Rumainum-HW3

Lince Rumainum

September 17, 2019

Problem 1

Mathematics of PCA

Problem 1-a (i)

Create the correlation matrix without the non-numeric column in R:

```
corMat <- cor(Glass[,-10])
```

Below is the correlation matrix of the Glass data sets without the duplicate:

```
##              RI          Na           Mg          Al          Si
## RI  1.000000000 -0.19880214 -0.127525799 -0.40097326 -0.53899979
## Na -0.198802137  1.00000000 -0.278419975  0.16773502 -0.06488462
## Mg -0.127525799 -0.27841998  1.000000000 -0.47957521 -0.16243673
## Al -0.400973259  0.16773502 -0.479575211  1.00000000 -0.01619465
## Si -0.538999788 -0.06488462 -0.162436734 -0.01619465  1.00000000
## K  -0.287645126 -0.26415757  0.007616762  0.32368278 -0.19728091
## Ca  0.811182947 -0.27819366 -0.446197111 -0.25806764 -0.20714455
## Ba  0.001679071  0.32907979 -0.491817790  0.48064237 -0.10438937
## Fe  0.147083050 -0.23937377  0.085425844 -0.08058344 -0.09771680
##               K         Ca           Ba           Fe
## RI -0.287645126  0.8111829  0.001679071  0.147083050
## Na -0.264157570 -0.2781937  0.329079794 -0.239373767
## Mg  0.007616762 -0.4461971 -0.491817790  0.085425844
## Al  0.323682777 -0.2580676  0.480642369 -0.080583442
## Si -0.197280914 -0.2071445 -0.104389371 -0.097716800
## K   1.000000000 -0.3170324 -0.043652592 -0.009372226
## Ca -0.317032427  1.0000000 -0.112207559  0.126314271
## Ba -0.043652592 -0.1122076  1.000000000 -0.059729016
## Fe -0.009372226  0.1263143 -0.059729016  1.000000000
```

Problem 1-a (ii)

Computing eigen values and eigen vectors of corMat accordingly in R:

```
(eigenValues <- eigen(corMat)$values)
## [1] 2.510152168 2.058169337 1.407484057 1.144693344 0.914768873 0.528593040
## [7] 0.370262639 0.064267543 0.001608997

(eigenVectors <- eigen(corMat)$vectors)
##             [,1]        [,2]        [,3]       [,4]        [,5]
##  [1,]  0.5432231 -0.28911804 -0.08849541  0.1479796  0.07670808
##  [2,] -0.2676141 -0.26909913  0.36710090  0.5010669 -0.14626769
##  [3,]  0.1093261  0.59215502 -0.02295318  0.3842440 -0.11610001
##  [4,] -0.4269512 -0.29636272 -0.32602906 -0.1488756 -0.01720068
##  [5,] -0.2239232  0.15874450  0.47979931 -0.6394962 -0.01763694
##  [6,] -0.2156587  0.15305116 -0.66349177 -0.0733491  0.30154622
##  [7,]  0.4924367 -0.34678973  0.01380151 -0.2743430  0.18431431
##  [8,] -0.2516459 -0.48262056 -0.07649040  0.1299431 -0.24970936
##  [9,]  0.1912640  0.06089167 -0.27223834 -0.2252596 -0.87828176
```

```
##              [,6]        [,7]        [,8]        [,9]
##  [1,] -0.11455615  0.08223530 -0.75177166 -0.02568051
##  [2,]  0.55790564  0.15419352 -0.12819398  0.31188932
##  [3,] -0.30585293 -0.20691746 -0.07799332  0.57732740
##  [4,]  0.02014091 -0.69982052 -0.27334224  0.19041178
##  [5,] -0.08850787  0.20945417 -0.38077660  0.29747147
##  [6,]  0.24107648  0.50515516 -0.11064442  0.26075531
##  [7,]  0.14957911 -0.09984144  0.39885229  0.57999243
##  [8,] -0.65986429  0.35043794  0.14497643  0.19853265
##  [9,]  0.24066617  0.07120579 -0.01650505  0.01459278
```

