ADVANCED STATISTICS ASSIGNMENT

GROUP 8

SatyaCharan Yadati

Subhakant

Satish S

Jestin Jose

Gowri R Vardhan

Ruthra

Contents

[Problem 1:Cereal Data Factor Analysis 2](#_Toc518232267)

[1 ) Exploratory data analysis 2](#_Toc518232268)

[2) Assumptions 3](#_Toc518232269)

[Bartlett’s Test 3](#_Toc518232270)

[KMO Test for measuring sampling accuracy 4](#_Toc518232271)

[3) Conducting PCA to determine number of factors for Factor Analysis 8](#_Toc518232272)

[Scree Plot to determine the number of factors to be selected 8](#_Toc518232273)

[4) Factor Analysis 9](#_Toc518232274)

[Exploring the model with 8 factors, without rotation 10](#_Toc518232275)

[Exploring the model with 8 factors, with rotation 12](#_Toc518232276)

[Removing variables with high Uniqueness values and running the factor analysis again 13](#_Toc518232277)

[Adding Factor Names 16](#_Toc518232278)

[Reliability of Factor 19](#_Toc518232279)

[Average Factor Scores grouped by the cereal. 20](#_Toc518232280)

[Problem 2 – LeslieSalt Company 21](#_Toc518232281)

[a) Question 1 21](#_Toc518232282)

[b) Question 3: 24](#_Toc518232283)

[c) Question 2: 26](#_Toc518232284)

[Problem 3 – All Greens Franchise 26](#_Toc518232285)

[5. Conclusion 31](#_Toc518232286)

# Problem 1:Cereal Data Factor Analysis

# 1 ) Exploratory data analysis

setwd("E:/Great Lakes/PGBABI/Course Material/Advanced Statistics/GA")  
cerealData<-read.csv("cereal.csv")  
summary(cerealData)

## Cereals Filling Natural Fibre   
## CornFlakes :27 Min. :1.000 Min. :1.000 Min. :1.000   
## Weetabix :27 1st Qu.:3.000 1st Qu.:3.000 1st Qu.:3.000   
## Vitabrit :25 Median :4.000 Median :4.000 Median :4.000   
## NutriGrain :24 Mean :3.881 Mean :3.783 Mean :3.528   
## SpecialK :23 3rd Qu.:4.500 3rd Qu.:4.000 3rd Qu.:4.000   
## RiceBubbles:21 Max. :5.000 Max. :5.000 Max. :5.000   
## (Other) :88   
## Sweet Easy Salt Satisfying   
## Min. :1.000 Min. :1.000 Min. :1.000 Min. :2.000   
## 1st Qu.:2.000 1st Qu.:4.000 1st Qu.:1.000 1st Qu.:3.000   
## Median :2.000 Median :5.000 Median :2.000 Median :4.000   
## Mean :2.506 Mean :4.532 Mean :1.991 Mean :4.004   
## 3rd Qu.:3.000 3rd Qu.:5.000 3rd Qu.:3.000 3rd Qu.:5.000   
## Max. :5.000 Max. :6.000 Max. :4.000 Max. :6.000   
##   
## Energy Fun Kids Soggy   
## Min. :1.000 Min. :1.000 Min. :1.000 Min. :1.000   
## 1st Qu.:3.000 1st Qu.:2.000 1st Qu.:3.000 1st Qu.:1.000   
## Median :4.000 Median :2.000 Median :4.000 Median :2.000   
## Mean :3.643 Mean :2.617 Mean :3.843 Mean :2.255   
## 3rd Qu.:4.000 3rd Qu.:3.000 3rd Qu.:5.000 3rd Qu.:3.000   
## Max. :5.000 Max. :5.000 Max. :6.000 Max. :5.000   
##   
## Economical Health Family Calories   
## Min. :1.000 Min. :1.000 Min. :1.000 Min. :1.000   
## 1st Qu.:3.000 1st Qu.:3.000 1st Qu.:3.000 1st Qu.:2.000   
## Median :3.000 Median :4.000 Median :4.000 Median :3.000   
## Mean :3.217 Mean :3.809 Mean :3.877 Mean :2.702   
## 3rd Qu.:4.000 3rd Qu.:4.000 3rd Qu.:5.000 3rd Qu.:3.000   
## Max. :5.000 Max. :5.000 Max. :6.000 Max. :5.000   
##   
## Plain Crisp Regular Sugar   
## Min. :1.000 Min. :1.000 Min. :1.000 Min. :1.000   
## 1st Qu.:1.000 1st Qu.:2.000 1st Qu.:2.000 1st Qu.:1.000   
## Median :2.000 Median :3.000 Median :3.000 Median :2.000   
## Mean :2.268 Mean :3.204 Mean :3.072 Mean :2.145   
## 3rd Qu.:3.000 3rd Qu.:4.000 3rd Qu.:4.000 3rd Qu.:3.000   
## Max. :5.000 Max. :6.000 Max. :5.000 Max. :5.000   
##   
## Fruit Process Quality Treat   
## Min. :1.000 Min. :1.000 Min. :1.000 Min. :1.00   
## 1st Qu.:1.000 1st Qu.:2.000 1st Qu.:3.000 1st Qu.:2.00   
## Median :1.000 Median :3.000 Median :4.000 Median :3.00   
## Mean :1.694 Mean :2.936 Mean :3.694 Mean :2.63   
## 3rd Qu.:3.000 3rd Qu.:4.000 3rd Qu.:4.000 3rd Qu.:3.00   
## Max. :5.000 Max. :6.000 Max. :5.000 Max. :6.00   
##   
## Boring Nutritious   
## Min. :1.00 Min. :1.000   
## 1st Qu.:1.00 1st Qu.:3.000   
## Median :2.00 Median :4.000   
## Mean :1.83 Mean :3.664   
## 3rd Qu.:2.00 3rd Qu.:4.000   
## Max. :5.00 Max. :5.000   
##

Here we see that there are few variables having max as 6, whereas the scale used is from 1 to 5. We need to replace these with 5.

cerealRefinedData<-cerealData[,-1]  
  
cerealRefinedData[cerealRefinedData==6] <- 5

# 2) Assumptions

We need to run few tests to check whether the used data set is sufficient and suitable for factor analysis. Here we would run two test :

### Bartlett’s Test

The Bartlett’s Test of Sphericity should be significant (p<.05) for factor analysis to be suitable.

Null Hypothesis: Variables are uncorrelated Alternative Hypothesis: Variables are correlated

library("psych")

## Warning: package 'psych' was built under R version 3.4.4

cerealMatrix <- cor(cerealRefinedData)  
cerealMatrix <- round(cerealMatrix, 2)  
cerealBartlett <- cortest.bartlett(cerealMatrix, n = nrow(cerealRefinedData))  
cerealBartlett

## $chisq  
## [1] 2878.65  
##   
## $p.value  
## [1] 0  
##   
## $df  
## [1] 300

Bartlett’s test has to be ‘significant’, telling us that the correlation matrix is not an identity matrix with the p-value <0.05. On running the test, we have obtained the p-value as 0.

Thus, based on the above results, we assume that the fair correlation exists between all the variables, so we can run the Factor Analysis on the given data set.

