GLM-SEM Model

Richard Lubega

2025-08-03

Model Development

Part 1: Generating Simulated Dataset

```
# Simulated Dataset:
pacman::p_load(tidymodels)
pacman::p_load(ggplot2)

set.seed(123)
n <- 500

freeway_data <- data.frame(
  vehicle_id = 1:n,
  std_speed = abs(rnorm(n, 5, 1.5)),  # speed variability
  gap_var = abs(rnorm(n, 3, 1.2)),    # gap size variation
  lane_density = rnorm(n, 30, 5),
  short_headway = rbinom(n, 1, 0.4),
  speed = rnorm(n, 100, 15),
  accel = rnorm(n, 0.5, 0.2),
  surrounding_gaps = rnorm(n, 2.5, 0.6),
  onramp_distance = runif(n, 50, 300),
  lane_change_freq = rpois(n, lambda = 2)  # target variable
)

head(freeway_data) # View first few rows of the dataset
```

```
##   vehicle_id std_speed  gap_var lane_density short_headway     speed     accel
## 1          1  4.159287 2.277729     25.02101             0  89.47742 0.2493328
## 2          2  4.654734 1.807562     24.80022             1 113.23352 0.4777336
## 3          3  7.338062 4.232142     29.91010             0  97.99944 0.2174373
## 4          4  5.105763 3.901274     29.33912             1  83.18982 0.1034092
## 5          5  5.193932 1.189000     17.25329             0 106.91789 0.6567191
## 6          6  7.572597 2.885823     35.20287             1 122.86214 0.6801739
##   surrounding_gaps onramp_distance lane_change_freq
## 1         1.829849        224.6009                4
## 2         1.965032        262.9375                2
## 3         3.023437        167.2628                0
## 4         3.621406        115.0119                3
## 5         2.425439        248.8313                0
## 6         2.564217        156.2903                3
```

```r
summary(freeway_data) # Get an overview of the dataset
```

```
##    vehicle_id      std_speed        gap_var          lane_density
##  Min.   :  1.0   Min.   :1.009   Min.   :0.009501   Min.   :16.52
##  1st Qu.:125.8   1st Qu.:4.138   1st Qu.:2.173938   1st Qu.:26.59
##  Median :250.5   Median :5.031   Median :2.998574   Median :30.30
##  Mean   :250.5   Mean   :5.052   Mean   :2.999211   Mean   :30.13
##  3rd Qu.:375.2   3rd Qu.:6.028   3rd Qu.:3.771901   3rd Qu.:33.30
##  Max.   :500.0   Max.   :9.862   Max.   :6.230057   Max.   :46.95
##  short_headway       speed            accel          surrounding_gaps
##  Min.   :0.000   Min.   : 60.56   Min.   :-0.06971   Min.   :0.7301
##  1st Qu.:0.000   1st Qu.: 90.18   1st Qu.: 0.36374   1st Qu.:2.1182
##  Median :0.000   Median : 99.49   Median : 0.49619   Median :2.4609
##  Mean   :0.406   Mean   : 99.78   Mean   : 0.49507   Mean   :2.5028
##  3rd Qu.:1.000   3rd Qu.:109.79   3rd Qu.: 0.63542   3rd Qu.:2.8795
##  Max.   :1.000   Max.   :142.24   Max.   : 1.10442   Max.   :4.5527
##  onramp_distance  lane_change_freq
##  Min.   : 50.22   Min.   :0.000
##  1st Qu.:115.15   1st Qu.:1.000
##  Median :172.66   Median :2.000
##  Mean   :174.33   Mean   :2.024
##  3rd Qu.:233.14   3rd Qu.:3.000
##  Max.   :299.43   Max.   :6.000
```

```r
str(freeway_data)
```

```
## 'data.frame':    500 obs. of  10 variables:
##  $ vehicle_id      : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ std_speed       : num  4.16 4.65 7.34 5.11 5.19 ...
##  $ gap_var         : num  2.28 1.81 4.23 3.9 1.19 ...
##  $ lane_density    : num  25 24.8 29.9 29.3 17.3 ...
##  $ short_headway   : int  0 1 0 1 0 1 1 0 1 1 ...
##  $ speed           : num  89.5 113.2 98 83.2 106.9 ...
##  $ accel           : num  0.249 0.478 0.217 0.103 0.657 ...
##  $ surrounding_gaps: num  1.83 1.97 3.02 3.62 2.43 ...
##  $ onramp_distance : num  225 263 167 115 249 ...
##  $ lane_change_freq: int  4 2 0 3 0 3 1 1 2 2 ...
```

```r
glimpse(freeway_data)
```

```
## Rows: 500
## Columns: 10
## $ vehicle_id       <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16~
## $ std_speed        <dbl> 4.159287, 4.654734, 7.338062, 5.105763, 5.193932, 7.5~
## $ gap_var          <dbl> 2.2777286, 1.8075617, 4.2321421, 3.9012736, 1.1890002~
## $ lane_density     <dbl> 25.02101, 24.80022, 29.91010, 29.33912, 17.25329, 35.~
## $ short_headway    <int> 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 0,~
## $ speed            <dbl> 89.47742, 113.23352, 97.99944, 83.18982, 106.91789, 1~
## $ accel            <dbl> 0.2493328, 0.4777336, 0.2174373, 0.1034092, 0.6567191~
## $ surrounding_gaps <dbl> 1.829849, 1.965032, 3.023437, 3.621406, 2.425439, 2.5~
## $ onramp_distance  <dbl> 224.60087, 262.93753, 167.26281, 115.01191, 248.83130~
## $ lane_change_freq <int> 4, 2, 0, 3, 0, 3, 1, 1, 2, 2, 2, 3, 4, 6, 1, 1, 0, 6,~
```

```
# # Boxplot for city mpg by cylinders
# boxplot(cty ~ cyl, data = mpg,
#         main = "City mpg by Number of Cylinders",
#         xlab = "Cylinders",
#         ylab = "City mpg (miles per gallon)")
#
# # Boxplot for highway mpg by cylinders
# boxplot(hwy ~ cyl, data = mpg,
#         main = "Highway mpg by Number of Cylinders",
#         xlab = "Cylinders",
#         ylab = "Highway mpg (miles per gallon)")
#
# par(mfrow = c(1, 1)) # Reset plot layout to default
#
# # Combine plots by faceting
```

