Knowledge & Motivation Instrument Correlations

This study set out to assess how prior survey instruments on sustainable behaviors, knowledge, and attitudes correlate.Many studies on environmental behavior measure people's motivation to show sustainable behavior and classify them as being highly or lowly motivated, and their knowledge about what the right behavior would look like. An example would be recycling.

Read data

```
pacman::p_load(dplyr,purrr,tidyr,here, haven,tibble,ggplot2,ggh4x,lme4,knitr,kableExtra,gt,pander,flextable,ggh
options(digits=2, scipen=999, dplyr.summarise.inform=FALSE)

library(gridExtra)
library(factoextra)
library(mgcv)
library(lavaan)
library(CCA)
library(qgraph)
library(rpart)
library(rpart.plot)
library(mclust)
library(tidyLPA)

select = dplyr::select

source(here("scripts","survey_functions.R"))

draw <- readRDS(here("data","draw.rds"))
dinst <- readRDS(here("data","dinst.rds"))

# Attari Energy Survey (Part 1)
aes1 <- draw |> select(id,ATT01:ATT18)
aes2 <- dinst |> select(id,ATT01:ATT18)
aes_combined <- bind_rows(aes1, aes2)
```

```r
att_useSave <- draw |> select(id,ATT19:ATT33)
att_useSave2 <- dinst |> select(id,ATT19:ATT33)
att2_combined <- bind_rows(att_useSave, att_useSave2)

els1 <- draw |> select(id,ELS01:ELS08)
els2 <- dinst |> select(id,ELS01:ELS08)
els <- bind_rows(els1,els2)

rs1 <- draw |> select(id,RS01:RS06)
rs2 <- dinst |> select(id,RS01:RS06)
rs <- bind_rows(rs1,rs2)

attari1 <- analyze_attari_survey_part1(aes_combined)
attari2_scores <- analyze_attari_survey(att2_combined)
els_scores <- analyze_els_survey(els)
rs_scores <- analyze_recycling_survey(rs)

# Combine all scores into one dataframe
combined_scores <- attari1 %>%
  left_join(attari2_scores, by="id") %>%
  left_join(els_scores, by="id") %>%
  left_join(rs_scores, by="id")

# Rename columns for clarity
names(combined_scores) <- c("id", "perceived_difficulty", "numeracy",
                          "energy_use", "energy_save",
                          "els_accuracy", "els_score",
                          "env_attitude", "env_attitude_z",
                          "pol_conservatism", "pol_conservatism_z")
```

preview data

```r
combined_scores |> head(5) |> kable() |> kable_styling("striped", full_width = F)
```

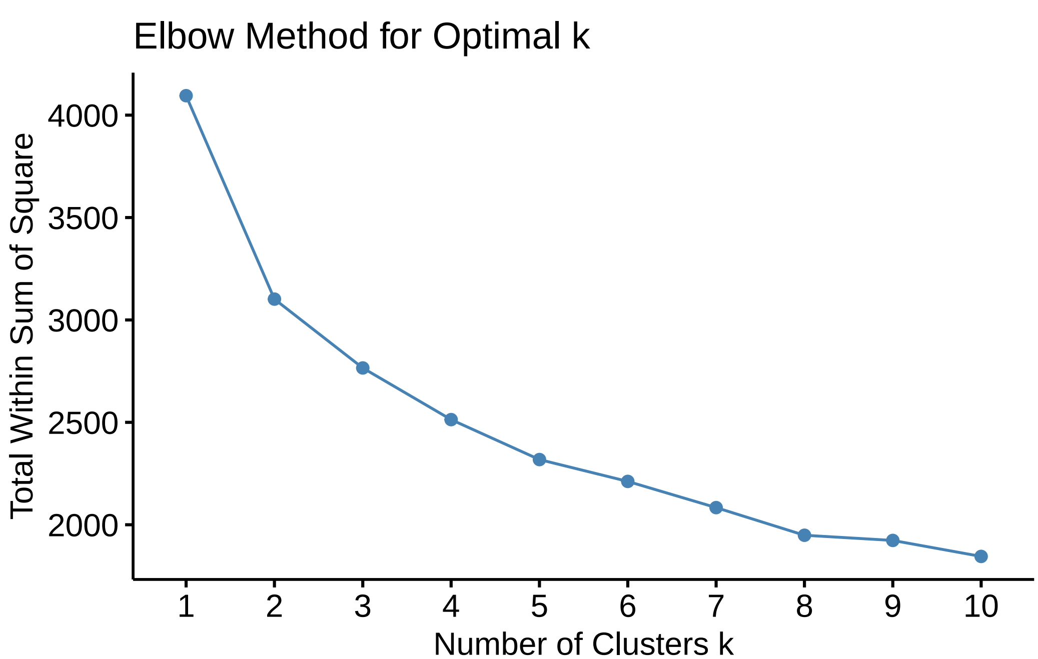
id	perceived_difficulty	numeracy	energy_use	energy_save	els_accuracy	els_score	env_attitude	env_attitude_z
1	0.61	1.5	1.101	1.01	6	0.74	3.2	-0.43
2	-0.45	1.5	0.137	-0.46	5	0.20	3.5	-0.11

3	2.09	-2.0	-1.440	0.70	4	-0.33	3.0	-0.76
4	-0.69	-1.3	1.346	2.16	2	-1.40	3.8	0.22
5	0.91	1.5	0.075	-0.52	3	-0.87	3.8	0.22

1a

::: {.cell}

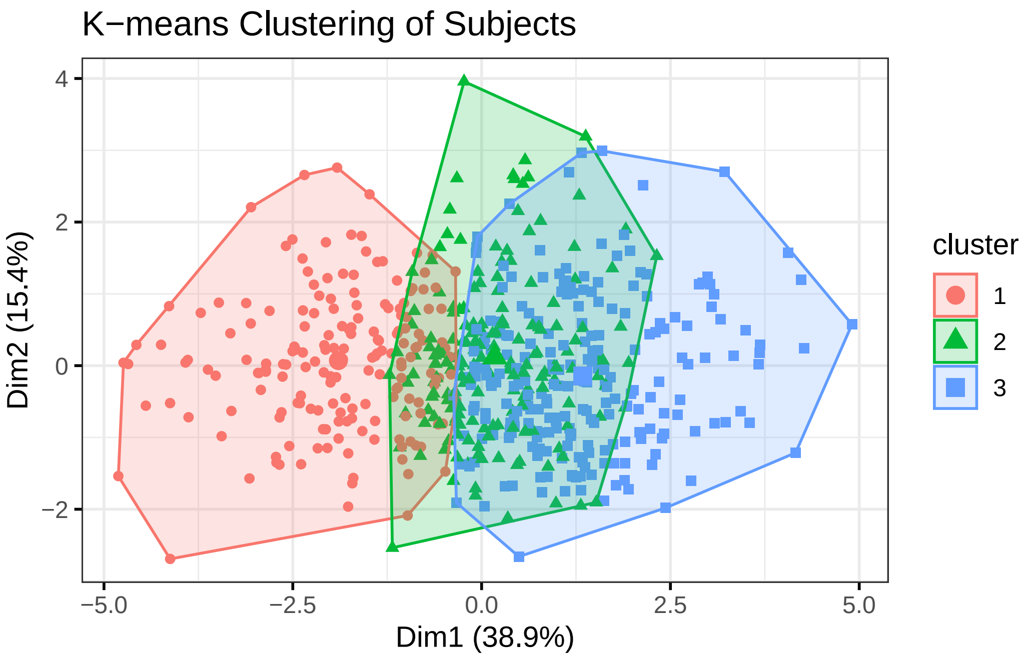
```
# 1. Cluster Analysis

# Prepare data for clustering (select relevant variables and scale)
cluster_data <- combined_scores %>%
 select(perceived_difficulty, numeracy, energy_use, energy_save, els_score, env_attitude_z, pol_conservatism_z
 na.omit() %>%
 scale()

# Determine optimal number of clusters using the elbow method
fviz_nbclust(cluster_data, kmeans, method = "wss") +
 labs(title = "Elbow Method for Optimal k", x = "Number of Clusters k")
```



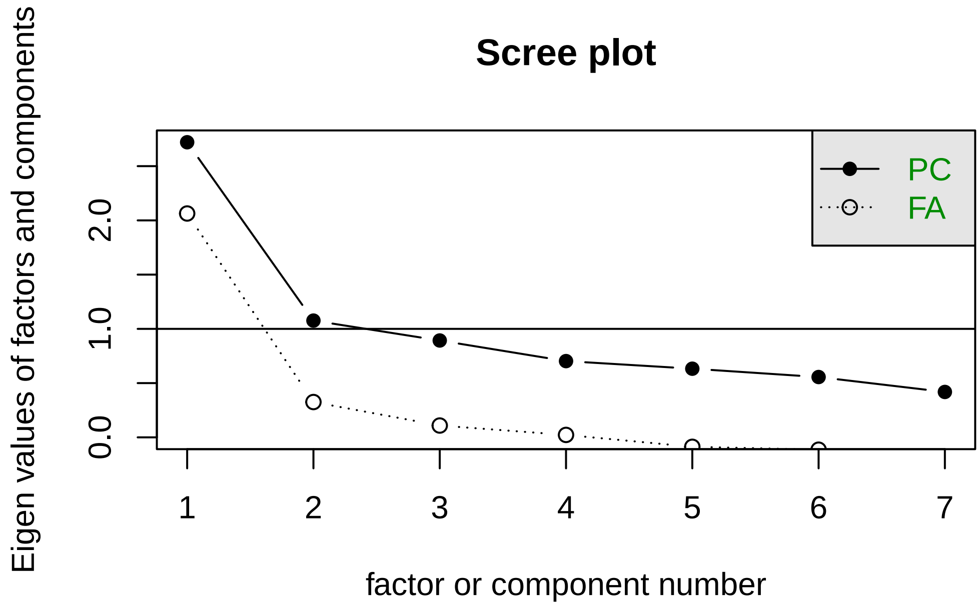
::: {.cell-output-display} ::: :::

```
# Perform k-means clustering (e.g., with 3 clusters)
set.seed(123)
km_result <- kmeans(cluster_data, centers = 3, nstart = 25)

# Visualize the clusters
```

```
fviz_cluster(km_result, data = cluster_data,
             geom = "point",
             ellipse.type = "convex",
             ggtheme = theme_bw()) +
  labs(title = "K-means Clustering of Subjects")
```




```
# Add cluster assignments to the main dataframe
combined_scores$cluster <- as.factor(km_result$cluster)


# 2. Enhanced Factor Analysis


# Scree plot to determine the number of factors
fa_data <- combined_scores %>%
  select(perceived_difficulty, numeracy, energy_use, energy_save, els_score, env_attitude_z, pol_conservatism_
  na.omit()
scree(fa_data)
```




```
# Perform factor analysis with, e.g., 3 factors
fa_result <- fa(fa_data, nfactors = 2, rotate = "varimax")
print(fa_result, cut = 0.3, sort = TRUE)
```

```
Factor Analysis using method =  minres
Call: fa(r = fa_data, nfactors = 2, rotate = "varimax")
Standardized loadings (pattern matrix) based upon correlation matrix
                     item  MR1   MR2   h2     u2 com
energy_use              3 0.77        0.61 0.3856 1.1
energy_save             4 0.68        0.49 0.5146 1.1
numeracy                2 0.52        0.29 0.7067 1.2
els_score               5 0.50        0.30 0.6954 1.4
pol_conservatism_z      7             0.14 0.8570 2.0
env_attitude_z          6       0.99  1.00 0.0035 1.0
perceived_difficulty    1      -0.36  0.19 0.8120 1.7

                       MR1  MR2
SS loadings           1.72 1.31
Proportion Var        0.25 0.19
Cumulative Var        0.25 0.43
Proportion Explained  0.57 0.43
Cumulative Proportion 0.57 1.00

Mean item complexity =  1.4
Test of the hypothesis that 2 factors are sufficient.

df null model =  21  with the objective function =  1.3 with Chi Square =  760
```

```
df of  the model are 8  and the objective function was  0.03

The root mean square of the residuals (RMSR) is  0.03
The df corrected root mean square of the residuals is  0.04

The harmonic n.obs is  586 with the empirical chi square  17  with prob <  0.035
The total n.obs was  586  with Likelihood Chi Square =  17  with prob <  0.029

Tucker Lewis Index of factoring reliability =  0.97
RMSEA index =  0.044  and the 90 % confidence intervals are  0.014 0.073
BIC =  -34
Fit based upon off diagonal values = 0.99
Measures of factor score adequacy
                                                   MR1  MR2
Correlation of (regression) scores with factors   0.87 1.00
Multiple R square of scores with factors          0.76 0.99
Minimum correlation of possible factor scores     0.52 0.99
```

```r
# 3. Enhanced Regression Models
# Model predicting ELS from motivation, controlling for other knowledge scores
model_els_enhanced <- lm(els_score ~ perceived_difficulty + env_attitude_z + pol_conservatism_z +
                           numeracy + energy_use + energy_save, data = combined_scores)
summary(model_els_enhanced)
```

```
Call:
lm(formula = els_score ~ perceived_difficulty + env_attitude_z +
    pol_conservatism_z + numeracy + energy_use + energy_save,
    data = combined_scores)

Residuals:
    Min      1Q  Median      3Q     Max
-2.8527 -0.5932 -0.0299  0.6199  1.8308

Coefficients:
                                 Estimate           Std. Error t value Pr(>|t|)
(Intercept)          -0.00000000000000163  0.03613218269822796    0.00   1.0000
perceived_difficulty -0.06114449214752567  0.04004735896976241   -1.53   0.1274
env_attitude_z        0.13430310791561964  0.04088654156087833    3.28   0.0011
pol_conservatism_z   -0.02729118081676693  0.03888339472559318   -0.70   0.4830
```

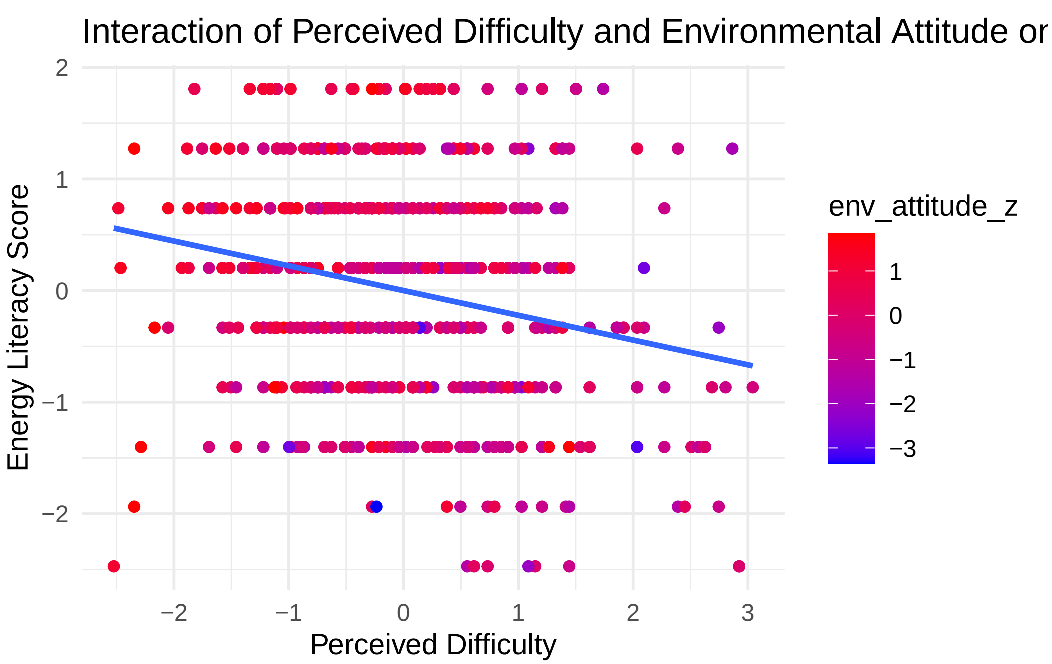
```
numeracy              0.16978503215872956  0.04078151886943691    4.16 0.000036
energy_use            0.19904507771344263  0.04637687030463776    4.29 0.000021
energy_save           0.13859460176947813  0.04521755406911262    3.07   0.0023

(Intercept)
perceived_difficulty
env_attitude_z       **
pol_conservatism_z
numeracy             ***
energy_use           ***
energy_save          **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.88 on 579 degrees of freedom
Multiple R-squared:  0.243, Adjusted R-squared:  0.235
F-statistic: 30.9 on 6 and 579 DF,  p-value: <0.0000000000000002
```

```
# 4. Interaction Effects in Regression
# Example: Interaction between environmental attitude and perceived difficulty on ELS
model_interaction <- lm(els_score ~ perceived_difficulty * env_attitude_z, data = combined_scores)
summary(model_interaction)
```

```
Call:
lm(formula = els_score ~ perceived_difficulty * env_attitude_z,
    data = combined_scores)

Residuals:
   Min     1Q Median     3Q    Max
-3.169 -0.678  0.026  0.689  2.285

Coefficients:
                                    Estimate Std. Error t value   Pr(>|t|)
(Intercept)                          -0.0130     0.0423   -0.31     0.7592
perceived_difficulty                 -0.1383     0.0428   -3.23     0.0013 **
env_attitude_z                        0.2187     0.0428    5.11 0.00000045 ***
perceived_difficulty:env_attitude_z  -0.0337     0.0393   -0.86     0.3915
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.96 on 582 degrees of freedom
Multiple R-squared:  0.0915,    Adjusted R-squared:  0.0868
F-statistic: 19.5 on 3 and 582 DF,  p-value: 0.00000000000438
```

```
# Visualize the interaction (example)
ggplot(combined_scores, aes(x = perceived_difficulty, y = els_score, color = env_attitude_z)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  scale_color_gradient(low = "blue", high = "red") +
  labs(title = "Interaction of Perceived Difficulty and Environmental Attitude on ELS",
       x = "Perceived Difficulty",
       y = "Energy Literacy Score") +
  theme_minimal()
```



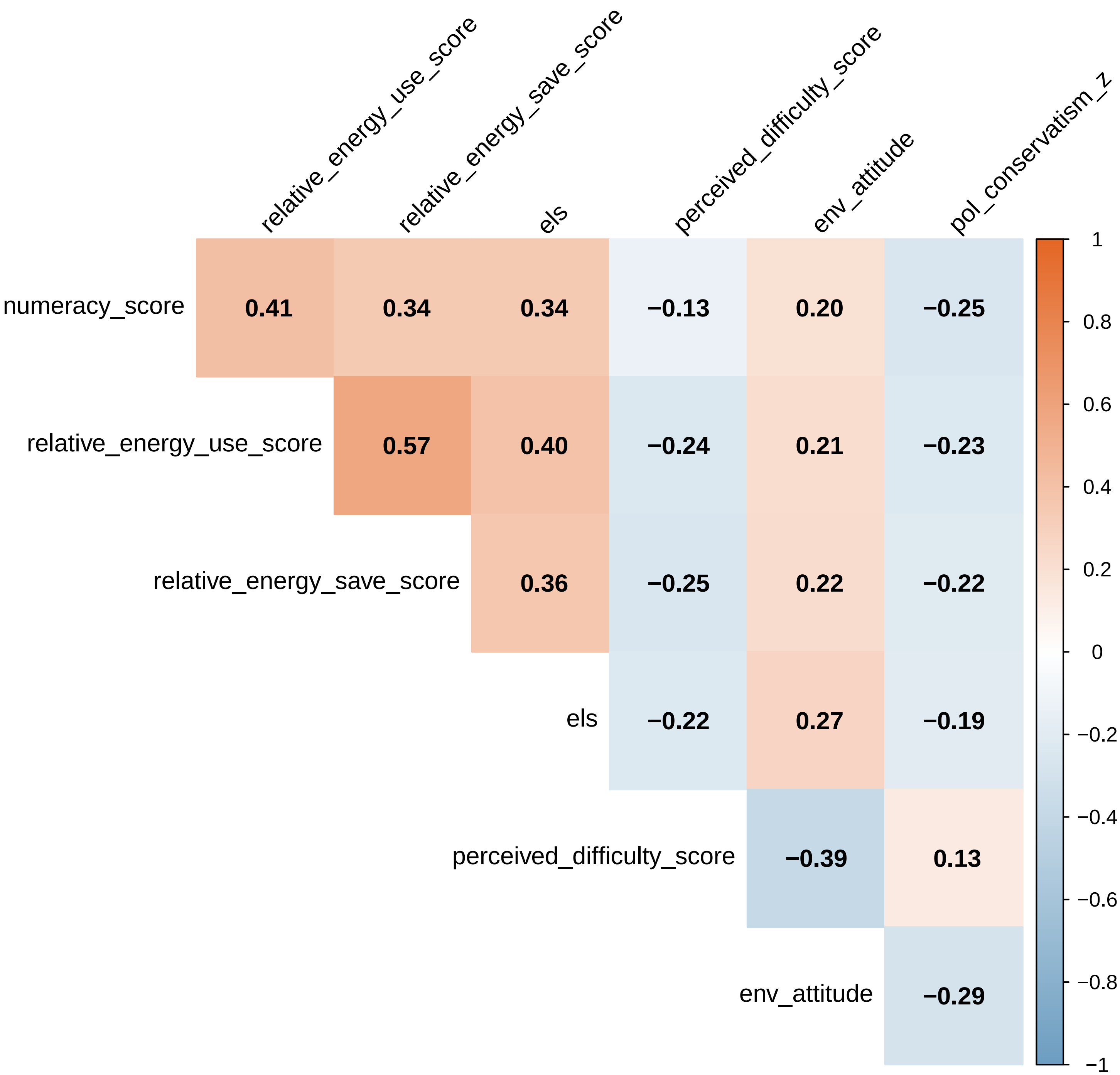
2a

```
combined_df <- attari1 %>%
  full_join(attari2_scores, by = "id") %>%
  full_join(els_scores,       by = "id") %>%
  full_join(rs_scores,        by = "id")



# 1. Create knowledge profiles using cluster analysis
knowledge_vars <- combined_df %>%
```

```
  select(numeracy_score, relative_energy_use_score,
         relative_energy_save_score, els)

set.seed(123)
clusters <- kmeans(scale(knowledge_vars), centers=3)

# Add cluster membership to data
combined_df$knowledge_cluster <- as.factor(clusters$cluster)

# Compare motivation scores across clusters
cluster_comparison <- combined_df %>%
  group_by(knowledge_cluster) %>%
  summarise(
    mean_env_attitude = mean(env_attitude, na.rm=TRUE),
    mean_difficulty = mean(perceived_difficulty_score, na.rm=TRUE)
  )

# 2. Test for non-linear relationships
gam_model <- gam(els ~ s(env_attitude) + s(perceived_difficulty_score),
                 data=combined_df)

# 3. Create interaction model between knowledge and motivation
interaction_model <- lm(els ~ env_attitude * perceived_difficulty_score +
                         numeracy_score, data=combined_df)
```

