Homework 3

1. Q: Perform latent class analysis of only the categorical variables for market segmentation using poLCA A: The categorical variables selected were Account Balance, Type of Apartment, Occupation, and Guarantors. These parameters were determined to be good identifiers of wealthy individuals.

```
set.seed(3)
German.Credit <- read.csv('German.Credit.csv')
my.german.new <- German.Credit %>%
  dplyr::select(Account.Balance,  Type.of.apartment, Occupation, Guarantors)
f1 <- with(my.german.new, cbind(Account.Balance,  Type.of.apartment ,Occupation, Guarantors)~1)
```

2. Q:Determine 2,3,..,K class/cluster solutions. Remember to run from multiple random starts. Use AIC criterion and intepretation based on graphs to interpret LCA solutions. A: Utilizing the AIC criterion, it is determined below that the model with the lowest AIC is three latent classes.

```
german.credit.train_ind <- sample(1:(nrow(my.german.new)),632, replace = F)
german.credit.train1<-my.german.new[german.credit.train_ind,]
min_aic <- 15000
for(i in 2:5){
  lc <- poLCA(f1, german.credit.train1, nclass=i, maxiter=3000,
              tol=1e-5, na.rm=FALSE,
              nrep=10, verbose=TRUE, calc.se=TRUE)
  if(lc$aic < min_aic){
    min_aic <- lc$aic
    LCA_best_model <- lc
  }
}
```

```
## Model 1: llik = -2112.524 ... best llik = -2112.524
## Model 2: llik = -2112.461 ... best llik = -2112.461
## Model 3: llik = -2112.524 ... best llik = -2112.461
## Model 4: llik = -2114.252 ... best llik = -2112.461
## Model 5: llik = -2112.46 ... best llik = -2112.46
## Model 6: llik = -2112.461 ... best llik = -2112.46
## Model 7: llik = -2114.252 ... best llik = -2112.46
## Model 8: llik = -2114.252 ... best llik = -2112.46
## Model 9: llik = -2112.461 ... best llik = -2112.46
## Model 10: llik = -2113.199 ... best llik = -2112.46
## Conditional item response (column) probabilities,
##  by outcome variable, for each class (row)
##
## $Account.Balance
##            Pr(1)  Pr(2)  Pr(3)  Pr(4)
## class 1:  0.2893 0.2823 0.0479 0.3805
## class 2:  0.2192 0.2946 0.0986 0.3876
##
## $Type.of.apartment
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.1968 0.7793 0.0239
## class 2:  0.1525 0.6018 0.2457
##
## $Occupation
##            Pr(1) Pr(2)  Pr(3)  Pr(4)
## class 1:  0.0000 0.321 0.6535 0.0255
## class 2:  0.0568 0.000 0.5731 0.3701
##
## $Guarantors
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.8847 0.0416 0.0737
## class 2:  0.9550 0.0450 0.0000
##
## Estimated class population shares
##  0.6654 0.3346
##
## Predicted class memberships (by modal posterior prob.)
##  0.7864 0.2136
##
## =========================================================
## Fit for 2 latent classes:
```

