Factor Analysis

Yuchen Hu

11/26/2018

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
library(psych)
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(nFactors)
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## Loading required package: boot
```

```
##
## Attaching package: 'boot'
```

```
## The following object is masked from 'package:psych':
##
##     logit
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'lattice'

## The following object is masked from 'package:boot':
##
##     melanoma

##
## Attaching package: 'nFactors'

## The following object is masked from 'package:lattice':
##
##     parallel
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'

## The following objects are masked from 'package:data.table':
##
##     dcast, melt
```

```
library(tidyr)
```

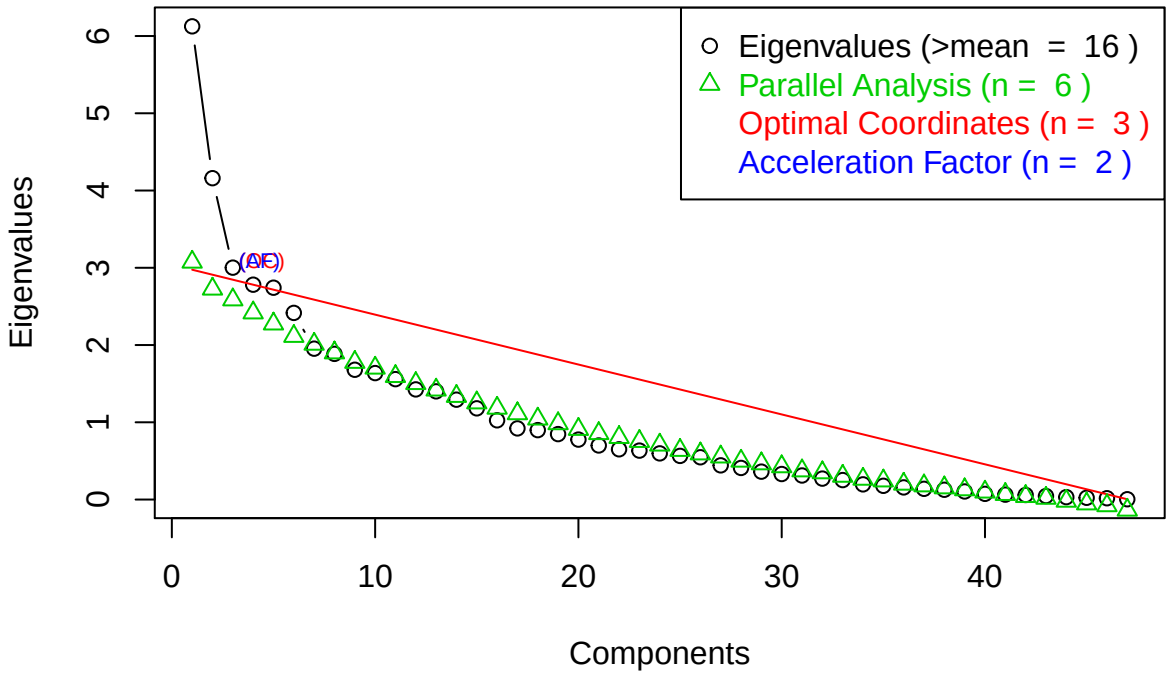
```
##
## Attaching package: 'tidyr'

## The following object is masked from 'package:reshape2':
##
##     smiths
```

```
food <- read.csv("food_coded.csv",stringsAsFactors = FALSE)
#summary(food)
```

basic FA

```
food_numeric <- food %>% select_if(is.numeric)
food_numeric <- food_numeric[complete.cases(food_numeric), ]
ev <- eigen(cor(food_numeric))
ap <- parallel(subject=nrow(food_numeric),var=ncol(food_numeric),
               rep=100,cent=.05)
nS <- nScree(x=ev$values, aparallel=ap$eigen$qevpea)
plotnScree(nS)
```




```
fa_numeric <- factanal(food_numeric, 5, rotation="varimax", scores="regression")
print(fa_numeric, digits=2, cutoff=.3, sort=TRUE)
```