Problem 1-a (iii)

Computing the principal components using **prcomp** with scale option in R:

```
glass.pca <- prcomp(Glass[,-10],scale = T)
```

Below is the principal components result from the **prcomp** function:

```
## Standard deviations (1, .., p=9):
## [1] 1.58434597 1.43463213 1.18637433 1.06990343 0.95643550 0.72704404
## [7] 0.60849210 0.25351044 0.04011231
##
## Rotation (n x k) = (9 x 9):
##           PC1         PC2         PC3        PC4         PC5         PC6
## RI -0.5432231  0.28911804 -0.08849541 -0.1479796  0.07670808 -0.11455615
## Na  0.2676141  0.26909913  0.36710090 -0.5010669 -0.14626769  0.55790564
## Mg -0.1093261 -0.59215502 -0.02295318 -0.3842440 -0.11610001 -0.30585293
## Al  0.4269512  0.29636272 -0.32602906  0.1488756 -0.01720068  0.02014091
## Si  0.2239232 -0.15874450  0.47979931  0.6394962 -0.01763694 -0.08850787
## K   0.2156587 -0.15305116 -0.66349177  0.0733491  0.30154622  0.24107648
## Ca -0.4924367  0.34678973  0.01380151  0.2743430  0.18431431  0.14957911
## Ba  0.2516459  0.48262056 -0.07649040 -0.1299431 -0.24970936 -0.65986429
## Fe -0.1912640 -0.06089167 -0.27223834  0.2252596 -0.87828176  0.24066617
##            PC7         PC8         PC9
## RI -0.08223530 -0.75177166 -0.02568051
## Na -0.15419352 -0.12819398  0.31188932
## Mg  0.20691746 -0.07799332  0.57732740
## Al  0.69982052 -0.27334224  0.19041178
## Si -0.20945417 -0.38077660  0.29747147
## K  -0.50515516 -0.11064442  0.26075531
## Ca  0.09984144  0.39885229  0.57999243
## Ba -0.35043794  0.14497643  0.19853265
## Fe -0.07120579 -0.01650505  0.01459278
```

Problem 1-a (iv)

The results from part (ii) and (iii) are the same because for the Principal Component Analysis model to be able to find a correlation between its variables, it has to do everything that part (ii) does. It reduces the original data's dimensions/features by computing its variance, its correlation matrix and a normalized eigen values and eigen vectors of that covariance matrix. Each of the principal components is responsible in explaining a certain percentage of the variation from the features in original data (highest percentage for PC1 and then PC2 (the next behind) and so on) and can be used as a linear combination to represent data that shows how it correlates with each other. Since the mathematics

behind **prcomp** method is the same, both (ii) and (iii) create the same results.

Problem 1-a (v)

Below show how to demonstrate that principal components 1 and 2 from **Problem 1-a (iii)** are orthogonal, in R:

Preview of PC1 and PC2 values:

```
##           PC1         PC2
## RI -0.5432231  0.28911804
## Na  0.2676141  0.26909913
## Mg -0.1093261 -0.59215502
## Al  0.4269512  0.29636272
## Si  0.2239232 -0.15874450
## K   0.2156587 -0.15305116
## Ca -0.4924367  0.34678973
## Ba  0.2516459  0.48262056
## Fe -0.1912640 -0.06089167
```

```
sum <- 0 # initial sum of dot product to zero
for (i in 1:nrow(glass.pca$rotation)){
  # calculate the dot product between PC1 and PC2
  sum <- sum + (glass.pca$rotation[i,1]*glass.pca$rotation[i,2])
}
sum # show the sum of the dot product
## [1] -4.857226e-17
```

Since their dot product is **-4.851226e-17**, which is very close to zero, it shows that they are orthogonal.

