
PO12Q - Introduction to Quantitative Political Analysis II:

Week 7 – Solutions to Additional Exercises

Dr Florian Reiche
F.Reiche@warwick.ac.uk



1 | Load Packages and data

Before starting, we need to load libraries and install packages if not already installed. In these exercises we will be using the following packages:

1. haven
2. ggplot2
3. stargazer

Set working directory and load the data. The file is on Moodle. It's called `simd.csv` and it's the "Scottish Index of Multiple Deprivation". More details here: <https://www.gov.scot/collections/scottish-index-of-multiple-deprivation-2020/>

Hint: Remember that the WD is the folder where you stored the data set and where all the outputs will be saved

Hint2: Note, the data is a `.csv` file

```
setwd("")

library(tidyverse)
library(haven)
library(stargazer)

sim <- read.csv("simd.csv", stringsAsFactors = T)
```

2 | Inspect your data

Here you can use several basic functions. The dataset does not contain too many variables, so you can start by using `names()`, `str()`, etc.

```
names(sim)
## [1] "data_zone" "intermediate_zone"
```

```
## [3] "council_area"      "health_board"
## [5] "pct_school_attend"  "alcohol"
## [7] "population"         "working_population"
## [9] "pct_income_deprived" "pct_employment_deprived"
## [11] "illness"            "mortality"
## [13] "drugs"              "pct_depress"
## [15] "pct_low_bw"         "hosp_emerg"
## [17] "noquals"            "crime"
## [19] "pct_no_heating"     "pct_overcrowded"
## [21] "simd_2016_quintile" "urban"
```

```
dim(sim)
## [1] 6976 22
```

3 | Preliminary Analysis

Now that you have a preliminary idea of the structure of the dataset, you can select two variables and test a possible relationship. The topic today is bivariate linear regression analysis, so remember that the outcome variable needs to be continuous.

Let's say we want to look at the relationship between alcohol consumption and mortality rate (yes, not a very funny topic, but interesting nonetheless). The two variables are, respectively, alcohol and mortality (rather intuitive this time).

Now, formulate the working (alternative) and the null hypothesis. Write them down.

H₀:

H₁:

Which is your dependent variable?

Run a frequency table on the mortality variable. What is the level of measurement?

```
table(sim$mortality)
##
##  0  6  8  9 11 12 13 15 17 18 19 20 21 22 23 24 25 26 27 28
##  4  1  1  1  2  2  1  3  7  4  2  1  5  3  2  4  2  5  5  8
## 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
##  3  9  7  7 12  8 17 10  7 25 16 18 31 26 23 23 21 28 28 36
## 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68
## 33 43 37 39 51 54 46 49 46 57 67 59 61 73 70 71 80 67 85 57
## 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88
## 87 64 71 86 76 84 97 88 70 70 79 74 84 81 83 68 68 74 98 70
## 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108
## 81 74 84 77 77 76 85 72 63 74 56 63 66 72 62 71 59 82 64 60
## 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128
## 61 63 52 62 61 47 66 53 53 42 44 47 54 46 59 52 43 39 40 45
```

```
## 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148
## 48 37 39 40 28 21 40 38 39 37 31 25 34 28 26 30 31 17 23 31
## 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168
## 22 21 21 11 20 22 19 25 18 21 21 18 25 17 13 16 17 15 9 8
## 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188
## 11 11 12 11 12 13 12 7 5 3 11 9 7 10 9 8 6 7 9 8
## 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 205 206 207 208 209
## 4 6 6 7 7 3 11 4 7 9 10 1 5 2 2 5 10 6 1 1
## 210 211 212 213 214 216 217 218 219 220 221 222 223 224 227 228 229 230 231 232
## 2 3 3 1 2 3 2 1 1 5 4 3 1 3 2 1 1 3 3 1
## 233 235 237 239 240 241 242 244 246 248 251 253 255 259 261 263 266 269 272 273
## 1 1 3 2 2 5 1 2 2 1 1 1 2 2 1 1 1 1 1 1
## 282 286 294 299 303 310 312 324 340 346 356 385 398 399 411 472 473 476 504 523
## 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## 557 563 589 669 719 950
## 1 1 1 1 1 1
```

```
class(sim$mortality)
## [1] "integer"
```