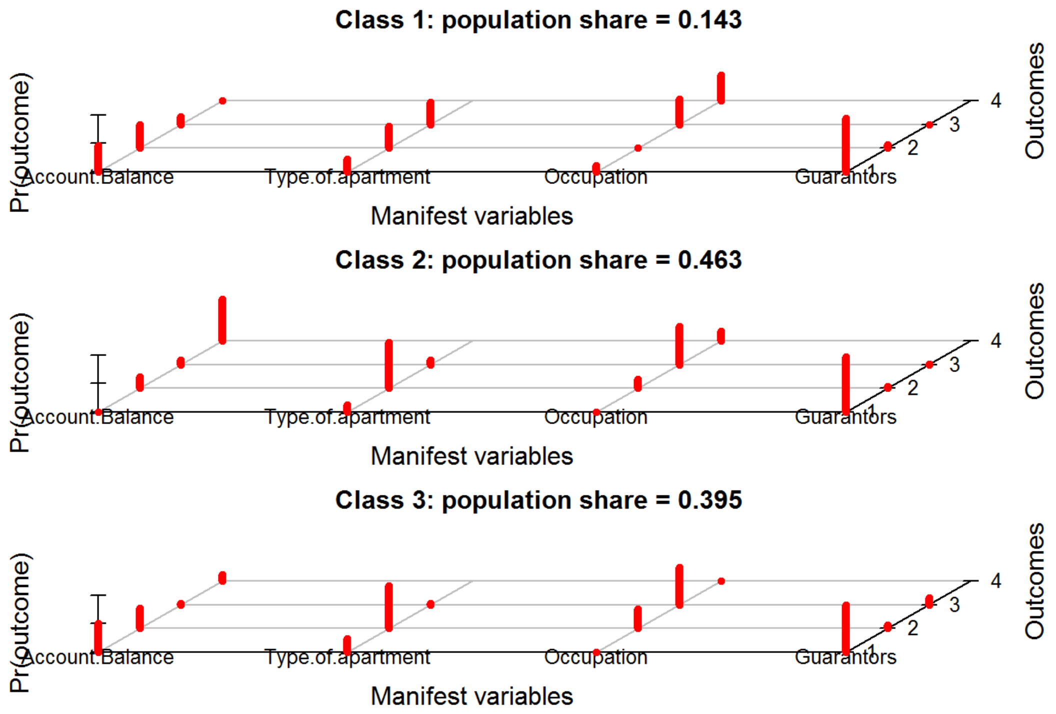
```
## =========================================================
## number of observations: 632
## number of estimated parameters: 21
## residual degrees of freedom: 122
## maximum log-likelihood: -2112.46
##
## AIC(2): 4266.921
## BIC(2): 4360.347
## G^2(2): 131.7005 (Likelihood ratio/deviance statistic)
## X^2(2): 169.4784 (Chi-square goodness of fit)
##
## Model 1: llik = -2092.882 ... best llik = -2092.882
## Model 2: llik = -2092.882 ... best llik = -2092.882
## Model 3: llik = -2092.881 ... best llik = -2092.881
## Model 4: llik = -2092.882 ... best llik = -2092.881
## Model 5: llik = -2092.882 ... best llik = -2092.881
## Model 6: llik = -2092.882 ... best llik = -2092.881
## Model 7: llik = -2092.882 ... best llik = -2092.881
## Model 8: llik = -2092.882 ... best llik = -2092.881
## Model 9: llik = -2092.882 ... best llik = -2092.881
## Model 10: llik = -2092.882 ... best llik = -2092.881
## Conditional item response (column) probabilities,
##  by outcome variable, for each class (row)
##
## $Account.Balance
##            Pr(1)  Pr(2)  Pr(3)  Pr(4)
## class 1:  0.0000 0.1996 0.0792 0.7212
## class 2:  0.5155 0.3488 0.0245 0.1113
## class 3:  0.4660 0.4049 0.1290 0.0000
##
## $Type.of.apartment
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.1259 0.8005 0.0736
## class 2:  0.2345 0.7435 0.0221
## class 3:  0.2246 0.3830 0.3923
##
## $Occupation
##            Pr(1)  Pr(2)  Pr(3)  Pr(4)
## class 1:  0.0067 0.1686 0.6591 0.1656
## class 2:  0.0000 0.3448 0.6552 0.0000
## class 3:  0.1132 0.0000 0.4373 0.4495
##
## $Guarantors
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.9652 0.0267 0.0081
## class 2:  0.8262 0.0576 0.1162
## class 3:  0.9445 0.0555 0.0000
##
## Estimated class population shares
##  0.4709 0.3893 0.1398
##
## Predicted class memberships (by modal posterior prob.)
##  0.4509 0.443 0.106
##
## =========================================================
## Fit for 3 latent classes:
## =========================================================
## number of observations: 632
## number of estimated parameters: 32
## residual degrees of freedom: 111
## maximum log-likelihood: -2092.881
##
## AIC(3): 4249.763
## BIC(3): 4392.127
## G^2(3): 92.54278 (Likelihood ratio/deviance statistic)
## X^2(3): 93.87396 (Chi-square goodness of fit)
##
## Model 1: llik = -2085.824 ... best llik = -2085.824
## Model 2: llik = -2086.361 ... best llik = -2085.824
## Model 3: llik = -2086.265 ... best llik = -2085.824
## Model 4: llik = -2085.985 ... best llik = -2085.824
## Model 5: llik = -2085.824 ... best llik = -2085.824
## Model 6: llik = -2086.317 ... best llik = -2085.824
## Model 7: llik = -2085.726 ... best llik = -2085.726
## Model 8: llik = -2085.984 ... best llik = -2085.726
## Model 9: llik = -2086.516 ... best llik = -2085.726
## Model 10: llik = -2086.265 ... best llik = -2085.726
## Conditional item response (column) probabilities,
##  by outcome variable, for each class (row)
##
## $Account.Balance
##            Pr(1)  Pr(2)  Pr(3)  Pr(4)
## class 1:  0.3115 0.4295 0.0729 0.1860
## class 2:  0.3624 0.4674 0.1702 0.0000
## class 3:  0.8750 0.0833 0.0417 0.0000
## class 4:  0.0000 0.0877 0.0055 0.9068
##
## $Type.of.apartment
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.1748 0.8252 0.0000
## class 2:  0.1740 0.4841 0.3418
## class 3:  0.4313 0.5107 0.0580
## class 4:  0.1269 0.7889 0.0842
##
## $Occupation
##            Pr(1)  Pr(2)  Pr(3)  Pr(4)
## class 1:  0.0000 0.3907 0.6093 0.0000
## class 2:  0.0779 0.0274 0.4727 0.4220
## class 3:  0.0290 0.1961 0.7749 0.0000
## class 4:  0.0052 0.1261 0.6884 0.1803
##
## $Guarantors
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.9042 0.0000 0.0958
## class 2:  0.9692 0.0308 0.0000
## class 3:  0.6424 0.2489 0.1087
## class 4:  0.9513 0.0402 0.0086
##
## Estimated class population shares
##  0.3757 0.1862 0.0929 0.3452
##
## Predicted class memberships (by modal posterior prob.)
##  0.4177 0.1377 0.068 0.3766
##
## =========================================================
## Fit for 4 latent classes:
## =========================================================
## number of observations: 632
## number of estimated parameters: 43
## residual degrees of freedom: 100
## maximum log-likelihood: -2085.726
##
## AIC(4): 4257.452
## BIC(4): 4448.754
## G^2(4): 78.2317 (Likelihood ratio/deviance statistic)
## X^2(4): 72.51695 (Chi-square goodness of fit)
##
## Model 1: llik = -2080.91 ... best llik = -2080.91
## Model 2: llik = -2080.208 ... best llik = -2080.208
## Model 3: llik = -2077.679 ... best llik = -2077.679
## Model 4: llik = -2082.61 ... best llik = -2077.679
## Model 5: llik = -2081.635 ... best llik = -2077.679
## Model 6: llik = -2081.731 ... best llik = -2077.679
## Model 7: llik = -2080.333 ... best llik = -2077.679
## Model 8: llik = -2081.732 ... best llik = -2077.679
## Model 9: llik = -2080.372 ... best llik = -2077.679
## Model 10: llik = -2080.675 ... best llik = -2077.679
## Conditional item response (column) probabilities,
##  by outcome variable, for each class (row)
##
## $Account.Balance
##            Pr(1)  Pr(2)  Pr(3)  Pr(4)
## class 1:  0.0000 0.0000 0.0200 0.9800
## class 2:  0.5699 0.2886 0.1414 0.0000
## class 3:  0.3355 0.4751 0.0693 0.1201
## class 4:  0.0000 0.8318 0.1682 0.0000
## class 4:  0.0000 0.8318 0.1682 0.0000
## class 5:  0.8360 0.1149 0.0491 0.0000
##
## $Type.of.apartment
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.1351 0.7859 0.0790
## class 2:  0.3021 0.2903 0.4076
## class 3:  0.1596 0.8202 0.0202
## class 4:  0.0000 0.7228 0.2772
## class 5:  0.4923 0.5077 0.0000
##
## $Occupation
##            Pr(1)  Pr(2)  Pr(3)  Pr(4)
## class 1:  0.0049 0.1527 0.6762 0.1663
## class 2:  0.0000 0.0082 0.5970 0.3947
## class 3:  0.0000 0.3434 0.6566 0.0000
## class 4:  0.2500 0.0000 0.0000 0.7500
## class 5:  0.0777 0.2195 0.7028 0.0000
##
## $Guarantors
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.9517 0.0377 0.0106
## class 2:  0.9913 0.0000 0.0087
## class 3:  0.9086 0.0000 0.0914
## class 4:  0.9039 0.0961 0.0000
## class 5:  0.5249 0.3927 0.0824
##
## Estimated class population shares
##  0.3383 0.1204 0.4277 0.0494 0.0643
##
## Predicted class memberships (by modal posterior prob.)
##  0.3766 0.0759 0.4525 0.0617 0.0332
##
## =========================================================
## Fit for 5 latent classes:
## =========================================================
## number of observations: 632
## number of estimated parameters: 54
## residual degrees of freedom: 89
## maximum log-likelihood: -2077.679
##
## AIC(5): 4263.358
## BIC(5): 4503.598
## G^2(5): 62.13814 (Likelihood ratio/deviance statistic)
## X^2(5): 55.38656 (Chi-square goodness of fit)
##
```

```
LCA_best_model
```

```
## Conditional item response (column) probabilities,
##  by outcome variable, for each class (row)
##
## $Account.Balance
##            Pr(1)  Pr(2)  Pr(3)  Pr(4)
## class 1:  0.0000 0.1996 0.0792 0.7212
## class 2:  0.5155 0.3488 0.0245 0.1113
## class 3:  0.4660 0.4049 0.1290 0.0000
##
## $Type.of.apartment
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.1259 0.8005 0.0736
## class 2:  0.2345 0.7435 0.0221
## class 3:  0.2246 0.3830 0.3923
##
## $Occupation
##            Pr(1)  Pr(2)  Pr(3)  Pr(4)
## class 1:  0.0067 0.1686 0.6591 0.1656
## class 2:  0.0000 0.3448 0.6552 0.0000
## class 3:  0.1132 0.0000 0.4373 0.4495
##
## $Guarantors
##            Pr(1)  Pr(2)  Pr(3)
## class 1:  0.9652 0.0267 0.0081
## class 2:  0.8262 0.0576 0.1162
## class 3:  0.9445 0.0555 0.0000
##
## Estimated class population shares
##  0.4709 0.3893 0.1398
##
## Predicted class memberships (by modal posterior prob.)
##  0.4509 0.443 0.106
##
## =========================================================
## Fit for 3 latent classes:
## =========================================================
## number of observations: 632
## number of estimated parameters: 32
## residual degrees of freedom: 111
## maximum log-likelihood: -2092.881
##
## AIC(3): 4249.763
## BIC(3): 4392.127
## G^2(3): 92.54278 (Likelihood ratio/deviance statistic)
## X^2(3): 93.87396 (Chi-square goodness of fit)
##
```