Application of PCA

Problem 1-b (i)

Visualizations of the principal component analysis results from the Glass data:

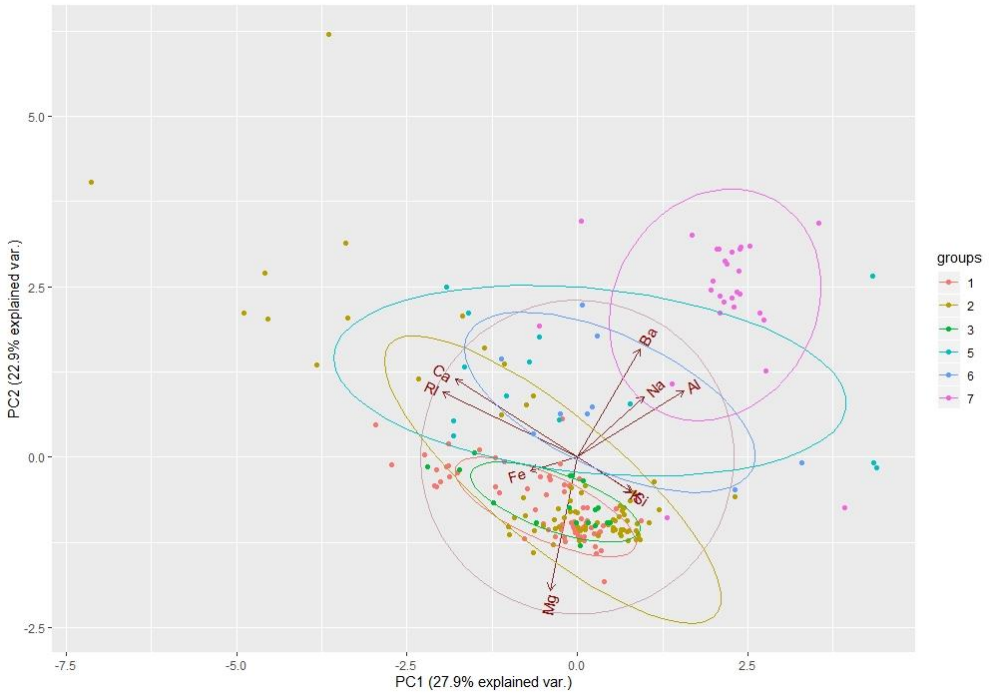

Figure 1 PC2 vs PC1 of Glass Data

Problem 1-b (ii)

From **Figure 2** and the preview of PC1 and PC2 values on Problem 1-a (v), each data shows the coefficient of Principal Component 1 (PC1) and Principal Component 2 (PC2). It shows each coefficient for RI, Na, Mg, Al, Si, K, Ca, Ba, and Fe for when each of them is represent to a reduce dimension through linear combination of PC1 and PC2. It also shows that PC1 explained the variations of the original data by 27.9% and 22.9% for PC2. When using PC1 and PC2 the data present have 50.8% explained variance of the original data. PC1 distinguished a glass that contains Iron (Fe) with higher refractive index (RI), Calcium (Ca) and less Silicon (Si), Barium (Ba), Sodium (Na) and Aluminum (Al). While PC2 distinguished the type of glass that is either float or non-float processed type contains more Magnesium (Mg) and Fe while the other three types (containers, tableware, and headlamps) contains less of those and more Ca, Ba, Na, and Al.

Problem 1-b (ii)

Based on the PCA summary results below and the cumulative sum plot below:

```
## Importance of components:
##                           PC1    PC2    PC3    PC4    PC5     PC6     PC7
## Standard deviation     1.5843 1.4346 1.1864 1.0699 0.9564 0.72704 0.60849
## Proportion of Variance 0.2789 0.2287 0.1564 0.1272 0.1016 0.05873 0.04114
## Cumulative Proportion  0.2789 0.5076 0.6640 0.7912 0.8928 0.95154 0.99268
##                            PC8     PC9
## Standard deviation     0.25351 0.04011
## Proportion of Variance 0.00714 0.00018
## Cumulative Proportion  0.99982 1.00000
```