```
##
## Call:
## factanal(x = food_numeric, factors = 5, scores = "regression",     rotation = "varimax")
##
## Uniquenesses:
##                          Gender                       breakfast
##                            0.86                            0.62
##               calories_chicken                    calories_day
##                            0.50                            0.51
##                 calories_scone                          coffee
##                            0.70                            0.87
##   comfort_food_reasons_coded                            cook
##                            0.03                            0.78
## comfort_food_reasons_coded.1                         cuisine
##                            0.00                            0.88
##             diet_current_coded                           drink
##                            0.89                            0.83
##           eating_changes_coded          eating_changes_coded1
##                            0.85                            0.97
##                     eating_out                      employment
##                            0.72                            0.84
##                    ethnic_food                        exercise
##                            0.43                            0.80
##               father_education             fav_cuisine_coded
##                            0.79                            0.81
##                       fav_food                           fries
##                            0.81                            0.91
```

```
##                    fruit_day                   grade_level
##                         0.34                          0.38
##                  greek_food               healthy_feeling
##                         0.32                          0.94
##             ideal_diet_coded                        income
##                         0.93                          0.59
##                  indian_food                  italian_food
##                         0.11                          0.80
##               life_rewarding                marital_status
##                         0.88                          0.81
##             mother_education             nutritional_check
##                         0.93                          0.76
##                on_off_campus                 parents_cook
##                         0.76                          0.90
##                 pay_meal_out                 persian_food
##                         0.89                          0.18
##       self_perception_weight                         soup
##                         0.80                          0.85
##                       sports                    thai_food
##                         0.84                          0.20
##            tortilla_calories               turkey_calories
##                         0.57                          0.62
##                  veggies_day                      vitamins
##                         0.24                          0.69
##              waffle_calories
##                         0.52
##
## Loadings:
##                               Factor1 Factor2 Factor3 Factor4 Factor5
## ethnic_food                    0.74
## greek_food                     0.80
## indian_food                    0.93
## persian_food                   0.86
## thai_food                      0.87
## calories_chicken                       0.56    0.35
## tortilla_calories                      0.64
## turkey_calories                        0.58
## waffle_calories                        0.69
## fruit_day                      0.35            0.68
## veggies_day                    0.30            0.79
## comfort_food_reasons_coded                             0.97
## comfort_food_reasons_coded.1                           0.98
## grade_level                                                     0.69
## income                                                         -0.56
## Gender
## breakfast                                     -0.50
## calories_day                   0.42    0.39    0.39
## calories_scone                         0.35            0.37
## coffee
## cook                                                           -0.41
## cuisine
## diet_current_coded
## drink
## eating_changes_coded                          0.33
```

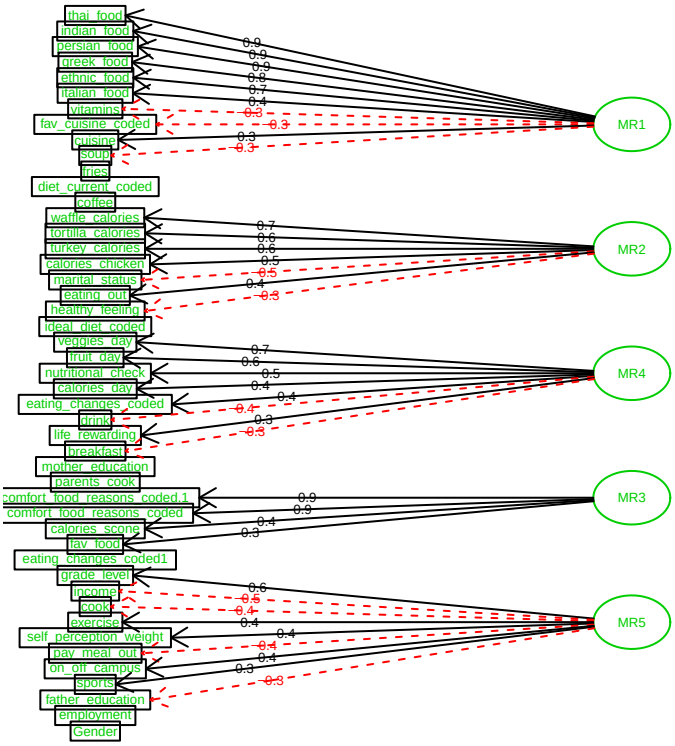
```
## eating_changes_coded1
## eating_out                            0.46
## employment
## exercise                                     -0.36
## father_education                                            -0.31
## fav_cuisine_coded
## fav_food
## fries
## healthy_feeling
## ideal_diet_coded
## italian_food                  0.38
## life_rewarding
## marital_status                       -0.38
## mother_education
## nutritional_check                     0.39
## on_off_campus                                               0.44
## parents_cook
## pay_meal_out
## self_perception_weight                       -0.30
## soup
## sports
## vitamins                     -0.43
##
##                Factor1 Factor2 Factor3 Factor4 Factor5
## SS loadings       5.12    3.01    2.86    2.69    2.06
## Proportion Var    0.11    0.06    0.06    0.06    0.04
## Cumulative Var    0.11    0.17    0.23    0.29    0.33
##
## Test of the hypothesis that 5 factors are sufficient.
## The chi square statistic is 946.87 on 856 degrees of freedom.
## The p-value is 0.0162
```

