

Interactive Effects of Smartphone Ownership, Education, and Life Satisfaction

Shen Chengrui (Alex)

2025-06-19

```
df_raw <- readxl::read_excel(
  "../data/F00013183-WVS_Wave_7_China_Excel_v5.1.xlsx",
  sheet = 1,
  col_names = TRUE
)

df <- df_raw %>%
  transmute(
    life_sat = as.numeric(`Q49: Satisfaction with your life`),
    age = as.numeric(`Q262: Age`),
    education = factor(
      as.numeric(`Q275R: Highest educational level: Respondent (recoded into 3 gr
      levels = 1:3,
      labels = c("Primary", "Secondary", "Tertiary")
    ),
    phone_use = if_else(as.numeric(`Q204: Information source: Mobile phone`) %in% 1:4, 1, 0),
    female = if_else(as.numeric(`Q260: Sex`) == 2, 1, 0),
    income_cat = factor(
      as.numeric(`Q288R: Income level (Recoded)`),
      levels = 1:3,
      labels = c("Low", "Middle", "High")
    )
  ) %>%
  filter(
    !is.na(life_sat),
    !is.na(age),
    !is.na(education),
    !is.na(phone_use),
    !is.na(income_cat)
  )

run_model <- function(subgroup) {
  df_sub <- filter(df, female == subgroup)
  lm(life_sat ~ education * phone_use + age + income_cat, data = df_sub)
}
```

```
models_by_gender <- map(c(0,1), run_model)
names(models_by_gender) <- c("Male", "Female")
modelsummary(models_by_gender, title = "Models by Gender", stars = TRUE)
```

```
skim(df)
```

Table 2: Data summary

Name	df
Number of rows	2989
Number of columns	6
Column type frequency:	
factor	2
numeric	4
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
education	0	1	FALSE	3	Pri: 1632, Sec: 685, Ter: 672
income_cat	0	1	FALSE	3	Mid: 1807, Low: 1102, Hig: 80

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
life_sat	0	1	7.38	2.11	-2	6	8	9	10	
age	0	1	44.58	14.52	18	32	45	56	70	
phone_use	0	1	0.84	0.37	0	1	1	1	1	
female	0	1	0.55	0.50	0	0	1	1	1	

```
model <- lm(
  life_sat ~ education * phone_use + female + age + income_cat,
  data = df
)
modelsummary(model, stars = TRUE, statistic = "std.error")
```

Table 1: Models by Gender

	Male	Female
(Intercept)	6.158*** (0.302)	5.329*** (0.294)
educationSecondary	−0.207 (0.493)	−0.809+ (0.461)
educationTertiary	−0.103 (0.738)	0.501 (0.801)
phone_use	−0.222 (0.188)	0.167 (0.159)
age	0.019*** (0.004)	0.030*** (0.004)
income_catMiddle	0.659*** (0.118)	0.887*** (0.111)
income_catHigh	1.624*** (0.349)	1.184*** (0.329)
educationSecondary × phone_use	0.361 (0.513)	0.892+ (0.481)
educationTertiary × phone_use	0.214 (0.748)	−0.378 (0.807)
Num.Obs.	1349	1640
R2	0.045	0.064
R2 Adj.	0.040	0.060
AIC	5745.4	7052.4
BIC	5797.4	7106.5
Log.Lik.	−2862.677	−3516.222
RMSE	2.02	2.06