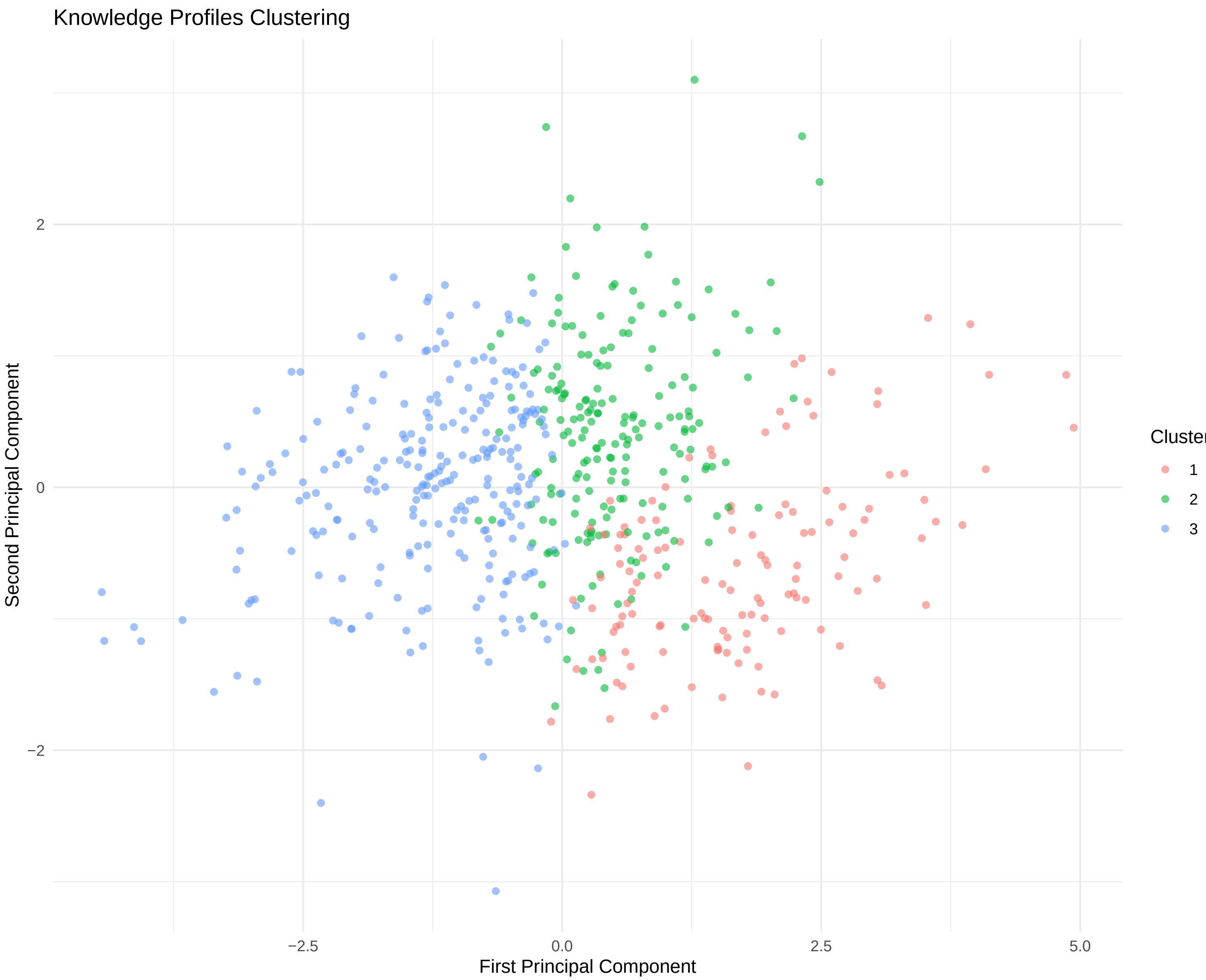
```
# 1. Enhanced Correlation Plot
cor_matrix <- combined_df %>%
  select(numeracy_score, relative_energy_use_score,
         relative_energy_save_score, els,
         perceived_difficulty_score, env_attitude,
         pol_conservatism_z) %>%
  cor(use = "pairwise.complete.obs")

corrplot(cor_matrix,
```

```
         method = "color",
         type = "upper",
         addCoef.col = "black",
         tl.col = "black",
         tl.srt = 45,
         diag = FALSE,
         col = colorRampPalette(c("#6D9EC1", "white", "#E46726"))(200))
```




```
# 2. Knowledge Profile Clustering
# Standardize knowledge variables
```

```r
knowledge_vars <- combined_df %>%
  select(numeracy_score, relative_energy_use_score,
         relative_energy_save_score, els) %>%
  scale()

# Determine optimal number of clusters
set.seed(123)
wss <- sapply(1:10, function(k) {
  kmeans(knowledge_vars, centers=k)$tot.withinss
})

# Perform k-means clustering
k <- 3  # Based on elbow plot inspection
clusters <- kmeans(knowledge_vars, centers=k)

# Add cluster membership to data
combined_df$knowledge_cluster <- as.factor(clusters$cluster)

# Visualize clusters
pca_result <- prcomp(knowledge_vars)
cluster_df <- data.frame(
  PC1 = pca_result$x[,1],
  PC2 = pca_result$x[,2],
  Cluster = combined_df$knowledge_cluster
)

# Create cluster visualization
p_clusters <- ggplot(cluster_df, aes(x=PC1, y=PC2, color=Cluster)) +
  geom_point(alpha=0.6) +
  theme_minimal() +
  labs(title="Knowledge Profiles Clustering",
       x="First Principal Component",
       y="Second Principal Component")
p_clusters
```



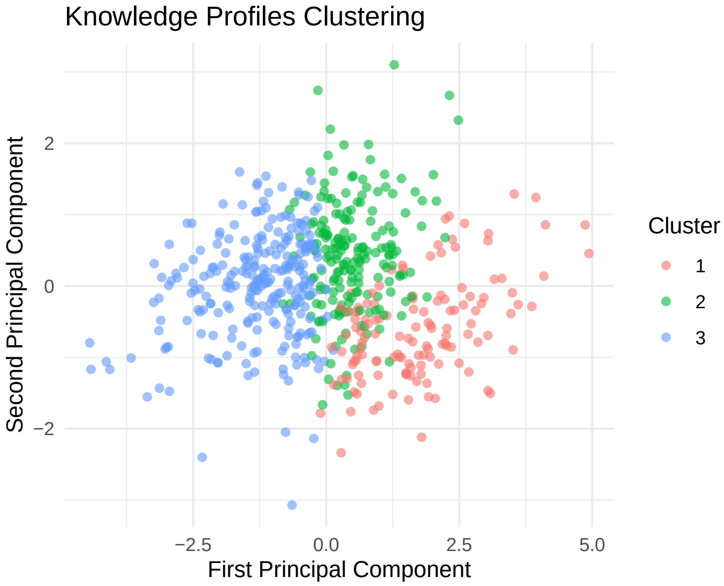

```
# 3. Non-linear GAM Analysis
gam_model <- gam(els ~ s(env_attitude) + s(perceived_difficulty_score),
                 data=combined_df)


# Create prediction grid for GAM visualization
env_grid <- seq(min(combined_df$env_attitude, na.rm=TRUE),
                max(combined_df$env_attitude, na.rm=TRUE),
                length.out=100)
diff_grid <- seq(min(combined_df$perceived_difficulty_score, na.rm=TRUE),
                 max(combined_df$perceived_difficulty_score, na.rm=TRUE),
                 length.out=100)


# Predict ELS scores
pred_env <- predict(gam_model,
```

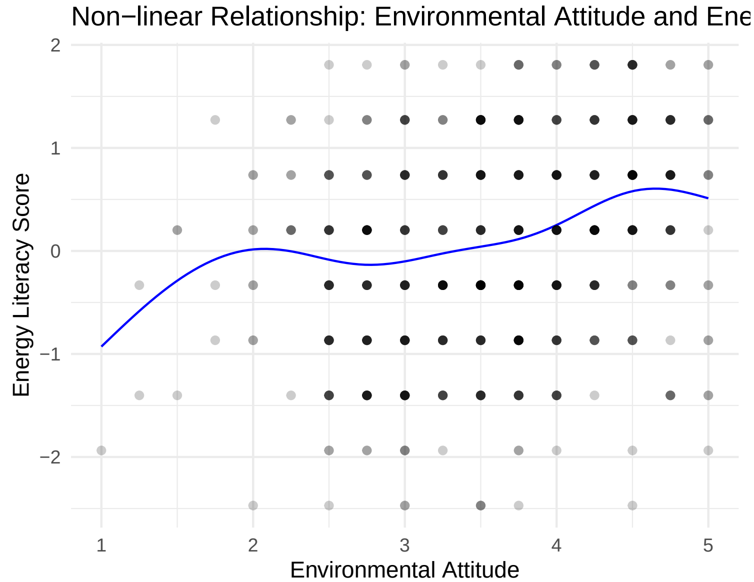
```
                    newdata=data.frame(env_attitude=env_grid,
                                       perceived_difficulty_score=mean(combined_df$perceived_difficulty_score, na
pred_diff <- predict(gam_model,
                     newdata=data.frame(perceived_difficulty_score=diff_grid,
                                        env_attitude=mean(combined_df$env_attitude, na.rm=TRUE)))

# Create GAM plots
p_gam_env <- ggplot() +
  geom_line(aes(x=env_grid, y=pred_env), color="blue") +
  geom_point(data=combined_df, aes(x=env_attitude, y=els), alpha=0.2) +
  theme_minimal() +
  labs(title="Non-linear Relationship: Environmental Attitude and Energy Literacy",
       x="Environmental Attitude",
       y="Energy Literacy Score")

p_gam_diff <- ggplot() +
  geom_line(aes(x=diff_grid, y=pred_diff), color="red") +
  geom_point(data=combined_df, aes(x=perceived_difficulty_score, y=els), alpha=0.2) +
  theme_minimal() +
  labs(title="Non-linear Relationship: Perceived Difficulty and Energy Literacy",
       x="Perceived Difficulty Score",
       y="Energy Literacy Score")

#p_gam_diff



# 4. Knowledge-Motivation Interaction Analysis
interaction_model <- lm(els ~ env_attitude * perceived_difficulty_score +
                         numeracy_score, data=combined_df)

summary(interaction_model)
```

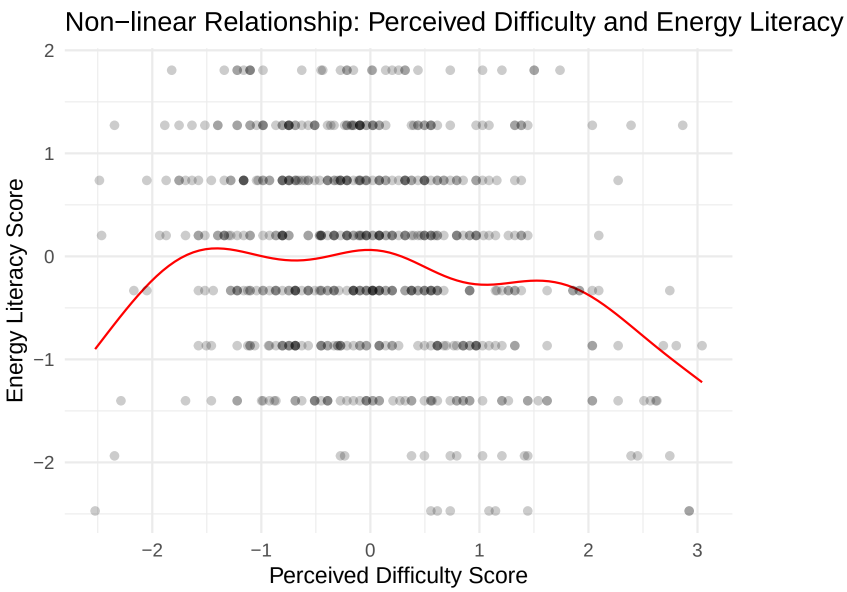
```
Call:
lm(formula = els ~ env_attitude * perceived_difficulty_score +
    numeracy_score, data = combined_df)

Residuals:
    Min      1Q  Median      3Q     Max
-3.0152 -0.6285  0.0088  0.6599  2.2174
```

```
Coefficients:
                                       Estimate Std. Error t value
(Intercept)                             -0.7906     0.1966   -4.02
env_attitude                             0.2192     0.0538    4.07
perceived_difficulty_score              -0.0550     0.1782   -0.31
numeracy_score                           0.2909     0.0387    7.52
env_attitude:perceived_difficulty_score -0.0177     0.0488   -0.36
                                                Pr(>|t|)
(Intercept)                             0.00006570557947 ***
env_attitude                            0.00005257587373 ***
perceived_difficulty_score                          0.76
numeracy_score                          0.00000000000021 ***
env_attitude:perceived_difficulty_score             0.72
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.91 on 581 degrees of freedom
Multiple R-squared:  0.172, Adjusted R-squared:  0.166
F-statistic: 30.2 on 4 and 581 DF,  p-value: <0.0000000000000002
```

```r
# Create interaction plot data
env_levels <- quantile(combined_df$env_attitude, probs=c(0.25, 0.75), na.rm=TRUE)
diff_seq <- seq(min(combined_df$perceived_difficulty_score, na.rm=TRUE),
                max(combined_df$perceived_difficulty_score, na.rm=TRUE),
                length.out=100)

# Arrange all plots
grid.arrange(p_clusters, p_gam_env, p_gam_diff, ncol=2)
```








```r
# Print statistical summaries
summary(gam_model)
```

```
Family: gaussian
Link function: identity

Formula:
els ~ s(env_attitude) + s(perceived_difficulty_score)

Parametric coefficients:
                        Estimate          Std. Error t value Pr(>|t|)
(Intercept) 0.0000000000000488 0.03867979305614088       0        1

Approximate significance of smooth terms:
```

```
                               edf Ref.df    F   p-value
s(env_attitude)               5.44   6.52 6.19 0.0000016 ***
s(perceived_difficulty_score) 6.28   7.42 3.76   0.00043 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.123   Deviance explained = 14.1%
GCV = 0.89618  Scale est. = 0.87673   n = 586
```

```
summary(interaction_model)
```

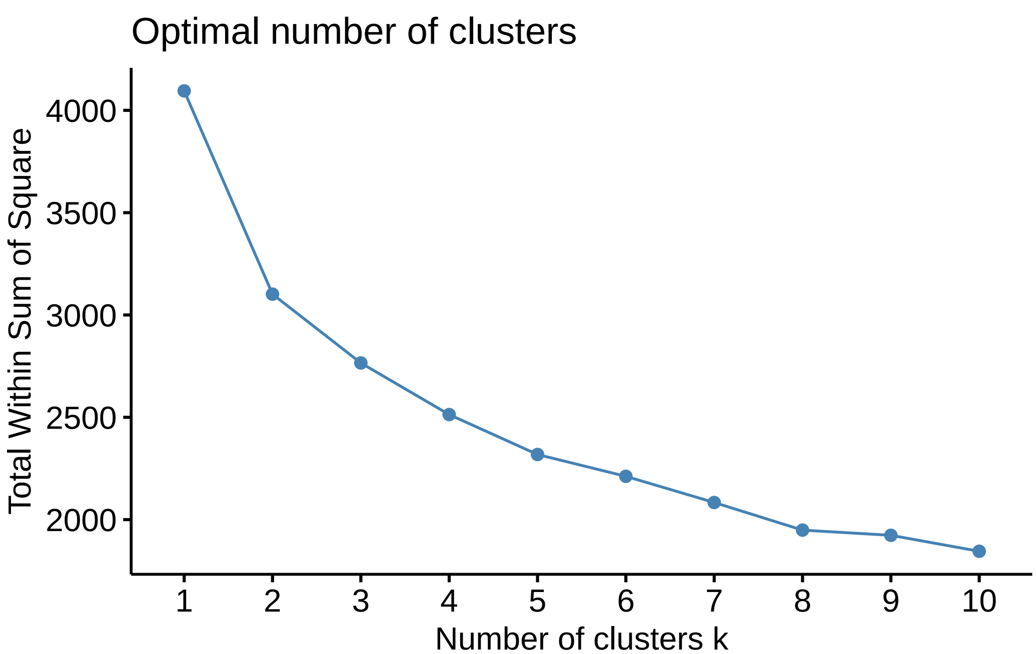
```

Call:
lm(formula = els ~ env_attitude * perceived_difficulty_score +
    numeracy_score, data = combined_df)

Residuals:
    Min      1Q  Median      3Q     Max
-3.0152 -0.6285  0.0088  0.6599  2.2174

Coefficients:
                                        Estimate Std. Error t value
(Intercept)                              -0.7906     0.1966   -4.02
env_attitude                              0.2192     0.0538    4.07
perceived_difficulty_score               -0.0550     0.1782   -0.31
numeracy_score                            0.2909     0.0387    7.52
env_attitude:perceived_difficulty_score  -0.0177     0.0488   -0.36
                                                Pr(>|t|)
(Intercept)                             0.00006570557947 ***
env_attitude                            0.00005257587373 ***
perceived_difficulty_score                          0.76
numeracy_score                          0.00000000000021 ***
env_attitude:perceived_difficulty_score             0.72
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.91 on 581 degrees of freedom
Multiple R-squared:  0.172, Adjusted R-squared:  0.166
F-statistic: 30.2 on 4 and 581 DF,  p-value: <0.0000000000000002
```

```r
# Cluster profile analysis
cluster_profiles <- combined_df %>%
  group_by(knowledge_cluster) %>%
  summarise(
    mean_numeracy = mean(numeracy_score, na.rm=TRUE),
    mean_energy_use = mean(relative_energy_use_score, na.rm=TRUE),
    mean_energy_save = mean(relative_energy_save_score, na.rm=TRUE),
    mean_els = mean(els, na.rm=TRUE),
    mean_env_attitude = mean(env_attitude, na.rm=TRUE),
    mean_difficulty = mean(perceived_difficulty_score, na.rm=TRUE),
    n = n()
  )

print(cluster_profiles)
```

```
# A tibble: 3 x 8
  knowledge_cluster mean_numeracy mean_energy_use mean_energy_save mean_els
  <fct>                     <dbl>           <dbl>            <dbl>    <dbl>
1 1                        -1.43           -0.670           -0.679   -0.790
2 2                         0.361          -0.386           -0.470   -0.463
3 3                         0.459           0.635            0.705    0.756
# i 3 more variables: mean_env_attitude <dbl>, mean_difficulty <dbl>, n <int>
```

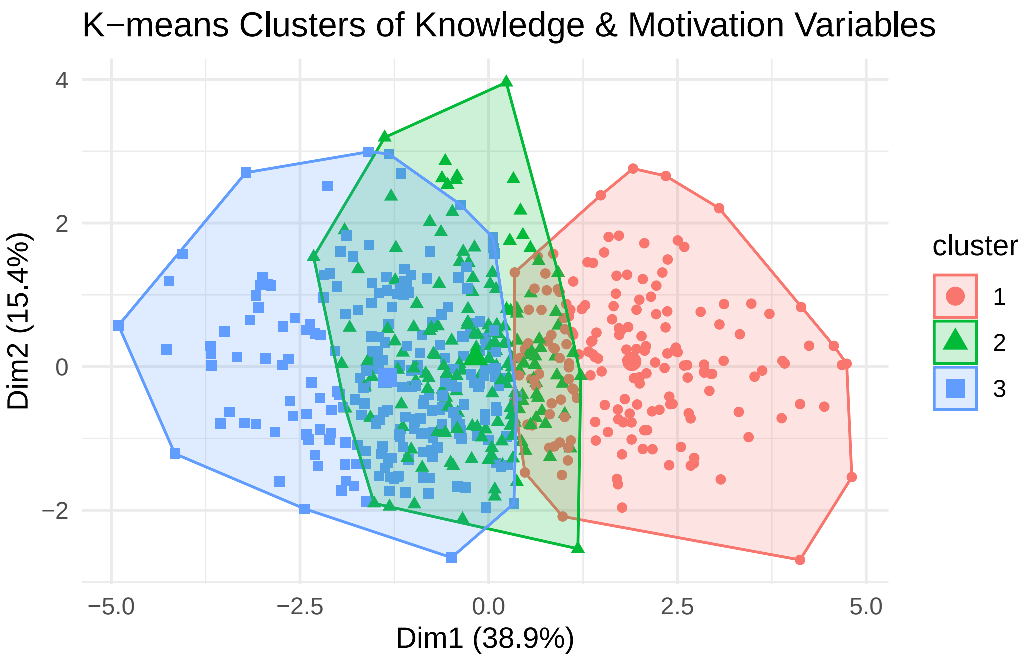
3a

```r
# Example: K-means clustering on knowledge + motivation
# Subset your knowledge & motivation columns
cluster_data <- combined_df %>%
  select(numeracy_score, relative_energy_use_score, relative_energy_save_score,
         els, perceived_difficulty_score, env_attitude, pol_conservatism) %>%
  na.omit()

# Scale them
cluster_data_scaled <- scale(cluster_data)

# Decide on number of clusters (e.g. 2-5) - use e.g. Elbow method
fviz_nbclust(cluster_data_scaled, kmeans, method = "wss")
```

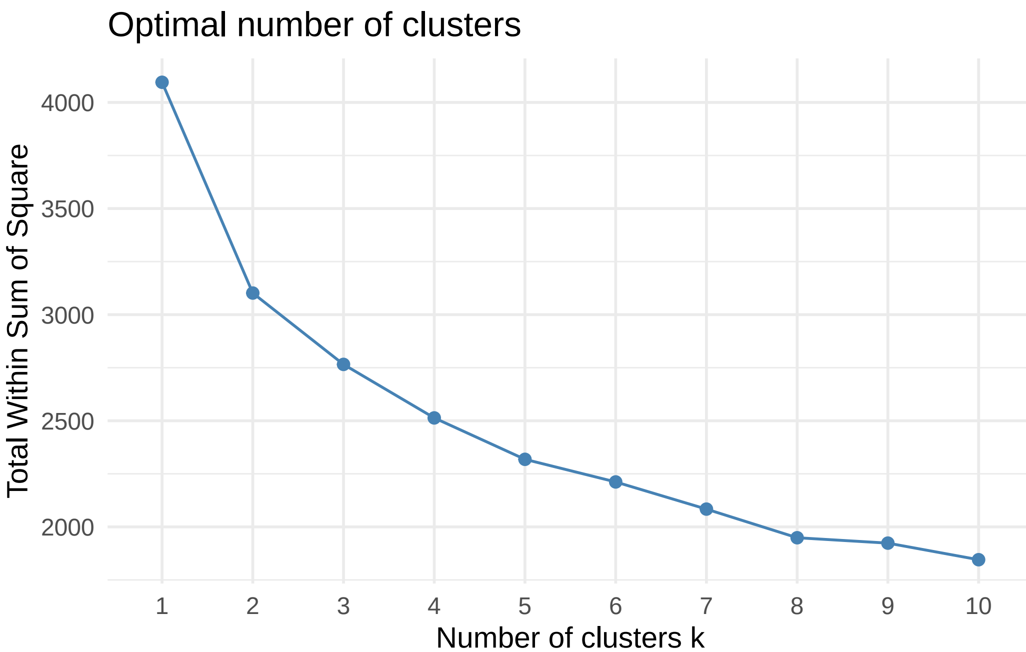



```r
# Suppose we choose 3 clusters as a demonstration
set.seed(123)
km_result <- kmeans(cluster_data_scaled, centers = 3, nstart = 25)

# Add cluster membership back into the original data
cluster_data$cluster <- factor(km_result$cluster)

# Visualize clusters in 2D (using PCA behind the scenes)
fviz_cluster(km_result, data = cluster_data_scaled,
             geom = "point", ellipse.type = "convex") +
  theme_minimal() +
```

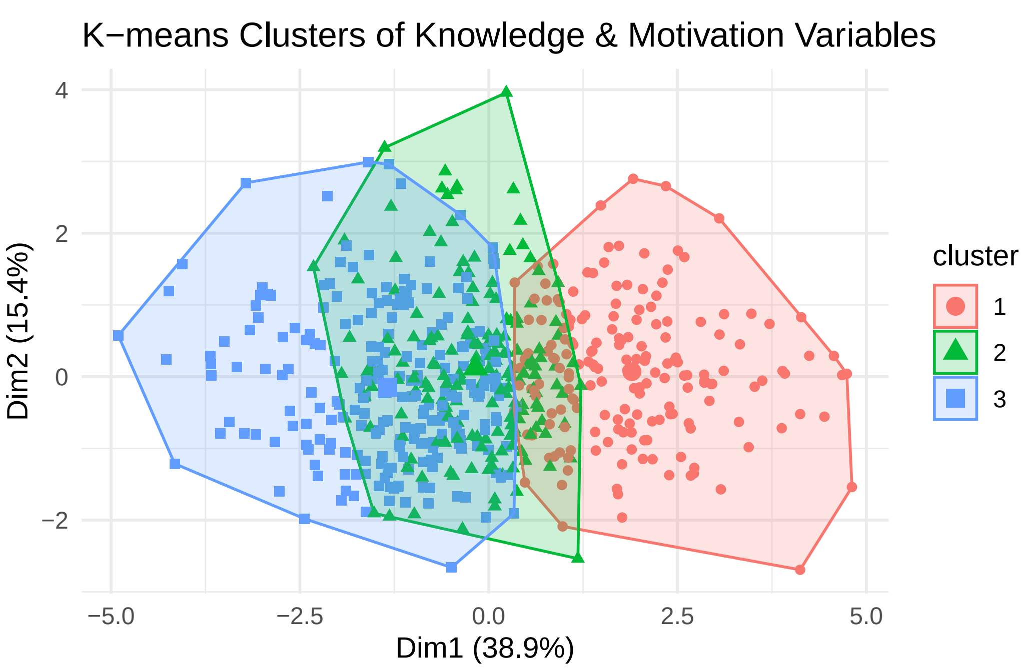
```
  labs(title = "K-means Clusters of Knowledge & Motivation Variables")
```




```
# 1. Grab relevant variables
cluster_data <- combined_df %>%
  select(numeracy_score, relative_energy_use_score, relative_energy_save_score,
         els, perceived_difficulty_score, env_attitude, pol_conservatism) %>%
  na.omit()

# 2. Standardize/scale them
cluster_data_scaled <- scale(cluster_data)

# 3. Determine the optimal number of clusters (Elbow or Silhouette methods)
fviz_nbclust(cluster_data_scaled, kmeans, method = "wss") +
  theme_minimal()
```