### KMO Test for measuring sampling accuracy

The KMO index, in particular, is recommended when the cases to variable ratio are less than 1:5. The KMO index ranges from 0 to 1, with 0.50 considered suitable for factor analysis.

cereal.KMO<-KMO(cerealRefinedData)  
cereal.KMO

## Kaiser-Meyer-Olkin factor adequacy  
## Call: KMO(r = cerealRefinedData)  
## Overall MSA = 0.85  
## MSA for each item =   
## Filling Natural Fibre Sweet Easy Salt   
## 0.89 0.90 0.88 0.78 0.83 0.82   
## Satisfying Energy Fun Kids Soggy Economical   
## 0.91 0.91 0.85 0.67 0.63 0.73   
## Health Family Calories Plain Crisp Regular   
## 0.92 0.73 0.86 0.82 0.83 0.87   
## Sugar Fruit Process Quality Treat Boring   
## 0.78 0.77 0.80 0.91 0.88 0.87   
## Nutritious   
## 0.92

Kaiser’s Criteria to determine the number of factors to be extracted

cerealCov<-cov(cerealRefinedData)  
cerealEV<-eigen(cerealCov)  
cerealEV

## eigen() decomposition  
## $values  
## [1] 6.2435725 4.3483221 3.0901775 1.8806804 1.3890966 1.0742392 0.9625560  
## [8] 0.8720933 0.7985735 0.7112658 0.6398052 0.6263464 0.5591410 0.5027648  
## [15] 0.4292656 0.4084484 0.3892509 0.3573892 0.2951947 0.2594110 0.2363838  
## [22] 0.2210170 0.1847529 0.1675338 0.1483178  
##   
## $vectors  
## [,1] [,2] [,3] [,4] [,5]  
## [1,] -0.24773514 0.06296020 0.046198687 -0.15982512 -0.01375283  
## [2,] -0.21741737 0.20029244 0.064807033 -0.11502032 -0.06190012  
## [3,] -0.24176579 0.22125944 0.184499412 -0.15104244 0.16588313  
## [4,] -0.12078931 -0.37249925 0.138662966 -0.25303393 -0.08162176  
## [5,] -0.10725876 -0.01031537 -0.086053265 -0.05886238 0.03317475  
## [6,] 0.03627618 -0.21046119 0.051157538 -0.24064276 0.14341206  
## [7,] -0.23206755 0.03201439 -0.056753568 -0.12219070 -0.01137831  
## [8,] -0.25053145 0.04132690 0.045963456 -0.14176011 0.02613975  
## [9,] -0.30129157 -0.25636558 -0.148871930 0.02418190 -0.20949433  
## [10,] -0.15299257 -0.11608608 -0.531685592 -0.08965353 -0.05039549  
## [11,] 0.10499117 0.14567637 -0.179269013 -0.60636428 -0.38445432  
## [12,] -0.04720862 0.16760719 -0.378957974 0.02690493 0.23323651  
## [13,] -0.23019585 0.22227977 0.064394933 -0.06788688 0.05636293  
## [14,] -0.17649102 -0.06277940 -0.443570835 -0.04761680 -0.11335620  
## [15,] 0.01302765 -0.29133931 0.096824696 -0.20902826 0.09342435  
## [16,] 0.19383453 0.15862573 -0.195290907 -0.27489209 0.26439322  
## [17,] -0.23144321 -0.23732473 -0.141415488 0.26137856 0.38578841  
## [18,] -0.26051079 0.18929115 0.152393487 -0.09160066 0.39490668  
## [19,] 0.02816414 -0.37869375 0.137501519 -0.23513157 0.14376317  
## [20,] -0.19030949 -0.07322025 0.354591165 -0.04024961 -0.27155795  
## [21,] 0.12173653 -0.21551941 -0.022030544 -0.27892496 0.40887139  
## [22,] -0.24462619 0.15465317 -0.008606573 -0.02597424 0.03045139  
## [23,] -0.33524035 -0.27202030 -0.022321767 0.09230512 -0.04315707  
## [24,] 0.18560835 0.07692412 0.027645048 -0.20085613 0.15564456  
## [25,] -0.24416700 0.19485411 0.079639874 -0.11778803 0.05140914  
## [,6] [,7] [,8] [,9] [,10]  
## [1,] 0.137969464 -0.152470920 -0.14813975 -0.141121503 -0.31257204  
## [2,] 0.066131464 -0.087590845 -0.10891012 0.165605994 -0.02961265  
## [3,] -0.043401706 -0.135422260 0.09772664 -0.005656813 -0.06036362  
## [4,] 0.334880709 0.016696692 0.03099932 -0.119058565 0.15955549  
## [5,] 0.113569730 0.007469636 0.13913080 0.186180212 -0.16499664  
## [6,] 0.175078807 0.047576889 -0.12582500 -0.032952612 0.02313316  
## [7,] 0.143562029 -0.019449165 -0.11434591 0.055566882 -0.13773674  
## [8,] 0.001358956 -0.052142114 -0.11256262 0.060841982 -0.36092517  
## [9,] -0.357088129 0.467535837 0.23535545 -0.240994361 -0.27857921  
## [10,] 0.065864540 -0.344420784 0.09315269 -0.137538841 0.05892209  
## [11,] -0.100972513 0.252381882 0.08194042 0.235164380 0.13033459  
## [12,] 0.251441047 0.513701942 -0.38518858 -0.191895546 0.22713231  
## [13,] 0.005261591 0.041928716 -0.09883693 0.121914195 0.06374605  
## [14,] -0.060553433 -0.405896550 0.14117497 -0.057773021 0.12879328  
## [15,] 0.218856935 -0.045553236 -0.20982026 0.057218277 -0.22017224  
## [16,] 0.138471233 0.170051912 0.38198104 0.199567122 -0.05336515  
## [17,] 0.167404188 0.152938909 0.27588048 0.488385350 -0.11689024  
## [18,] -0.088548789 -0.012228797 0.37321484 -0.411012834 0.29510665  
## [19,] 0.114246280 -0.006801957 0.02820944 -0.192028788 0.23291974  
## [20,] 0.007590793 0.102555350 0.26140434 0.175296586 0.23352792  
## [21,] -0.611886157 -0.133126043 -0.21906272 0.251860011 0.09694340  
## [22,] -0.123303758 0.011950311 -0.27071119 0.062677157 0.18663302  
## [23,] -0.256555391 0.151847034 -0.20881665 0.025389624 0.08526927  
## [24,] -0.133968087 0.081939044 0.07480437 -0.346330808 -0.45845597  
## [25,] -0.003610817 0.031637857 -0.03097393 0.049835746 0.04972495  
## [,11] [,12] [,13] [,14] [,15]  
## [1,] -0.0640113464 -0.003617388 0.016619455 -0.06921233 0.186021083  
## [2,] -0.0306648030 0.075833608 -0.018768236 0.03888964 -0.301938289  
## [3,] 0.0902615068 -0.017643994 -0.068021300 0.09708125 -0.070386614  
## [4,] 0.0803802295 -0.172471578 -0.200779661 -0.10416694 0.247741082  
## [5,] -0.0679369906 0.021080511 -0.641728067 0.12556652 -0.420368540  
## [6,] 0.1633666471 0.137729656 -0.167727489 -0.08454161 -0.064724104  
## [7,] -0.1578958334 -0.063275512 -0.075042388 -0.15661175 0.123397634  
## [8,] -0.0821626734 -0.280003662 0.136030760 -0.09282156 0.358299567  
## [9,] -0.2266171793 -0.168597738 -0.091068766 0.19681748 -0.055831449  
## [10,] -0.0002423504 0.078570605 0.008015605 -0.10153004 -0.166073003  
## [11,] 0.3379730430 -0.148672259 0.215378278 0.13642460 -0.029621141  
## [12,] -0.1753692231 -0.109886143 0.058127465 -0.24015329 -0.134103369  
## [13,] 0.0297735294 0.049281211 -0.105564492 -0.02967702 0.095016687  
## [14,] -0.0428869479 0.032155005 0.135242936 -0.12326140 0.016741482  
## [15,] -0.3012583702 0.181857197 0.491164566 0.37241579 -0.345781574  
## [16,] -0.3408845504 0.429116606 -0.075644354 0.07086781 0.380966828  
## [17,] 0.3705167943 -0.186996426 0.244346741 -0.08904176 -0.050732155  
## [18,] -0.0208675055 -0.061643445 0.199423544 0.18595663 -0.051286874  
## [19,] 0.1113693597 -0.075299593 -0.114213435 0.09593544 -0.003966305  
## [20,] -0.3372810726 0.128259482 0.166025006 -0.56822330 -0.223700804  
## [21,] -0.2260646732 -0.193951561 -0.091846761 -0.19046780 -0.027488542  
## [22,] 0.0740442014 0.035653954 -0.030716058 0.16826185 -0.040841998  
## [23,] 0.2295484921 0.652419686 0.003443193 0.03149346 0.206417218  
## [24,] 0.3463922924 0.209523432 0.066029679 -0.43805556 -0.233409618  
## [25,] 0.1349085029 0.059817060 -0.044612444 -0.01340296 -0.071089815  
## [,16] [,17] [,18] [,19] [,20]  
## [1,] 0.1944017193 0.193241918 0.111037834 -0.19124175 0.084961403  
## [2,] -0.1025172742 -0.205642391 0.014464428 -0.41763369 -0.466300600  
## [3,] 0.0002971134 -0.220991169 0.044210152 -0.46960531 0.380718813  
## [4,] 0.1387898144 -0.334777274 -0.431433756 0.01207451 0.213668698  
## [5,] 0.3329064612 0.238636422 -0.102342156 0.17146229 -0.043657570  
## [6,] -0.4346253919 0.111325448 0.561705307 0.11602044 0.216795983  
## [7,] 0.0994585433 0.252025652 0.179040562 0.21393394 0.046176176  
## [8,] -0.0227281042 -0.025903688 0.025338558 0.17840016 -0.378456624  
## [9,] -0.2702774424 -0.071810646 0.034173107 -0.09969578 0.078128930  
## [10,] -0.0096480436 -0.517586259 0.183182455 0.23653754 -0.159554457  
## [11,] 0.2077002931 0.083669638 0.092480560 0.02496881 0.012965027  
## [12,] 0.1290035669 0.004880566 -0.029311992 -0.18980935 0.026533832  
## [13,] -0.1315896416 -0.031011261 -0.040727453 0.05922091 0.082729365  
## [14,] -0.1556793565 0.473489503 -0.221317003 -0.24498629 0.202844191  
## [15,] 0.0413634768 -0.007169259 -0.138463341 0.10594940 0.160703674  
## [16,] -0.1524091849 -0.057173090 -0.119625487 -0.05495428 -0.077242488  
## [17,] -0.0214375529 0.029212063 -0.024735265 -0.04202036 0.005786718  
## [18,] 0.2609408686 0.096420355 0.127608409 0.23946112 -0.073349198  
## [19,] -0.1767343766 0.259377125 -0.021808469 -0.25469667 -0.480973601  
## [20,] -0.0292137661 0.046355674 0.021338851 0.05898226 -0.024480848  
## [21,] 0.1117488709 -0.086949774 -0.006690743 -0.02931758 0.065731030  
## [22,] -0.3844524247 0.119387717 -0.446479525 0.31555520 -0.060002502  
## [23,] 0.3284538133 0.003099086 0.053770293 -0.05325106 -0.073584365  
## [24,] -0.0375067637 0.028078626 -0.302298946 0.05943856 -0.056493695  
## [25,] -0.2294927684 -0.123243844 -0.030632390 0.18250629 0.156155679  
## [,21] [,22] [,23] [,24] [,25]  
## [1,] 0.452929096 0.078670524 0.515002509 -0.23920961 -0.127010232  
## [2,] 0.098140970 0.340911622 -0.141886491 0.28734754 -0.256914404  
## [3,] -0.148421437 -0.265086533 -0.304956950 -0.34801628 0.156931206  
## [4,] 0.003776711 0.230992394 -0.080216134 0.12764628 -0.124903786  
## [5,] -0.197389252 0.046910378 0.086529436 -0.11272734 0.044247703  
## [6,] -0.121281748 0.357242680 -0.060016353 -0.09881732 -0.069539621  
## [7,] 0.311430997 -0.334708751 -0.578497597 0.30394130 -0.083501194  
## [8,] -0.550687155 0.070324002 -0.031379746 -0.17705846 0.072118803  
## [9,] 0.090455412 0.031964678 0.009102173 0.07107545 0.001175925  
## [10,] 0.135967875 -0.145864612 0.050034178 -0.15655597 0.148206074  
## [11,] 0.035279544 0.002694497 0.007220608 -0.05941260 0.026743177  
## [12,] -0.145486812 -0.029001191 0.030241099 -0.08910536 0.008221153  
## [13,] 0.096243546 0.134479629 0.200620164 0.38323128 0.765552579  
## [14,] -0.259723656 0.135590576 -0.013141145 0.15080627 -0.028204406  
## [15,] -0.076394953 -0.039780456 0.009734834 0.07658103 0.112760091  
## [16,] 0.049243206 -0.009724248 0.005093774 -0.05528508 -0.073229624  
## [17,] 0.155395908 0.020879341 0.036345021 -0.02511133 -0.023150272  
## [18,] 0.008700723 0.234251215 -0.038583537 0.07768912 -0.073187481  
## [19,] 0.048393414 -0.418904057 0.089491598 -0.02208856 0.184569786  
## [20,] -0.007428240 -0.037311885 0.059414322 -0.21512987 0.060527644  
## [21,] 0.078075204 0.039054448 0.066289762 0.06342811 -0.069432638  
## [22,] 0.257338504 0.075381431 -0.140081953 -0.43326762 -0.075298589  
## [23,] -0.130438657 -0.010144402 -0.061735514 -0.01361835 -0.015397296  
## [24,] 0.040193421 0.012667243 -0.096277983 0.06506036 0.060931060  
## [25,] -0.240068609 -0.454111233 0.415851602 0.32778157 -0.415827602