```
# factor1: food choices; 2:calories guess; 3:life style ; 4:comfort food; 5:education
fa_numeric <- fa(food_numeric, 5)
```

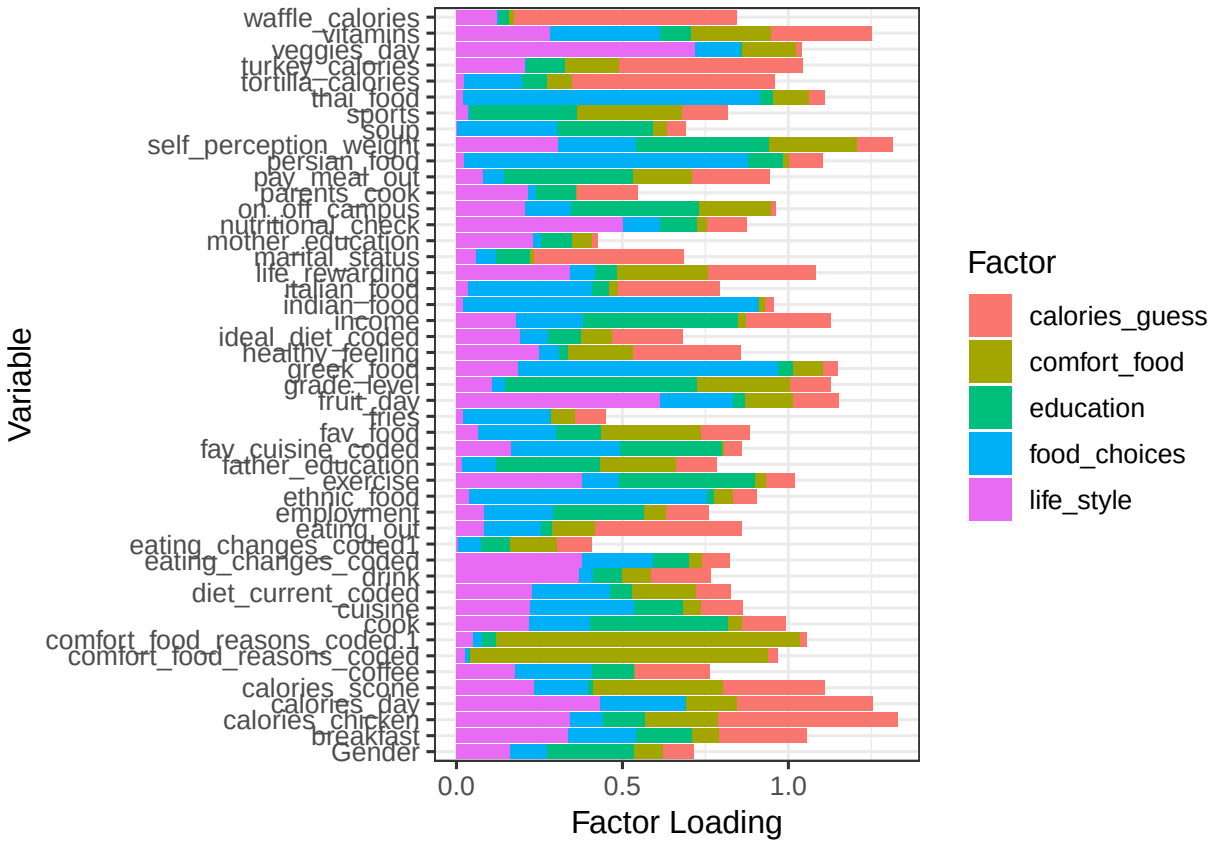
```
## Loading required namespace: GPArotation
```