3. Q:Perform Holdout Validation Training

```
results.4.train <- poLCA(f1,german.credit.train1, nclass=3, nrep= 100, tol=.001, verbose=F, graphs=T)
```

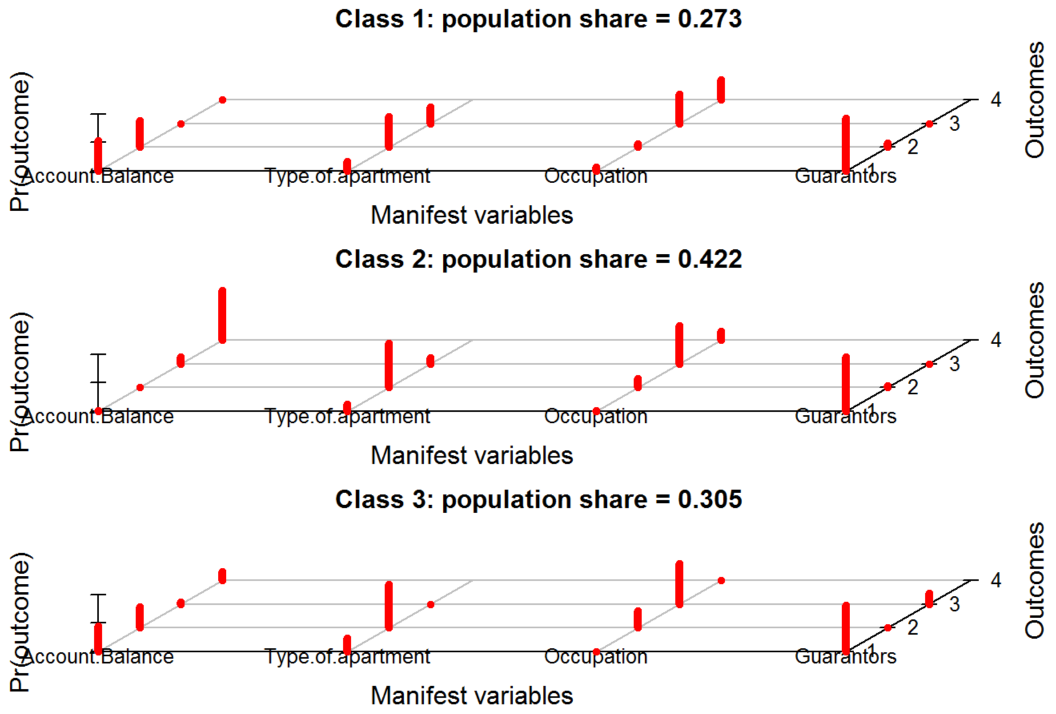


Class 1: population share = 0.143

Class 2: population share = 0.463

Class 3: population share = 0.395

```
german.credit.test1<-my.german.new[-german.credit.train_ind,]
results.4.test <- poLCA(f1,german.credit.test1,probs.start = results.4.train$probs, nclass=3,  nrep= 100, to
l=.001, verbose=F, graphs=T)
```



Class 1: population share = 0.273

Class 2: population share = 0.422

Class 3: population share = 0.305

4. Q: Provide implications/commentary on the goodness, interpretability, stability and adequacy of solutions. Looking at goodniss of fit of the two models: Interpretibility? stability? adequacy?