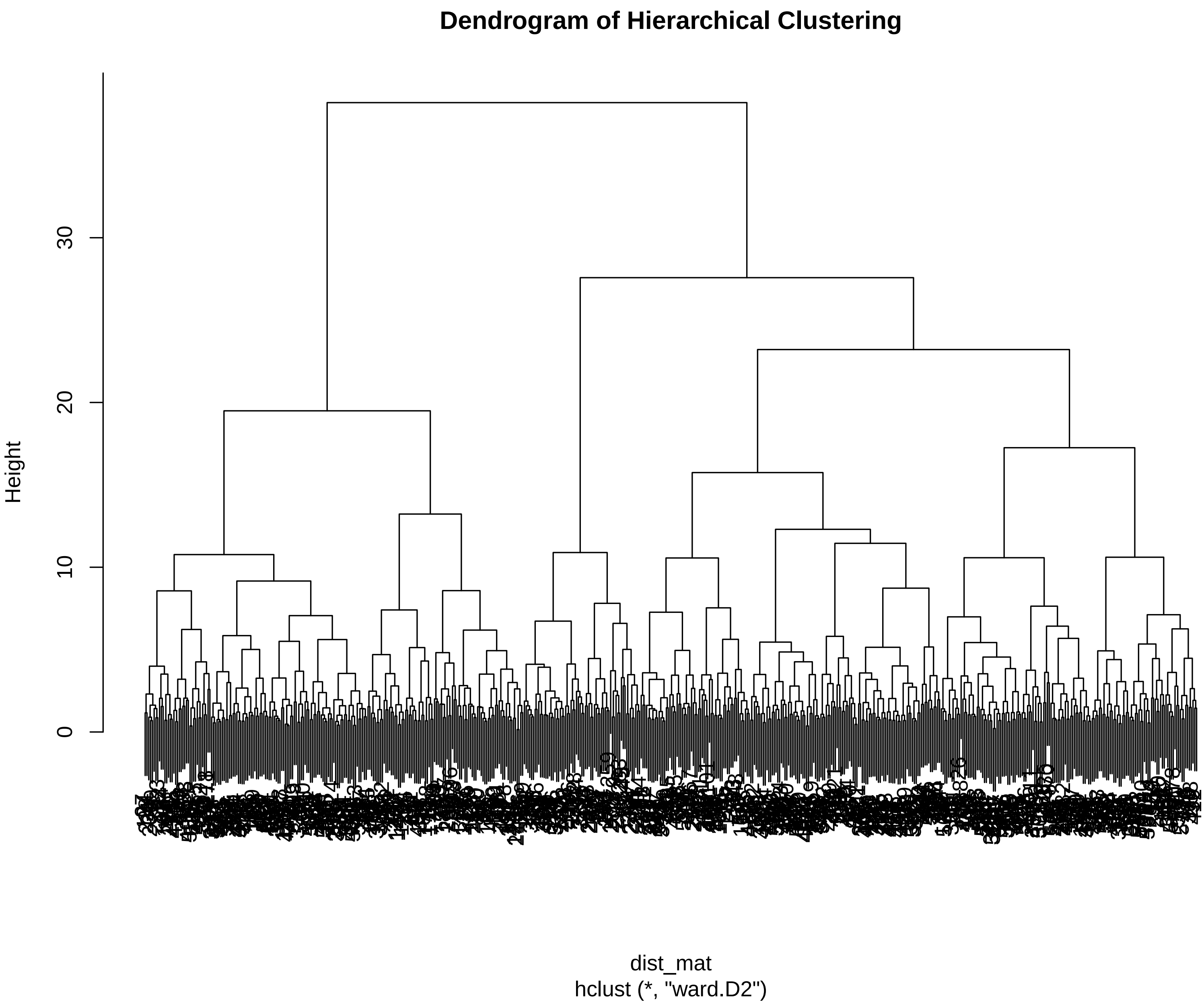

```
# 4. Run k-means with your chosen number of clusters (say k = 3)
set.seed(123)
km_res <- kmeans(cluster_data_scaled, centers = 3, nstart = 25)

# 5. Visualize
fviz_cluster(km_res, data = cluster_data_scaled,
             geom = "point", ellipse.type = "convex") +
  theme_minimal() +
  labs(title = "K-means Clusters of Knowledge & Motivation Variables")
```



```r
# 6. Inspect cluster means
cluster_centers <- as.data.frame(km_res$centers)
colnames(cluster_centers) <- colnames(cluster_data)
cluster_centers
```

```
  numeracy_score relative_energy_use_score relative_energy_save_score   els
1          -0.80                     -0.74                      -0.76 -0.89
2           0.19                      0.12                       0.13  0.33
3           0.48                      0.48                       0.49  0.44
  perceived_difficulty_score env_attitude pol_conservatism
1                       0.69      -0.6788             0.41
2                      -0.35       0.0008             0.84
3                      -0.27       0.5253            -0.96
```

```r
# Example of hierarchical clustering if that is preferred
dist_mat <- dist(cluster_data_scaled, method = "euclidean")
hc_res <- hclust(dist_mat, method = "ward.D2")
plot(hc_res, main = "Dendrogram of Hierarchical Clustering")
```




```r
# Cut tree at chosen k
clusters <- cutree(hc_res, k = 3)
table(clusters)
```

```
clusters
  1   2   3
211  65 310
```

```r
# Example mediation: knowledge -> perceived_difficulty -> env_attitude
model_mediation <- '
  # direct effect
  env_attitude ~ c*els
  # mediator
  perceived_difficulty_score ~ a*els
  env_attitude ~ b*perceived_difficulty_score
  # indirect effect
  ab := a*b
  # total effect
  total := c + (a*b)
'

fit_mediation <- sem(model_mediation, data = combined_df, missing="fiml")
summary(fit_mediation, fit.measures=TRUE, standardized=TRUE, rsquare=TRUE)
```

```
lavaan 0.6-19 ended normally after 1 iteration

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                         7

  Number of observations                           586
  Number of missing patterns                         1

Model Test User Model:

  Test statistic                                 0.000
  Degrees of freedom                                 0

Model Test Baseline Model:

  Test statistic                               149.690
  Degrees of freedom                                 3
  P-value                                        0.000

User Model versus Baseline Model:

  Comparative Fit Index (CFI)                    1.000
  Tucker-Lewis Index (TLI)                       1.000
```

```
  Robust Comparative Fit Index (CFI)             1.000
  Robust Tucker-Lewis Index (TLI)                1.000

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -1434.548
  Loglikelihood unrestricted model (H1)      -1434.548

  Akaike (AIC)                                2883.095
  Bayesian (BIC)                              2913.709
  Sample-size adjusted Bayesian (SABIC)       2891.486

Root Mean Square Error of Approximation:

  RMSEA                                          0.000
  90 Percent confidence interval - lower         0.000
  90 Percent confidence interval - upper         0.000
  P-value H_0: RMSEA <= 0.050                       NA
  P-value H_0: RMSEA >= 0.080                       NA

  Robust RMSEA                                   0.000
  90 Percent confidence interval - lower         0.000
  90 Percent confidence interval - upper         0.000
  P-value H_0: Robust RMSEA <= 0.050                NA
  P-value H_0: Robust RMSEA >= 0.080                NA

Standardized Root Mean Square Residual:

  SRMR                                           0.000

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Observed
  Observed information based on                Hessian

Regressions:
                               Estimate  Std.Err  z-value  P(>|z|)   Std.lv
  env_attitude ~
```

```
    els        (c)               0.152    0.029    5.151    0.000    0.152
  perceived_difficulty_score ~
    els        (a)              -0.222    0.040   -5.506    0.000   -0.222
  env_attitude ~
    prcvd_dff_ (b)              -0.263    0.029   -8.934    0.000   -0.263
  Std.all

    0.197

   -0.222

   -0.342

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .env_attitude      3.583    0.029  124.796    0.000    3.583    4.653
   .prcvd_dffclty_    0.000    0.040    0.000    1.000    0.000    0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .env_attitude      0.483    0.028   17.117    0.000    0.483    0.815
   .prcvd_dffclty_    0.949    0.055   17.117    0.000    0.949    0.951

R-Square:
                   Estimate
    env_attitude      0.185
    prcvd_dffclty_    0.049

Defined Parameters:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    ab                0.058    0.012    4.687    0.000    0.058    0.076
    total             0.210    0.031    6.862    0.000    0.210    0.273
```

4a

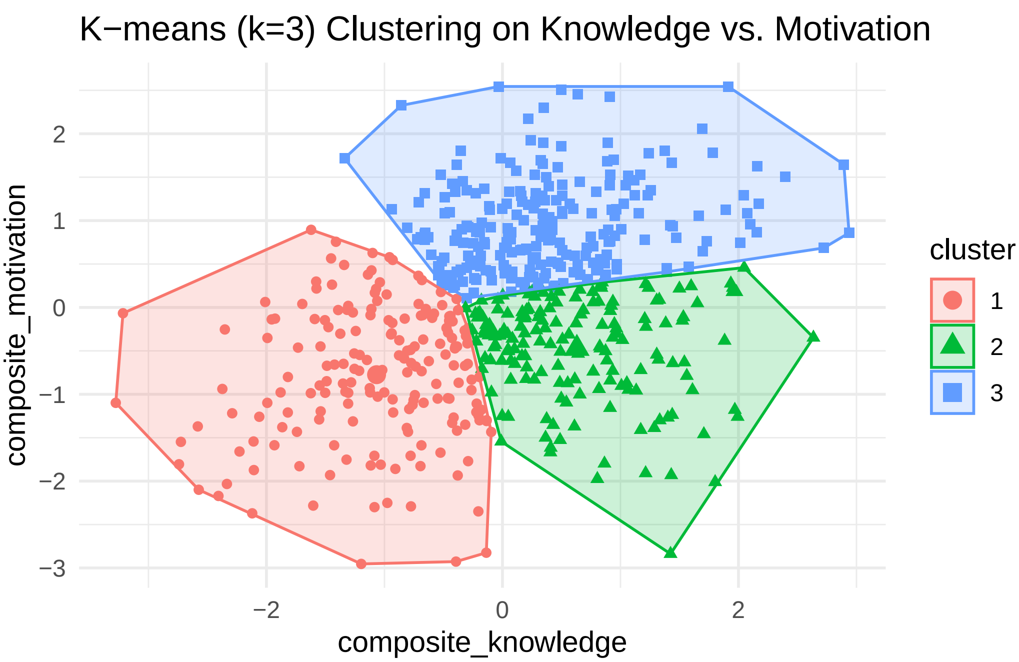
```
combined_scores <- combined_df %>%
  mutate(
    # Example composite for knowledge: average of (z-scored) numeracy,
    # energy_use, energy_save, ELS.
    # (You can also sum them, but average is convenient.)
    composite_knowledge = rowMeans(
      cbind(numeracy_score, relative_energy_use_score,
            relative_energy_save_score, els),
      na.rm = FALSE  # If a row has missing for any item, result = NA
    ),

    # Example composite for motivation:
    # env_attitude might be already in a favorable direction, but if
    # perceived_difficulty is "difficulty," consider reversing so that
    # higher = "less difficulty" = "higher motivation."
    # For example: reverse_diff = (-1)*perceived_difficulty_score
    # Then average with env_attitude (if you want them combined).
    # If you are including pol_conservatism as well, you must decide
    # how to handle that in the composite. Possibly reverse-coded
    # so that higher # = more liberal or more "pro-environment" stance.
    # (It's your theoretical call.)

    # For now, let's do a small composite with environmental attitude
    # and reversed difficulty:
    reverse_diff = -1 * perceived_difficulty_score,

    composite_motivation = rowMeans(
      cbind(env_attitude, reverse_diff),
      na.rm = FALSE
    )
  )

# We'll create a small data frame with just the two composites,
# removing any incomplete cases
cluster_data <- combined_scores %>%
  select(composite_knowledge, composite_motivation) %>%
  na.omit()
```

```r
# Decide on number of clusters "k". Let's try k = 3:
set.seed(123)
km3 <- kmeans(cluster_data, centers = 3, nstart = 25)

# Inspect results
km3
```

```
K-means clustering with 3 clusters of sizes 184, 167, 235

Cluster means:
  composite_knowledge composite_motivation
1               -0.79                  1.2
2                0.47                  1.4
3                0.29                  2.5

Clustering vector:
  [1] 2 3 1 3 2 3 3 1 2 3 3 3 1 3 1 1 1 1 1 3 3 2 1 3 3 3 2 3 1 1 2 2 2 3 2 2 2
 [38] 1 3 2 1 3 2 1 2 3 3 1 1 1 2 2 3 1 3 2 3 3 1 3 2 3 1 2 3 2 1 3 1 3 3 1 3 2
 [75] 2 3 1 3 1 3 3 3 3 2 2 2 1 2 3 1 2 3 3 3 2 3 1 3 2 3 3 3 3 2 2 2 3 3 3 3 2
[112] 3 2 3 3 1 3 3 2 2 2 1 3 3 2 2 3 3 2 2 3 3 3 3 1 1 3 3 3 1 3 1 3 2 3 3 2 2
[149] 2 3 3 2 2 1 3 3 1 2 2 1 2 3 3 2 3 1 3 3 2 2 3 2 2 1 2 3 1 3 2 2 2 3 3 1 2
[186] 1 2 2 3 3 3 2 1 2 2 2 3 2 3 2 3 2 3 1 1 1 3 3 3 2 1 2 3 1 2 1 3 2 2 3 2 2
[223] 3 1 3 1 2 2 3 2 1 1 2 2 3 2 1 3 1 1 1 2 2 1 1 1 3 1 3 1 3 1 1 2 1 2 1 1 1
[260] 3 1 3 1 3 3 1 1 1 3 1 1 1 1 1 3 3 1 2 1 1 1 1 1 1 1 3 1 3 2 1 1 3 1 2 2 2
[297] 2 3 2 2 3 2 1 3 3 1 2 1 1 1 1 2 2 3 2 1 3 1 1 2 3 3 2 2 3 3 1 1 1 2 2 1 3
[334] 1 3 3 3 1 1 1 2 2 2 1 3 3 3 1 1 3 1 2 2 3 2 2 1 1 1 1 1 3 3 1 2 3 3 2 1 3
[371] 1 2 3 3 3 2 1 3 2 3 3 3 3 2 1 3 2 3 3 3 1 1 1 1 2 2 1 3 3 3 3 2 3 1 1 3 1
[408] 3 2 2 2 1 2 2 3 1 2 3 3 2 3 1 2 3 1 1 3 3 1 1 3 2 2 1 3 1 3 2 2 3 2 3 1 1
[445] 2 1 1 2 3 3 2 1 3 1 2 2 2 2 3 1 1 2 2 3 3 2 3 3 1 3 2 3 1 3 3 1 2 3 3 1 3
[482] 3 3 2 3 3 2 2 3 1 1 1 2 1 3 1 1 3 1 1 3 1 3 3 3 1 3 3 3 3 3 3 1 2 2 3 3 1
[519] 3 1 2 3 2 1 2 2 1 2 3 3 3 3 3 3 1 3 3 3 3 1 3 1 3 2 1 1 2 3 3 1 2 1 3 1 1
[556] 2 1 1 1 1 3 1 3 2 1 3 3 2 2 3 3 3 2 1 3 3 1 1 1 3 3 2 3 3 3 2

Within cluster sum of squares by cluster:
[1] 103  67 108
 (between_SS / total_SS =  57.0 %)

Available components:
```

```
[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

```
# Visualize
fviz_cluster(km3, data = cluster_data,
             geom = "point", ellipse.type = "convex") +
  theme_minimal() +
  labs(title = "K-means (k=3) Clustering on Knowledge vs. Motivation")
```

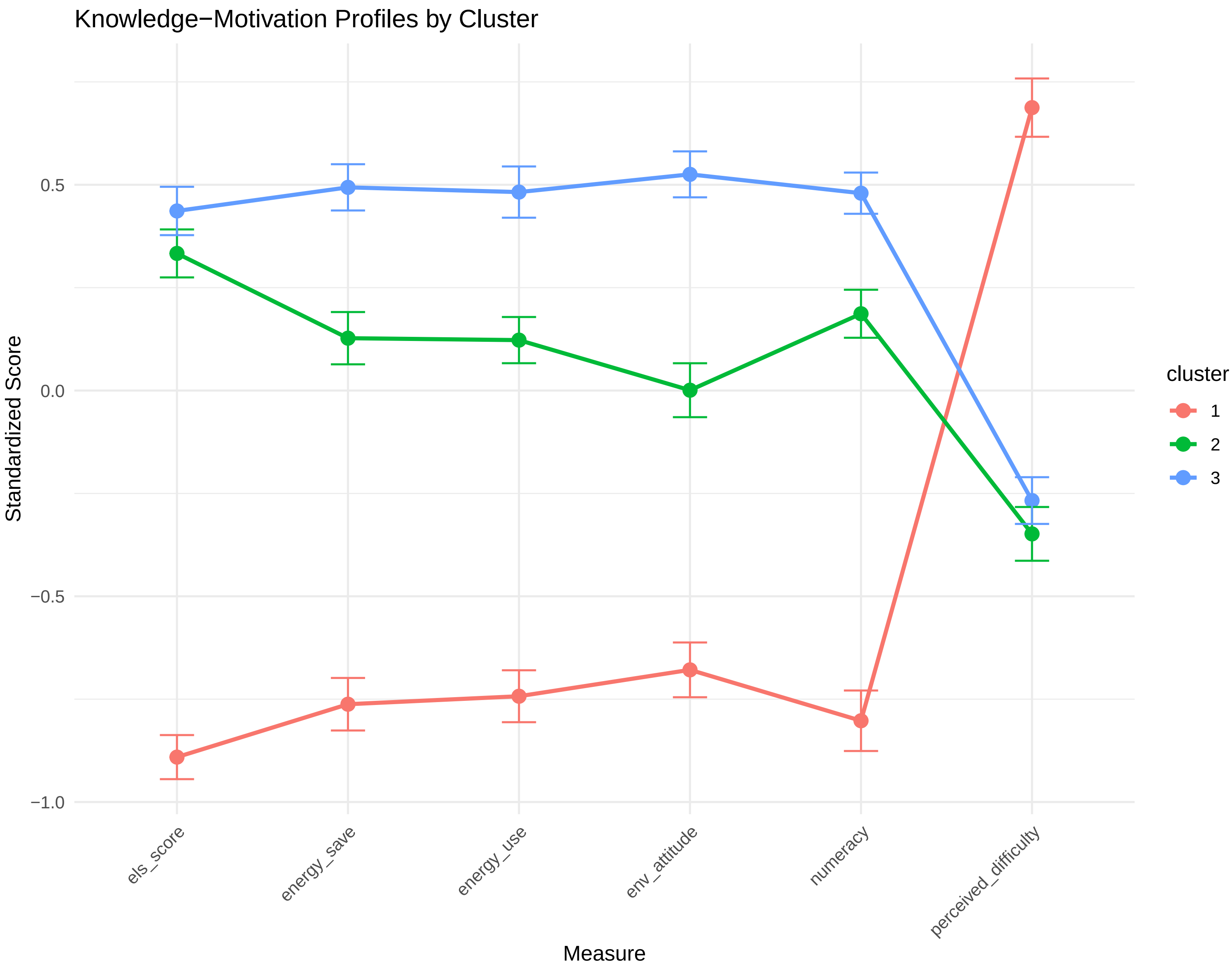



```
combined_scores$cluster <- factor(km3$cluster)

# Compare mean knowledge & motivation by cluster
combined_scores %>%
  group_by(cluster) %>%
  summarize(
    n = n(),
    mean_knowledge = mean(composite_knowledge, na.rm = TRUE),
    mean_motivation = mean(composite_motivation, na.rm = TRUE)
  )
```

```
# A tibble: 3 x 4
  cluster     n mean_knowledge mean_motivation
  <fct>   <int>          <dbl>           <dbl>
1 1         184         -0.794            1.22
2 2         167          0.469            1.44
3 3         235          0.288            2.49
```

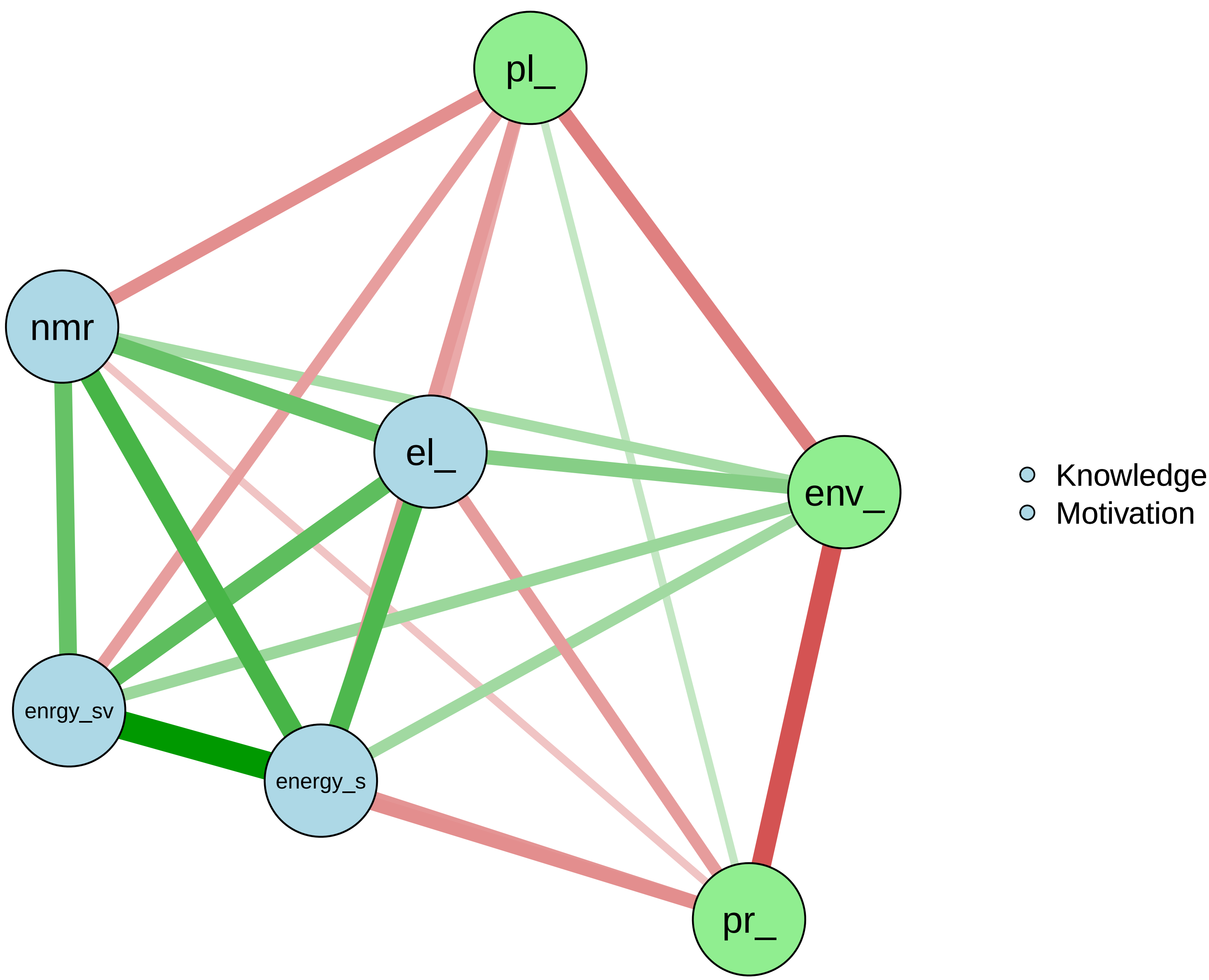
5a

```
combined_scores <- attari1 %>%
  left_join(attari2_scores, by="id") %>%
  left_join(els_scores, by="id") %>%
  left_join(rs_scores, by="id")

# Rename columns for clarity
names(combined_scores) <- c("id", "perceived_difficulty", "numeracy",
                            "energy_use", "energy_save",
                            "els_accuracy", "els_score",
                            "env_attitude", "env_attitude_z",
                            "pol_conservatism", "pol_conservatism_z")

combined_scores$cluster <- as.factor(km_result$cluster)

# Create composite knowledge score
combined_scores$composite_knowledge <- rowMeans(combined_scores[, c("numeracy", "energy_use", "energy_save", "

# Create standardized scores for profile analysis
profile_data <- combined_scores %>%
  select(id, cluster, numeracy, energy_use, energy_save,
         els_score, env_attitude, perceived_difficulty) %>%
  gather(measure, value, -id, -cluster) %>%
  group_by(measure) %>%
  mutate(z_score = scale(value)[,1]) %>%
  ungroup()

