Towards Effective Human Intervention in Algorithmic Decision-Making

Anonymized.

August 29, 2024

Contents

1 Importing the Packages

We will start with importing the required packages for the analysis.

```r
library(tidyverse)
library(readxl)
library(scales)
library(gridExtra)
library(gplots)
library(RColorBrewer)
library(FactoMineR)
library(nlme)
library(rcompanion)
library(here)
library(ggalluvial)
library(corrplot)
library(wesanderson)
library(psych)
library(mgcv)
library(mblm)
library(quantreg)
library(ggpubr)
library(factoextra)
library(mediation)
library(semPlot)
library(flexplot)
library(rstatix)
```

2 Loading Data

```r
# Set the path.
path = paste0(here(), "/Data/")
# Read the XLSX file.
dt = read_excel(paste0(path, "cleaned_data.xlsx"))
# Cleaning up.
rm(path)
```

Printing the names of the columns.

```r
# Print column names.
names(dt)
```

```
##  [1] "...1"            "UserID"          "Time"            "Age"
##  [5] "Education"       "LesseeAirbnb_1"  "LesseeAirbnb_2"  "AI_literacy"
##  [9] "Aff_tech"        "ExperienceHuman" "ExperienceAI"    "GoodPublicAdmin"
## [13] "AgreementPolicy" "Complexity"      "Profile"         "Model_type"
## [17] "Data_type"       "Ability"         "Ability_1"       "Ability_2"
## [21] "Ability_3"       "Ability_4"       "Ability_5"       "Ability_6"
## [25] "Benevolence"     "Benevolence_1"   "Benevolence_2"   "Benevolence_3"
## [29] "Benevolence_4"   "Benevolence_5"   "Integrity"       "Integrity_1"
## [33] "Integrity_2"     "Integrity_3"     "Integrity_4"     "Integrity_5"
## [37] "Integrity_6"     "Overall"         "Able"            "Benevolent"
## [41] "Integer"
```

3 Cleaning Data

First, let's drop a few columns which are not necessary for our analysis.

```
# Remove unnecessary columns.
dt = dt[, c(2:5, 8:41)]
# Removing Time, Age, Education, Able, Benevolent, Integer
dt = dt[, c(1, 5:35)]
# dt = dt %>% select(!c("Time", "Age", "Education", "Able",
#                       "Benevolent", "Integer"))
```

Second, let's rename the column names to comply with R standards.

```
# Rename columns.
names(dt) = c("ID",
              "AI.Literacy.Lik", "Affinity.Technology.Lik", "Exp.Human.Lik",
              "Exp.AI.Lik", "Good.Public.Admin.Lik", "Agreement.Policy.Lik",
              "Complexity.Lik", "Profile.Cat", "Model.Type.Cat",
              "Data.Type.Cat", "Ability.Agg.Dbl", "Ability.Q1.Lik",
              "Ability.Q2.Lik", "Ability.Q3.Lik", "Ability.Q4.Lik",
              "Ability.Q5.Lik", "Ability.Q6.Lik", "Benevolence.Agg.Dbl",
              "Benevolence.Q1.Lik", "Benevolence.Q2.Lik", "Benevolence.Q3.Lik",
              "Benevolence.Q4.Lik", "Benevolence.Q5.Lik", "Integrity.Agg.Dbl",
              "Integrity.Q1.Lik", "Integrity.Q2.Lik", "Integrity.Q3.Lik",
              "Integrity.Q4.Lik", "Integrity.Q5.Lik", "Integrity.Q6.Lik",
              "Fairness.Perception.Lik")
```

4 Hypothesis Testing

We will conduct the assumption tests (for normality and homoscedasticity) before proceeding with the actual –parametric and non-parametric– tests.

4.1 H_{1a}: Profile -> Ability (***)

a) Shapiro test for normality.

```
# Shapiro test.
shapiro.test(dt$Ability.Agg.Dbl)

##
##  Shapiro-Wilk normality test
##
## data:  dt$Ability.Agg.Dbl
## W = 0.96304, p-value = 1.528e-05
```

The results show that since the `p<.05` the data is `not normal`.

b) Bartlett test for homoscedasticity.

```
# Bartlett test.
bartlett.test(dt$Ability.Agg.Dbl ~ dt$Profile.Cat)

##
##  Bartlett test of homogeneity of variances
##
## data:  dt$Ability.Agg.Dbl by dt$Profile.Cat
## Bartlett's K-squared = 2.6174, df = 1, p-value = 0.1057
```

The results show that since the `p>.05` the variance is `homogeneous`.

c) ANOVA or Kurskal-Wallis.

```
# t-test
t.test(dt$Ability.Agg.Dbl ~ dt$Profile.Cat)
```

```
## 
##  Welch Two Sample t-test
## 
## data:  dt$Ability.Agg.Dbl by dt$Profile.Cat
## t = -10.262, df = 218.41, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group Fully automated and group Hybrid is not
## 95 percent confidence interval:
##  -2.092455 -1.418195
## sample estimates:
## mean in group Fully automated          mean in group Hybrid
##                     -0.4005848                     1.3547401

# Mann-Whitney U
wilcox.test(dt$Ability.Agg.Dbl ~ dt$Profile.Cat)

## 
##  Wilcoxon rank sum test with continuity correction
## 
## data:  dt$Ability.Agg.Dbl by dt$Profile.Cat
## W = 2130.5, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0

# ANOVA
oneway.test(dt$Ability.Agg.Dbl ~ dt$Profile.Cat)

## 
##  One-way analysis of means (not assuming equal variances)
## 
## data:  dt$Ability.Agg.Dbl and dt$Profile.Cat
## F = 105.3, num df = 1.00, denom df = 218.41, p-value < 2.2e-16

# Kruskal-Wallis
kruskal.test(dt$Ability.Agg.Dbl ~ dt$Profile.Cat)

## 
##  Kruskal-Wallis rank sum test
## 
## data:  dt$Ability.Agg.Dbl by dt$Profile.Cat
## Kruskal-Wallis chi-squared = 72.013, df = 1, p-value < 2.2e-16

epsilonSquared(x = dt$Ability.Agg.Dbl,
               g = dt$Profile.Cat)

## epsilon.squared
##           0.324
```
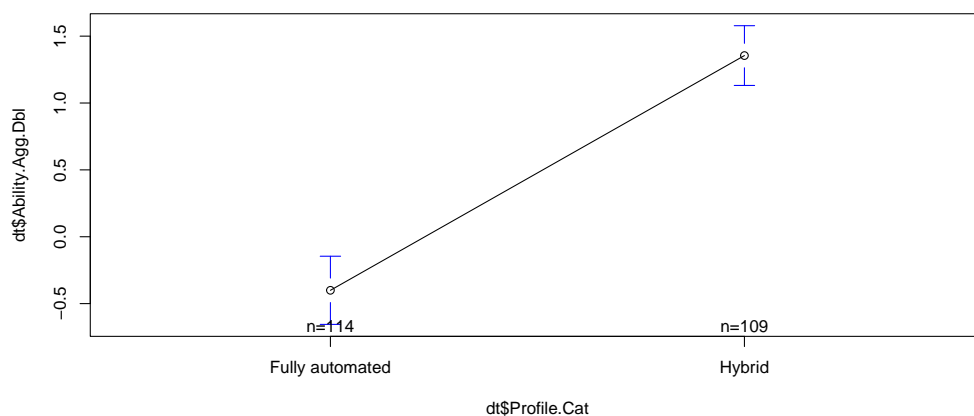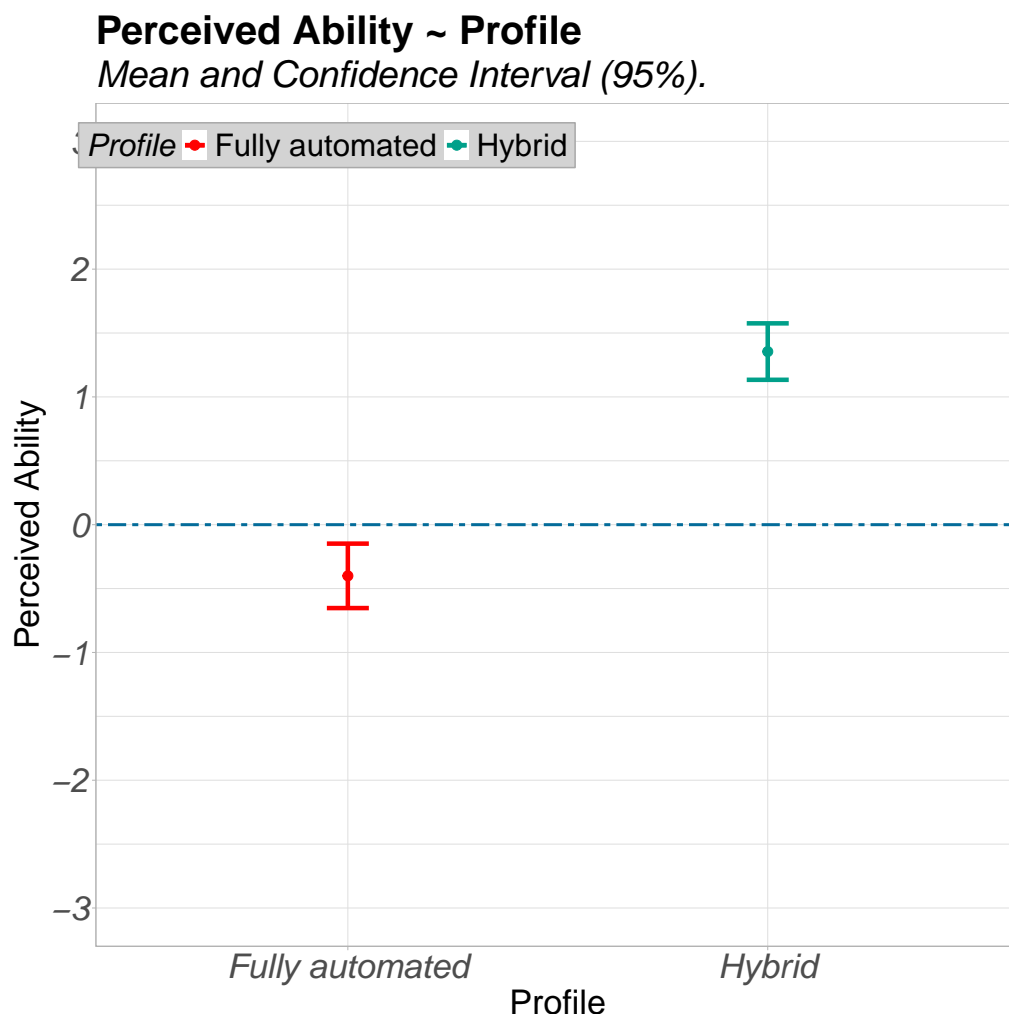
d) Visualize.

Using GGPlot to make a publication ready plot.

```
## Warning:  Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was generated.
## Warning:  The 'size' argument of 'element_rect()' is deprecated as of ggplot2 3.4.0.
## i Please use the 'linewidth' argument instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was generated.
## Warning:  A numeric 'legend.position' argument in 'theme()' was deprecated in ggplot2 3.5.0.
## i Please use the 'legend.position.inside' argument of 'theme()' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was generated.
```


Perceived Ability ~ Profile
Mean and Confidence Interval (95%).

e) Effect sizes (Eta-Square and Cohen's F).

```
# Effect Size ANOVA (Generalized Eta-Sq and Cohen's F-Value)
eff = anova_test(data = dt, Ability.Agg.Dbl ~ Profile.Cat, effect.size = "ges")
# Eta-Square Value.
eff

## ANOVA Table (type II tests)
##
##        Effect DFn DFd       F        p p<.05   ges
## 1 Profile.Cat   1 221 104.578 2.39e-20     * 0.321

# Now computing Cohen's f using eta-sq.
f = sqrt(eff$ges/(1-eff$ges))
# Cohen's F
f

## [1] 0.6875711
```

```
# Kruskal-Wallis Eta-Sq
ordinalEtaSquared(x = dt$Ability.Agg.Dbl,
                  g = dt$Profile.Cat)

## eta.squared
##      0.321
```

4.2 H_{1b}: Profile -> Benevolence (***)

a) Shapiro test for normality.

```
# Shapiro test.
shapiro.test(dt$Benevolence.Agg.Dbl)

##
##  Shapiro-Wilk normality test
##
## data:  dt$Benevolence.Agg.Dbl
## W = 0.93656, p-value = 2.992e-08
```

The results show that since the `p<.05` the data is `not normal`.

b) Bartlett test for homoscedasticity.

```
# Bartlett test.
bartlett.test(dt$Benevolence.Agg.Dbl ~ dt$Profile.Cat)

##
##  Bartlett test of homogeneity of variances
##
## data:  dt$Benevolence.Agg.Dbl by dt$Profile.Cat
## Bartlett's K-squared = 6.5296, df = 1, p-value = 0.01061
```

The results show that since the `p<.05` the variance is `not homogeneous`.

c) ANOVA or Kurskal-Wallis.

```
# t-test
t.test(dt$Benevolence.Agg.Dbl ~ dt$Profile.Cat)

##
##  Welch Two Sample t-test
##
## data:  dt$Benevolence.Agg.Dbl by dt$Profile.Cat
## t = -6.6192, df = 204.57, p-value = 3.11e-10
## alternative hypothesis: true difference in means between group Fully automated and group Hybrid is not
## 95 percent confidence interval:
##  -1.5455950 -0.8361529
## sample estimates:
## mean in group Fully automated        mean in group Hybrid
##                    -1.6789474                  -0.4880734

# Mann-Whitney U
wilcox.test(dt$Benevolence.Agg.Dbl ~ dt$Profile.Cat)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  dt$Benevolence.Agg.Dbl by dt$Profile.Cat
## W = 3182, p-value = 2.834e-10
## alternative hypothesis: true location shift is not equal to 0

# ANOVA
oneway.test(dt$Benevolence.Agg.Dbl ~ dt$Profile.Cat)
```

```
## 
##  One-way analysis of means (not assuming equal variances)
## 
## data:  dt$Benevolence.Agg.Dbl and dt$Profile.Cat
## F = 43.813, num df = 1.00, denom df = 204.57, p-value = 3.11e-10
```

```r
# Kruskal-Wallis
kruskal.test(dt$Benevolence.Agg.Dbl ~ dt$Profile.Cat)
```

```
## 
##  Kruskal-Wallis rank sum test
## 
## data:  dt$Benevolence.Agg.Dbl by dt$Profile.Cat
## Kruskal-Wallis chi-squared = 39.799, df = 1, p-value = 2.816e-10
```

```r
epsilonSquared(x = dt$Benevolence.Agg.Dbl,
               g = dt$Profile.Cat)
```

```
## epsilon.squared
##           0.179
```

d) Visualize.

Using GGPlot to make a publication ready plot.

Perceived Benevolence ~ Profile
Mean and Confidence Interval (95%).

e) Effect sizes (Eta-Square and Cohen's F).

```
# Effect Size ANOVA (Generalized Eta-Sq and Cohen's F-Value)
eff = anova_test(data = dt, Benevolence.Agg.Dbl ~ Profile.Cat, effect.size = "ges")
# Eta-Square Value.
eff

## ANOVA Table (type II tests)
##
##        Effect DFn DFd      F        p p<.05   ges
## 1 Profile.Cat   1 221 44.289 2.2e-10     * 0.167

# Now computing Cohen's f using eta-sq.
f = sqrt(eff$ges/(1-eff$ges))
# Cohen's F
f

## [1] 0.4477501

# Kruskal-Wallis Eta-Sq
ordinalEtaSquared(x = dt$Benevolence.Agg.Dbl,
                  g = dt$Profile.Cat)

## eta.squared
##       0.176
```

4.3   H$_{Ix'}$: Profile -> Integrity (***)

a) Shapiro test for normality.

```
# Shapiro test.
shapiro.test(dt$Integrity.Agg.Dbl)

##
##  Shapiro-Wilk normality test
##
## data:  dt$Integrity.Agg.Dbl
## W = 0.98623, p-value = 0.02964
```

The results show that since the `p<.05` the data is `not normal`.

b) Bartlett test for homoscedasticity.

```
# Bartlett test.
bartlett.test(dt$Integrity.Agg.Dbl ~ dt$Profile.Cat)

##
##  Bartlett test of homogeneity of variances
##
## data:  dt$Integrity.Agg.Dbl by dt$Profile.Cat
## Bartlett's K-squared = 0.00035548, df = 1, p-value = 0.985
```

The results show that since the `p>.05` the variance is `homogeneous`.