Do the same for the other variable. And guess what is the level of measurement.

```
table(sim$alcohol)
##
## 0 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
## 117 4 47 64 53 60 52 51 42 36 63 50 55 60 63 42
## 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34
## 51 57 64 60 54 58 63 68 54 82 56 41 65 80 70 54
## 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
## 62 54 61 58 54 47 43 45 49 31 54 46 60 55 34 44
## 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66
## 44 53 40 47 39 37 41 32 52 40 49 35 42 35 39 30
## 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82
## 39 39 51 34 36 45 35 42 30 35 40 38 40 35 29 28
## 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98
## 43 24 31 30 26 39 31 32 29 26 28 29 28 38 25 31
## 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114
## 27 21 18 27 23 13 31 22 31 37 23 23 21 23 23 20
## 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130
## 25 29 28 17 21 28 26 19 20 26 17 17 25 28 19 16
## 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146
## 19 20 22 21 13 21 16 9 25 18 12 11 14 16 19 16
## 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162
## 13 13 23 15 16 14 17 19 19 13 17 16 9 15 12 18
## 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178
## 8 12 9 16 8 12 10 16 11 12 16 9 11 13 8 7
## 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194
## 9 6 16 15 13 5 7 17 8 10 9 11 10 9 21 9
## 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210
## 13 15 15 6 6 4 10 13 14 11 7 11 10 11 8 14
## 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226
## 6 9 9 12 10 8 4 5 6 13 10 6 6 5 3 9
## 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242
```

```
## 6 5 8 6 8 5 4 8 6 8 9 8 6 8 4 3
## 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258
## 6 8 11 4 6 4 9 7 9 11 3 11 5 8 2 10
## 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274
## 6 6 9 2 8 5 4 5 5 3 5 1 5 7 3 6
## 275 276 277 278 279 280 282 283 284 285 286 287 288 289 290 291
## 3 7 2 5 7 2 2 6 6 4 2 3 5 3 3 1
## 292 293 294 295 296 297 298 299 301 302 303 304 306 307 308 309
## 5 4 1 7 2 6 6 4 4 1 1 1 5 2 3 1
## 310 312 313 314 315 316 317 318 319 320 321 322 324 325 326 327
## 3 1 1 2 1 3 1 2 3 2 1 2 1 4 2 1
## 329 330 331 333 334 335 336 337 338 339 340 342 343 344 345 346
## 2 2 4 2 2 2 1 1 3 3 3 2 5 1 1 1
## 347 348 349 350 351 353 354 355 356 357 358 359 360 361 362 363
## 2 3 3 3 2 2 1 2 2 2 1 1 3 2 2 3
## 364 365 366 367 369 371 372 373 374 375 376 377 379 380 381 382
## 3 1 4 1 1 4 5 2 2 1 1 2 1 1 5 1
## 384 385 386 390 391 392 393 395 396 397 398 400 402 403 404 405
## 1 2 1 2 1 4 1 1 1 2 1 2 2 4 1 1
## 406 408 409 416 417 418 419 421 423 424 426 428 429 430 432 433
## 2 1 2 1 1 2 1 1 1 1 1 1 1 2 3 1
## 438 441 442 443 444 445 447 448 449 450 451 453 454 455 458 460
## 2 1 1 1 4 1 1 3 1 2 1 1 1 2 1 2
## 462 463 466 469 470 472 474 480 482 484 485 488 491 492 493 494
## 1 1 1 1 1 1 1 1 2 1 2 1 1 1 2 1
## 501 504 505 506 511 513 514 516 521 526 528 529 531 533 538 539
## 2 1 1 1 1 2 1 1 1 2 1 2 1 1 1 2
## 541 542 543 545 549 550 552 553 557 558 564 568 574 584 600 619
## 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1
## 629 635 636 646 652 670 675 694 699 715 754 766 769 795 845 889
## 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## 1703 2351
## 1 1
```

```
class(sim$alcohol)
## [1] "integer"
```