# Create profile plot
ggplot(profile_data, aes(x = measure, y = z_score, color = cluster, group = cluster)) +
  stat_summary(fun = mean, geom = "line", size = 1) +
  stat_summary(fun = mean, geom = "point", size = 3) +
  stat_summary(fun.data = mean_se, geom = "errorbar", width = 0.2) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Knowledge-Motivation Profiles by Cluster",
       x = "Measure", y = "Standardized Score")
```




```
# 2. Canonical Correlation Analysis between Knowledge and Motivation Sets
# Prepare matrices
knowledge_vars <- combined_scores %>% select(numeracy, energy_use, energy_save, els_score) %>%
  as.matrix()

motivation_vars <- combined_scores %>%
  select(env_attitude, perceived_difficulty, pol_conservatism) %>%
  as.matrix()

# Perform CCA
cc_result <- cancor(knowledge_vars, motivation_vars)

# 3. Network Analysis to Visualize Variable Relationships

# Create correlation matrix
```

```r
cor_matrix <- cor(combined_scores %>%
                    select(numeracy, energy_use, energy_save, els_score,
                           env_attitude, perceived_difficulty, pol_conservatism),
                  use = "pairwise.complete.obs")

# Create network plot
qgraph(cor_matrix,
       layout = "spring",
       groups = list(Knowledge = 1:4, Motivation = 5:7),
       color = c(rep("lightblue", 4), rep("lightgreen", 3)))
```




```r
# 4. Mixed Effects Model to Account for Potential Group-Level Effects
mixed_model <- lmer(els_score ~ env_attitude + perceived_difficulty +
                      (1|cluster), data = combined_scores)
summary(mixed_model)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: els_score ~ env_attitude + perceived_difficulty + (1 | cluster)
   Data: combined_scores

REML criterion at convergence: 1432

Scaled residuals:
   Min     1Q Median     3Q    Max
-3.446 -0.743  0.019  0.716  2.054

Random effects:
 Groups   Name        Variance Std.Dev.
 cluster  (Intercept) 0.583    0.763
 Residual             0.650    0.806
Number of obs: 586, groups:  cluster, 3

Fixed effects:
                     Estimate Std. Error t value
(Intercept)           -0.1027     0.4782   -0.21
env_attitude           0.0172     0.0515    0.33
perceived_difficulty   0.0600     0.0386    1.55

Correlation of Fixed Effects:
            (Intr) env_tt
env_attitud -0.382
prcvd_dffcl -0.095  0.244
```

```
# 5. Structural Equation Model for Path Analysis

# Define model
model <- '
  # Measurement model
  knowledge =~ numeracy + energy_use + energy_save + els_score
  motivation =~ env_attitude + perceived_difficulty + pol_conservatism

  # Structural model
  knowledge ~ motivation
'
```

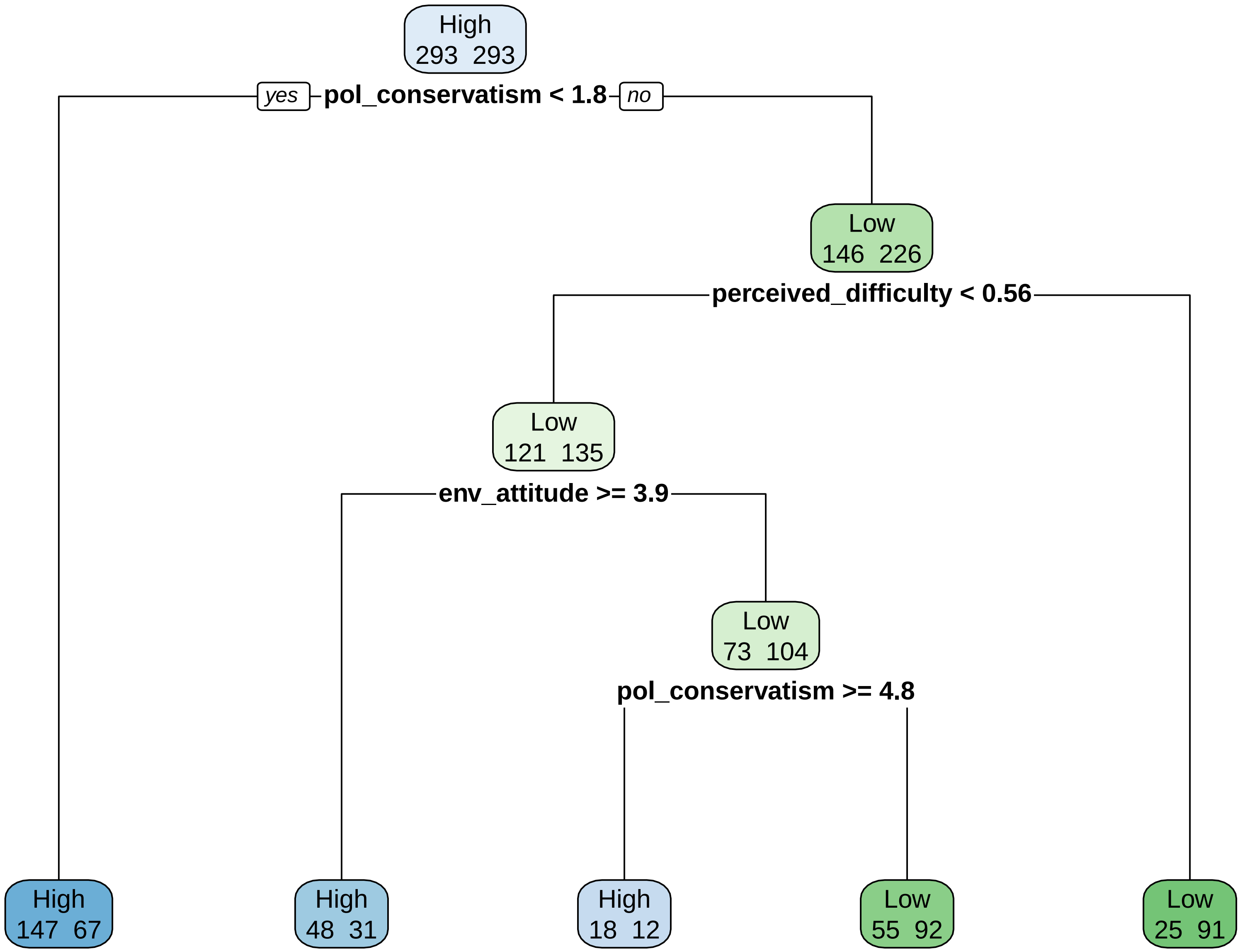
```r
# Fit model
fit <- sem(model, data = combined_scores)
summary(fit, standardized = TRUE, fit.measures = TRUE)
```

```
lavaan 0.6-19 ended normally after 36 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                        15

  Number of observations                           586

Model Test User Model:

  Test statistic                                48.061
  Degrees of freedom                                13
  P-value (Chi-square)                           0.000

Model Test Baseline Model:

  Test statistic                               765.733
  Degrees of freedom                                21
  P-value                                        0.000

User Model versus Baseline Model:

  Comparative Fit Index (CFI)                    0.953
  Tucker-Lewis Index (TLI)                       0.924

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -5510.805
  Loglikelihood unrestricted model (H1)      -5486.775

  Akaike (AIC)                               11051.610
  Bayesian (BIC)                             11117.210
  Sample-size adjusted Bayesian (SABIC)      11069.590

Root Mean Square Error of Approximation:
```

```
  RMSEA                                          0.068
  90 Percent confidence interval - lower         0.048
  90 Percent confidence interval - upper         0.089
  P-value H_0: RMSEA <= 0.050                    0.068
  P-value H_0: RMSEA >= 0.080                    0.180

Standardized Root Mean Square Residual:

  SRMR                                           0.045

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  knowledge =~
    numeracy          1.000                               0.534    0.534
    energy_use        1.441    0.131   10.978    0.000    0.769    0.769
    energy_save       1.331    0.123   10.786    0.000    0.710    0.711
    els_score         1.024    0.109    9.393    0.000    0.546    0.547
  motivation =~
    env_attitude      1.000                               0.506    0.657
    percvd_dffclty   -1.068    0.142   -7.515    0.000   -0.540   -0.541
    pol_conservtsm   -1.159    0.175   -6.635    0.000   -0.586   -0.413

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  knowledge ~
    motivation        0.609    0.094    6.497    0.000    0.577    0.577

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .numeracy          0.714    0.047   15.120    0.000    0.714    0.715
   .energy_use        0.408    0.042    9.619    0.000    0.408    0.408
   .energy_save       0.494    0.042   11.665    0.000    0.494    0.495
   .els_score         0.700    0.047   14.976    0.000    0.700    0.701
   .env_attitude      0.337    0.037    9.074    0.000    0.337    0.569
```

```
 .percvd_dffclty    0.706    0.056   12.591    0.000    0.706    0.708
 .pol_conservtsm    1.668    0.111   14.976    0.000    1.668    0.829
 .knowledge         0.190    0.034    5.553    0.000    0.667    0.667
  motivation        0.256    0.042    6.021    0.000    1.000    1.000
```

```
# 6. Classification Tree for Predicting Knowledge Levels - rpart functions
# Create binary knowledge indicator (high/low) based on median split
combined_scores$knowledge_level <- factor(ifelse(combined_scores$composite_knowledge >
                                               median(combined_scores$composite_knowledge, na.rm = TRUE),
                                               "High", "Low"))

# Fit tree
tree_model <- rpart(knowledge_level ~ env_attitude + perceived_difficulty +
                      pol_conservatism, data = combined_scores)

# Plot tree
rpart.plot(tree_model, extra = 1)
```

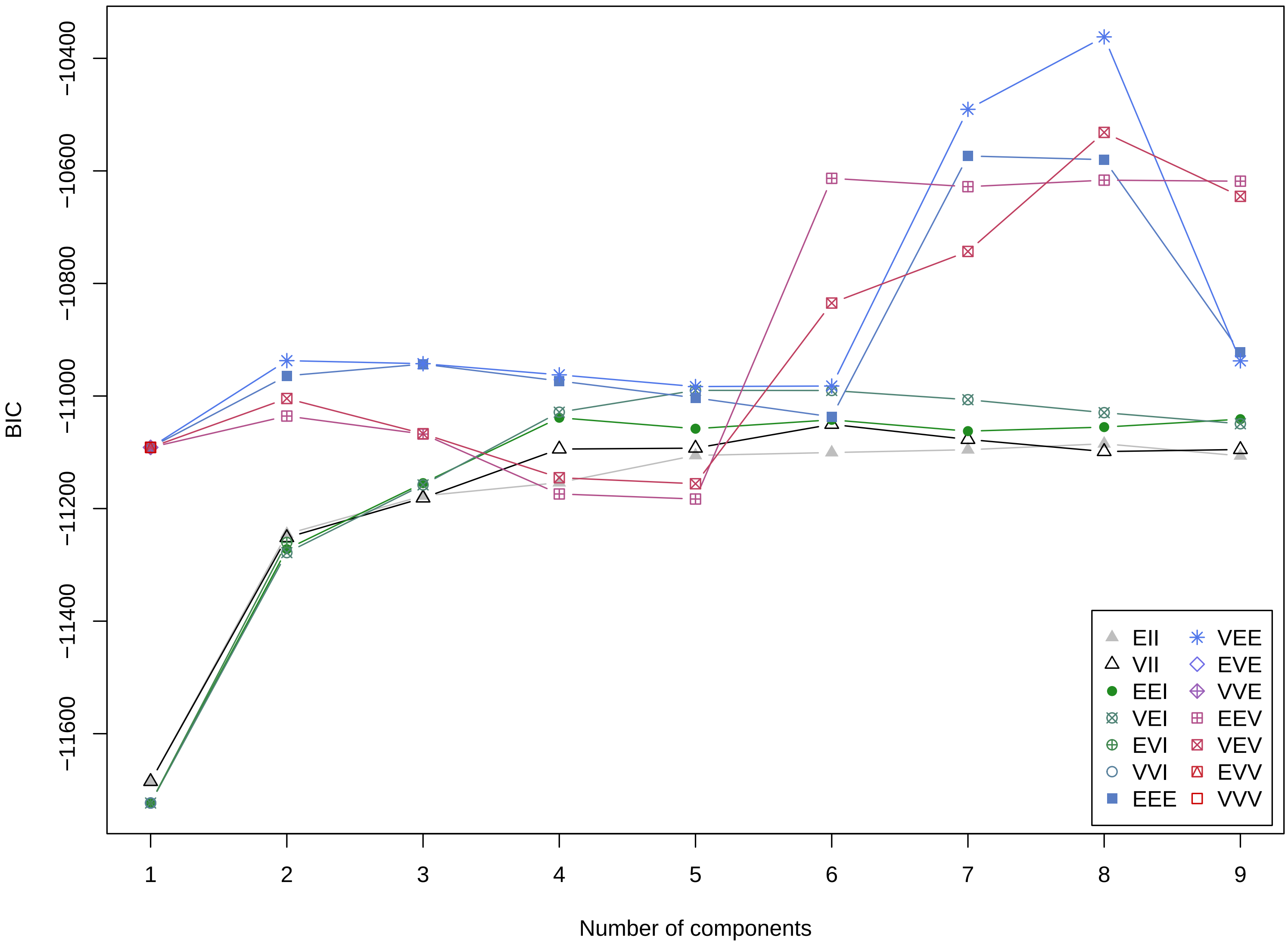

High
293 293
yes
pol_conservatism < 1.8
no
Low
146 226
perceived_difficulty < 0.56
Low
121 135
env_attitude >= 3.9
Low
73 104
pol_conservatism >= 4.8
High
147 67
High
48 31
High
18 12
Low
55 92
Low
25 91

1b

```r
lpa_model <- Mclust(cluster_data_scaled)
summary(lpa_model)
```

```
----------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
----------------------------------------------------

Mclust VEE (ellipsoidal, equal shape and orientation) model with 8 components:

 log-likelihood   n df    BIC    ICL
          -4869 586 98 -10362 -10419

Clustering table:
  1   2   3   4   5   6   7   8
 54 123 115  30  81  68  83  32
```

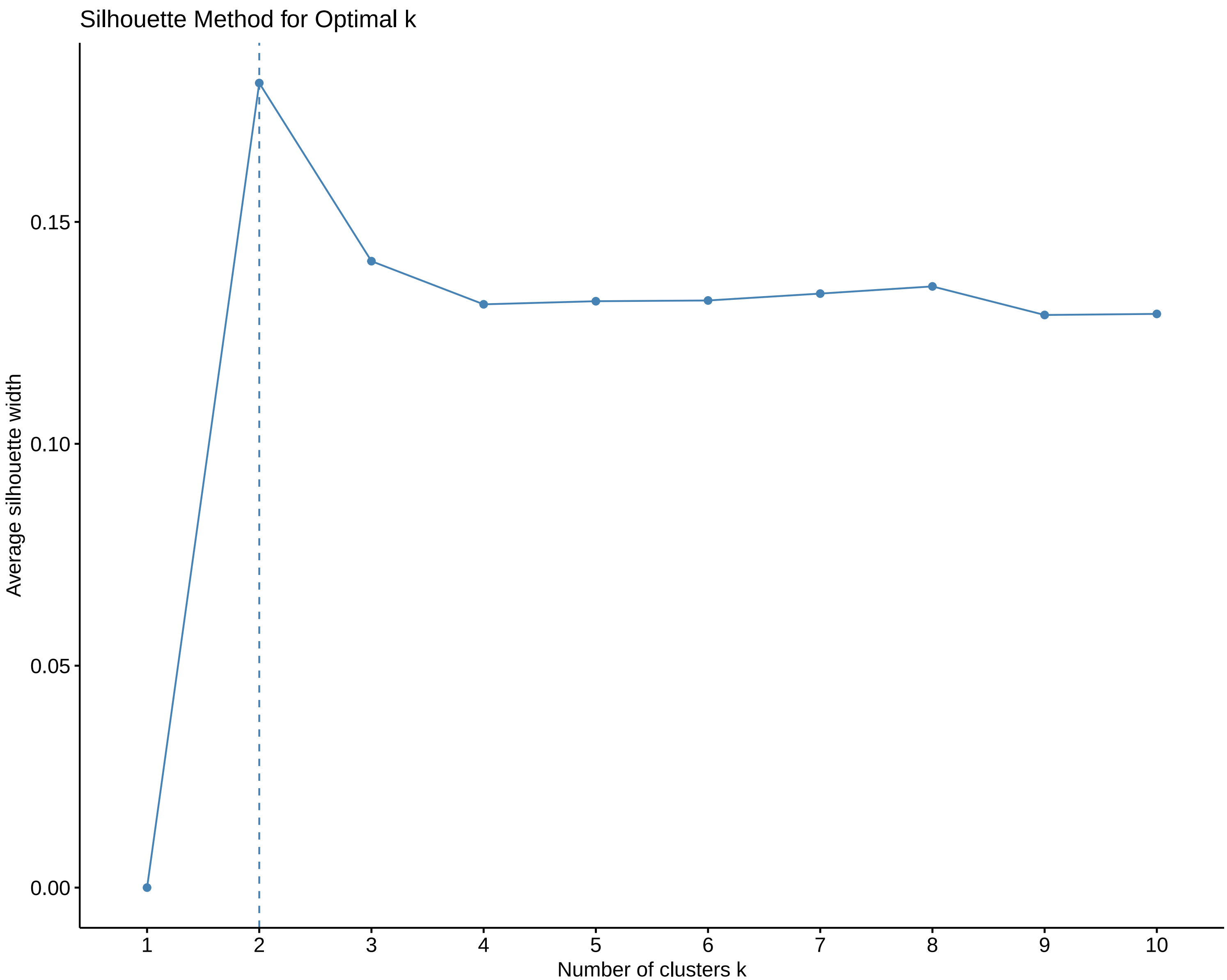
```r
plot(lpa_model, "BIC")
```




```r
can_cor <- cancor(select(combined_scores, numeracy, energy_use, energy_save),
                  select(combined_scores, env_attitude_z, perceived_difficulty))
print(can_cor$cor)
```

```
[1] 0.324 0.084
```

```r
fviz_nbclust(cluster_data_scaled, cluster::pam, method = "silhouette") +
  labs(title = "Silhouette Method for Optimal k")
```




```r
sem_model <- '
  knowledge =~ numeracy + energy_use + energy_save + els_accuracy
  motivation =~ env_attitude_z + perceived_difficulty
  knowledge ~ motivation
'
fit <- sem(sem_model, data = combined_scores)
summary(fit, standardized = TRUE)
```

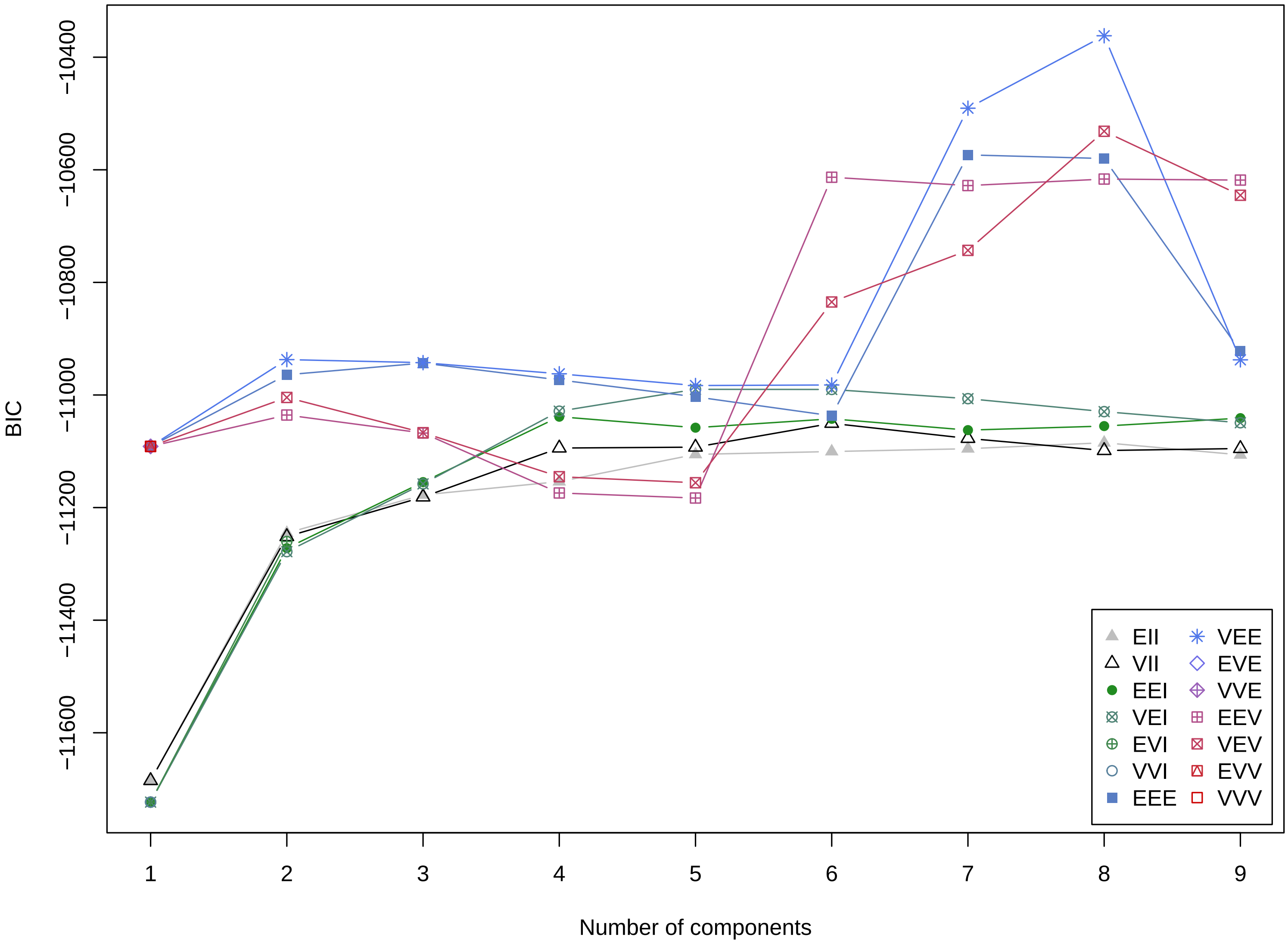
```
lavaan 0.6-19 ended normally after 34 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                        13
```

```
  Number of observations                           586

Model Test User Model:

  Test statistic                                23.012
  Degrees of freedom                                 8
  P-value (Chi-square)                           0.003

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  knowledge =~
    numeracy          1.000                               0.527    0.527
    energy_use        1.469    0.136   10.825    0.000    0.774    0.774
    energy_save       1.352    0.127   10.662    0.000    0.712    0.713
    els_accuracy      1.925    0.208    9.263    0.000    1.014    0.543
  motivation =~
    env_attitude_z    1.000                               0.618    0.619
    percvd_dffclty   -1.006    0.160   -6.283    0.000   -0.622   -0.623

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  knowledge ~
    motivation        0.450    0.076    5.903    0.000    0.529    0.529

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .numeracy          0.721    0.047   15.176    0.000    0.721    0.722
   .energy_use        0.400    0.043    9.282    0.000    0.400    0.400
   .energy_save       0.491    0.043   11.489    0.000    0.491    0.492
   .els_accuracy      2.463    0.164   15.004    0.000    2.463    0.705
   .env_attitude_z    0.616    0.070    8.860    0.000    0.616    0.617
   .percvd_dffclty    0.611    0.070    8.726    0.000    0.611    0.612
   .knowledge         0.200    0.036    5.586    0.000    0.721    0.721
    motivation        0.382    0.075    5.087    0.000    1.000    1.000
```

```r
combined_scores %>%
  group_by(cluster) %>%
  summarise(across(c(numeracy, env_attitude_z),
                   list(mean = mean, sd = sd)))
```

```
# A tibble: 3 x 5
  cluster numeracy_mean numeracy_sd env_attitude_z_mean env_attitude_z_sd
  <fct>           <dbl>       <dbl>               <dbl>             <dbl>
1 1              -0.802       0.983           -0.679                0.890
2 2               0.187       0.774            0.000799             0.869
3 3               0.480       0.761            0.525                0.848
```

