cult_data_portfolio_2

Christian Stenbro

2025-01-04

Portfolio Assignment 2 – Introduction to Cultural Data Science

Part 1.

Load the 'divorce_margarine' dataset from the 'dslabs' package. Investigate the correlation between margarine consumption and divorce rates in Maine. Would an increase in the preference for margarine lead to skyrocketing divorce rates?

1.1 Installing packages and loading data

```
install.packages("dslabs")
library(dslabs)
```

```
# loading data
divorce_margarine <- dslabs::divorce_margarine
dim(divorce_margarine)
```

```
## [1] 10  3
```

```
head(divorce_margarine)
```

```
##   divorce_rate_maine margarine_consumption_per_capita year
## 1                5.0                               8.2 2000
## 2                4.7                               7.0 2001
## 3                4.6                               6.5 2002
## 4                4.4                               5.3 2003
## 5                4.3                               5.2 2004
## 6                4.1                               4.0 2005
```

1.2 Investigating a possible correlation

To test whether the margarine consumption per capita correlates with the divorce_rate, we could use the cor.test function:

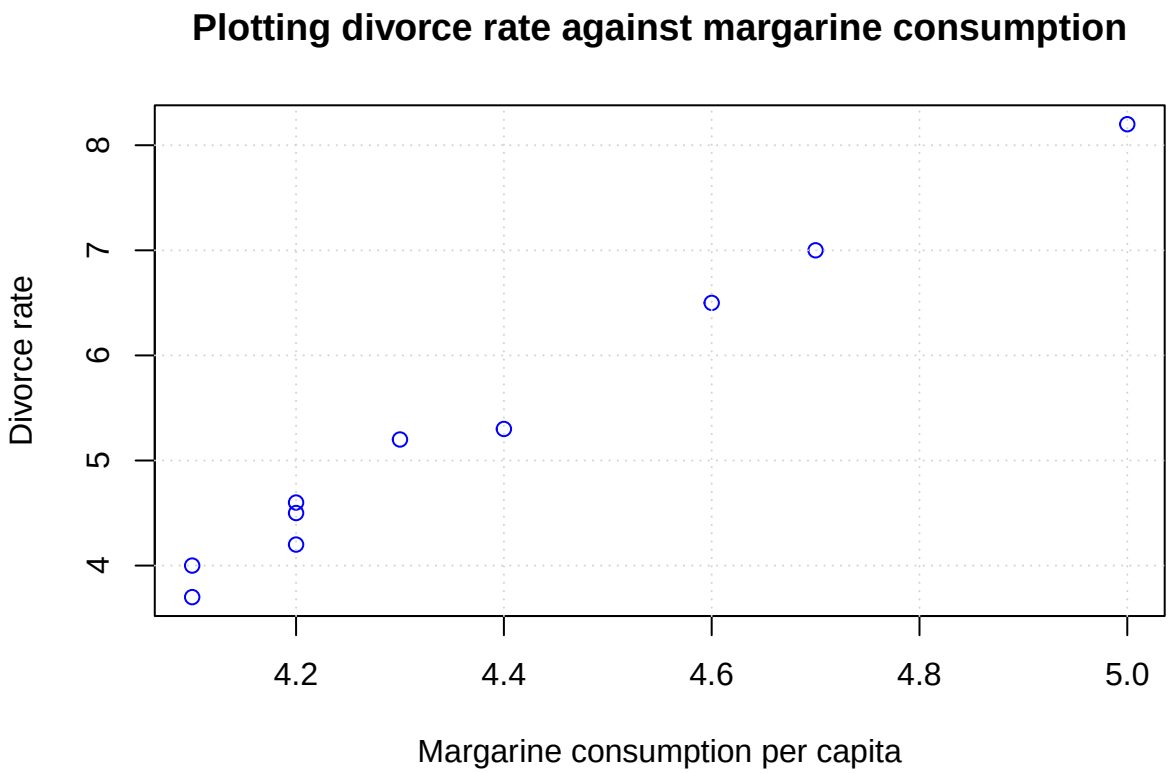
```
x = divorce_margarine$divorce_rate_maine
y = divorce_margarine$margarine_consumption_per_capita

cor.test(x = x,
         y = y)
```

```
##
##  Pearson's product-moment correlation
##
## data:  x and y
## t = 23.055, df = 8, p-value = 1.33e-08
```

```
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9676666 0.9983038
## sample estimates:
##       cor
## 0.9925585
```

I will also plot the variables:

```
plot(x, y,
     xlab = "Margarine consumption per capita",
     ylab = "Divorce rate",
     main = "Plotting divorce rate against margarine consumption",
     col = "blue")
grid()
```



The correlation test suggests a strong correlation, with a low p-value; this is also visible in the plot, which displays a clear positive relationship between divorce rate and margarine consumption.

