Lesson 11 - Register Variation and PCA/EFA

Erin M. Buchanan

03/28/2019

Language Topics Discussed

- Registers as an extension of dialect and other cultural norms
- The LIWC will be used for the assignment

Registers?

- A register is a language variety associated with the way you are using the language (i.e., email versus face-to-face)
- Contextual factors that change registers:
  - Communication channel: writing, speech
  - Relationship between participants: personal, work, status
  - Communication purpose: social, transfer of information
  - Setting: Private, public

Registers are linguistic

- Registers can also be related to specific linguistic features
  - So we may see more first person pronouns in person than through email
  - We can use clustering to understand these conversations and give them labels without hand coding it

Biber's work

- Biber (1988) used factor analysis to analyze conversations for different register variations
  - Showed dimensions such as:
    - Involved/Informal Production
    - Narrative/Non-narrative contexts

Relation to previous work

- These variations are obviously related to each other, rather than being distinct categories that we can cleanly separate
- This sort of overlap does make defining register difficult, but the general categories remain
- The idea of register is often tied to genre, stylistic choices, jargon, dialect (as defined last week)
- Sometimes defined as pragmatics - the social use of language

Thinking about formality

- Register can be considered a ranking of formality (tenor), if we think about it in a social way
  - Frozen: "static" text, like the Pledge of Allegiance
  - Formal: one way conversation aimed at delivering technical knowledge, like a conference presentation
  - Consultative: two way conversation with some formality usually delivering knowledge, like student/teacher
  - Casual: friends/acquaintances, slang, social settings
  - Intimate: family, close friends, non-verbal messages, non-public

Analyzing register

- We will look at the British National Corpus which has been coded for:
  - 69 observations of 11 variables
  - Things like: `Ncomm`: frequencies of common nouns, `Vpres`: frequencies of third present tense verbs, `P1`: first person pronouns, `ConjCoord`: coordinating conjunctions, etc.

Let's look at the data

```r
library(Rling)
library(psych)
data(reg_bnc)
head(reg_bnc)
```

```
##                  Reg     Ncomm       Nprop      Vpres
## S_brdcst_disc   Spok 0.1696076 0.026968511 0.03550390 0.02
## S_brdcst_doc    Spok 0.2050599 0.024979040 0.03910835 0.02
## S_brdcst_news   Spok 0.2055274 0.046801903 0.03663561 0.02
## S_classroom     Spok 0.1362944 0.011201051 0.04851445 0.01
## S_consult       Spok 0.1327101 0.009851242 0.04519086 0.01
## S_conv          Spok 0.1197967 0.019950370 0.04425219 0.03
##                          P2        Adj   ConjCoord     ConjS
## S_brdcst_disc   0.018323103 0.05357844 0.03949722 0.03104
## S_brdcst_doc    0.011367559 0.05851457 0.03397939 0.02764
## S_brdcst_news   0.007748555 0.05961002 0.03347093 0.02323
## S_classroom     0.037485571 0.04069626 0.03388688 0.03145
## S_consult       0.037029706 0.04460466 0.03840766 0.02828
```

A brief note

- PCA: components are orthogonal (i.e. uncorrelated) linear combinations that maximize capturing the total variance
  - Often used when you want separate distinction solutions, clustering things together
- EFA: factors are linear combinations that maximize the shared portions of the variance
  - Often used when you want to identify the underlying "latent" variables, allowing them to be correlated to each other

Steps to Analysis

- Things to check before you begin
- How many factors or components should you use?
- Simple structure
- Adequate solutions

Before you begin

- Check correlations - you want things to be correlated, `cortest.bartlett`
- Check your sampling adequacy, `KMO`
- At least 3-4 items per grouping
- Interval measurement
- Normal/linear/normal parametric assumptions

Correlations

- If non-significant implies that your items are not correlated enough

```
correlations = cor(reg_bnc[ , -1 ])
cortest.bartlett(correlations, n = nrow(reg_bnc))
```

```
## $chisq
## [1] 536.3401
##
## $p.value
## [1] 0.000000000000000000000000000000000000000000000000000000
##
## $df
## [1] 55
```

Sampling adequacy

- Kaiser-Meyer-Olkin (KMO) test
- Compares the ratio between r2 and pr2
- Scores closer to 1 are better, closer to 0 are bad

Sampling adequacy

```r
KMO(correlations)
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = correlations)
## Overall MSA =  0.72
## MSA for each item =
##     Ncomm     Nprop     Vpres     Vpast        P1
##      0.85      0.61      0.61      0.29      0.86
## ConjCoord   ConjSub Interject       Num
##      0.49      0.72      0.76      0.53
```