c) ANOVA or Kurskal-Wallis.

```
# t-test
t.test(dt$Integrity.Agg.Dbl ~ dt$Profile.Cat)

##
##  Welch Two Sample t-test
##
## data:  dt$Integrity.Agg.Dbl by dt$Profile.Cat
## t = -8.0436, df = 220.59, p-value = 5.315e-14
## alternative hypothesis: true difference in means between group Fully automated and group Hybrid is not
## 95 percent confidence interval:
##  -1.4145405 -0.8577917
## sample estimates:
## mean in group Fully automated        mean in group Hybrid
##                    -0.1315789                   1.0045872

# Mann-Whitney U
wilcox.test(dt$Integrity.Agg.Dbl ~ dt$Profile.Cat)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  dt$Integrity.Agg.Dbl by dt$Profile.Cat
## W = 2708, p-value = 3.234e-13
## alternative hypothesis: true location shift is not equal to 0

# ANOVA
oneway.test(dt$Integrity.Agg.Dbl ~ dt$Profile.Cat)

##
##  One-way analysis of means (not assuming equal variances)
##
## data:  dt$Integrity.Agg.Dbl and dt$Profile.Cat
## F = 64.699, num df = 1.00, denom df = 220.59, p-value = 5.315e-14

# Kruskal-Wallis
kruskal.test(dt$Integrity.Agg.Dbl ~ dt$Profile.Cat)
```

```
## 
##  Kruskal-Wallis rank sum test
## 
## data:  dt$Integrity.Agg.Dbl by dt$Profile.Cat
## Kruskal-Wallis chi-squared = 53.076, df = 1, p-value = 3.209e-13

epsilonSquared(x = dt$Integrity.Agg.Dbl,
               g = dt$Profile.Cat)

## epsilon.squared
##           0.239
```
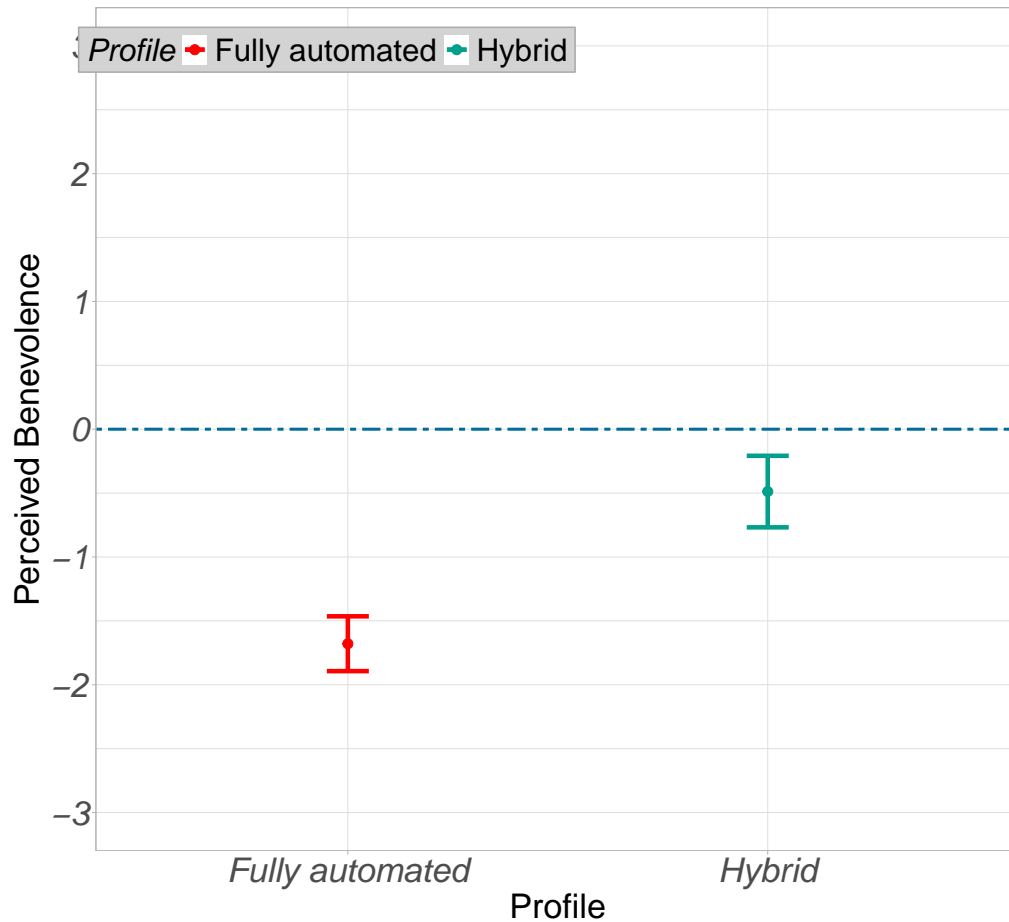
d) Visualize.

Using GGPlot to make a publication ready plot.

Perceived Integrity ~ Profile
Mean and Confidence Interval (95%).

e) Effect sizes (Eta-Square and Cohen's F).

```
# Effect Size ANOVA (Generalized Eta-Sq and Cohen's F-Value)
eff = anova_test(data = dt, Integrity.Agg.Dbl ~ Profile.Cat, effect.size = "ges")
# Eta-Square Value.
eff

## ANOVA Table (type II tests)
##
##        Effect DFn DFd      F        p p<.05   ges
## 1 Profile.Cat   1 221 64.694 5.29e-14     * 0.226

# Now computing Cohen's f using eta-sq.
f = sqrt(eff$ges/(1-eff$ges))
# Cohen's F
f

## [1] 0.5403607

# Kruskal-Wallis Eta-Sq
ordinalEtaSquared(x = dt$Integrity.Agg.Dbl,
                  g = dt$Profile.Cat)

## eta.squared
##       0.236
```

4.4 H_{lc}: Model Type -> Integrity (N.S.)

a) Shapiro test for normality.

```
# Shapiro test.
shapiro.test(dt$Integrity.Agg.Dbl)

##
##  Shapiro-Wilk normality test
##
## data:  dt$Integrity.Agg.Dbl
## W = 0.98623, p-value = 0.02964
```

The results show that since the `p<.05` the data is `not normal`.

b) Bartlett test for homoscedasticity.

```
# Bartlett test.
bartlett.test(dt$Integrity.Agg.Dbl ~ dt$Model.Type.Cat)

##
##  Bartlett test of homogeneity of variances
##
## data:  dt$Integrity.Agg.Dbl by dt$Model.Type.Cat
## Bartlett's K-squared = 0.13954, df = 1, p-value = 0.7087
```

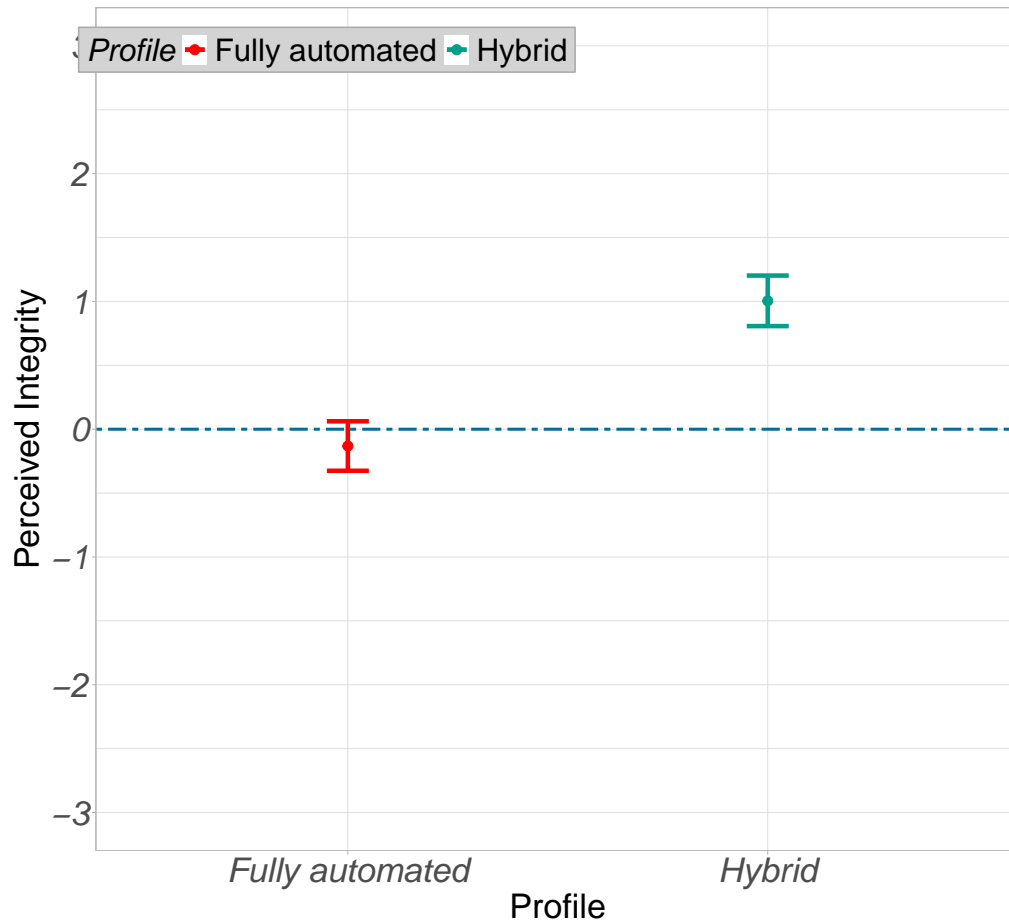The results show that since the `p>.05` the variance is `homogeneous`.

c) ANOVA or Kurskal-Wallis.

```
# t-test
t.test(dt$Integrity.Agg.Dbl ~ dt$Model.Type.Cat)

##
##  Welch Two Sample t-test
##
## data:  dt$Integrity.Agg.Dbl by dt$Model.Type.Cat
## t = -0.032974, df = 220.56, p-value = 0.9737
## alternative hypothesis: true difference in means between group Probabilistic and group Rule-based is no
```

```
## 95 percent confidence interval:
##  -0.3217894  0.3111985
## sample estimates:
## mean in group Probabilistic    mean in group Rule-based
##                   0.4211310                    0.4264264

# Mann-Whitney U
wilcox.test(dt$Integrity.Agg.Dbl ~ dt$Model.Type.Cat)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  dt$Integrity.Agg.Dbl by dt$Model.Type.Cat
## W = 6094.5, p-value = 0.8015
## alternative hypothesis: true location shift is not equal to 0

# ANOVA
oneway.test(dt$Integrity.Agg.Dbl ~ dt$Model.Type.Cat)

##
##  One-way analysis of means (not assuming equal variances)
##
## data:  dt$Integrity.Agg.Dbl and dt$Model.Type.Cat
## F = 0.0010873, num df = 1.00, denom df = 220.56, p-value = 0.9737

# Kruskal-Wallis
kruskal.test(dt$Integrity.Agg.Dbl ~ dt$Model.Type.Cat)

##
##  Kruskal-Wallis rank sum test
##
## data:  dt$Integrity.Agg.Dbl by dt$Model.Type.Cat
## Kruskal-Wallis chi-squared = 0.063748, df = 1, p-value = 0.8007

epsilonSquared(x = dt$Integrity.Agg.Dbl,
               g = dt$Model.Type.Cat)

## epsilon.squared
##        0.000287
```
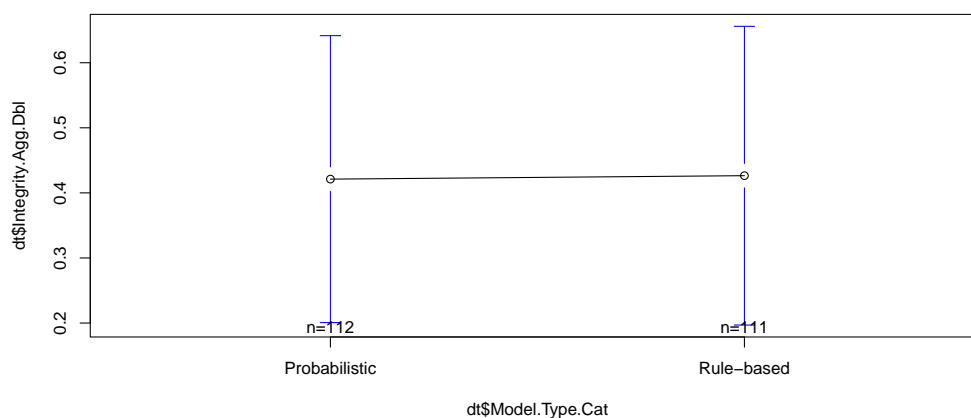
d) Visualize.

Using GGPlot to make a publication ready plot.

Perceived Integrity ~ Model Type
Mean and Confidence Interval (95%).

e) Effect sizes (Eta-Square and Cohen's F).

```
# Effect Size ANOVA (Generalized Eta-Sq and Cohen's F-Value)
eff = anova_test(data = dt, Integrity.Agg.Dbl ~ Model.Type.Cat, effect.size = "ges")
# Eta-Square Value.
eff

## ANOVA Table (type II tests)
##
##          Effect DFn DFd     F     p p<.05      ges
## 1 Model.Type.Cat   1 221 0.001 0.974       4.92e-06

# Now computing Cohen's f using eta-sq.
f = sqrt(eff$ges/(1-eff$ges))
# Cohen's F
f

## [1] 0.002218113

# Kruskal-Wallis Eta-Sq
ordinalEtaSquared(x = dt$Integrity.Agg.Dbl,
                  g = dt$Model.Type.Cat)

## eta.squared
##    -0.00424
```

4.5 H_{ld}: Data Type -> Integrity (N.S.)

a) Shapiro test for normality.

```
# Shapiro test.
shapiro.test(dt$Integrity.Agg.Dbl)

##
##  Shapiro-Wilk normality test
##
## data:  dt$Integrity.Agg.Dbl
## W = 0.98623, p-value = 0.02964
```

The results show that since the `p<.05` the data is `not normal`.

b) Bartlett test for homoscedasticity.

```
# Bartlett test.
bartlett.test(dt$Integrity.Agg.Dbl ~ dt$Data.Type.Cat)

##
##  Bartlett test of homogeneity of variances
##
## data:  dt$Integrity.Agg.Dbl by dt$Data.Type.Cat
## Bartlett's K-squared = 1.0281, df = 1, p-value = 0.3106
```

The results show that since the `p>.05` the variance is `homogeneous`.

c) ANOVA or Kurskal-Wallis.

```
# t-test
t.test(dt$Integrity.Agg.Dbl ~ dt$Data.Type.Cat)

##
##  Welch Two Sample t-test
##
## data:  dt$Integrity.Agg.Dbl by dt$Data.Type.Cat
## t = -1.4453, df = 219.32, p-value = 0.1498
## alternative hypothesis: true difference in means between group Camera and group Public Registry is not
## 95 percent confidence interval:
##  -0.54571591  0.08395057
## sample estimates:
##         mean in group Camera mean in group Public Registry
##                    0.3078078                     0.5386905

# Mann-Whitney U
wilcox.test(dt$Integrity.Agg.Dbl ~ dt$Data.Type.Cat)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  dt$Integrity.Agg.Dbl by dt$Data.Type.Cat
## W = 5426, p-value = 0.1009
## alternative hypothesis: true location shift is not equal to 0

# ANOVA
oneway.test(dt$Integrity.Agg.Dbl ~ dt$Data.Type.Cat)

##
##  One-way analysis of means (not assuming equal variances)
##
## data:  dt$Integrity.Agg.Dbl and dt$Data.Type.Cat
## F = 2.0889, num df = 1.00, denom df = 219.32, p-value = 0.1498

# Kruskal-Wallis
kruskal.test(dt$Integrity.Agg.Dbl ~ dt$Data.Type.Cat)
```

```
## 
##  Kruskal-Wallis rank sum test
## 
## data:  dt$Integrity.Agg.Dbl by dt$Data.Type.Cat
## Kruskal-Wallis chi-squared = 2.695, df = 1, p-value = 0.1007

epsilonSquared(x = dt$Integrity.Agg.Dbl,
               g = dt$Data.Type.Cat)

## epsilon.squared
##          0.0121
```
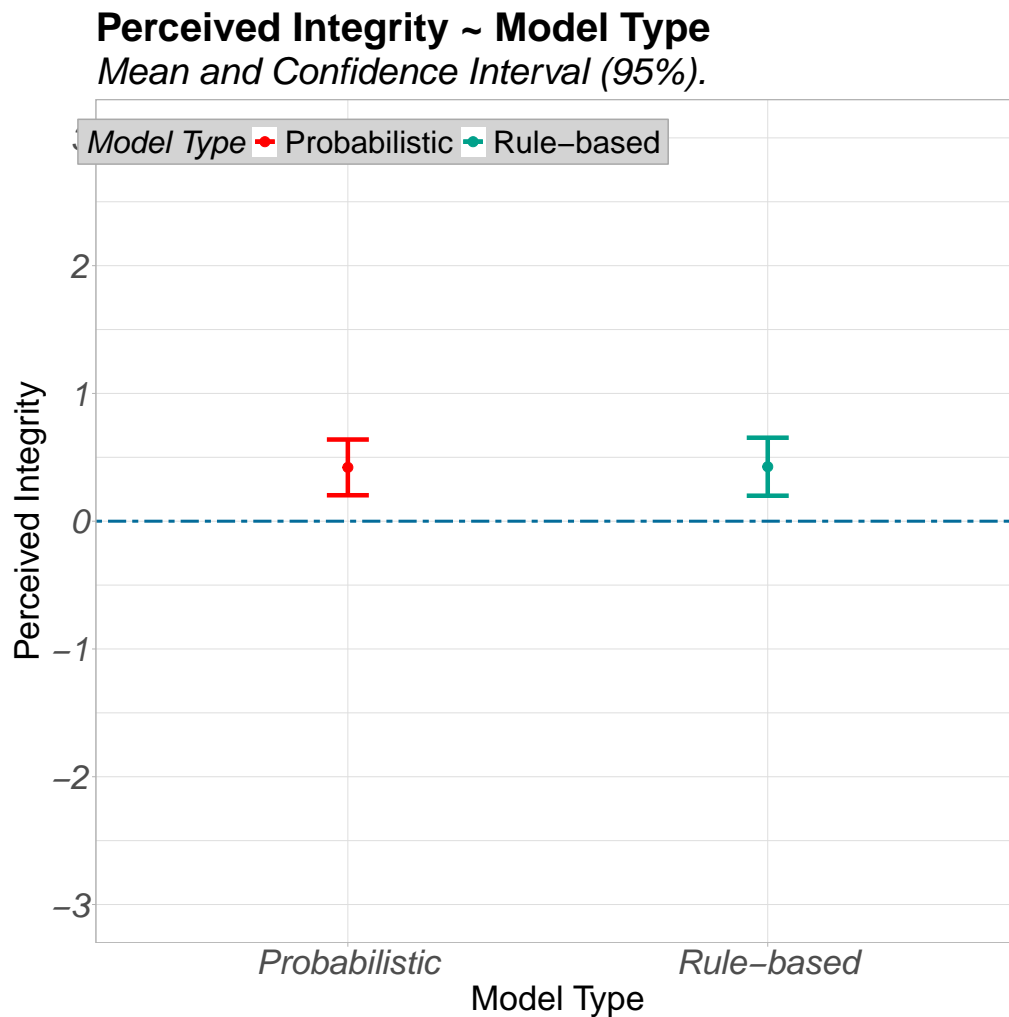
d) Visualize.