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

	(1)
(Intercept)	5.685*** (0.218)
educationSecondary	−0.501 (0.337)
educationTertiary	0.183 (0.543)
phone__use	−0.007 (0.121)
female	0.099 (0.076)
age	0.025*** (0.003)
income_catMiddle	0.776*** (0.081)
income_catHigh	1.391*** (0.239)
educationSecondary × phone__use	0.607+ (0.350)
educationTertiary × phone__use	−0.077 (0.549)
Num.Obs.	2989
R2	0.053
R2 Adj.	0.050
AIC	12 790.3
BIC	12 856.4
Log.Lik.	−6384.161
RMSE	2.05
+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001	

```
robust_mod <- lm_robust(
  life_sat ~ education * phone_use + female + age + income_cat,
  data = df, se_type = "HC2"
)
modelsummary(
  list(OLS = model, RobustSE = robust_mod),
  stars = TRUE, statistic = "std.error",
  title = "OLS vs. Robust SE Comparison"
)
```

```
boot_fn <- function(data, idx) {
  d <- data[idx, ]
  coef(lm(life_sat ~ education * phone_use + female + age + income_cat, data = d))
}
set.seed(2025)
boot_res <- boot(df, boot_fn, R = 1000)
boot.ci(boot_res, index = 5, type = c("perc", "bca"))
```

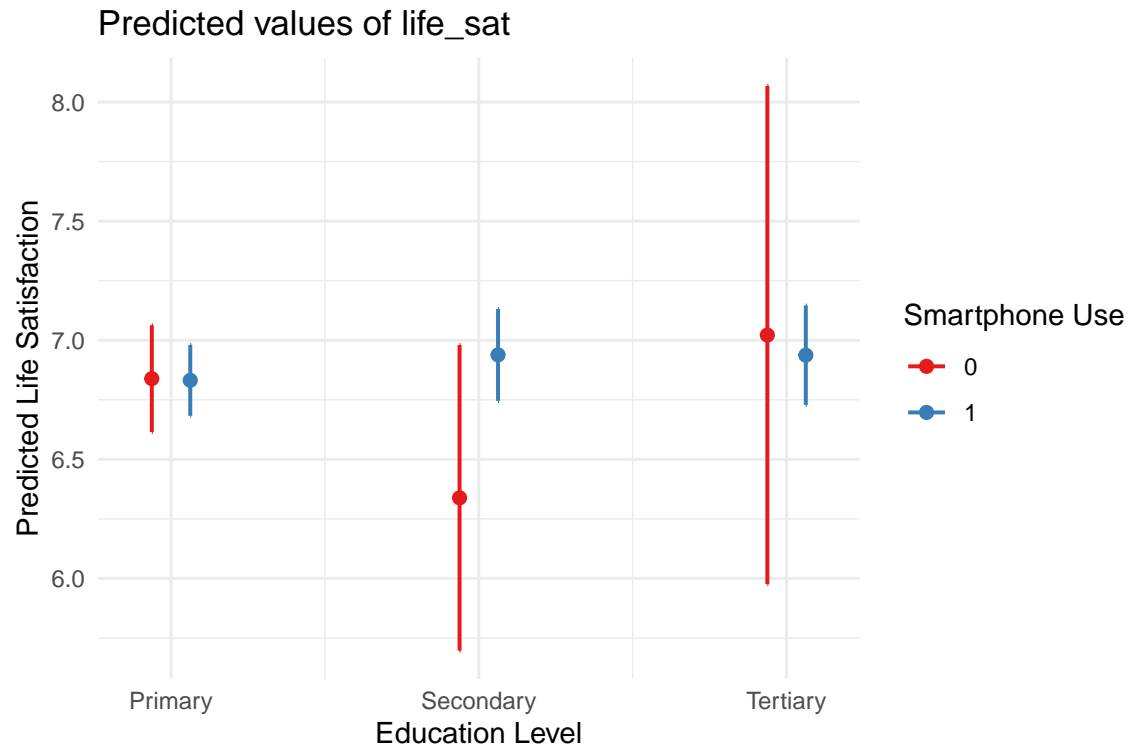
```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 1000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_res, type = c("perc", "bca"), index = 5)
##
## Intervals :
## Level      Percentile          BCa
## 95%      (-0.0525,  0.2417 )  (-0.0577,  0.2363 )
## Calculations and Intervals on Original Scale
```

```
pred <- ggpredict(model, terms = c("education", "phone_use"))
plot(pred) +
  labs(
    x = "Education Level",
    y = "Predicted Life Satisfaction",
    color = "Smartphone Use"
  ) +
  theme_minimal()
```