Let's fit a regression model to learn more about the relationship between the two variables. I will standardize the predictor to ease interpretations of the variables (using the standardize function from the rethinking package):

```
install.packages("rethinking")
library(rethinking)
```

```
mdl_1 <- glm(divorce_rate_maine ~ standardize(margarine_consumption_per_capita),
    family = gaussian,
    data = divorce_margarine)

summary(mdl_1)
```

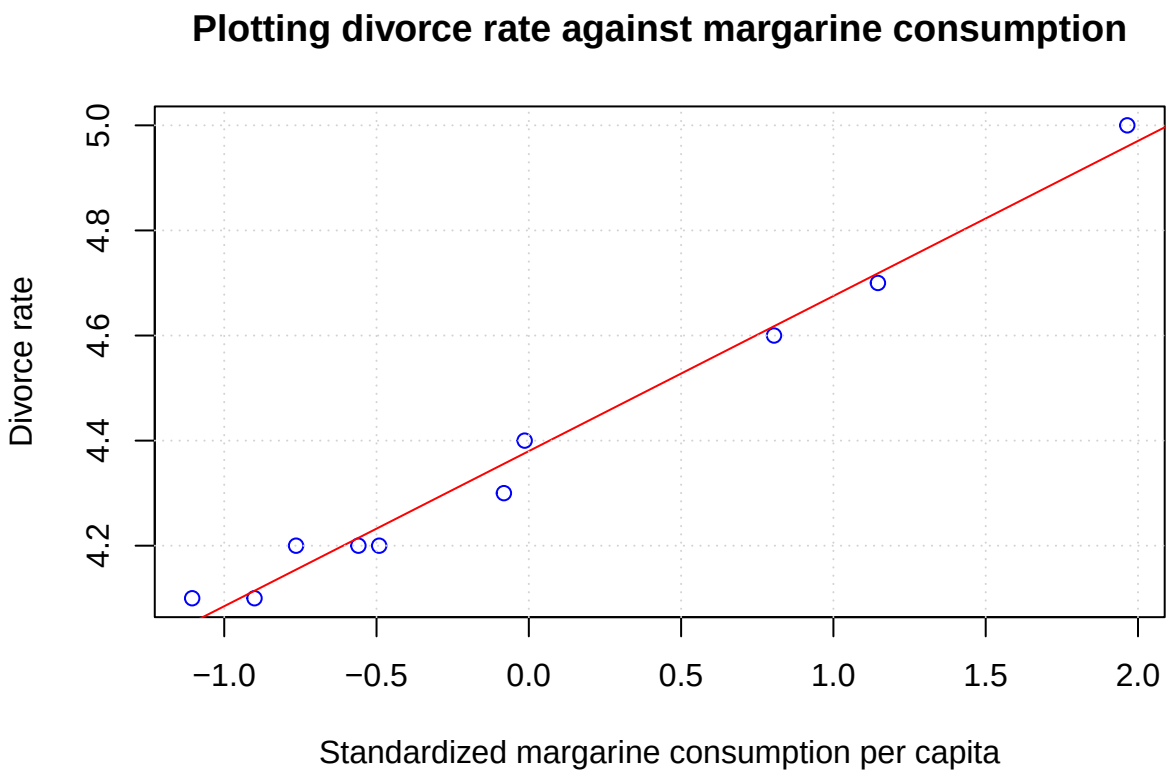
```
##
## Call:
## glm(formula = divorce_rate_maine ~ standardize(margarine_consumption_per_capita),
##     family = gaussian, data = divorce_margarine)
##
## Coefficients:
##                                             Estimate Std. Error t value
## (Intercept)                                  4.38000    0.01215  360.60
## standardize(margarine_consumption_per_capita)  0.29518    0.01280   23.05
##                                             Pr(>|t|)
## (Intercept)                                  < 2e-16 ***
## standardize(margarine_consumption_per_capita) 1.33e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.001475357)
##
##     Null deviance: 0.796000  on 9  degrees of freedom
## Residual deviance: 0.011803  on 8  degrees of freedom
## AIC: -33.041
##
## Number of Fisher Scoring iterations: 2
```

The model summary provides the following information:

- There is a positive relationship between margarine consumption variable and the divorce rate variable: when margarine consumption increases with 1 standard deviation, the divorce rate increases with approximately 0.3 units.
- The models predicts that divorce rate is at 4.38 at the mean level of margarine consumption (this is represented in the intercept).
- Both estimates (the margarine consumption coefficient and the intercept) are fairly certain, as indicated by the small std. error values.

We can add the regression line to the plot (here I have not attempted to visualize the uncertainty of the regression line):

```
plot(x = standardize(divorce_margarine$margarine_consumption_per_capita), y = divorce_margarine$divorce_
     xlab = "Standardized margarine consumption per capita",
     ylab = "Divorce rate",
     main = "Plotting divorce rate against margarine consumption",
     col = "blue")
abline(a = mdl_1$coefficients[[1]], b = mdl_1$coefficients[[2]], col = "red")
grid()
```



1.3 Would divorce rates skyrocket?

Even if the linear model suggests a positive relationship between the variables, we would need further evidence to confidently say that divorce rates would skyrocket if the margarine consumption increased. First of all, we lack data for more extreme values of margarine consumption; it could be that the association between the variables changes when the margarine consumption becomes more extreme.

Second, it is difficult to know whether the relationship is indeed suggestive of a causal connection between the variables. There is no reason to think that margarine consumption would directly lead to the deterioration of marriages, so there could be other 'hidden' variables correlating with margarine consumption which would be more obvious candidates (to be fair, I faintly remember this to be the case, as the same data set is used as an example in McElreath (2020); and critical thinking leads to the same conclusion, even without knowing the hidden variable).

Part 2.

