EFA

```r
set.seed(42)

library(igraph)
```

```
##
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats':
##
##     decompose, spectrum

## The following object is masked from 'package:base':
##
##     union
```

```r
library(QuantPsyc) # for the multivariate normality test
```

```
## Loading required package: boot

## Loading required package: dplyr

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:igraph':
##
##     as_data_frame, groups, union

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

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

## Loading required package: purrr

##
## Attaching package: 'purrr'

## The following objects are masked from 'package:igraph':
##
##     compose, simplify

## Loading required package: MASS

##
## Attaching package: 'MASS'

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

```
##
## Attaching package: 'QuantPsyc'

## The following object is masked from 'package:base':
##
##     norm
```

```r
library(nFactors) # for the scree plot
```

```
## 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
```

```r
library(psych) # for PA FA
```

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

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

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v readr     2.1.5

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x lubridate::%--%()       masks igraph::%--%()
## x ggplot2::%+%()          masks psych::%+%()
## x ggplot2::alpha()        masks psych::alpha()
## x tibble::as_data_frame() masks dplyr::as_data_frame(), igraph::as_data_frame()
## x purrr::compose()        masks igraph::compose()
## x tidyr::crossing()       masks igraph::crossing()
## x dplyr::filter()         masks stats::filter()
## x dplyr::lag()            masks stats::lag()
## x MASS::select()          masks dplyr::select()
## x purrr::simplify()       masks igraph::simplify()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```r
library(paletteer) # color palettes

library(conflicted) # to resolve QuantPsyc x dplyr conflicts
conflict_prefer("select", "dplyr")
```

```
## [conflicted] Will prefer dplyr::select over any other package.
```

```r
conflict_prefer("filter", "dplyr")
```

```
## [conflicted] Will prefer dplyr::filter over any other package.
```

Load and tidy data

```r
pretty_names <- read_csv("../feat_name_mapping.csv")
```

```
## Rows: 85 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): name_orig, name_pretty
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
data <- read_csv("../measurements/measurements.csv")
```

```
## Rows: 754 Columns: 108
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (20): fpath, KUK_ID, FileName, FileFormat, FolderPath, subcorpus, Source...
## dbl (85): RuleAbstractNouns, RuleAmbiguousRegards, RuleAnaphoricReferences, ...
## lgl  (3): ClarityPursuit, SyllogismBased, Bindingness
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
.firstnonmetacolumn <- 17

data_clean <- data %>%
  select(!c(
    fpath,
    # KUK_ID,
    # FileName,
    FolderPath,
    # subcorpus,
    DocumentTitle,
    ClarityPursuit,
    Readability,
    SyllogismBased,
    SourceDB
  )) %>%
  # replace -1s in variation coefficients with NAs
  mutate(across(c(
    `RuleDoubleAdpos.max_allowable_distance.v`,
    `RuleTooManyNegations.max_negation_frac.v`,
    `RuleTooManyNegations.max_allowable_negations.v`,
    `RuleTooManyNominalConstructions.max_noun_frac.v`,
    `RuleTooManyNominalConstructions.max_allowable_nouns.v`,
    `RuleCaseRepetition.max_repetition_count.v`,
    `RuleCaseRepetition.max_repetition_frac.v`,
    `RulePredSubjDistance.max_distance.v`,
```

```r
    `RulePredObjDistance.max_distance.v`,
    `RuleInfVerbDistance.max_distance.v`,
    `RuleMultiPartVerbs.max_distance.v`,
    `RuleLongSentences.max_length.v`,
    `RulePredAtClauseBeginning.max_order.v`,
    `mattr.v`,
    `maentropy.v`
  ), ~ na_if(.x, -1))) %>%
  # replace NAs with 0s
  replace_na(list(
    RuleGPcoordovs = 0,
    RuleGPdeverbaddr = 0,
    RuleGPpatinstr = 0,
    RuleGPdeverbsubj = 0,
    RuleGPadjective = 0,
    RuleGPpatbenperson = 0,
    RuleGPwordorder = 0,
    RuleDoubleAdpos = 0,
    RuleDoubleAdpos.max_allowable_distance = 0,
    RuleDoubleAdpos.max_allowable_distance.v = 0,
    RuleAmbiguousRegards = 0,
    RuleReflexivePassWithAnimSubj = 0,
    RuleTooManyNegations = 0,
    RuleTooManyNegations.max_negation_frac = 0,
    RuleTooManyNegations.max_negation_frac.v = 0,
    RuleTooManyNegations.max_allowable_negations = 0,
    RuleTooManyNegations.max_allowable_negations.v = 0,
    RuleTooManyNominalConstructions.max_noun_frac.v = 0,
    RuleTooManyNominalConstructions.max_allowable_nouns.v = 0,
    RuleFunctionWordRepetition = 0,
    RuleCaseRepetition.max_repetition_count.v = 0,
    RuleCaseRepetition.max_repetition_frac.v = 0,
    RuleWeakMeaningWords = 0,
    RuleAbstractNouns = 0,
    RuleRelativisticExpressions = 0,
    RuleConfirmationExpressions = 0,
    RuleRedundantExpressions = 0,
    RuleTooLongExpressions = 0,
    RuleAnaphoricReferences = 0,
    RuleLiteraryStyle = 0,
    RulePassive = 0,
    RulePredSubjDistance = 0,
    RulePredSubjDistance.max_distance = 0,
    RulePredSubjDistance.max_distance.v = 0,
    RulePredObjDistance = 0,
    RulePredObjDistance.max_distance = 0,
    RulePredObjDistance.max_distance.v = 0,
    RuleInfVerbDistance = 0,
    RuleInfVerbDistance.max_distance = 0,
    RuleInfVerbDistance.max_distance.v = 0,
    RuleMultiPartVerbs = 0,
    RuleMultiPartVerbs.max_distance = 0,
    RuleMultiPartVerbs.max_distance.v = 0,
```

```
  RuleLongSentences.max_length.v = 0,
  RulePredAtClauseBeginning.max_order.v = 0,
  RuleVerbalNouns = 0,
  RuleDoubleComparison = 0,
  RuleWrongValencyCase = 0,
  RuleWrongVerbonominalCase = 0,
  RuleIncompleteConjunction = 0
)) %>%
# norm data expected to correlate with text length
mutate(across(c(
  RuleGPcoordovs,
  RuleGPdeverbaddr,
  RuleGPpatinstr,
  RuleGPdeverbsubj,
  RuleGPadjective,
  RuleGPpatbenperson,
  RuleGPwordorder,
  RuleDoubleAdpos,
  RuleAmbiguousRegards,
  RuleFunctionWordRepetition,
  RuleWeakMeaningWords,
  RuleAbstractNouns,
  RuleRelativisticExpressions,
  RuleConfirmationExpressions,
  RuleRedundantExpressions,
  RuleTooLongExpressions,
  RuleAnaphoricReferences,
  RuleLiteraryStyle,
  RulePassive,
  RuleVerbalNouns,
  RuleDoubleComparison,
  RuleWrongValencyCase,
  RuleWrongVerbonominalCase,
  RuleIncompleteConjunction,
  num_hapax,
  RuleReflexivePassWithAnimSubj,
  RuleTooManyNominalConstructions,
  RulePredSubjDistance,
  RuleMultiPartVerbs,
  RulePredAtClauseBeginning
), ~ .x / word_count)) %>%
mutate(across(c(
  RuleTooFewVerbs,
  RuleTooManyNegations,
  RuleCaseRepetition,
  RuleLongSentences,
  RulePredObjDistance,
  RuleInfVerbDistance
), ~ .x / sent_count)) %>%
# remove variables identified as "u counts"
select(!c(
  RuleTooFewVerbs,
  RuleTooManyNegations,
```

```r
  RuleTooManyNominalConstructions,
  RuleCaseRepetition,
  RuleLongSentences,
  RulePredAtClauseBeginning,
  sent_count,
  word_count,
  syllab_count,
  char_count
)) %>%
# remove variables identified as unreliable
select(!c(
  RuleAmbiguousRegards,
  RuleFunctionWordRepetition,
  RuleDoubleComparison,
  RuleWrongValencyCase,
  RuleWrongVerbonominalCase
)) %>%
# remove artificially limited variables
select(!c(
  RuleCaseRepetition.max_repetition_frac,
  RuleCaseRepetition.max_repetition_frac.v
)) %>%
# remove further variables belonging to the 'acceptability' category
select(!c(RuleIncompleteConjunction)) %>%
# # remove variation coefficients theoretically coinciding with their means too strongly
# select(!c(
#   RuleDoubleAdpos.max_allowable_distance.v,
#   RuleTooManyNegations.max_negation_frac.v,
#   RuleTooManyNegations.max_allowable_negations.v
# )) %>%
# remove features expected to have low communalities
select(!c(
  RuleDoubleAdpos.max_allowable_distance,
  RuleDoubleAdpos.max_allowable_distance.v,
  RuleGPwordorder,
  RuleLiteraryStyle,
  maentropy.v,
  RuleTooManyNegations.max_negation_frac,
  RulePredSubjDistance.max_distance,
  RuleTooManyNegations.max_allowable_negations,
  RuleTooManyNegations.max_allowable_negations.v,
  RuleTooManyNominalConstructions.max_allowable_nouns.v,
  RuleTooFewVerbs.min_verb_frac.v,
  RulePredObjDistance.max_distance.v,
  RulePredObjDistance.max_distance,
  # RuleInfVerbDistance.max_distance,
  RulePredAtClauseBeginning.max_order.v,
  RuleInfVerbDistance
  # RulePredSubjDistance
)) %>%
# remove features expected to have low loadings
select(!c(
  RuleMultiPartVerbs.max_distance.v,
```

```r
    RulePredSubjDistance.max_distance.v,
    RuleLongSentences.max_length
  )) %>%
  mutate(across(c(class), ~ as.factor(.x)))

# no NAs should be present now
data_clean[!complete.cases(data_clean), ]
```

```
## # A tibble: 754 x 65
##    KUK_ID                 FileName FileFormat subcorpus SourceID DocumentVersion
##    <chr>                  <chr>    <chr>      <chr>     <chr>    <chr>
##  1 673b7a37c6537d54ff062~ 002_Kom~ TXT        KUKY      <NA>     Original
##  2 673b7a37c6537d54ff062~ 006_Chc~ TXT        KUKY      <NA>     Redesign
##  3 673b7a37c6537d54ff062~ 004_Nev~ TXT        KUKY      <NA>     Original
##  4 673b7a37c6537d54ff062~ 008_Pol~ TXT        KUKY      <NA>     Original
##  5 673b7a37c6537d54ff062~ 005_Och~ TXT        KUKY      <NA>     Original
##  6 673b7a37c6537d54ff062~ 016_Obc~ TXT        KUKY      <NA>     Original
##  7 673b7a37c6537d54ff062~ 019_Dět~ TXT        KUKY      <NA>     Redesign
##  8 673b7a37c6537d54ff062~ 007_DŮC~ TXT        KUKY      <NA>     Redesign
##  9 673b7a37c6537d54ff062~ 024_Opa~ TXT        KUKY      <NA>     Original
## 10 673b7a37c6537d54ff062~ 047_Dav~ TXT        KUKY      <NA>     Original
## # i 744 more rows
## # i 59 more variables: ParentDocumentID <chr>, LegalActType <chr>,
## #   Objectivity <chr>, Bindingness <lgl>, AuthorType <chr>,
## #   RecipientType <chr>, RecipientIndividuation <chr>, Anonymized <chr>,
## #   `Recipient Type` <chr>, class <fct>, RuleAbstractNouns <dbl>,
## #   RuleAnaphoricReferences <dbl>,
## #   RuleCaseRepetition.max_repetition_count <dbl>, ...
```

```r
data_clean_scaled <- data_clean %>%
  mutate(across(class, ~ .x == "good")) %>%
  mutate(across(.firstnonmetacolumn:length(names(data_clean)), ~ scale(.x)))
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `across(.firstnonmetacolumn:length(names(data_clean)),
##   ~scale(.x))`.
## Caused by warning:
## ! Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(.firstnonmetacolumn)
##
##   # Now:
##   data %>% select(all_of(.firstnonmetacolumn))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
```