Part 1: Model Building and Analysis

```
pacman::p_load(lavaan)

model <- '
  # Measurement model
  FI =~ std_speed + gap_var + lane_density
  DU =~ speed + short_headway + accel
  PO =~ surrounding_gaps + onramp_distance

  # Structural model
  lane_change_freq ~ FI + DU + PO
'

fit <- sem(model, data = freeway_data, estimator = "MLM")
```

```
## Warning: lavaan->lav_data_full():
##    some observed variances are (at least) a factor 1000 times larger than
##    others; use varTable(fit) to investigate
```

```
## Warning: lavaan->lav_lavaan_step11_estoptim():
##    Model estimation FAILED! Returning starting values.
```

```
summary(fit, fit.measures = TRUE, standardized = TRUE)
```

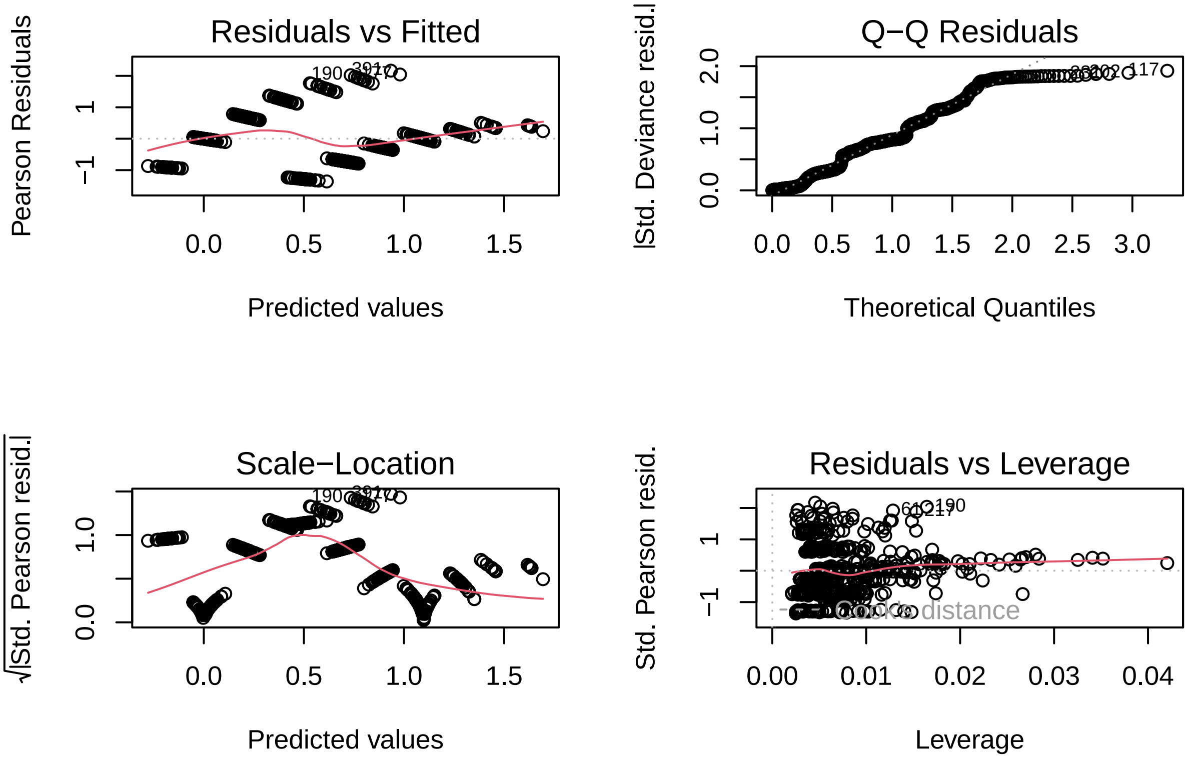
```
## Warning: lavaan->lav_object_summary():
##    fit measures not available if model did not converge
```

```
## lavaan 0.6-19 did NOT end normally after 3101 iterations
## ** WARNING ** Estimates below are most likely unreliable
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        23
##
##   Number of observations                           500
##
##
## Parameter Estimates:
##
##   Standard errors                           Robust.sem
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                    Estimate     Std.Err  z-value  P(>|z|)   Std.lv     Std.all
##   FI =~
##     std_speed            1.000                                  0.007    0.005
##     gap_var              0.546       NA                         0.004    0.003
##     lane_density     10778.055       NA                        80.408   16.185
##   DU =~
##     speed                1.000                                     NA       NA
##     short_headway     -974.153       NA                            NA       NA
##     accel               38.854       NA                            NA       NA
##   PO =~
##     surroundng_gps       1.000                                  0.002    0.002
##     onramp_distanc  294720.137       NA                       450.266    6.363
##
## Regressions:
##                      Estimate     Std.Err  z-value  P(>|z|)   Std.lv
##   lane_change_freq ~
##     FI                     0.120       NA                         0.001
##     DU                  2150.286       NA                            NA
##     PO                     4.949       NA                         0.008
##   Std.all
##
##     0.001
##        NA
##     0.005
##
## Covariances:
##                    Estimate     Std.Err  z-value  P(>|z|)   Std.lv     Std.all
##   FI ~~
##     DU                  -0.000       NA                        -0.007   -0.007
##     PO                  -0.000       NA                        -0.000   -0.000
##   DU ~~
##     PO                  -0.000       NA                        -0.026   -0.026
##
## Variances:
##                    Estimate     Std.Err  z-value  P(>|z|)   Std.lv     Std.all
##    .std_speed            2.131       NA                         2.131    1.000
##    .gap_var              1.457       NA                         1.457    1.000
##    .lane_density     -6440.690       NA                     -6440.690 -260.940
##    .speed              210.914       NA                       210.914    1.000
##    .short_headway        0.251       NA                         0.251    1.040
##    .accel                0.039       NA                         0.039    1.000
##    .surroundng_gps       0.374       NA                         0.374    1.000
##    .onramp_distanc -197731.387       NA                    -197731.387  -39.486
##    .lane_chang_frq       1.966       NA                          1.966    1.024
##     FI                   0.000       NA                          1.000    1.000
##     DU                  -0.000       NA                             NA       NA
##     PO                   0.000       NA                          1.000    1.000
```