Using GGPlot to make a publication ready plot.

Perceived Integrity ~ Data Provenance
Mean and Confidence Interval (95%).

e) Effect sizes (Eta-Square and Cohen's F).

15

```
# Effect Size ANOVA (Generalized Eta-Sq and Cohen's F-Value)
eff = anova_test(data = dt, Integrity.Agg.Dbl ~ Data.Type.Cat, effect.size = "ges")
# Eta-Square Value.
eff

## ANOVA Table (type II tests)
##
##          Effect DFn DFd     F    p p<.05   ges
## 1 Data.Type.Cat   1 221 2.087 0.15        0.009

# Now computing Cohen's f using eta-sq.
f = sqrt(eff$ges/(1-eff$ges))
# Cohen's F
f

## [1] 0.09529814

# Kruskal-Wallis Eta-Sq
ordinalEtaSquared(x = dt$Integrity.Agg.Dbl,
                  g = dt$Data.Type.Cat)

## eta.squared
##     0.00767
```
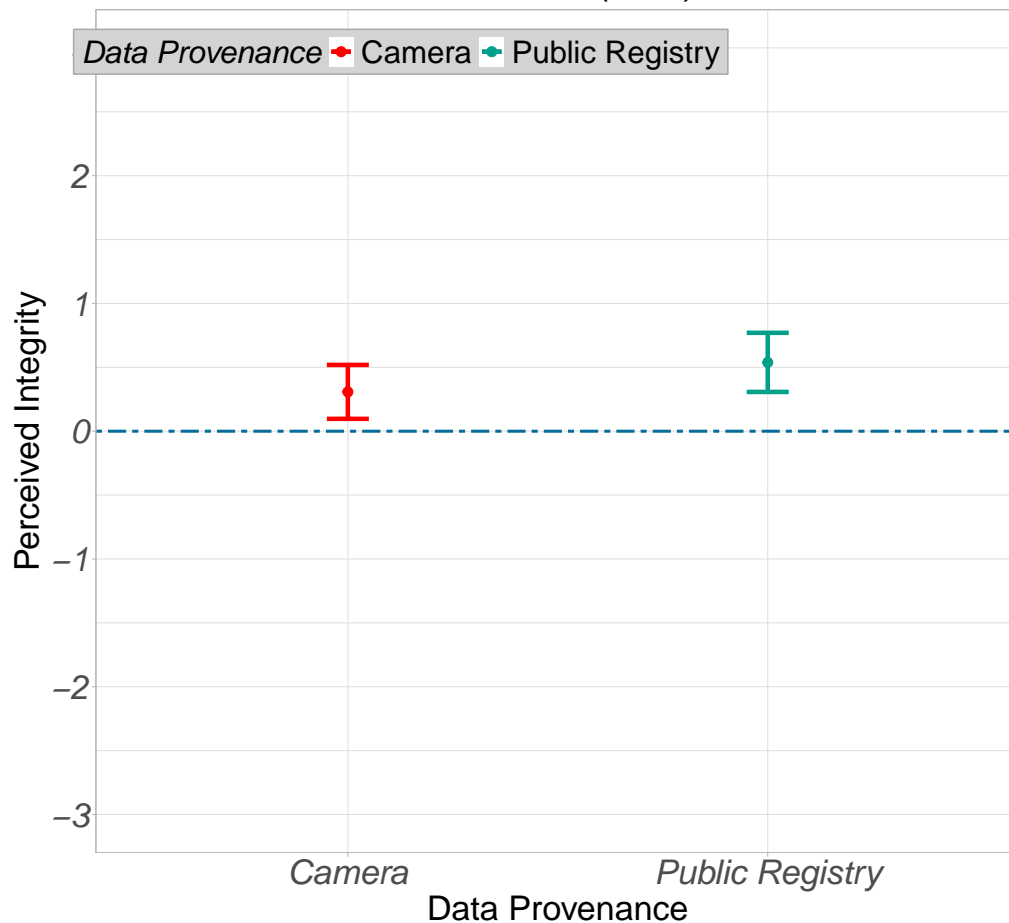
4.6   H$_{2a}$: Ability -> Fairness (***)

a) Shapiro test for normality.

```
# Shapiro test.
shapiro.test(dt$Fairness.Perception.Lik)

##
##  Shapiro-Wilk normality test
##
## data:  dt$Fairness.Perception.Lik
## W = 0.90486, p-value = 1.03e-10
```

The results show that since the `p<.05` the data is `not normal`.

в) Create a categorical variable.

```
# Median Cut.
dt$Ability.Agg.Cat = ifelse(dt$Ability.Agg.Dbl >= median(dt$Ability.Agg.Dbl),
                            "High", "Low")
```

c) Bartlett test for homoscedasticity.

```
# Bartlett test.
bartlett.test(dt$Fairness.Perception.Lik ~ dt$Ability.Agg.Cat)

##
##  Bartlett test of homogeneity of variances
##
## data:  dt$Fairness.Perception.Lik by dt$Ability.Agg.Cat
## Bartlett's K-squared = 17.019, df = 1, p-value = 3.701e-05
```

The results show that since the `p<.05` the variance is `not homogeneous`.

d) ANOVA or Kurskal-Wallis.

```
# t-test
t.test(dt$Fairness.Perception.Lik ~ dt$Ability.Agg.Cat)
```

```
## 
##  Welch Two Sample t-test
## 
## data:  dt$Fairness.Perception.Lik by dt$Ability.Agg.Cat
## t = 11.117, df = 192.24, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group High and group Low is not equal to 0
## 95 percent confidence interval:
##  1.573711 2.252544
## sample estimates:
## mean in group High  mean in group Low
##          1.6428571         -0.2702703

# Mann-Whitney U
wilcox.test(dt$Fairness.Perception.Lik ~ dt$Ability.Agg.Cat)

## 
##  Wilcoxon rank sum test with continuity correction
## 
## data:  dt$Fairness.Perception.Lik by dt$Ability.Agg.Cat
## W = 10594, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0

# ANOVA
oneway.test(dt$Fairness.Perception.Lik ~ dt$Ability.Agg.Cat)

## 
##  One-way analysis of means (not assuming equal variances)
## 
## data:  dt$Fairness.Perception.Lik and dt$Ability.Agg.Cat
## F = 123.6, num df = 1.00, denom df = 192.24, p-value < 2.2e-16

# Kruskal-Wallis
kruskal.test(dt$Fairness.Perception.Lik ~ dt$Ability.Agg.Cat)

## 
##  Kruskal-Wallis rank sum test
## 
## data:  dt$Fairness.Perception.Lik by dt$Ability.Agg.Cat
## Kruskal-Wallis chi-squared = 86.807, df = 1, p-value < 2.2e-16

epsilonSquared(x = dt$Fairness.Perception.Lik,
               g = dt$Ability.Agg.Cat)

## epsilon.squared
##           0.391
```

e) Visualize.

e) Effect sizes (Eta-Square and Cohen's F).

17

```
# Effect Size ANOVA (Generalized Eta-Sq and Cohen's F-Value)
eff = anova_test(data = dt, Fairness.Perception.Lik ~ Ability.Agg.Cat, effect.size = "ges")
# Eta-Square Value.
eff

## ANOVA Table (type II tests)
##
##          Effect DFn DFd       F        p p<.05    ges
## 1 Ability.Agg.Cat   1 221 124.018 3.73e-23     * 0.359

# Now computing Cohen's f using eta-sq.
f = sqrt(eff$ges/(1-eff$ges))
# Cohen's F
f

## [1] 0.7483732

# Kruskal-Wallis Eta-Sq
ordinalEtaSquared(x = dt$Fairness.Perception.Lik,
                  g = dt$Ability.Agg.Cat)

## eta.squared
##       0.388
```

4.7   H$_{2b}$: Benevolence -> Fairness (***)

a) Shapiro test for normality.

```
# Shapiro test.
shapiro.test(dt$Fairness.Perception.Lik)

##
##  Shapiro-Wilk normality test
##
## data:  dt$Fairness.Perception.Lik
## W = 0.90486, p-value = 1.03e-10
```

The results show that since the `p<.05` the data is `not normal`.

b) Create a categorical variable.

```
# Median Cut.
dt$Benevolence.Agg.Cat = ifelse(dt$Benevolence.Agg.Dbl >= median(dt$Benevolence.Agg.Dbl),
                          "High", "Low")
```

c) Bartlett test for homoscedasticity.

```
# Bartlett test.
bartlett.test(dt$Fairness.Perception.Lik ~ dt$Benevolence.Agg.Cat)

##
##  Bartlett test of homogeneity of variances
##
## data:  dt$Fairness.Perception.Lik by dt$Benevolence.Agg.Cat
## Bartlett's K-squared = 14.122, df = 1, p-value = 0.0001713
```

The results show that since the `p<.05` the variance is `not homogeneous`.

d) ANOVA or Kurskal-Wallis.

```
# t-test
t.test(dt$Fairness.Perception.Lik ~ dt$Benevolence.Agg.Cat)
```

```
## 
##  Welch Two Sample t-test
## 
## data:  dt$Fairness.Perception.Lik by dt$Benevolence.Agg.Cat
## t = 6.3919, df = 185.34, p-value = 1.293e-09
## alternative hypothesis: true difference in means between group High and group Low is not equal to 0
## 95 percent confidence interval:
##  0.8851199 1.6754349
## sample estimates:
## mean in group High  mean in group Low
##         1.29914530         0.01886792

# Mann-Whitney U
wilcox.test(dt$Fairness.Perception.Lik ~ dt$Benevolence.Agg.Cat)

## 
##  Wilcoxon rank sum test with continuity correction
## 
## data:  dt$Fairness.Perception.Lik by dt$Benevolence.Agg.Cat
## W = 8885, p-value = 1.079e-08
## alternative hypothesis: true location shift is not equal to 0

# ANOVA
oneway.test(dt$Fairness.Perception.Lik ~ dt$Benevolence.Agg.Cat)

## 
##  One-way analysis of means (not assuming equal variances)
## 
## data:  dt$Fairness.Perception.Lik and dt$Benevolence.Agg.Cat
## F = 40.856, num df = 1.00, denom df = 185.34, p-value = 1.293e-09

# Kruskal-Wallis
kruskal.test(dt$Fairness.Perception.Lik ~ dt$Benevolence.Agg.Cat)

## 
##  Kruskal-Wallis rank sum test
## 
## data:  dt$Fairness.Perception.Lik by dt$Benevolence.Agg.Cat
## Kruskal-Wallis chi-squared = 32.705, df = 1, p-value = 1.073e-08

epsilonSquared(x = dt$Fairness.Perception.Lik,
               g = dt$Benevolence.Agg.Cat)

## epsilon.squared
##           0.147
```

e) Visualize.

f) Effect sizes (Eta-Square and Cohen's F).

```
# Effect Size ANOVA (Generalized Eta-Sq and Cohen's F-Value)
eff = anova_test(data = dt, Fairness.Perception.Lik ~ Benevolence.Agg.Cat, effect.size = "ges")
# Eta-Square Value.
eff

## ANOVA Table (type II tests)
##
##                Effect DFn DFd     F        p p<.05   ges
## 1 Benevolence.Agg.Cat   1 221 42.28 5.2e-10     * 0.161

# Now computing Cohen's f using eta-sq.
f = sqrt(eff$ges/(1-eff$ges))
# Cohen's F
f

## [1] 0.4380583

# Kruskal-Wallis Eta-Sq
ordinalEtaSquared(x = dt$Fairness.Perception.Lik,
                  g = dt$Benevolence.Agg.Cat)

## eta.squared
##       0.143
```

4.8   H$_{2c}$: Integrity -> Fairness (***)

a) Shapiro test for normality.

```
# Shapiro test.
shapiro.test(dt$Fairness.Perception.Lik)

##
##  Shapiro-Wilk normality test
##
## data:  dt$Fairness.Perception.Lik
## W = 0.90486, p-value = 1.03e-10
```

The results show that since the `p<.05` the data is `not normal`.

b) Create a categorical variable.

```
# Median Cut.
dt$Integrity.Agg.Cat = ifelse(dt$Integrity.Agg.Dbl >= median(dt$Integrity.Agg.Dbl),
                       "High", "Low")
```

c) Bartlett test for homoscedasticity.

```
# Bartlett test.
bartlett.test(dt$Fairness.Perception.Lik ~ dt$Integrity.Agg.Cat)

##
##  Bartlett test of homogeneity of variances
##
## data:  dt$Fairness.Perception.Lik by dt$Integrity.Agg.Cat
## Bartlett's K-squared = 17.13, df = 1, p-value = 3.49e-05
```

The results show that since the `p<.05` the variance is `not homogeneous`.

d) ANOVA or Kurskal-Wallis.

```
# t-test
t.test(dt$Fairness.Perception.Lik ~ dt$Integrity.Agg.Cat)
```

```
##
##  Welch Two Sample t-test
##
## data:  dt$Fairness.Perception.Lik by dt$Integrity.Agg.Cat
## t = 13.666, df = 164.5, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group High and group Low is not equal to 0
## 95 percent confidence interval:
##  1.901720 2.544027
## sample estimates:
## mean in group High  mean in group Low
##           1.6774194         -0.5454545

# Mann-Whitney U
wilcox.test(dt$Fairness.Perception.Lik ~ dt$Integrity.Agg.Cat)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  dt$Fairness.Perception.Lik by dt$Integrity.Agg.Cat
## W = 11144, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0

# ANOVA
oneway.test(dt$Fairness.Perception.Lik ~ dt$Integrity.Agg.Cat)

##
##  One-way analysis of means (not assuming equal variances)
##
## data:  dt$Fairness.Perception.Lik and dt$Integrity.Agg.Cat
## F = 186.77, num df = 1.0, denom df = 164.5, p-value < 2.2e-16

# Kruskal-Wallis
kruskal.test(dt$Fairness.Perception.Lik ~ dt$Integrity.Agg.Cat)

##
##  Kruskal-Wallis rank sum test
##
## data:  dt$Fairness.Perception.Lik by dt$Integrity.Agg.Cat
## Kruskal-Wallis chi-squared = 114.94, df = 1, p-value < 2.2e-16

epsilonSquared(x = dt$Fairness.Perception.Lik,
               g = dt$Integrity.Agg.Cat)

## epsilon.squared
##           0.518
```

e) Visualize.

f) Effect sizes (Eta-Square and Cohen's F).

```r
# Effect Size ANOVA (Generalized Eta-Sq and Cohen's F-Value)
eff = anova_test(data = dt, Fairness.Perception.Lik ~ Integrity.Agg.Cat, effect.size = "ges")
# Eta-Square Value.
eff

## ANOVA Table (type II tests)
##
##              Effect DFn DFd       F        p p<.05   ges
## 1 Integrity.Agg.Cat   1 221 203.333 3.81e-33     * 0.479

# Now computing Cohen's f using eta-sq.
f = sqrt(eff$ges/(1-eff$ges))
# Cohen's F
f

## [1] 0.9588461

# Kruskal-Wallis Eta-Sq
ordinalEtaSquared(x = dt$Fairness.Perception.Lik,
                  g = dt$Integrity.Agg.Cat)

## eta.squared
##       0.516
```