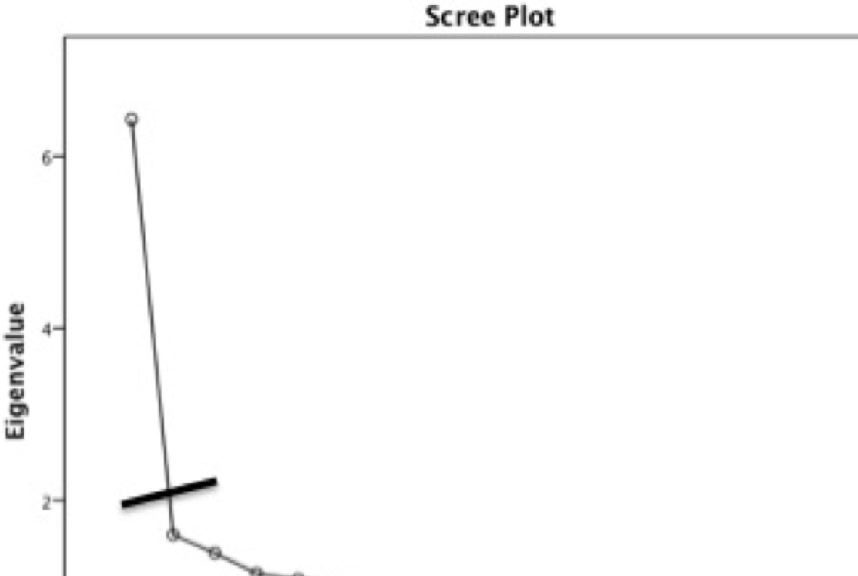
How many factors or components do I have?

- Theory
- Kaiser criterion
- Scree plots
- Parallel analysis

Kaiser criterion

- Old rule: extract the number of eigenvalues over 1
- New rule: extract the number of eigenvalues over .7
- What the heck is an eigenvalue?
  - A mathematical representation of the variance accounted for by that grouping of items

Scree plots

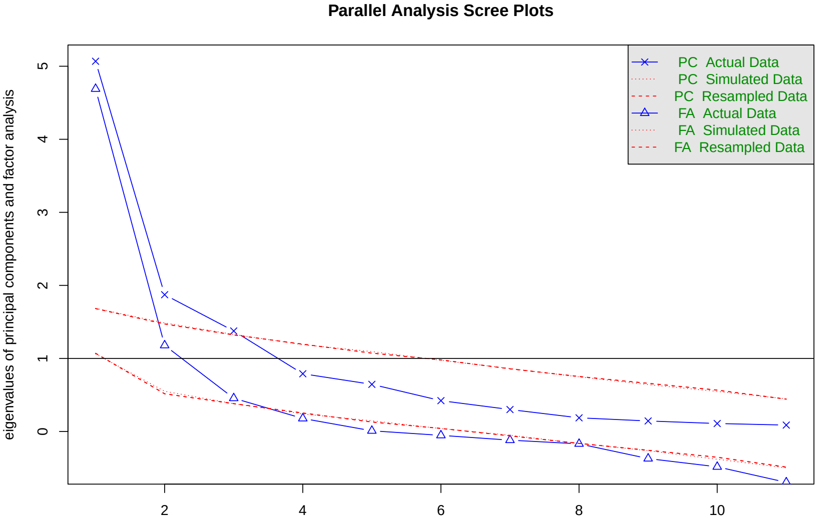


Parallel analysis

- A statistical test to tell you how many eigenvalues are greater than chance
  - Calculates the eigenvalues for your data
  - Randomizes your data and recalculates the eigenvalues
  - Then compares them to determine if they are equal

Finding factors/components

```
number_items = fa.parallel(reg_bnc[, -1], ##dataset
                           fm = "ml", ##type of math
                           fa = "both") #look at both efa/p
```



Eigenvalues

```r
sum(number_items$fa.values > 1)
```

```
## [1] 2
```

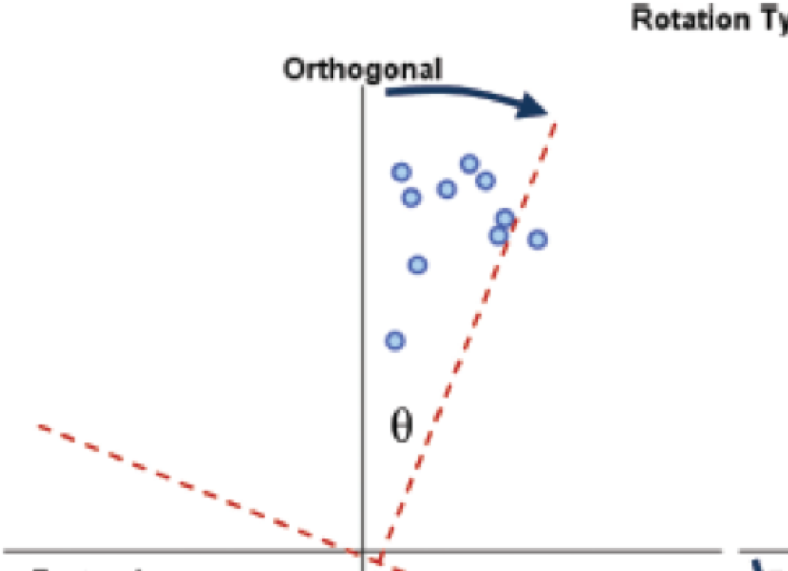
```r
sum(number_items$fa.values > .7)
```

```
## [1] 2
```