2.1 Loading data

Load the 'GSSvocab' dataset from the 'car' package. This dataset contains people's scores on an English vocabulary test and includes demographic information.

```
# installing the car package
install.packages("carData")
library(carData)
```

```
# loading data
GSSvocab <- carData::GSSvocab

# assessing the data
dim(GSSvocab)
```

```
## [1] 28867     8
```

```
head(GSSvocab)
```

```
##        year gender nativeBorn ageGroup educGroup vocab age educ
## 1978.1 1978 female        yes    50-59    12 yrs    10  52   12
## 1978.2 1978 female        yes      60+   <12 yrs     6  74    9
## 1978.3 1978   male        yes    30-39   <12 yrs     4  35   10
## 1978.4 1978 female        yes    50-59    12 yrs     9  50   12
## 1978.5 1978 female        yes    40-49    12 yrs     6  41   12
## 1978.6 1978   male        yes    18-29    12 yrs     6  19   12
```

This is a large data set!

2.2 Filtering

Filter for the year 1978 and remove rows with missing values (the function na.exclude() is one way to do this – check out the documentation!).

```
install.packages("tidyverse")
library(tidyverse)
```

```
# making a new subset with rows matching 1978 and removing NAs
GSSvocab_subset <- GSSvocab %>% filter(year == 1978) %>% na.exclude(.)

# testing that all NAs are removed successfully
unique(is.na(GSSvocab_subset))
```

```
##         year gender nativeBorn ageGroup educGroup vocab   age  educ
## 1978.1 FALSE  FALSE      FALSE    FALSE     FALSE FALSE FALSE FALSE
```

2.3 Model a: Vocab. test score ~ education level

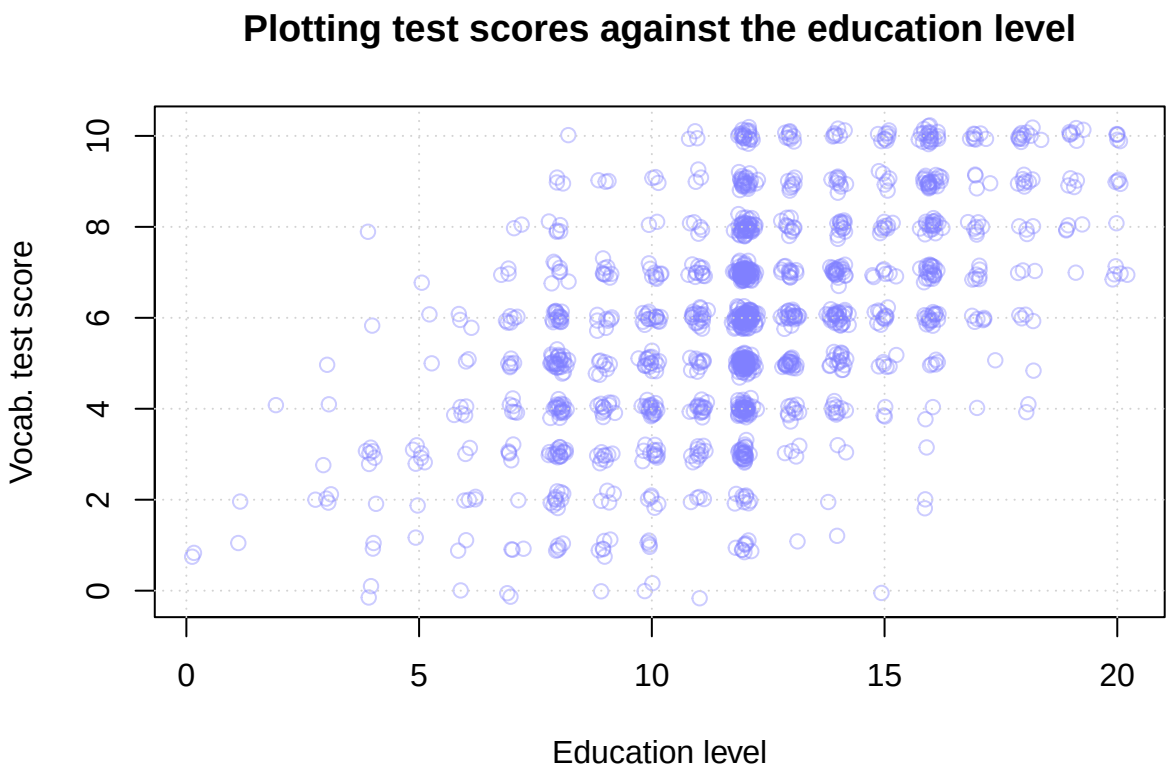
Is a person's score on the vocabulary test ('vocab') significantly impacted by their level of education ('educ')? Visualize the relationship in a plot and build a model. Briefly explain the results.