Important features identification

```r
data_clean_good <- data_clean_scaled %>% filter(class == "good")
data_clean_bad <- data_clean_scaled %>% filter(class == "bad")

feature_importances <- tibble(
```

```
  feat_name = character(), p_value = numeric()
)

for (i in .firstnonmetacolumn:ncol(data_clean)) {
  fname <- names(data_clean)[i]

  formula_single <- reformulate(fname, "class")

  glm_model <- glm(formula_single, data_clean, family = "binomial")
  glm_coefficients <- summary(glm_model)$coefficients
  row_index <- which(rownames(glm_coefficients) == fname)
  p_value <- glm_coefficients[row_index, 4]

  feature_importances <- feature_importances %>%
    add_row(feat_name = fname, p_value = p_value)
}
feature_importances
```

```
## # A tibble: 49 x 2
##    feat_name                               p_value
##    <chr>                                     <dbl>
##  1 RuleAbstractNouns                       0.00187
##  2 RuleAnaphoricReferences                 0.660
##  3 RuleCaseRepetition.max_repetition_count   0.0722
##  4 RuleCaseRepetition.max_repetition_count.v 0.00479
##  5 RuleConfirmationExpressions             0.0985
##  6 RuleDoubleAdpos                         0.312
##  7 RuleGPadjective                         0.380
##  8 RuleGPcoordovs                          0.828
##  9 RuleGPdeverbaddr                        0.0112
## 10 RuleGPdeverbsubj                        0.0133
## # i 39 more rows
```

```
selected_features <- feature_importances %>%
  filter(p_value <= 0.05) %>%
  pull(feat_name)
```

Correlations

See Levshina (2015: 353–54).

```
analyze_correlation <- function(data) {
  cor_matrix <- cor(data)

  cor_tibble_long <- cor_matrix %>%
    as_tibble() %>%
    mutate(feat1 = rownames(cor_matrix)) %>%
    pivot_longer(!feat1, names_to = "feat2", values_to = "cor") %>%
    mutate(abs_cor = abs(cor))

  cor_matrix_upper <- cor_matrix
  cor_matrix_upper[lower.tri(cor_matrix_upper)] <- 0

  cor_tibble_long_upper <- cor_matrix_upper %>%
```

```r
    as_tibble() %>%
    mutate(feat1 = rownames(cor_matrix)) %>%
    pivot_longer(!feat1, names_to = "feat2", values_to = "cor") %>%
    mutate(abs_cor = abs(cor)) %>%
    filter(feat1 != feat2 & abs_cor > 0)

  list(
    cor_matrix = cor_matrix,
    cor_matrix_upper = cor_matrix_upper,
    cor_tibble_long = cor_tibble_long,
    cor_tibble_long_upper = cor_tibble_long_upper
  )
}

data_purish <- data_clean %>% select(any_of(selected_features))
```

Extremely non-normal data

```r
# # remove where median == 0?
# keep <- character()
# for (i in seq_along(colnames(data_purish))) {
#   cname <- colnames(data_purish)[i]
#   q <- quantile(data_purish[, i][[1]], probs = 0.10)[[1]]
#   if (q > 0) {
#     keep <- c(keep, cname)
#     cat("keep", cname, "\n")
#   } else {
#     cat("throw out", cname, "\n")
#   }
# }
# data_purish <- data_purish %>% select(any_of(keep))
```

High correlations

```r
.hcorrcutoff <- 0.9

analyze_correlation(data_purish)$cor_tibble_long %>%
  filter(feat1 != feat2 & abs_cor > .hcorrcutoff) %>%
  arrange(feat1, -abs_cor) %>%
  print(n = 100)
```

```
## # A tibble: 16 x 4
##    feat1     feat2       cor abs_cor
##    <chr>     <chr>     <dbl>   <dbl>
##  1 ari       fkgl      0.984   0.984
##  2 ari       gf        0.978   0.978
##  3 ari       smog      0.951   0.951
##  4 atl       cli       0.960   0.960
##  5 cli       atl       0.960   0.960
##  6 fkgl      ari       0.984   0.984
##  7 fkgl      gf        0.967   0.967
##  8 fkgl      smog      0.949   0.949
##  9 gf        smog      0.987   0.987
```

```
## 10 gf        ari       0.978   0.978
## 11 gf        fkgl      0.967   0.967
## 12 maentropy mattr     0.964   0.964
## 13 mattr     maentropy 0.964   0.964
## 14 smog      gf        0.987   0.987
## 15 smog      ari       0.951   0.951
## 16 smog      fkgl      0.949   0.949
```

exclude:

- **ari:** corr. w/ RuleLongSentences.max_length > 0.94; sentence length seems more universal, let's make it a substitute
- **gf:** corr. w/ RuleLongSentences.max_length > 0.92; sentence length seems more universal, let's make it a substitute
- **maentropy:** corr. w/ mattr > 0.96, but mattr is implemented in QuitaUp. besides, the interesting thing about maentropy is its variation
- **smog:** corr. w/ fkgl almost 0.95, but fkgl coefficients adjusted for Czech are available
- **atl:** corr. w/ cli around 0.96; unlike cli, atl is not a readability metric

```
data_pureish_striphigh <- data_purish %>% select(!c(
  ari, gf, maentropy, smog, atl
))

analyze_correlation(data_pureish_striphigh)$cor_tibble_long %>%
  filter(feat1 != feat2 & abs_cor > .hcorrcutoff) %>%
  arrange(feat1, -abs_cor) %>%
  print(n = 100)
```

```
## # A tibble: 0 x 4
## # i 4 variables: feat1 <chr>, feat2 <chr>, cor <dbl>, abs_cor <dbl>
```

Low correlations

```
# 0.35 instead of 0.3 otherwise the FA bootstrapping would freeze
.lcorrcutoff <- 0.35

low_correlating_features <- analyze_correlation(data_pureish_striphigh)$
  cor_tibble_long %>%
  filter(feat1 != feat2) %>%
  group_by(feat1) %>%
  summarize(max_cor = max(abs_cor)) %>%
  filter(max_cor < .lcorrcutoff) %>%
  pull(feat1)

feature_importances %>% filter(feat_name %in% low_correlating_features)
```

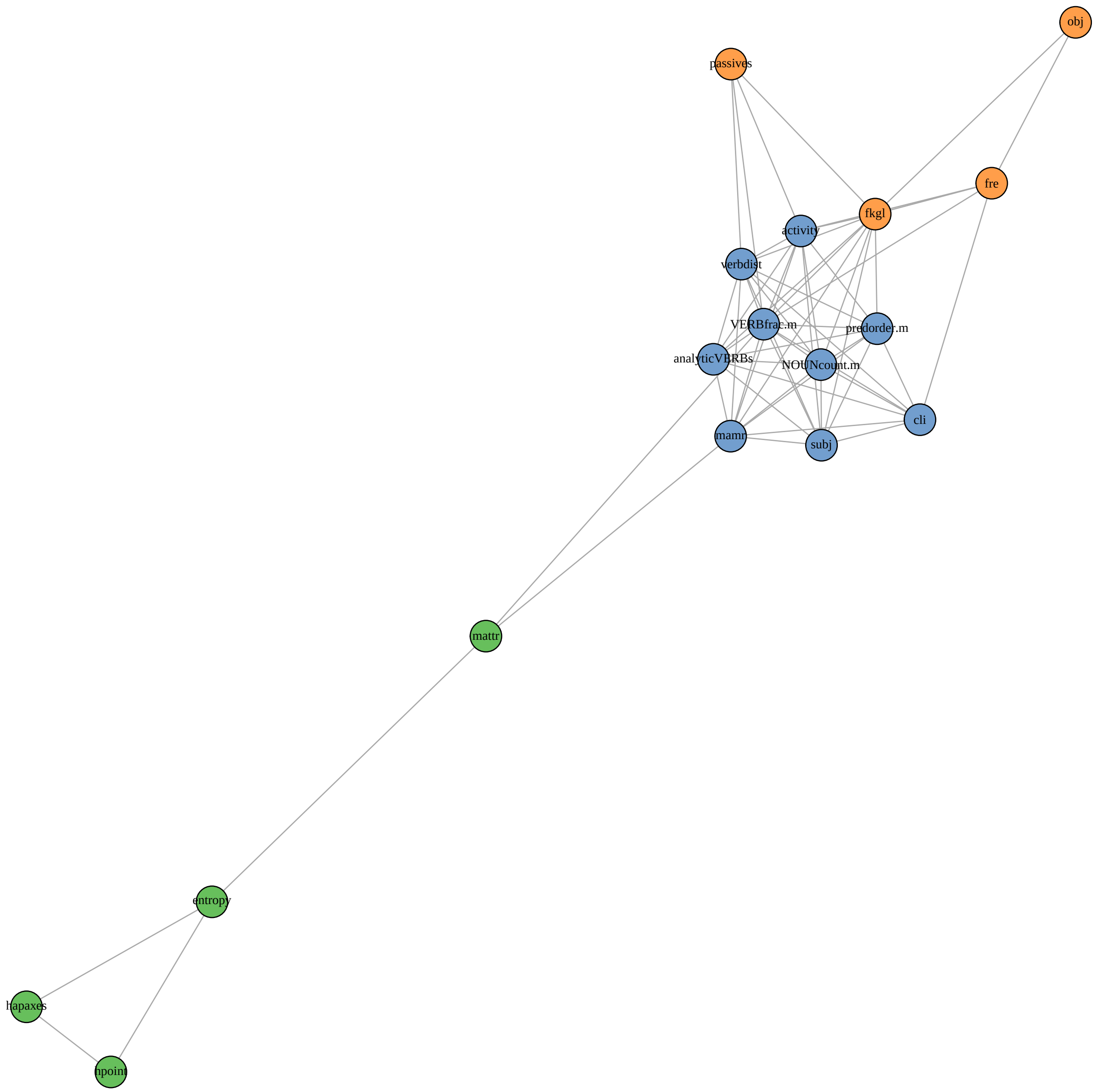
```
## # A tibble: 11 x 2
##    feat_name                                     p_value
##    <chr>                                           <dbl>
##  1 RuleAbstractNouns                             0.00187
##  2 RuleCaseRepetition.max_repetition_count.v     0.00479
##  3 RuleGPdeverbaddr                              0.0112
##  4 RuleGPdeverbsubj                              0.0133
##  5 RuleMultiPartVerbs.max_distance               0.00320
##  6 RuleRedundantExpressions                      0.0104
##  7 RuleRelativisticExpressions                   0.00205
```

```
##   8 RuleTooManyNegations.max_negation_frac.v        0.0365
##   9 RuleTooManyNominalConstructions.max_noun_frac.v 0.00000311
## 10 RuleVerbalNouns                                  0.0000748
## 11 RuleWeakMeaningWords                             0.0386
```

```
data_pure <- data_pureish_striphigh %>%
  select(!any_of(low_correlating_features))

cnames <- map(
  colnames(data_pure),
  function(x) {
    pull(pretty_names %>%
      filter(name_orig == x), name_pretty)
  }
) %>% unlist()

colnames(data_pure) <- cnames
```

Visualisation

```
my_colors <- paletteer::paletteer_d("ggthemes::Classic_10_Medium")

network_edges <- analyze_correlation(data_pure)$cor_tibble_long_upper %>%
  filter(abs_cor > 0.3)

network <- graph_from_data_frame(
  network_edges,
  directed = FALSE
)
E(network)$weight <- network_edges$abs_cor
network_communities <- cluster_optimal(network)

network_membership <- membership(network_communities)

plot(
  network,
  layout = layout.fruchterman.reingold,
  vertex.color = map(
    network_communities$membership,
    function(x) my_colors[x]
  ) %>% unlist(use.names = FALSE),
  vertex.size = 6,
  vertex.label.color = "black",
  vertex.label.cex = 0.7
)
```



Scaling

```
data_scaled <- data_pure %>%
  mutate(across(1:length(colnames(data_pure)), ~ scale(.x)[, 1]))
```

Check for normality

```
mult.norm(data_scaled %>% as.data.frame())$mult.test
```

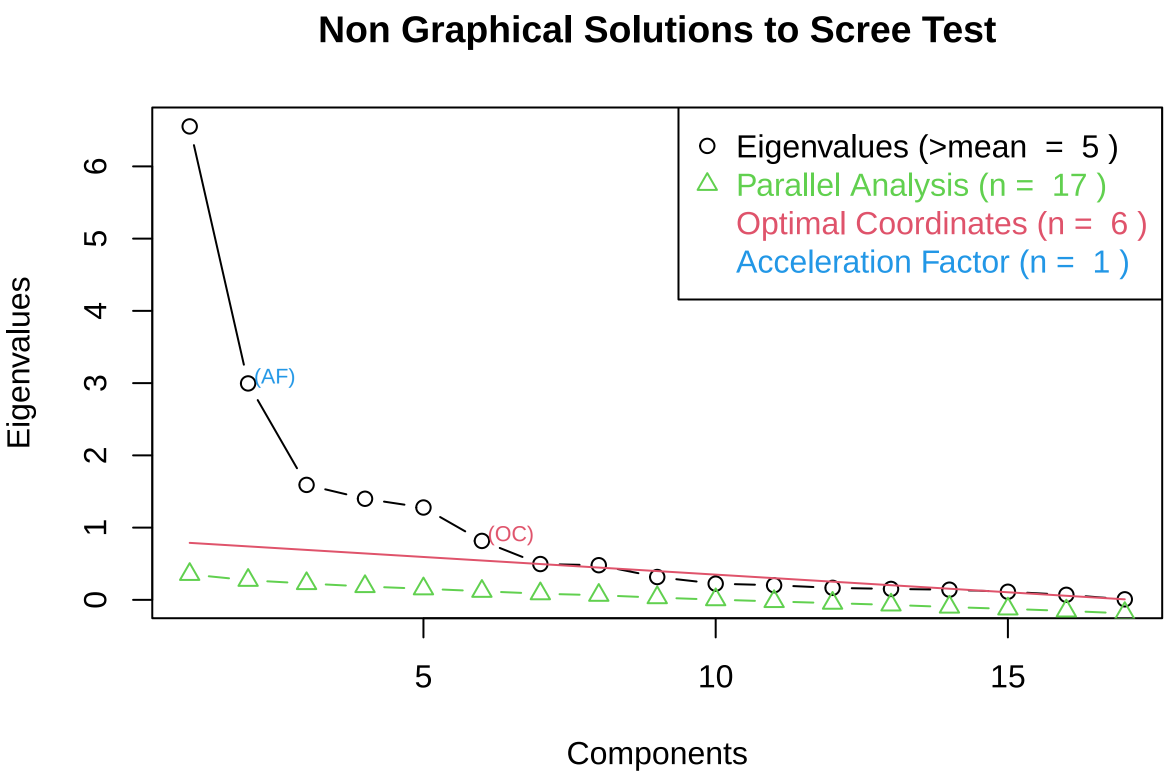
```
##          Beta-hat      kappa p-val
## Skewness 351.5182 44174.1153     0
## Kurtosis 858.5678   289.3036     0
```

Low (null) p-values show that we can reject the hypothesis that the data would be in a multivariate normal distribution. I.e. the distribution isn't multivariate normal.