Simple structure

- Simple structure covers two pieces:
  - The math used to achieve the solution
    - PCA: principle components
    - EFA: maximum likelihood
  - The rotation to increase communality between items and aid in interpretation (EFA only)

Rotation



Rotation

- Orthogonal assume uncorrelated factors: varimax, quartermax, equamax
- Oblique allows factors to be correlated: oblimin, promax
- Why would we even use orthogonal?

Simple structure/solution

- Looking at the loadings: the relationship between the item and the factor/component
  - Want these to be related at least .3
  - Remember that r = .3 is a medium effect size that is ~10% variance
  - Can eliminate items that load poorly
  - Difference here in scale development versus exploratory clustering

Run a PCA

```
PCA_fit = principal(reg_bnc[, -1], #data
           nfactors = 2, #number of components
           rotate = "none")
```

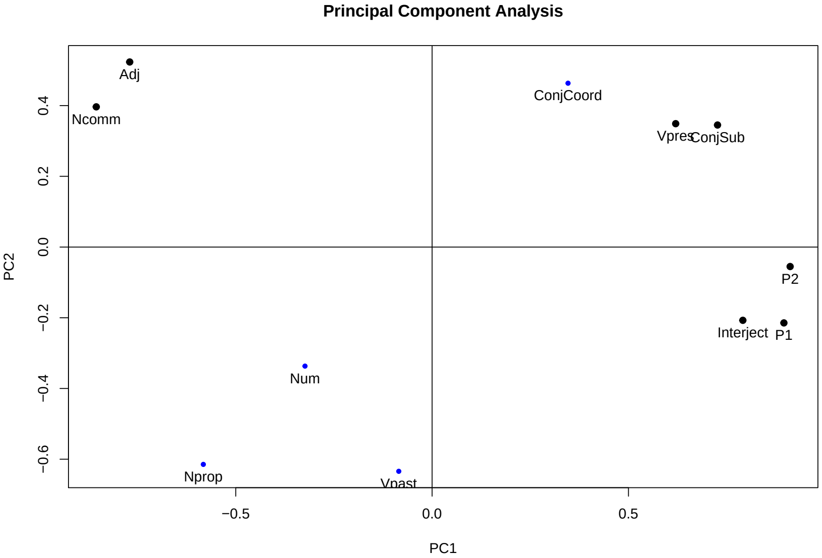
Look at the results

```
PCA_fit$loadings ##view full output in R
```

```
##
## Loadings:
##              PC1    PC2
## Ncomm       -0.855  0.397
## Nprop       -0.583 -0.615
## Vpres        0.620  0.349
## Vpast              -0.634
## P1           0.896 -0.215
## P2           0.912
## Adj         -0.770  0.523
## ConjCoord    0.346  0.463
## ConjSub      0.727  0.345
## Interject    0.791 -0.207
## Num         -0.324 -0.337
##
##                  PC1   PC2
## SS loadings    5.068 1.872
```

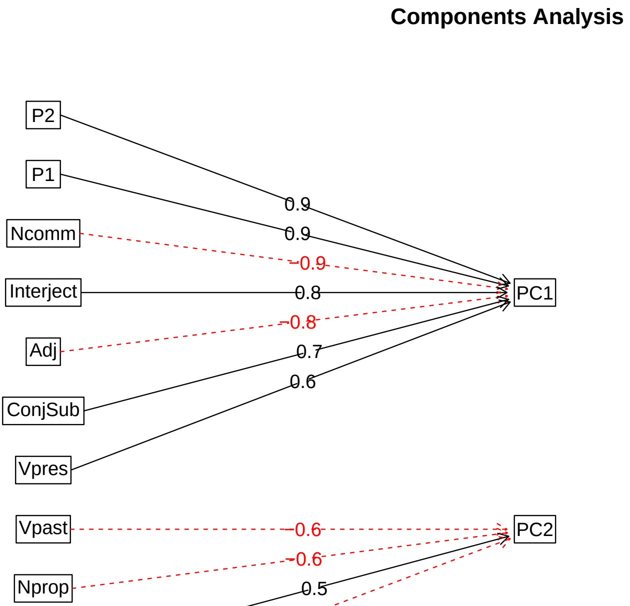
Plots of the results

```
fa.plot(PCA_fit,
     labels = colnames(reg_bnc[ , -1]))
```