Eigenvalues of the correlation matrix, or the covariance matrix of the active variables play an important role in computing the principal components. In addition to determining the factor coordinates of the variables and cases, they give a fairly good idea about the variance that can be explained by the given number of factors. This information can further be utilized for determining the order by which you can afford to reduce the dimensions of the original space of variables or cases, without losing much information. Based on the eigenvalues, many criteria can be used to decide the ideal number of factors in a given situation. Since the sum of eigenvalues is equal to the number of ‘active’ variables, the average of the eigenvalues is 1,and the general approach is to first start with the eigenvalues that are greater than 1.

# 3) Conducting PCA to determine number of factors for Factor Analysis

pcacerealRefinedData = princomp(cerealRefinedData,cor=TRUE,scores = FALSE)   
summary(pcacerealRefinedData)

## Importance of components:  
## Comp.1 Comp.2 Comp.3 Comp.4  
## Standard deviation 2.5515645 1.9473508 1.57931249 1.29699429  
## Proportion of Variance 0.2604193 0.1516870 0.09976912 0.06728777  
## Cumulative Proportion 0.2604193 0.4121063 0.51187538 0.57916315  
## Comp.5 Comp.6 Comp.7 Comp.8  
## Standard deviation 1.04196617 0.97215571 0.92371682 0.88941254  
## Proportion of Variance 0.04342774 0.03780347 0.03413011 0.03164219  
## Cumulative Proportion 0.62259089 0.66039436 0.69452447 0.72616666  
## Comp.9 Comp.10 Comp.11 Comp.12  
## Standard deviation 0.85594266 0.83528812 0.80508009 0.74210795  
## Proportion of Variance 0.02930551 0.02790825 0.02592616 0.02202897  
## Cumulative Proportion 0.75547217 0.78338042 0.80930658 0.83133555  
## Comp.13 Comp.14 Comp.15 Comp.16  
## Standard deviation 0.72900837 0.69819277 0.6456121 0.62203549  
## Proportion of Variance 0.02125813 0.01949893 0.0166726 0.01547713  
## Cumulative Proportion 0.85259368 0.87209260 0.8887652 0.90424233  
## Comp.17 Comp.18 Comp.19 Comp.20  
## Standard deviation 0.60340597 0.60072703 0.55329582 0.52496345  
## Proportion of Variance 0.01456395 0.01443492 0.01224545 0.01102346  
## Cumulative Proportion 0.91880628 0.93324119 0.94548664 0.95651011  
## Comp.21 Comp.22 Comp.23 Comp.24  
## Standard deviation 0.51267066 0.492791255 0.467311607 0.445682156  
## Proportion of Variance 0.01051325 0.009713729 0.008735206 0.007945303  
## Cumulative Proportion 0.96702336 0.976737086 0.985472292 0.993417595  
## Comp.25  
## Standard deviation 0.405660102  
## Proportion of Variance 0.006582405  
## Cumulative Proportion 1.000000000

### Scree Plot to determine the number of factors to be selected

library("factoextra")

## Warning: package 'factoextra' was built under R version 3.4.4

## Loading required package: ggplot2

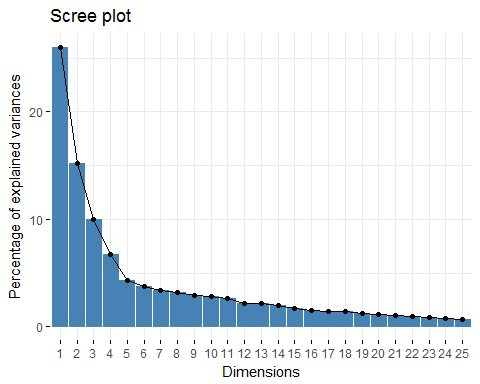
## Warning: package 'ggplot2' was built under R version 3.4.4