FA

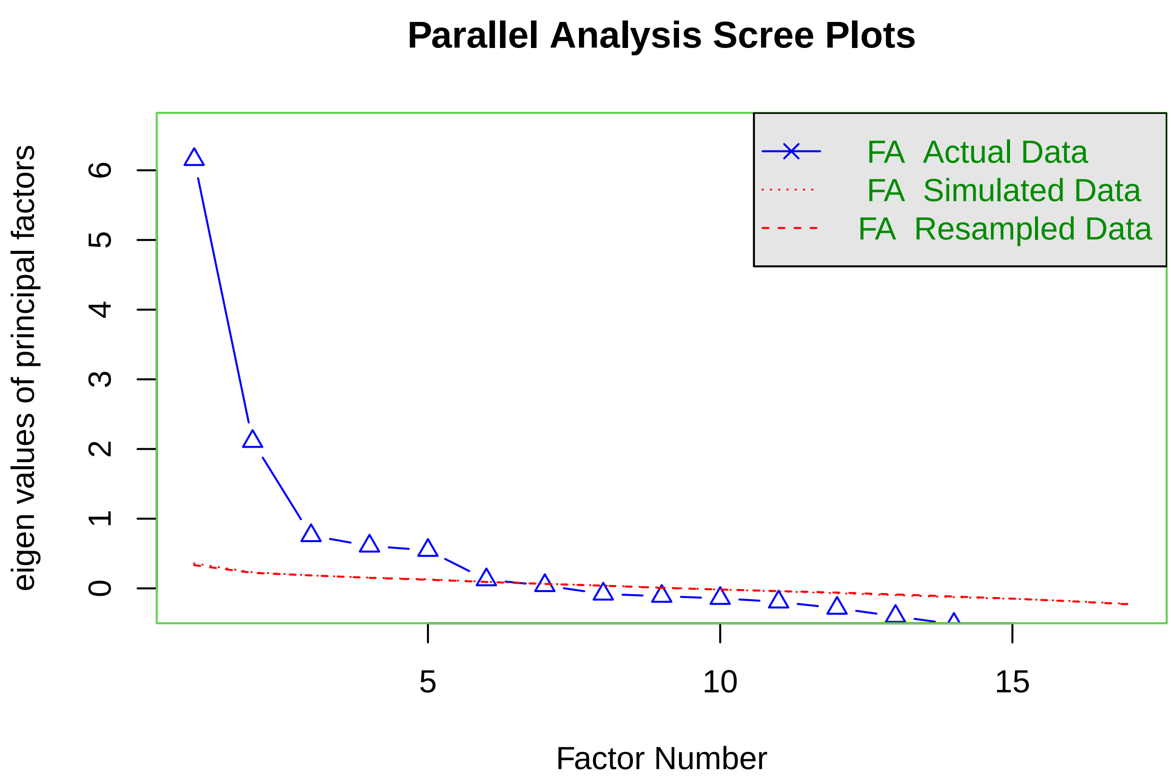
No. of factors

```
eigen <- eigen(cor(data_scaled))
par <- nFactors::parallel(
  subject = nrow(data_scaled),
  var = ncol(data_scaled),
  rep = 100,
  quantile = .95,
  model = "factors"
)
scree <- nScree(x = eigen$values, aparallel = par$eigen$qevpea)
plotnScree(scree)
```




```
fa.parallel(data_scaled, fm = "pa", fa = "fa", n.iter = 20)
```

```
## Warning in fa.stats(r = r, f = f, phi = phi, n.obs = n.obs, np.obs = np.obs, :
## The estimated weights for the factor scores are probably incorrect.  Try a
## different factor score estimation method.
```




```
## Parallel analysis suggests that the number of factors =  6  and the number of components =  NA
```

Model

https://www.rdocumentation.org/packages/psych/versions/2.5.3/topics/fa

```r
# appears to be the happiest when nfactors = 6 or 7
# throws the The estimated weights for the factor scores are probably incorrect.
# Try a different factor score estimation method. warning otherwise
fa_res <- fa(
  data_scaled,
  nfactors = 6,
  fm = "pa",
  rotate = "promax",
  oblique.scores = TRUE,
  scores = "tenBerge",
  n.iter = 20
)
```

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

```r
fa_res
```

```
## Factor Analysis with confidence intervals using method = fa(r = data_scaled, nfactors = 6, n.iter = 2
##     scores = "tenBerge", fm = "pa", oblique.scores = TRUE)
## Factor Analysis using method =  pa
## Call: fa(r = data_scaled, nfactors = 6, n.iter = 20, rotate = "promax",
##     scores = "tenBerge", fm = "pa", oblique.scores = TRUE)
## Standardized loadings (pattern matrix) based upon correlation matrix
##                   PA1   PA2   PA6   PA3   PA5   PA4   h2     u2 com
## analyticVERBs  0.86 -0.06 -0.06  0.05  0.36  0.02 0.64 0.3600 1.4
## passives       0.12 -0.02 -0.02 -0.20  0.90  0.04 0.63 0.3716 1.1
```

```
## predorder.m   -0.68 -0.05  0.17 -0.05  0.01 -0.11 0.54 0.4609 1.2
## obj            0.19  0.01  0.91 -0.17 -0.10 -0.02 0.68 0.3200 1.2
## subj           0.68  0.12 -0.02  0.07  0.13 -0.15 0.51 0.4854 1.3
## VERBfrac.m     0.82 -0.06  0.03  0.01 -0.21 -0.03 0.87 0.1295 1.2
## NOUNcount.m   -1.04  0.04 -0.13  0.13 -0.12 -0.07 0.86 0.1438 1.1
## activity       0.78 -0.04  0.20 -0.12 -0.35 -0.03 0.89 0.1076 1.6
## cli            0.18 -0.03 -0.14  0.96 -0.28  0.08 0.91 0.0927 1.3
## entropy        0.12  0.74 -0.04  0.05  0.01  0.54 0.95 0.0482 1.9
## fkgl          -0.37  0.02  0.63  0.09  0.25  0.04 1.00 0.0046 2.0
## fre            0.13 -0.01 -0.57 -0.55 -0.12 -0.04 0.98 0.0224 2.2
## hpoint         0.01  0.94  0.01 -0.02 -0.02 -0.02 0.87 0.1325 1.0
## mamr           0.68 -0.05 -0.05  0.22  0.02 -0.32 0.75 0.2484 1.7
## mattr         -0.06 -0.12 -0.01  0.08  0.05  0.83 0.72 0.2769 1.1
## hapaxes        0.08 -0.93 -0.03  0.03  0.02  0.29 0.86 0.1441 1.2
## verbdist      -0.87 -0.01 -0.21 -0.06  0.16 -0.11 0.79 0.2101 1.2
##
##                        PA1  PA2  PA6  PA3  PA5  PA4
## SS loadings           5.53 2.33 1.70 1.33 1.29 1.26
## Proportion Var        0.33 0.14 0.10 0.08 0.08 0.07
## Cumulative Var        0.33 0.46 0.56 0.64 0.72 0.79
## Proportion Explained  0.41 0.17 0.13 0.10 0.10 0.09
## Cumulative Proportion 0.41 0.59 0.71 0.81 0.91 1.00
##
##  With factor correlations of
##       PA1  PA2   PA6  PA3   PA5   PA4
## PA1  1.00 0.02 -0.35 0.08 -0.44 -0.28
## PA2  0.02 1.00  0.29 0.16  0.16  0.17
## PA6 -0.35 0.29  1.00 0.26  0.25  0.14
## PA3  0.08 0.16  0.26 1.00  0.36  0.10
## PA5 -0.44 0.16  0.25 0.36  1.00  0.10
## PA4 -0.28 0.17  0.14 0.10  0.10  1.00
##
## Mean item complexity =  1.4
## Test of the hypothesis that 6 factors are sufficient.
##
## df null model =  136  with the objective function =  17.23 with Chi Square =  12859.06
## df of  the model are 49  and the objective function was  0.93
##
## The root mean square of the residuals (RMSR) is  0.01
## The df corrected root mean square of the residuals is  0.02
##
## The harmonic n.obs is  754 with the empirical chi square  43.8  with prob <  0.68
## The total n.obs was  754  with Likelihood Chi Square =  692.45  with prob <  1.7e-114
##
## Tucker Lewis Index of factoring reliability =  0.859
## RMSEA index =  0.132  and the 90 % confidence intervals are  0.123 0.141
## BIC =  367.81
## Fit based upon off diagonal values = 1
## Measures of factor score adequacy
##                                                    PA1  PA2  PA6  PA3  PA5  PA4
## Correlation of (regression) scores with factors   0.99 0.98 0.99 0.98 0.92 0.95
## Multiple R square of scores with factors          0.97 0.96 0.98 0.97 0.85 0.90
## Minimum correlation of possible factor scores     0.95 0.91 0.96 0.93 0.70 0.79
##
```

```
##  Coefficients and bootstrapped confidence intervals
##                low   PA1 upper   low   PA2 upper   low   PA6 upper   low   PA3
## analyticVERBs  0.81  0.86  0.93 -0.10 -0.06 -0.04 -0.10 -0.06  0.00 -0.01  0.05
## passives       0.08  0.12  0.16 -0.05 -0.02  0.01 -0.06 -0.02  0.02 -0.24 -0.20
## predorder.m   -0.73 -0.68 -0.61 -0.12 -0.05  0.00  0.11  0.17  0.27 -0.19 -0.05
## obj            0.13  0.19  0.25 -0.01  0.01  0.05  0.84  0.91  0.95 -0.23 -0.17
## subj           0.58  0.68  0.76  0.07  0.12  0.20 -0.09 -0.02  0.05  0.02  0.07
## VERBfrac.m     0.75  0.82  0.88 -0.10 -0.06 -0.03  0.00  0.03  0.08 -0.04  0.01
## NOUNcount.m   -1.07 -1.04 -1.00  0.01  0.04  0.07 -0.16 -0.13 -0.09  0.09  0.13
## activity       0.75  0.78  0.82 -0.06 -0.04 -0.01  0.17  0.20  0.24 -0.16 -0.12
## cli            0.16  0.18  0.23 -0.06 -0.03  0.00 -0.18 -0.14 -0.10  0.91  0.96
## entropy        0.09  0.12  0.16  0.70  0.74  0.77 -0.06 -0.04 -0.01  0.01  0.05
## fkgl          -0.42 -0.37 -0.32 -0.01  0.02  0.04  0.55  0.63  0.70  0.05  0.09
## fre            0.10  0.13  0.17 -0.03 -0.01  0.01 -0.65 -0.57 -0.50 -0.63 -0.55
## hpoint        -0.03  0.01  0.05  0.91  0.94  0.96 -0.02  0.01  0.04 -0.05 -0.02
## mamr           0.62  0.68  0.72 -0.09 -0.05 -0.01 -0.08 -0.05 -0.02  0.16  0.22
## mattr         -0.10 -0.06 -0.03 -0.15 -0.12 -0.09 -0.03 -0.01  0.03  0.05  0.08
## hapaxes        0.05  0.08  0.12 -0.96 -0.93 -0.90 -0.07 -0.03  0.00  0.00  0.03
## verbdist      -0.94 -0.87 -0.83 -0.05 -0.01  0.03 -0.27 -0.21 -0.14 -0.12 -0.06
##               upper   low   PA5 upper   low   PA4 upper
## analyticVERBs  0.12  0.30  0.36  0.44 -0.04  0.02  0.08
## passives      -0.14  0.84  0.90  0.94  0.02  0.04  0.07
## predorder.m    0.14 -0.12  0.01  0.10 -0.19 -0.11 -0.06
## obj           -0.10 -0.16 -0.10 -0.05 -0.05 -0.02  0.02
## subj           0.14  0.05  0.13  0.18 -0.23 -0.15 -0.08
## VERBfrac.m     0.08 -0.28 -0.21 -0.15 -0.08 -0.03  0.01
## NOUNcount.m    0.16 -0.15 -0.12 -0.08 -0.11 -0.07 -0.03
## activity      -0.09 -0.39 -0.35 -0.29 -0.08 -0.03  0.01
## cli            1.01 -0.31 -0.28 -0.23  0.07  0.08  0.11
## entropy        0.09 -0.02  0.01  0.03  0.50  0.54  0.58
## fkgl           0.12  0.20  0.25  0.29  0.01  0.04  0.05
## fre           -0.47 -0.16 -0.12 -0.09 -0.06 -0.04 -0.01
## hpoint         0.00 -0.04 -0.02  0.01 -0.04 -0.02  0.01
## mamr           0.28 -0.05  0.02  0.08 -0.37 -0.32 -0.27
## mattr          0.13  0.01  0.05  0.08  0.78  0.83  0.86
## hapaxes        0.06 -0.01  0.02  0.04  0.26  0.29  0.33
## verbdist      -0.02  0.07  0.16  0.27 -0.14 -0.11 -0.07
##
##  Interfactor correlations and bootstrapped confidence intervals
##            lower estimate  upper
## PA1-PA2 -3.6e-02    0.024  0.079
## PA1-PA6 -5.6e-01   -0.348 -0.123
## PA1-PA3 -6.9e-01    0.082  0.274
## PA1-PA5 -6.7e-01   -0.440  0.319
## PA1-PA4 -6.4e-01   -0.277  0.134
## PA2-PA6  1.6e-01    0.290  0.374
## PA2-PA3  1.7e-02    0.155  0.265
## PA2-PA5  4.1e-02    0.159  0.249
## PA2-PA4  5.3e-02    0.165  0.262
## PA6-PA3  8.8e-02    0.255  0.357
## PA6-PA5  7.1e-02    0.248  0.378
## PA6-PA4 -4.7e-05    0.138  0.348
## PA3-PA5 -2.7e-02    0.360  0.460
## PA3-PA4 -1.1e-01    0.097  0.383
```

```
## PA5-PA4 -9.5e-02    0.095  0.320
```