4 | Visualisation

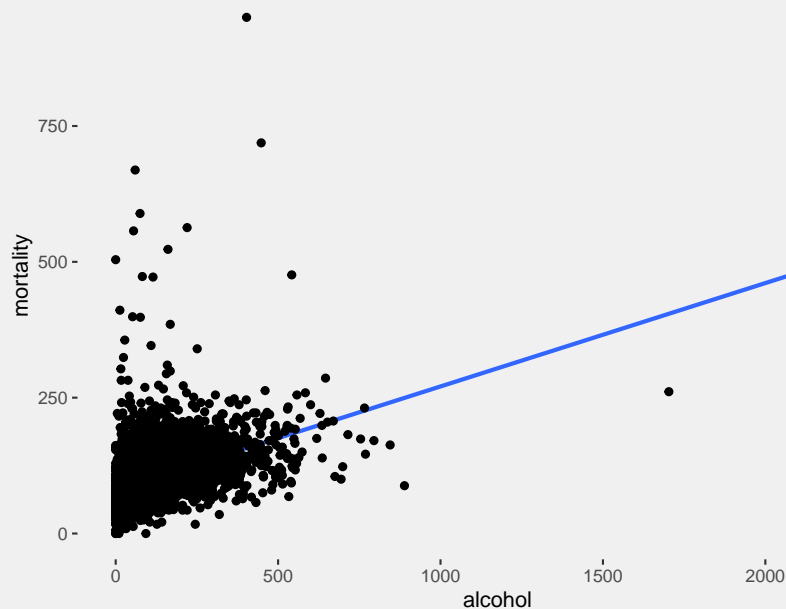
Let's start with the visualisation of the relationship between the two variables. What is the best way to visualise the relationship considering the level of measurement of our variables?

Hint: Probably a scatterplot, right? So, use a scatterplot to visualise the relationship and add the regression line.

You can use `ggplot`, but also the standard `plot()` function.

```
ggplot(sim, aes(x = alcohol, y = mortality)) +
  geom_smooth(method = lm, se=FALSE) +
  geom_point()+
```

```
theme(
  panel.background = element_rect(fill='transparent'), #transparent panel bg
  plot.background = element_rect(fill='transparent', color=NA), #transparent plot bg
  panel.grid.major = element_blank(), #remove major gridlines
  panel.grid.minor = element_blank(), #remove minor gridlines
  legend.background = element_rect(fill='transparent'), #transparent legend bg
  legend.box.background = element_rect(fill='transparent') #transparent legend panel
)
```



Improve the graph by:

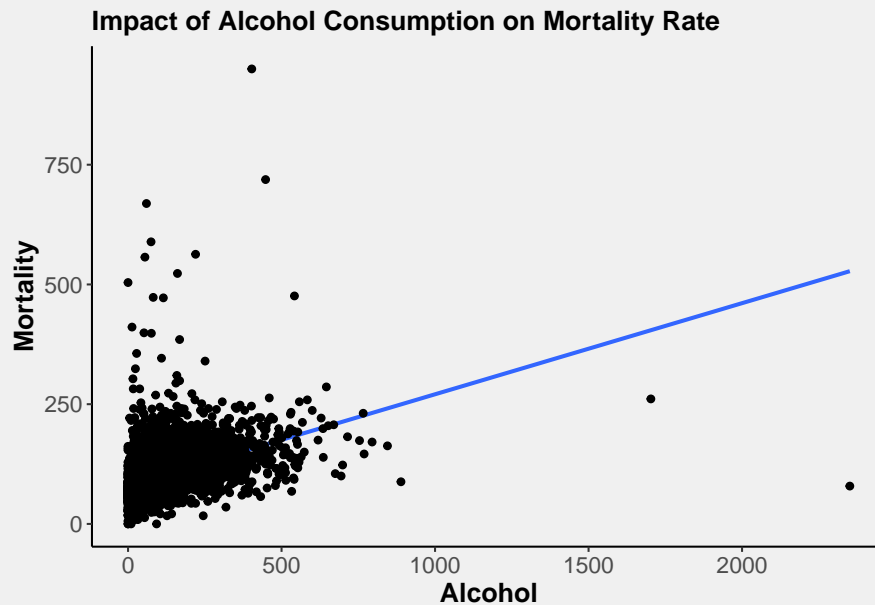
1. Adding a regression line.
2. Adding up a relevant title, also possibly a subtitle.
3. Adding axes labels and making them readable.
4. Remove the grid in the background.

```
ggplot(sim, aes(x = alcohol, y = mortality)) +
  geom_smooth(method = lm, se=FALSE) +
  geom_point() +
  theme_classic() +
  xlab('Alcohol') +
  ylab('Mortality') +
  ggtitle("Impact of Alcohol Consumption on Mortality Rate") +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14,face="bold")) +
  theme(plot.title = element_text(size = 14, face = "bold"))+
  theme(
    panel.background = element_rect(fill='transparent'), #transparent panel bg
    plot.background = element_rect(fill='transparent', color=NA), #transparent plot bg
    panel.grid.major = element_blank(), #remove major gridlines
    panel.grid.minor = element_blank(), #remove minor gridlines
  )
```

```

legend.background = element_rect(fill='transparent'), #transparent legend bg
legend.box.background = element_rect(fill='transparent') #transparent legend panel
)

```



If you have done everything correctly, you should see that the dots are rather concentrated in the bottom left corner of the scatterplot with some outliers far away from the cloud of our data. It is not a big deal, but we might want to get rid of the outliers and this way improve our visualisation.

There are several ways to do that, of course. But let's say we want to transform our variables, excluding all values over a certain point. For instance, we want to exclude the values above 750 of our alcohol variable and above 500 for our mortality variable. How would you do that?

```

# Option 1 with tidyverse
simred <- filter(sim, alcohol < 750 & mortality < 500)

# Option 2 with base R
simred <- subset(sim, alcohol < 750 & mortality < 500)

```

Hint: *there are several solutions. One could be, creating a new dataset subsetting the original. Another solution could be again, creating a new variable telling R to transform all the values above our treshold in NA (missing values). Try to find an apply the appropriate code. Use Google if necessary, it helps a lot.*