##   
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':  
##   
## %+%, alpha

## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ

fviz\_eig(pcacerealRefinedData,ncp = 25)



# 4) Factor Analysis

The factor loadings explain the weightage of the variables on the Factors.

fa1<-factanal(cerealRefinedData,factors=6)  
fa1

##   
## Call:  
## factanal(x = cerealRefinedData, factors = 6)  
##   
## Uniquenesses:  
## Filling Natural Fibre Sweet Easy Salt   
## 0.306 0.388 0.127 0.358 0.846 0.472   
## Satisfying Energy Fun Kids Soggy Economical   
## 0.337 0.434 0.528 0.215 0.754 0.632   
## Health Family Calories Plain Crisp Regular   
## 0.180 0.339 0.575 0.544 0.648 0.485   
## Sugar Fruit Process Quality Treat Boring   
## 0.211 0.560 0.759 0.362 0.389 0.667   
## Nutritious   
## 0.260   
##   
## Loadings:  
## Factor1 Factor2 Factor3 Factor4 Factor5 Factor6  
## Filling 0.653 0.149 0.201 0.438   
## Natural 0.727 -0.215 0.173   
## Fibre 0.884 -0.119 -0.261   
## Sweet 0.697 0.348 0.154   
## Easy 0.233 0.286   
## Salt 0.712   
## Satisfying 0.563 0.186 0.355 0.410 0.117   
## Energy 0.607 0.227 0.167 0.340   
## Fun 0.123 0.152 0.534 0.373   
## Kids 0.884   
## Soggy -0.461 0.109 0.120   
## Economical -0.233 -0.209 0.358 0.368   
## Health 0.821 -0.258 0.257   
## Family 0.122 0.798   
## Calories -0.151 0.596 0.117 0.157   
## Plain -0.110 -0.640 -0.137   
## Crisp 0.147 0.456 0.325   
## Regular 0.677 0.104 -0.174 -0.108   
## Sugar -0.174 0.840 0.179 -0.110   
## Fruit 0.326 0.157 0.442 -0.277 0.177   
## Process -0.204 0.376 -0.103 -0.215   
## Quality 0.652 -0.213 0.230 0.147 0.305   
## Treat 0.224 0.212 0.653 0.280 0.102   
## Boring -0.140 -0.503 -0.187 -0.112   
## Nutritious 0.828 -0.156 0.101 0.138   
##   
## Factor1 Factor2 Factor3 Factor4 Factor5 Factor6  
## SS loadings 5.041 2.586 2.429 2.314 0.745 0.505  
## Proportion Var 0.202 0.103 0.097 0.093 0.030 0.020  
## Cumulative Var 0.202 0.305 0.402 0.495 0.525 0.545  
##   
## Test of the hypothesis that 6 factors are sufficient.  
## The chi square statistic is 256.19 on 165 degrees of freedom.  
## The p-value is 6.89e-06

It can be observed that the significance level of the chi square fit statistic is very small. This indicates that the hypothesis of perfect model fit is rejected. The 6 factors are able to explain about 54% of the data.

### Exploring the model with 8 factors, without rotation

fa2<-factanal(cerealRefinedData,factors=8,rotation = "none")  
fa2

##   
## Call:  
## factanal(x = cerealRefinedData, factors = 8, rotation = "none")  
##   
## Uniquenesses:  
## Filling Natural Fibre Sweet Easy Salt   
## 0.132 0.367 0.164 0.332 0.793 0.483   
## Satisfying Energy Fun Kids Soggy Economical   
## 0.340 0.455 0.544 0.227 0.543 0.549   
## Health Family Calories Plain Crisp Regular   
## 0.203 0.291 0.584 0.491 0.421 0.461   
## Sugar Fruit Process Quality Treat Boring   
## 0.200 0.376 0.730 0.201 0.416 0.638   
## Nutritious   
## 0.252   
##   
## Loadings:  
## Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7 Factor8  
## Filling 0.772 0.248 0.352 -0.262   
## Natural 0.761 -0.155 0.151   
## Fibre 0.786 -0.173 -0.271 0.159 -0.277   
## Sweet 0.748 -0.280 0.121   
## Easy 0.285 0.183 0.171 0.228   
## Salt -0.224 0.490 -0.236 0.214 0.296 -0.101 0.125 0.111   
## Satisfying 0.696 0.282 0.156 0.118 0.101 0.200   
## Energy 0.692 0.212   
## Fun 0.287 0.520 0.178 -0.246   
## Kids 0.147 0.386 0.746 0.113 0.111 -0.108   
## Soggy -0.184 0.158 0.369 0.253 0.343 0.274   
## Economical 0.152 -0.167 0.490 0.142 -0.270 0.137 0.213   
## Health 0.841 -0.224 0.105 0.113   
## Family 0.249 0.330 0.697 0.146 -0.147   
## Calories -0.194 0.540 -0.237 0.142   
## Plain -0.249 -0.342 0.196 0.369 0.315 0.234   
## Crisp 0.194 0.460 0.174 -0.335 -0.247 -0.268 0.232   
## Regular 0.620 -0.173 0.191 -0.269   
## Sugar -0.304 0.702 -0.361 0.267   
## Fruit 0.340 0.233 -0.471 -0.237 -0.167 0.364 0.122   
## Process -0.317 0.220 0.290 -0.131   
## Quality 0.751 0.104 -0.263 0.186 0.274 -0.176   
## Treat 0.376 0.566 -0.337   
## Boring -0.302 -0.275 -0.120 0.380 0.147   
## Nutritious 0.823 -0.135 0.189   
##   
## Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7  
## SS loadings 6.090 3.353 2.156 1.080 0.737 0.526 0.467  
## Proportion Var 0.244 0.134 0.086 0.043 0.029 0.021 0.019  
## Cumulative Var 0.244 0.378 0.464 0.507 0.537 0.558 0.576  
## Factor8  
## SS loadings 0.399  
## Proportion Var 0.016  
## Cumulative Var 0.592  
##   
## Test of the hypothesis that 8 factors are sufficient.  
## The chi square statistic is 156.07 on 128 degrees of freedom.  
## The p-value is 0.0464

We observe that the loadings are not very clear, hence it has to be rotated to increase the very clear interpretability. It is done to maximize the high loadings and minimize low loadings.

### Exploring the model with 8 factors, with rotation

To select the maximum number of factors we check the following condition : pm + p <= p(p+1)/2 where p is the number of variables and m is the number of factors. Since pm+p = 225 and p(p+1)/2 = 325 the above condition os met.

fa3<-factanal(cerealRefinedData,factors=8,rotation = "varimax")  
fa3

##   
## Call:  
## factanal(x = cerealRefinedData, factors = 8, rotation = "varimax")  
##   
## Uniquenesses:  
## Filling Natural Fibre Sweet Easy Salt   
## 0.132 0.367 0.164 0.332 0.793 0.483   
## Satisfying Energy Fun Kids Soggy Economical   
## 0.340 0.455 0.544 0.227 0.543 0.549   
## Health Family Calories Plain Crisp Regular   
## 0.203 0.291 0.584 0.491 0.421 0.461   
## Sugar Fruit Process Quality Treat Boring   
## 0.200 0.376 0.730 0.201 0.416 0.638   
## Nutritious   
## 0.252   
##   
## Loadings:  
## Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7 Factor8  
## Filling 0.618 0.194 0.656   
## Natural 0.708 -0.231 0.136 0.152 0.120   
## Fibre 0.875 -0.124 0.117 -0.186   
## Sweet 0.683 0.151 0.165 0.345 0.114 0.133   
## Easy 0.220 0.270 0.266   
## Salt 0.703   
## Satisfying 0.524 0.343 0.112 0.397 0.133 0.281   
## Energy 0.582 0.180 0.160 0.105 0.346 0.114   
## Fun 0.119 0.160 0.440 0.364 0.206 0.154 0.127   
## Kids 0.857 -0.179   
## Soggy 0.120 -0.654   
## Economical -0.206 0.228 -0.525 0.163 0.212   
## Health 0.804 -0.259 0.101 0.213 0.166   
## Family 0.827   
## Calories -0.161 0.590 0.113 0.148   
## Plain -0.103 -0.469 -0.414 -0.169 -0.218 0.163   
## Crisp 0.165 0.313 0.602 0.267   
## Regular 0.698 0.165 -0.130   
## Sugar -0.162 0.849 0.119 0.159   
## Fruit 0.298 0.112 -0.128 0.153 0.657 0.196   
## Process -0.178 0.397 -0.127 -0.186 -0.163   
## Quality 0.677 -0.180 0.166 0.131 0.502   
## Treat 0.224 0.226 0.352 0.491 0.218 0.242   
## Boring -0.128 -0.265 -0.359 -0.236 -0.233 -0.178   
## Nutritious 0.825 -0.158 0.149 0.115   
##   
## Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7  
## SS loadings 4.875 2.560 2.315 1.695 1.305 0.886 0.643  
## Proportion Var 0.195 0.102 0.093 0.068 0.052 0.035 0.026  
## Cumulative Var 0.195 0.297 0.390 0.458 0.510 0.545 0.571  
## Factor8  
## SS loadings 0.529  
## Proportion Var 0.021  
## Cumulative Var 0.592  
##   
## Test of the hypothesis that 8 factors are sufficient.  
## The chi square statistic is 156.07 on 128 degrees of freedom.  
## The p-value is 0.0464