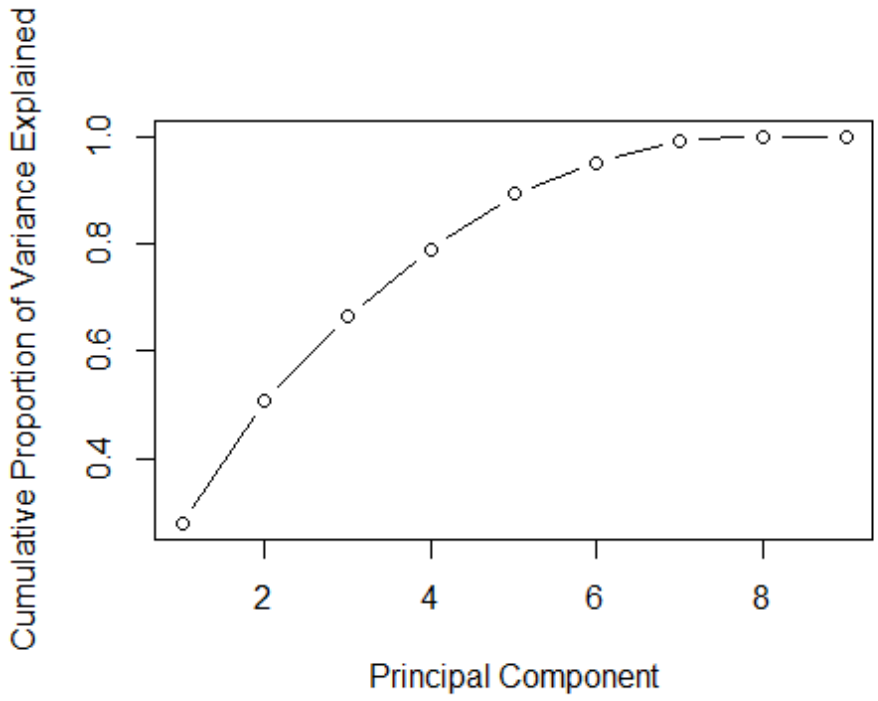

Figure 2 The Cummulative Proportion of Variance Explained vs. Principal Component

It is clear that the dimension can be reduce to 7-dimension for over 95% representation and could easily cut by about half (4-dimension) for close to 80% variations explained instead of using all 9-dimension.

Application of LDA

Problem 1-c (i)

Below shows the result of the LDA model for the Glass data:

```
## Prior probabilities of groups:
##          1          2          3          5          6          7
## 0.32394366 0.35680751 0.07981221 0.06103286 0.04225352 0.13615023
##
## Group means:
##            RI          Na         Mg          Al          Si           K
## 1  0.10586803 -0.21529537  0.6021709 -0.55566964 -0.03051835 -0.07127292
## 2  0.08928758 -0.35801082  0.2236651 -0.08333047 -0.07370057  0.03395576
## 3 -0.12668032  0.04037692  0.5986928 -0.50069475 -0.32346915 -0.14146475
## 5  0.19121411 -0.70579005 -1.3197807  1.17832829 -0.37327809  1.48675612
## 6 -0.29416453  1.52153712 -0.9514821 -0.16699477  0.71265830 -0.76375492
## 7 -0.40605128  1.27100801 -1.4829529  1.35761437  0.40154052 -0.26592900
##            Ca          Ba           Fe
## 1 -0.11782012 -0.32708816  0.005626553
## 2  0.08387772 -0.25209570  0.230146481
## 3 -0.12002632 -0.33526691 -0.002235610
## 5  0.82037781  0.02373083  0.035785004
## 6  0.28233912 -0.35297613 -0.586918470
## 7 -0.32450463  1.73435095 -0.449113729
##
## Coefficients of linear discriminants:
##            LD1        LD2       LD3          LD4        LD5
## RI  0.94568441 0.07527438 1.0863607 -0.818100075  2.3954624
## Na  1.93653461 2.56835550 0.3710009 -5.620203296 -2.1990082
## Mg  1.06291733 4.27171386 2.2661557 -9.823318476 -4.4675280
## Al  1.65414159 0.83192958 1.1005447 -3.192430110 -0.6006326
## Si  1.89406393 2.29624206 1.3286024 -5.857566773 -0.9918384
## K   1.02248031 1.19667679 0.8324237 -5.232631319 -2.0664499
## Ca  1.42618954 3.35575714 0.9098493 -9.431977031 -5.6893657
## Ba  1.14918330 1.69975242 1.2850261 -3.150403464 -2.3353652
## Fe -0.04927568 0.01955888 0.1195656 -0.002693645  0.1268396
##
## Proportion of trace:
##    LD1    LD2    LD3    LD4    LD5
## 0.8145 0.1168 0.0417 0.0158 0.0111
```