A: Class 1 has a favorable population for levels 1 and 2 of account balance. There also seems to be a favorable population for occupation levels 3 and 4 in class 1. Class 2 differentiates itself from the other classes by having a very favorable population for the fourth categtory of Account Balances. All the models seem to mimic one another for guarantors where all class have a very favorable population towards the first category of guarantors. All classes share a distinguishable population size for category 3 for occupation. The Holding set performs adequately to mimic the training class. The last time this simulation was run, the following changes in population shares were noted: Class 1 train: .143 to Class 1 test: .273 Class 2 train: .463 to Class 2 test: .422 Class 3 train: .395 to Class 3 test: .305

Chisq moves lower with our test data which shows a strong goodness of fit for our test.

```
c("Train Chi.sqr "= results.4.train$Chisq,
"Test Chi.sqr "= results.4.test$Chisq)
```

```
## Train Chi.sqr   Test Chi.sqr
##       93.70351       67.34272
```

5. Q: Comment on the similarity/differences between the clustering solutions you generated in Assignment 1 with the solution you generated using LCA.

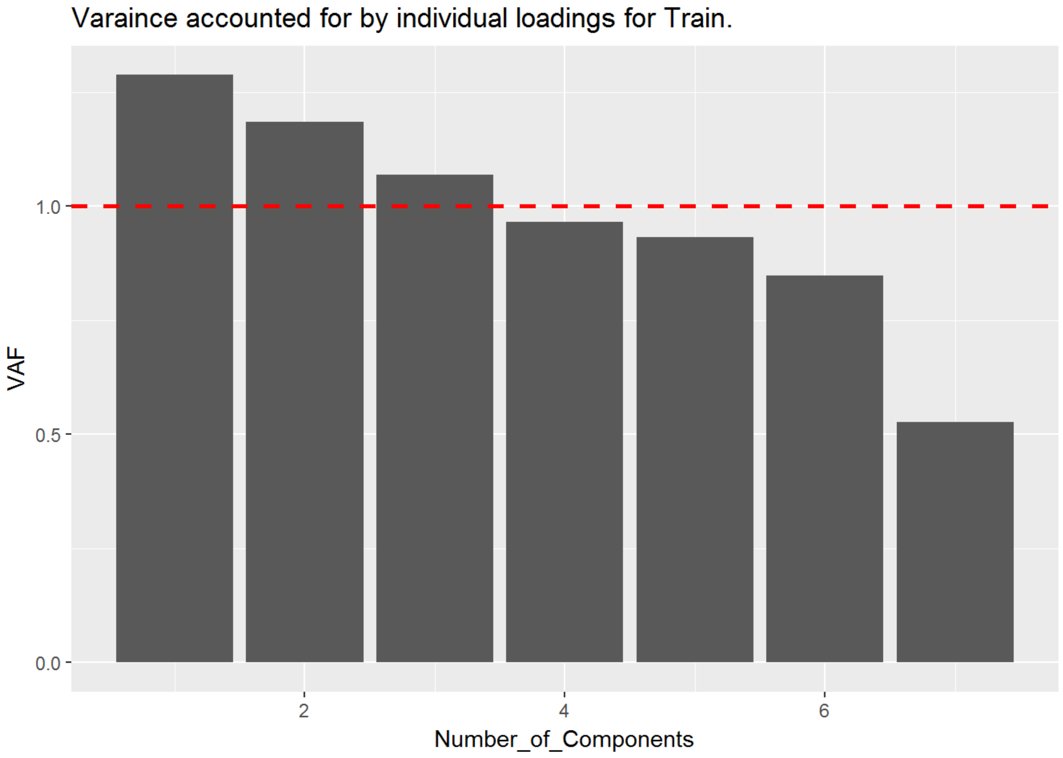
A: With the clustering analysis utilizing kmeans, 4 paramters were observed, and it was deemed most fitting to create 5 clusters. With the LCA analysis, 4 paramters were also observed, but it was deemed most fitting to have 3 latent classes decided upon by utilizing AIC.

Part 2 1) Q: Split the sample into two random samples sizes 70% and 30%. 2) Q: Perfomr prinicipal components of variables 1:7 from the GermanCredit Data on training sample.

```
data(GermanCredit)
GermanCredit <- data.frame(scale(GermanCredit[,1:7]))
german.credit.train_ind <- sample(1:(nrow(GermanCredit)),700)
german.credit_1to7.train <- GermanCredit[german.credit.train_ind,]
x = prcomp(german.credit_1to7.train, scale = T, retx = T, tol =.4)
```

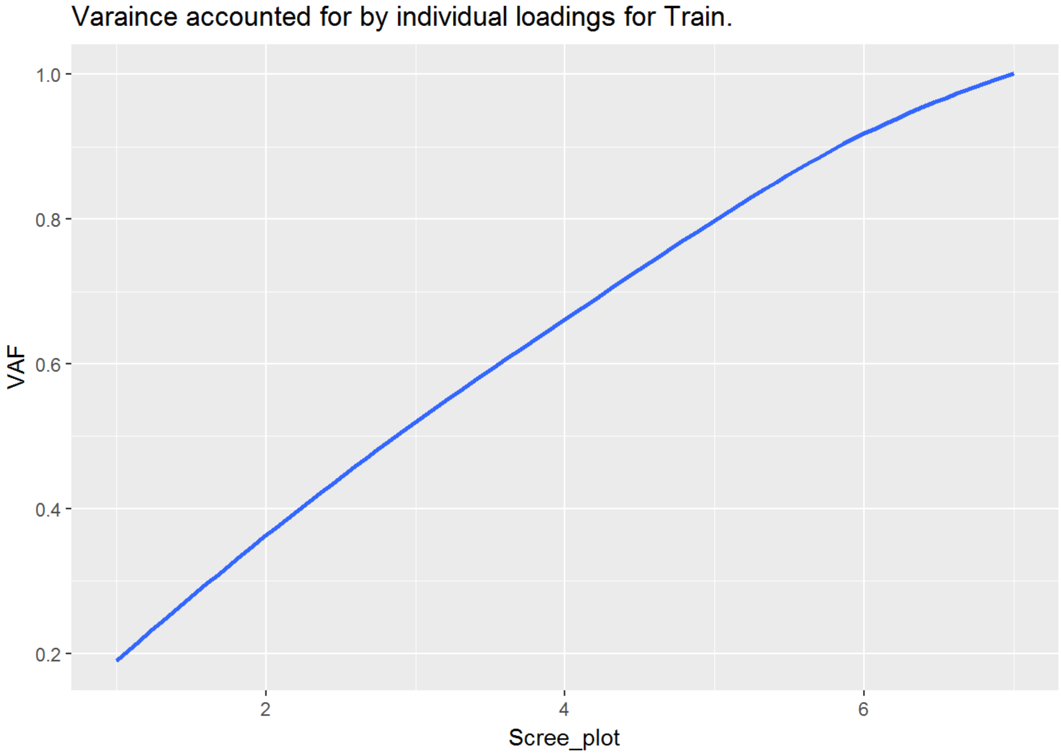
3. Generate Scree Plots and select number of components you would retain. From the Bar Chart below, the first three components should be utilized because they all share a VAF larger than 1.

```
cum.VAF.gm1 <- data.frame("Number_of_Components" = c(seq(1,7)), "VAF" = (c(x$sdev)))
ggplot(cum.VAF.gm1, aes(x=Number_of_Components, y=VAF, colours = VAF)) + geom_bar(stat = 'identity') + ggtit
le("Varaince accounted for by individual loadings for Train.") + geom_hline(yintercept = 1, linetype = 'dash
ed', size = 1, color = 'red')
```



```
cum.VAF.gm <- data.frame("Scree_plot" = c(seq(1,7)), "VAF" = cumsum(c(x$sdev)/sum(x$sdev)))
ggplot(cum.VAF.gm, aes(x=Scree_plot, y=VAF, colours = VAF)) + geom_smooth(se = F) + ggtitle("Varaince accoun
ted for by individual loadings for Train.")
```

```
## `geom_smooth()` using method = 'loess'
```