Loadings

```r
fa_res$loadings
```

```
##
## Loadings:
##                PA1    PA2    PA6    PA3    PA5    PA4
## analyticVERBs  0.863                       0.363
## passives       0.117                -0.195  0.896
## predorder.m   -0.676          0.166                -0.109
## obj            0.192          0.906 -0.166 -0.103
## subj           0.676  0.125                 0.132 -0.146
## VERBfrac.m     0.817                       -0.214
## NOUNcount.m   -1.041         -0.131  0.129 -0.117
## activity       0.777          0.197 -0.124 -0.348
## cli            0.184         -0.139  0.961 -0.283
## entropy        0.122  0.738                        0.538
## fkgl          -0.369          0.634         0.249
## fre            0.132         -0.572 -0.551 -0.119
## hpoint                0.936
## mamr           0.676                 0.218         -0.315
## mattr                -0.122                        0.828
## hapaxes              -0.928                        0.294
## verbdist      -0.873         -0.214         0.162 -0.106
##
##                  PA1   PA2   PA6   PA3   PA5   PA4
## SS loadings    5.495 2.331 1.708 1.403 1.329 1.225
## Proportion Var 0.323 0.137 0.100 0.083 0.078 0.072
## Cumulative Var 0.323 0.460 0.561 0.643 0.722 0.794
```

```r
for (i in 1:fa_res$factors) {
  cat("\n-----", colnames(fa_res$loadings)[i], "-----\n")

  loadings <- fa_res$loadings[, i]
  load_df <- data.frame(loading = loadings)

  load_df_filtered <- load_df %>%
    mutate(abs_l = abs(loading)) %>%
    mutate(str = case_when(
      abs_l > 0.7 ~ "***",
      abs_l <= 0.7 & abs_l > 0.5 ~ "** ",
      abs_l <= 0.5 & abs_l > 0.3 ~ "*  ",
      abs_l <= 0.3 & abs_l > 0.1 ~ ".  ",
      .default = ""
    )) %>%
    arrange(-abs_l) %>%
    filter(abs_l > 0.1)

  load_df_filtered %>%
    mutate(across(c(loading, abs_l), ~ round(.x, 3))) %>%
    print()
```

```r
  cat("\n")
}
```

```
##
## ----- PA1 -----
##                loading abs_l str
## NOUNcount.m     -1.041 1.041 ***
## verbdist        -0.873 0.873 ***
## analyticVERBs    0.863 0.863 ***
## VERBfrac.m       0.817 0.817 ***
## activity         0.777 0.777 ***
## mamr             0.676 0.676 **
## subj             0.676 0.676 **
## predorder.m     -0.676 0.676 **
## fkgl            -0.369 0.369 *
## obj              0.192 0.192 .
## cli              0.184 0.184 .
## fre              0.132 0.132 .
## entropy          0.122 0.122 .
## passives         0.117 0.117 .
##
##
## ----- PA2 -----
##          loading abs_l str
## hpoint    0.936 0.936 ***
## hapaxes  -0.928 0.928 ***
## entropy   0.738 0.738 ***
## subj      0.125 0.125 .
## mattr    -0.122 0.122 .
##
##
## ----- PA6 -----
##              loading abs_l str
## obj            0.906 0.906 ***
## fkgl           0.634 0.634 **
## fre           -0.572 0.572 **
## verbdist      -0.214 0.214 .
## activity       0.197 0.197 .
## predorder.m    0.166 0.166 .
## cli           -0.139 0.139 .
## NOUNcount.m   -0.131 0.131 .
##
##
## ----- PA3 -----
##              loading abs_l str
## cli            0.961 0.961 ***
## fre           -0.551 0.551 **
## mamr           0.218 0.218 .
## passives      -0.195 0.195 .
## obj           -0.166 0.166 .
## NOUNcount.m    0.129 0.129 .
## activity      -0.124 0.124 .
##
##
```

```
## ----- PA5 -----
##               loading abs_l str
## passives        0.896 0.896 ***
## analyticVERBs   0.363 0.363 *
## activity       -0.348 0.348 *
## cli            -0.283 0.283 .
## fkgl            0.249 0.249 .
## VERBfrac.m     -0.214 0.214 .
## verbdist        0.162 0.162 .
## subj            0.132 0.132 .
## fre            -0.119 0.119 .
## NOUNcount.m    -0.117 0.117 .
## obj            -0.103 0.103 .
##
##
## ----- PA4 -----
##             loading abs_l str
## mattr         0.828 0.828 ***
## entropy       0.538 0.538 **
## mamr         -0.315 0.315 *
## hapaxes       0.294 0.294 .
## subj         -0.146 0.146 .
## predorder.m  -0.109 0.109 .
## verbdist     -0.106 0.106 .
```

hypotheses:

- **PA1:** register – narrativity, richness of expression; non-technicality (not sticking to terminology as much etc.?)
- **PA2:** text length
- **PA6:** sentence complexity (more clauses)
  - slightly longer nominal constructions / more objects, more years of education necessary, predicates slightly further in the clause, slightly more verbs
- **PA3:** unit lengths (sentence length & word length)
  - slightly more passives, slightly more objects, slightly less verbal overall / slightly longer nom. constructions, slightly morphologically richer, many years of education necessary
  - more enumerations? but one would expect higher `activity` differences to occur if that was the case
- **PA5:** passives? (there's probably more to it)
- **PA4:** lexical diversity?

strong correlations:

- **PA1–PA6:** non-technical texts likely more to the point overall, making them shorter
- … other ones

hypotheses **ON AN OLD ANALYSIS**:

- **PA1:** written, formal register (complex) vs. more spoken-like register
  - long, severely complex, nominalized sentences / shorter, more verbal sentences
  - narrativity? (1st and 2nd persons etc.)
- **PA4:** structure size? elaboratedness of expression? advancement (in years of age)?
  - short words, short sentences, more negations / long words, long sentences, more objects
  - cli: word complexity - sentence easiness
  - the negations might be because of the varying sentence length
    - FrBo more instructional than CzCDC, meaning less negation (the text tells the reader what to do, not what *not* to do)

- **PA2:** text length & enumerations
- **PA3:** intra-text (syntactic, possibly content-related) variation
  - note that the loadings of `VERBfrac.v` and `NEGcount.v` are negligible
  - however, the loading of `entropy.v` is significant
- **PA5:** negation
- **PA6:** passive / active
  - more passives => more tokens in a sentence, but the same no. of verbs (passive participles classified as ADJ in UD)
- **PA7:** unique words

  NOTE: variables with low communalities are excluded from the analysis, yet still likely play a role in legal writing readability. this includes both those selected for the analysis and the excluded ones.

  NOTE: some high-correlating variables were excluded from the FA.

Strong correlations **ON AN OLD ANALYSIS**:

- **PA1–PA3:** possible register switching
- **PA4–PA5:** expression sophisticatedness

Healthiness diagnostics

```
fa_res$loadings[] %>%
  as_tibble() %>%
  mutate(feat = cnames) %>%
  select(feat, everything()) %>%
  pivot_longer(!feat) %>%
  mutate(value = abs(value)) %>%
  group_by(feat) %>%
  summarize(maxload = max(value)) %>%
  arrange(maxload)
```

```
## # A tibble: 17 x 2
##    feat          maxload
##    <chr>           <dbl>
##  1 fre             0.572
##  2 fkgl            0.634
##  3 predorder.m     0.676
##  4 subj            0.676
##  5 mamr            0.676
##  6 entropy         0.738
##  7 activity        0.777
##  8 VERBfrac.m      0.817
##  9 mattr           0.828
## 10 analyticVERBs   0.863
## 11 verbdist        0.873
## 12 passives        0.896
## 13 obj             0.906
## 14 hapaxes         0.928
## 15 hpoint          0.936
## 16 cli             0.961
## 17 NOUNcount.m     1.04
```

```
fa_res$communality %>% sort()
```

```
##          subj   predorder.m      passives analyticVERBs           obj
```

```
##     0.5145745     0.5391177     0.6284491     0.6400169     0.6800036
##         mattr          mamr      verbdist       hapaxes   NOUNcount.m
##     0.7231000     0.7516432     0.7898567     0.8558636     0.8561580
##        hpoint    VERBfrac.m      activity           cli       entropy
##     0.8674644     0.8704664     0.8923625     0.9072700     0.9518002
##           fre          fkgl
##     0.9776249     0.9953918
```