5 Linear Models

5.1 Profile + Model + Data -> Integrity (*)

5.1.1 Parametric LR
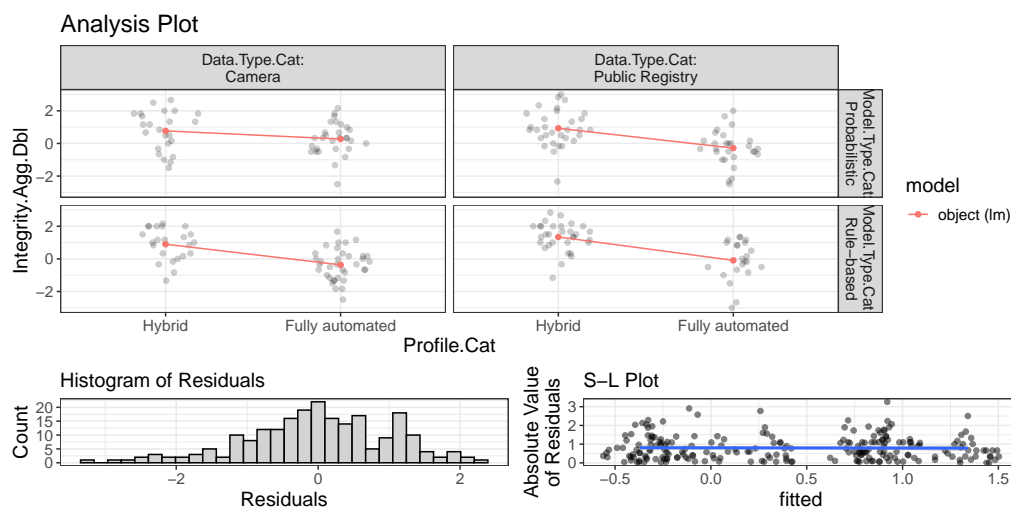
a) Fit the model.

```r
# Linear Model.
m = lm(Integrity.Agg.Dbl ~ Profile.Cat * Model.Type.Cat * Data.Type.Cat,
       data = dt)
# Summary of model.
summary(m)

##
## Call:
## lm(formula = Integrity.Agg.Dbl ~ Profile.Cat * Model.Type.Cat *
##     Data.Type.Cat, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2634 -0.6034  0.0333  0.6613  2.2857
##
## Coefficients:
##                                                                     Estimate
## (Intercept)                                                           0.2701
## Profile.CatHybrid                                                     0.5007
## Model.Type.CatRule-based                                             -0.6368
## Data.Type.CatPublic Registry                                        -0.5558
## Profile.CatHybrid:Model.Type.CatRule-based                           0.7645
## Profile.CatHybrid:Data.Type.CatPublic Registry                       0.7151
## Model.Type.CatRule-based:Data.Type.CatPublic Registry                0.8316
## Profile.CatHybrid:Model.Type.CatRule-based:Data.Type.CatPublic Registry  -0.5507
##                                                                    Std. Error
## (Intercept)                                                           0.1934
## Profile.CatHybrid                                                     0.2874
## Model.Type.CatRule-based                                             0.2615
## Data.Type.CatPublic Registry                                         0.2759
```
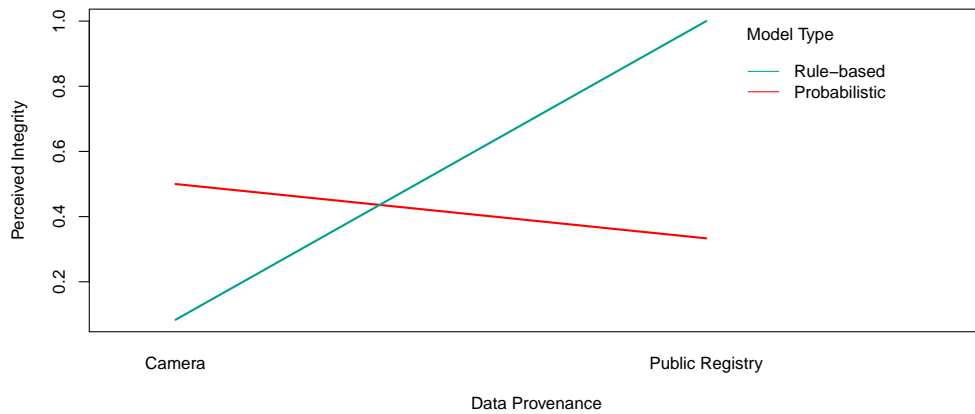
```
## Profile.CatHybrid:Model.Type.CatRule-based                                      0.4009
## Profile.CatHybrid:Data.Type.CatPublic Registry                                  0.3954
## Model.Type.CatRule-based:Data.Type.CatPublic Registry                           0.3955
## Profile.CatHybrid:Model.Type.CatRule-based:Data.Type.CatPublic Registry         0.5646
##                                                                          t value Pr(>|t|)
## (Intercept)                                                                1.397   0.1639
## Profile.CatHybrid                                                          1.742   0.0829
## Model.Type.CatRule-based                                                  -2.435   0.0157
## Data.Type.CatPublic Registry                                             -2.014   0.0452
## Profile.CatHybrid:Model.Type.CatRule-based                                 1.907   0.0579
## Profile.CatHybrid:Data.Type.CatPublic Registry                            1.809   0.0719
## Model.Type.CatRule-based:Data.Type.CatPublic Registry                     2.103   0.0367
## Profile.CatHybrid:Model.Type.CatRule-based:Data.Type.CatPublic Registry  -0.975   0.3305
##
## (Intercept)
## Profile.CatHybrid                                                        .
## Model.Type.CatRule-based                                                 *
## Data.Type.CatPublic Registry                                             *
## Profile.CatHybrid:Model.Type.CatRule-based                               .
## Profile.CatHybrid:Data.Type.CatPublic Registry                          .
## Model.Type.CatRule-based:Data.Type.CatPublic Registry                    *
## Profile.CatHybrid:Model.Type.CatRule-based:Data.Type.CatPublic Registry
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.041 on 215 degrees of freedom
## Multiple R-squared:  0.2659,Adjusted R-squared:  0.242
## F-statistic: 11.12 on 7 and 215 DF,  p-value: 5.438e-12
```
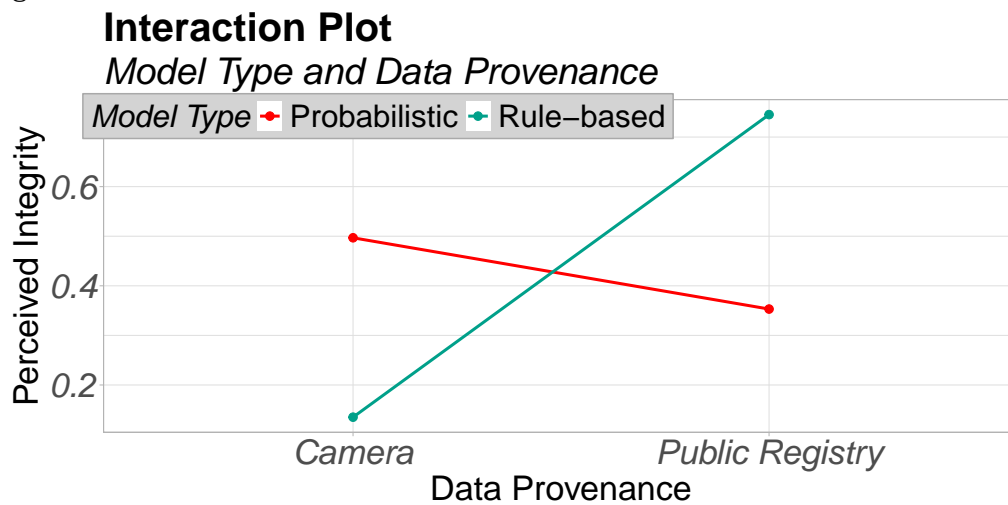
b) Visualize the model.

The results show that all `Model Type`, and `Data Type`, have `significant` direct effects. In addition, there is a `significant` interaction effect between `Model Type` and `Data Type`.

c) Visualize the interaction effects.

Generating Interaction Plot for Publication to ToCHI.

5.2   H$_{2a}$: Ability -> Fairness (***)
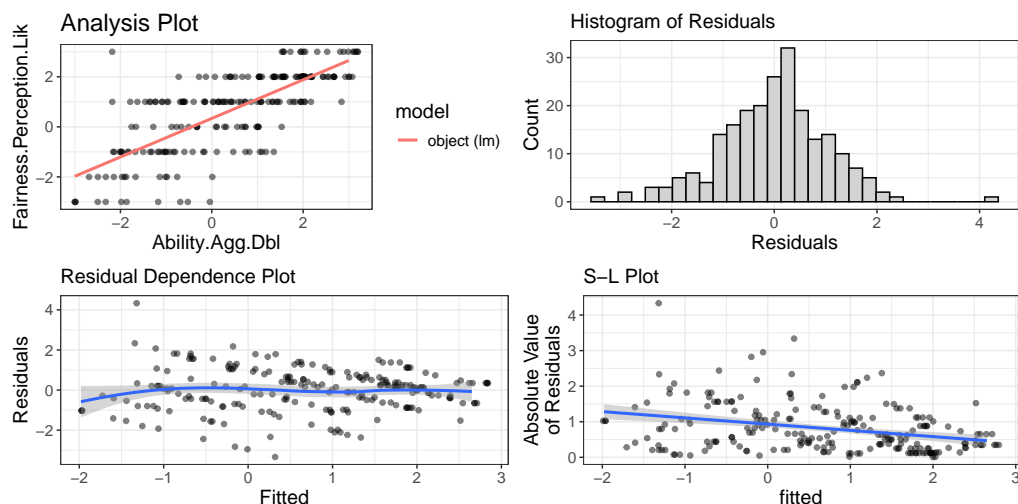
5.2.1   Parametric LR

a) Fit the model.

```
# Linear Model.
m = lm(Fairness.Perception.Lik ~ Ability.Agg.Dbl,
       data = dt)
# Summary of model.
summary(m)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Ability.Agg.Dbl, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3378 -0.6515  0.1197  0.6339  4.3332
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.33783    0.07423   4.551  8.8e-06 ***
## Ability.Agg.Dbl  0.77122    0.04598  16.773  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.063 on 221 degrees of freedom
## Multiple R-squared:  0.5601,Adjusted R-squared:  0.5581
```

```
## F-statistic: 281.3 on 1 and 221 DF,  p-value: < 2.2e-16
```

b) Visualize the model.

5.2.2 Non-Parametric LR

a) Fit the model.

```
# Kendall-Theil Sen Siegel Nonparametric Linear Regression.
m = mblm(Fairness.Perception.Lik ~ Ability.Agg.Dbl,
        data = dt)
# Summary of MBLM model.
summary.mblm(m)

##
## Call:
## mblm(formula = Fairness.Perception.Lik ~ Ability.Agg.Dbl, dataframe = dt)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.500 -0.750  0.000  0.500  4.125
##
## Coefficients:
##                 Estimate    MAD V value Pr(>|V|)
## (Intercept)       0.5000 0.5998   19184 2.83e-14 ***
## Ability.Agg.Dbl   0.7500 0.2926   21887  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.075 on 221 degrees of freedom

# Summary of LM model (for reference).
summary.lm(m)

##
## Call:
## mblm(formula = Fairness.Perception.Lik ~ Ability.Agg.Dbl, dataframe = dt)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.500 -0.750  0.000  0.500  4.125
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.50000    0.07503   6.664 2.09e-10 ***
## Ability.Agg.Dbl  0.75000    0.04648  16.137  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.075 on 221 degrees of freedom
## Multiple R-squared:  0.5409,Adjusted R-squared:  0.5389
## F-statistic: 260.4 on 1 and 221 DF,  p-value: < 2.2e-16

# Efron R2
efronRSquared(m)

## EfronRSquared
##         0.551
```

5.3 H$_{2b}$: Benevolence -> Fairness (***)
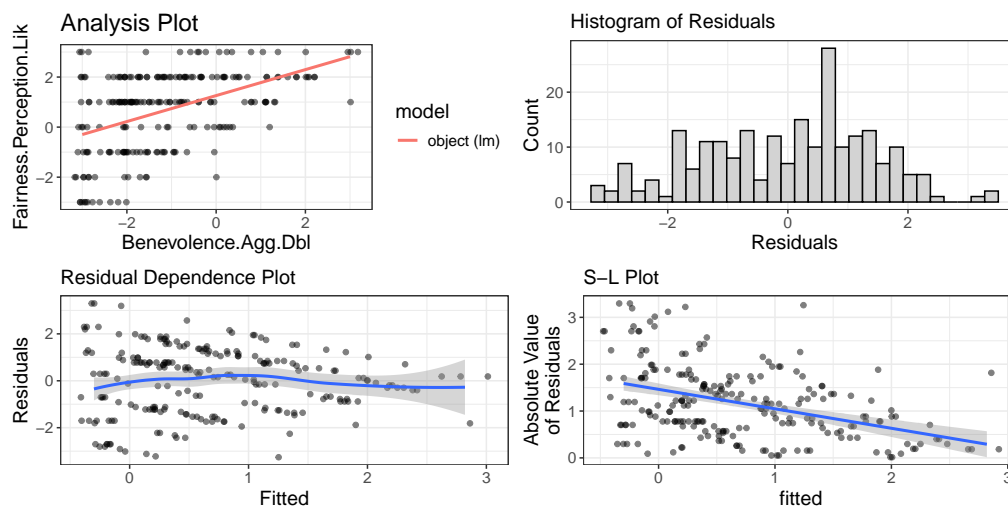
5.3.1 Parametric LR

a) Fit the model.

```
# Linear Model.
m = lm(Fairness.Perception.Lik ~ Benevolence.Agg.Dbl,
       data = dt)
# Summary of model.
summary(m)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Benevolence.Agg.Dbl, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2595 -1.1371  0.1845  1.0330  3.2965
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          1.25950    0.11831  10.645  < 2e-16 ***
## Benevolence.Agg.Dbl  0.51868    0.06488   7.995 7.18e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.411 on 221 degrees of freedom
## Multiple R-squared:  0.2243,Adjusted R-squared:  0.2208
## F-statistic: 63.92 on 1 and 221 DF,  p-value: 7.181e-14
```

b) Visualize the model.

5.3.2 Non-Parametric LR

a) Fit the model.

```
# Kendall-Theil Sen Siegel Nonparametric Linear Regression.
m = mblm(Fairness.Perception.Lik ~ Benevolence.Agg.Dbl,
         data = dt)
# Summary of MBLM model.
summary.mblm(m)

##
## Call:
## mblm(formula = Fairness.Perception.Lik ~ Benevolence.Agg.Dbl,
##     dataframe = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7857 -1.5714 -0.0714  0.5714  2.5714
##
## Coefficients:
##                      Estimate    MAD V value Pr(>|V|)
## (Intercept)           1.5000 0.7413    21316   <2e-16 ***
## Benevolence.Agg.Dbl   0.3571 0.5295    12109   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.491 on 221 degrees of freedom

# Summary of LM model (for reference).
summary.lm(m)

##
## Call:
## mblm(formula = Fairness.Perception.Lik ~ Benevolence.Agg.Dbl,
##     dataframe = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7857 -1.5714 -0.0714  0.5714  2.5714
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.50000    0.12501   12.00  < 2e-16 ***
## Benevolence.Agg.Dbl   0.35714    0.06855    5.21 4.32e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.491 on 221 degrees of freedom
## Multiple R-squared:  0.1094,Adjusted R-squared:  0.1054
## F-statistic: 27.14 on 1 and 221 DF,  p-value: 4.322e-07

# Efron R2
efronRSquared(m)

## EfronRSquared
##        0.134
```

5.4 H$_{2c}$: Integrity -> Fairness (***)
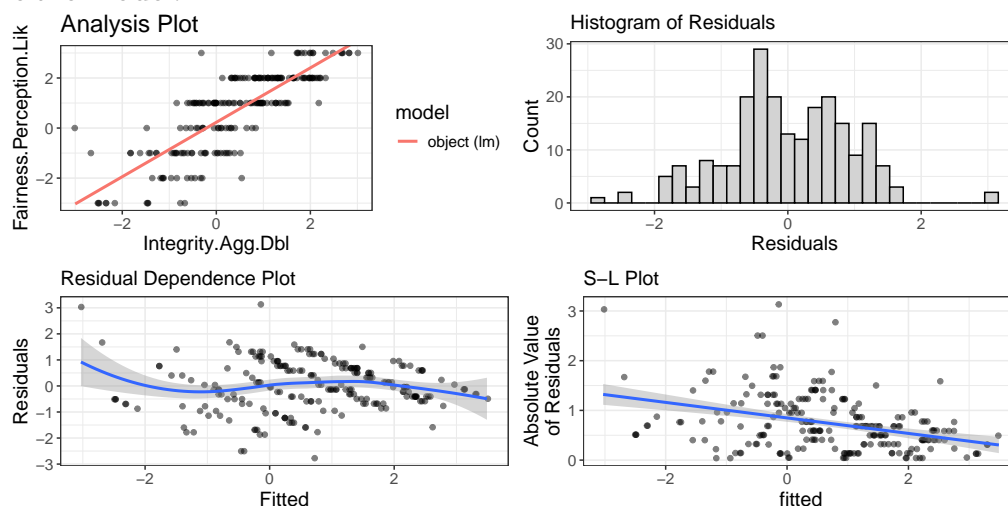
5.4.1 Parametric LR

a) Fit the model.

```
# Linear Model.
m = lm(Fairness.Perception.Lik ~ Integrity.Agg.Dbl,
       data = dt)
# Summary of model.
summary(m)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Integrity.Agg.Dbl, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7735 -0.5858 -0.0421  0.6828  3.1327
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.22978    0.06624   3.469 0.000629 ***
## Integrity.Agg.Dbl  1.08739    0.05231  20.789  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9322 on 221 degrees of freedom
## Multiple R-squared:  0.6617,Adjusted R-squared:  0.6601
## F-statistic: 432.2 on 1 and 221 DF,  p-value: < 2.2e-16
```

b) Visualize the model.