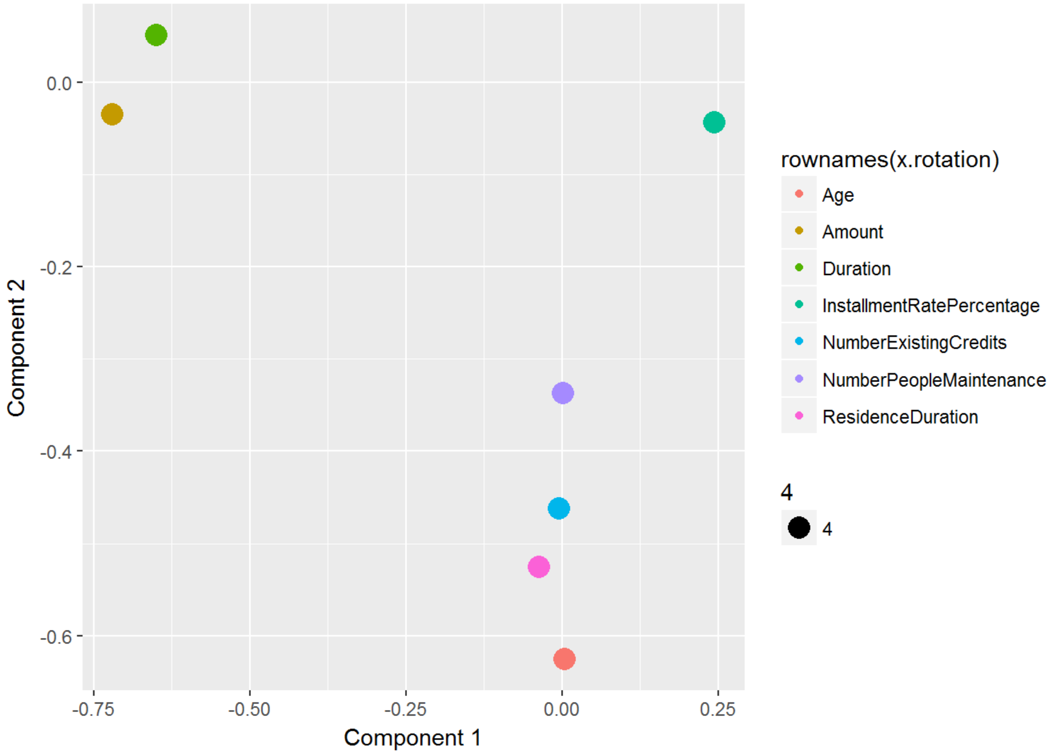


Q4) Plot component 1 loadings (x-axis) versus Component 2 loadings (y-axis). Use this plot to interpret and name the components.

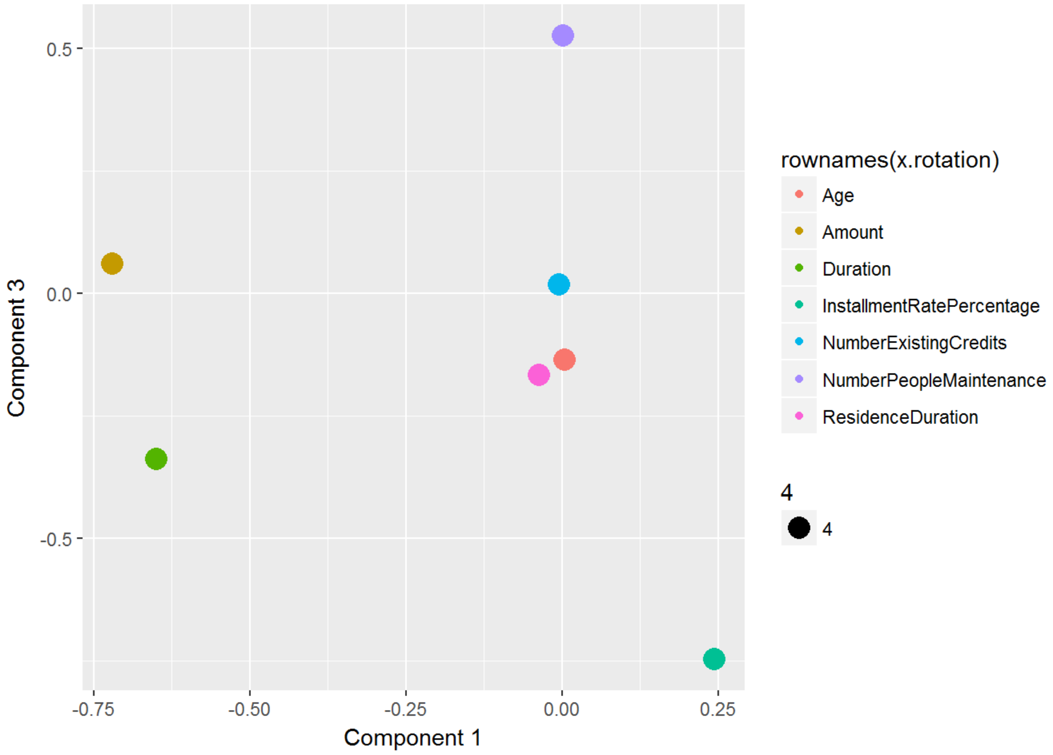
A: For the 1st component against the 2nd component, The 1st component has a very strong negative effect for Duration and Amount. The first component has a near 0 effect on 4 of the paramters. The second component has a negative effect on all paramters except for Duration.