Plots of the results

```r
fa.diagram(PCA_fit)
```



Run an EFA

```
EFA_fit = fa(reg_bnc[, -1], #data
             nfactors = 2, #number of factors
             rotate = "oblimin", #rotation
             fm = "ml") #math
```

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

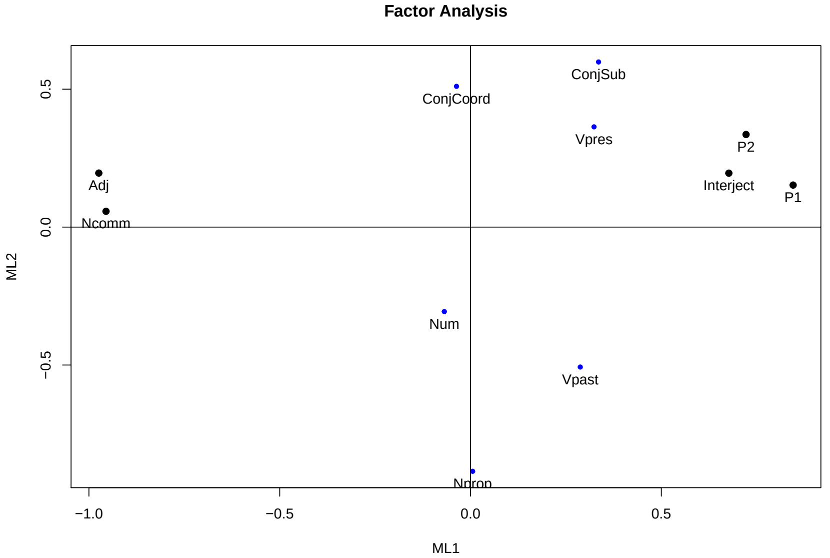
Look at the results

```
EFA_fit$loadings #look at the full results
```

```
##
## Loadings:
##           ML1    ML2
## Ncomm     -0.955
## Nprop            -0.885
## Vpres      0.324  0.363
## Vpast      0.288 -0.507
## P1         0.845  0.152
## P2         0.722  0.336
## Adj       -0.974  0.196
## ConjCoord         0.510
## ConjSub    0.336  0.599
## Interject  0.677  0.196
## Num              -0.306
##
##                  ML1   ML2
## SS loadings    3.862 2.101
```

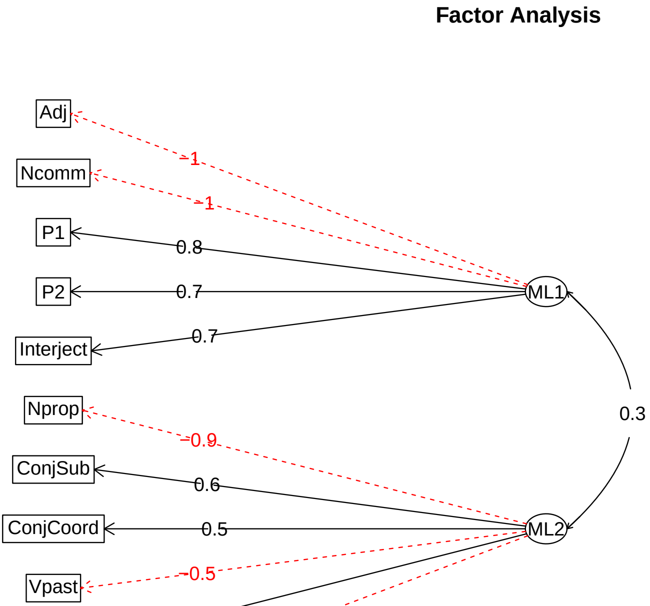
Plots of the results

```
fa.plot(EFA_fit,
    labels = colnames(reg_bnc[ , -1]))
```



Plots of the results

```
fa.diagram(EFA_fit)
```



Adequate solution

- Fit indices: a measure of how well the model matches the data
  - Goodness of fit statistics: measure the overlap between the reproduced correlation matrix and the original, want high numbers close to 1
  - Badness of fit statistics (residual): measure the mismatch, want low numbers close to zero
- Theory/interpretability

Fit statistics

```
PCA_fit$rms #Root mean square of the residuals
```

```
## [1] 0.105979
```

```
EFA_fit$rms
```

```
## [1] 0.08936898
```

```
EFA_fit$RMSEA #root mean squared error of approximation
```

```
##      RMSEA      lower      upper confidence
##  0.1894160  0.1402170  0.2161319  0.9000000
```

```
EFA_fit$TLI #tucker lewis index
```

```
## [1] 0.7490712
```

Summary

- You learned about registers and how they can be used to classify language
- You learned how to examine how these group together into orthogonal components versus latent factors
- You learned about EFA and PCA and the steps to running them