Table 5: OLS vs. Robust SE Comparison

	OLS	RobustSE
(Intercept)	5.685*** (0.218)	5.685*** (0.232)
educationSecondary	-0.501 (0.337)	-0.501 (0.442)
educationTertiary	0.183 (0.543)	0.183 (0.513)
phone_use	-0.007 (0.121)	-0.007 (0.136)
female	0.099 (0.076)	0.099 (0.076)
age	0.025*** (0.003)	0.025*** (0.003)
income_catMiddle	0.776*** (0.081)	0.776*** (0.085)
income_catHigh	1.391*** (0.239)	1.391*** (0.184)
educationSecondary \times phone_use	0.607+ (0.350)	0.607 (0.452)
educationTertiary \times phone_use	-0.077 (0.549)	-0.077 (0.517)
Num.Obs.	2989	2989
R2	0.053	0.053
R2 Adj.	0.050	0.050
AIC	12 790.3	12 790.3
BIC	12 856.4	12 856.4
Log.Lik.	-6384.161	
RMSE	2.05	2.05

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001



```
df <- df %>% mutate(edu_num = as.numeric(education))
ggplot(df, aes(x = edu_num, y = life_sat,
               color = factor(phone_use), group = phone_use)) +
  geom_jitter(width = 0.2, alpha = 0.3, size = 1) +
  geom_smooth(method = "lm", se = FALSE) +
  facet_wrap(~ female,
             labeller = labeller(female = c(`0`="Male", `1`="Female")))) +
  scale_x_continuous(breaks = 1:3,
                    labels = levels(df$education)) +
  labs(x = "Education Level",
       y = "Life Satisfaction",
       color = "Smartphone Use") +
  theme_minimal()
```



```
tbl <- modelsummary(model, output = "data.frame")
kable(tbl, caption = "OLS Regression Results", booktabs = TRUE) %>%
  kable_styling(full_width = FALSE, position = "left") %>%
  row_spec(0, bold = TRUE)
```

Table 6: OLS Regression Results

part	term	statistic	(1)
estimates	(Intercept)	estimate	5.685
estimates	(Intercept)	std.error	(0.218)
estimates	educationSecondary	estimate	-0.501
estimates	educationSecondary	std.error	(0.337)
estimates	educationTertiary	estimate	0.183
estimates	educationTertiary	std.error	(0.543)
estimates	phone_use	estimate	-0.007
estimates	phone_use	std.error	(0.121)
estimates	female	estimate	0.099
estimates	female	std.error	(0.076)
estimates	age	estimate	0.025
estimates	age	std.error	(0.003)
estimates	income_catMiddle	estimate	0.776
estimates	income_catMiddle	std.error	(0.081)
estimates	income_catHigh	estimate	1.391

estimates	income_catHigh	std.error	(0.239)
estimates	educationSecondary × phone_use	estimate	0.607
estimates	educationSecondary × phone_use	std.error	(0.350)
estimates	educationTertiary × phone_use	estimate	-0.077
estimates	educationTertiary × phone_use	std.error	(0.549)
gof	Num.Obs.		2989
gof	R2		0.053
gof	R2 Adj.		0.050
gof	AIC		12790.3
gof	BIC		12856.4
gof	Log.Lik.		-6384.161
gof	RMSE		2.05