It can be observed that the significance level of the chi square fit statistic is 0.0464. This indicates that the hypothesis that the factor model holds.The 8 factors are able to explain 59% of the data.

However, variables such as ‘Easy’,‘Process’ and ‘Boring’ have high ’Uniquenesses - means it does not correlate with other variables.Higher the uniqueness lesser the communality and implies factors don’t explain variation.

### Removing variables with high Uniqueness values and running the factor analysis again

After removing high uniqueness values it is necessary to again run the KMO test and Kaiser criteria test:

**1)KMO Test**

cerealRefinedData = cerealRefinedData [-c(5,21,24)]

KMO(cerealRefinedData)

**Output:**

KMO(new\_data)

Kaiser-Meyer-Olkin factor adequacy

Call: KMO(r = cerealRefinedData)

Overall MSA = 0.85

MSA for each item =

Filling Natural Fibre Sweet Salt Satisfying Energy Fun Kids Soggy Economical

0.89 0.91 0.89 0.79 0.80 0.92 0.93 0.85 0.65 0.60 0.73

Health Family Calories Plain Crisp Regular Sugar Fruit Quality Treat Nutritious

0.91 0.71 0.86 0.81 0.80 0.86 0.78 0.76 0.90 0.87 0.93

All the variables are high correlation and values are also greater than 0.5

**2) Kaiser Criteria Test: To determine number of factors**

eigen(cov(cerealRefinedData))$values

eigen(cov(cerealRefinedData))$values

[1] 5.9180225 4.1699627 3.0670089 1.7724548 1.2821949 0.9611486 0.8822224 0.7661686 0.6753200 0.6396474 0.6241716 0.4941430

[13] 0.4241265 0.4016186 0.3789063 0.3039951 0.2651844 0.2516832 0.2228581 0.1941443 0.1757777 0.1527354

There are 5 values which are greater than 1 and hence we will proceed with 5 factors

fa5=factanal(new\_data,factors = 5,rotation = "varimax")

fa5

Call:

factanal(x = new\_data, factors = 5, rotation = "varimax")

Uniquenesses:

Filling Natural Fibre Sweet Salt Satisfying Energy Fun Kids Soggy Economical Health

0.291 0.391 0.316 0.355 0.526 0.377 0.411 0.500 0.225 0.786 0.711 0.214

Family Calories Plain Crisp Regular Sugar Fruit Quality Treat Nutritious

0.351 0.578 0.574 0.660 0.555 0.205 0.582 0.395 0.353 0.237

Loadings:

Factor1 Factor2 Factor3 Factor4 Factor5

Filling 0.657 0.104 0.159 0.142 0.470

Natural 0.745 -0.176 0.143

Fibre 0.814 -0.112

Sweet 0.724 0.322 0.120

Salt -0.105 0.675

Satisfying 0.589 0.342 0.171 0.352

Energy 0.619 0.137 0.226 0.364

Fun 0.135 0.156 0.368 0.562

Kids 0.876

Soggy 0.127 -0.444

Economical 0.112 -0.252 0.405 -0.201

Health 0.851 -0.242

Family 0.790 0.105

Calories -0.170 0.590 0.123 0.168

Plain -0.131 -0.614 -0.136

Crisp 0.152 0.320 0.450

Regular 0.654

Sugar -0.201 0.843 0.167 -0.114

Fruit 0.345 0.192 -0.290 0.406 0.116

Quality 0.695 -0.205 0.182 0.201

Treat 0.242 0.209 0.286 0.680

Nutritious 0.857 -0.128

Factor1 Factor2 Factor3 Factor4 Factor5

SS loadings 5.068 2.415 2.203 2.078 0.644

Proportion Var 0.230 0.110 0.100 0.094 0.029

Cumulative Var 0.230 0.340 0.440 0.535 0.564

Test of the hypothesis that 5 factors are sufficient.

The chi square statistic is 244.95 on 131 degrees of freedom.

The p-value is 6.27e-09

This indicates that the hypothesis that the factor model holds. The number of factors are reduced to 5.The 5 factors are able to explain about 56% of the data.

### Adding Factor Names

Factor1 = Health -> Filling, Natural, Fibre, Satisfying, Energy, Health, Regular, Quality, Nutritious

Factor2 = Taste -> Sweet, Salt, Calories, Sugar

Factor3 = Family -> Satisfying,Fun,Kids,Family, Crisp

Factor4 = Texture -> Fun, Soggy, Plain, Crisp, Treat

Factor5 = Experience -> Filling, Satisfying, Energy

> colnames(fa5$loadings) <- c("Health","Taste","Family","Texture","Experience")

> fa5$loadings

Loadings:

Health Taste Family Texture Experience

Filling 0.657 0.104 0.159 0.142 0.470

Natural 0.745 -0.176 0.143

Fibre 0.814 -0.112

Sweet 0.724 0.322 0.120

Salt -0.105 0.675

Satisfying 0.589 0.342 0.171 0.352

Energy 0.619 0.137 0.226 0.364

Fun 0.135 0.156 0.368 0.562

Kids 0.876

Soggy 0.127 -0.444

Economical 0.112 -0.252 0.405 -0.201

Health 0.851 -0.242

Family 0.790 0.105

Calories -0.170 0.590 0.123 0.168

Plain -0.131 -0.614 -0.136

Crisp 0.152 0.320 0.450

Regular 0.654

Sugar -0.201 0.843 0.167 -0.114

Fruit 0.345 0.192 -0.290 0.406 0.116

Quality 0.695 -0.205 0.182 0.201

Treat 0.242 0.209 0.286 0.680

Nutritious 0.857 -0.128

Health Taste Family Texture Experience

SS loadings 5.068 2.415 2.203 2.078 0.644

Proportion Var 0.230 0.110 0.100 0.094 0.029

Cumulative Var 0.230 0.340 0.440 0.535 0.564

We can see that the some of the variables are present in more than one factor.This is known as the ‘Communality’ effect.

print(loadings(fa5), digits = 2, cutoff = 0.3, sort = TRUE)

Loadings:

Health Taste Family Texture Experience

Filling 0.66 0.47

Natural 0.75

Fibre 0.81

Satisfying 0.59 0.34 0.35

Energy 0.62 0.36

Health 0.85

Regular 0.65

Quality 0.69

Nutritious 0.86

Sweet 0.72 0.32

Salt 0.67

Calories 0.59

Sugar 0.84

Kids 0.88

Family 0.79

Fun 0.37 0.56

Plain -0.61

Treat 0.68

Soggy -0.44

Economical 0.41

Crisp 0.32 0.45

Fruit 0.34 0.41

Health Taste Family Texture Experience

SS loadings 5.07 2.41 2.20 2.08 0.64

Proportion Var 0.23 0.11 0.10 0.09 0.03

Cumulative Var 0.23 0.34 0.44 0.53 0.56

### Reliability of Factor

#reliability of factors

factor1 <- c(1,2,3,6,7,12,17,20,22)

factor2 <- c(4,5,14,18)

factor3 <- c(6,8,9,13,16)

factor4 <- c(8,10,15,16,21)

factor5 <- c(1,6,7)

factor1alpha <- psych::alpha(new\_data[,factor1], check.keys = TRUE)

factor2alpha <- psych::alpha(new\_data[,factor2], check.keys = TRUE)

factor3alpha <- psych::alpha(new\_data[,factor3], check.keys = TRUE)

factor4alpha <- psych::alpha(new\_data[,factor4], check.keys = TRUE)

factor5alpha <- psych::alpha(new\_data[,factor5], check.keys = TRUE)

factor1alpha$total$raw\_alpha

factor2alpha$total$raw\_alpha

factor3alpha$total$raw\_alpha

factor4alpha$total$raw\_alpha

factor5alpha$total$raw\_alpha

Output:

factor1alpha$total$raw\_alpha

[1] 0.9131105

> factor1alpha$total$raw\_alpha

[1] 0.9131105

> factor2alpha$total$raw\_alpha

[1] 0.809945

> factor3alpha$total$raw\_alpha

[1] 0.7379407

> factor4alpha$total$raw\_alpha

[1] 0.7254357

> factor5alpha$total$raw\_alpha

[1] 0.8359789

As all the reliability factors are greater than 0.7 the factors are all reliable.