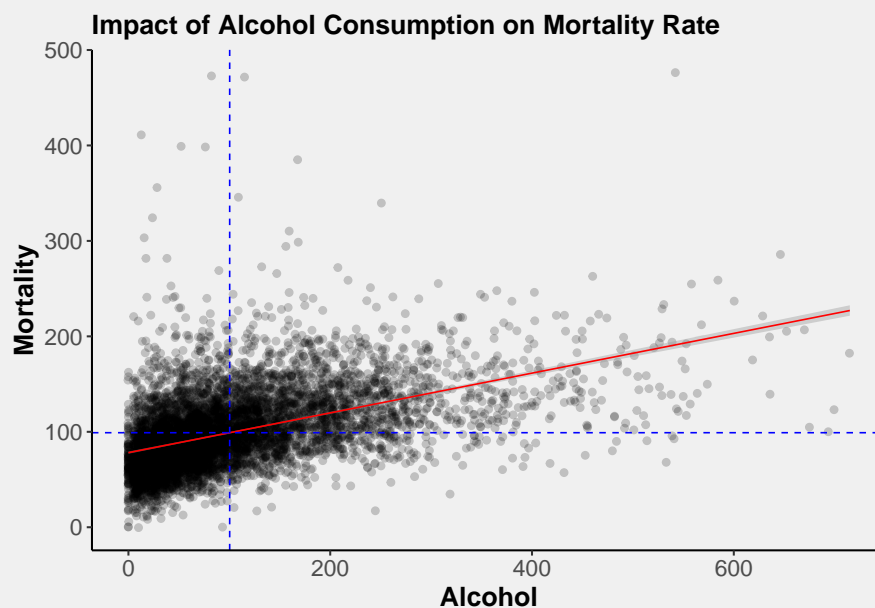
5 | Visualisation 2.0

Now, visualise the relationship using the new variables. What can you see?

You can improve the scatterplot using a series of arguments (e.g., `alpha`) in the `geom_point()` function in `ggplot`. Try to improve the Aesthetics of the scatterplot playing with `alpha`, for instance. (see: https://www.rdocumentation.org/packages/ggplot2/versions/3.4.0/topics/geom_point).

Also, you can draw a vertical and horizontal line corresponding to the mean of your variables using `geom_hline` and `geom_vline`. You can thus check if the regression line passes through the mean of X and Y. (see: https://www.rdocumentation.org/packages/ggplot2/versions/0.9.1/topics/geom_hline).

```
ggplot(simred, aes(alc, mortality)) +
  geom_point(position='jitter', alpha = 1/5) +
  xlab('Alcohol') +
  ylab('Mortality') +
  ggtitle("Impact of Alcohol Consumption on Mortality Rate") +
  theme_classic() +
  geom_smooth(method = 'lm', se=T, colour = 'red', lwd=0.4) +
  geom_hline(yintercept = mean(simred$mortality, na.rm=TRUE), color='blue', lty='dashed',
    lwd=0.4) +
  geom_vline(xintercept = mean(simred$alc, na.rm=TRUE), color='blue', lty='dashed', lwd=
    =0.4) +
  theme(axis.text=element_text(size=12),
    axis.title=element_text(size=14, face="bold")) +
  theme(plot.title = element_text(size = 14, face = "bold")) +
  theme(
    panel.background = element_rect(fill='transparent'), #transparent panel bg
    plot.background = element_rect(fill='transparent', color=NA), #transparent plot bg
    panel.grid.major = element_blank(), #remove major gridlines
    panel.grid.minor = element_blank(), #remove minor gridlines
    legend.background = element_rect(fill='transparent'), #transparent legend bg
    legend.box.background = element_rect(fill='transparent') #transparent legend panel
  )
```



6 | Saving the Scatterplot

You can also save a graph as .png, .JPG (even .pdf) that you can then import in a word document. Although there are many way to use your R output, saving a graph might be sometimes useful.

Use the function `ggsave()` to save your scatterplot. Again, there are tons of examples online, google it.

Hint: You first need to store the graph in an object.

Hint 2: The file will end up in your working directory

```
# First, create an object #
scatterplot<-ggplot(simred, aes(alcchol, mortality)) +
  geom_point(position='jitter', alpha = 1/5) +
  xlab('Alcohol') +
  ylab('Mortality') +
  ggtitle("Impact of Alcohol Consumption on Mortality Rate") +
  theme_classic() +
  geom_smooth(method = 'lm', se=T, colour = 'red', lwd=0.4)+
  geom_hline(yintercept = mean(simred$mortality, na.rm=TRUE), color='blue', lty='dashed',
    lwd=0.4)+
  geom_vline(xintercept = mean(simred$alcchol, na.rm=TRUE), color='blue', lty='dashed', lwd
    =0.4)+
  theme(axis.text=element_text(size=12),
    axis.title=element_text(size=14,face="bold")) +
  theme(plot.title = element_text(size = 14, face = "bold")) +
  theme(
    panel.background = element_rect(fill='transparent'), #transparent panel bg
    plot.background = element_rect(fill='transparent', color=NA), #transparent plot bg
    panel.grid.major = element_blank(), #remove major gridlines
    panel.grid.minor = element_blank(), #remove minor gridlines
    legend.background = element_rect(fill='transparent'), #transparent legend bg
    legend.box.background = element_rect(fill='transparent') #transparent legend panel
  )

# Then save the file #
save_plot("scatterplot_mortality.png", scatterplot)
```