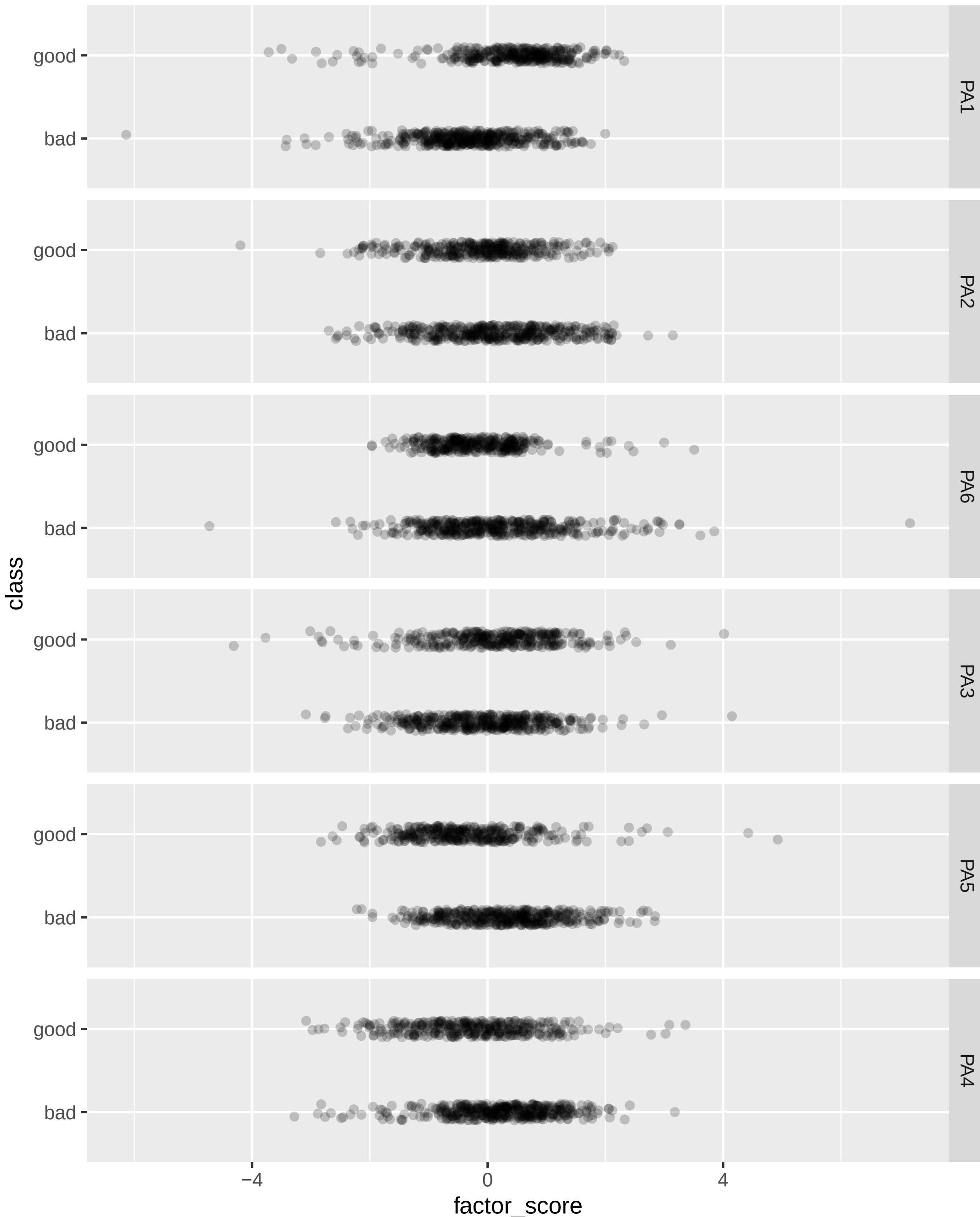
Uniquenesses

```
fa_res$uniquenesses %>% round(3)
```

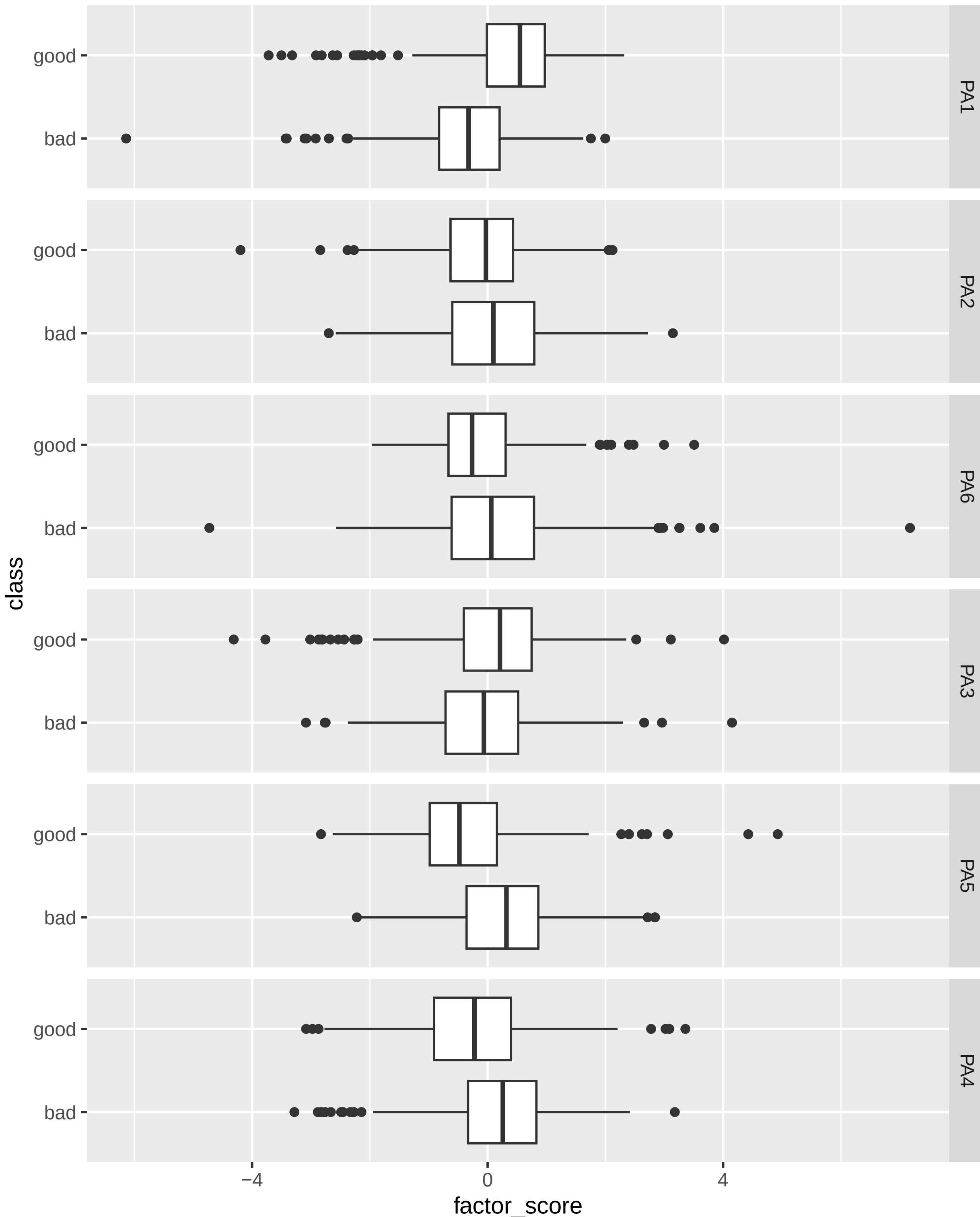
```
## analyticVERBs      passives   predorder.m           obj          subj
##         0.360         0.372         0.461         0.320         0.485
##    VERBfrac.m   NOUNcount.m      activity           cli       entropy
##         0.130         0.144         0.108         0.093         0.048
##          fkgl           fre        hpoint          mamr         mattr
##         0.005         0.022         0.133         0.248         0.277
##       hapaxes      verbdist
##         0.144         0.210
```

Plots

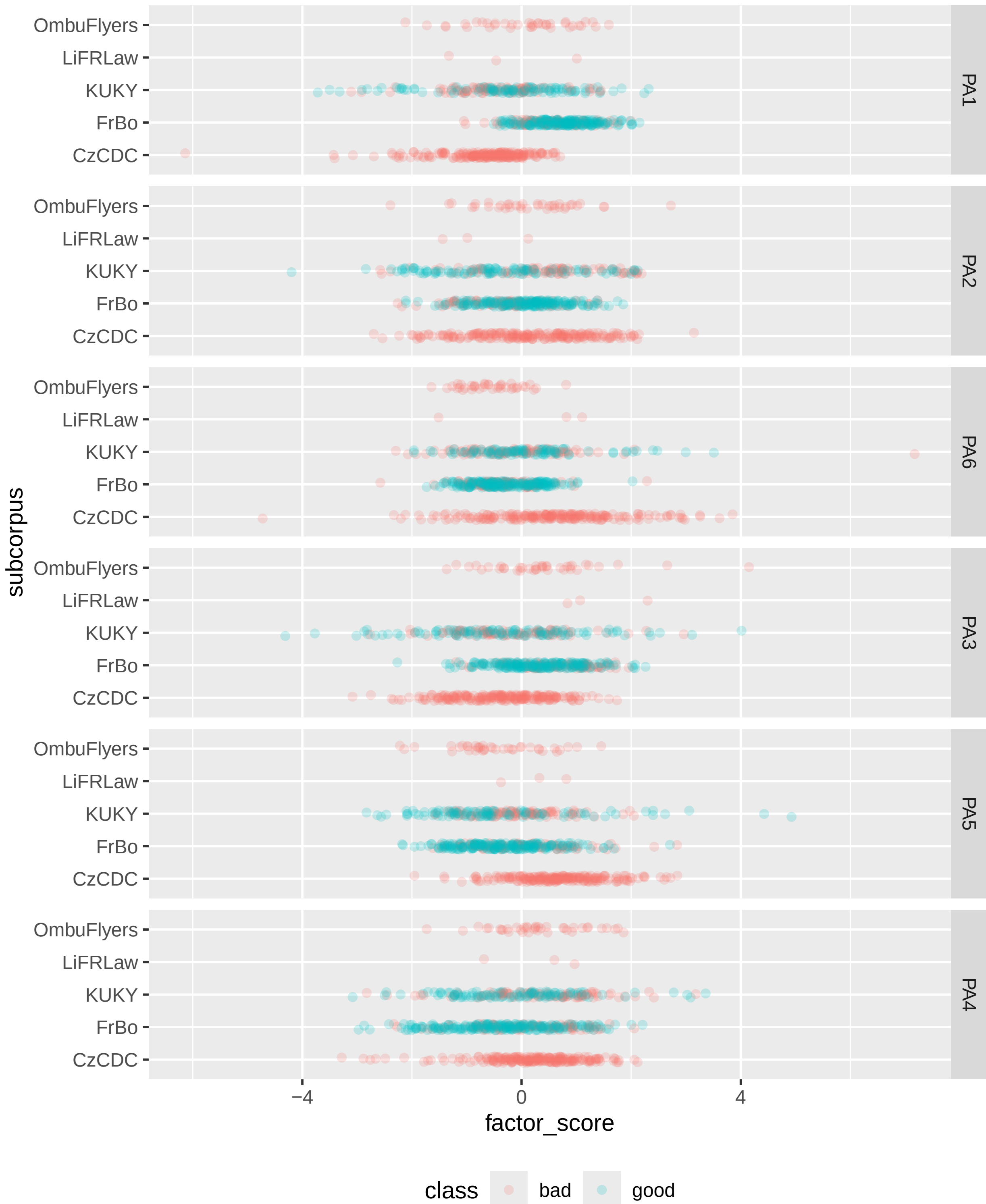
```
data_factors <- bind_cols(data_clean, fa_res$scores %>% as.data.frame())
data_factors_long <- data_factors %>%
  pivot_longer(PA1:PA4, names_to = "factor", values_to = "factor_score") %>%
  mutate(across(
    factor,
    ~ factor(.x, levels = c("PA1", "PA2", "PA6", "PA3", "PA5", "PA4"))
  ))

data_factors_long %>%
  ggplot(aes(x = factor_score, y = class)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_jitter(width = 0, height = 0.1, alpha = 0.2)
```



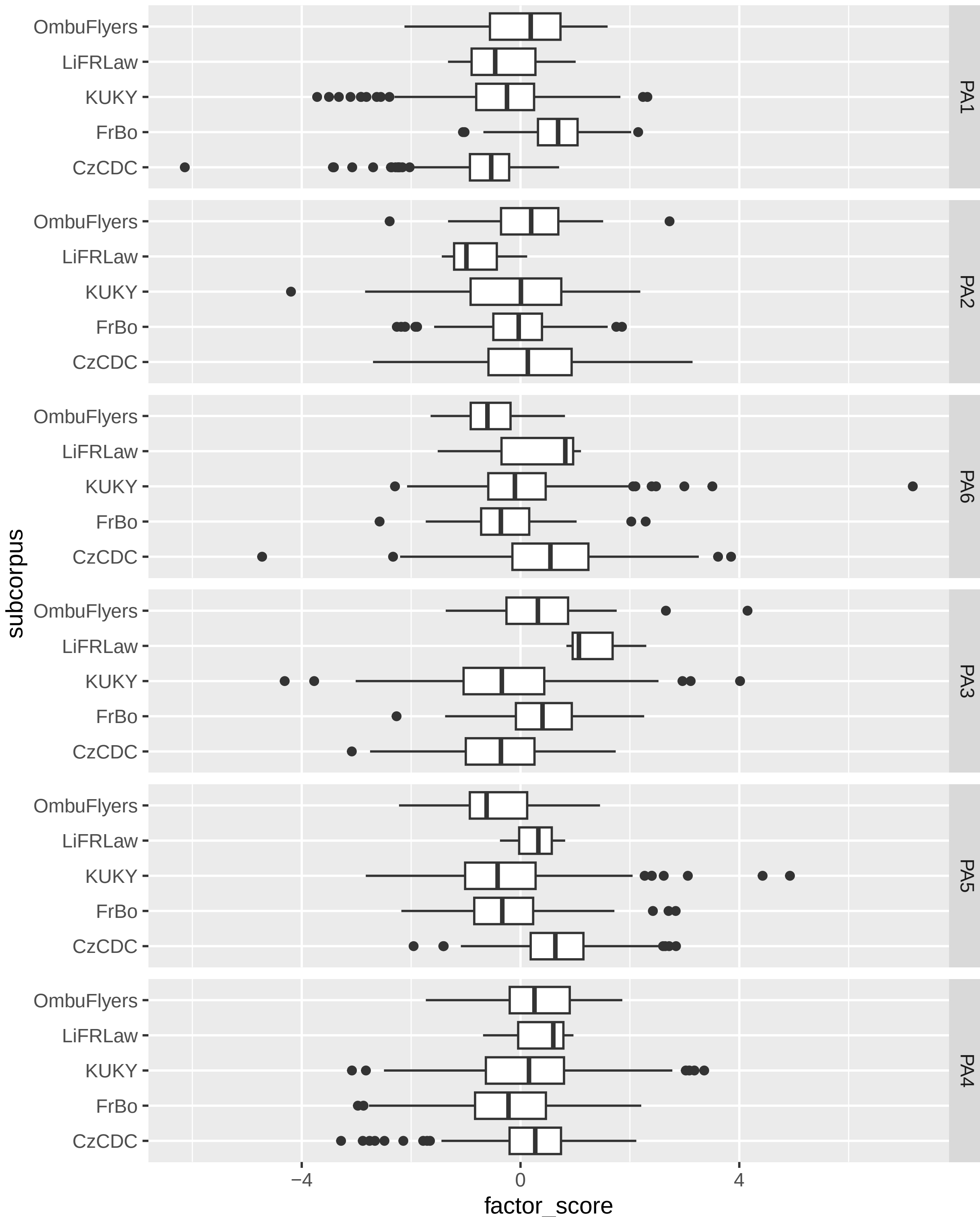

```r
data_factors_long %>% ggplot(aes(x = factor_score, y = class)) +
  geom_boxplot() +
  facet_grid(factor ~ .)
```