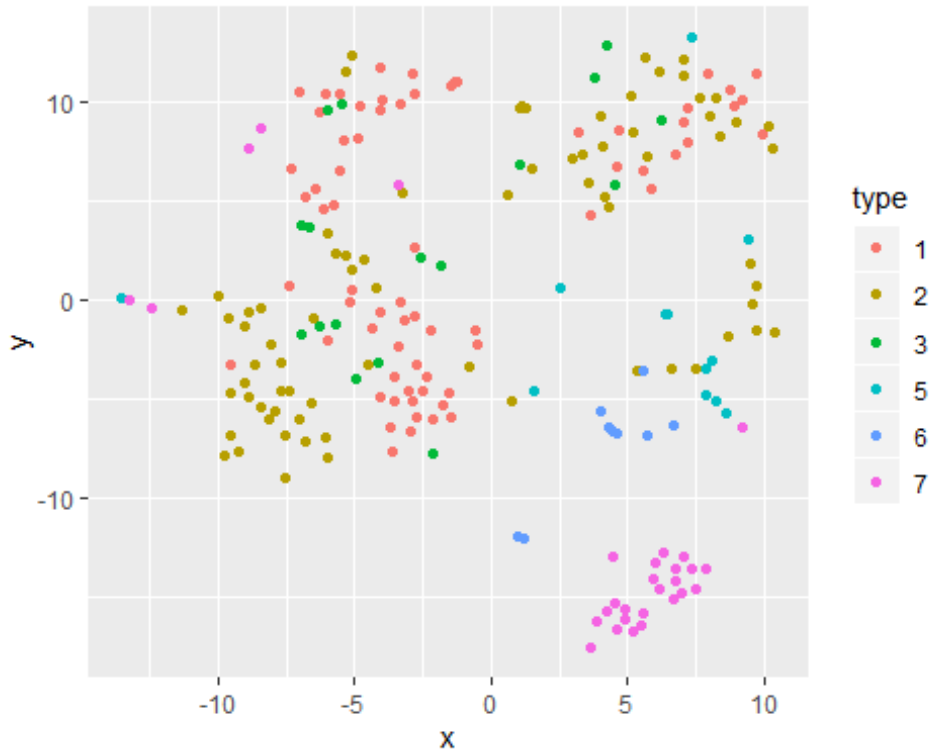

Figure 3 LD2 vs LD1 Using t-SNE Method

Problem 1-c (ii)

So, as stated in the coefficient of linear discriminant of LD1 in Problem 1-c (i), it shows each coefficient for RI, Na, Mg, Al, Si, K, Ca, Ba, and Fe for them to be represent to a reduce dimension. It also shows in the proportion of trace part that the LD1 explained the variations of the original data by 81.45%. When using LD1 with LD2, which explained 93.13% of the original data, it shows that those maximizes the separation in the glass type especially on type 7 (see **Figure 3**).