5.4.2 Non-Parametric LR

a) Fit the model.

```
# Kendall-Theil Sen Siegel Nonparametric Linear Regression.
m = mblm(Fairness.Perception.Lik ~ Integrity.Agg.Dbl,
         data = dt)
# Summary of MBLM model.
summary.mblm(m)

##
## Call:
## mblm(formula = Fairness.Perception.Lik ~ Integrity.Agg.Dbl, dataframe = dt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.87719 -0.71053 -0.04386  0.62281  2.95614
##
## Coefficients:
##                   Estimate    MAD V value Pr(>|V|)
```

```
## (Intercept)              0.3772 0.6763   17651 3.69e-11 ***
## Integrity.Agg.Dbl     1.0000 0.3094   24500  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9446 on 221 degrees of freedom

# Summary of LM model (for reference).
summary.lm(m)

##
## Call:
## mblm(formula = Fairness.Perception.Lik ~ Integrity.Agg.Dbl, dataframe = dt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.87719 -0.71053 -0.04386  0.62281  2.95614
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.37719    0.06713   5.619 5.74e-08 ***
## Integrity.Agg.Dbl 1.00000    0.05300  18.867  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9446 on 221 degrees of freedom
## Multiple R-squared:  0.617,Adjusted R-squared:  0.6152
## F-statistic:   356 on 1 and 221 DF,  p-value: < 2.2e-16

# Efron R2
efronRSquared(m)

## EfronRSquared
##        0.653
```

5.5   H$_2$: Ability + Benevolence + Integrity -> Fairness (***)
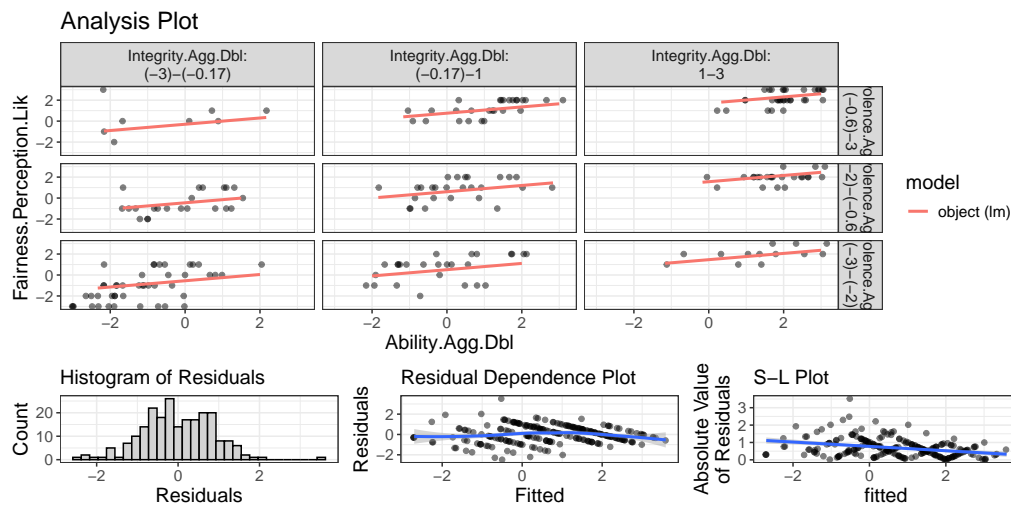
5.5.1   Parametric LR

a) Fit the model.

```
# Linear Model.
m = lm(Fairness.Perception.Lik ~ Ability.Agg.Dbl + Benevolence.Agg.Dbl + Integrity.Agg.Dbl,
      data = dt)
# Summary of model.
summary(m)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Ability.Agg.Dbl + Benevolence.Agg.Dbl +
##     Integrity.Agg.Dbl, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4802 -0.5950 -0.0252  0.5871  3.5185
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.31872    0.08814   3.616 0.000371 ***
## Ability.Agg.Dbl      0.29876    0.05912   5.053 9.15e-07 ***
## Benevolence.Agg.Dbl  0.07735    0.04675   1.655 0.099410 .
## Integrity.Agg.Dbl    0.75526    0.07372  10.245  < 2e-16 ***
## ---
```

29

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.867 on 219 degrees of freedom
## Multiple R-squared:   0.71,Adjusted R-squared:  0.706
## F-statistic: 178.7 on 3 and 219 DF,  p-value: < 2.2e-16
```

b) Visualize the model.

The results show that all `Ability` , and `Integrity` have `significant` direct effects. However, `Benevolence` has `no significant` direct effect.

5.5.2 Non-Parametric LR

a) Fit the model.

```
# Quantile Regression
# Our model.
m = rq(Fairness.Perception.Lik ~ Ability.Agg.Dbl + Benevolence.Agg.Dbl + Integrity.Agg.Dbl,
       data = dt,
       tau = 0.5)
# Null model.
m0 = rq(Fairness.Perception.Lik ~ 1,
       data = dt,
       tau = 0.5)
# Sumamary of the model.
summary(m, se = "ker")

##
## Call: rq(formula = Fairness.Perception.Lik ~ Ability.Agg.Dbl + Benevolence.Agg.Dbl +
##     Integrity.Agg.Dbl, tau = 0.5, data = dt)
##
## tau: [1] 0.5
##
## Coefficients:
##                     Value   Std. Error t value Pr(>|t|)
## (Intercept)         0.26215 0.13000    2.01662 0.04496
## Ability.Agg.Dbl     0.42037 0.10184    4.12790 0.00005
## Benevolence.Agg.Dbl 0.07239 0.06796    1.06522 0.28795
## Integrity.Agg.Dbl   0.62978 0.12443    5.06132 0.00000

# p-value for the overall model.
anova(m, m0)

## Quantile Regression Analysis of Deviance Table
##
## Model 1: Fairness.Perception.Lik ~ Ability.Agg.Dbl + Benevolence.Agg.Dbl + Integrity.Agg.Dbl
## Model 2: Fairness.Perception.Lik ~ 1
##   Df Resid Df F value     Pr(>F)
```

```
## 1  3      219  93.356 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# R2 values for the regression.
nagelkerke(m)

## $Models
##
## Model: "rq, Fairness.Perception.Lik ~ Ability.Agg.Dbl + Benevolence.Agg.Dbl + Integrity.Agg.Dbl, 0.5, d
## Null:  "rq, Fairness.Perception.Lik ~ 1, 0.5, dt"
##
## $Pseudo.R.squared.for.model.vs.null
##                               Pseudo.R.squared
## McFadden                             0.310615
## Cox and Snell (ML)                   0.693348
## Nagelkerke (Cragg and Uhler)         0.709125
##
## $Likelihood.ratio.test
##  Df.diff LogLik.diff Chisq   p.value
##      -3      -131.8 263.6 7.499e-57
##
## $Number.of.observations
##
## Model: 223
## Null:  223
##
## $Messages
## [1] "Note: For models fit with REML, these statistics are based on refitting with ML"
##
## $Warnings
## [1] "None"

# Gaussian Additive Model.
#m = gam(Fairness.Perception.Lik ~ Ability.Agg.Dbl + Benevolence.Agg.Dbl + Integrity.Agg.Dbl,
#        data = dt,
#        family = gaussian())
#m = rq(Fairness.Perception.Lik ~ Ability.Agg.Dbl + Benevolence.Agg.Dbl + Integrity.Agg.Dbl,
#        data = dt,
#        tau = 0.5)
#m0 = rq(Fairness.Perception.Lik ~ 1,
#        data = dt,
#        tau = 0.5)
```

The results show that all `Ability`, and `Integrity` have `significant` direct effects. However, `Benevolence` has `no significant` direct effect.

5.6 Policy Agreement -> Integrity (**)

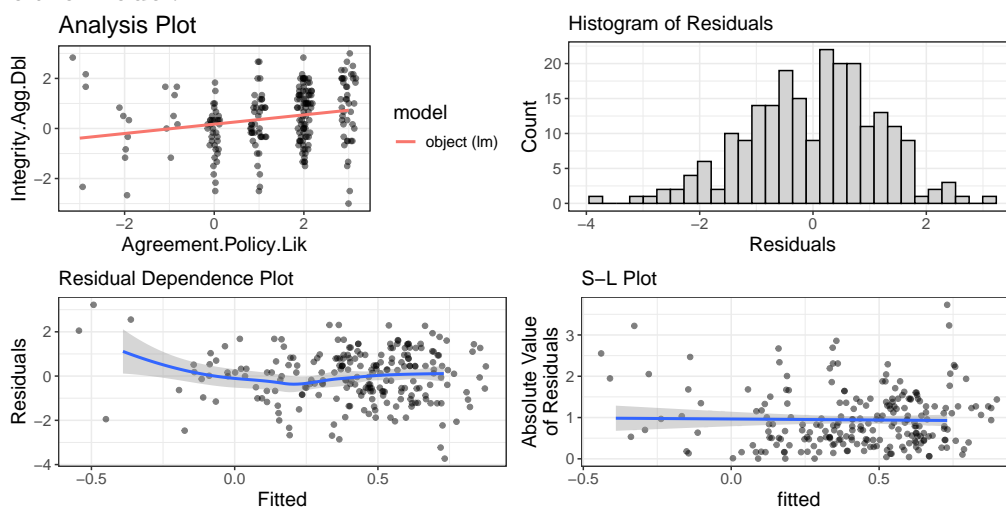### 5.6.1 Parametric LR

a) Fit the model.

```
# Linear Model.
m = lm(Integrity.Agg.Dbl ~ Agreement.Policy.Lik,
       data = dt)
# Summary of model.
summary(m)

##
## Call:
## lm(formula = Integrity.Agg.Dbl ~ Agreement.Policy.Lik, data = dt)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7299 -0.7105  0.1421  0.8088  3.2194
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.1719     0.1105   1.555  0.12138
## Agreement.Policy.Lik  0.1860     0.0575   3.234  0.00141 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.171 on 221 degrees of freedom
## Multiple R-squared:  0.0452,Adjusted R-squared:  0.04088
## F-statistic: 10.46 on 1 and 221 DF,  p-value: 0.001406
```

b) Visualize the model.

5.6.2 Non-Parametric LR

a) Fit the model.

```
# Kendall-Theil Sen Siegel Nonparametric Linear Regression.
m = mblm(Integrity.Agg.Dbl ~ Agreement.Policy.Lik,
         data = dt)
# Summary of MBLM model.
summary.mblm(m)

##
## Call:
## mblm(formula = Integrity.Agg.Dbl ~ Agreement.Policy.Lik, dataframe = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1667 -1.0000  0.0000  0.6667  3.6667
##
## Coefficients:
##                     Estimate    MAD V value Pr(>|V|)
## (Intercept)           0.1667 0.4942   13744   0.0925 .
## Agreement.Policy.Lik  0.3333 0.3912   18402 7.21e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.205 on 221 degrees of freedom

# Summary of LM model (for reference).
summary.lm(m)
```

```
##
## Call:
## mblm(formula = Integrity.Agg.Dbl ~ Agreement.Policy.Lik, dataframe = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1667 -1.0000  0.0000  0.6667  3.6667
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.16667    0.11367   1.466    0.144
## Agreement.Policy.Lik  0.33333    0.05913   5.637 5.25e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.205 on 221 degrees of freedom
## Multiple R-squared:  0.1257,Adjusted R-squared:  0.1217
## F-statistic: 31.77 on 1 and 221 DF,  p-value: 5.251e-08

# Efron R2
efronRSquared(m)

## EfronRSquared
##      -0.00968
```

Now using Quantile Regression.

```
# Quantile Regression
# Our model.
m = rq(Integrity.Agg.Dbl ~ Agreement.Policy.Lik,
       data = dt,
       tau = 0.5)
# Null model.
m0 = rq(Integrity.Agg.Dbl ~ 1,
       data = dt,
       tau = 0.5)
# Sumamary of the model.
summary(m, se = "ker")

##
## Call: rq(formula = Integrity.Agg.Dbl ~ Agreement.Policy.Lik, tau = 0.5,
##     data = dt)
##
## tau: [1] 0.5
##
## Coefficients:
##                      Value   Std. Error t value Pr(>|t|)
## (Intercept)          0.25000 0.20037    1.24768 0.21347
## Agreement.Policy.Lik 0.25000 0.11231    2.22598 0.02703

# p-value for the overall model.
anova(m, m0)

## Quantile Regression Analysis of Deviance Table
##
## Model 1: Integrity.Agg.Dbl ~ Agreement.Policy.Lik
## Model 2: Integrity.Agg.Dbl ~ 1
##   Df Resid Df F value     Pr(>F)
## 1  1      221  20.067 1.199e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# R2 values for the regression.
nagelkerke(m)
```

```
## $Models
##
## Model: "rq, Integrity.Agg.Dbl ~ Agreement.Policy.Lik, 0.5, dt"
## Null:  "rq, Integrity.Agg.Dbl ~ 1, 0.5, dt"
##
## $Pseudo.R.squared.for.model.vs.null
##                              Pseudo.R.squared
## McFadden                           0.0209536
## Cox and Snell (ML)                 0.0673407
## Nagelkerke (Cragg and Uhler)       0.0698479
##
## $Likelihood.ratio.test
##  Df.diff LogLik.diff  Chisq  p.value
##       -1     -7.7733 15.547 8.05e-05
##
## $Number.of.observations
##
## Model: 223
## Null:  223
##
## $Messages
## [1] "Note: For models fit with REML, these statistics are based on refitting with ML"
##
## $Warnings
## [1] "None"
```

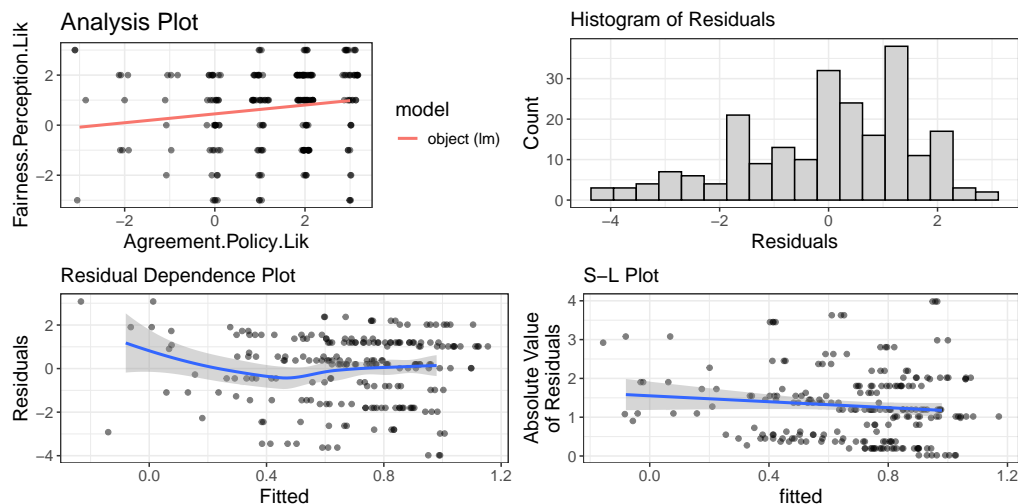## 5.7   Policy Agreement -> Fairness (*)

5.7.1   Parametric LR

a) Fit the model.

```
# Linear Model.
m = lm(Fairness.Perception.Lik ~ Agreement.Policy.Lik,
       data = dt)
# Summary of model.
summary(m)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Agreement.Policy.Lik,
##     data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9818 -1.0970  0.1951  1.1951  3.0800
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.45092    0.14948   3.017  0.00286 **
## Agreement.Policy.Lik  0.17697    0.07776   2.276  0.02382 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.584 on 221 degrees of freedom
## Multiple R-squared:  0.0229,Adjusted R-squared:  0.01848
## F-statistic: 5.179 on 1 and 221 DF,  p-value: 0.02382
```

b) Visualize the model.

5.7.2 Non-Parametric LR

a) Fit the model.

```
# Kendall-Theil Sen Siegel Nonparametric Linear Regression.
m = mblm(Fairness.Perception.Lik ~ Agreement.Policy.Lik,
         data = dt)
# Summary of MBLM model.
summary.mblm(m)

##
## Call:
## mblm(formula = Fairness.Perception.Lik ~ Agreement.Policy.Lik,
##     dataframe = dt)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##     -4     -2      0      1      2
##
## Coefficients:
##                     Estimate    MAD V value Pr(>|V|)
## (Intercept)           1.0000 1.1861   12015 1.51e-10 ***
## Agreement.Policy.Lik  0.0000 0.7413    8274 2.65e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.633 on 221 degrees of freedom

# Summary of LM model (for reference).
summary.lm(m)

##
## Call:
## mblm(formula = Fairness.Perception.Lik ~ Agreement.Policy.Lik,
##     dataframe = dt)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##     -4     -2      0      1      2
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          1.00000    0.15404   6.492 5.52e-10 ***
## Agreement.Policy.Lik 0.00000    0.08014   0.000        1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

35

```
##
## Residual standard error: 1.633 on 221 degrees of freedom
## Multiple R-squared:      0,Adjusted R-squared:  -0.004525
## F-statistic:      0 on 1 and 221 DF,  p-value: 1

# Efron R2
efronRSquared(m)

## EfronRSquared
##       -0.0376
```

Since the estimate value acquired through Kendall-Thiel method is shown as Zero, we will also conduct a Quantile Regression below.

```
# Quantile Regression
# Our model.
m = rq(Fairness.Perception.Lik ~ Agreement.Policy.Lik,
       data = dt,
       tau = 0.5)

## Warning in rq.fit.br(x, y, tau = tau, ...):  Solution may be nonunique

# Null model.
m0 = rq(Fairness.Perception.Lik ~ 1,
        data = dt,
        tau = 0.5)
# Sumamary of the model.
summary(m, se = "ker")

##
## Call: rq(formula = Fairness.Perception.Lik ~ Agreement.Policy.Lik,
##     tau = 0.5, data = dt)
##
## tau: [1] 0.5
##
## Coefficients:
##                      Value   Std. Error t value Pr(>|t|)
## (Intercept)          1.00000 0.26515    3.77152 0.00021
## Agreement.Policy.Lik 0.00000 0.14150    0.00000 1.00000

# p-value for the overall model.
anova(m, m0)

## Quantile Regression Analysis of Deviance Table
##
## Model 1: Fairness.Perception.Lik ~ Agreement.Policy.Lik
## Model 2: Fairness.Perception.Lik ~ 1
##   Df Resid Df F value Pr(>F)
## 1  1     221       0      1

# R2 values for the regression.
nagelkerke(m)

## $Models
##
## Model: "rq, Fairness.Perception.Lik ~ Agreement.Policy.Lik, 0.5, dt"
## Null:  "rq, Fairness.Perception.Lik ~ 1, 0.5, dt"
##
## $Pseudo.R.squared.for.model.vs.null
##                            Pseudo.R.squared
## McFadden                                  0
## Cox and Snell (ML)                        0
## Nagelkerke (Cragg and Uhler)              0
##
## $Likelihood.ratio.test
```

```
##  Df.diff LogLik.diff Chisq p.value
##      -1            0     0       1
##
## $Number.of.observations
##
## Model: 223
## Null:  223
##
## $Messages
## [1] "Note: For models fit with REML, these statistics are based on refitting with ML"
##
## $Warnings
## [1] "None"
```

6 Mediation Analysis

In this section, we will conduct the `mediation analysis`. Before we proceed with the analysis, let's briefly summarize the conditions for conducting mediation analysis, and whether there is a mediation effect.