sessionInfo()

```
## R version 4.5.0 (2025-04-11 ucrt)
## Platform: x86_64-w64-mingw32/x64
## Running under: Windows 11 x64 (build 26100)
##
## Matrix products: default
##   LAPACK version 3.12.1
##
## locale:
## [1] LC_COLLATE=Chinese (Simplified)_China.utf8
## [2] LC_CTYPE=Chinese (Simplified)_China.utf8
## [3] LC_MONETARY=Chinese (Simplified)_China.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=Chinese (Simplified)_China.utf8
##
## time zone: Asia/Shanghai
## tzcode source: internal
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods
## [7] base
##
## other attached packages:
##  [1] rmarkdown_2.29      here_1.0.1          kableExtra_1.4.0
##  [4] boot_1.3-31         estimatr_1.0.6      skimr_2.1.5
##  [7] ggeffects_2.3.0     modelsummary_2.4.0  lubridate_1.9.4
## [10] forcats_1.0.0       stringr_1.5.1       purrr_1.0.4
## [13] readr_2.1.5         tidyr_1.3.1         tibble_3.2.1
## [16] ggplot2_3.5.2       tidyverse_2.0.0     readxl_1.4.5
## [19] semPlot_1.1.6       lavaan_0.6-19       dplyr_1.1.4
## [22] haven_2.5.4
##
```

```
## loaded via a namespace (and not attached):
## [1] rstudioapi_0.17.1      jsonlite_2.0.0         datawizard_1.1.0
## [4] magrittr_2.0.3         farver_2.1.2           nloptr_2.2.1
## [7] vctr_0.6.5             minqa_1.2.8            base64enc_0.1-3
## [10] tinytex_0.57           htmltools_0.5.8.1      cellranger_1.1.0
## [13] Formula_1.2-5          htmlwidgets_1.6.4      plyr_1.8.9
## [16] igraph_2.1.4           lifecycle_1.0.4        pkgconfig_2.0.3
## [19] Matrix_1.7-3           R6_2.6.1               fastmap_1.2.0
## [22] rbibutils_2.3          digest_0.6.37          OpenMx_2.21.13
## [25] fdrtool_1.2.18         colorspace_2.1-1       rprojroot_2.0.4
## [28] textshaping_1.0.1      Hmisc_5.2-3            labeling_0.4.3
## [31] fansi_1.0.6            timechange_0.3.0       tinytable_0.9.0
## [34] abind_1.4-8            mgcv_1.9-1             compiler_4.5.0
## [37] withr_3.0.2            glasso_1.11            htmlTable_2.4.3
## [40] backports_1.5.0        carData_3.0-5          psych_2.5.3
## [43] performance_0.14.0     MASS_7.3-65            corpcor_1.6.10
## [46] gtools_3.9.5           tools_4.5.0            pbivnorm_0.6.0
## [49] foreign_0.8-90         zip_2.3.2              nnet_7.3-20
## [52] glue_1.8.0             quadprog_1.5-8         nlme_3.1-168
## [55] lisrelToR_0.3          grid_4.5.0             rsconnect_1.3.4
## [58] checkmate_2.3.2        cluster_2.1.8.1        reshape2_1.4.4
## [61] generics_0.1.3         gtable_0.3.6           tzdb_0.5.0
## [64] data.table_1.17.0      hms_1.1.3              xml2_1.3.8
## [67] sem_3.1-16             tables_0.9.31          pillar_1.10.2
## [70] rockchalk_1.8.157      splines_4.5.0          lattice_0.22-6
## [73] kutils_1.73            tidyselect_1.2.1       pbapply_1.7-2
## [76] knitr_1.50             reformulas_0.4.1        gridExtra_2.3
## [79] svglite_2.2.1          stats4_4.5.0           xfun_0.52
## [82] qgraph_1.9.8           arm_1.14-4             stringi_1.8.7
## [85] yaml_2.3.10            evaluate_1.0.3         mi_1.1
## [88] cli_3.6.4              RcppParallel_5.1.10    rpart_4.1.24
## [91] xtable_1.8-4           parameters_0.26.0       systemfonts_1.2.3
## [94] Rdpack_2.6.4           repr_1.1.7             munsell_0.5.1
## [97] Rcpp_1.0.14            coda_0.19-4.1          png_0.1-8
## [100] XML_3.99-0.18          parallel_4.5.0         bayestestR_0.16.0
## [103] jpeg_0.1-11            lme4_1.1-37            viridisLite_0.4.2
## [106] scales_1.3.0           openxlsx_4.2.8         insight_1.3.0
## [109] rlang_1.1.6            mnormt_2.1.1
```