```r
plot(lpa_model, "BIC") # Visualize model selection
```




```r
lpa_3class <- Mclust(cluster_data_scaled, G=3) # Force 3-class solution
summary(lpa_3class, parameters=TRUE)
```

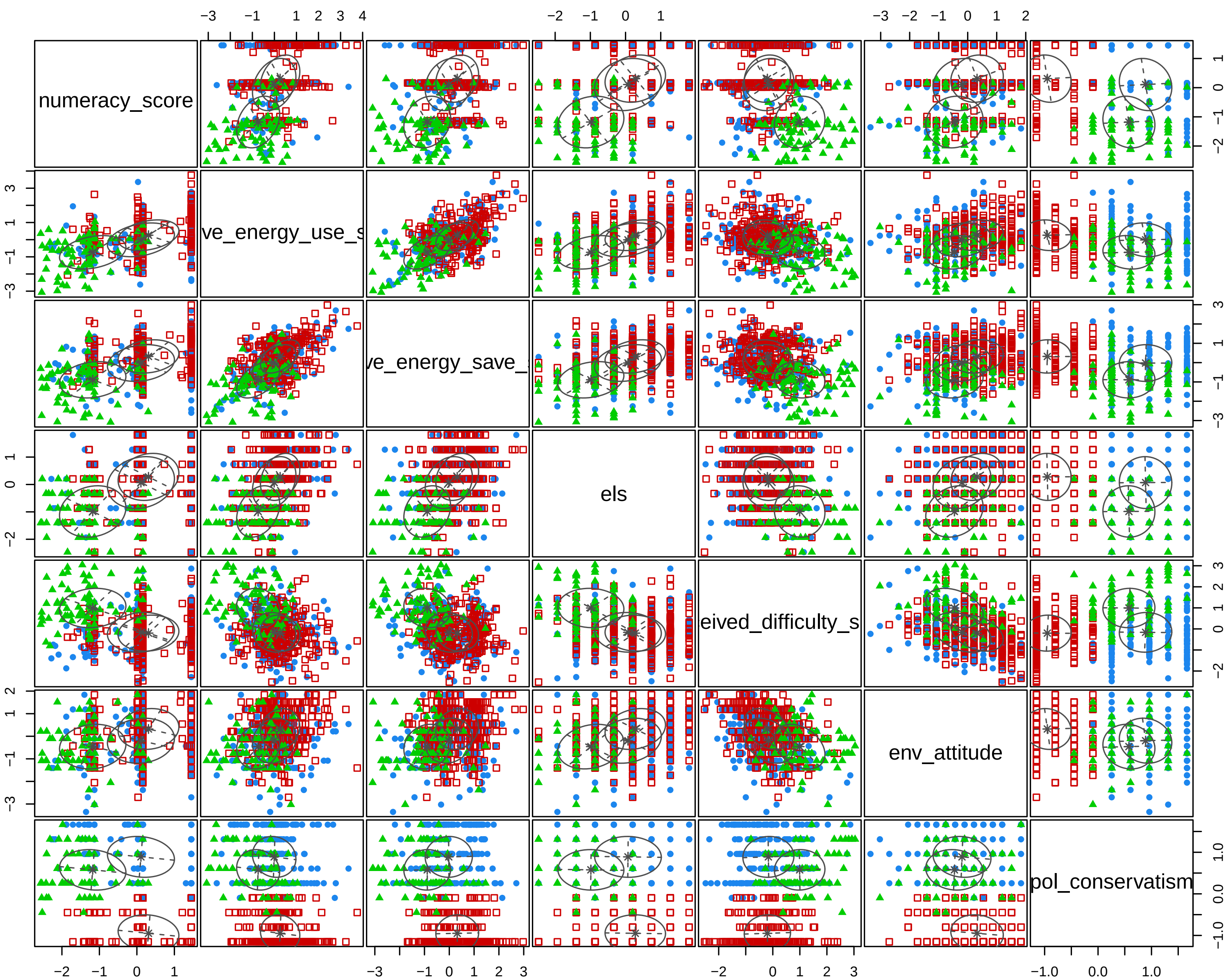
```
---------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
---------------------------------------------------

Mclust VEE (ellipsoidal, equal shape and orientation) model with 3 components:

 log-likelihood   n df    BIC    ICL
          -5302 586 53 -10943 -11055

Clustering table:
  1   2   3
236 270  80

Mixing probabilities:
   1    2    3
0.39 0.46 0.16

Means:
                             [,1]  [,2]  [,3]
numeracy_score              0.108  0.31 -1.18
relative_energy_use_score  -0.010  0.26 -0.74
relative_energy_save_score -0.021  0.32 -0.89
els                         0.069  0.28 -0.98
perceived_difficulty_score -0.163 -0.20  0.99
env_attitude               -0.196  0.32 -0.46
pol_conservatism            0.889 -0.95  0.58

Variances:
[,,1]
                           numeracy_score relative_energy_use_score
numeracy_score                      0.800                      0.27
relative_energy_use_score           0.266                      0.97
relative_energy_save_score          0.151                      0.47
els                                 0.127                      0.27
perceived_difficulty_score          0.087                     -0.11
env_attitude                        0.088                      0.13
pol_conservatism                   -0.051                     -0.05
                           relative_energy_save_score     els
numeracy_score                                 0.1511  0.1272
relative_energy_use_score                      0.4697  0.2750
```

```
relative_energy_save_score                          0.9038  0.2033
els                                                 0.2033  0.8974
perceived_difficulty_score                         -0.0940 -0.0432
env_attitude                                        0.1237  0.1850
pol_conservatism                                    0.0079 -0.0076
                           perceived_difficulty_score env_attitude
numeracy_score                                      0.087        0.088
relative_energy_use_score                          -0.108        0.133
relative_energy_save_score                         -0.094        0.124
els                                                -0.043        0.185
perceived_difficulty_score                          0.890       -0.326
env_attitude                                       -0.326        0.996
pol_conservatism                                    0.014       -0.044
                           pol_conservatism
numeracy_score                      -0.0507
relative_energy_use_score           -0.0504
relative_energy_save_score           0.0079
els                                 -0.0076
perceived_difficulty_score           0.0139
env_attitude                        -0.0441
pol_conservatism                     0.2415
[,,2]
                           numeracy_score relative_energy_use_score
numeracy_score                      0.658                     0.219
relative_energy_use_score           0.219                     0.798
relative_energy_save_score          0.124                     0.386
els                                 0.105                     0.226
perceived_difficulty_score          0.071                    -0.089
env_attitude                        0.073                     0.109
pol_conservatism                   -0.042                    -0.041
                           relative_energy_save_score    els
numeracy_score                                      0.1243  0.1046
relative_energy_use_score                           0.3863  0.2262
relative_energy_save_score                          0.7434  0.1672
els                                                 0.1672  0.7381
perceived_difficulty_score                         -0.0773 -0.0355
env_attitude                                        0.1018  0.1522
pol_conservatism                                    0.0065 -0.0063
                           perceived_difficulty_score env_attitude
numeracy_score                                      0.071        0.073
```

```
relative_energy_use_score                        -0.089       0.109
relative_energy_save_score                       -0.077       0.102
els                                              -0.036       0.152
perceived_difficulty_score                        0.732      -0.268
env_attitude                                     -0.268       0.819
pol_conservatism                                  0.011      -0.036
                           pol_conservatism
numeracy_score                      -0.0417
relative_energy_use_score           -0.0415
relative_energy_save_score           0.0065
els                                 -0.0063
perceived_difficulty_score           0.0114
env_attitude                        -0.0363
pol_conservatism                     0.1987
[,,3]
                           numeracy_score relative_energy_use_score
numeracy_score                      0.785                     0.261
relative_energy_use_score           0.261                     0.952
relative_energy_save_score          0.148                     0.461
els                                 0.125                     0.270
perceived_difficulty_score          0.085                    -0.106
env_attitude                        0.087                     0.130
pol_conservatism                   -0.050                    -0.049
                           relative_energy_save_score     els
numeracy_score                                 0.1483  0.1248
relative_energy_use_score                      0.4608  0.2698
relative_energy_save_score                     0.8868  0.1994
els                                            0.1994  0.8806
perceived_difficulty_score                    -0.0922 -0.0424
env_attitude                                   0.1214  0.1816
pol_conservatism                               0.0078 -0.0075
                           perceived_difficulty_score env_attitude
numeracy_score                                  0.085        0.087
relative_energy_use_score                      -0.106        0.130
relative_energy_save_score                     -0.092        0.121
els                                            -0.042        0.182
perceived_difficulty_score                      0.874       -0.320
env_attitude                                   -0.320        0.977
pol_conservatism                                0.014       -0.043
                           pol_conservatism
```

```
numeracy_score                  -0.0498
relative_energy_use_score       -0.0494
relative_energy_save_score       0.0078
els                             -0.0075
perceived_difficulty_score       0.0136
env_attitude                    -0.0432
pol_conservatism                 0.2370
```

```
plot(lpa_3class, what="classification") # Visualize classification
```




```
# 2. Interpret canonical variables
cancor_loadings <- can_cor$xcoef %>%
  as.data.frame() %>%
  rownames_to_column("variable") %>%
  rename(Dimension1=V1, Dimension2=V2, Dimension3=V3)
```

```r
print(cancor_loadings)
```

```
     variable Dimension1 Dimension2 Dimension3
1    numeracy      0.010      0.045    0.00041
2  energy_use      0.017     -0.019    0.04573
3 energy_save      0.023     -0.012   -0.04353
```

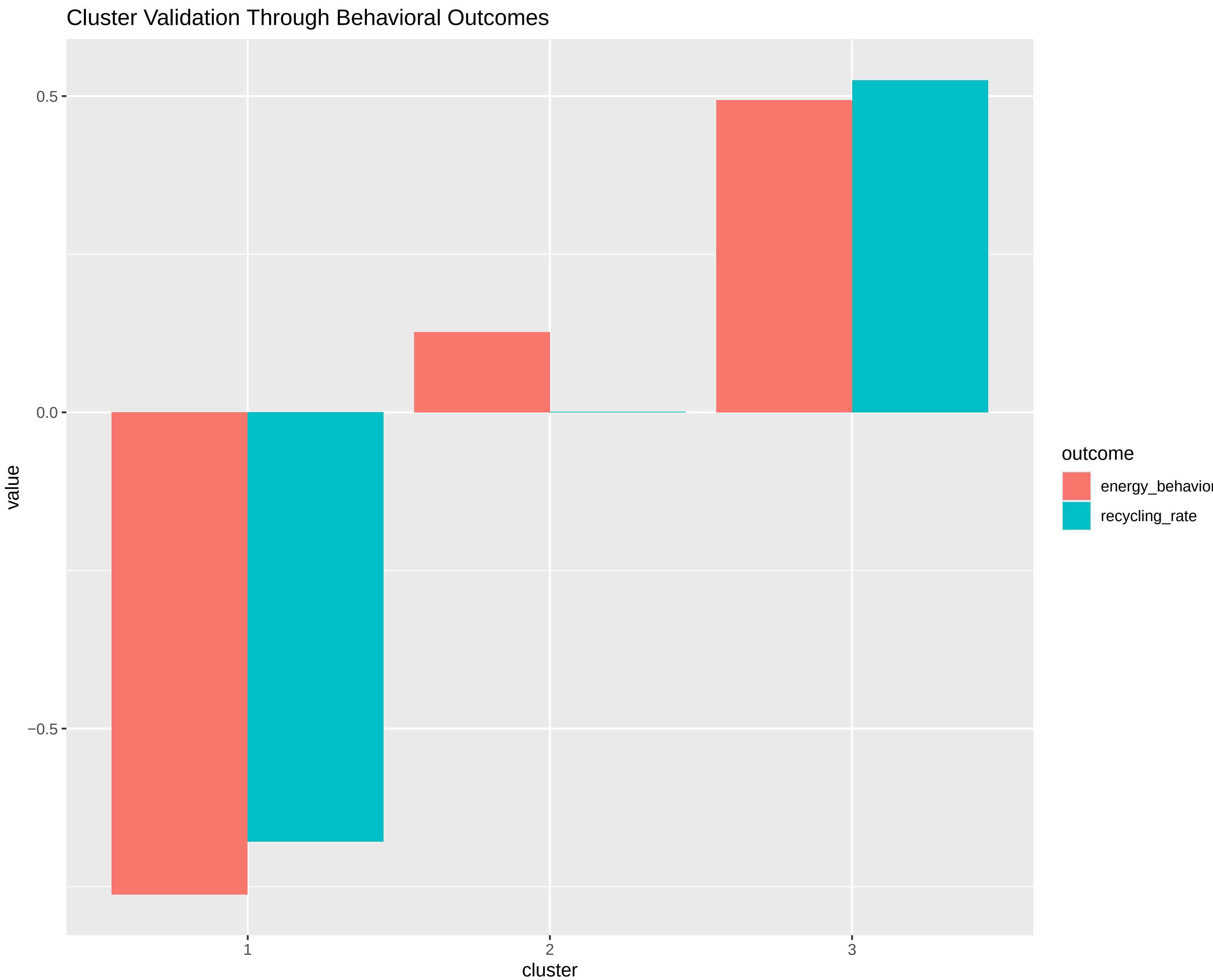
```r
# 3. Improve SEM specification
sem_improved <- '
  knowledge =~ numeracy + energy_use + energy_save + els_accuracy
  motivation =~ env_attitude_z + perceived_difficulty
  knowledge ~ motivation
  els_accuracy ~~ energy_use # Add residual covariance
'
fit_improved <- sem(sem_improved, data=combined_scores)
summary(fit_improved, fit.measures=TRUE)
```

```
lavaan 0.6-19 ended normally after 37 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                        14

  Number of observations                           586

Model Test User Model:

  Test statistic                                19.280
  Degrees of freedom                                 7
  P-value (Chi-square)                           0.007

Model Test Baseline Model:

  Test statistic                               680.231
  Degrees of freedom                                15
  P-value                                        0.000

User Model versus Baseline Model:

  Comparative Fit Index (CFI)                    0.982
```

```
  Tucker-Lewis Index (TLI)                       0.960

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -5022.340
  Loglikelihood unrestricted model (H1)      -5012.700

  Akaike (AIC)                               10072.681
  Bayesian (BIC)                             10133.907
  Sample-size adjusted Bayesian (SABIC)      10089.462

Root Mean Square Error of Approximation:

  RMSEA                                          0.055
  90 Percent confidence interval - lower         0.026
  90 Percent confidence interval - upper         0.085
  P-value H_0: RMSEA <= 0.050                    0.350
  P-value H_0: RMSEA >= 0.080                    0.085

Standardized Root Mean Square Residual:

  SRMR                                           0.029

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  knowledge =~
    numeracy          1.000
    energy_use        1.551    0.154   10.076    0.000
    energy_save       1.312    0.122   10.768    0.000
    els_accuracy      2.123    0.248    8.545    0.000
  motivation =~
    env_attitude_z    1.000
    percvd_dffclty   -0.999    0.158   -6.311    0.000
```

```
Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  knowledge ~
    motivation        0.436    0.075    5.812    0.000

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)
 .energy_use ~~
   .els_accuracy     -0.150    0.077   -1.946    0.052

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .numeracy          0.727    0.048   15.278    0.000
   .energy_use        0.346    0.054    6.423    0.000
   .energy_save       0.531    0.046   11.596    0.000
   .els_accuracy      2.269    0.189   11.994    0.000
   .env_attitude_z    0.613    0.070    8.811    0.000
   .percvd_dffclty    0.614    0.069    8.836    0.000
   .knowledge         0.198    0.035    5.650    0.000
    motivation        0.385    0.075    5.111    0.000
```

```r
# 4. Validate clusters with outcomes
combined_scores %>%
  group_by(cluster) %>%
  summarise(recycling_rate = mean(env_attitude_z, na.rm=TRUE),
            energy_behavior = mean(energy_save, na.rm=TRUE)) %>%
  pivot_longer(-cluster, names_to="outcome") %>%
  ggplot(aes(x=cluster, y=value, fill=outcome)) +
  geom_col(position="dodge") +
  labs(title="Cluster Validation Through Behavioral Outcomes")
```


Cluster Validation Through Behavioral Outcomes
0.5
0.0
−0.5
value
1
2
3
cluster
outcome
energy_behavior
recycling_rate

2b

```r
# Combine all items into a single dataframe
all_items <- full_join(aes_combined, att2_combined, by = "id") %>%
  full_join(els, by = "id") %>%
  full_join(rs, by = "id")

# Select only item columns for factor analysis
item_columns <- setdiff(names(all_items), "id")
item_data <- all_items[, item_columns]

# Perform factor analysis
fa_items <- fa(item_data, nfactors = 5, rotate = "varimax")  # Adjust nfactors as needed
print(fa_items, cut = 0.3, sort = TRUE)
```

```
Factor Analysis using method =  minres
Call: fa(r = item_data, nfactors = 5, rotate = "varimax")
Standardized loadings (pattern matrix) based upon correlation matrix
      item   MR1   MR2   MR5   MR3   MR4    h2    u2 com
ATT25   25  0.94                          0.903 0.097 1.0
ATT23   23  0.90                          0.865 0.135 1.1
ATT27   27  0.89                          0.804 0.196 1.0
ATT26   26  0.89                          0.810 0.190 1.0
ATT24   24  0.82                          0.767 0.233 1.3
ATT33   33  0.70                          0.622 0.378 1.5
ATT32   32  0.61                          0.424 0.576 1.3
ATT30   30  0.56        0.40              0.572 0.428 2.6
ATT31   31  0.42                          0.258 0.742 1.9
ELS08   41                                0.026 0.974 3.2
ATT10   10        0.63                    0.455 0.545 1.3
ATT15   15        0.63       -0.37        0.537 0.463 1.7
ATT09    9        0.62                    0.456 0.544 1.3
ATT14   14        0.62       -0.34        0.516 0.484 1.6
ATT06    6        0.61                    0.401 0.599 1.2
ATT07    7        0.56                    0.337 0.663 1.1
ATT08    8        0.55                    0.313 0.687 1.0
ATT13   13        0.54                    0.317 0.683 1.2
ATT03    3        0.49        0.36        0.367 0.633 1.9
ATT12   12        0.48                    0.256 0.744 1.2
ATT05    5        0.48        0.37        0.362 0.638 1.9
```

ATT04	4		0.47				0.235	0.765	1.1
ATT01	1		0.42		0.31		0.275	0.725	1.9
RS01	42		-0.40				0.248	0.752	2.0
RS02	43						0.083	0.917	1.2
ATT11	11						0.070	0.930	1.5
ELS01	34						0.037	0.963	2.8
ATT20	20			0.92			0.915	0.085	1.2
ATT21	21	0.35		0.79			0.759	0.241	1.4
ATT22	22			0.73			0.610	0.390	1.3
RS03	44		-0.37		0.59		0.503	0.497	1.8
RS04	45				0.46		0.256	0.744	1.5
RS05	46				0.44		0.212	0.788	1.2
RS06	47				0.38		0.158	0.842	1.2
ATT17	17				-0.36		0.166	0.834	1.5
ELS02	35				0.34		0.132	0.868	1.2
ATT18	18						0.139	0.861	2.6
ELS03	36						0.073	0.927	1.4
ATT02	2						0.111	0.889	2.2
ELS04	37						0.038	0.962	1.3
ELS07	40						0.039	0.961	1.9
ATT19	19						0.028	0.972	1.1
ELS05	38						0.022	0.978	1.4
ATT28	28					0.94	0.888	0.112	1.0
ATT29	29					0.90	0.827	0.173	1.0
ATT16	16						0.024	0.976	1.1
ELS06	39						0.027	0.973	4.0

	MR1	MR2	MR5	MR3	MR4
SS loadings	5.69	4.67	2.52	2.38	1.98
Proportion Var	0.12	0.10	0.05	0.05	0.04
Cumulative Var	0.12	0.22	0.27	0.32	0.37
Proportion Explained	0.33	0.27	0.15	0.14	0.11
Cumulative Proportion	0.33	0.60	0.75	0.89	1.00

Mean item complexity = 1.6

Test of the hypothesis that 5 factors are sufficient.

df null model = 1081 with the objective function = 27 with Chi Square = 15130

df of the model are 856 and the objective function was 8.8

```
The root mean square of the residuals (RMSR) is  0.05
The df corrected root mean square of the residuals is  0.06

The harmonic n.obs is  586 with the empirical chi square  3813  with prob <  0
The total n.obs was  586  with Likelihood Chi Square =  4950  with prob <  0

Tucker Lewis Index of factoring reliability =  0.63
RMSEA index =  0.09  and the 90 % confidence intervals are  0.088 0.093
BIC =  -506
Fit based upon off diagonal values = 0.91
Measures of factor score adequacy
                                                   MR1  MR2  MR5  MR3  MR4
Correlation of (regression) scores with factors   0.98 0.93 0.97 0.87 0.97
Multiple R square of scores with factors          0.97 0.87 0.94 0.76 0.93
Minimum correlation of possible factor scores     0.94 0.75 0.88 0.52 0.87
```

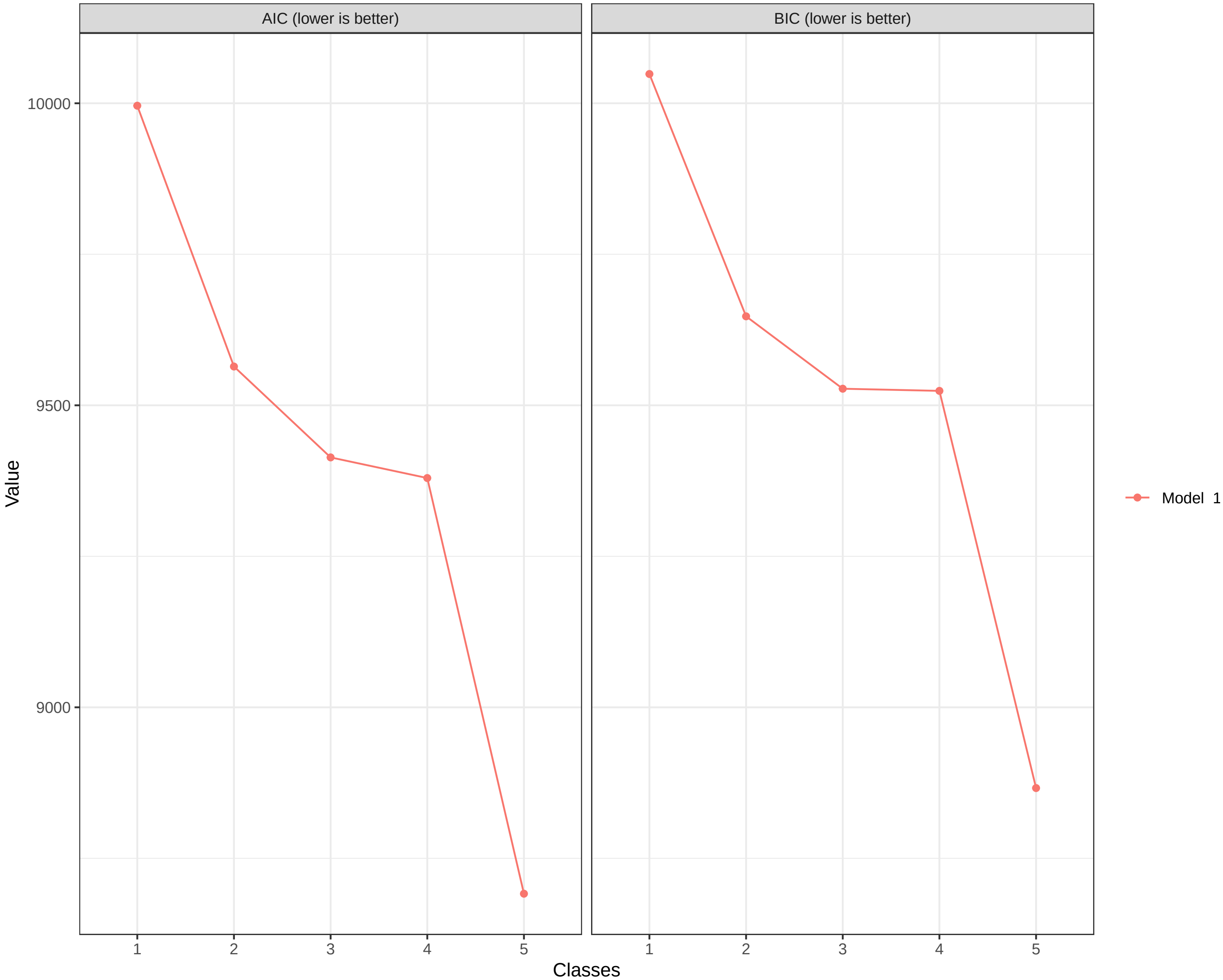
```r
model <- '
  # Measurement model
  Knowledge =~ numeracy + energy_use + energy_save + els_score
  Motivation =~ env_attitude_z + perceived_difficulty

  # Structural model
  Knowledge ~ Motivation
'

fit <- sem(model, data = combined_scores)
summary(fit, fit.measures = TRUE, standardized = TRUE)
```

```
lavaan 0.6-19 ended normally after 32 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                        13

  Number of observations                           586

Model Test User Model:

  Test statistic                                23.012
  Degrees of freedom                                 8
```

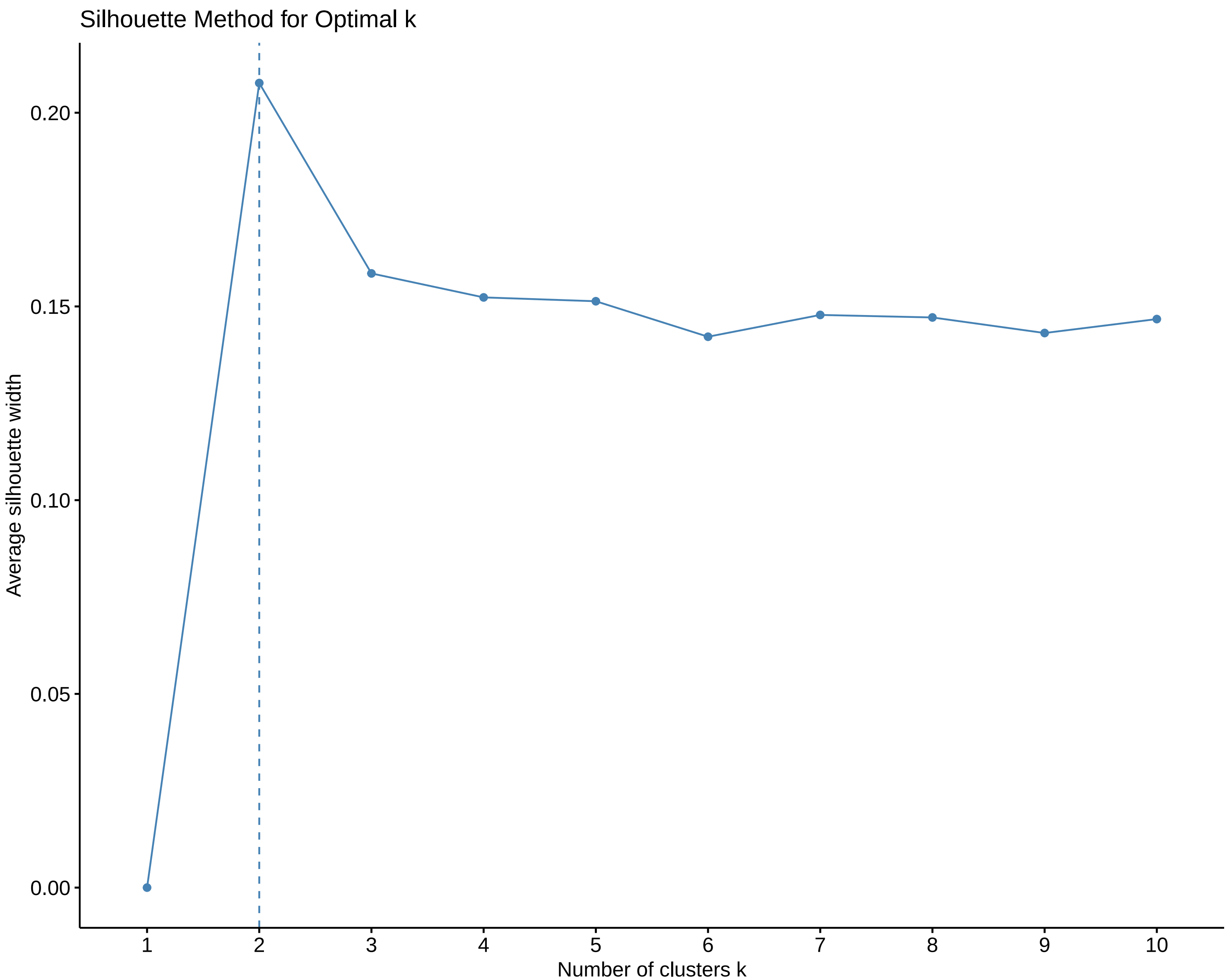
```
  P-value (Chi-square)                           0.003

Model Test Baseline Model:

  Test statistic                               680.231
  Degrees of freedom                                15
  P-value                                        0.000

User Model versus Baseline Model:

  Comparative Fit Index (CFI)                    0.977
  Tucker-Lewis Index (TLI)                       0.958

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -4657.376
  Loglikelihood unrestricted model (H1)      -4645.870

  Akaike (AIC)                                9340.753
  Bayesian (BIC)                              9397.606
  Sample-size adjusted Bayesian (SABIC)       9356.335

Root Mean Square Error of Approximation:

  RMSEA                                          0.057
  90 Percent confidence interval - lower         0.030
  90 Percent confidence interval - upper         0.084
  P-value H_0: RMSEA <= 0.050                    0.305
  P-value H_0: RMSEA >= 0.080                    0.086

Standardized Root Mean Square Residual:

  SRMR                                           0.032

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured
```

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  Knowledge =~
    numeracy          1.000                               0.527    0.527
    energy_use        1.469    0.136   10.825    0.000    0.774    0.774
    energy_save       1.352    0.127   10.662    0.000    0.712    0.713
    els_score         1.029    0.111    9.263    0.000    0.542    0.543
  Motivation =~
    env_attitude_z    1.000                               0.618    0.619
    percvd_dffclty   -1.006    0.160   -6.283    0.000   -0.622   -0.623

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  Knowledge ~
    Motivation        0.450    0.076    5.903    0.000    0.529    0.529

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .numeracy          0.721    0.047   15.176    0.000    0.721    0.722
   .energy_use        0.400    0.043    9.282    0.000    0.400    0.400
   .energy_save       0.491    0.043   11.489    0.000    0.491    0.492
   .els_score         0.704    0.047   15.004    0.000    0.704    0.705
   .env_attitude_z    0.616    0.070    8.860    0.000    0.616    0.617
   .percvd_dffclty    0.611    0.070    8.726    0.000    0.611    0.612
   .Knowledge         0.200    0.036    5.586    0.000    0.721    0.721
    Motivation        0.382    0.075    5.087    0.000    1.000    1.000
```

```r
# Example LPA (using tidyLPA)
lpa_data <- combined_scores %>%
  select(numeracy, energy_use, energy_save, els_score, env_attitude_z, perceived_difficulty) %>%
  na.omit() |>
  # convert all to numeric
    mutate_all(as.numeric)

lpa_results <- lpa_data %>%
  estimate_profiles(n_profiles = 1:5) %>% # Estimate models with 1-5 profiles
  compare_solutions(statistics = c("AIC", "BIC"))

plot(lpa_results)
```



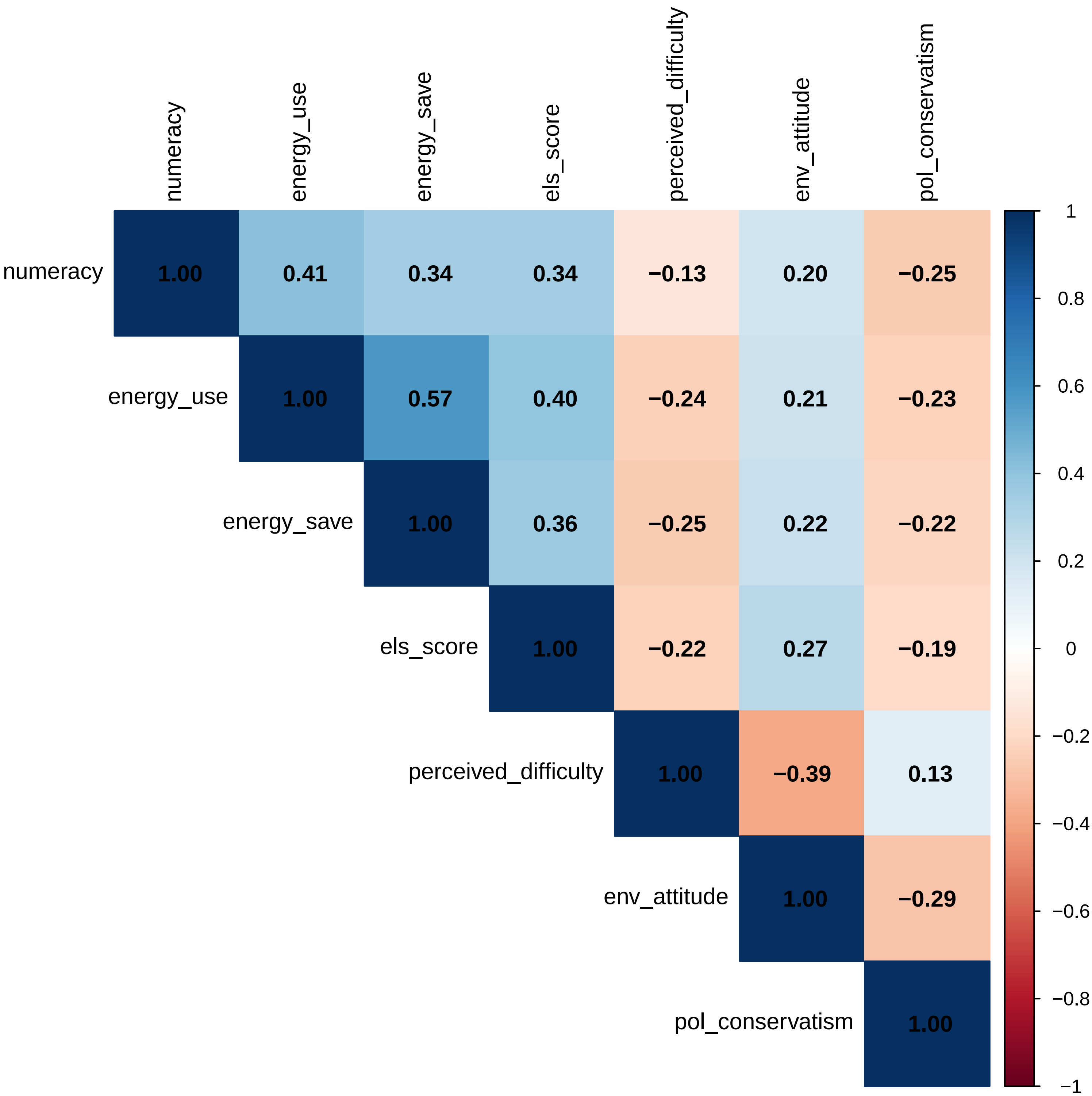

```r
# Determine optimal k using silhouette method
fviz_nbclust(cluster_data_scaled, kmeans, method = "silhouette") +
  labs(title = "Silhouette Method for Optimal k")
```



3b

```
# 1) Create a correlation matrix of the key knowledge & motivation subscales
#    ensuring no duplicates (e.g., pick either 'env_attitude' or 'env_attitude_z').

cor_vars <- combined_scores %>%
  select(numeracy, energy_use, energy_save, els_score,
         perceived_difficulty, env_attitude, pol_conservatism)

# 2) Compute correlations
cor_matrix <- cor(cor_vars, use = "pairwise.complete.obs")
```

```r
# 3) Visualize
corrplot::corrplot(cor_matrix, method = "color", type="upper",
                   tl.col="black", addCoef.col="black")
```




```r
fa_data <- combined_scores %>%
  select(numeracy, energy_use, energy_save, els_score,
         perceived_difficulty, env_attitude, pol_conservatism) %>%
```

```r
  na.omit()

fa_result <- fa(fa_data, nfactors = 2, rotate = "varimax", fm = "ml")
print(fa_result, cut=0.3, sort=TRUE)
```

```
Factor Analysis using method =  ml
Call: fa(r = fa_data, nfactors = 2, rotate = "varimax", fm = "ml")
Standardized loadings (pattern matrix) based upon correlation matrix
                     item   ML2   ML1   h2    u2 com
energy_use              2  0.78       0.63 0.374 1.1
energy_save             3  0.69       0.50 0.497 1.1
numeracy                1  0.51       0.28 0.720 1.2
els_score               4  0.49       0.29 0.709 1.4
pol_conservatism        7             0.14 0.860 2.0
env_attitude            6        0.99 1.00 0.005 1.0
perceived_difficulty    5       -0.37 0.19 0.807 1.7

                       ML2  ML1
SS loadings           1.71 1.31
Proportion Var        0.24 0.19
Cumulative Var        0.24 0.43
Proportion Explained  0.57 0.43
Cumulative Proportion 0.57 1.00

Mean item complexity =  1.4
Test of the hypothesis that 2 factors are sufficient.

df null model =  21  with the objective function =  1.3 with Chi Square =  760
df of  the model are 8  and the objective function was  0.03

The root mean square of the residuals (RMSR) is  0.03
The df corrected root mean square of the residuals is  0.04

The harmonic n.obs is  586 with the empirical chi square  18  with prob <  0.025
The total n.obs was  586  with Likelihood Chi Square =  16  with prob <  0.036

Tucker Lewis Index of factoring reliability =  0.97
RMSEA index =  0.042  and the 90 % confidence intervals are  0.01 0.072
BIC =  -35
Fit based upon off diagonal values = 0.99
```

```
Measures of factor score adequacy
                                                   ML2  ML1
Correlation of (regression) scores with factors   0.87 1.00
Multiple R square of scores with factors          0.77 0.99
Minimum correlation of possible factor scores     0.53 0.99
```

```r
model_motivation <- lm(env_attitude ~ els_score + numeracy + pol_conservatism,
                       data=combined_scores)
summary(model_motivation)
```

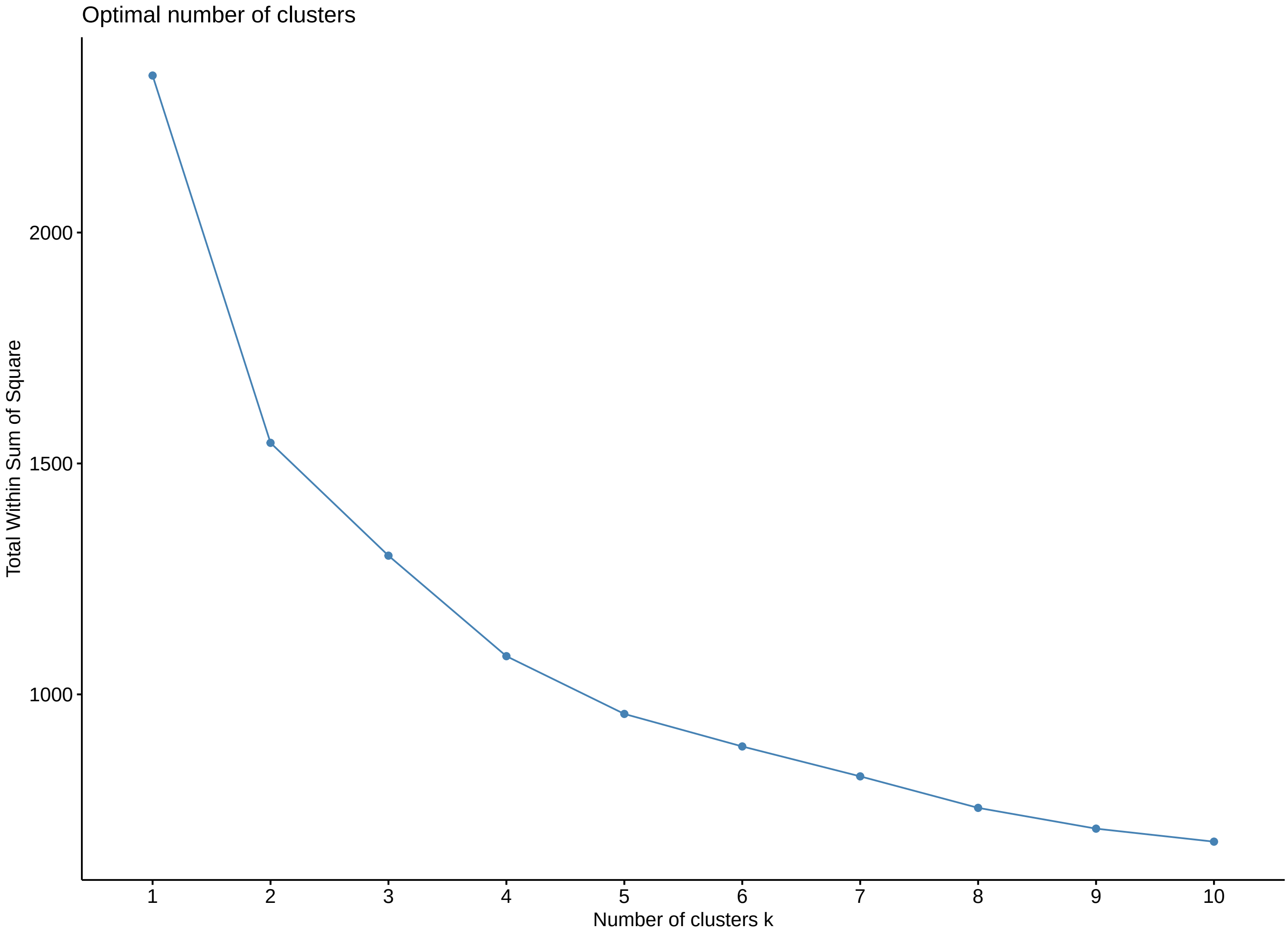
```

Call:
lm(formula = env_attitude ~ els_score + numeracy + pol_conservatism,
    data = combined_scores)

Residuals:
   Min     1Q Median     3Q    Max
-2.320 -0.486  0.020  0.514  1.944

Coefficients:
                 Estimate Std. Error t value             Pr(>|t|)
(Intercept)        3.9098     0.0646   60.48 < 0.0000000000000002 ***
els_score          0.1575     0.0318    4.95           0.00000098 ***
numeracy           0.0555     0.0323    1.72                0.086 .
pol_conservatism  -0.1239     0.0218   -5.69           0.00000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.72 on 582 degrees of freedom
Multiple R-squared:  0.135, Adjusted R-squared:  0.131
F-statistic: 30.3 on 3 and 582 DF,  p-value: <0.0000000000000002
```

```r
model_knowledge <- lm(els_score ~ perceived_difficulty + env_attitude + pol_conservatism,
                      data=combined_scores)
summary(model_knowledge)
```

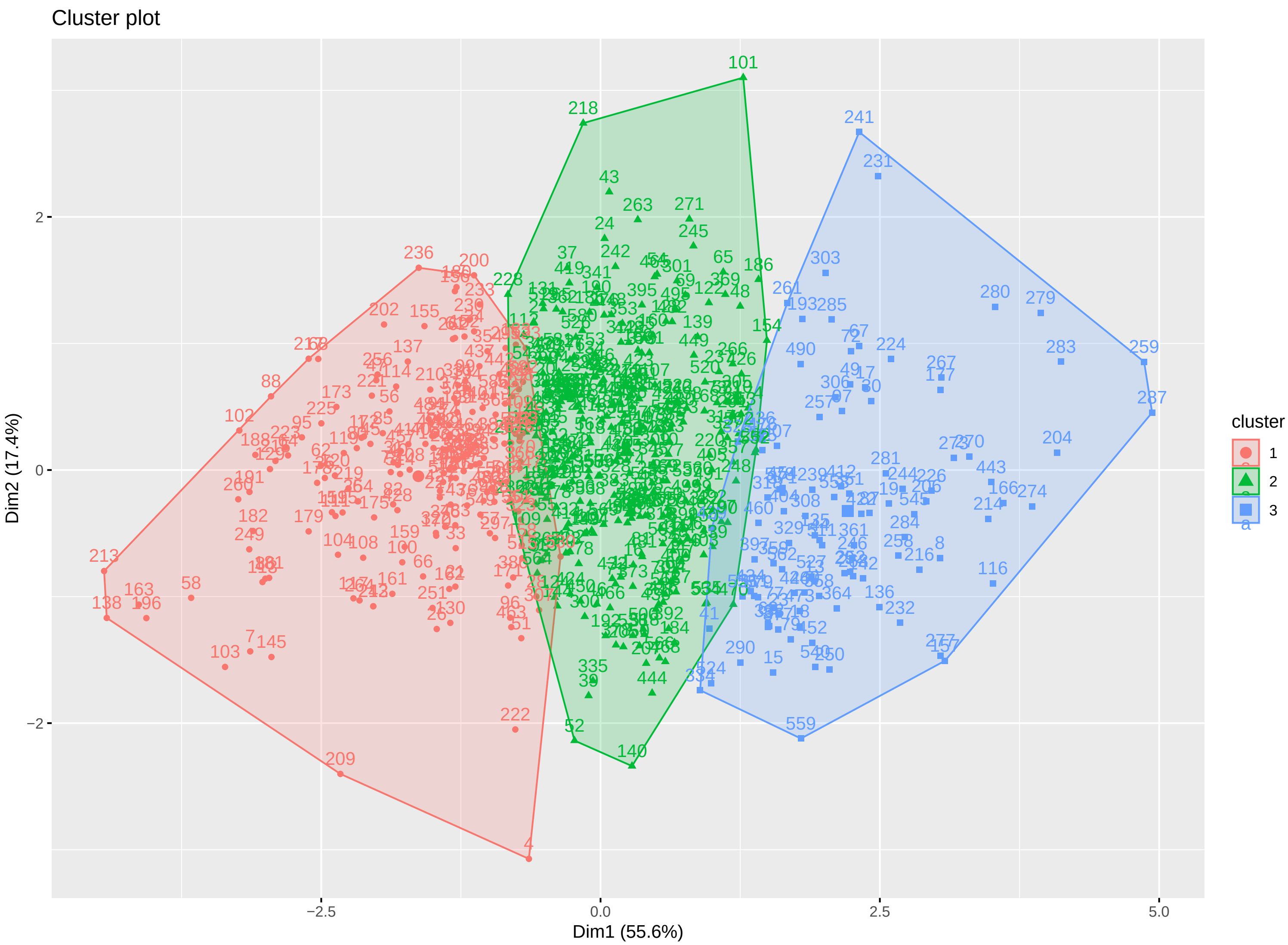
```

Call:
lm(formula = els_score ~ perceived_difficulty + env_attitude +
    pol_conservatism, data = combined_scores)
```

```
Residuals:
   Min     1Q Median     3Q    Max
-3.171 -0.646 -0.004  0.716  2.333

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)           -0.6422     0.2390   -2.69   0.0074 **
perceived_difficulty  -0.1338     0.0425   -3.15   0.0017 **
env_attitude           0.2421     0.0571    4.24 0.000026 ***
pol_conservatism      -0.0854     0.0289   -2.96   0.0032 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.95 on 582 degrees of freedom
Multiple R-squared:  0.104, Adjusted R-squared:  0.0992
F-statistic: 22.5 on 3 and 582 DF,  p-value: 0.0000000000000872
```

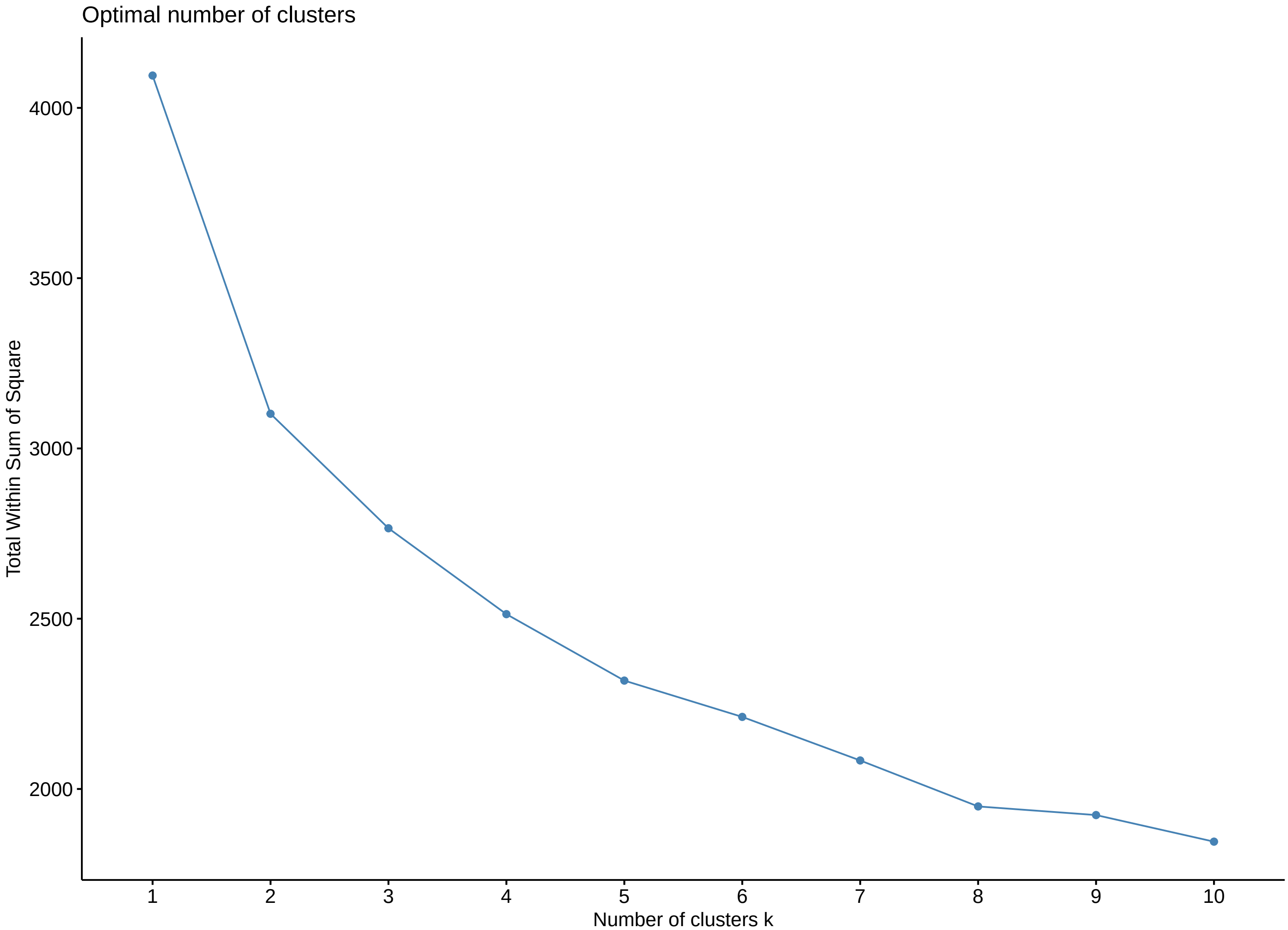
```r
knowledge_only <- combined_scores %>%
  select(numeracy, energy_use, energy_save, els_score) %>%
  na.omit() %>%
  scale()

set.seed(123)
# Decide k with elbow or silhouette
fviz_nbclust(knowledge_only, kmeans, method="wss")
```