6.0.1 Step 1: IV → DV

First, we must confirm that:

The `Independent Variable (IV)` is a **significant** predictor of `Dependent Variable (DV)`.

$$Y = \beta_{10} + \beta_{11}X + \epsilon_1$$

Here, β_{11} should be **significant**.

6.0.2 Step 2: IV → MV

Second, we must confirm that:

The `Independent Variable (IV)` is a **significant** predictor of the `Mediator Variable (MV)`. If this effect is not significant, then there is not association between the independent variable and the mediator, and thus, it cannot mediate anything.

$$M = \beta_{20} + \beta_{21}X + \epsilon_2$$

Here, β_{21} should be **significant**.

6.0.3 Step 3: IV + MV → DV

Third, we must confirm that:

a) The `Mediator Variable (MV)` is a **significant** predictor of the `Dependent Variable (DV)`.

b) The strength of the coefficient of the previously significant `Independent Variable (IV)` in **Step #1** is now **greatly reduced, if not rendered non-significant**.

$$Y = \beta_{30} + \beta_{31}X + \beta_{32}M + \epsilon_3$$

Here, β_{32} should be **significant** and $\beta_{31} \ll \beta_{11}$.
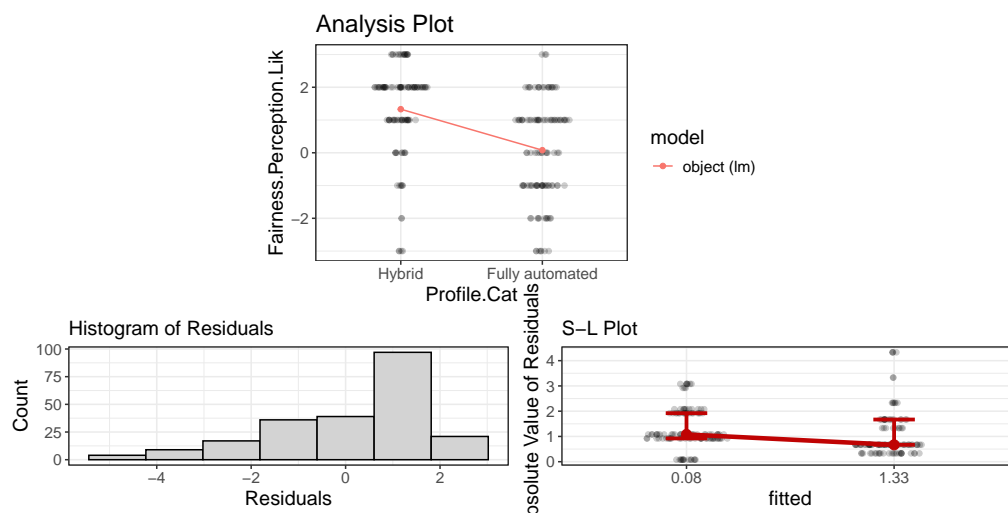
6.1 Profile, Ability, and Perceived Fairness (***)

`Profile (X)` $\Rightarrow$ `Ability (M)` $\Rightarrow$ `Perceived Fairness (Y)`

Step 1: Predicting the dependent variable with the independent variable.

```
# Step 1: Predict the DV.
base_model = lm(Fairness.Perception.Lik ~ Profile.Cat,
                data = dt)
summary(base_model)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Profile.Cat, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.3303 -1.0789  0.6697  0.9211  2.9211
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.07895    0.13809   0.572    0.568
## Profile.CatHybrid  1.25133    0.19751   6.335 1.31e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.474 on 221 degrees of freedom
## Multiple R-squared:  0.1537,Adjusted R-squared:  0.1499
## F-statistic: 40.14 on 1 and 221 DF,  p-value: 1.309e-09

visualize(base_model)
```


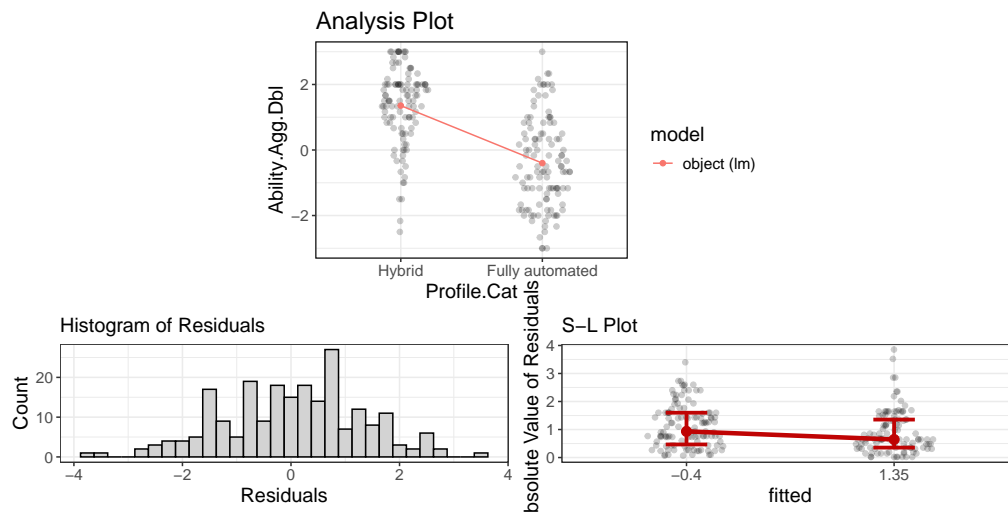The results show a `Significant` direct effect.
Step 2: Predicting the mediator with the independent variable.

```
# Step 2: Predict the Mediator.
mediate_model = lm(Ability.Agg.Dbl ~ Profile.Cat,
                data = dt)
summary(mediate_model)

##
## Call:
## lm(formula = Ability.Agg.Dbl ~ Profile.Cat, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8547 -0.7661  0.0673  0.7339  3.4006
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -0.4006     0.1200  -3.338  0.00099 ***
## Profile.CatHybrid   1.7553     0.1716  10.226  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.281 on 221 degrees of freedom
## Multiple R-squared:  0.3212,Adjusted R-squared:  0.3181
## F-statistic: 104.6 on 1 and 221 DF,  p-value: < 2.2e-16
```

```
visualize(mediate_model)
```


The results show a `Significant` effect of Independent Variable on the Mediator.
Step 3: The full model, including the mediator.

```
# Step 3: Full model.
full_model = lm(Fairness.Perception.Lik ~ Ability.Agg.Dbl + Profile.Cat,
                data = dt)
summary(full_model)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Ability.Agg.Dbl + Profile.Cat,
##     data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2481 -0.6460  0.1336  0.6099  4.3319
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.39895    0.10209   3.908 0.000124 ***
## Ability.Agg.Dbl    0.79883    0.05584  14.306  < 2e-16 ***
## Profile.CatHybrid -0.15088    0.17294  -0.872 0.383915
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.064 on 220 degrees of freedom
## Multiple R-squared:  0.5616,Adjusted R-squared:  0.5576
## F-statistic: 140.9 on 2 and 220 DF,  p-value: < 2.2e-16

# Mediation Analysis Results.
results = mediate(mediate_model, full_model,
                  treat = "Profile.Cat",
                  mediator = "Ability.Agg.Dbl",
                  boot = TRUE,
                  sims = 500)
summary(results)

##
## Causal Mediation Analysis
```
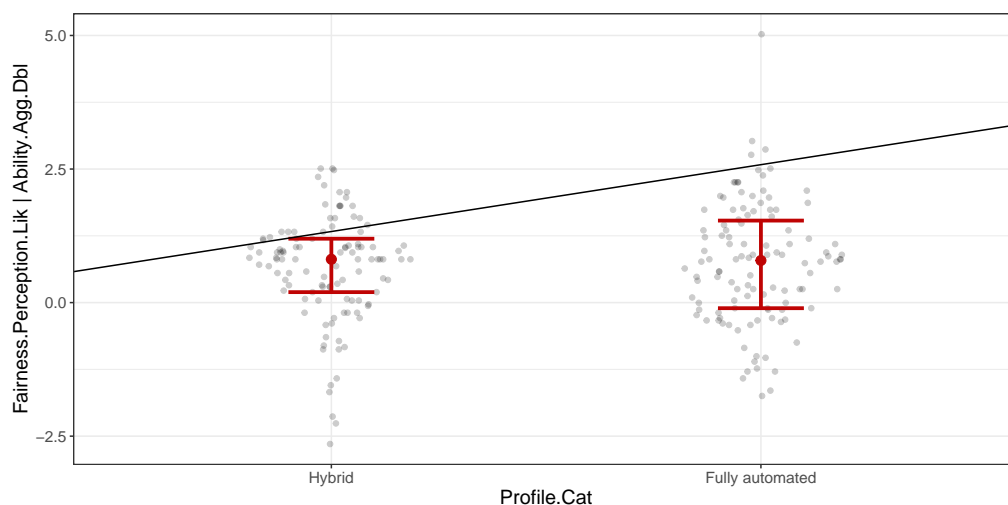
```
## 
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
## 
##                Estimate 95% CI Lower 95% CI Upper p-value
## ACME              1.402        1.053         1.72  <2e-16 ***
## ADE              -0.151       -0.484         0.23    0.42
## Total Effect      1.251        0.875         1.65  <2e-16 ***
## Prop. Mediated    1.121        0.847         1.48  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Sample Size Used: 223
## 
## 
## Simulations: 500

# Visualizing the Medeation Results.
mediate_plot(Fairness.Perception.Lik ~ Ability.Agg.Dbl + Profile.Cat, data = dt)
```


The `Total.Effects` shows the *slope* of the effect of Independent Variable (i.e., treatment) on the Dependent Variable (i.e., outcome). It shows that this effect is `significant`. Next, we have the `ADE (Average Direct Effect)`, which is the net effect of IV on DV when we substract the mediator variable. Here this slope is `not significant`. Finally, we look at the `ACME (Average Causal Mediation Effect)` (the most important aspect of this analysis), which is the total mediation effect or the strength of the mediation effect, i.e., how much does the slope change when a mediator variable is added, which in our case is statistically `significant`.

The results show a `mediation` effect.

6.2   Profile, Integrity, and Perceived Fairness (***)

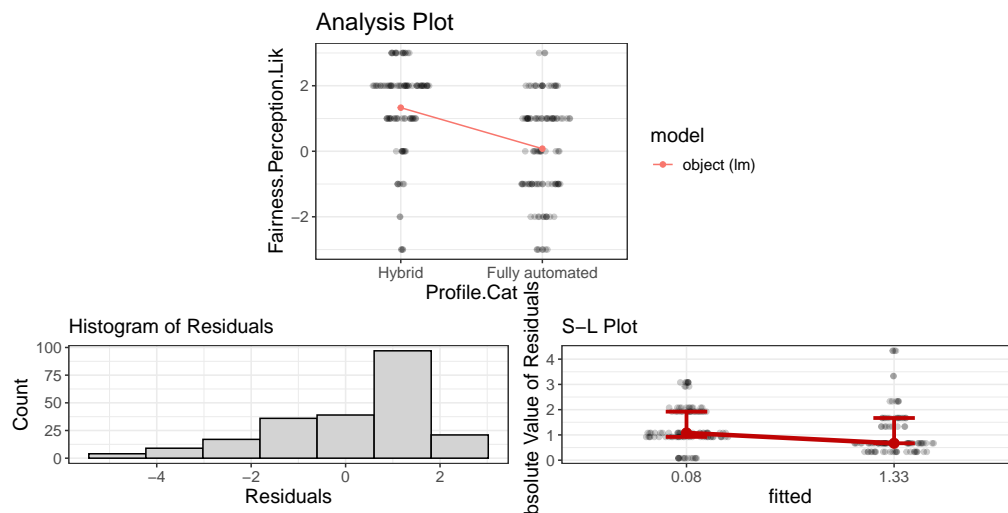`Profile (X)` ⇒ `Integrity (M)` ⇒ `Perceived Fairness (Y)`

Step 1: Predicting the dependent variable with the independent variable.

```
# Step 1: Predict the DV.
base_model = lm(Fairness.Perception.Lik ~ Profile.Cat,
                data = dt)
summary(base_model)

## 
## Call:
## lm(formula = Fairness.Perception.Lik ~ Profile.Cat, data = dt)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -4.3303 -1.0789  0.6697  0.9211  2.9211
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.07895    0.13809   0.572    0.568
## Profile.CatHybrid  1.25133    0.19751   6.335 1.31e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.474 on 221 degrees of freedom
## Multiple R-squared:  0.1537,Adjusted R-squared:  0.1499
## F-statistic: 40.14 on 1 and 221 DF,  p-value: 1.309e-09
```
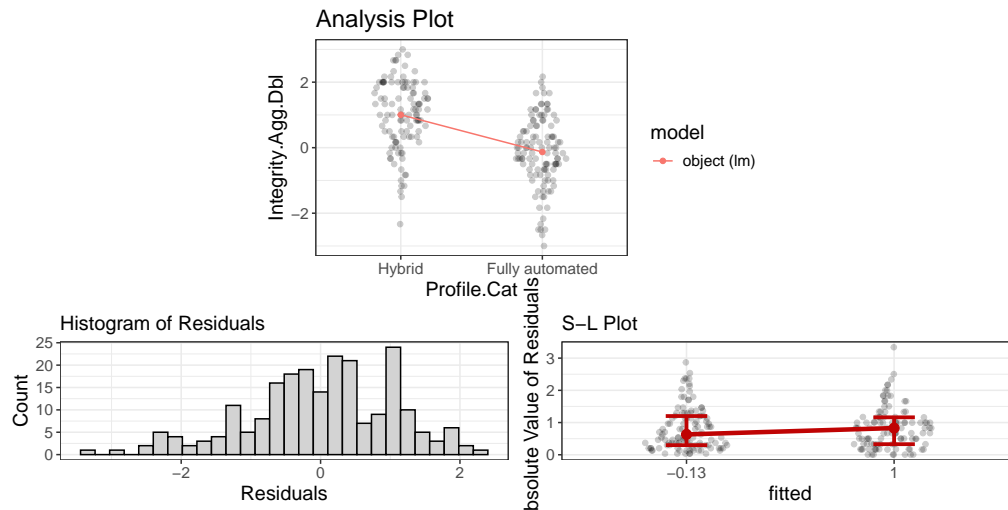
```
visualize(base_model)
```


The results show a `Significant` direct effect.