I will make the following model:

vocab ~ educ

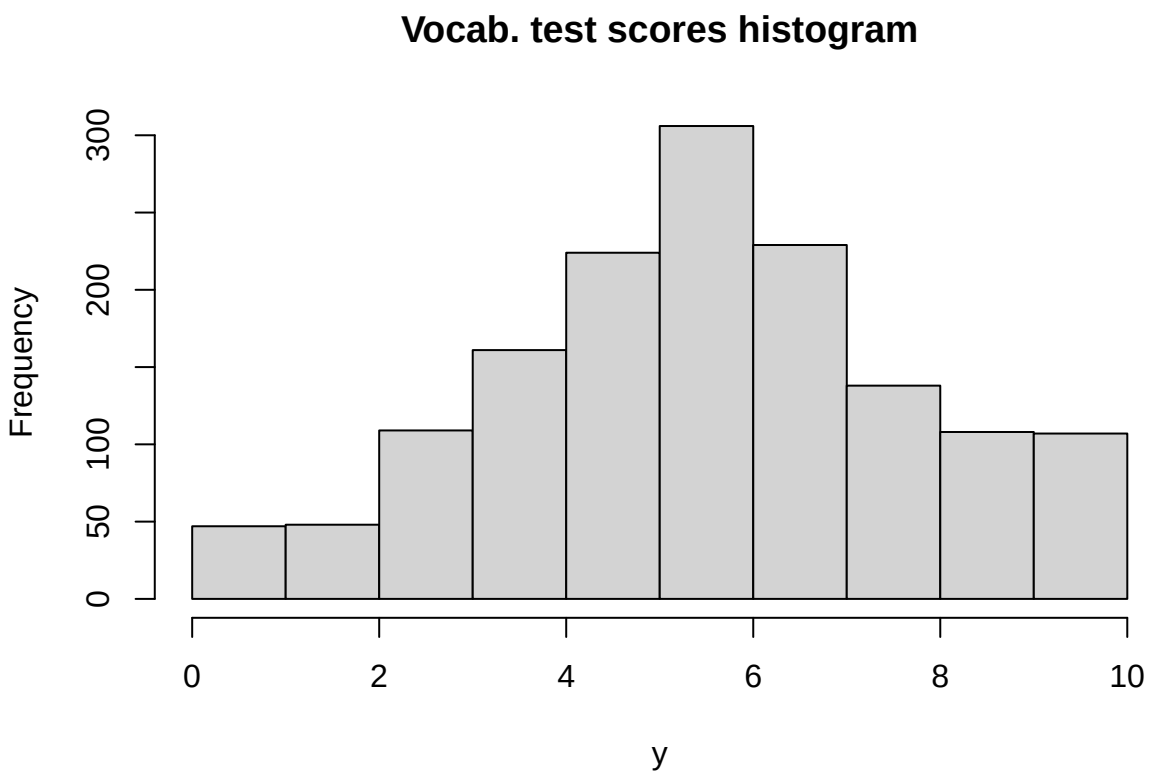
First, I plot the variables against each other:

```
# Plotting the variables (with a bit of jitter added to show all data points)
x <- GSSvocab_subset$educ
y <- GSSvocab_subset$vocab

jitt_x <- rnorm(length(x), 0, 0.1)
jitt_y <- rnorm(length(y), 0, 0.1)

plot(x = x + jitt_x, y = y + jitt_y,
     xlab = "Education level",
     ylab = "Vocab. test score",
     main = "Plotting test scores against the education level",
     col = col.alpha(rangi2, 0.4))
grid()
```



Then, I visualise the outcome variable indivudally:

```
hist(y, main = "Vocab. test scores histogram")
```



Comment: The distribution looks normal, but both variables are technically zero-bounded and categorical (this fact is also visible in the scatter plot). For the model, I will treat it as a continuous variable.

```r
# fitting a linear model. The educ predictor is standardized
mdl_a <- glm(vocab ~ standardize(educ), family = gaussian, data = GSSvocab_subset)

summary(mdl_a)
```

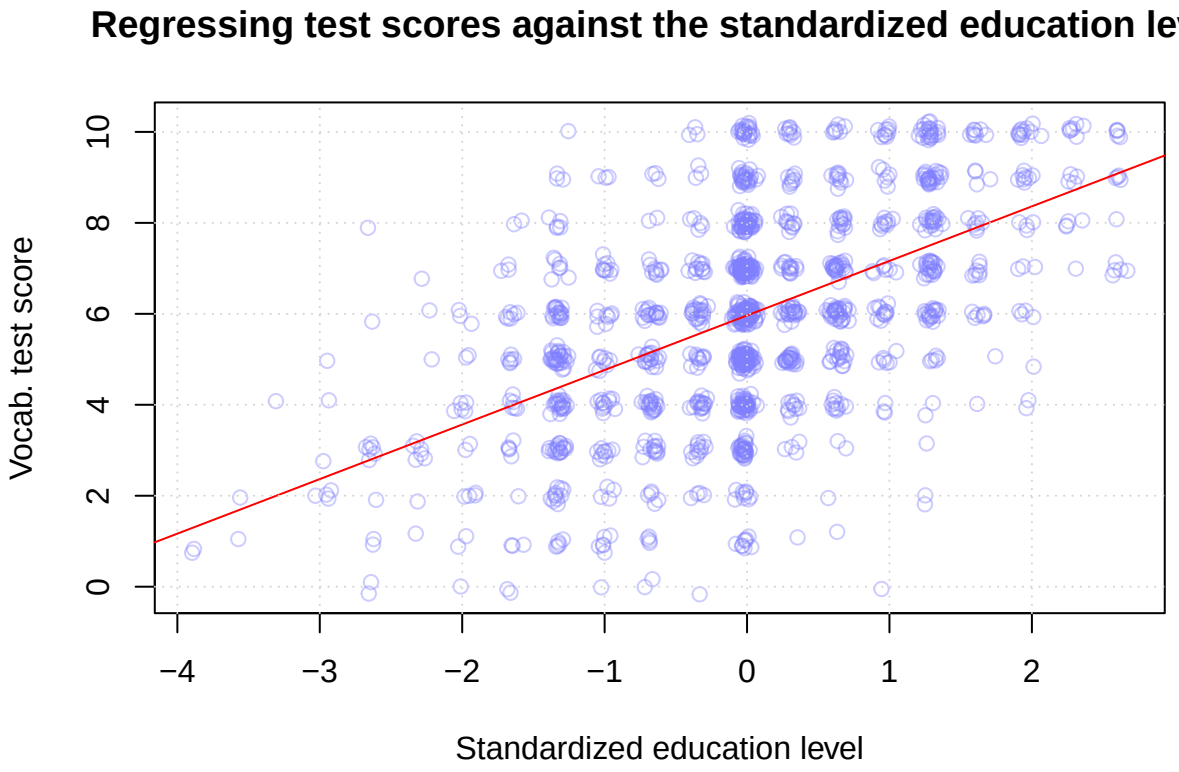
```
##
## Call:
## glm(formula = vocab ~ standardize(educ), family = gaussian, data = GSSvocab_subset)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        5.96412    0.04905  121.59   <2e-16 ***
## standardize(educ)  1.19939    0.04907   24.44   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 3.553782)
##
##     Null deviance: 7365.1  on 1476  degrees of freedom
## Residual deviance: 5241.8  on 1475  degrees of freedom
## AIC: 6068.4
##
## Number of Fisher Scoring iterations: 2
```