### Average Factor Scores grouped by the cereal.

Factors

new\_data$Cerals <- c(data[,1])

new\_data$factor1Score <- apply(new\_data[,factor1],1,mean)

new\_data$factor2Score <- apply(new\_data[,factor2],1,mean)

new\_data$factor3Score <- apply(new\_data[,factor3],1,mean)

new\_data$factor4Score <- apply(new\_data[,factor4],1,mean)

new\_data$factor5Score <- apply(new\_data[,factor5],1,mean)

colnames(new\_data)[23:27] <- c("health","taste","family","texture","experience")

aggregateCereal<-aggregate(new\_data[,23:27], list(new\_data[,"Cerals"]), mean)

format(aggregateCereal, digits = 2)

Output:

Group.1 health taste family texture experience

AllBran 3.9 2.0 2.8 2.1 3.6

CMuesli 4.0 2.8 3.6 2.5 4.2

CornFlakes 3.3 2.6 3.8 2.7 3.6

JustRight 3.6 2.5 3.4 2.5 3.7

Komplete 4.0 2.6 3.1 2.6 4.0

NutriGrain 3.4 3.0 4.0 2.9 3.9

PMuesli 4.1 2.9 3.6 2.8 4.2

RiceBubbles 2.9 2.0 3.8 2.8 3.2

SpecialK 3.5 2.1 3.6 2.6 3.7

Sustain 4.2 2.1 3.8 2.8 4.3

Vitabrit 3.9 1.7 3.3 2.5 4.1

Weetabix 3.9 1.9 3.1 2.3 3.8

Above table helps in analyzing the impact of factors on each and every cereal brand.

# Problem 2 – LeslieSalt Company

# Question 1

**Q1: What is the nature of each of the variables? Which variable is dependent variable and what are the independent variables in the model?**

* Loading and reading the Data
* Variables Price, Size, Elevation, Sewer, date, and Distance are numeric in nature.
* County and Flood variables are dummy variables represented by 0 and 1 which means San Mateo and Santa Clara for County and Flood by tidal action and otherwise for Flood variable respectively.
* Below code gives the string of the variables.

> mydata = read.csv("leslie\_salt.csv",1)

> attach(mydata)

> str(mydata)

**'data.frame': 31 obs. of 8 variables:**

**$ Price :** num 4.5 10.6 1.7 5 5 3.3 5.7 6.2 19.4 3.2 ...

**$ County :** int 1 1 0 0 0 1 1 1 1 1 ...

**$ Size :** num 138.4 52 16.1 1695.2 845 ...

**$ Elevation:** int 10 4 0 1 1 2 4 4 20 0 ...

**$ Sewer :** int 3000 0 2640 3500 1000 10000 0 0 1300 6000 ...

**$ Date :** int -103 -103 -98 -93 -92 -86 -68 -64 -63 -62 ...

**$ Flood :** int 0 0 1 0 1 0 0 0 0 0 ...

**$ Distance :** num 0.3 2.5 10.3 14 14 0 0 0 1.2 0 ...

Taking summary of the attached data, we can see that each variable has Minimum, 1st quadrant, 3rd quadrant, Median, Mean and Maximum.

> summary(mydata)

**Price County Size Elevation Sewer**

Min.: 1.70 Min.:0.0000 Min.: 6.90 Min.: 0.000 Min.: 0

1st Qu.: 5.35 1st Qu.:0.0000 1st Qu.:20.35 1st Qu.:2.000 1st Qu.:0

Median:11.70 Median:1.0000 Median:51.40 Median:4.000 Median:900

Mean:11.95 Mean:0.6129 Mean:139.97 Mean:4.645 Mean:1981

3rd Qu.:16.05 3rd Qu.:1.0000 3rd Qu.:104.10 3rd Qu.:7.000 3rd Qu.: 3450

Max.:37.20 Max.:1.0000 Max.:1695.20 Max.:20.000 Max.:10000

**Date Flood Distance**

Min. :-103.00 Min. :0.0000 Min. : 0.000

1st Qu.: -63.50 1st Qu.:0.0000 1st Qu.: 0.850

Median : -59.00 Median :0.0000 Median : 4.900

Mean : -58.65 Mean :0.1613 Mean : 5.132

3rd Qu.: -51.00 3rd Qu.:0.0000 3rd Qu.: 5.500

Max. : -4.00 Max. :1.0000 Max. :16.500

* Below code and output gives the scatter plot between variables and infer the relation. Here the target or dependent variable is Price since we need to predict the fair market value of the land and other 7 variables are independent variables.
* Price is not correlated with County and Flood.
* Elevation, Size, Distance and Date variables are positively correlated with Price variable. That is if all the Elevation, Size, Distance and Date increases the fair market price of the land increases.
* Sewer is negatively correlated with Price variable. That is If the distance from nearest sewer increases the fair market value of the Land decreases.

> pairs(mydata)

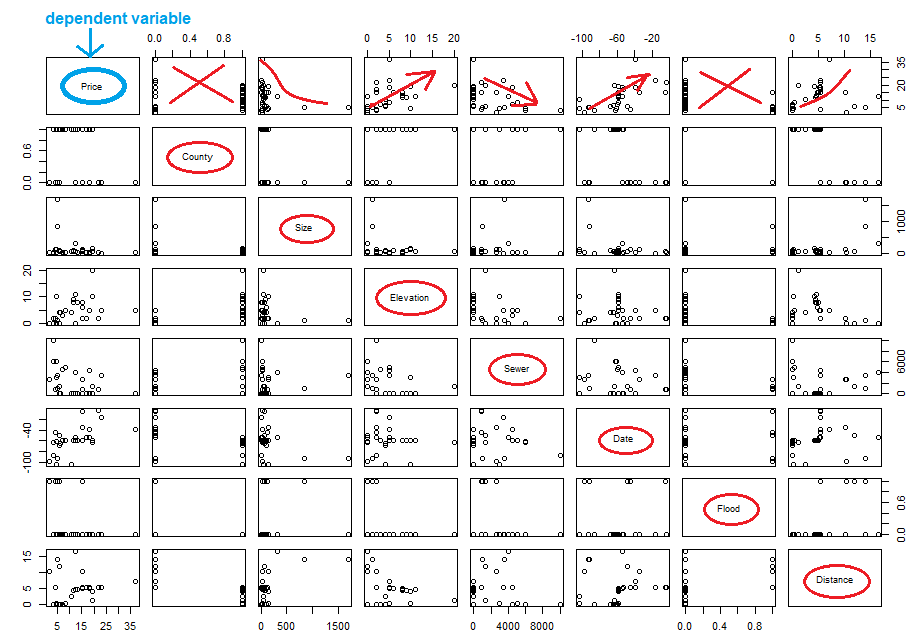


Fig.1: Scatterplot of Leslie Salt company datasets.

* Below code uses correlation function to find the strength of correlation between variables. Considering the correlation only between Price variable and other independent variables, we can conclude with the below:
* Price is negatively correlated with Sewer and Flood with 39% and 32% relationship, that is as sewer distance and Flood affinity due to tidal actions increases the Market value of the land decreases.
* Similarly, Price is positively correlated with Elevation and Date with 35% and 59.46% relationship, inferring that Price increases with increase in elevation and date of sales.