Step 2: Predicting the mediator with the independent variable.

```
# Step 2: Predict the Mediator.
mediate_model = lm(Integrity.Agg.Dbl ~ Profile.Cat,
                   data = dt)
summary(mediate_model)
```

```
##
## Call:
## lm(formula = Integrity.Agg.Dbl ~ Profile.Cat, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3379 -0.6032 -0.0046  0.8135  2.2982
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -0.13158    0.09876  -1.332    0.184
## Profile.CatHybrid  1.13617    0.14126   8.043 5.29e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.054 on 221 degrees of freedom
## Multiple R-squared:  0.2264,Adjusted R-squared:  0.2229
## F-statistic: 64.69 on 1 and 221 DF,  p-value: 5.292e-14
```

```
visualize(mediate_model)
```

Analysis Plot

Histogram of Residuals

S–L Plot

The results show a `Significant` effect of Independent Variable on the Mediator.
Step 3: The full model, including the mediator.

```
# Step 3: Full model.
full_model = lm(Fairness.Perception.Lik ~ Integrity.Agg.Dbl + Profile.Cat,
                data = dt)
summary(full_model)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Integrity.Agg.Dbl + Profile.Cat,
##     data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7631 -0.5859 -0.0475  0.6747  3.1396
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.22149    0.08786   2.521   0.0124 *
## Integrity.Agg.Dbl  1.08330    0.05960  18.175   <2e-16 ***
## Profile.CatHybrid  0.02052    0.14231   0.144   0.8855
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9343 on 220 degrees of freedom
## Multiple R-squared:  0.6617,Adjusted R-squared:  0.6586
## F-statistic: 215.1 on 2 and 220 DF,  p-value: < 2.2e-16

# Mediation Analysis Results.
results = mediate(mediate_model, full_model,
                  treat = "Profile.Cat",
                  mediator = "Integrity.Agg.Dbl",
                  boot = TRUE,
                  sims = 500)
summary(results)

##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                Estimate 95% CI Lower 95% CI Upper p-value
## ACME             1.2308       0.9053         1.52  <2e-16 ***
## ADE              0.0205      -0.2360         0.27    0.88
## Total Effect     1.2513       0.8739         1.63  <2e-16 ***
## Prop. Mediated   0.9836       0.8039         1.23  <2e-16 ***
```
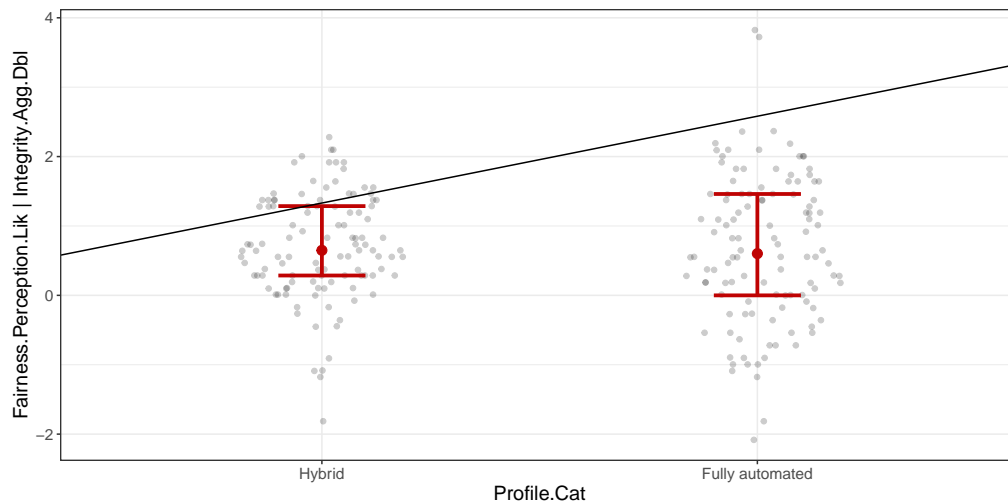
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 223
##
##
## Simulations: 500

# Visualizing the Medeation Results.
mediate_plot(Fairness.Perception.Lik ~ Integrity.Agg.Dbl + Profile.Cat, data = dt)
```


The `Total.Effects` shows the *slope* of the effect of Independent Variable (i.e., treatment) on the Dependent Variable (i.e., outcome). It shows that this effect is `significant`. Next, we have the `ADE (Average Direct Effect)`, which is the net effect of IV on DV when we substract the mediator variable. Here this slope is `not significant`. Finally, we look at the `ACME (Average Causal Mediation Effect)` (the most important aspect of this analysis), which is the total mediation effect or the strength of the mediation effect, i.e., how much does the slope change when a mediator variable is added, which in our case is statistically `significant`.

The results show a `mediation` effect.

6.3 Policy Agreement, Integrity, and Perceived Fairness (***)

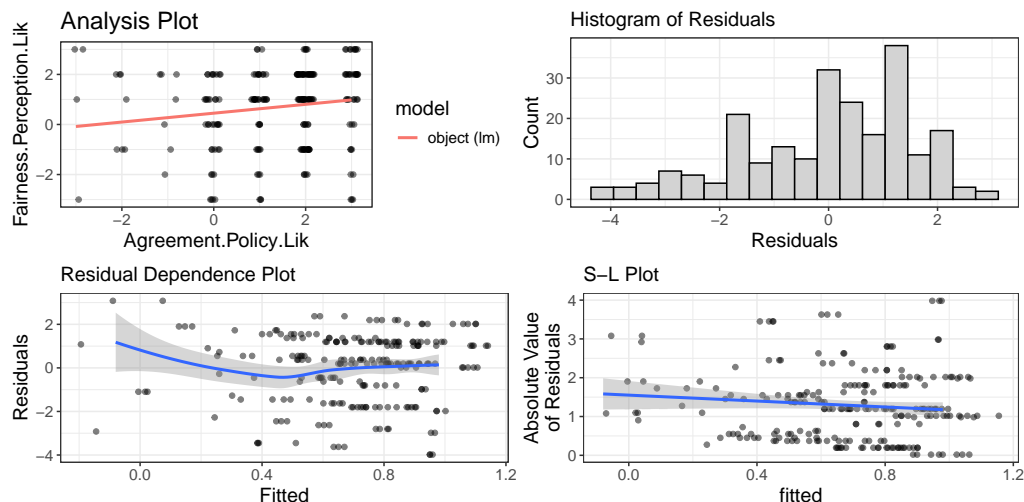`Policy Agreement (X)` ⇒ `Integrity (M)` ⇒ `Perceived Fairness (Y)`

Step 1: Predicting the dependent variable with the independent variable.

```
# Step 1: Predict the DV.
base_model = lm(Fairness.Perception.Lik ~ Agreement.Policy.Lik,
                data = dt)
summary(base_model)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Agreement.Policy.Lik,
##     data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9818 -1.0970  0.1951  1.1951  3.0800
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.45092    0.14948   3.017  0.00286 **
## Agreement.Policy.Lik   0.17697    0.07776   2.276  0.02382 *
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.584 on 221 degrees of freedom
## Multiple R-squared:  0.0229,Adjusted R-squared:  0.01848
## F-statistic: 5.179 on 1 and 221 DF,  p-value: 0.02382

visualize(base_model)
```
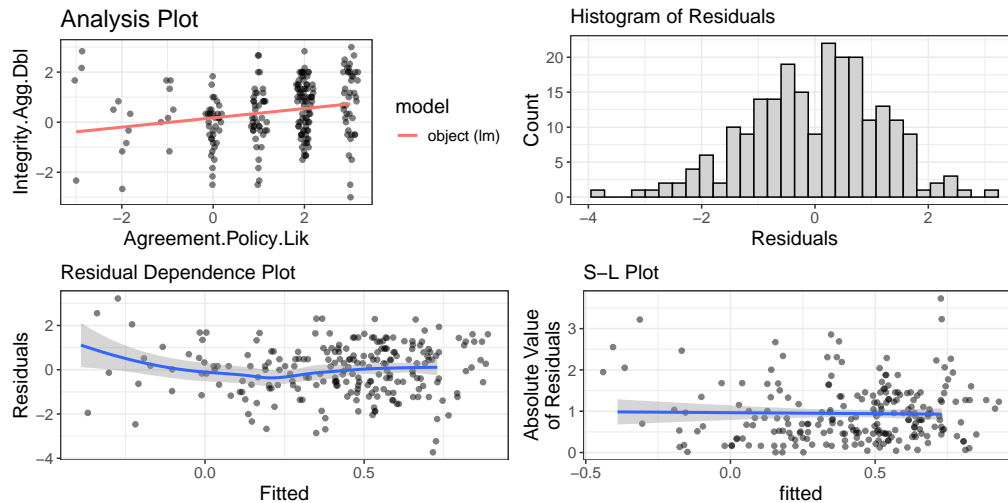

The results show a `Significant` direct effect.

Step 2: Predicting the mediator with the independent variable.

```
# Step 2: Predict the Mediator.
mediate_model = lm(Integrity.Agg.Dbl ~ Agreement.Policy.Lik,
                   data = dt)
summary(mediate_model)

##
## Call:
## lm(formula = Integrity.Agg.Dbl ~ Agreement.Policy.Lik, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7299 -0.7105  0.1421  0.8088  3.2194
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.1719     0.1105   1.555  0.12138
## Agreement.Policy.Lik   0.1860     0.0575   3.234  0.00141 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.171 on 221 degrees of freedom
## Multiple R-squared:  0.0452,Adjusted R-squared:  0.04088
## F-statistic: 10.46 on 1 and 221 DF,  p-value: 0.001406

visualize(mediate_model)
```

The results show a `Significant` effect of Independent Variable on the Mediator.
Step 3: The full model, including the mediator.

```r
# Step 3: Full model.
full_model = lm(Fairness.Perception.Lik ~ Integrity.Agg.Dbl + Agreement.Policy.Lik,
                data = dt)
summary(full_model)

##
## Call:
## lm(formula = Fairness.Perception.Lik ~ Integrity.Agg.Dbl + Agreement.Policy.Lik,
##     data = dt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.73040 -0.55343 -0.05413  0.69622  3.12817
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.26291    0.08858   2.968  0.00333 **
## Integrity.Agg.Dbl     1.09382    0.05361  20.402  < 2e-16 ***
## Agreement.Policy.Lik -0.02647    0.04690  -0.564  0.57304
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9337 on 220 degrees of freedom
## Multiple R-squared:  0.6621,Adjusted R-squared:  0.6591
## F-statistic: 215.6 on 2 and 220 DF,  p-value: < 2.2e-16

# Mediation Analysis Results.
results = mediate(mediate_model, full_model,
                  treat = "Agreement.Policy.Lik",
                  mediator = "Integrity.Agg.Dbl",
                  boot = TRUE,
                  sims = 500)
summary(results)

##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                Estimate 95% CI Lower 95% CI Upper p-value
## ACME            0.20344      0.04768         0.39   0.004 **
## ADE            -0.02647     -0.13783         0.08   0.568
## Total Effect    0.17697     -0.00136         0.37   0.060 .
## Prop. Mediated  1.14960     -0.09837         3.59   0.056 .
```
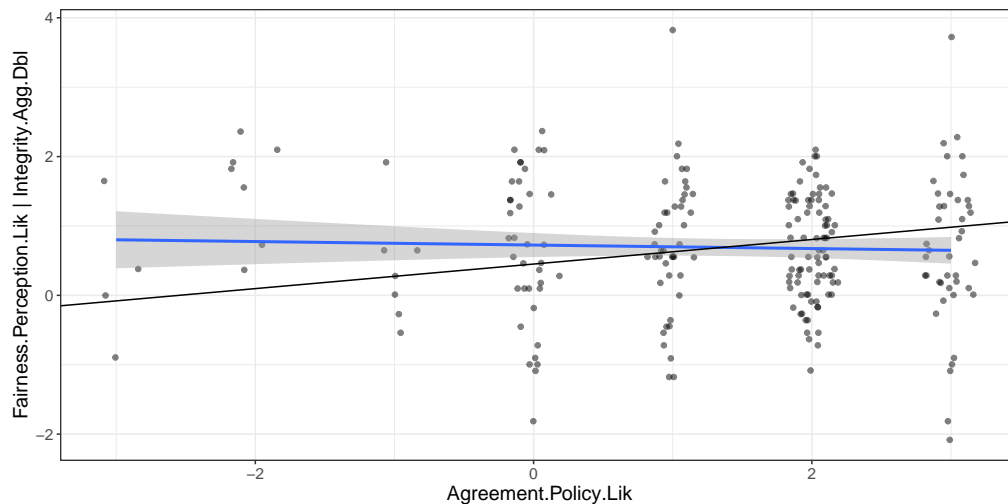
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 223
##
##
## Simulations: 500

# Visualizing the Medeation Results.
mediate_plot(Fairness.Perception.Lik ~ Integrity.Agg.Dbl + Agreement.Policy.Lik, data = dt)
```


The `Total.Effects` shows the *slope* of the effect of Independent Variable (i.e., treatment) on the Dependent Variable (i.e., outcome). It shows that this effect is `significant`. Next, we have the `ADE (Average Direct Effect)`, which is the net effect of IV on DV when we substract the mediator variable. Here this slope is `not significant`. Finally, we look at the `ACME (Average Causal Mediation Effect)` (the most important aspect of this analysis), which is the total mediation effect or the strength of the mediation effect, i.e., how much does the slope change when a mediator variable is added, which in our case is statistically `significant`.

The results show a `mediation` effect.

7   Principal Component Analysis

For this experiment, we will specifically do the `PCA` or Factor Analysis to check whether the different questions capturing `Integrity`, `Benevolence`, and `Ability` can be aggregated into a single variable. Therefore, we will conduct *three* PCAs.

7.1   Ability

... preparing the data for PCA. Including `Profile` as a qualitative supplementary variable (as it demonstrated the most discriminatory effect).

```
# Selecting the columns for PCA.
dt.pca = dt[, c(11, 14:20)]
```

... conducting the PCA.
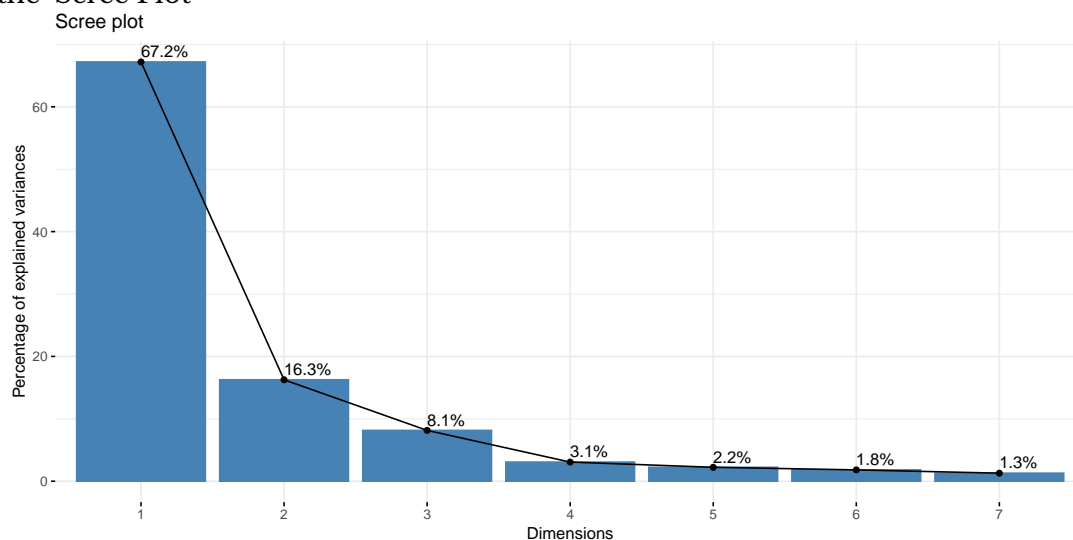
```
# Principal Component Analysis
res.pca = PCA(dt.pca,
              scale.unit = TRUE,
              quali.sup = c(1),
              ncp = 10,
              graph = FALSE)
```

... having a look at the eigen values.

```
# Eigen Values
get_eigenvalue(res.pca)

##       eigenvalue variance.percent cumulative.variance.percent
## Dim.1 4.70338279        67.191183                    67.19118
## Dim.2 1.13798406        16.256915                    83.44810
## Dim.3 0.57022601         8.146086                    91.59418
## Dim.4 0.21476686         3.068098                    94.66228
## Dim.5 0.15618957         2.231280                    96.89356
## Dim.6 0.12713084         1.816155                    98.70972
## Dim.7 0.09031988         1.290284                   100.00000
```
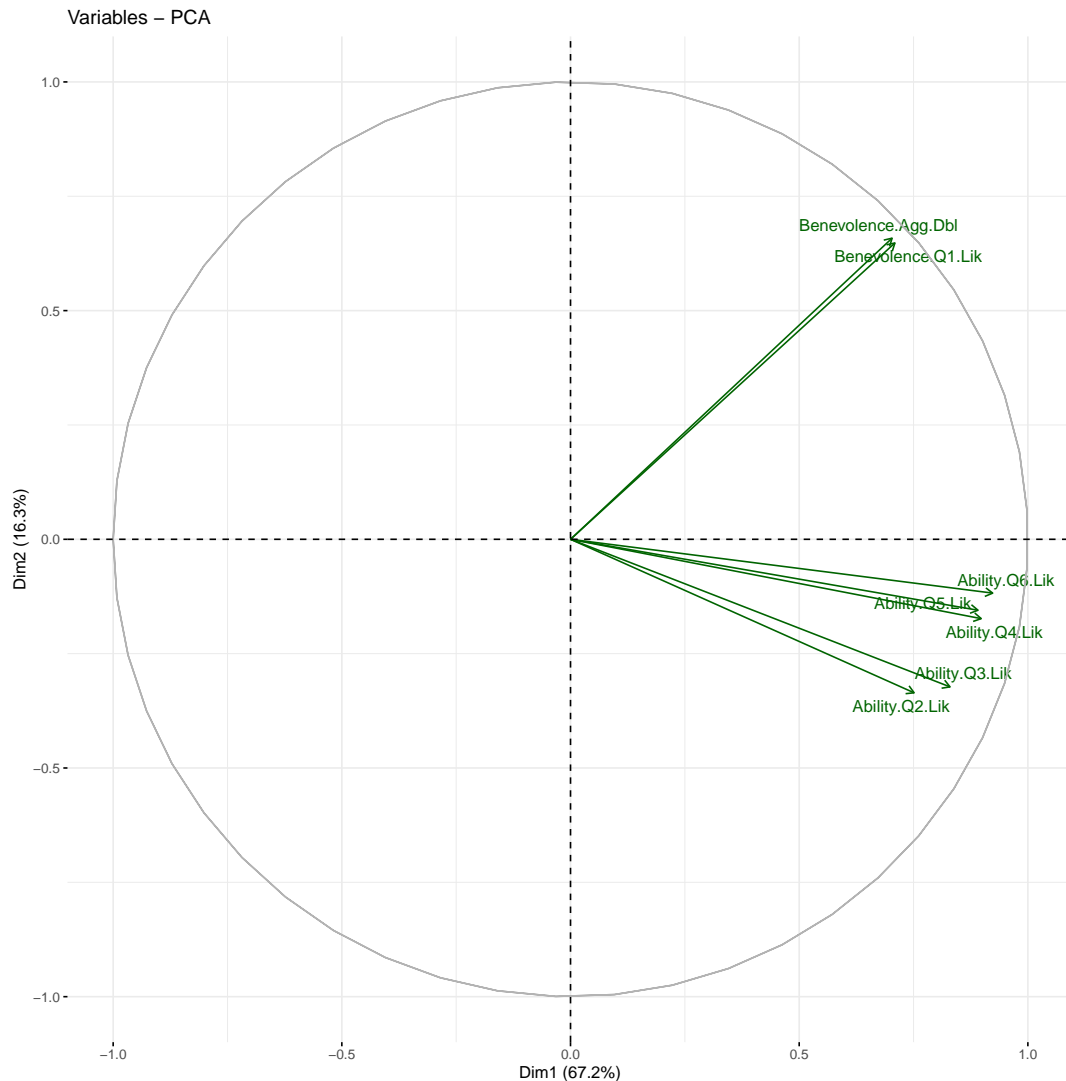
... and the 'Scree Plot'

We see that the first two dimensions already explain `83.4480978 %` of variance in the data. However, the first dimension is the only one which has the eigenvalue higher than 1.0 (i.e., it explains more than itself). As a conclusion, only the `first dimension (or Principal Component)` should be considered.