Replication materials

All analysis code, the knitted PDF, and the one-line build script are openly available on GitHub:
https://github.com/Alex-Shen114514/WVS_report_final_files

Effect of Education and Residence on Life Satisfaction – WVS China 2018

Alex Shen

2025-07-11

Data and Methods

World Values Survey, Wave 7 (China, 2018). Logistic regression predicting high life satisfaction (score ≥ 7) by education, urban/rural status, age, and sex, with education \times urban interaction.

```
# Adjust the path as needed
df_raw <- read_excel("data/F00013183-WVS_Wave_7_China_Excel_v5.1.xlsx", sheet = 1)
df <- df_raw %>%
  transmute(
    life_satisfaction = as.numeric(`Q49: Satisfaction with your life`),
    education = factor(
      as.numeric(`Q275R: Highest educational level: Respondent (recoded into 3 groups)`),
      levels = 1:3,
      labels = c("Primary", "Secondary", "Tertiary")
    ),
    urban = as.numeric(`H_URBRURAL: Urban-Rural`),
    age = as.numeric(`Q262: Age`),
    sex = as.numeric(`Q260: Sex`)
  ) %>%
  filter(
    !is.na(life_satisfaction), !is.na(education), !is.na(urban), !is.na(age), !is.na(sex)
  ) %>%
  mutate(
    urban_bin = if_else(urban == 1, 1, 0),
    satisfied = if_else(life_satisfaction >= 7, 1, 0)
  )
nrow(df)
```