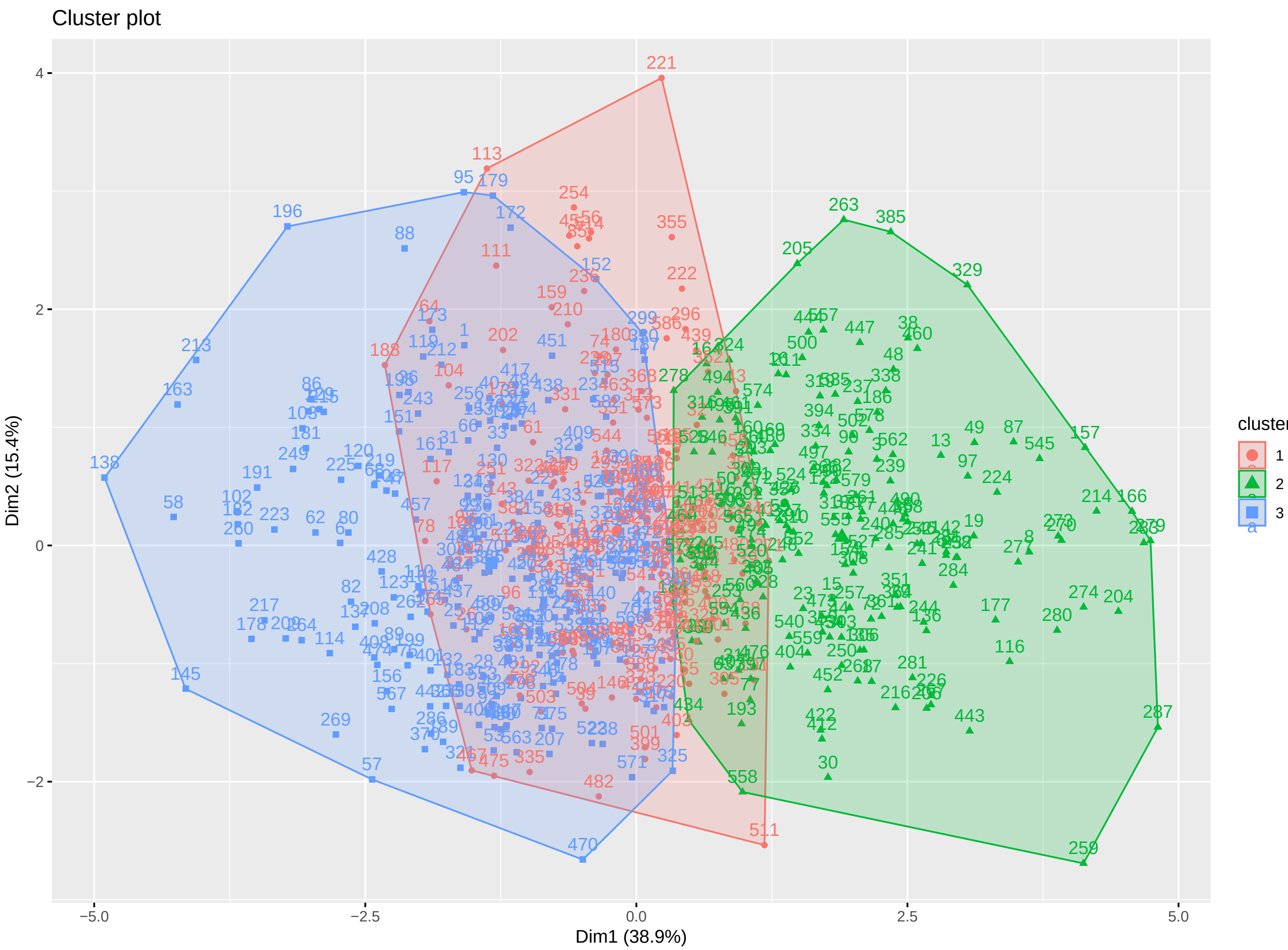

```r
km_knowl <- kmeans(knowledge_only, centers=3, nstart=25)
fviz_cluster(km_knowl, data = knowledge_only)
```




```r
all_vars <- combined_scores %>%
  select(numeracy, energy_use, energy_save, els_score,
         perceived_difficulty, env_attitude, pol_conservatism) %>%
  na.omit() %>%
  scale()


set.seed(123)
fviz_nbclust(all_vars, kmeans, method="wss")
```




```
km_all <- kmeans(all_vars, centers=3, nstart=25)
fviz_cluster(km_all, data=all_vars)
```



Cluster plot
Dim1 (38.9%)
Dim2 (15.4%)
cluster
1
2
3

4b

```r
# 1. Summarize clusters on an extra measure
combined_scores %>%
  group_by(cluster) %>%
  summarise(
    mean_recycling = mean(env_attitude_z, na.rm=TRUE),
    sd_recycling   = sd(env_attitude_z, na.rm=TRUE),
    n = n()
  ) %>%
  arrange(cluster)
```

```
# A tibble: 3 x 4
  cluster mean_recycling sd_recycling     n
  <fct>            <dbl>        <dbl> <int>
1 1            -0.679           0.890   179
2 2             0.000799        0.869   176
3 3             0.525           0.848   231
```

```r
# 2. ANOVA to test whether clusters differ significantly
anova_result <- aov(env_attitude_z ~ cluster, data = combined_scores)
summary(anova_result)
```

```
             Df Sum Sq Mean Sq F value              Pr(>F)
cluster       2    146    73.1    97.1 <0.0000000000000002 ***
Residuals   583    439     0.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# 3. Pairwise comparisons if ANOVA is significant
TukeyHSD(anova_result)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = env_attitude_z ~ cluster, data = combined_scores)

$cluster
    diff  lwr  upr p adj
2-1 0.68 0.46 0.90     0
```

```
3-1 1.20 1.00 1.41     0
3-2 0.52 0.32 0.73     0
```

```r
# Subset data to knowledge & motivation variables
lpa_data <- combined_scores %>%
  select(numeracy, energy_use, energy_save, els_score,
         perceived_difficulty, env_attitude, pol_conservatism) %>%
  na.omit() %>%
  scale()

# Model-based clustering
lpa_model <- Mclust(lpa_data)
summary(lpa_model)  # Tells you how many clusters & the type of covariance structure
```

```
----------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
----------------------------------------------------

Mclust VEE (ellipsoidal, equal shape and orientation) model with 8 components:

 log-likelihood   n df    BIC    ICL
          -4869 586 98 -10362 -10419

Clustering table:
  1   2   3   4   5   6   7   8
 54 123 115  30  81  68  83  32
```

```r
# Extract membership
combined_scores$LPA_cluster <- as.factor(lpa_model$classification)
table(combined_scores$LPA_cluster)
```

```

  1   2   3   4   5   6   7   8
 54 123 115  30  81  68  83  32
```

```r
# Compare means across the new LPA-based clusters
combined_scores %>%
  group_by(LPA_cluster) %>%
  summarise(
    across(numeracy:pol_conservatism, mean, na.rm=TRUE)
  )
```

```
# A tibble: 8 x 9
  LPA_cluster numeracy energy_use energy_save els_accuracy els_score
  <fct>          <dbl>      <dbl>       <dbl>        <dbl>     <dbl>
1 1             1.45       0.453      0.258           5.04    0.222
2 2             0.106      0.0661     0.146           4.84    0.116
3 3            -1.50      -0.687     -0.671           3.17   -0.774
4 4            -1.17       0.146      0.239           4.37   -0.136
5 5             0.117     -0.130     -0.193           4.79    0.0903
6 6             1.43       0.785      0.659           5.18    0.297
7 7             0.0666    -0.0178     0.00612         4.67    0.0286
8 8             0.133      0.0197     0.261           6.78    1.16
# i 3 more variables: env_attitude <dbl>, env_attitude_z <dbl>,
#   pol_conservatism <dbl>
```

1c

```
# Combine all individual survey items from each instrument
all_items <- bind_rows(aes_combined, att2_combined) %>%
  full_join(els, by = "id") %>%
  full_join(rs, by = "id")

# Select only item columns for analysis
item_data <- all_items %>% select(-id)

subscale_cors <- combined_scores %>%
  select(perceived_difficulty, numeracy, energy_use, energy_save,
         els_score, env_attitude_z, pol_conservatism_z) %>%
  cor(use = "pairwise.complete.obs")



# 3. Knowledge-Motivation Relationship Analyses
# Create composite scores with explicit content alignment
knowledge_vars <- c("numeracy", "energy_use", "energy_save", "els_score")
motivation_vars <- c("env_attitude_z", "perceived_difficulty")

combined_scores <- combined_scores %>%
  mutate(
    knowledge = scale(rowMeans(select(., all_of(knowledge_vars)), na.rm = TRUE)),
    motivation = scale(rowMeans(select(., all_of(motivation_vars)) * c(1, -1), na.rm = TRUE))
  )

# 3a. Bivariate Correlation
with(combined_scores, cor.test(knowledge, motivation))
```

```
    Pearson's product-moment correlation

data:  knowledge and motivation
t = -0.4, df = 584, p-value = 0.7
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.099  0.063
sample estimates:
   cor
```

```
-0.018
```

```r
# 3b. Hierarchical Regression
model <- lm(knowledge ~ motivation + pol_conservatism_z + cluster,
            data = combined_scores)
summary(model)
```

```

Call:
lm(formula = knowledge ~ motivation + pol_conservatism_z + cluster,
    data = combined_scores)

Residuals:
    Min      1Q  Median      3Q     Max
-2.2080 -0.4889 -0.0304  0.4339  2.2729

Coefficients:
                   Estimate Std. Error t value            Pr(>|t|)
(Intercept)         -1.0897     0.0546  -19.96 <0.0000000000000002 ***
motivation          -0.0437     0.0284   -1.54                0.12
pol_conservatism_z   0.0339     0.0464    0.73                0.46
cluster2             1.3164     0.0753   17.47 <0.0000000000000002 ***
cluster3             1.7613     0.0933   18.88 <0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.68 on 581 degrees of freedom
Multiple R-squared:  0.535, Adjusted R-squared:  0.532
F-statistic:  167 on 4 and 581 DF,  p-value: <0.0000000000000002
```

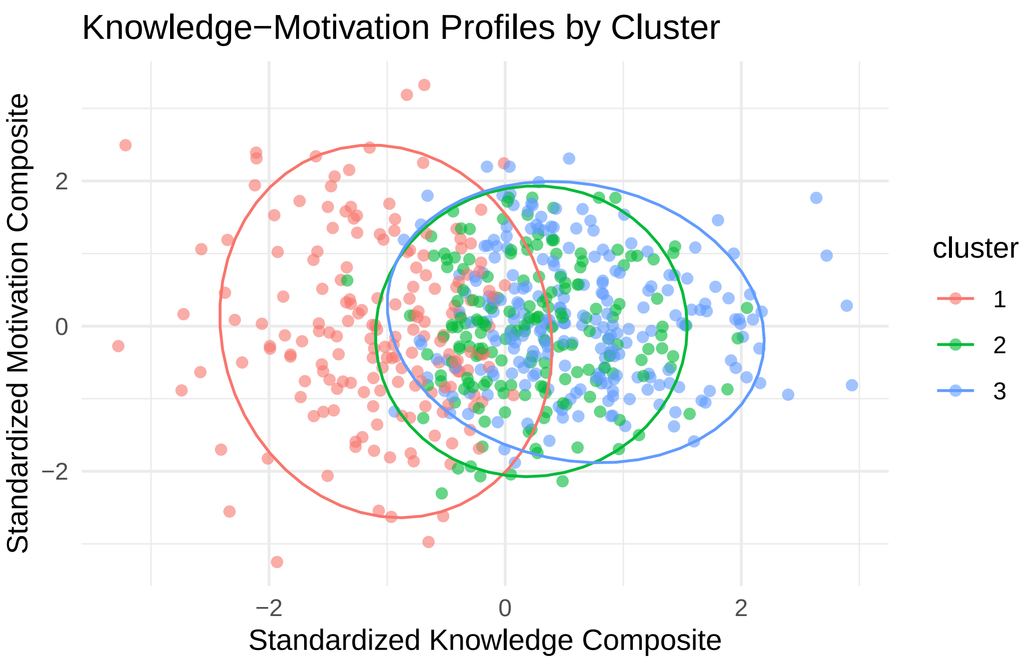
```r
# 3c. Path Analysis
path_model <- '
  motivation ~ a * knowledge
  els_score ~ b * motivation + c * knowledge
  indirect := a * b
  total := c + indirect
'
fit <- sem(path_model, data = combined_scores)
summary(fit, standardized = TRUE)
```

```
lavaan 0.6-19 ended normally after 5 iterations
```

```
  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                         5

  Number of observations                           586

Model Test User Model:

  Test statistic                                 0.000
  Degrees of freedom                                 0

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  motivation ~
    knowledge  (a)   -0.018    0.041   -0.439    0.660   -0.018   -0.018
  els_score ~
    motivation (b)    0.036    0.029    1.229    0.219    0.036    0.036
    knowledge  (c)    0.707    0.029   24.181    0.000    0.707    0.707

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .motivation        0.998    0.058   17.117    0.000    0.998    1.000
   .els_score         0.499    0.029   17.117    0.000    0.499    0.500

Defined Parameters:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    indirect         -0.001    0.002   -0.414    0.679   -0.001   -0.001
    total             0.706    0.029   24.132    0.000    0.706    0.706
```

```r
# 4. Cluster Validation by Motivation-Knowledge Profiles
ggplot(combined_scores, aes(x = knowledge, y = motivation, color = cluster)) +
  geom_point(alpha = 0.6) +
  stat_ellipse(level = 0.95) +
```

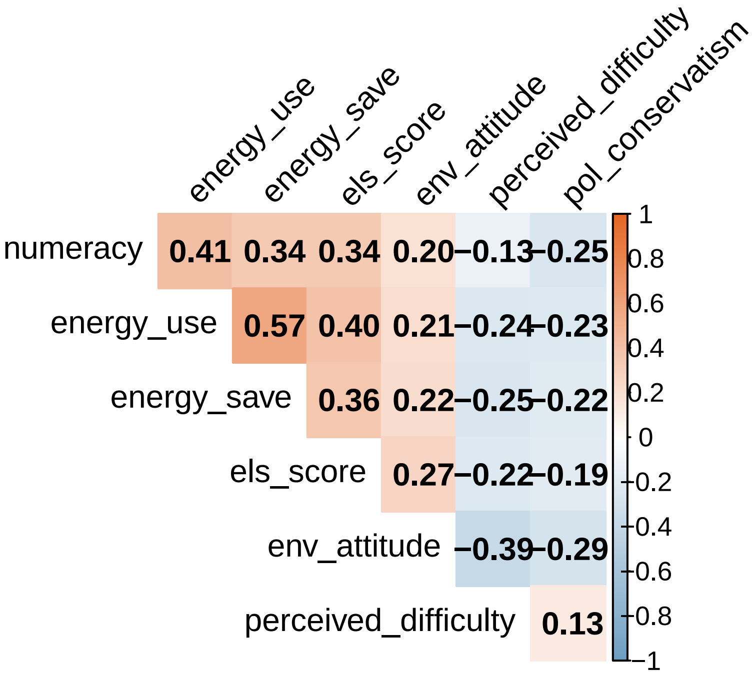
```
  labs(title = "Knowledge-Motivation Profiles by Cluster",
       x = "Standardized Knowledge Composite",
       y = "Standardized Motivation Composite") +
  theme_minimal()
```



2c

```
# Combine key measures into correlation matrix
key_measures <- combined_scores %>%
  select(
    # Knowledge measures
    numeracy, energy_use, energy_save, els_score,
    # Motivation/attitude measures
    env_attitude, perceived_difficulty, pol_conservatism
  ) %>%
  na.omit()

# Compute and visualize correlation matrix
cor_matrix <- cor(key_measures, use="pairwise.complete.obs")
corrplot(cor_matrix,
         method="color",
         type="upper",
         addCoef.col = "black",
         tl.col="black",
```

```
        tl.srt=45,
        diag=FALSE,
        col=colorRampPalette(c("#6D9EC1", "white", "#E46726"))(200))
```



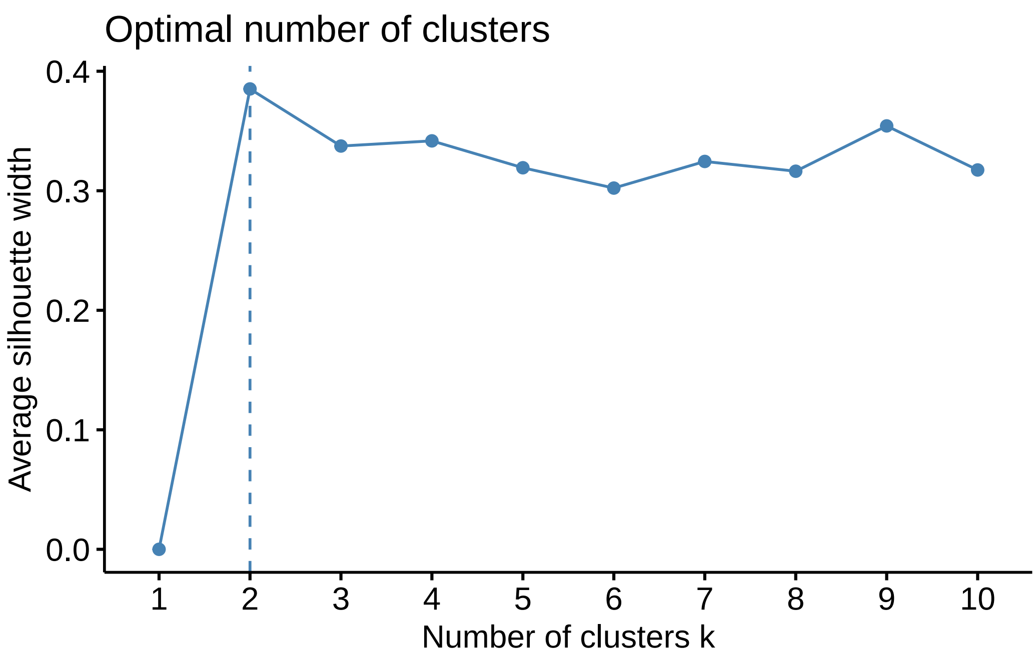

```
# 2. Factor Analysis to examine underlying structure
fa_results <- fa(key_measures, nfactors=2, rotate="varimax")
print(fa_results, cut=0.3, sort=TRUE)
```

```
Factor Analysis using method =  minres
Call: fa(r = key_measures, nfactors = 2, rotate = "varimax")
Standardized loadings (pattern matrix) based upon correlation matrix
                     item  MR1   MR2   h2     u2 com
energy_use              2  0.77        0.61 0.3856 1.1
energy_save             3  0.68        0.49 0.5146 1.1
numeracy                1  0.52        0.29 0.7067 1.2
els_score               4  0.50        0.30 0.6954 1.4
pol_conservatism        7              0.14 0.8570 2.0
env_attitude            5        0.99 1.00 0.0035 1.0
perceived_difficulty    6       -0.36 0.19 0.8120 1.7

                       MR1  MR2
SS loadings           1.72 1.31
Proportion Var        0.25 0.19
Cumulative Var        0.25 0.43
Proportion Explained  0.57 0.43
```

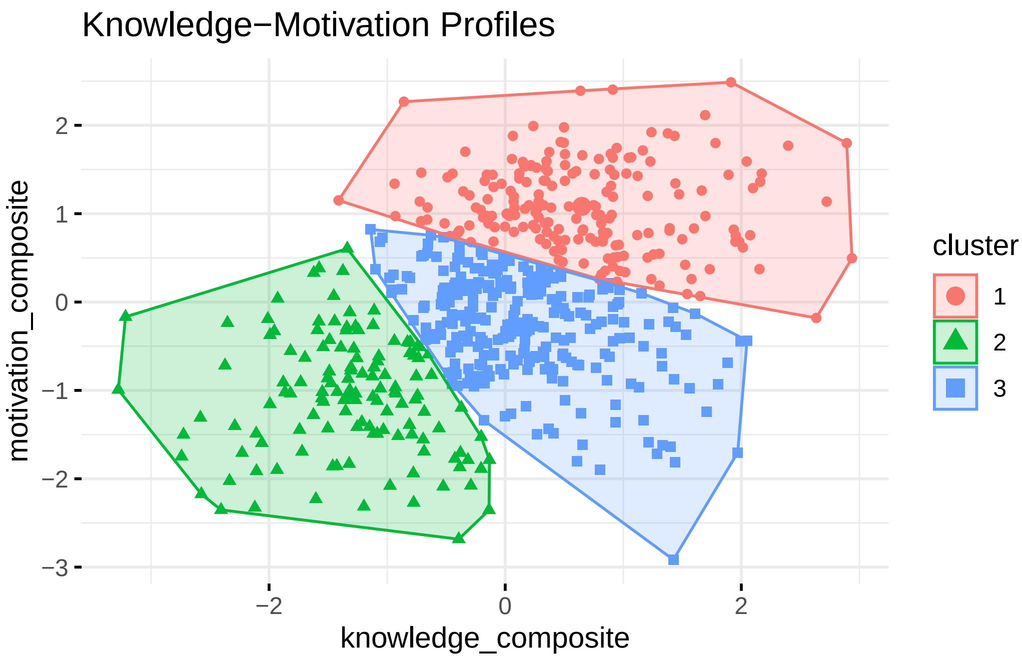
```
Cumulative Proportion 0.57 1.00

Mean item complexity =  1.4
Test of the hypothesis that 2 factors are sufficient.

df null model =  21  with the objective function =  1.3 with Chi Square =  760
df of  the model are 8  and the objective function was  0.03

The root mean square of the residuals (RMSR) is  0.03
The df corrected root mean square of the residuals is  0.04

The harmonic n.obs is  586 with the empirical chi square  17  with prob <  0.035
The total n.obs was  586  with Likelihood Chi Square =  17  with prob <  0.029

Tucker Lewis Index of factoring reliability =  0.97
RMSEA index =  0.044  and the 90 % confidence intervals are  0.014 0.073
BIC =  -34
Fit based upon off diagonal values = 0.99
Measures of factor score adequacy
                                                   MR1  MR2
Correlation of (regression) scores with factors   0.87 1.00
Multiple R square of scores with factors          0.76 0.99
Minimum correlation of possible factor scores     0.52 0.99
```

```r
# 3. Create composite scores and analyze relationships
combined_scores <- combined_scores %>%
  mutate(
    # Knowledge composite (z-score mean)
    knowledge_composite = scale(rowMeans(
      cbind(scale(numeracy), scale(energy_use),
            scale(energy_save), scale(els_score)))),
    # Motivation composite
    motivation_composite = scale(rowMeans(
      cbind(scale(env_attitude), -scale(perceived_difficulty),
            -scale(pol_conservatism))))
  )

# 4. Profile Analysis using cluster analysis
# Standardize variables for clustering
cluster_vars <- combined_scores %>%
```

```r
  select(knowledge_composite, motivation_composite) %>%
  na.omit() %>%
  scale()

# Determine optimal number of clusters
fviz_nbclust(cluster_vars, kmeans, method="silhouette")
```




```r
# Perform k-means clustering
set.seed(123)
k <- 3  # Based on silhouette plot
clusters <- kmeans(cluster_vars, centers=k, nstart=25)

# Visualize clusters
fviz_cluster(clusters, data=cluster_vars,
             geom="point",
             ellipse.type="convex",
             ggtheme=theme_minimal()) +
  labs(title="Knowledge-Motivation Profiles")
```




```
# 5. Regression Analysis
# Model 1: Knowledge predicting motivation
model_1 <- lm(motivation_composite ~ knowledge_composite, data=combined_scores)

# Model 2: Controlling for demographics if available
model_2 <- lm(motivation_composite ~ knowledge_composite + pol_conservatism,
              data=combined_scores)

summary(model_1)
```

```

Call:
lm(formula = motivation_composite ~ knowledge_composite, data = combined_scores)

Residuals:
   Min     1Q Median     3Q    Max
-3.506 -0.616  0.004  0.651  2.623

Coefficients:
                                Estimate           Std. Error t value
(Intercept)         -0.000000000000000109  0.037634207769266410       0
knowledge_composite  0.414061913664111936  0.037666360024399721      11
                              Pr(>|t|)
(Intercept)                          1
knowledge_composite <0.0000000000000002 ***
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.91 on 584 degrees of freedom
Multiple R-squared:  0.171, Adjusted R-squared:  0.17
F-statistic:  121 on 1 and 584 DF,  p-value: <0.0000000000000002
```

```r
summary(model_2)
```

```
Call:
lm(formula = motivation_composite ~ knowledge_composite + pol_conservatism,
    data = combined_scores)

Residuals:
   Min     1Q Median     3Q    Max
-2.273 -0.465  0.018  0.482  2.119

Coefficients:
                    Estimate Std. Error t value             Pr(>|t|)
(Intercept)           1.0955     0.0649   16.89 < 0.0000000000000002 ***
knowledge_composite   0.2377     0.0311    7.65    0.000000000000081 ***
pol_conservatism     -0.4154     0.0219  -18.99 < 0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.72 on 583 degrees of freedom
Multiple R-squared:  0.488, Adjusted R-squared:  0.486
F-statistic:  278 on 2 and 583 DF,  p-value: <0.0000000000000002
```

```r
# 6. Examine cluster profiles
cluster_profiles <- combined_scores %>%
  mutate(cluster = factor(clusters$cluster)) %>%
  group_by(cluster) %>%
  summarise(
    n = n(),
    mean_knowledge = mean(knowledge_composite, na.rm=TRUE),
    sd_knowledge = sd(knowledge_composite, na.rm=TRUE),
    mean_motivation = mean(motivation_composite, na.rm=TRUE),
    sd_motivation = sd(motivation_composite, na.rm=TRUE)
  )
```

```r
print(cluster_profiles)
```

```
# A tibble: 3 x 6
  cluster     n mean_knowledge sd_knowledge mean_motivation sd_motivation
  <fct>   <int>          <dbl>        <dbl>           <dbl>         <dbl>
1 1         194          0.649        0.787           1.08          0.487
2 2         125         -1.32         0.632          -1.07          0.676
3 3         267          0.145        0.649          -0.285         0.582
```