> cor(mydata)

**Price County Size Elevation Sewer Date Flood**

**Price** **1.00** -0.1822 -0.2397 **0.3518** **-0.3912** **0.5946** **-0.3230**

**County** -0.1822 **1.00** **-0.3394** **0.4751** -0.0500 **-0.3698 -0.5518**

**Size** **-0.2397 -0.3394** **1.00** -0.2094 0.0533 **-0.3494** 0.1089

**Elevation** **0.3518 0.4751** -0.2094 **1.00** **-0.3594** -0.0565 **-0.3730**

**Sewer** **-0.3912** -0.0500 0.0533 **-0.3594** **1.00** -0.1514 -0.1130

**Date** **0.5946** **-0.3698 -0.3494** -0.0565 -0.1514 **1.00** 0.0153

**Flood** **-0.3230 -0.5518** 0.1089 **-0.3730** -0.1130 0.0153 **1.00**

**Distance** 0.0933 **-0.7422** **0.5569** -0.3624 -0.1586 0.0443 **0.4233**

**Distance**

**Price** 0.09331133

**County** **-0.74220440**

**Size** **0.55694587**

**Elevation** **-0.36246039**

**Sewer** -0.15865389

**Date** 0.04438251

**Flood** **0.42330840**

**Distance** **1.00000000**

# b) Question 3:

**Q3: Set up a regression equation, run the model and discuss your results**

Below is code for multiple linear regression model set up, which provide us the below result.

> Model1 = lm (Price ~ County + Size + Elevation + Sewer + Date + Flood + Distance, data = mydata)

> summary(Model1)

**Call:**

lm (formula = Price ~ County + Size + Elevation + Sewer + Date +

Flood + Distance, data = mydata)

**Residuals:**

Min 1Q Median 3Q Max

-5.169 -2.957 -0.256 2.070 13.031

**Coefficients:**

**Estimate Std. Error t value Pr(>|t|)**

**(Intercept)** 2.364e+01 3.829e+00 6.174 2.68e-06 \*\*\*

**County** -8.789e+00 3.652e+00 -2.407 0.024532 \*

**Size** -6.043e-03 3.501e-03 -1.726 0.097702 .

**Elevation** 5.193e-01 2.386e-01 2.177 0.040030 \*

**Sewer** -9.573e-04 4.169e-04 -2.296 0.031126 \*

**Date** 8.508e-02 4.865e-02 1.749 0.093646 .

**Flood** -1.202e+01 2.989e+00 -4.020 0.000536 \*\*\*

**Distance** 1.858e-01 3.395e-01 0.547 0.589386

**---------------------------------------------------------------**

**Signif. codes:** 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

**Residual standard error:** 4.431 on 23 degrees of freedom

**Multiple R-squared:** 0.747, **Adjusted R-squared:** 0.67

**F-statistic:** 9.703 on 7 and 23 DF, **p-value:** 1.351e-05

--------------------------------------------------------------------------------------------------------

**Inference:** From the above output, we can see that County, Elevation, Sewer, and Flood has a significant difference affecting the Market price value of the Land. Other variables like Size, Date and Distance does not affect the Market Fair value of the Leslie salt’s land.

* Referring to the Multiple R Squared from the output, we are 74.7% sure of the variations due to the significant variables affecting the Market Fair value.
* We can model another multiple linear regression but removing the insignificant variables like size, date and distance, If the Adjusted goes beyond 67% then we need to drop model2 and stick with model 1 itself for predicting the Market Fair value of the land.
* We use Predict function to predict the Market Fair value using the Model 1:

> myprediction = Predict (Price, data.Frame(“County”=0, “Elevation”=12, “Sewer”=8900, “Flood”=1, data=Model1))

> summary (my prediction)

* By changing the values fed to the data frame we can predict the market fair value of the land within the given range of the dataset.

**NOTE:** We cannot predict the value of the Price outside the data set range.

# c) Question 2:

**Q2: Check whether the variables require any transformation individually**

Price.log=log(mydata$Price)

County.log=log(mydata$County)

Size.log=log(mydata$Size)

Elevation.log=log(mydata$Elevation)

Sewer.log=log(mydata$Sewer)

Date.log=log(mydata$Date)

Flood.log=log(mydata$Flood)

Distance.log=log(mydata$Distance)

new.log.variable=cbind(Price.log,Size.log,Elevation.log,Sewer.log,Distance.log)

mylogdata=cbind(mydata,new.log.variable)

head(new.log.variable,32)

     Price.log Size.log Elevation.log Sewer.log Distance.log

[1,] 1.5040774 4.930148     2.3025851 8.006368 -1.2039728

[2,] 2.3608540 3.951244     1.3862944 -Inf 0.9162907

[3,] 0.5306283 2.778819          -Inf 7.878534 2.3321439

[4,] 1.6094379 7.435556     0.0000000 8.160518 2.6390573

[5,] 1.6094379 6.739337     0.0000000 6.907755 2.6390573

[6,] 1.1939225 1.931521     0.6931472 9.210340 -Inf

[7,] 1.7404662 4.662495     1.3862944 -Inf -Inf

[8,] 1.8245493 4.036009     1.3862944 -Inf -Inf

[9,] 2.9652731 3.939638     2.9957323 7.170120 0.1823216

[10,] 1.1631508 3.095578          -Inf 8.699515 -Inf

[11,] 1.5475625 3.095578          -Inf 8.699515 -Inf

[12,] 1.9315214 3.321432     1.0986123 8.411833 -Inf

[13,] 2.0918641 2.923162     1.6094379 8.517193 -0.6931472

[14,] 2.4510051 4.247066     2.0794415 -Inf 1.4816045

[15,] 2.9601051 4.981550     2.3025851 -Inf 1.4350845

[16,] 2.4595888 4.346399     2.1972246 -Inf 1.5040774

[17,] 2.5877640 3.265759     2.0794415 -Inf 1.5475625

[18,] 2.7146947 4.627910     1.7917595 -Inf 1.5892352

[19,] 2.5176965 3.901973     2.3978953 -Inf 1.5260563

[20,] 2.7278528 2.501436     2.0794415 -Inf 1.6094379

[21,] 2.5014360 5.770194          -Inf 8.294050 2.8033604

[22,] 2.8959119 2.292535     1.6094379 -Inf 1.6486586

[23,] 2.8213789 2.727853     0.6931472 -Inf 1.7047481

[24,] 1.7749524 4.010963          -Inf 7.185387 2.4765384

[25,] 1.3862944 4.755313     0.6931472 6.802395 1.7047481

[26,] 3.6163088 2.708050     1.6094379 -Inf 1.9740810

[27,] 2.9014216 3.152736     1.6094379 8.393895 1.7047481

[28,] 2.7146947 4.888844     0.6931472 7.878534 2.3223877

[29,] 3.1311369 2.484907     1.6094379 8.131531 1.7047481

[30,] 2.7212954 4.204693     0.6931472 6.802395 1.7047481

[31,] 3.0864866 3.427515     0.6931472 6.802395 1.7047481

**Only price.log and size.log can be made use of as the other field values has infinity in it.**

The below Model with the log transformations:

> Model9 = lm (Price.log ~ County + Size.log + Elevation +

+                Sewer + Date + Flood + Distance, data = mylogdata)

> summary(Model9)

Call:

lm(formula = Price.log ~ County + Size.log + Elevation + Sewer +

   Date + Flood + Distance, data = mylogdata)

Residuals:

    Min     1Q Median       3Q Max

-0.53335 -0.20482 -0.04464  0.19631 0.64438

Coefficients:

             Estimate Std. Error t value Pr(>|t|)

(Intercept)  3.418e+00 3.071e-01  11.129 9.79e-11 \*\*\*

County      -1.624e-01 2.553e-01  -0.636 0.530995

Size.log    -9.886e-02 5.834e-02  -1.694 0.103692

Elevation    5.338e-02 1.695e-02   3.148 0.004501 \*\*

Sewer       -9.277e-05 3.006e-05  -3.086 0.005220 \*\*

Date         1.448e-02 3.261e-03   4.439 0.000188 \*\*\*

Flood       -9.471e-01 2.046e-01  -4.629 0.000117 \*\*\*

Distance     5.159e-02 2.363e-02   2.183 0.039478 \*

---

Signif. codes:  0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3138 on 23 degrees of freedom

Multiple R-squared:  0.853, Adjusted R-squared:  0.8083

F-statistic: 19.07 on 7 and 23 DF,  p-value: 3.533e-08

# Problem 3 – All Greens Franchise

The variables have been categorized into two classes- Independent and dependent. The average sale price has been classified as the dependent variable and the rest variables have been classified as independent variable.