Problem 1-c (ii)

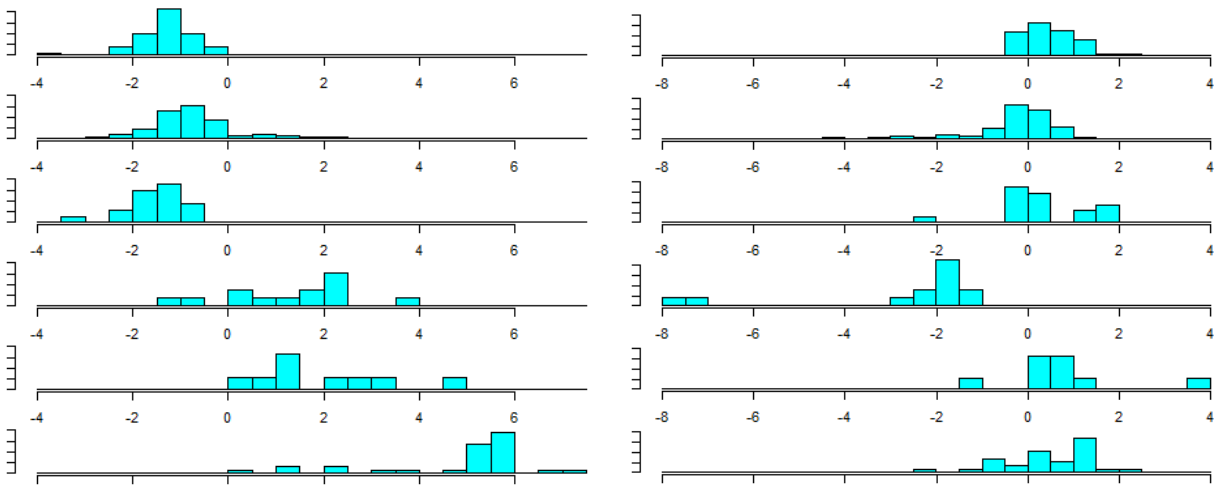

Figure 4 LD1 Histogram (left) and LD2 Histogram (right)

Both LDA histograms in **Figure 4** shows the variations of each Glass Type (Type 1, 2, 3, 5, 6, and 7 from top to bottom, respectively). From those histogram, it clearly shows how LD1 separated each type better than LD2 and how type 1, 2, and 3 (building windows float processed, building windows non-float processed, and vehicle windows float processed, respectively) are closely related which make sense since they are most likely to be made of similar elements than type 5, 6, and 7 (containers, tableware, and headlamps).

Problem 2

Problem 2-a

From analyzing the 11 evaluation features in Table 2 of the Moro et al. (2016) (excluding the total interactions), a Principal Component Analysis was made which resulted to the graph below. The zoom in version of the graph is made for a clearer version on how the features behave on each Principal Components. As you can see, link does not really engage other users or posted often as the other type of posting. Status is mostly posted but not as effective as posting a photo or video since it receives less likes, shares, and comments. Video creates the most impact out of all the other type of posting on Facebook because it shows that it has more positive impact on all the 11 features where photo is the next behind it. A better representation might be useful since PC1 and PC2 cumulative proportion is 69.26%.