3c

```r
# Merge them into one "long" dataframe with all items.
all_items <- aes_combined %>%
  full_join(att2_combined, by = "id") %>%
  full_join(els,           by = "id") %>%
  full_join(rs,            by = "id")

# Inspect how many rows (should match total unique respondents if IDs match)
dim(all_items)
```

```
[1] 586  48
```

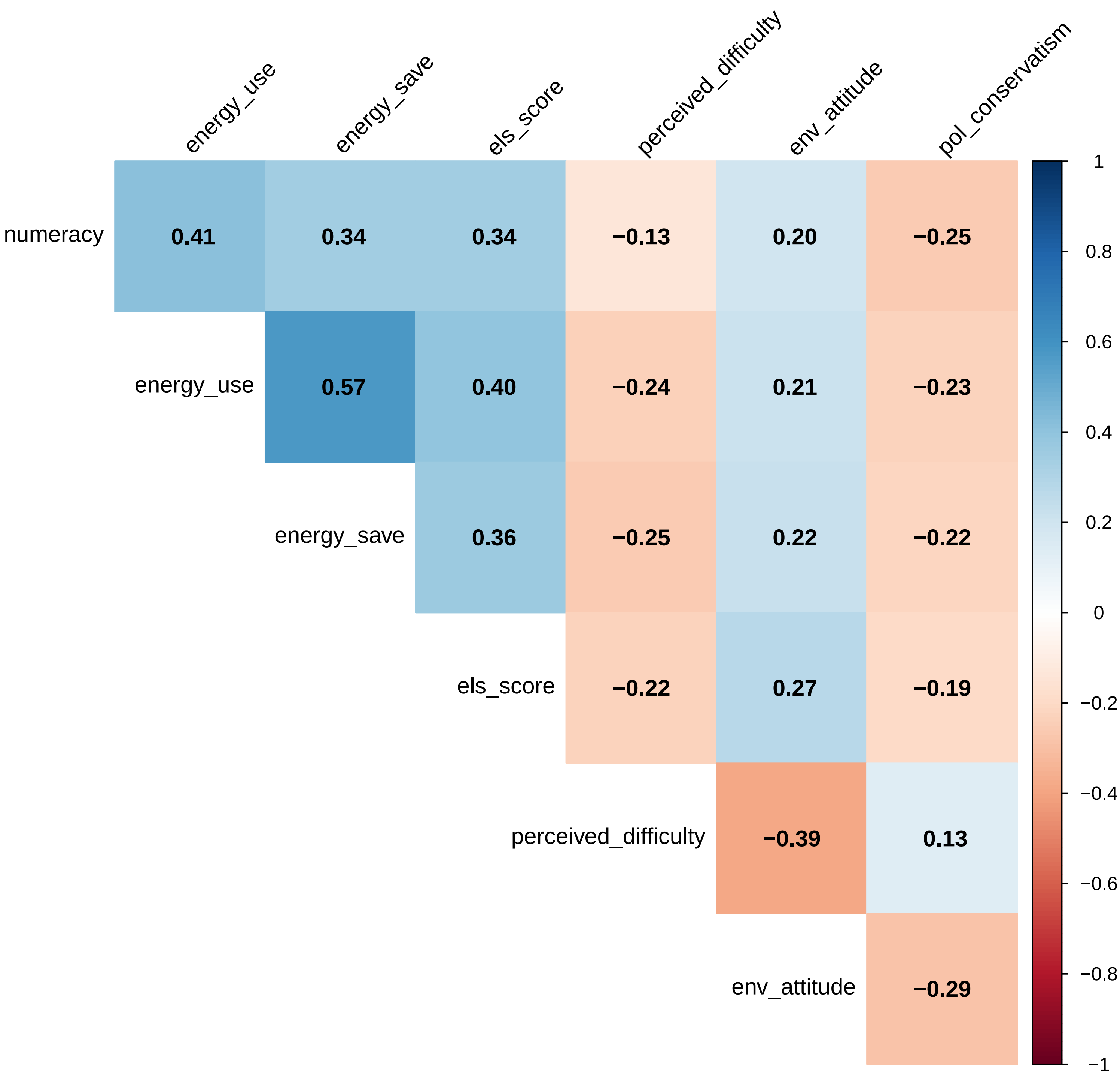
```r
head(all_items)
```

```
# A tibble: 6 x 48
     id ATT01    ATT02   ATT03   ATT04   ATT05   ATT06   ATT07   ATT08   ATT09
  <int> <dbl+lb> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl+l>
1     1 6 [Some~ 3 [Ver~ 5 [Nei~ 6 [Som~ 6 [Som~ 5 [Nei~ 5 [Nei~ 5 [Nei~ 4 [Som~
2     2 7 [Very~ 2 [Ext~ 5 [Nei~ 1 [Do ~ 7 [Ver~ 1 [Do ~ 1 [Do ~ 4 [Som~ 4 [Som~
3     3 7 [Very~ 6 [Som~ 8 [Ext~ 6 [Som~ 8 [Ext~ 6 [Som~ 8 [Ext~ 6 [Som~ 5 [Nei~
4     4 6 [Some~ 5 [Nei~ 7 [Ver~ 3 [Ver~ 6 [Som~ 1 [Do ~ 1 [Do ~ 5 [Nei~ 1 [Do ~
5     5 5 [Neit~ 6 [Som~ 4 [Som~ 5 [Nei~ 4 [Som~ 5 [Nei~ 6 [Som~ 6 [Som~ 5 [Nei~
6     6 6 [Some~ 4 [Som~ 6 [Som~ 2 [Ext~ 8 [Ext~ 1 [Do ~ 1 [Do ~ 3 [Ver~ 1 [Do ~
# i 38 more variables: ATT10 <dbl+lbl>, ATT11 <dbl+lbl>, ATT12 <dbl+lbl>,
#   ATT13 <dbl+lbl>, ATT14 <dbl+lbl>, ATT15 <dbl+lbl>, ATT16 <dbl>,
#   ATT17 <dbl>, ATT18 <dbl>, ATT19 <dbl>, ATT20 <dbl>, ATT21 <dbl>,
#   ATT22 <dbl>, ATT23 <dbl>, ATT24 <dbl>, ATT25 <dbl>, ATT26 <dbl>,
#   ATT27 <dbl>, ATT28 <dbl>, ATT29 <dbl>, ATT30 <dbl>, ATT31 <dbl>,
```

```
#   ATT32 <dbl>, ATT33 <dbl>, ELS01 <dbl+lbl>, ELS02 <dbl+lbl>,
#   ELS03 <dbl+lbl>, ELS04 <dbl+lbl>, ELS05 <dbl+lbl>, ELS06 <dbl+lbl>, ...
```

```r
# Alternatively, if you want only the *summarized scale scores*
# for each instrument (as in your existing code):
#   - attari1, attari2_scores, els_scores, rs_scores
# we can merge those:

combined_scores <- attari1 %>%
  left_join(attari2_scores, by = "id") %>%
  left_join(els_scores,     by = "id") %>%
  left_join(rs_scores,      by = "id")

# Rename columns for clarity (optional)
names(combined_scores) <- c(
  "id",
  "perceived_difficulty",  # from Attari Part 1
  "numeracy",
  "energy_use",
  "energy_save",
  "els_accuracy",
  "els_score",
  "env_attitude",
  "env_attitude_z",
  "pol_conservatism",
  "pol_conservatism_z"
)



# 2. Examine correlations and underlying structure
# A. Correlation plot among the *scale-level* variables
cor_vars <- combined_scores %>%
  select(numeracy, energy_use, energy_save, els_score,
         perceived_difficulty, env_attitude, pol_conservatism)

cor_matrix <- cor(cor_vars, use = "pairwise.complete.obs")

# Visualize correlation matrix
corrplot::corrplot(
  cor_matrix,
```

```
  method = "color",
  addCoef.col = "black",
  type = "upper",
  tl.col  = "black",
  tl.srt  = 45,
  diag    = FALSE
)
```



```r
# B. Factor Analysis (to see if knowledge & motivation load differently)
#    Using, e.g., 2-factor solution as a demonstration:
fa_data <- cor_vars %>%
  na.omit()

two_factor <- fa(fa_data, nfactors = 2, rotate = "varimax", fm = "ml")
print(two_factor, cut = 0.30, sort = TRUE)
```

```
Factor Analysis using method =  ml
Call: fa(r = fa_data, nfactors = 2, rotate = "varimax", fm = "ml")
Standardized loadings (pattern matrix) based upon correlation matrix
                     item   ML2   ML1   h2    u2 com
energy_use              2  0.78        0.63 0.374 1.1
energy_save             3  0.69        0.50 0.497 1.1
numeracy                1  0.51        0.28 0.720 1.2
els_score               4  0.49        0.29 0.709 1.4
pol_conservatism        7              0.14 0.860 2.0
env_attitude            6        0.99 1.00 0.005 1.0
perceived_difficulty    5       -0.37 0.19 0.807 1.7

                       ML2  ML1
SS loadings           1.71 1.31
Proportion Var        0.24 0.19
Cumulative Var        0.24 0.43
Proportion Explained  0.57 0.43
Cumulative Proportion 0.57 1.00

Mean item complexity =  1.4
Test of the hypothesis that 2 factors are sufficient.

df null model =  21  with the objective function =  1.3 with Chi Square =  760
df of  the model are 8  and the objective function was  0.03

The root mean square of the residuals (RMSR) is  0.03
The df corrected root mean square of the residuals is  0.04

The harmonic n.obs is  586 with the empirical chi square  18  with prob <  0.025
The total n.obs was  586  with Likelihood Chi Square =  16  with prob <  0.036

Tucker Lewis Index of factoring reliability =  0.97
```

```
RMSEA index =  0.042  and the 90 % confidence intervals are  0.01 0.072
BIC =  -35
Fit based upon off diagonal values = 0.99
Measures of factor score adequacy
                                                   ML2  ML1
Correlation of (regression) scores with factors   0.87 1.00
Multiple R square of scores with factors          0.77 0.99
Minimum correlation of possible factor scores     0.53 0.99
```

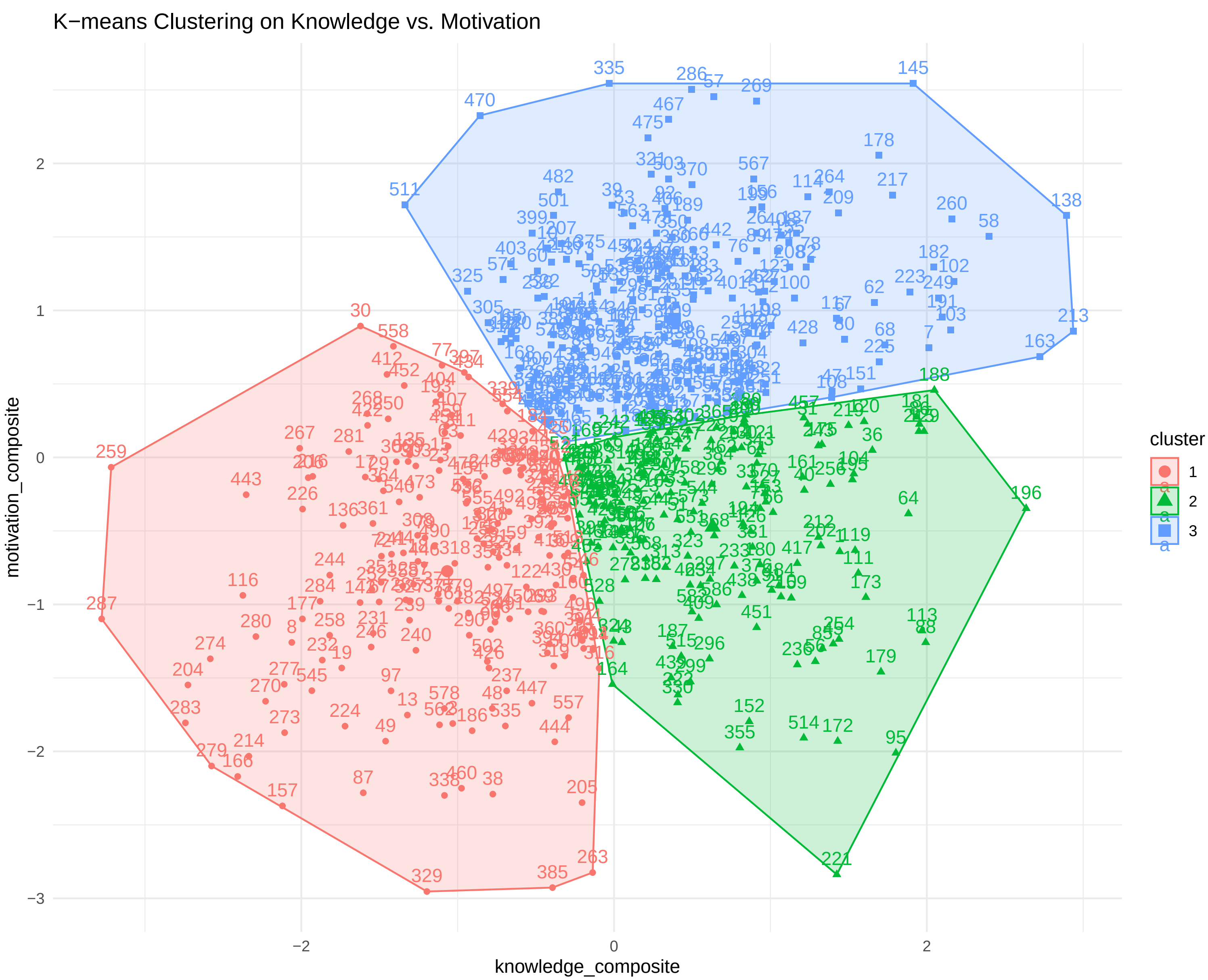
```
# ============================================================
# 3. Inspect the relation between "motivation" and "knowledge"
# ============================================================

# --- (a) Simple regressions
# For example: Predict ELS knowledge (els_score) from motivation variables
model_els <- lm(els_score ~ env_attitude + perceived_difficulty + pol_conservatism,
                data = combined_scores)
summary(model_els)
```

```

Call:
lm(formula = els_score ~ env_attitude + perceived_difficulty +
    pol_conservatism, data = combined_scores)

Residuals:
   Min     1Q Median     3Q    Max
-3.171 -0.646 -0.004  0.716  2.333

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)           -0.6422     0.2390   -2.69   0.0074 **
env_attitude           0.2421     0.0571    4.24 0.000026 ***
perceived_difficulty  -0.1338     0.0425   -3.15   0.0017 **
pol_conservatism      -0.0854     0.0289   -2.96   0.0032 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.95 on 582 degrees of freedom
Multiple R-squared:  0.104, Adjusted R-squared:  0.0992
F-statistic: 22.5 on 3 and 582 DF,  p-value: 0.0000000000000872
```

```r
# --- (b) Create composites and correlate them
# Composite for "knowledge": average of numeracy, energy_use, energy_save, els_score
combined_scores <- combined_scores %>%
  mutate(
    knowledge_composite = rowMeans(
      cbind(numeracy, energy_use, energy_save, els_score),
      na.rm = TRUE
    ),
    # For "motivation," you might choose env_attitude and reverse-coded difficulty,
    # or some other conceptual combination. Example:
    motivation_composite = rowMeans(
      cbind(env_attitude, -1 * perceived_difficulty),
      na.rm = TRUE
    )
  )

cor(combined_scores$knowledge_composite, combined_scores$motivation_composite,
    use = "pairwise.complete.obs")
```

```
[1] 0.35
```

```r
cluster_data <- combined_scores %>%
  select(knowledge_composite, motivation_composite) %>%
  na.omit() %>%
  scale()

# Decide the number of clusters (k). Let's try k = 3 for illustration:
set.seed(123)
km_fit <- kmeans(cluster_data, centers = 3, nstart = 25)

# Visualize clusters
fviz_cluster(km_fit, data = cluster_data) +
  labs(title = "K-means Clustering on Knowledge vs. Motivation") +
  theme_minimal()
```




```
# Add cluster membership back to your main dataframe
combined_scores$km_cluster <- factor(km_fit$cluster)


# Compare mean knowledge & motivation by cluster
combined_scores %>%
  group_by(km_cluster) %>%
  summarise(
    mean_knowledge = mean(knowledge_composite, na.rm = TRUE),
    mean_motivation = mean(motivation_composite, na.rm = TRUE),
    n = n()
  )
```

```
# A tibble: 3 x 4
  km_cluster mean_knowledge mean_motivation     n
```

```
  <fct>                 <dbl>          <dbl> <int>
1 1                    -0.794           1.22   184
2 2                     0.469           1.44   167
3 3                     0.288           2.49   235
```

```r
# --- (d) Canonical Correlation Analysis (CCA)
# Splitting your knowledge vs. motivation sets

knowledge_set <- combined_scores %>%
  select(numeracy, energy_use, energy_save, els_score) %>%
  na.omit()

motivation_set <- combined_scores %>%
  select(env_attitude, perceived_difficulty, pol_conservatism) %>%
  na.omit()

# We need same rows for both sets, so do a quick align:
cca_df <- na.omit(data.frame(knowledge_set, motivation_set))
K <- cca_df[, 1:4]
M <- cca_df[, 5:7]

cca_res <- cancor(K, M)
cca_res$cor  # canonical correlations
```

```
[1] 0.418 0.117 0.088
```

```r
# Inspect canonical weights and loadings
cca_res$xcoef
```

```
             [,1]   [,2]   [,3]    [,4]
numeracy    0.011  0.044  0.004 -0.0076
energy_use  0.011 -0.006 -0.029  0.0432
energy_save 0.016 -0.016 -0.013 -0.0444
els_score   0.017 -0.016  0.039  0.0086
```

```r
cca_res$ycoef
```

```
                       [,1]    [,2]  [,3]
env_attitude          0.022 -0.0009 0.056
perceived_difficulty -0.019  0.0312 0.026
pol_conservatism     -0.015 -0.0222 0.014
```

```r
# Hypothetical model:
#   - latent Knowledge from numeracy, energy_use, energy_save, els_score
#   - latent Motivation from env_attitude, perceived_difficulty
#   - regression: Knowledge ~ Motivation

sem_model <- '
  Knowledge =~ numeracy + energy_use + energy_save + els_score
  Motivation =~ env_attitude + perceived_difficulty
  Knowledge ~ Motivation
'
fit_sem <- sem(sem_model, data = combined_scores, missing = "fiml")  # handle missing if needed
summary(fit_sem, fit.measures = TRUE, standardized = TRUE)
```

```
lavaan 0.6-19 ended normally after 36 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                        19

  Number of observations                           586
  Number of missing patterns                         1

Model Test User Model:

  Test statistic                                23.012
  Degrees of freedom                                 8
  P-value (Chi-square)                           0.003

Model Test Baseline Model:

  Test statistic                               680.231
  Degrees of freedom                                15
  P-value                                        0.000

User Model versus Baseline Model:

  Comparative Fit Index (CFI)                    0.977
  Tucker-Lewis Index (TLI)                       0.958

  Robust Comparative Fit Index (CFI)             0.977
```

```
  Robust Tucker-Lewis Index (TLI)                0.958

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -4504.774
  Loglikelihood unrestricted model (H1)      -4493.268

  Akaike (AIC)                                9047.548
  Bayesian (BIC)                              9130.641
  Sample-size adjusted Bayesian (SABIC)       9070.323

Root Mean Square Error of Approximation:

  RMSEA                                          0.057
  90 Percent confidence interval - lower         0.030
  90 Percent confidence interval - upper         0.084
  P-value H_0: RMSEA <= 0.050                    0.305
  P-value H_0: RMSEA >= 0.080                    0.086

  Robust RMSEA                                   0.057
  90 Percent confidence interval - lower         0.030
  90 Percent confidence interval - upper         0.084
  P-value H_0: Robust RMSEA <= 0.050             0.305
  P-value H_0: Robust RMSEA >= 0.080             0.086

Standardized Root Mean Square Residual:

  SRMR                                           0.028

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Observed
  Observed information based on                Hessian

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  Knowledge =~
    numeracy          1.000                               0.527    0.527
    energy_use        1.469    0.135   10.849    0.000    0.774    0.774
```

```
    energy_save       1.352    0.129   10.452    0.000    0.712    0.713
    els_score         1.029    0.110    9.400    0.000    0.542    0.543
  Motivation =~
    env_attitude      1.000                               0.477    0.619
    percvd_dffclty   -1.306    0.208   -6.269    0.000   -0.622   -0.623

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  Knowledge ~
    Motivation        0.584    0.099    5.923    0.000    0.529    0.529

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .numeracy          0.000    0.041    0.000    1.000    0.000    0.000
   .energy_use       -0.000    0.041   -0.000    1.000   -0.000   -0.000
   .energy_save      -0.000    0.041   -0.000    1.000   -0.000   -0.000
   .els_score         0.000    0.041    0.000    1.000    0.000    0.000
   .env_attitude      3.583    0.032  112.638    0.000    3.583    4.653
   .percvd_dffclty    0.000    0.041    0.000    1.000    0.000    0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .numeracy          0.721    0.048   15.124    0.000    0.721    0.722
   .energy_use        0.400    0.042    9.432    0.000    0.400    0.400
   .energy_save       0.491    0.042   11.751    0.000    0.491    0.492
   .els_score         0.704    0.047   14.845    0.000    0.704    0.705
   .env_attitude      0.366    0.041    8.844    0.000    0.366    0.617
   .percvd_dffclty    0.611    0.070    8.711    0.000    0.611    0.612
   .Knowledge         0.200    0.036    5.567    0.000    0.721    0.721
    Motivation        0.227    0.045    5.079    0.000    1.000    1.000
```