7 | Regression Analysis (yes, finally)

Now we can finally run a linear regression with mortality as the outcome variable and alcohol as the predictor using the `lm()` function. Store the results in an object called `model` and visualise the regression output using `summary()`. Use the reduced data set without outliers.

```
# Store the results in an object called model #
model<-lm(mortality ~ alcchol, simred)
# Visualise the regression output using summary() #
```



```
summary(model)
##
## Call:
## lm(formula = mortality ~ alcohol, data = simred)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -122.69  -22.52   -4.71   17.27  377.77
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  78.155160   0.619648  126.13  <2e-16 ***
## alcohol      0.208270   0.004476   46.53  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.61 on 6956 degrees of freedom
## Multiple R-squared:  0.2374, Adjusted R-squared:  0.2373
## F-statistic: 2165 on 1 and 6956 DF, p-value: < 2.2e-16
```

You can also extract specific blocks of the output table. One way of doing it is to use the brackets `[]` after the `summary()` function. For example `summary()[8]`. Try to extract the block of Coefficients from the table, like this:

```
summary(model)[4]
## $coefficients
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  78.1551599  0.619648419  126.12823      0
## alcohol      0.2082697  0.004476011   46.53021      0
```

8 | Interpretation

Interpret the results, starting with model evaluation.

1. Is the p-value of the F-statistics statistically significant? We will be discussing this in our following lectures.
2. How much variation in the outcome variable does the model explain? What does this tell us about the model?
3. What's the value of the slope? What does it mean?
4. What's the value of the regression coefficient? How do we interpret it? Is it statistically significant? What does it mean in practice?
5. Interpret the results (in plain language) referring to the hypothesis you formulated above.

9 | Exporting the Results

As for the graphs, you can also export and save the results of the regression model in a Word table. To do that you can use the `stargazer` package. Try and export the table.

Hint 1: The function needs to contain, in order: the name of the R object where you stored the regression results, the option `header=F` to suppress the annoying immortalisation of the author, the option `type="html"`, and the option `out="documentname.doc"` which places a word document with that file name in your working directory.

Hint 2: Remember that Google is your best friend when you learn to code. And after that as well. No one actually remembers all the functions and all their arguments. The secret is to know how to google the things you need.

```
stargazer(model,
           header=F,
           type="html",
           out="documentname.doc")
```

Table 1

<i>Dependent variable:</i>	
	mortality
alcohol	0.208*** (0.004)
Constant	78.155*** (0.620)
Observations	6,958
R ²	0.237
Adjusted R ²	0.237
Residual Std. Error	35.610 (df = 6956)
F Statistic	2,165.060*** (df = 1; 6956)
Note:	*p<0.1; **p<0.05; ***p<0.01

You can improve the table in many ways. For example, you need to replace the variable names with the variable labels, You could add a name of the model and suppress unneeded statistics (see https://www.rdocumentation.org/packages/stargazer/versions/5.2.3/topics/stargazer_stat_code_list).

```
stargazer(model,
           header=F,
           font.size = "scriptsize",
           omit.stat = c("adj.rsq", "ser", "f"),
           dep.var.labels = "Mortality",
           covariate.labels = "Alcohol",
           type="html",
```

Table 2

<i>Dependent variable:</i>	
Mortality	
Alcohol	0.208*** (0.004)
Constant	78.155*** (0.620)
Observations	6,958
R ²	0.237
<i>Note:</i> *p<0.1; **p<0.05; ***p<0.01	

10 | Comparing models

You can now run another regression model with a different independent variable. Can you compare your original model with the new one? How? How do you know which independent variable is doing a better job in explaining your dependent variable?