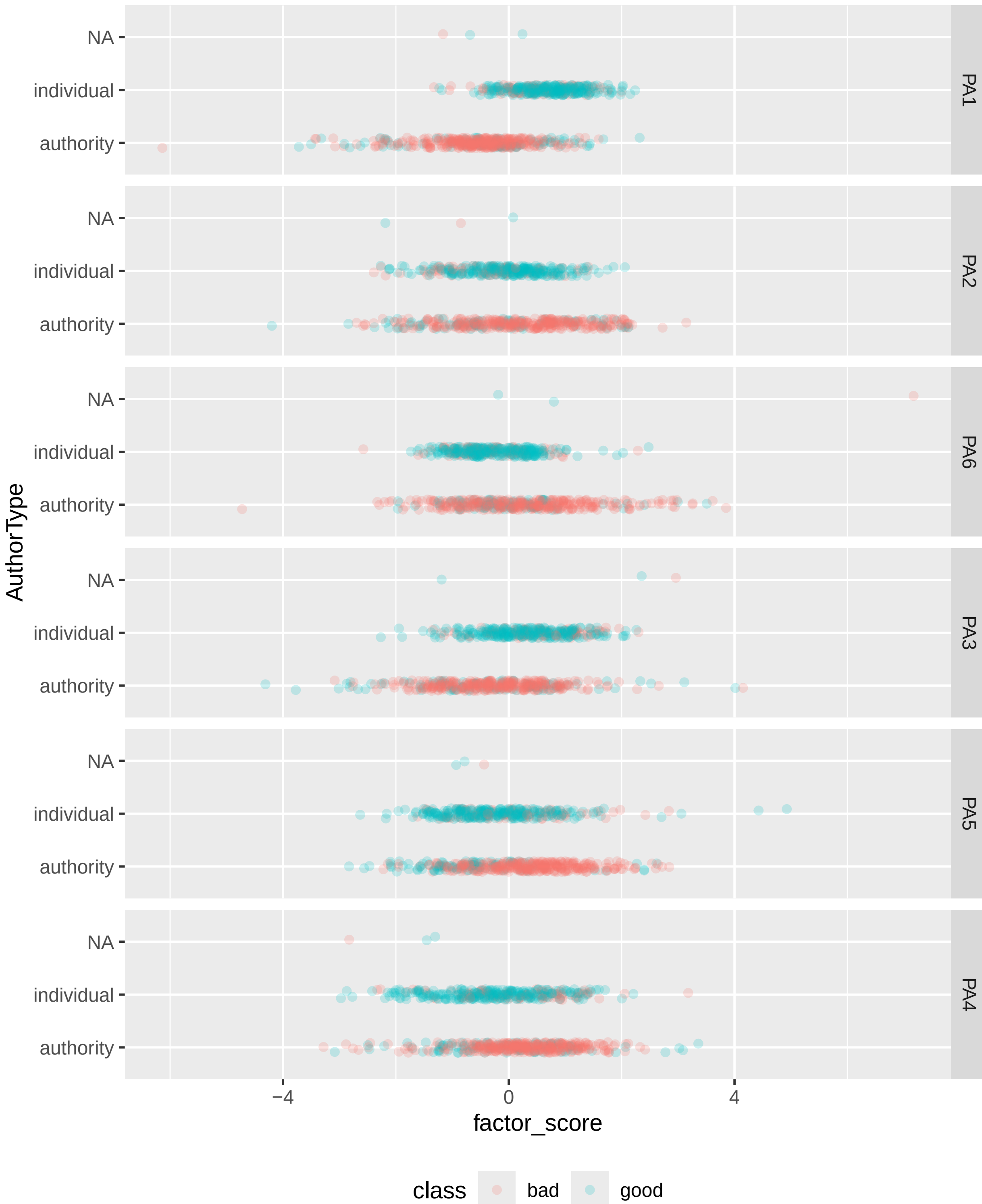

```r
data_factors_long %>%
  ggplot(aes(x = factor_score, y = subcorpus, color = class)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_jitter(width = 0, height = 0.1, alpha = 0.2)
```



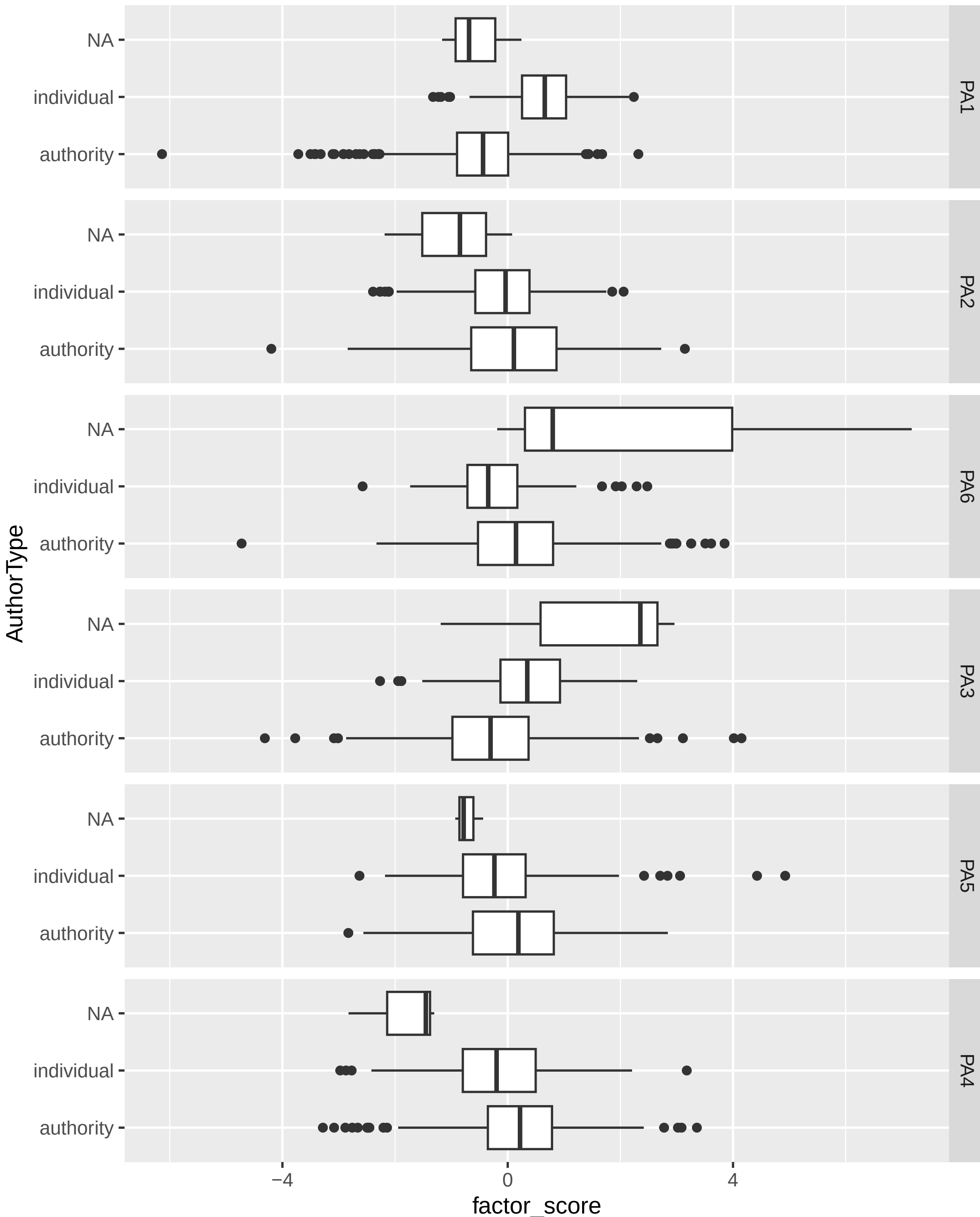

```
data_factors_long %>% ggplot(aes(x = factor_score, y = subcorpus)) +
  geom_boxplot() +
  facet_grid(factor ~ .)
```



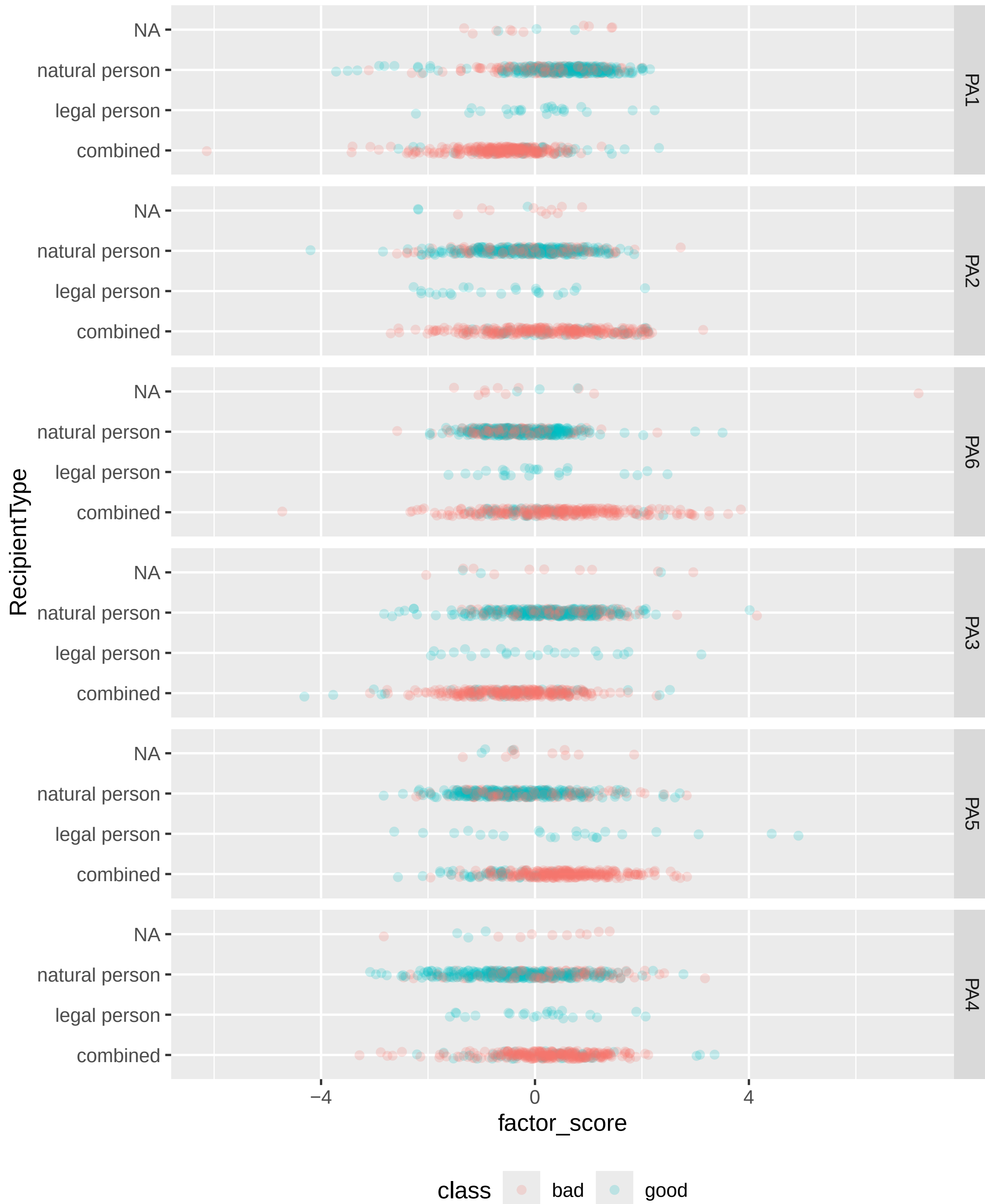

```r
data_factors_long %>%
  ggplot(aes(x = factor_score, y = AuthorType, color = class)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_jitter(width = 0, height = 0.1, alpha = 0.2)
```