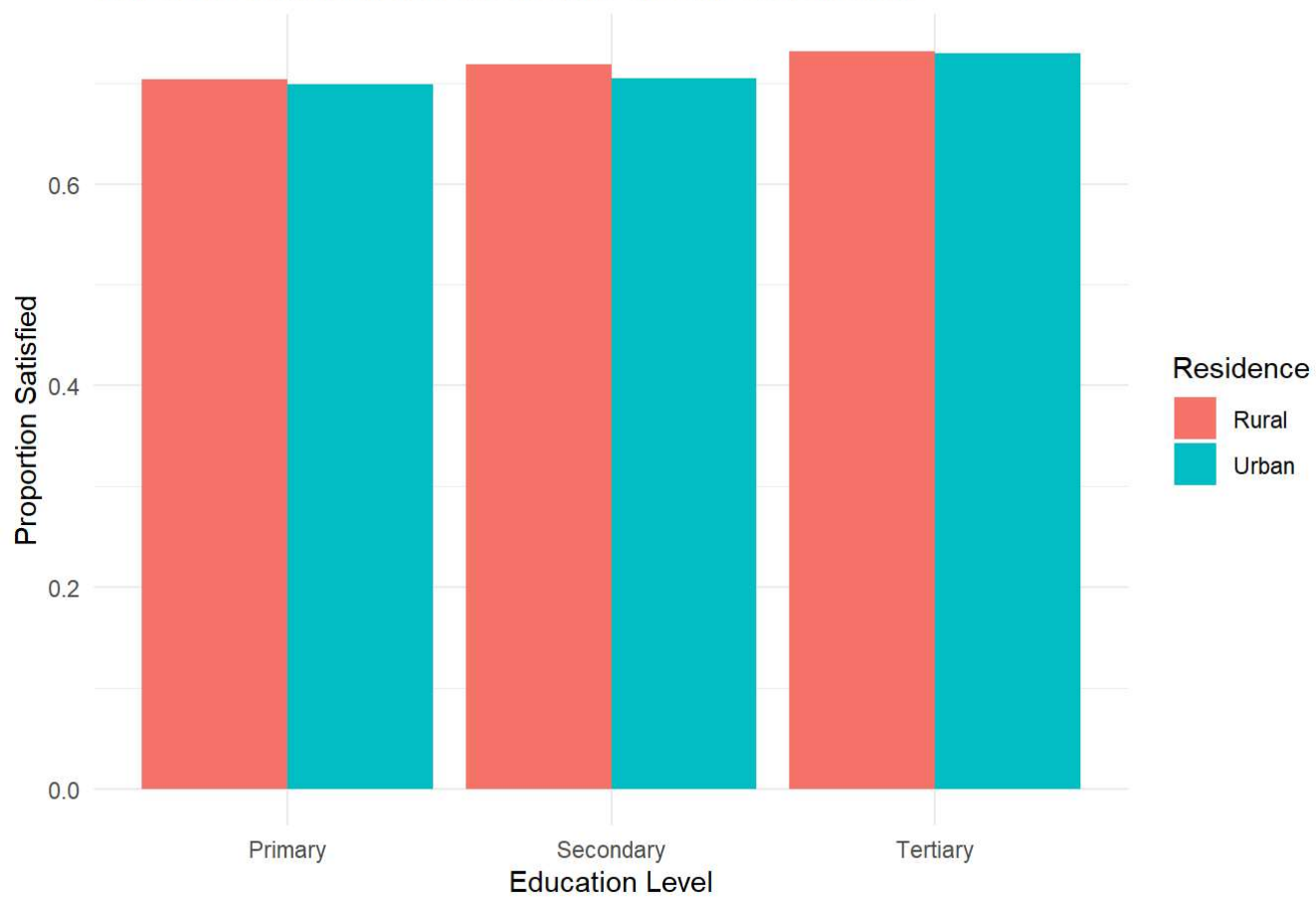
```
## [1] 3006
```

```
model <- glm(satisfied ~ education * urban_bin + age + sex, data = df, family = "binomial")
summary(model)
```

```
##
## Call:
## glm(formula = satisfied ~ education * urban_bin + age + sex,
##      family = "binomial", data = df)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -0.135423    0.222617  -0.608   0.5430
## educationSecondary    0.295802    0.188457   1.570   0.1165
## educationTertiary    0.496872    0.250355   1.985   0.0472 *
## urban_bin          -0.022624    0.108451  -0.209   0.8347
## age                0.017344    0.003169   5.473 4.43e-08 ***
## sex                0.087271    0.081581   1.070   0.2847
## educationSecondary:urban_bin -0.122208    0.222885  -0.548   0.5835
## educationTertiary:urban_bin -0.076170    0.271535  -0.281   0.7791
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 3624.2  on 3005  degrees of freedom
## Residual deviance: 3591.3  on 2998  degrees of freedom
## AIC: 3607.3
##
## Number of Fisher Scoring iterations: 4
```

```
df %>%
  group_by(education, urban_bin) %>%
  summarise(mean_satisfied = mean(satisfied, na.rm = TRUE)) %>%
  ggplot(aes(x = education, y = mean_satisfied, fill = factor(urban_bin, labels = c("Rural", "Urban")))) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(
    x = "Education Level",
    y = "Proportion Satisfied",
    fill = "Residence",
    title = "Effect of Education and Residence on Life Satisfaction"
  ) +
  theme_minimal()
```

Effect of Education and Residence on Life Satisfaction



Trust to Life Satisfaction (WVS China 2018)

Alex Shen

```
df_plot <- df %>%  
  group_by(urban, trustGrp) %>%  
  summarise(mean_sat = mean(life_sat), .groups = "drop")
```

```
gap_tbl <- df_plot %>%  
  pivot_wider(names_from = trustGrp, values_from = mean_sat) %>%  
  mutate(gap = `High trust` - `Low trust`)  
  
print(df_plot)
```

```
## # A tibble: 4 x 3  
##   urban trustGrp   mean_sat  
##   <fct> <chr>       <dbl>  
## 1 Urban High trust    7.45  
## 2 Urban Low trust     7.32  
## 3 Rural High trust    7.51  
## 4 Rural Low trust     7.07
```

```
print(gap_tbl)
```

```
## # A tibble: 2 x 4  
##   urban 'High trust' 'Low trust'   gap  
##   <fct>       <dbl>       <dbl> <dbl>  
## 1 Urban         7.45         7.32 0.132  
## 2 Rural         7.51         7.07 0.446
```