**1**. The below results show that there are 27 rows and 6 variables and all are of type number or integer. The variables have been categorized into two classes- Independent and dependent. The average sale price has been classified as the dependent variable and the rest variables have been classified as independent variable.

*Dataset\_All\_Greens\_Franchise <- read.csv("Dataset\_All Greens Franchise.csv",TRUE)*

*dim(Dataset\_All\_Greens\_Franchise)*

*str(Dataset\_All\_Greens\_Franchise)*

'data.frame': 27 obs. of 6 variables:

$ X1: num 231 156 10 519 437 487 299 195 20 68 ...

$ X2: num 3 2.2 0.5 5.5 4.4 ...

$ X3: int 294 232 149 600 567 571 512 347 212 102 ...

$ X4: num 8.2 6.9 3 12 10.6 ...

$ X5: num 8.2 4.1 4.3 16.1 14.1 ...

$ X6: int 11 12 15 1 5 4 10 12 15 8 ...

**2**. Further to combine the data frame I used cbind and then made a correlation plot to see the correlation of both dependent and independent variable. . The figures below figure shows that there is a very high correlation between X1 and all other dependent variables. The other dependent variables are also highly correlated with each other so there can be a problem of multi collinearity. Let do the regression analysis of complete dataset:

*cbind(colSums(is.na(Dataset\_All\_Greens\_Franchise)))*

X1 0

X2 0

X3 0

X4 0

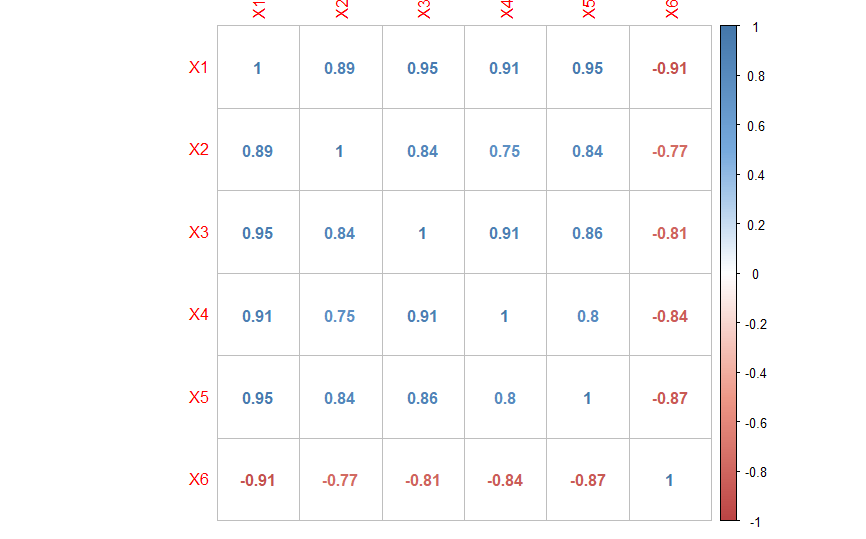
X5 0

X6 0

*install.packages("corrplot")*

*corel<-cor(Dataset\_All\_Greens\_Franchise)*

*corrplot(corel,method ="number") [,1]*



**3**. Model -1 summary below shows that correlation, both R-squared and R-squared adjusted are greater than 99%. That means mean price is strongly correlated with 5 independent variables. Coefficient of determination or the Multiple R-Square value is 0.9932. That means 99.3% changes in the sales can be explained by this model. Value of adjusted R-Square is 0. 991 indicates only 99.1% variation in performance can be measured by this model after considering all related factors.

*reg1 <- lm(X1 ~ ., data= Dataset\_All\_Greens\_Franchise)*

*summary(reg1)*

Call:

lm(formula = X1 ~ ., data = Dataset\_All\_Greens\_Franchise)

Residuals:

Min 1Q Median 3Q Max

-26.338 -9.699 -4.496 4.040 41.139

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -18.85941 30.15023 -0.626 0.538372

X2 16.20157 3.54444 4.571 0.000166 \*\*\*

X3 0.17464 0.05761 3.032 0.006347 \*\*

X4 11.52627 2.53210 4.552 0.000174 \*\*\*

X5 13.58031 1.77046 7.671 1.61e-07 \*\*\*

X6 -5.31097 1.70543 -3.114 0.005249 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 17.65 on 21 degrees of freedom

Multiple R-squared: 0.9932, Adjusted R-squared: 0.9916

F-statistic: 611.6 on 5 and 21 DF, p-value: < 2.2e-16

Now let us find the variance:

*install.packages("car")*

*library(car)*

*vif(reg1)*

X2 X3 X4 X5 X6

4.240914 10.122480 7.624391 6.912318 5.818768

Since the value for X3 is greater than 10 which is unacceptable. So let us drop this variable from the model and try to build another one.

3. Model -2 summary below shows that correlation, both R-squared and R-squared adjusted are greater than 99%. That means mean price is strongly correlated with 5 independent variables. Coefficient of determination or the Multiple R-Square value is 0.9902. That means 99.02% changes in the sales can be explained by this model. Value of adjusted R-Square is 0. 988 indicates only 98.8% variation in performance can be measured by this model after considering all related factors.

*reg2 <- lm(X1 ~ .-X3, data= Dataset\_All\_Greens\_Franchise)*

*summary(reg2)*

Call:

lm(formula = X1 ~ . - X3, data = Dataset\_All\_Greens\_Franchise)

Residuals:

Min 1Q Median 3Q Max

-30.422 -12.858 -6.477 16.160 45.255

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -39.460 34.411 -1.147 0.2638

X2 20.444 3.815 5.359 2.22e-05 \*\*\*

X4 16.966 2.093 8.107 4.73e-08 \*\*\*

X5 15.673 1.910 8.206 3.86e-08 \*\*\*

X6 -4.043 1.937 -2.088 0.0486 \*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 20.68 on 22 degrees of freedom

Multiple R-squared: 0.9902, Adjusted R-squared: 0.9884

F-statistic: 555.4 on 4 and 22 DF, p-value: < 2.2e-16

Now let us find the variance:

*vif(reg2)*

X2 X4 X5 X6

3.579850 3.795323 5.861520 5.468943

Now since none of the variances are greater than 10 so this model is good.

**4.** Let’s resolve the issue of multicollinearity using PCA

*pca1 <- prcomp(Dataset\_All\_Greens\_Franchise[,-1], scale. = T)*

*summary(pca1)*

Importance of components:

PC1 PC2 PC3 PC4 PC5

Standard deviation 2.0769 0.5277 0.47972 0.35207 0.23260

Proportion of Variance 0.8627 0.0557 0.04603 0.02479 0.01082

Cumulative Proportion 0.8627 0.9184 0.96439 0.98918 1.00000

From the values for Proportion of Variance that the first factor which is PC1 is sufficient and it is explaining 86% of the variance. Second component can only explain <6% variance. Lets use the first component for building the regression model:

*loadings(pca1)*

*data1<- data.frame(Dataset\_All\_Greens\_Franchise[,1], pca1$x[,1])*

*colnames(data1) <- c("X1", "PC1")*

*reg3 <- lm(X1 ~ ., data=data1)*

*summary(reg3)*

Call:

lm(formula = X1 ~ ., data = data1)

Residuals:

Min 1Q Median 3Q Max

-32.168 -15.059 -3.809 11.915 47.944

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 286.574 3.774 75.94 <2e-16 \*\*\*

PC1 92.013 1.852 49.69 <2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 19.61 on 25 degrees of freedom

Multiple R-squared: 0.99, Adjusted R-squared: 0.9896

F-statistic: 2469 on 1 and 25 DF, p-value: < 2.2e-16

# 5. Conclusion

Thus, by using the techniques like PCA and linear regression we are able reduced multi-collinearity in the data and also we could explain the importance of all the independent variables on X1 which is Annual net sales.