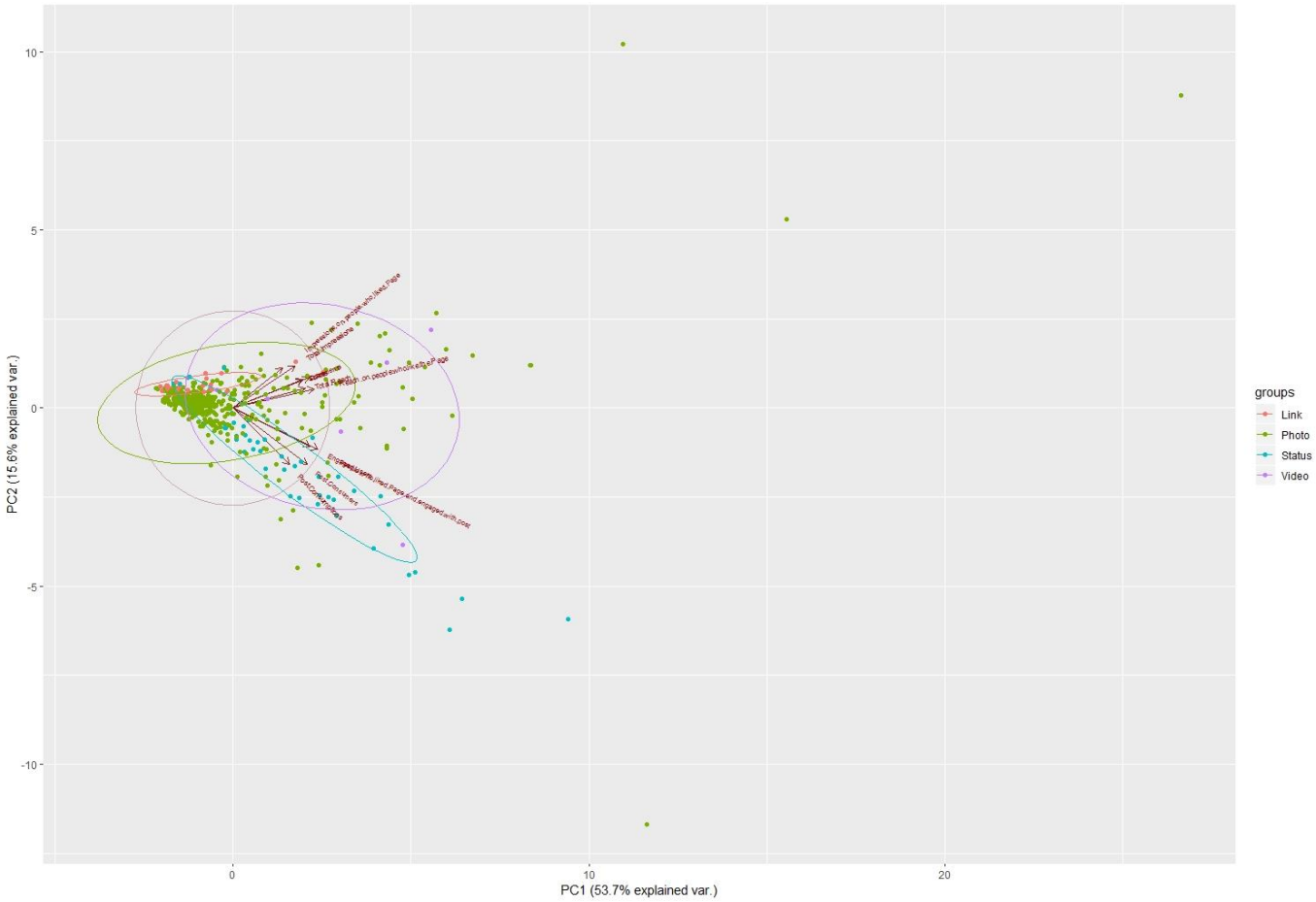

Figure 5 PC2 vs PC1 of fb.metrics Data

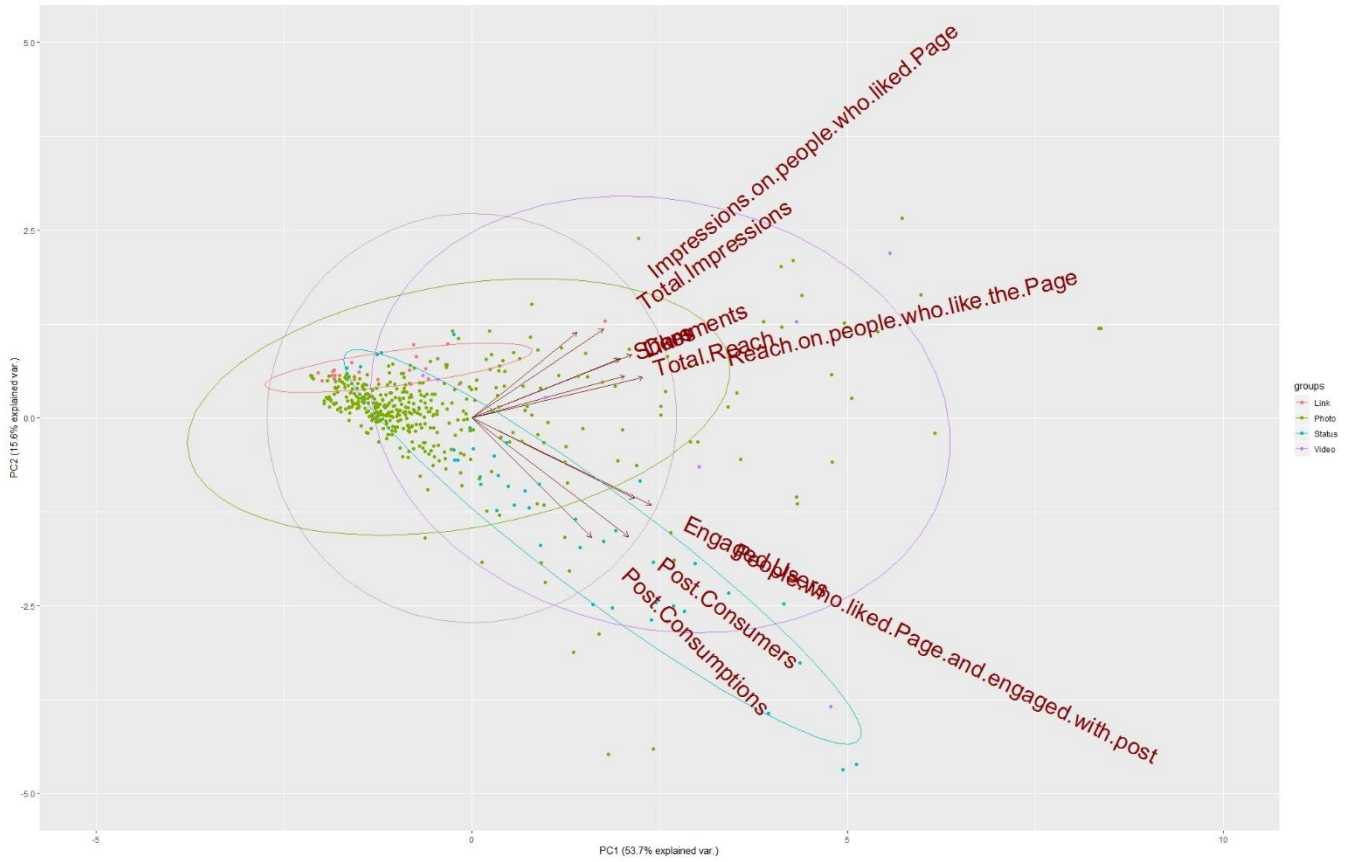

Figure 6 Zoom-In Version of PC2 vs PC1 of fb.metrics Data (Figure 5)

The 11 evaluation features in Table 2 of the Moro et al. (2016) (excluding the total interactions) are lifetime post total reach, lifetime post total impressions, lifetime engaged users, lifetime post consumers, lifetime post consumptions, lifetime post impressions by people who have liked a page, lifetime post reach by people who like a page, lifetime people who have liked a page and engaged with a post, comment, likes, and shares.

Since some of the features are close related to each other in the PCA plot (Figure 6), it shows that shares, likes and comments are almost on the same vectors and engaged users and people who have liked a page and engaged with a post are also. From the graph it makes sense that the total impressions and total reach are close to its impressions and reach on people who like the page, respectively. Share, likes, and comments closely related on how others react to a product and how post consumers and consumptions are closely related to how consumers engaged to that product(s).

Problem 2-b

Now a Linear Discriminant Analysis (LDA) is made on the same data (the 11 evaluation features in Table 2 of the Moro et al. (2016) (excluding the total interactions)). In the LDA, the separation for each type of posting are showing much clearer. The links are on the bottom right corner while status on the middle left side. Both have a little variation on LD1 with more variation on the LD2 while photos have a lot more variations in both directions than any other type. From the summary of the LDA model, the LD1 in this case has 78.41% of proportion of trace while LD2 has 14.63%. The cumulative proportion of this LDA1 and LDA2 model is 93.04% which is much better than the cumulative proportion of PC1 and PC2 model (69.26%) in problem 2-a.