For the 1st component against the 3rd component, the 1st component, again, has a very strong negative effect on Amount and Duration. The 1st component also has 4 paramters on it’s 0 axis, while the second component has 3 parameters above 0.

```
x.rotation<-data.frame(x$rotation)
ggplot(x.rotation, aes(x=x.rotation[,1], y= x.rotation[,2], colour = rownames(x.rotation), size =4)) + geom_
point() + xlab("Component 1") + ylab("Component 2")
```



```
ggplot(x.rotation, aes(x=x.rotation[,1], y= x.rotation[,3], colour = rownames(x.rotation), size =4)) + geom_
point()  + xlab("Component 1") + ylab("Component 3")
```



```
?prcomp
```

```
## starting httpd help server ... done
```

Q5) Show that eigenvectors are orthogonal

```
round(x$rotation %*% t(x$rotation),2)
```

```
##                           Duration Amount InstallmentRatePercentage ResidenceDuration Age
## Duration                         1      0                         0                 0   0
## Amount                           0      1                         0                 0   0
## InstallmentRatePercentage        0      0                         1                 0   0
## ResidenceDuration                0      0                         0                 1   0
## Age                              0      0                         0                 0   1
## NumberExistingCredits            0      0                         0                 0   0
## NumberPeopleMaintenance          0      0                         0                 0   0
##                           NumberExistingCredits NumberPeopleMaintenance
## Duration                                      0                       0
## Amount                                        0                       0
## InstallmentRatePercentage                     0                       0
## ResidenceDuration                             0                       0
## Age                                           0                       0
## NumberExistingCredits                         1                       0
## NumberPeopleMaintenance                       0                       1
```

Q6) Show that component scores are orthogonal

```
round(cov(x$x),2)
```

```
##      PC1 PC2  PC3  PC4  PC5  PC6  PC7
## PC1 1.66 0.0 0.00 0.00 0.00 0.00 0.00
## PC2 0.00 1.4 0.00 0.00 0.00 0.00 0.00
## PC3 0.00 0.0 1.14 0.00 0.00 0.00 0.00
## PC4 0.00 0.0 0.00 0.93 0.00 0.00 0.00
## PC5 0.00 0.0 0.00 0.00 0.87 0.00 0.00
## PC6 0.00 0.0 0.00 0.00 0.00 0.72 0.00
## PC7 0.00 0.0 0.00 0.00 0.00 0.00 0.28
```

7. Perform holdout validation of principal component solution

```
german.Credit_1to7.test <- data.frame(GermanCredit[-german.credit.train_ind,])
y=predict(x, newdata = german.Credit_1to7.test)
?tr
tr(round(cor(as.vector(german.credit_1to7.train),(x$x %*% t(x$rotation))),2))/7
```

```
## [1] 1
```

```
tr(round(cor((german.Credit_1to7.test),(y%*%t(x$rotation))),2)/7)
```

```
## [1] 1
```

8. Compute the Variance Account for (R^2) in the Holdout sample. We can see below hat the R^2 for both the training and holding are quite similar

```
tr(round(cor(as.vector(german.credit_1to7.train),(x$x[,1:3]%*% t(x$rotation)[1:3,])),2)/7)^2
```

```
## [1] 0.5841327
```

```
tr((round(cor((german.Credit_1to7.test),(y[,1:3]%*%t(x$rotation)[1:3,])),2))/7)^2
```

```
## [1] 0.5732653
```

```
?cor
```

9. Rotate the component loadings using varimax rotation.

```
x$rotation[,1:3]
```

```
##                                     PC1         PC2         PC3
## Duration                   -0.649179002  0.05083661 -0.33789436
## Amount                     -0.719513968 -0.03486720  0.06079512
## InstallmentRatePercentage   0.243781929 -0.04366199 -0.74745927
## ResidenceDuration          -0.037184373 -0.52545214 -0.16671620
## Age                         0.004451873 -0.62549386 -0.13575387
## NumberExistingCredits      -0.005449509 -0.46172196  0.01813294
## NumberPeopleMaintenance     0.002109508 -0.33728899  0.52619748
```

```
y=varimax(x$rotation[,1:3])
y$loadings
```

```
## 
## Loadings:
##                           PC1    PC2    PC3   
## Duration                  -0.701        -0.216
## Amount                    -0.695         0.198
## InstallmentRatePercentage  0.106 -0.184 -0.758
## ResidenceDuration                -0.548       
## Age                              -0.640       
## NumberExistingCredits            -0.450  0.104
## NumberPeopleMaintenance    0.109 -0.232  0.570
## 
##                  PC1   PC2   PC3
## SS loadings    1.000 1.000 1.000
## Proportion Var 0.143 0.143 0.143
## Cumulative Var 0.143 0.286 0.429
```

10. Plot rotated loadings(1) versus rotated loadings (2) and (3). Do you think Principal Components reduced this data a lot? Do you like the solution?

A: In the visualization with the 1st loading against the 2nd, Duration barely moved against component 2 to have a negative. There was a masisive change with the Verimax function where NumberPeopleMaintenance moved from a 0 value for component 1 to a value of 10%.

In the visualizaion with the 2nd loading agast the 3rd loading, Components 1 absolute value for the parameters seemed to decrease for Amount, Duration, and InstallmentRatePercentage. It seemed that components 3's values didn't change much whether if the Verimax function was used or not. I personally don’t think I would use the Verimax function very often, but I might in a time of need.