Based on the summary, we can say the following about the relationship:

- The model's expected test score for a person with the average education level is approximately 5.96 (represented in the intercept estimate).
- As the education level increases with 1 standard deviation, the vocabulary test score is expected to increase with approximately 1.99 points (represented in the standardized educ coefficient estimate).
- Both estimates are fairly certain (small std. errors).
- The summary also tells us that the sigma of the model is estimated at approximately 3.55. In the scale of the outcome variable (test scores), this is a quite high number. In other words, the model is confidently predicting a positive relationship between the variables, but there is a lot of noise in the predictions. Intuitively, this makes sense considering the wide spread of test scores for each education level.

We can now visualise the relationship by drawing the regression line:

```r
# Plotting the variables
plot(x = standardize(GSSvocab_subset$educ + jitt_x), y = GSSvocab_subset$vocab + jitt_y,
     xlab = "Standardized education level",
     ylab = "Vocab. test score",
     main = "Regressing test scores against the standardized education level",
     col = col.alpha(rangi2, 0.4))
abline(a = mdl_a$coefficients[[1]], b = mdl_a$coefficients[[2]], col = "red")
grid()
```



2.4 Adding 'nativeBorn' as a predictor

Whether a person is the native of an English-speaking country ('nativeBorn') could potentially have an impact on the size of their vocabulary. Visualize the relationship and add the predictor to the model. Briefly explain the results.

Since the nativeBorn is a binary variable:

```
class(GSSvocab_subset$nativeBorn)
```

```
## [1] "factor"
```

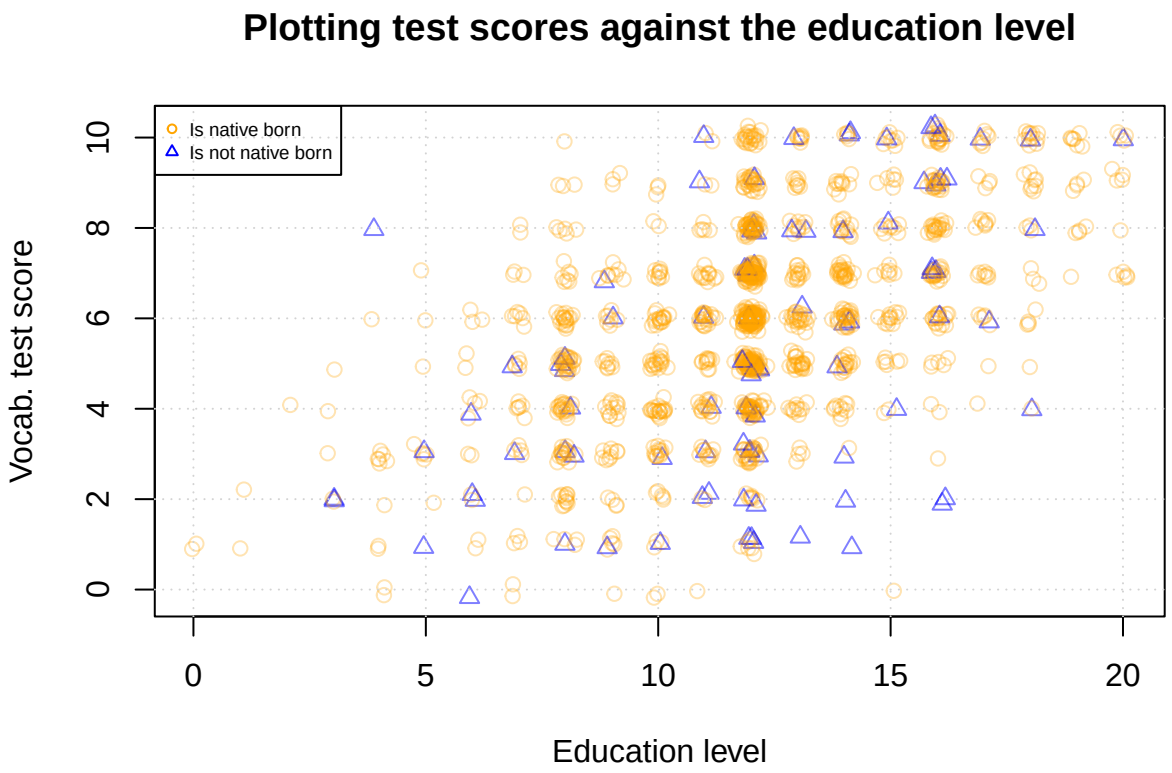
```
levels(GSSvocab_subset$nativeBorn)
```

```
## [1] "no"  "yes"
```