```r
# Check model degrees of freedom
# lavInspect(fit, "df")

# Print modification indices
# modindices(fit, sort = TRUE, minimum.value = 10)

# Step 2: Extract factor scores
latent_scores <- lavPredict(fit)

# Merge with original data
hybrid_data <- cbind(freeway_data, latent_scores)

# Poisson (Count Model)
glm_fit <- glm(lane_change_freq ~ FI + DU + PO,
               data = hybrid_data,
               family = poisson(link = "log"))

summary(glm_fit)
```

```
##
## Call:
## glm(formula = lane_change_freq ~ FI + DU + PO, family = poisson(link = "log"),
##     data = hybrid_data)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.005e+00  3.549e-02  28.313  < 2e-16 ***
## FI          -1.563e+00  2.884e-01  -5.420 5.96e-08 ***
## DU          -1.627e+04  1.252e+03 -12.993  < 2e-16 ***
## PO          -2.803e+01  3.896e+00  -7.195 6.26e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 545.58  on 499  degrees of freedom
## Residual deviance: 371.95  on 496  degrees of freedom
## AIC: 1529.9
##
## Number of Fisher Scoring iterations: 5
```

```r
# Using base R for diagnostics
par(mfrow = c(2, 2))
plot(glm_fit)
```



```
par(mfrow = c(1,1))

pacman::p_load(BiocManager)
# pacman::p_load(countreg)
# rootogram(glm_fit)  # visually compares observed vs. predicted counts



# Negative Binomial (in case of overdispersion)
pacman::p_load(MASS)

nb_fit <- glm.nb(lane_change_freq ~ FI + DU + PO, data = hybrid_data)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
summary(nb_fit)
```

```
##
## Call:
## glm.nb(formula = lane_change_freq ~ FI + DU + PO, data = hybrid_data,
##     init.theta = 48242.13318, link = log)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.005e+00  3.549e-02  28.312  < 2e-16 ***
## FI          -1.563e+00  2.885e-01  -5.420 5.96e-08 ***
## DU          -1.627e+04  1.252e+03 -12.992  < 2e-16 ***
## PO          -2.803e+01  3.896e+00  -7.194 6.27e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(48242.13) family taken to be 1)
##
##     Null deviance: 545.56  on 499  degrees of freedom
## Residual deviance: 371.94  on 496  degrees of freedom
## AIC: 1531.9
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  48242
##           Std. Err.:  270434
## Warning while fitting theta: iteration limit reached
##
##  2 x log-likelihood:  -1521.912
```

```r
# Logistic Model (Lane-Change Occurrence)
# modeling whether a vehicle changed lanes or not (i.e., binary variable)
hybrid_data$changed_lane <- ifelse(hybrid_data$lane_change_freq > 0, 1, 0)

logit_fit <- glm(changed_lane ~ FI + DU + PO,
                 data = hybrid_data,
                 family = binomial(link = "logit"))

summary(logit_fit)
```

```
##
## Call:
## glm(formula = changed_lane ~ FI + DU + PO, family = binomial(link = "logit"),
##     data = hybrid_data)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.451e+00  3.156e-01  10.935  < 2e-16 ***
## FI          -2.639e+00  1.350e+00  -1.956 0.050522 .
## DU          -4.367e+04  6.624e+03  -6.592 4.33e-11 ***
## PO          -6.428e+01  1.881e+01  -3.418 0.000631 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 374.82  on 499  degrees of freedom
## Residual deviance: 316.69  on 496  degrees of freedom
## AIC: 324.69
##
## Number of Fisher Scoring iterations: 6
```

Therefore, based on the analysis, …….