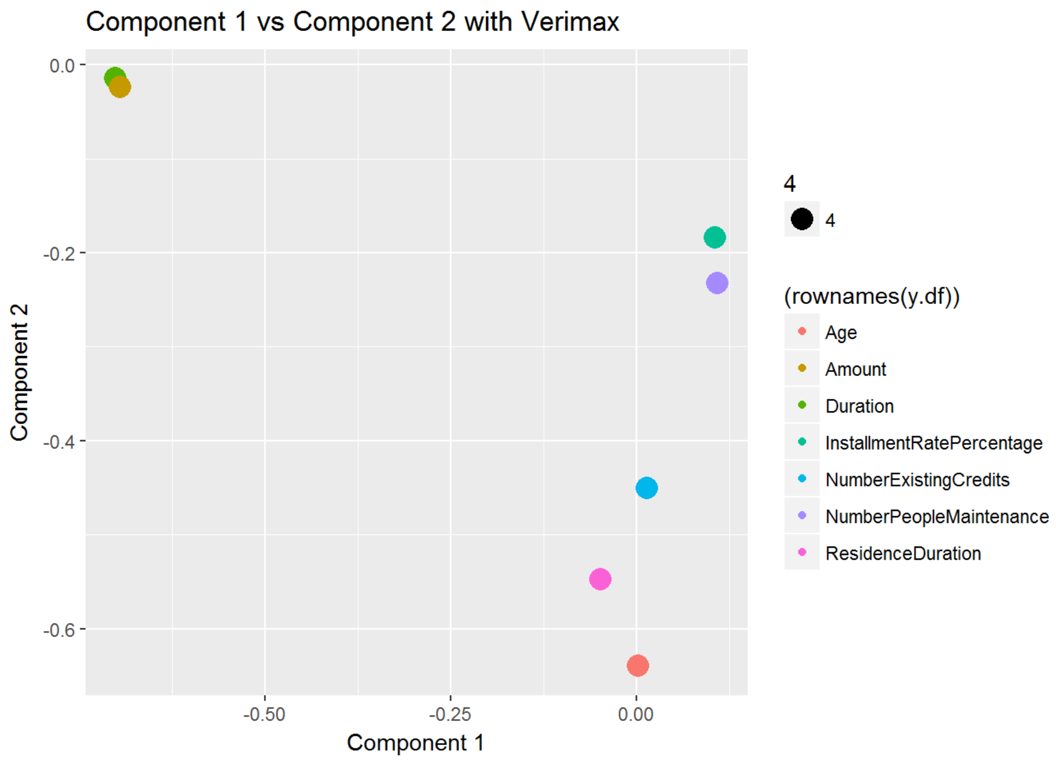
```
(y.df<-data.frame(matrix(y$loadings, byrow = F, ncol = 3),row.names = rownames(y$loadings)))
```

```
##                                      X1          X2          X3
## Duration                   -0.700948132 -0.01451518 -0.21598926
## Amount                     -0.694966632 -0.02351741  0.19768771
## InstallmentRatePercentage  0.106097712 -0.18367355 -0.75831304
## ResidenceDuration          -0.047881535 -0.54751042 -0.05671526
## Age                         0.002215546 -0.63986859 -0.01596185
## NumberExistingCredits       0.014434957 -0.45001014  0.10405849
## NumberPeopleMaintenance     0.109226613 -0.23193031  0.57002630
```

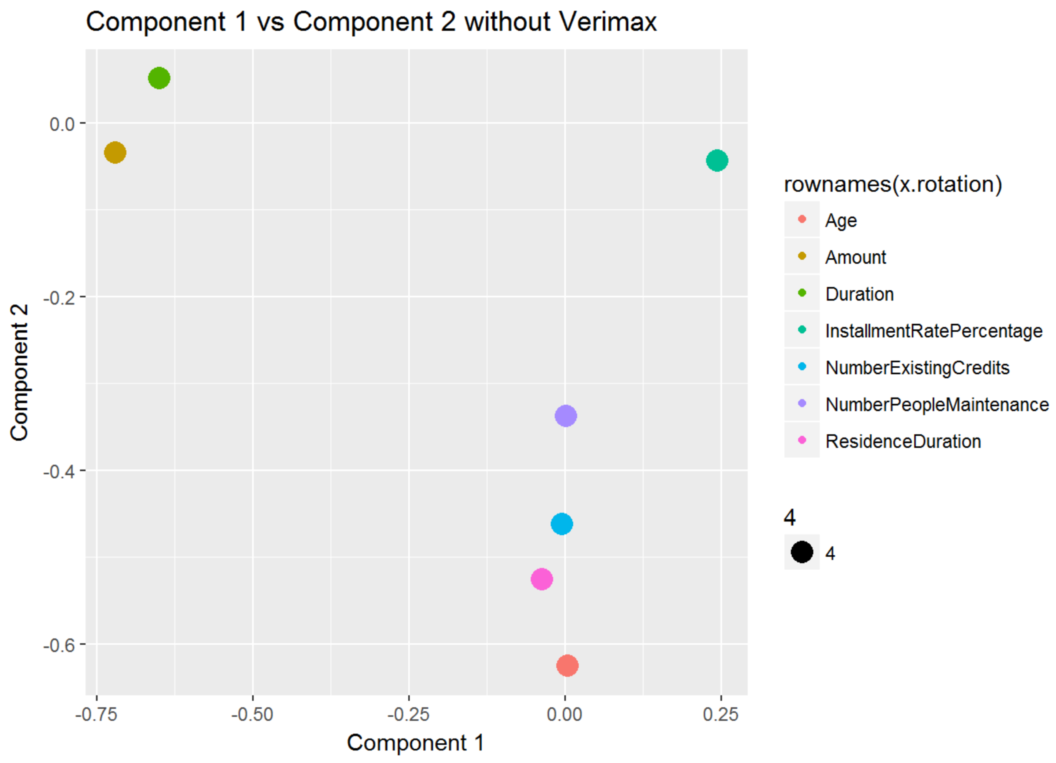
```
x$rotation[,1:3]
```

```
##                                     PC1         PC2         PC3
## Duration                   -0.649179002  0.05083661 -0.33789436
## Amount                     -0.719513968 -0.03486720  0.06079512
## InstallmentRatePercentage  0.243781929 -0.04366199 -0.74745927
## ResidenceDuration          -0.037184373 -0.52545214 -0.16671620
## Age                         0.004451873 -0.62549386 -0.13575387
## NumberExistingCredits      -0.005449509 -0.46172196  0.01813294
## NumberPeopleMaintenance     0.002109508 -0.33728899  0.52619748
```