. . . the information could be added to the previous plot by alternating the color/symbol according to the value of the nativeBorn variable:

```
# making color + symbol coding vectors
colors <- ifelse(GSSvocab_subset$nativeBorn == "yes", col.alpha("orange", 0.3), col.alpha("blue", 0.5))
symbols <- ifelse(GSSvocab_subset$nativeBorn == "yes", 1, 2)

# making a new plot, this time with the native born variable
x <- GSSvocab_subset$educ
y <- GSSvocab_subset$vocab

jitt_x <- rnorm(length(x), 0, 0.1)
jitt_y <- rnorm(length(y), 0, 0.1)

plot(x = x + jitt_x, y = y + jitt_y,
     xlab = "Education level",
     ylab = "Vocab. test score",
```

```
     main = "Plotting test scores against the education level",
     col = colors,
     pch = symbols)
grid()
legend("topleft",
       legend = c("Is native born", "Is not native born"),
       col = c("orange", "blue"),
       pch = c(1,2),
       cex = 0.6)
```



Now, the predictor is added to the model:

```
# fitting model
mdl_a2 <- glm(vocab ~ standardize(educ) + nativeBorn,
              family = gaussian,
              data = GSSvocab_subset)

summary(mdl_a2)
```

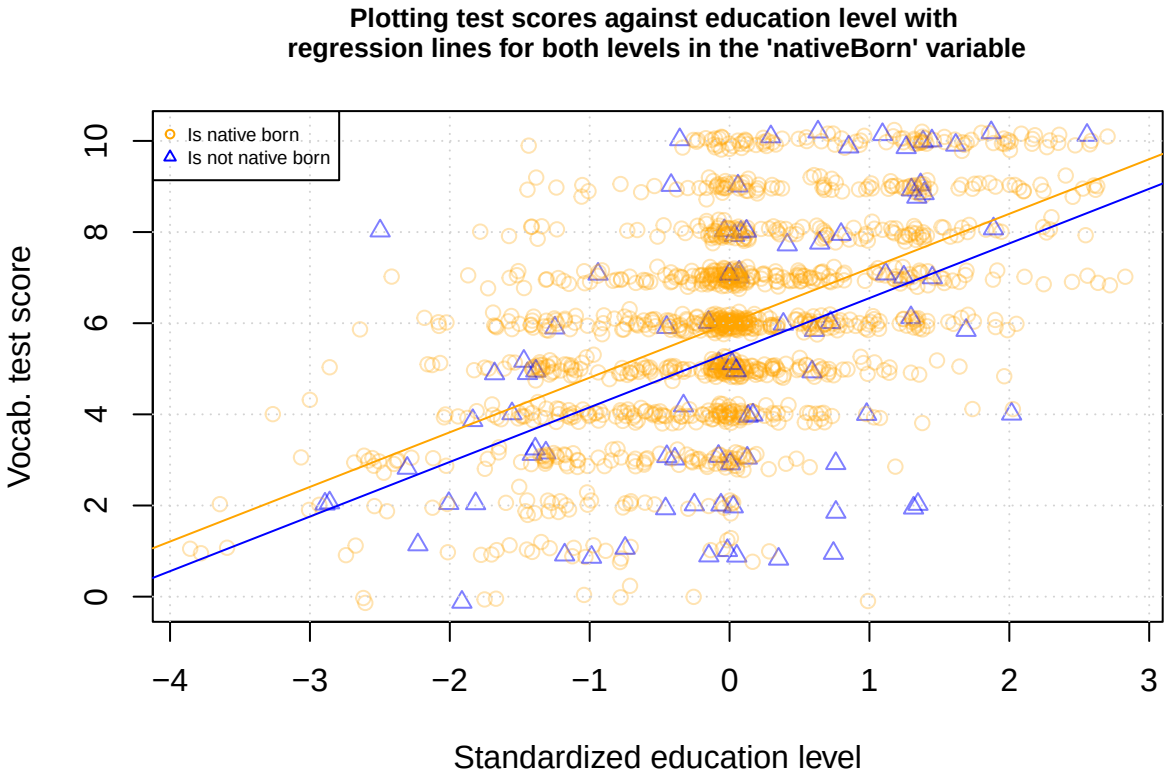
```
##
## Call:
## glm(formula = vocab ~ standardize(educ) + nativeBorn, family = gaussian,
##     data = GSSvocab_subset)
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         5.35298    0.19922  26.870  < 2e-16 ***
## standardize(educ)   1.19850    0.04892  24.499  < 2e-16 ***
## nativeBornyes       0.65032    0.20551   3.164  0.00159 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 3.532197)
##
##     Null deviance: 7365.1  on 1476  degrees of freedom
## Residual deviance: 5206.5  on 1474  degrees of freedom
## AIC: 6060.4
##
## Number of Fisher Scoring iterations: 2
```

While the new predictor has not changed the estimate for the effect of the standardized education level, the intercept is now slightly lower. This makes sense, because adding the binary 'nativeBorn' variable can be conceptualised as having a model with two regression lines, with a similar slope, but with different intersects. In this case, the model predicts higher test scores for people that are native english speakers, although it is worth noting that the estimate is not as certain as is the case for the education level.