... next, we plot the `Variable Factor Map`

Variables – PCA

... show the correlation of the variables along the first dimension.

```
# Correlation table
res.pca$var$cor[, c(1,2)]

##                         Dim.1       Dim.2
## Ability.Q2.Lik       0.7512896 -0.3356001
## Ability.Q3.Lik       0.8298969 -0.3226787
## Ability.Q4.Lik       0.8980338 -0.1734353
## Ability.Q5.Lik       0.8910578 -0.1551387
## Ability.Q6.Lik       0.9231546 -0.1176419
## Benevolence.Agg.Dbl  0.7031487  0.6583988
## Benevolence.Q1.Lik   0.7093214  0.6478878
```

If we examine the above figure and the correlation table, we see that there is a `strong correlation` of all the variables with the first principal component. Also, since the aggregate variable is horizontal to the first dimension, and as the other questions on `Ability` are also strongly correlated, we can safely combine all the questions into a single variable.

... and the `Individual Factor Map`

```
## Error in ans[ypos] <- rep(yes, length.out = len)[ypos]:  replacement has length zero
```

7.2   Benevolence

... preparing the data for PCA. Including `Profile` as a qualitative supplementary variable (as it demonstrated the most discriminatory effect).

48

```
# Selecting the columns for PCA.
dt.pca = dt[, c(11, 21:26)]
```

... conducting the PCA.
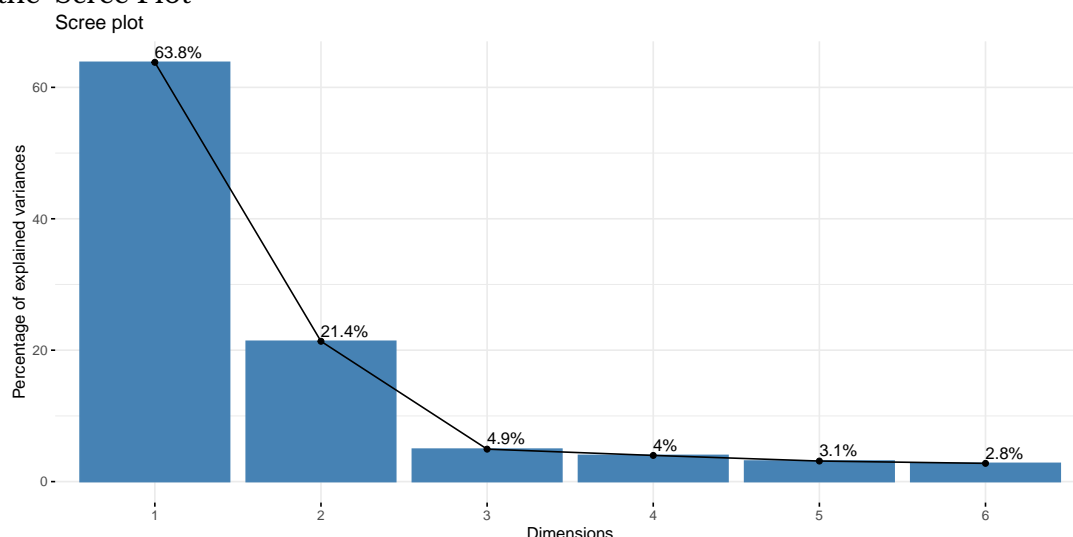
```
# Principal Component Analysis
res.pca = PCA(dt.pca,
              scale.unit = TRUE,
              quali.sup = c(1),
              ncp = 10,
              graph = FALSE)
```

... having a look at the eigen values.

```
# Eigen Values
get_eigenvalue(res.pca)

##        eigenvalue variance.percent cumulative.variance.percent
## Dim.1  3.8287318         63.812197                    63.81220
## Dim.2  1.2813307         21.355511                    85.16771
## Dim.3  0.2964207          4.940345                    90.10805
## Dim.4  0.2391773          3.986288                    94.09434
## Dim.5  0.1877204          3.128674                    97.22302
## Dim.6  0.1666190          2.776984                   100.00000
```
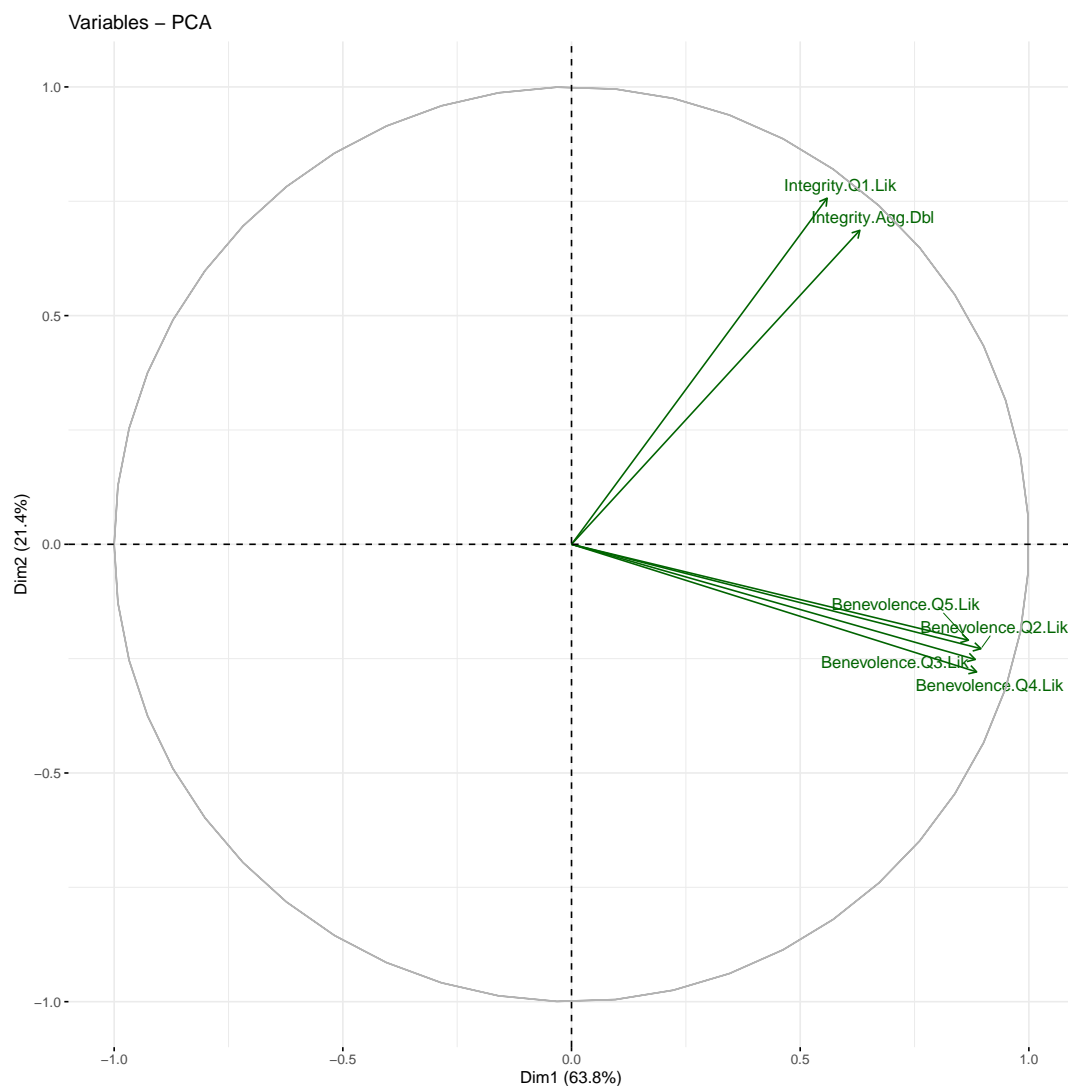
... and the 'Scree Plot'

We see that the first two dimensions already explain `85.1677084 %` of variance in the data. However, the first dimension is the only one which has the eigenvalue higher than 1.0 (i.e., it explains more than itself). As a conclusion, only the `first dimension (or Principal Component)` should be considered.

... next, we plot the `Variable Factor Map`

Variables – PCA

... show the correlation of the variables along the first dimension.

```
# Correlation table
res.pca$var$cor[, c(1,2)]

##                       Dim.1      Dim.2
## Benevolence.Q2.Lik 0.8952071 -0.2282723
## Benevolence.Q3.Lik 0.8828861 -0.2517888
## Benevolence.Q4.Lik 0.8859045 -0.2790032
## Benevolence.Q5.Lik 0.8678711 -0.2096526
## Integrity.Agg.Dbl  0.6304391  0.6865211
## Integrity.Q1.Lik   0.5588986  0.7567805
```

If we examine the above figure and the correlation table, we see that there is a `strong correlation` of all the variables with the first principal component. Also, since the aggregate variable is horizontal to the first dimension, and as the other questions on `Benevolence` are also strongly correlated, we can safely combine all the questions into a single variable.

... and the `Individual Factor Map`

```
## Error in ans[ypos] <- rep(yes, length.out = len)[ypos]:  replacement has length zero
```

7.3   Integrity

... preparing the data for PCA. Including `Profile` as a qualitative supplementary variable (as it demonstrated the most discriminatory effect).

```
# Selecting the columns for PCA.
dt.pca = dt[, c(11, 27:33)]
```

... conducting the PCA.

```
# Principal Component Analysis
res.pca = PCA(dt.pca,
              scale.unit = TRUE,
              quali.sup = c(1),
              ncp = 10,
              graph = FALSE)
```
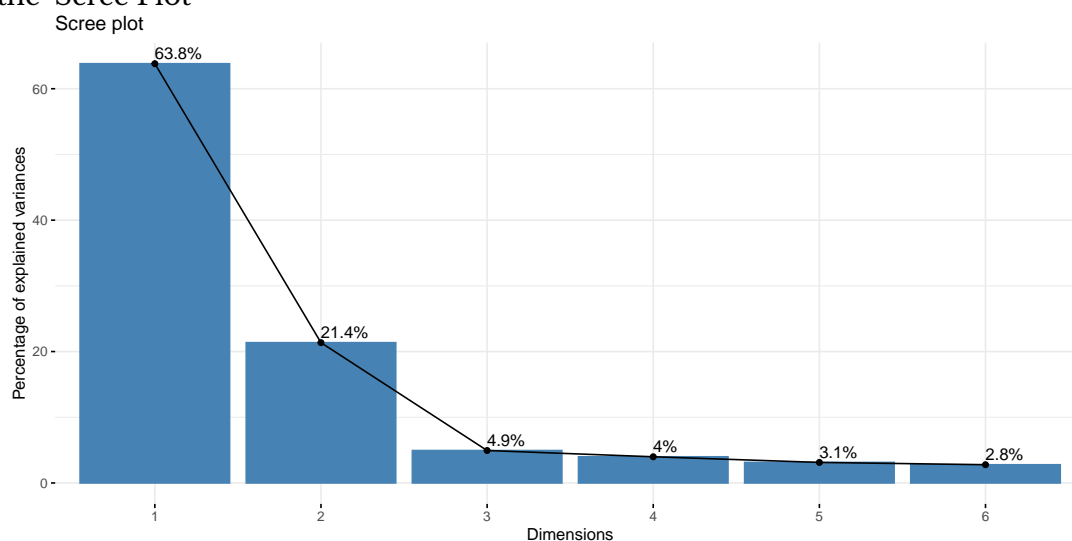
```
## Error in PCA(dt.pca, scale.unit = TRUE, quali.sup = c(1), ncp = 10, graph = FALSE):
## The following variables are not quantitative:   Ability.Agg.Cat
```

... having a look at the eigen values.

```
# Eigen Values
get_eigenvalue(res.pca)
```

```
##       eigenvalue variance.percent cumulative.variance.percent
## Dim.1  3.8287318        63.812197                    63.81220
## Dim.2  1.2813307        21.355511                    85.16771
## Dim.3  0.2964207         4.940345                    90.10805
## Dim.4  0.2391773         3.986288                    94.09434
## Dim.5  0.1877204         3.128674                    97.22302
## Dim.6  0.1666190         2.776984                   100.00000
```
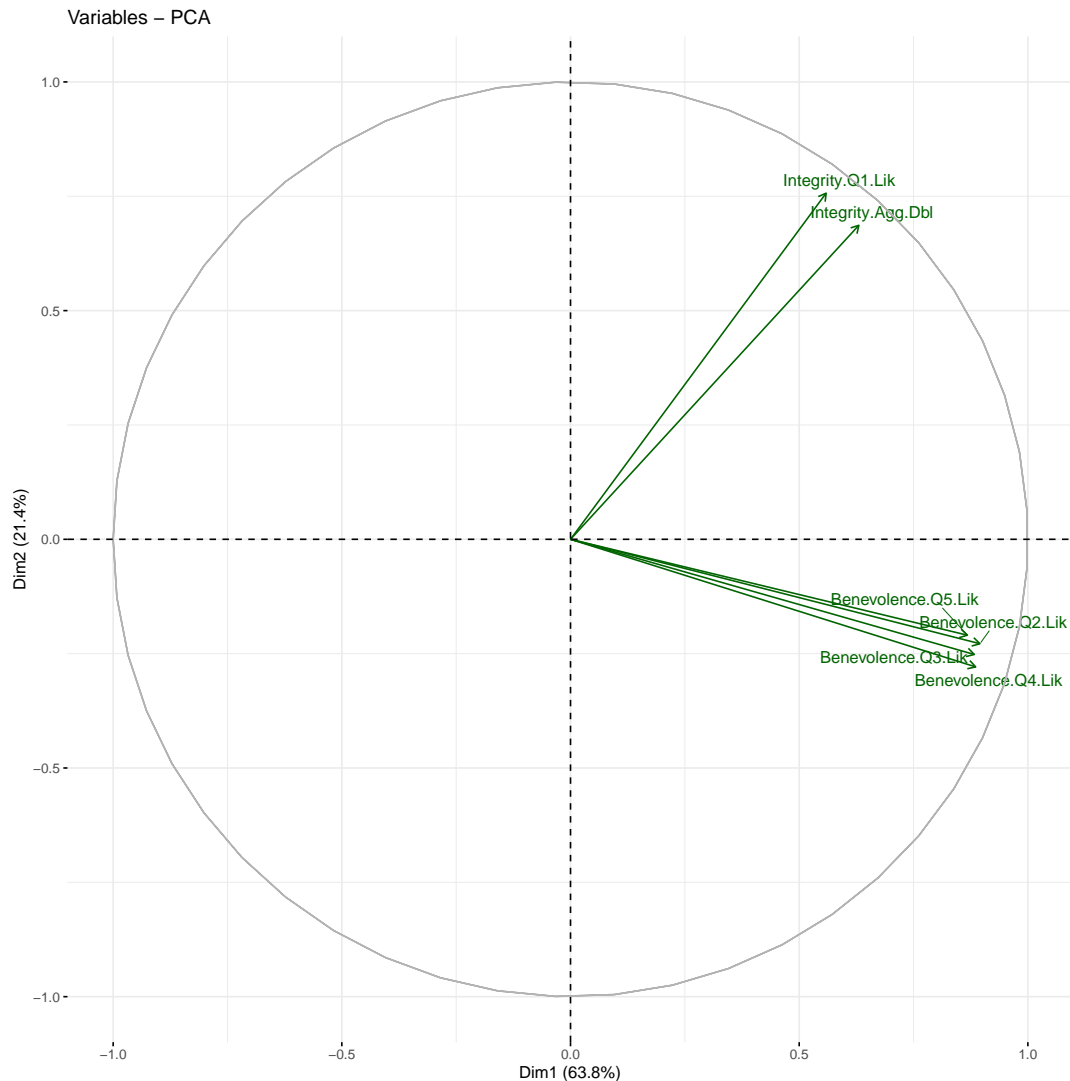
... and the 'Scree Plot'

We see that the first two dimensions already explain `85.1677084 %` of variance in the data. However, the first dimension is the only one which has the eigenvalue higher than 1.0 (i.e., it explains more than itself). As a conclusion, only the `first dimension (or Principal Component)` should be considered.

... next, we plot the `Variable Factor Map`

Variables – PCA

... show the correlation of the variables along the first dimension.

```
# Correlation table
res.pca$var$cor[, c(1,2)]

##                         Dim.1      Dim.2
## Benevolence.Q2.Lik  0.8952071 -0.2282723
## Benevolence.Q3.Lik  0.8828861 -0.2517888
## Benevolence.Q4.Lik  0.8859045 -0.2790032
## Benevolence.Q5.Lik  0.8678711 -0.2096526
## Integrity.Agg.Dbl   0.6304391  0.6865211
## Integrity.Q1.Lik    0.5588986  0.7567805
```

If we examine the above figure and the correlation table, we see that there is a `strong correlation` of all the variables, except `Integrity.Q4.Lik` with the first principal component. Also, since the aggregate variable is horizontal to the first dimension, and as the other questions on `Integrity` are also strongly correlated, we can safely combine all the questions into a single variable. This can be done despite moderate correlation value of `Integrity.Q4.Lik` because the second dimension has a eigenvalue less than 1.0.

... and the `Individual Factor Map`

```
## Error in ans[ypos] <- rep(yes, length.out = len)[ypos]:  replacement has length zero
```