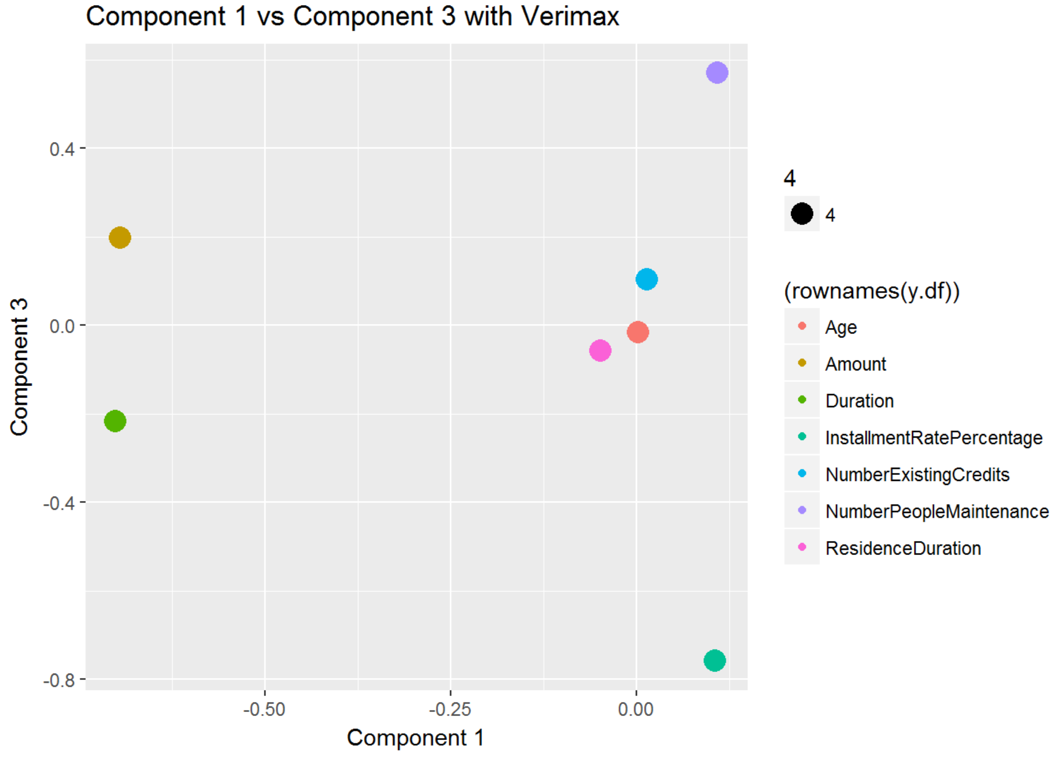
```
ggplot(y.df,aes(x=(y.df[,1]),y=y.df[,2], colour=(rownames(y.df)), size = 4)) + geom_point() + xlab('Componen
t 1') + ylab('Component 2') + ggtitle("Component 1 vs Component 2 with Verimax")
```



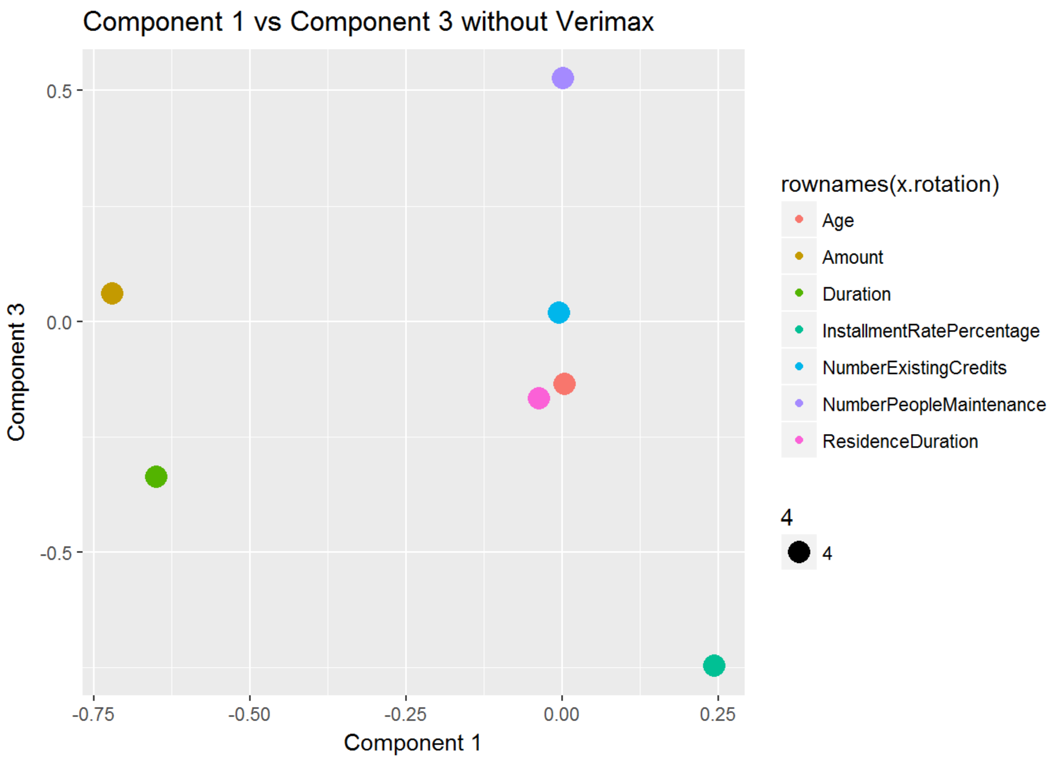
```
ggplot(x.rotation, aes(x=x.rotation[,1], y= x.rotation[,2], colour = rownames(x.rotation), size =4)) + geom_
point() + xlab("Component 1") + ylab("Component 2") + ggtitle("Component 1 vs Component 2 without Verimax")
```



```
ggplot(y.df,aes(x=(y.df[,1]),y=y.df[,3], colour=(rownames(y.df)), size = 4)) + geom_point() + xlab('Component 1') + ylab('Component 3') + ggtitle("Component 1 vs Component 3 with Verimax")
```



```
ggplot(x.rotation, aes(x=x.rotation[,1], y= x.rotation[,3], colour = rownames(x.rotation), size =4)) + geom_point()  + xlab("Component 1") + ylab("Component 3")+ ggtitle("Component 1 vs Component 3 without Verimax")
```


Component 1 vs Component 3 without Verimax
0.5
0.0
-0.5
Component 3
-0.75
-0.50
-0.25
0.00
0.25
Component 1
rownames(x.rotation)
Age
Amount
Duration
InstallmentRatePercentage
NumberExistingCredits
NumberPeopleMaintenance
ResidenceDuration
4
4