```
fa.diagram(fa_numeric,side=1)
```

Factor Analysis




```
# plot the result
gathering <- as.data.frame(matrix(nrow = dim(food_numeric)[2], ncol =5))
gathering$Variable <- colnames(food_numeric)
for (i in 1:5) {
  for (j in 1:dim(food_numeric)[2]) {
    gathering[j, i] <- fa_numeric$loadings[j, i]
  }
}
colnames(gathering) <- c("food_choices","calories_guess",
                         "life_style","comfort_food","education", "Variable")
gathering <- gathering %>% gather("Factor", "Value", 1:5)
ggplot(gathering, aes(Variable, abs(Value), fill=Factor)) +
  geom_bar(stat="identity") + coord_flip() +
  ylab("Factor Loading") +
  theme_bw(base_size=12)
```



run regression on score

```
factor_gpa <- merge(as.numeric(food$GPA),fa_numeric$scores,
                    by="row.names",all.x=FALSE)[,-1]
```

```
## Warning in merge(as.numeric(food$GPA), fa_numeric$scores, by =
## "row.names", : NAs introduced by coercion
```

```
names(factor_gpa) <- c("GPA","food_choices","calories_guess","life_style",
                       "comfort_food","education")
# regression based on numerical GPA
fit.factor <- lm(GPA~food_choices+calories_guess+life_style+comfort_food+education,
                 data=factor_gpa)
summary(fit.factor)
```

```
##
## Call:
## lm(formula = GPA ~ food_choices + calories_guess + life_style +
##     comfort_food + education, data = factor_gpa)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.64709 -0.23254  0.06347  0.17809  0.67057
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)     3.45408    0.04089  84.464  < 2e-16 ***
## food_choices    0.02948    0.04334   0.680  0.49933
## calories_guess  0.15661    0.04671   3.353  0.00148 **
## life_style     -0.06132    0.04638  -1.322  0.19179
## comfort_food   -0.11280    0.04160  -2.712  0.00900 **
## education       0.02096    0.04648   0.451  0.65389
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3139 on 53 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.2728, Adjusted R-squared:  0.2042
## F-statistic: 3.977 on 5 and 53 DF,  p-value: 0.003891
```

```r
# regression based on categorical GPA
factor_gpa$GPA <- cut(factor_gpa$GPA, c(0,2.5,3,3.25,3.5,3.75,4.0))
factor_gpa$GPA <- as.numeric(factor_gpa$GPA)
fit.factor <- lm(GPA~food_choices+calories_guess+life_style+comfort_food+education,
                 data=factor_gpa)
summary(fit.factor)
```

```
##
## Call:
## lm(formula = GPA ~ food_choices + calories_guess + life_style +
##     comfort_food + education, data = factor_gpa)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2939 -0.8507  0.1867  0.9839  2.3880
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     4.22067    0.16666  25.325  < 2e-16 ***
## food_choices    0.12323    0.17664   0.698  0.48846
## calories_guess  0.54759    0.19035   2.877  0.00578 **
## life_style     -0.25663    0.18902  -1.358  0.18030
## comfort_food   -0.36102    0.16952  -2.130  0.03786 *
## education       0.04968    0.18943   0.262  0.79412
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.279 on 53 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.2085, Adjusted R-squared:  0.1338
## F-statistic: 2.792 on 5 and 53 DF,  p-value: 0.02605
```