We can now visualise the two regression lines for both levels of the binary variable:

```
# making color + symbol coding vectors
colors <- ifelse(GSSvocab_subset$nativeBorn == "yes", col.alpha("orange", 0.3), col.alpha("blue", 0.5))
symbols <- ifelse(GSSvocab_subset$nativeBorn == "yes", 1, 2)

# plotting variables and regression lines
x <- standardize(GSSvocab_subset$educ)
y <- GSSvocab_subset$vocab

jitt_x <- rnorm(length(x), 0, 0.1)
jitt_y <- rnorm(length(y), 0, 0.1)

plot(x = x + jitt_x, y = y + jitt_y,
     xlab = "Standardized education level",
     ylab = "Vocab. test score",
     main = "Plotting test scores against education level with \nregression lines for both levels in the
     col = colors,
     pch = symbols)
grid()
legend("topleft",
       legend = c("Is native born", "Is not native born"),
       col = c("orange", "blue"),
       pch = c(1,2),
       cex = 0.6)
abline(a = mdl_a2$coefficients[[1]] + mdl_a2$coefficients[[3]], b = mdl_a2$coefficients[[2]], col = "ora
abline(a = mdl_a2$coefficients[[1]], b = mdl_a2$coefficients[[2]], col = "blue")
```

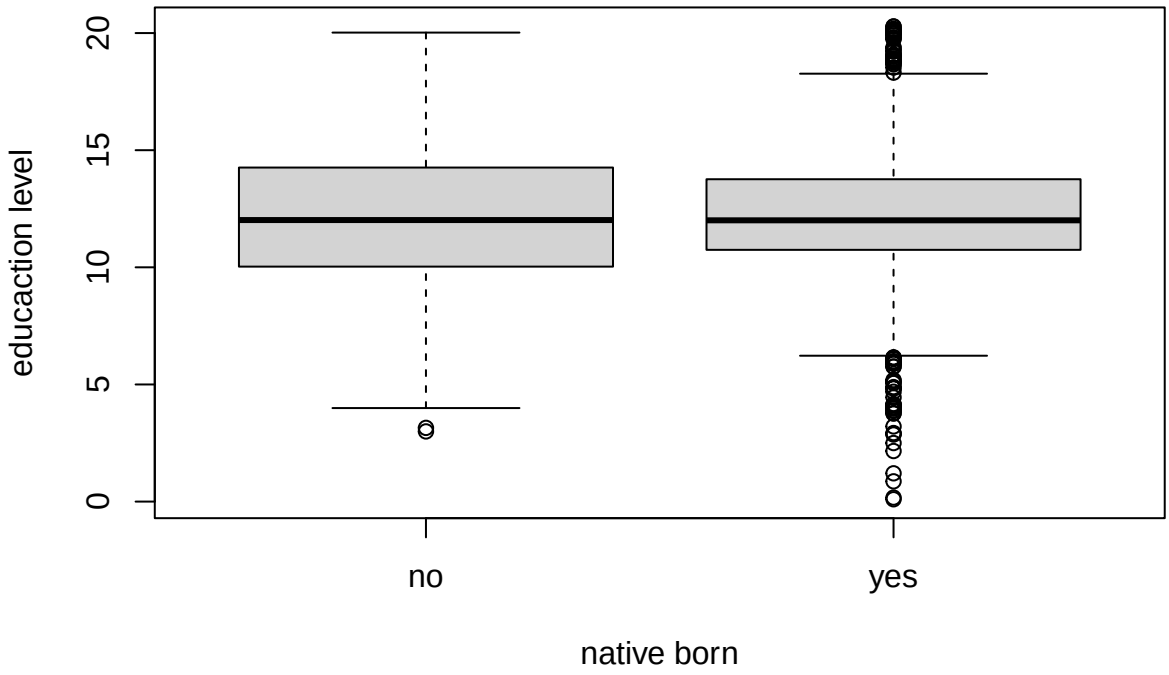


2.5 Model b: education level ~ education level * nativeBorn

Does a person's level of education depend on whether they are a native of the country? Visualize the relationship. Do you think it makes sense to add the relationship as an interaction term? Try creating the model and briefly explain the results.

Before modelling, the variables are plotted against each other:

```
x = GSSvocab_subset$nativeBorn
y = GSSvocab_subset$educ

jitt_y = rnorm(length(y), 0, 0.2)

plot(x = x, y = y + jitt_y,
     xlab = "native born", ylab = "educaction level",
     main = "Education levels plotted against the levels of the 'nativeBorn' variable")
```



From looking at the plot, we can tell that the median is almost the same for both groups. Yet, the variance of the native born group is larger.