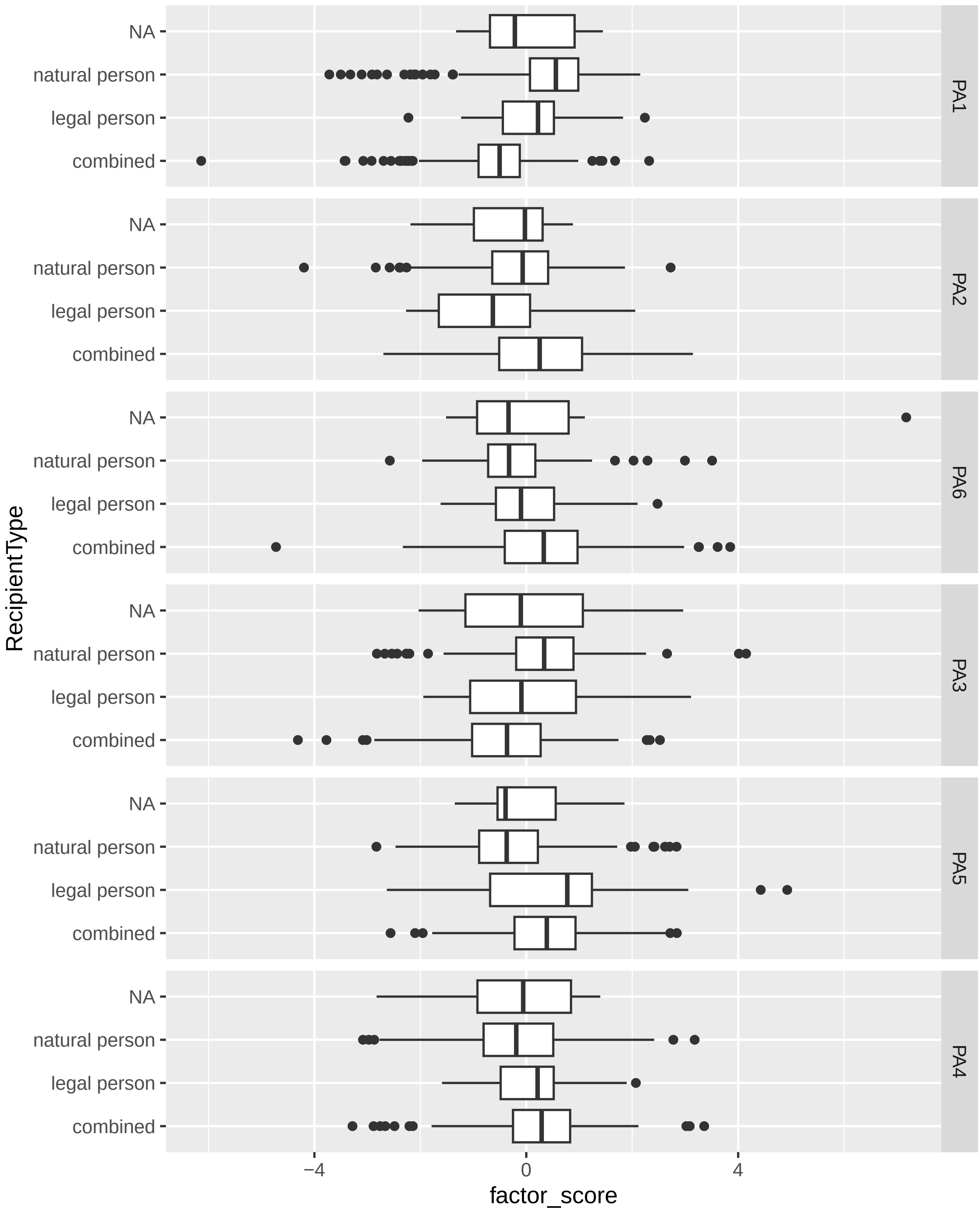

```
data_factors_long %>%
  ggplot(aes(x = factor_score, y = AuthorType)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_boxplot()
```




```r
data_factors_long %>%
  ggplot(aes(x = factor_score, y = RecipientType, color = class)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_jitter(width = 0, height = 0.1, alpha = 0.2)
```



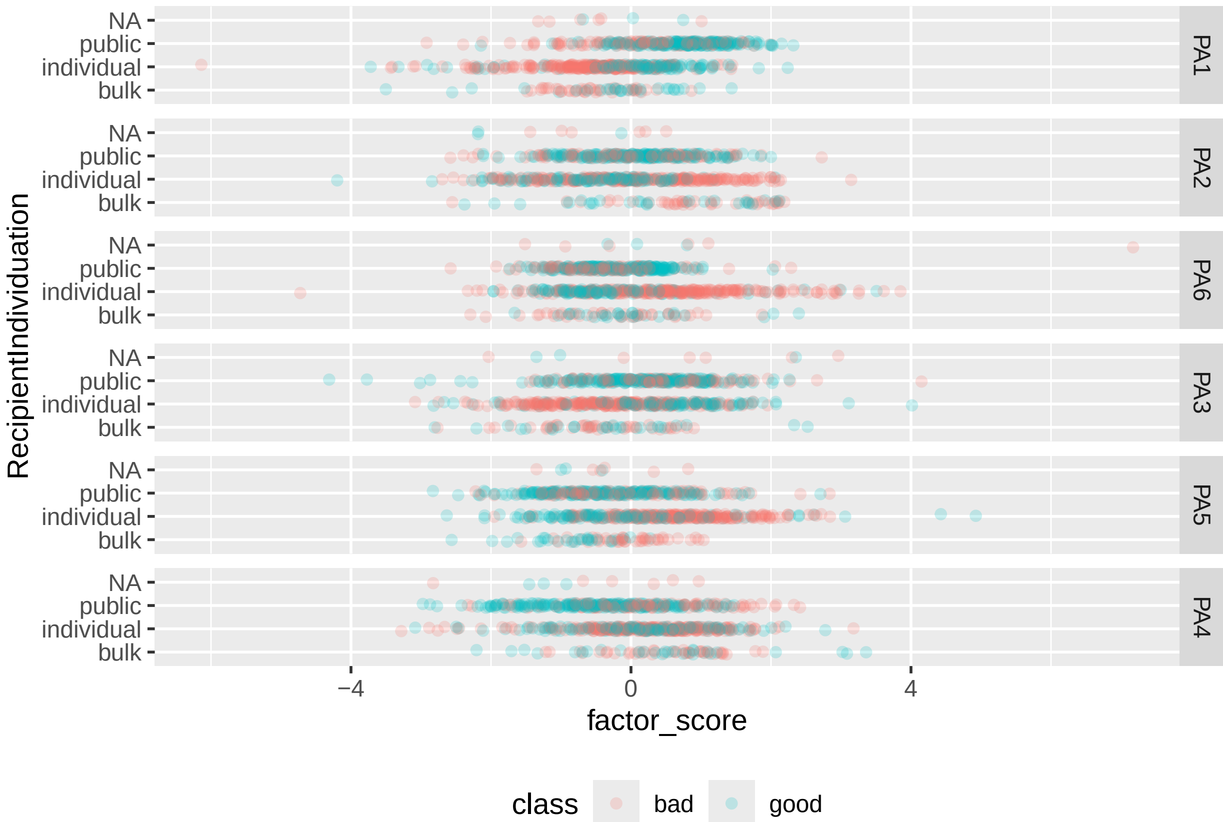

```r
data_factors_long %>%
  ggplot(aes(x = factor_score, y = RecipientType)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_boxplot()
```



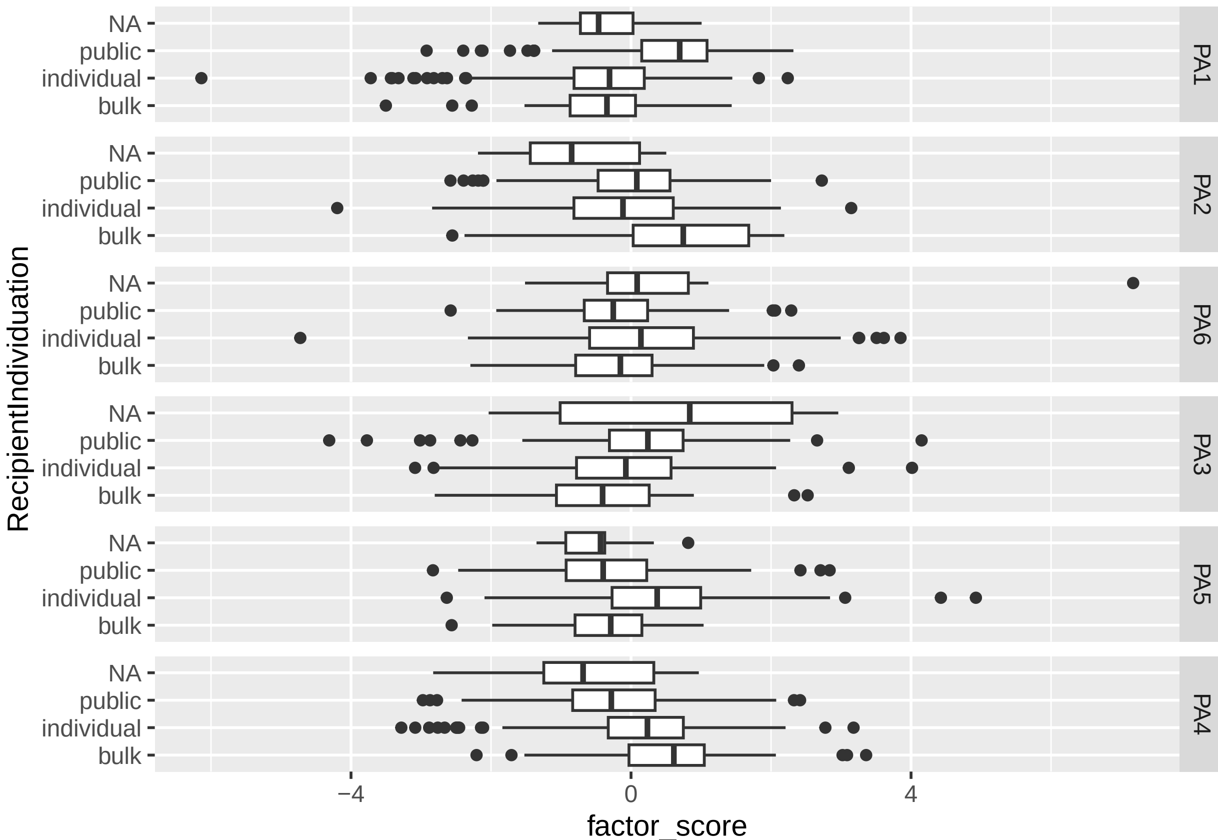
court decisions often `combined`.

```
data_factors_long %>%
  ggplot(aes(x = factor_score, y = RecipientIndividuation, color = class)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
```

