0.1031	0.0796	0.0482
0.7884	0.1769	0.1428	0.0754	0.0753	0.0656
0.8685	0.0983	0.0171	0.1167	-0.0302	0.0831
0.6831	0.1883	0.1028	0.1332	0.0083	0.1651
0.7987	0.2331	0.0938	0.0482	0.0461	0.1263
0.0852	-0.0984	0.1680	0.5152	0.8078	-0.0084
0.2573	0.0906	0.2712	0.5981	0.1014	0.1211
0.0567	0.1747	0.0609	0.5372	0.2643	0.1108
0.1098	0.3438	0.0881	0.6408	0.0261	0.0640
0.1315	0.0305	0.6159	0.0218	-0.0358	0.1612
-0.0420	0.1889	0.6288	-0.0650	0.0741	-0.0480
0.1217	0.3600	0.4549	0.1100	0.0136	0.2173
0.0478	-0.0145	0.5506	0.2509	0.1277	0.0558
0.0891	0.1223	0.4399	0.2038	0.0342	0.1216
0.1837	0.1013	0.2713	0.1488	-0.0671	0.4973
0.2871	0.4541	0.2184	-0.0508	0.0693	0.2443
0.2314	0.3945	0.1419	0.2158	0.2977	0.3241
0.4359	0.3909	0.0983	0.0940	0.0105	0.3208
0.3550	0.5538	0.1539	0.0775	0.1877	0.2768
0.3719	0.1707	0.1927	0.1920	0.2891	0.4548

R Code:

```
psych<-read.csv("24psychtests.csv",header = TRUE)
psychtest<-psych[, 5:28]
cov<-cov(psychtest)
corr<-cov2cor(cov)
I<-matrix(rep(1,576),24,24)
D<-I-corr

#Hiearchical for dissimilarity matrix
tree.avg<-hclust(dist(D),method='average')
plot(tree.avg)
membership<-cutree(tree.avg, k=4)
membership
```

```
d <- dist(D)
fit <- cmdscale(d,eig=TRUE, k=2)
x <- fit$points[,1]
y <- fit$points[,2]
plot(x, y, xlab="Coordinate 1", ylab="Coordinate 2",
   main="MDS",        type="n")
text(x, y, labels = row.names(D), cex=.7,col=membership)
```

STAT 2221 HW#3

-1 ~~10~~ 9

1. Solution:

① Varimax rotation doesn't change the value of estimated communality h_j

② $\hat{h}_j^2 = \sum_{i=1}^{k} \hat{\lambda}_{ji}^2 = (\hat{\lambda}_{j1}\ \hat{\lambda}_{j2} \cdots \hat{\lambda}_{jk}) \begin{pmatrix} \hat{\lambda}_{j1} \\ \hat{\lambda}_{j2} \\ \vdots \\ \hat{\lambda}_{jk} \end{pmatrix}$ 10

$$= \underset{\sim}{a}' \begin{pmatrix} \hat{\lambda}_{11} & \hat{\lambda}_{12} & \cdots & \hat{\lambda}_{1k} \\ \hat{\lambda}_{21} & \hat{\lambda}_{22} & \cdots & \hat{\lambda}_{2k} \\ \vdots & \vdots & & \\ \hat{\lambda}_{p1} & \hat{\lambda}_{p2} & \cdots & \hat{\lambda}_{pk} \end{pmatrix} \left(\underset{\sim}{a}' \begin{pmatrix} \hat{\lambda}_{11} & \hat{\lambda}_{12} & \cdots & \hat{\lambda}_{1k} \\ \hat{\lambda}_{21} & \hat{\lambda}_{22} & \cdots & \hat{\lambda}_{2k} \\ \vdots & \vdots & & \\ \hat{\lambda}_{p1} & \hat{\lambda}_{p2} & \cdots & \hat{\lambda}_{pk} \end{pmatrix} \right)'$$

where $\underset{\sim}{a}' = (0, 0, \cdots, 1, 0 \cdots 0)$ (jth element is 1, all the other elements are 0), and $\hat{\Lambda} = \begin{pmatrix} \hat{\lambda}_{11} & \hat{\lambda}_{12} & \cdots & \hat{\lambda}_{1k} \\ \hat{\lambda}_{21} & \hat{\lambda}_{22} & \cdots & \hat{\lambda}_{2k} \\ \vdots & & & \\ \hat{\lambda}_{p1} & \hat{\lambda}_{p2} & \cdots & \hat{\lambda}_{pk} \end{pmatrix}$

i.e. $\hat{h}_j^2 = \underset{\sim}{a}' \hat{\Lambda} \hat{\Lambda}' \underset{\sim}{a}$

Now, if we do Varimax rotation, i.e. $\hat{\Lambda}^+ = \hat{\Lambda} \cdot G^+$, where $G^+ = \underset{G}{\operatorname{argmax}} \sum_{i=1}^{k} \{ Var(\{(\lambda_{1i}^*)^2, \cdots, (\lambda_{pi}^*)^2\}) \}$, then the new estimated communality, denoted as $\hat{h}_j^{+2} = \underset{\sim}{a}' \hat{\Lambda}^+ \hat{\Lambda}^{+\prime} \underset{\sim}{a}$

$$= \underset{\sim}{a}' \hat{\Lambda} G^+ G^{+\prime} \hat{\Lambda} \underset{\sim}{a}$$

As G^+ is an orthogonal matrix, so $G^+ G^{+\prime} = I$, so

$\hat{h}_j^{+2} = \underset{\sim}{a}' \hat{\Lambda} I \hat{\Lambda}' \underset{\sim}{a} = \underset{\sim}{a}' \hat{\Lambda} \hat{\Lambda}' \underset{\sim}{a} = \hat{h}_j^2$

In this way, we have Varimax rotation doesn't change the value of estimated communality h_j.