I think it does make sense to add the predictor as an interaction term, considering that we are really interested in whether the effect of education level differs between the two groups. Conceptually, this means that we are creating a model with different slopes depending on the value of the 'nativeBorn' variable:

```
# fitting interaction model
mdl_b <- glm(vocab ~ standardize(educ) * nativeBorn,
             family = gaussian,
             data = GSSvocab_subset)

summary(mdl_b)
```

```
##
## Call:
## glm(formula = vocab ~ standardize(educ) * nativeBorn, family = gaussian,
##     data = GSSvocab_subset)
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      5.35456    0.19931  26.866  < 2e-16 ***
## standardize(educ)                1.26843    0.16794   7.553 7.45e-14 ***
## nativeBornyes                    0.64875    0.20560   3.155  0.00163 **
## standardize(educ):nativeBornyes -0.07641    0.17556  -0.435  0.66344
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 3.53414)
##
##     Null deviance: 7365.1  on 1476  degrees of freedom
```

```
## Residual deviance: 5205.8  on 1473  degrees of freedom
## AIC: 6062.2
##
## Number of Fisher Scoring iterations: 2
```

There is only a very small effect of the new interaction between the education level and the 'nativeBorn' variable, suggesting that the education level matters less for the test scores of native born english speakers. The std. error is very large for this estimate (larger than the effect size!) which means that the estimate is very uncertain (this is also reflected in the insignificant p value).

2.6 Comparing the models

Which model performs best?

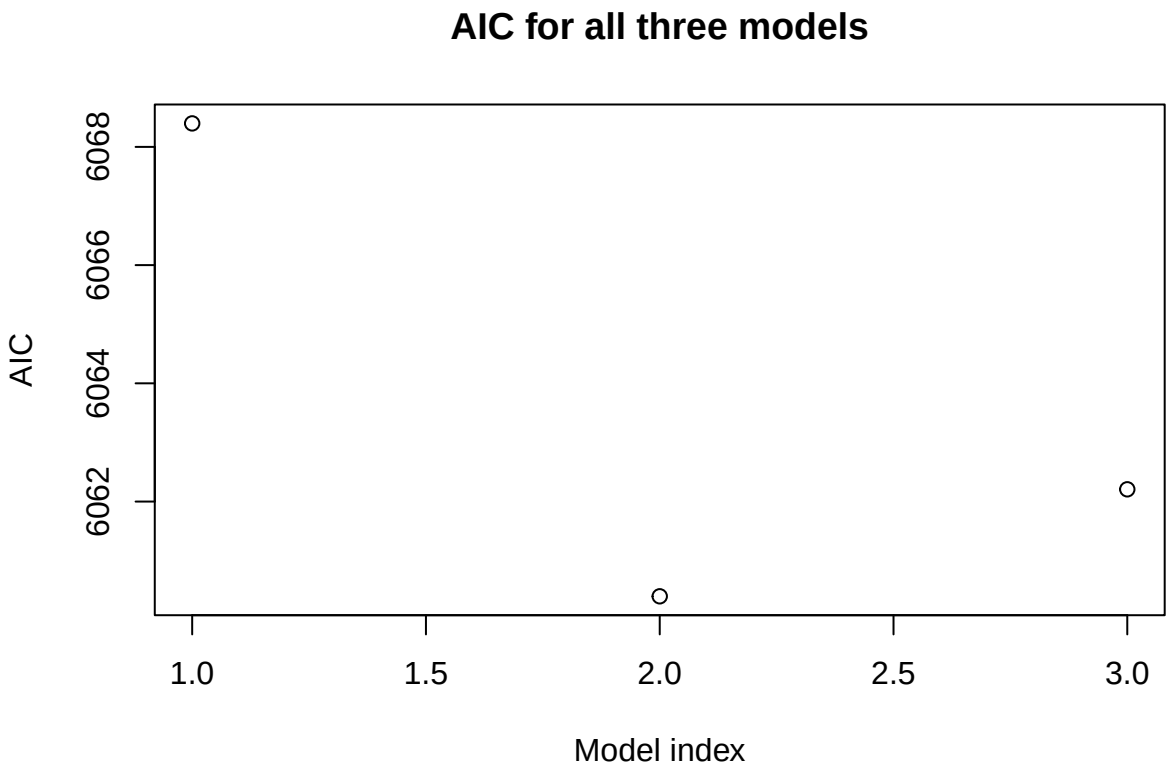
From a 'simple is better' perspective, model a2 (vocab ~ educ + nativeBorn) would seem a better choice than model b, since the interaction doesn't add any strong explanatory power to our model, yet complicates the model by adding another coefficient.

Approaching the question more quantitatively, we can compare the AIC scores of the respective models fitted for this part of the assignment:

```
cat("mdl_a =", mdl_a$aic,
    "\nmdl_a2 =", mdl_a2$aic,
    "\nmdl_b =", mdl_b$aic)
```

```
## mdl_a = 6068.397
## mdl_a2 = 6060.397
## mdl_b = 6062.207
```

```
plot(c(mdl_a$aic, mdl_a2$aic, mdl_b$aic),
     ylab = "AIC", xlab = "Model index",
     main = "AIC for all three models")
```



This method confirms that model a2 is the stronger candidate, as it has the lowest AIC score of the three. As AIC penalizes the fit according to the number of predictors (Field et al., 2012), it makes sense that mdl_b would have a higher score.

References

Field, A., Miles, J., & Field, Z. (2012). Discovering Statistics Using R (1st edition). SAGE Publications Ltd.

McElreath, R. (2020). Statistical Rethinking: A Bayesian Course with Examples in R and Stan (Second edition). CRC Press. https://doi.org/10.1201/9780429029608