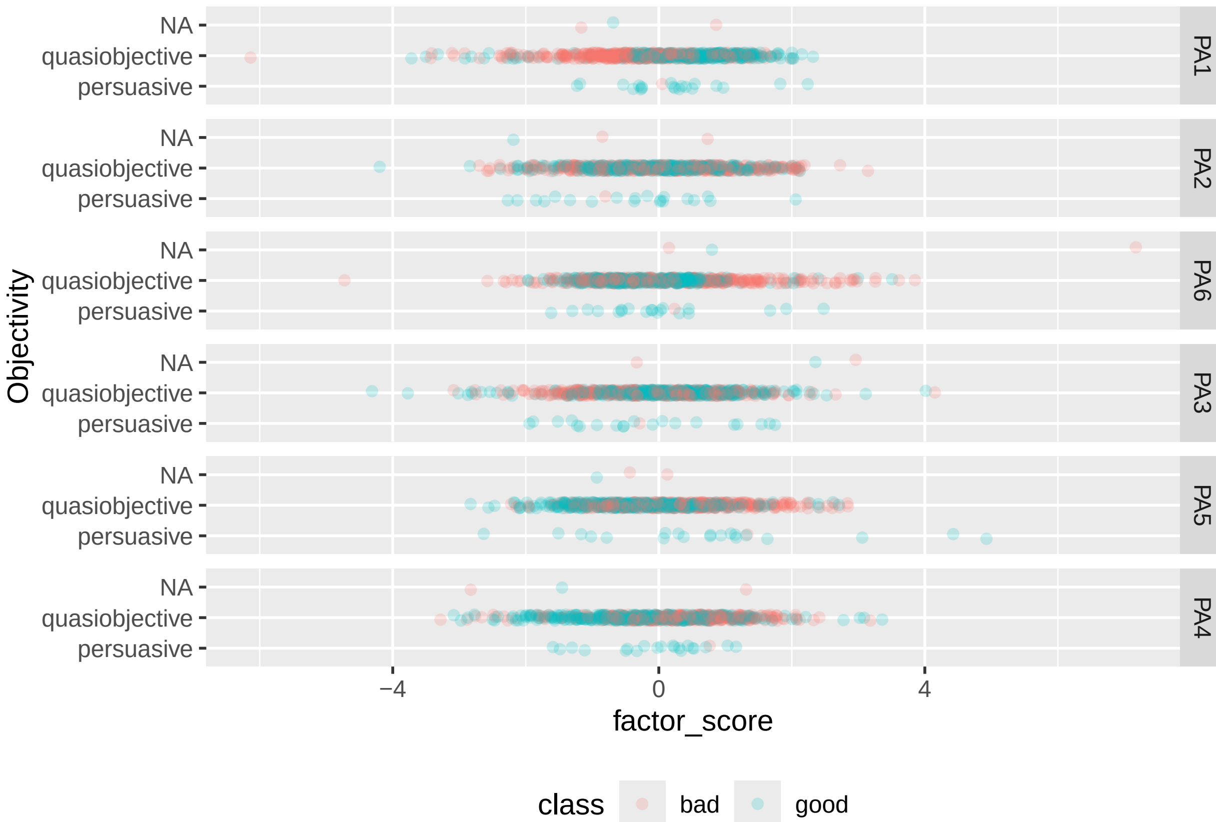
```r
  geom_jitter(width = 0, height = 0.1, alpha = 0.2)
```



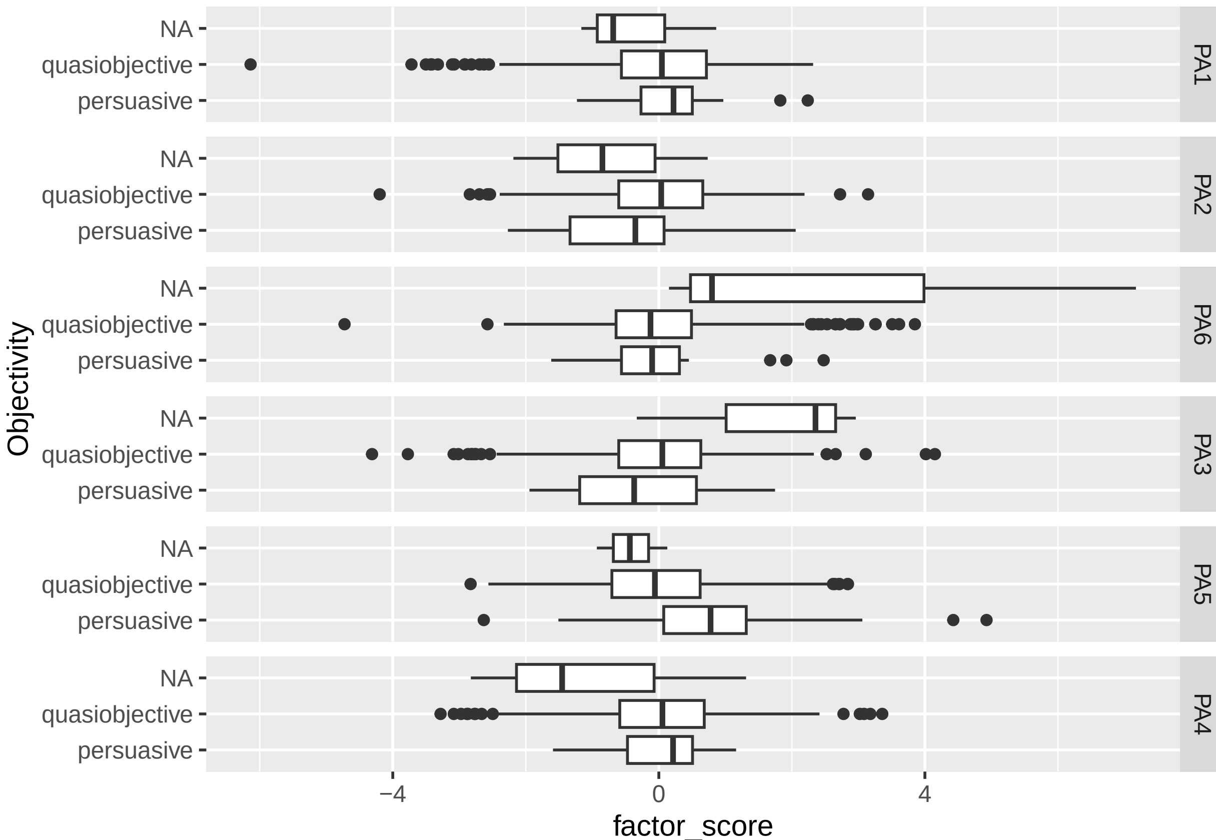

```r
data_factors_long %>%
  ggplot(aes(x = factor_score, y = RecipientIndividuation)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_boxplot()
```



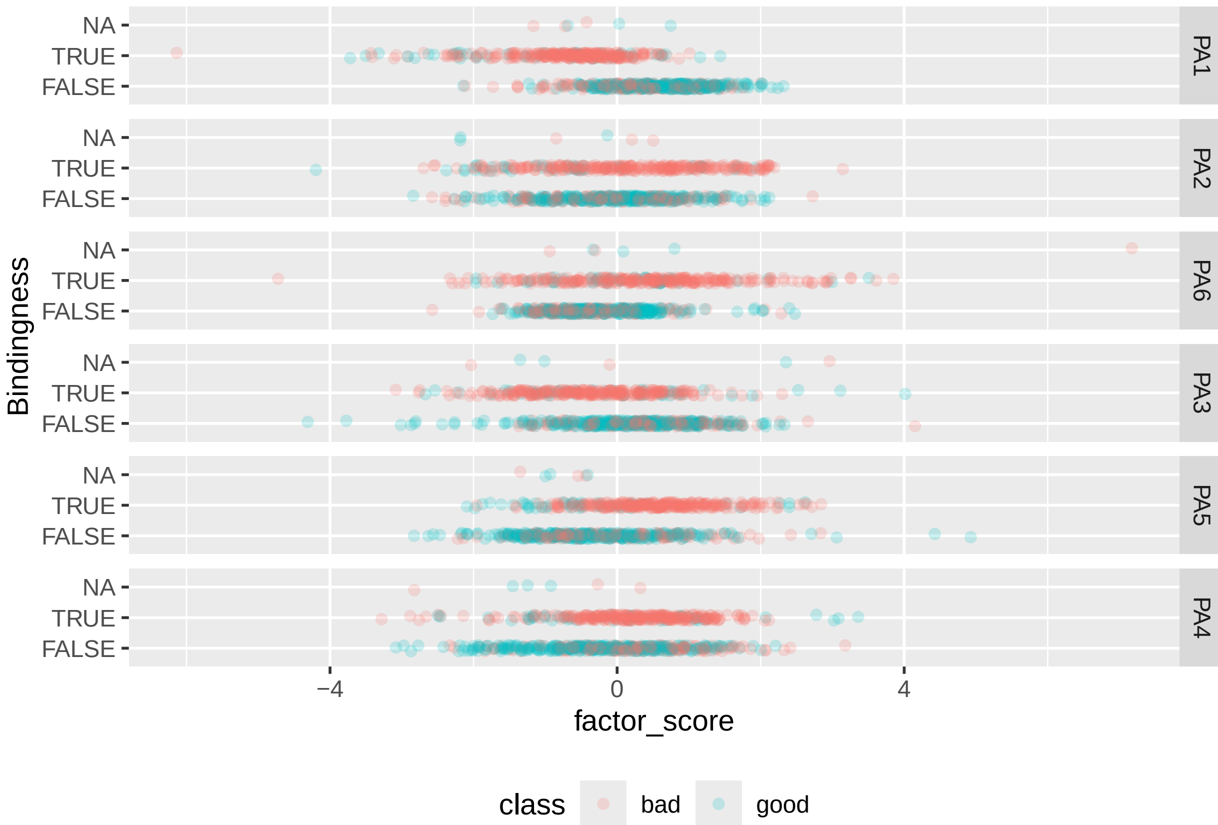

```
data_factors_long %>%
  ggplot(aes(x = factor_score, y = Objectivity, color = class)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_jitter(width = 0, height = 0.1, alpha = 0.2)
```



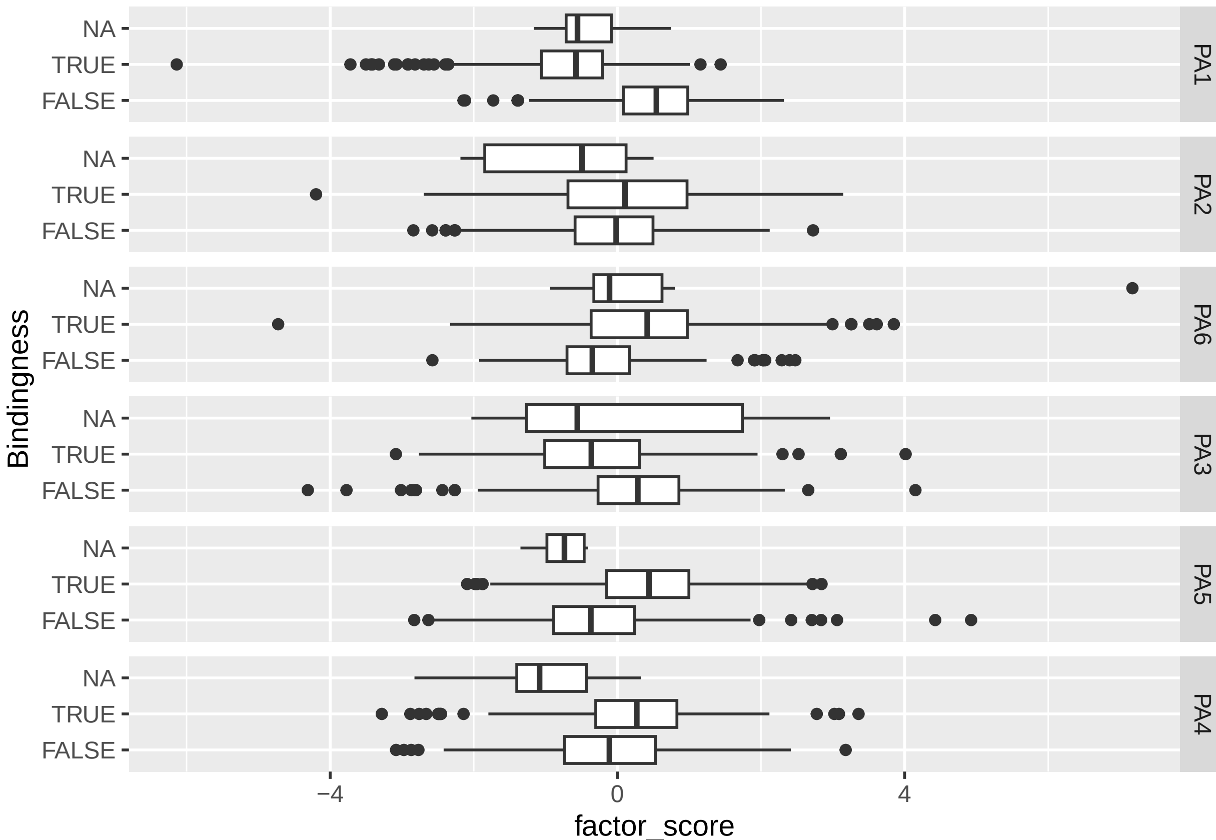

```r
data_factors_long %>%
  ggplot(aes(x = factor_score, y = Objectivity)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_boxplot()
```




```
data_factors_long %>%
  ggplot(aes(x = factor_score, y = Bindingness, color = class)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_jitter(width = 0, height = 0.1, alpha = 0.2)
```




```
data_factors_long %>%
  ggplot(aes(x = factor_score, y = Bindingness)) +
  facet_grid(factor ~ .) +
  theme(legend.position = "bottom") +
  geom_boxplot()
```



Bindingness
NA
TRUE
FALSE
PA1
PA2
PA6
PA3
PA5
PA4
−4
0
4
factor_score